Quantizer Design for Channel Codes with Soft Output Decoding

J. Bakus and A. K. Khandani

Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1

Technical Report UW-E&CE#2001-12

September 2, 2001

Abstract

A new method of combined source-channel coding for the scalar quantization of a discrete memoryless source is presented which takes advantage of the reliability information produced by a soft output channel decoder. Numerical results are presented for a memoryless Gaussian source in conjunction with Turbo-code showing up to 1 dB improvement in the end-to-end distortion with respect to a traditional channel optimized scalar quantizer. Furthermore, certain issues related to the effect of the channel mismatch and spectral efficiency of the system are studied. It is shown that the increase in distortion due to a channel mismatch can be substantially reduced by using an adaptive receiver.

1 Introduction

This work studies the transmission of a discrete time continuous amplitude signal over a noisy channel using a scalar quantizer in conjunction with a channel coding scheme which relies on a soft output channel decoder[1]. An improvement in the end-to-end quantization distortion is achieved by providing a soft reconstruction rule using the reliability information generated by the soft output channel decoder. The corresponding quantization and reconstruction rules are iteratively optimized using a procedure similar to the Lloyd-Max algorithm [3, 4]. Numerical results are presented for the case of a memoryless Gaussian source where Turbo-code [2] is used for forward error correction. Note that in the case of Turbo-code, the required reliability information is available as part of the normal decoding operation. Numerical results show an improvement over channel optimized scalar quantizer operating in conjunction with Turbo-code.

Optimum fixed-rate scalar quantizers, introduced by Max [3] and Lloyd [4], minimize the average distortion for a given number of threshold points. This approach was later extended by Linde, Buzo, and Gray to a vector quantizer [5]. The original Lloyd-Max algorithm does not consider the effect of channel noise. Kurtenbach and Wintz were among the first researchers who investigated the effect of the channel noise in a quantization system [6]. Later, Rydbeck and Sundberg showed that the code word assignment for a quantizer operating over a noisy channel also plays an important role in the system performance [7].

In [8], Farvardin and Vaishampayan presented an algorithm based on the Lloyd-Max for the quantizer design over a noisy channel resulting in the so called *Channel Optimized Scalar Quantizer* (COSQ). The COSQ algorithm was further extended by the same researchers to vector sources in [9, 10] which is known as the *Channel Optimized Vector Quantizer* (COVQ).

One of the first attempts to design a quantizer with soft reconstruction decoding was made by Vaishampayan and Farvardin by extending the COVQ design algorithm to include the modulation signal set [11]. Another approach to soft reconstruction was proposed by Phamdo and Alajaji, who applied the COVQ

[1] A soft output channel decoder can be efficiently implemented by applying the BCJR algorithm [1] to the trellis diagram of the code with iterations between decoders in the case of Turbo-code [2].

algorithm to the case that the demodulator output is uniformly quantized to allow for a soft decision decoding [12, 13]. This increases the number of channel output symbols and results in a finer reconstruction than the classical COVQ. This approach has successfully been used by Zhu and Alajaji [14] to design a COVQ for Turbo-code channel.

Skoglund and Hedelin in [15, 16] study the problem of transmitting a source via vector quantization over a noisy channel. They present a soft decoder which is optimal in the mean square sense and discuss rules to design the corresponding quantizer and reconstructor pair. They propose an efficient algorithm for optimum decoding, as well as a variety of sub-optimum methods of reduced complexity, where the key idea is to use a Hadamard formulation of the optimum decoder. This idea is further adapted for image transmission [17], multiuser decoding [18], and channels with memory [19].

The algorithms based on Lloyd Max quantizers are essentially gradient descent algorithms, which are not guaranteed to result in a global minimum. To avoid this problem, a number of techniques have been proposed that introduce randomness to help push the solution out of local minima. As the algorithm proceeds the amount of this randomness is reduced, resulting in a solution closer to the global minimum. Some of the techniques include deterministic annealing [20], and noisy channel relaxation [21]. An excellent overview of such probabilistic techniques can be found in [22].

Ho [23] uses the reliability information and the same soft reconstruction rule as in this work. The design algorithm is based on the iterative COVQ, and the effects of the channel noise are incorporated by transmitting the encoded training sequence through a Turbo-code channel at every iteration. Several other works have been devoted to using Turbo-code as part of a combined source channel coding system for image transmission [24, 25]. In these works, the image is source coded using the JPEG standard [26], and then channel coded using Turbo-code. An image model is used to estimate the probability of each pixel from adjacent pixels, and these probabilities are fed back to the Turbo-code decoder to improve the channel decoding.

The current work, and the above mentioned works, are in some way or other based on the following two basic principles:

1. Quantizer partitions are selected to minimize the distortion due to the individual input samples (refer to Eq. 10).

2. Quantizer output is selected as a proper combination of a set of reconstruction values. In [27, 28, 16, 23], as well as in the current work, this is selected as the sum of the reconstruction values multiplied by their probabilities (refer to Eq. 14).

The main contribution of this report is to formulate these two basic principles explicitly for the case of a scalar quantizer and a channel decoder producing analog (un-quantized) bit Log Likelihood Ratio

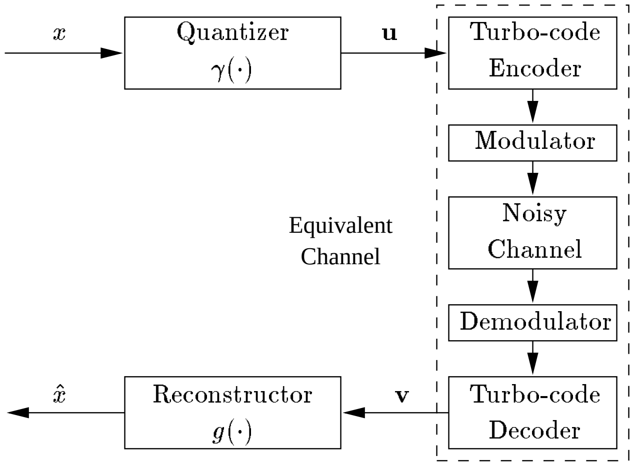


Figure 1: System block diagram.

(LLR) values. The main differences between the current report and our earlier related work [27, 28] are: (i) improvement in the reconstruction level design by solving a system of linear equations, and (ii) using closed form expressions to design the quantizers for the Gaussian distributed samples (instead of a training sequence).

This report is organized as follows. Section 2 outlines the overall system including the design procedure for the quantizer and reconstructor. This is followed by several models for the behavior of the channel coding structure. Finally, Section 3 presents some numerical results for a memoryless Gaussian source in conjunction with Turbo-code. This includes a study of certain issues related to the effect of the channel mismatch and spectral efficiency of the system.

2 System Overview

The block diagram of the system is shown in Fig. 1. The scalar quantizer $\gamma(\cdot)$ maps the input sample x to a quantizer partition $S \in \{S_1, \ldots, S_N\}$. Each quantizer partition is represented by a binary code word $\mathbf{u} \in \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$ composed of r bits, where $N \leq 2^r$. At the receiver side, for each transmitted code word $\mathbf{u}$, the soft output decoder produces a reliability information vector $\mathbf{v}$ composed of r Log Likelihood Ratio (LLR) values. The reconstructor $g(\cdot)$ maps the reliability information vector to an output sample $\hat{x}$.

The encoder, noisy channel, and the decoder can be considered as an equivalent channel with input $\mathbf{u}$ and output $\mathbf{v}$, as shown in Fig. 1. To design the quantizer and the reconstructor, the relationship between the input $\mathbf{u}$ and the output $\mathbf{v}$ of this equivalent channel are required, and different models to approximate this relationship will be presented.

We assume that the equivalent channel between $\mathbf{u}$ and $\mathbf{v}$ is memoryless, i.e.,

$$P(\mathbf{v}|\mathbf{u}) = \prod_{i=1}^{r} P(v_i|u_i), \tag{1}$$

where $u_i, v_i, i = 1, \ldots, r$ are the components of $\mathbf{u}$, $\mathbf{v}$, respectively. We also assume that the components of $\mathbf{u}$ are independent of each other, i.e.,

$$P(\mathbf{u}) = \prod_{i=1}^{r} P(u_i). \tag{2}$$

Note that Eqs. 1 and 2 are the basic assumptions behind the algorithms known for the soft output decoding of channel codes (e.g., BCJR algorithm [1]). Using Eqs. 1, 2, we obtain,

$$P(\mathbf{u}|\mathbf{v}) = \prod_{i=1}^{r} P(u_i|v_i). \tag{3}$$

In practice, the bit independence assumption in Eq. 3 is usually not entirely valid. However, the same assumptions are used in all the schemes that are compared with each other. The validity of the independence assumption can be improved by interleaving the quantizer output bits before encoding. Numerical results are presented which clarifies the improvement obtained by using such an extra stage of interleaving.

In a system with soft reconstruction, the probabilities $P(\mathbf{u}|\mathbf{v})$ are used to reconstruct the final output. Using the independence assumption in Eq. 3, these probabilities are computed by finding the related conditional probabilities at the bit level and extending the results to composite code words by multiplying the corresponding bit values together. To compute the conditional bit probabilities, consider a given bit u of $\mathbf{u}$ and the corresponding LLR value v in $\mathbf{v}$. We have,

$$v = \ln \frac{P(u = 1|\ \textit{channel output})}{P(u = 0|\ \textit{channel output})}. \tag{4}$$

Using the definition of LLR in Eq. 4, we obtain,

$$\begin{aligned} P(u = 0|\ \textit{channel output}) &= \frac{1}{1 + \exp(v)}, \\ P(u = 1|\ \textit{channel output}) &= \frac{\exp(v)}{1 + \exp(v)}. \end{aligned} \tag{5}$$

We assume,

$$\begin{aligned} P(u = 0|\ \textit{channel output}) &= P(u = 0|v), \\ P(u = 1|\ \textit{channel output}) &= P(u = 1|v), \end{aligned} \tag{6}$$

This assumption relies on a perfect decoder where all the information about u is captured in v.

To design the quantizer, consider the distortion of a sample x quantized to partition i as:

$$D_i(x) = \int_{\mathbf{v}} P(\mathbf{v}|\mathbf{u} = \mathbf{u}_i) \left[x - g(\mathbf{v})\right]^2 d\mathbf{v}, \tag{7}$$

where $P(\mathbf{v}|\mathbf{u} = \mathbf{u}_i) = P(\mathbf{v}|x \in S_i)$ is the probability of receiving a reliability vector $\mathbf{v}$ conditioned on transmitting $\mathbf{u}_i$ and $g(\cdot)$ is the reconstruction function. Eq. 7 can be written as,

$$D_i(x) = x^2 - 2xB_i + C_i, \tag{8}$$

where,

$$\begin{aligned} B_i &= \int_{\mathbf{v}} g(\mathbf{v}) P(\mathbf{v}|\mathbf{u} = \mathbf{u}_i) d\mathbf{v}, \\ C_i &= \int_{\mathbf{v}} g(\mathbf{v})^2 P(\mathbf{v}|\mathbf{u} = \mathbf{u}_i) d\mathbf{v}. \end{aligned} \tag{9}$$

The structure of the quantizer is captured in the quantities B_i and C_i given in Eq. 9.

The quantizer is designed by fixing the reconstruction function, and finding the best partitions $S_i, i = 1, \ldots, N$ where,

$$S_i = \{x : D_i(x) \leq D_j(x),\ \forall\ j \neq i\}, \tag{10}$$

with $D_i(x)$ given in Eq. 8. Averaging $D_i(x)$ over the samples in each partition and summing over all the partitions result in the total distortion,

$$D = \sum_{i=1}^{N} \int_{S_i} P(x) \int_{\mathbf{v}} P(\mathbf{v}|\mathbf{u} = \mathbf{u}_i) \left[x - g(\mathbf{v})\right]^2 d\mathbf{v} dx, \tag{11}$$

or,

$$D = \sum_{i=1}^{N} \int_{S_i} P(x) \left(x^2 - 2xB_i + C_i\right) dx, \tag{12}$$

where $P(x)$ is the source probability distribution.

We assume that there are a set of reconstruction levels $R_1, \ldots, R_N$ associated with quantizer partitions which will be used to describe the reconstruction function $g(\cdot)$. The reconstructor inputs a reliability vector $\mathbf{v}$ and calculates the probabilities $P(\mathbf{u} = \mathbf{u}_i|\mathbf{v}), i = 1, \ldots, N$ where $P(\mathbf{u} = \mathbf{u}_i|\mathbf{v})$ is the probability that the code word $\mathbf{u}_i$ is transmitted conditioned on the channel output $\mathbf{v}$. These probabilities are used to reconstruct the output sample $\hat{x}$ using one of two possible reconstruction rules: *hard reconstruction rule*, and *soft reconstruction rule*.

The hard reconstruction rule selects the code word which is most likely transmitted and outputs the corresponding reconstruction level, i.e.,

$$\hat{x} = R_i : P(\mathbf{u} = \mathbf{u}_i|\mathbf{v}) > P(\mathbf{u} = \mathbf{u}_j|\mathbf{v}),\ \ \forall\ j \neq i. \tag{13}$$

Systems with this reconstruction rule are well established in the literature [3, 4, 6, 8, 9, 10].

The soft reconstruction rule generates the reconstructed sample as a sum of the code word probabilities multiplied with their corresponding reconstruction levels, i.e.,

$$\hat{x} = g(\mathbf{v}) = \sum_{i=1}^{N} R_i P(\mathbf{u} = \mathbf{u}_i | \mathbf{v}). \tag{14}$$

In our earlier related work [27, 28], the reconstruction levels are simply selected as, $R_i = E(x|x \in S_i)$, which is the same rule as used in [8]. In the current work, we use an improved approach based on minimizing the average distortion in Eq. 12 through setting its derivative with respect to R_m, $m = 1, \ldots, N$, equal to zero. This results in N equations in terms of the N reconstruction levels:

$$2\sum_{i=1}^{N} \int_{S_i} xP(x)dx \frac{dB_i}{dR_m} = \sum_{i=1}^{N} \int_{S_i} P(x)dx \frac{dC_i}{dR_m}, \quad m = 1, \ldots, N. \tag{15}$$

The quantizer is designed using an iterative method by optimizing the encoder with Eq. 10 and the decoder with Eqs. 15. To design quantizers for different channels, the corresponding values B_i and C_i are calculated using appropriate equivalent channel model and reconstruction function.

In this work, we use a Gaussian source distribution, and as a result a closed form expression exists for the integrals over the source sample distribution. In case that the probability distribution of the input source is unknown, we can use a training sequence to estimate these integrals. In this case, for each training sample x, the partition which gives the lowest distortion is found. As each sample is quantized, statistics are collected for the partitions to estimate $P(x \in S_i)$ and $E(x|x \in S_i)$ and then calculate the necessary integrals with:

$$\begin{aligned} \int_{S_i} xP(x)dx &= P(x \in S_i)E(x|x \in S_i), \\ \int_{S_i} P(x)dx &= P(x \in S_i). \end{aligned} \tag{16}$$

2.1 Hard Decision Quantizer Design

Using the hard reconstruction function in Eq.13, and an appropriate equivalent channel, the presented formulation becomes either the classical Lloyd-Max quantizer [3, 4] or the channel optimized scalar quantizer [8]. These two are then compared with our proposed model based on a soft reconstruction rule.

The *Hard Noiseless Channel* model ignores the effect of the channel noise, where $\mathbf{v} = \mathbf{u}$, and the quantities B_i and C_i in Eq. 9 become:

$$\begin{aligned} B_i &= R_i, \\ C_i &= R_i^2. \end{aligned} \tag{17}$$

Solving for the set of Eqs. 15 results in a classical Lloyd-Max design algorithm [3, 4], where the reconstruction levels are given by:

$$R_m = \frac{\int_{S_m} xP(x)dx}{\int_{S_m} P(x)dx}, \quad m = 1, \dots, N \tag{18}$$

The *Hard Binary Symmetric Channel* model replaces the equivalent channel model with a binary symmetric channel with a bit error probability of ϵ. In this case, the relationship between the transmitted code word, $\mathbf{u}$, and the received code word, $\mathbf{v} = \hat{\mathbf{u}}$, is given by,

$$\begin{aligned} P(j|i) &\equiv P(\hat{\mathbf{u}} = \mathbf{u}_j|\mathbf{u}_i), \\ &= (\epsilon)^{h_{i,j}}(1-\epsilon)^{(r-h_{i,j})}, \end{aligned} \tag{19}$$

where $\hat{\mathbf{u}}$ is the received binary code word and $h_{i,j}$ is the Hamming distance between code words $\mathbf{u}_i$ and $\mathbf{u}_j$. With this model and using the hard reconstruction rule, the quantities B_i and C_i in Eq. 9 become:

$$\begin{aligned} B_i &= \sum_{j=1}^{N} P(j|i)R_j, \\ C_i &= \sum_{j=1}^{N} P(j|i)R_j^2, \end{aligned} \tag{20}$$

Solving the set of Eqs. 15 results in the channel optimized quantizer of [8] with the reconstruction levels given by,

$$R_m = \frac{\sum_{i=1}^{N} P(m|i) \int_{S_i} xP(x)dx}{\sum_{i=1}^{N} P(m|i) \int_{S_i} P(x)dx}, \quad m = 1, \dots, N. \tag{21}$$

2.2 Soft Decision Quantizer Design

For a system with soft decision, the reconstruction function in Eq. 14 is substituted into the distortion expression in Eq. 9, resulting in:

$$\begin{aligned} B_i &= \sum_{k=1}^{N} R_k \Theta_i(k), \\ C_i &= \sum_{j=1}^{N}\sum_{k=1}^{N} R_j R_k \Psi_i(j,k), \end{aligned} \tag{22}$$

where,

$$\begin{aligned} \Theta_i(k) &= \int_{\mathbf{v}} P(\mathbf{v}|\mathbf{u}=\mathbf{u}_i)P(\mathbf{u}=\mathbf{u}_k|\mathbf{v})d\mathbf{v}, \\ \Psi_i(j,k) &= \int_{\mathbf{v}} P(\mathbf{v}|\mathbf{u}=\mathbf{u}_i)P(\mathbf{u}=\mathbf{u}_j|\mathbf{v})P(\mathbf{u}=\mathbf{u}_k|\mathbf{v})d\mathbf{v}. \end{aligned} \tag{23}$$

The two sets of parameters, $\Theta_i(k)$ and $\Psi_i(j,k)$, capture the characteristics of the channel with soft reconstruction rule. The indices i, j and k denote the individual code words, and $\Theta_i(k)$, $\Psi_i(j,k)$ are computed for all such possible code word combinations.

To calculate these parameters, we first compute the single bit versions $\theta_i(k)$ and $\psi_i(j,k)$ where,

$$\begin{aligned} \theta_i(k) &= \int_{v} P(v|u=i)P(u=k|v)dv, \\ \psi_i(j,k) &= \int_{v} P(v|u=i)P(u=j|v)P(u=k|v)dv, \end{aligned} \tag{24}$$

The indices i, j and k in Eq. 24 can take the binary values 0 or 1. The two quantities are evaluated for all possible bit combinations, resulting in 4 different values for $\theta_i(k)$ and 8 different values for $\psi_i(j,k)$. Using the independence assumptions in Eqs. 1, 3, the set of parameters $\Theta_i(k)$ and $\Psi_i(j,k)$ are calculated by multiplying their single bit components, i.e., $\theta_i(k)$ and $\psi_i(j,k)$.

The quantizer partitions are determined by substituting B_i and C_i in Eq. 10. The reconstructor is designed by substituting B_i and C_i in Eq. 15, resulting in a set of linear equations,

$$\sum_{i=1}^{N} \Theta_i(k) \int_{S_i} xP(x)dx = \sum_{j=1}^{N} R_j \sum_{i=1}^{N} \Psi_i(j,k) \int_{S_i} P(x)dx, \tag{25}$$

where $k = 1, \dots, N$. This set of linear equations can be expressed in matrix form as:

$$\begin{bmatrix} E(1) \\ \vdots \\ E(N) \end{bmatrix} = \begin{bmatrix} F(1,1) & \dots & F(1,N) \\ \vdots & \ddots & \vdots \\ F(N,1) & \dots & F(N,N) \end{bmatrix} \begin{bmatrix} R_1 \\ \vdots \\ R_N \end{bmatrix}, \tag{26}$$

where

$$\begin{aligned} E(k) &= \sum_{i=1}^{N} \Theta_i(k) \int_{S_i} xP(x)dx, \\ F(j,k) &= \sum_{i=1}^{N} \Psi_i(j,k) \int_{S_i} P(x)dx. \end{aligned} \tag{27}$$

The solutions $R_i, i = 1, \dots, N$ to this set of equations result in the optimal reconstruction levels.

By a direct replacement of 23 in 25 and using the identity,

$$P(\mathbf{v}|\mathbf{u}=\mathbf{u}_i|) = \frac{P(\mathbf{u}=\mathbf{u}_i|\mathbf{v})}{P(\mathbf{u}=\mathbf{u}_i)}P(\mathbf{v}), \text{ for } P(\mathbf{u}=\mathbf{u}_i) \neq 0 \tag{28}$$

it can be shown that the set of Eqs. 25 simplifies to a trivial case where the the reconstruction level R_i is equal to expected value of the source samples in the corresponding partition S_i (this requires all the partitions to be non-empty in order to use the identity in Eq. 28), i.e.,

$$R_i = \frac{\int_{S_i} xP(x)dx}{\int_{S_i} P(x)dx}, \text{ if } \int_{S_i} P(x)dx > 0, \quad \forall i. \tag{29}$$

However, as the channel noise increases some of the quantizer partitions become empty. In mathematical terms, the mth partition will be empty if for all inputs x, we have,

$$\forall x \quad \exists i \neq m \quad \text{such that} \quad D_i(x) < D_m(x) \tag{30}$$

where $D_i(x)$, $D_m(x)$ are given in 8. Note that such empty partitions are decided by the design algorithm to optimize the overall performance and should not be confused with the cases of having empty partitions due to using a small training data sequence (the algorithms presented in this work do not use any training data and all the computations are based on closed form expressions). It is also crucial to note that the existence of such empty partitions can not be captured in our model. This is because the conditional symbol probabilities are computed by multiplying the conditional probabilities of their bit components. In other words, using Eq. 3 and Eq. 5, one obtains a non-zero probability $P(\mathbf{u} = \mathbf{u}_i|\mathbf{v})$ even for an empty partition S_i, and consequently, one needs the correct value for the corresponding reconstruction level R_i to be used in conjunction with the $P(\mathbf{u} = \mathbf{u}_i|\mathbf{v})$ in 14. The use of a more sophisticated model to capture the fact that some of the channel input symbols do not occur at all would substantially increase the overall complexity and is beyond the scope of the current work. In summary, by solving the system of equation in 25, one computes the correct reconstruction value for all the partitions to be used in 14 including the partitions that are empty. We will later provide numerical results comparing these two cases (expected value vs. system of equations). To have a fair comparison, in the case of expected value, we have deleted the term corresponding to the empty partitions in 14 and have renormalized the probabilities to add to unity. It is observed that this strategy results in a large improvement for the method based on the expected value as compared to using the mean value of the source (zero) as the default value for the reconstruction level of the empty partitions.

In order to estimate the channel parameters $\theta_i(j)$ and $\psi_i(j,k)$, the equivalent channel in Fig. 1 is replaced by a *Reliability Information Channel* model. It based on the assumption that the equivalent channel model output a reliability information value $P(u = k|v)$ for every input bit.

To compute $\theta_i(k)$, we define the reliability information as a random variable X_k:

$$X_k = P(u = k|v), \quad k = 0, 1. \tag{31}$$

Note that X_k, as a function of v, is a random variable. The identities in Eqs. 5, 6 provide a one-to-one relationship between v and X_k, and the value of $k = 0, 1$ determines which of the two identities are used for this purpose. With this one-to-one mapping, the probability distribution of X_k is expressed in terms of the probability distribution of v by the transformation:

$$P(X_k)dX_k = P(v)dv. \tag{32}$$

Conditioning both sides of the equation on $u = i$ results in:

$$P(X_k|u = i)dX_k = P(v|u = i)dv. \tag{33}$$

Substituting Eqs. 31, 33 in Eq. 24 results in,

$$\begin{aligned} \theta_i(k) &= \int_v P(v|u = i)P(u = k|v)dv, \\ &= \int_0^1 X_k P(X_k|u = i)dX_k, \\ &= E(X_k|u = i). \end{aligned} \tag{34}$$

The quantity $E(X_k|u = i)$ is measured by passing a test bit stream through the channel and observing the channel output.

Similarly, the quantity $\psi_i(j, k)$ is calculated by defining a random variable $Y_{j,k}$ as,

$$Y_{j,k} = P(u = j|v)P(u = k|v), \quad j, k = 0, 1. \tag{35}$$

Using the identities in Eqs. 5, 6, we obtain,

$$P(Y_{j,k})dY_{j,k} = P(v)dv. \tag{36}$$

Conditioning both sides on $u = i$ results in,

$$P(Y_{j,k}|u = i)dY_{j,k} = P(v|u = i)dv. \tag{37}$$

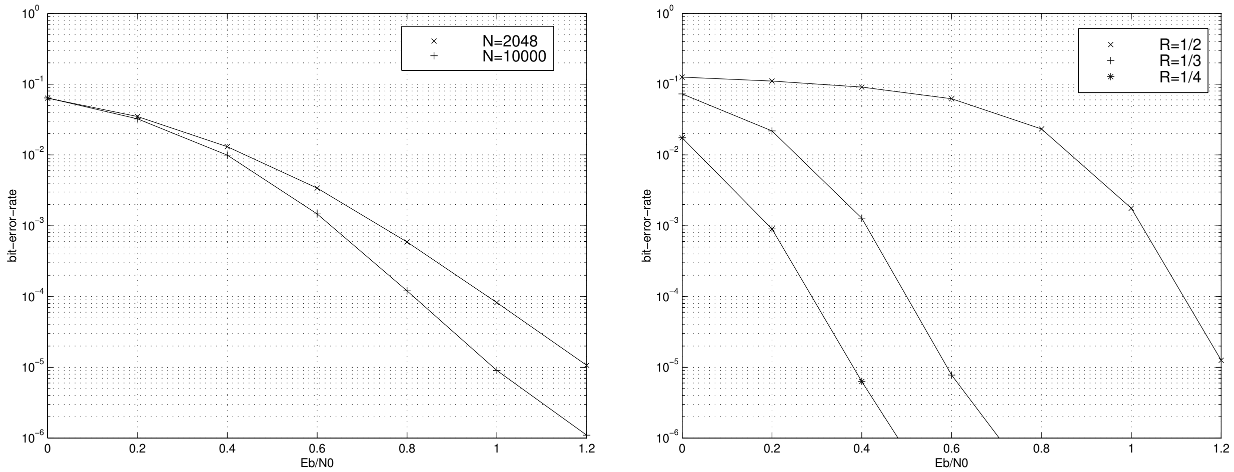
Substituting Eqs. 35, 37 in Eq. 24 results in,

$$\begin{aligned} \psi_i(j, k) &= \int_v P(v|u = i)P(u = j|v)P(u = k|v)dv, \\ &= \int_0^1 Y_{j,k} P(Y_{j,k}|u = i)dY_{j,k}, \\ &= E(Y_{j,k} \mid u = i). \end{aligned} \tag{38}$$

The quantity $E(Y_{j,k}|u = i)$ is also measured by passing a test bit stream through the channel and observing the output. Using the parameters, $\theta_i(j)$ and $\psi_i(j, k)$, we calculate the parameters, $\Theta_i(j)$ and $\Psi_i(j, k)$, and design the the quantizer as outlined earlier.

3 Numerical Results

3.1 Simulation Setup



(a) $M = 2$, Rate $1/3$

(b) $M = 3$, $N = 10000$

Figure 2: Bit-error-rate curves for Turbo-code channel coding.

The proposed quantization scheme has been simulated for an independent identically distributed Gaussian input. The channel is Additive White Gaussian Noise (AWGN) with Binary Phase Shift Keying (BPSK) modulation[2]. The quantizers are designed and tested with a Turbo-code with a block length of 10000 bits. The Turbo-codes presented use an S-random interleaver [29] with $S = 15$, and the number of iterations for the Turbo decoder is equal to 5. The main simulation is performed using a rate 1/3 Turbo-code, composed of two component codes with $M = 2$ memory elements, forward polynomial (101) and backward polynomial (111). The bit-error-rate curves for these codes are shown in Fig. 2(a).

To show the trade-off between the source and channel coding, some simulations are also performed with rates 1/2, 1/3, and 1/4 Turbo-code. The Turbo-code used for these simulations has $M = 3$ memory elements, and block length of 10000 bits. To generate the different rates, the basic Turbo-code structure has been modified by replacing each of the component codes with a pair of encoders. The two encoders in each pair have the forward polynomials (1011), (1111), and the same backward polynomial (1101). Different coding rates are achieved by puncturing the encoder outputs. The bit-error-rate curves for these

[2]The specific model of AWGN channel with BPSK modulation is used as a framework to present numerical results. However, the quantization methods proposed are general and can be used in conjunction with other channel models and/or modulation schemes.

Quantizer	Channel Model
Lloyd-Max	Noiseless
Channel Optimized	Binary Symmetric
Soft Decision (Expected Value)	Reliability Information
Soft Decision (System of Equations)	Reliability Information

Table 1: Quantizers tested.

codes using a block length of 10000 are shown in Fig. 2(b).

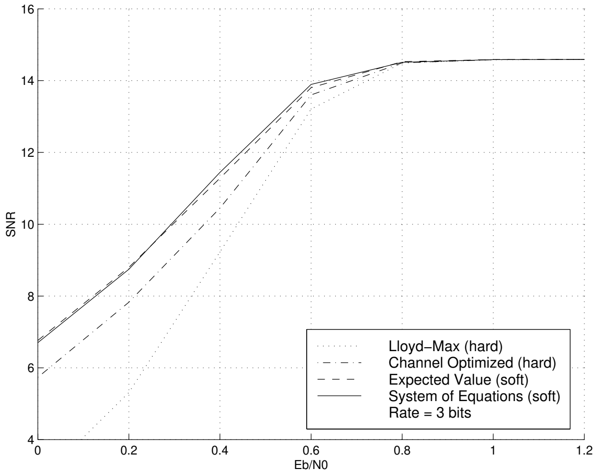
Total of four different quantizers are designed and tested, as listed in Table 1. The first quantizer is the classical Lloyd-Max Quantizer designed assuming noiseless channel and tested over a noisy channel with hard decision rule. The channel optimized quantizer is designed taking into effect the effect of the noise and tested with a hard decision rule. Two soft decision rule quantizers are designed using the reliability information channel. The first is designed using the expected value in Eq. 29 to calculate the reconstruction levels (where the undefined reconstruction levels are selected as a default value equal to the mean of the source sample distribution), while the second one uses the system of equations in Eq. 25.

To reduce the impact of the initialization, the code words are initialized randomly and best of 100 designs is chosen. In each case, the quantizer and reconstructor design algorithms are iterated until the relative change in the mean square error is less that 10^{-3}. Since the source sample distribution is known to be Gaussian, the quantizers are designed by a direct substitution in Eq. 16 without the use of a training sequence.

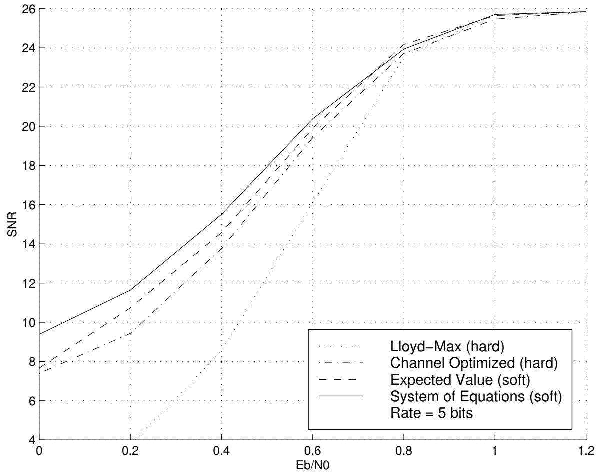
3.2 Quantizer Results

The results for a 3 and 5 bit quantizers are shown in Fig. 3. As expected, the Lloyd-Max quantizer operating with hard reconstruction rule results in the worst performance. The channel optimized quantizer also uses a hard reconstruction rule, however, the design takes the effect of the channel noise into account and the performance is substantially improved. The soft reconstruction quantizers offer an improvement over both of the hard reconstruction quantizers. Of the two soft decision quantizers, the system of equations design performs better than the the expected value design. Fig. 4 shows the effect of using soft reconstruction rule with the Lloyd-Max and Channel Optimized quantizers, where the performance of both quantizers is improved using the soft reconstruction rule.

The soft decision quantizer is designed using both the system of equations in Eq. 25 and the expected value in Eq. 29. During the design, Eq. 3 relies on the assumption that the bits within a code word are independent of each other, however in practice this assumption is not entirely valid. To improve the

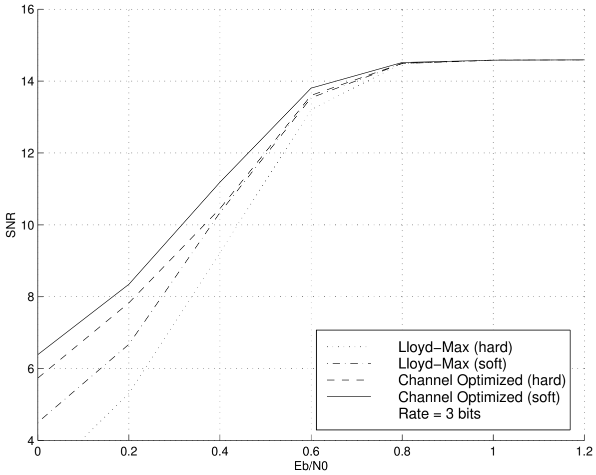


(a) 3 bit quantizers

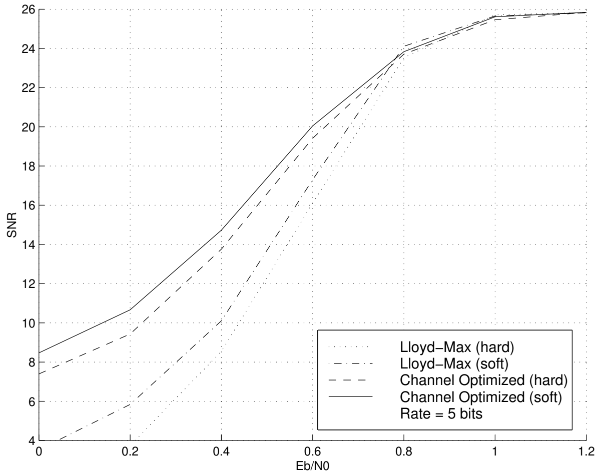


(b) 5 bit quantizers

Figure 3: Distortion for 3 and 5 bit quantizers with rate 1/3 Turbo-code with $M = 2$ memory elements.

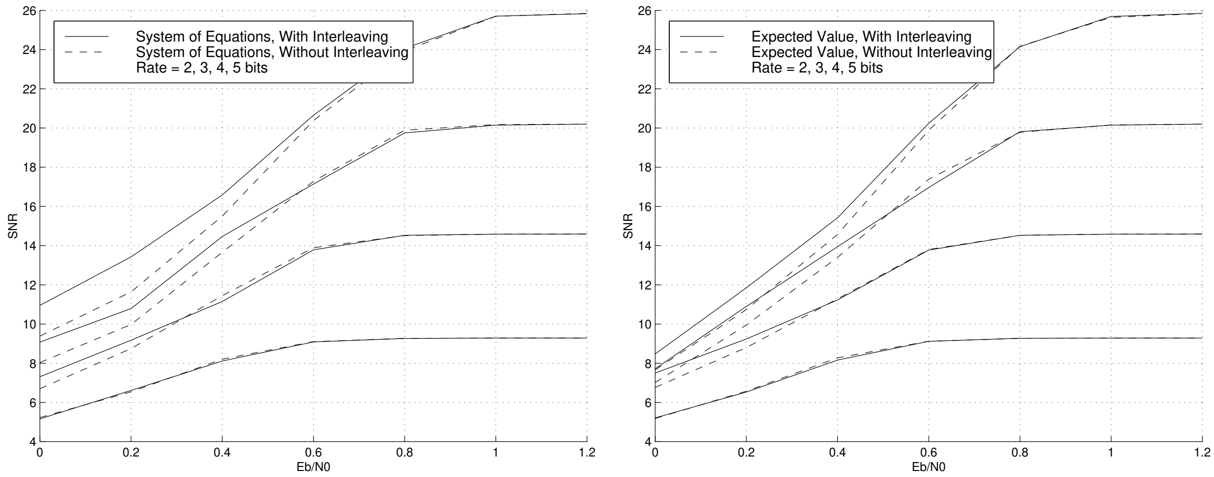


(a) 3 bit quantizers



(b) 5 bit quantizers

Figure 4: Distortion for 3 and 5 bit Lloyd-Max and Channel Optimized quantizers using hard and soft reconstruction rules with rate 1/3 Turbo-code with $M = 2$ memory elements.



(a) Quantizer designed with system of equations

(b) Quantizer designed with expected value

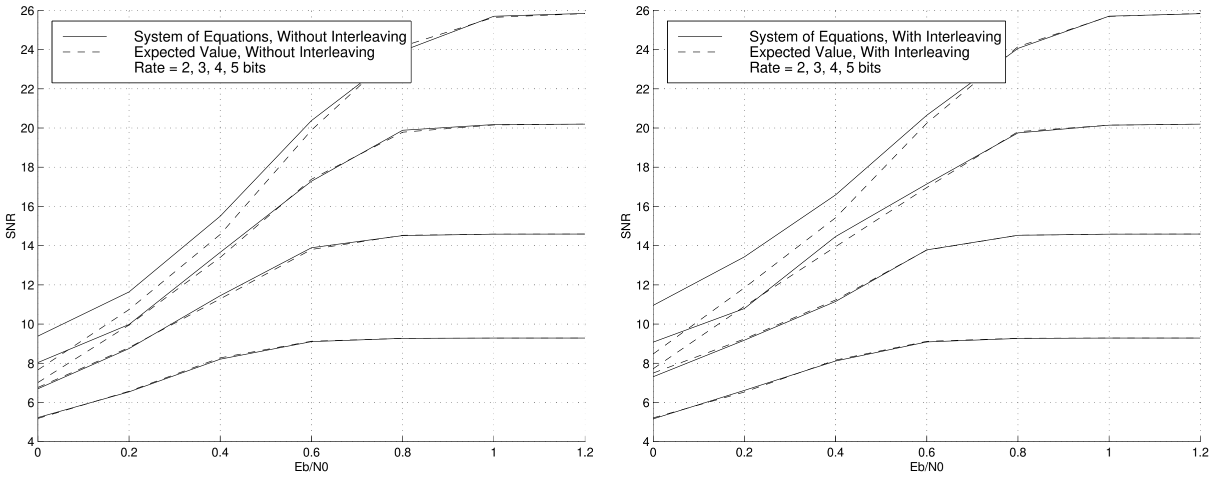
Figure 5: Soft decision quantizer with rate 1/3 Turbo-code with $M = 2$ memory elements, illustrating the effect of interleaving the quantized bits before channel coding.

independence assumption, the entire block of 10000 bits is interleaved with an S-random interleaver before encoding with Turbo-code. Fig. 5 shows the effect of interleaving the bits before the channel code, and an improvement in performance is achieved for both of the soft decision quantizers. The results shown in Fig. 6 indicate that the quantizer designed with the system of equation offers an improvement over the expected value in both cases with and without interleaving of the bits.

3.3 Spectral Efficiency Comparison

To study the effect of the spectral efficiency, both the Turbo-code rate and the quantizer rate have been varied to achieve systems that operate with the same overall transmitted rate per source sample. The Turbo-code used for this comparison is the encoder with $M = 3$ described earlier, with a block length of 10000 bits.

Fig. 7(a) shows the 2 bit quantizers with a rate 1/3 Turbo-code, compared to the 3 bit quantizers with rate 1/2 Turbo-code. At low E_b/N_0, the channel noise has a significant effect on the quantizer performance, and the quantizers that allocate more bits toward the channel coding have better performance. At high E_b/N_0, the channel noise has little impact on the quantizer performance, and the systems that allocate more bits toward the source coding have better performance. Similar results are shown in Fig. 7(b), where Turbo-code rate 1/2 with 4 bit quantizer is compared with Turbo-code rate 1/4 with 2 bit quantizer. Fig. 7(c) shows another comparison of Turbo-code rate 1/3 with 4 bit quantizer compared with Turbo-



(a) Quantizer with no bit interleaving (b) Quantizer with bit interleaving

Figure 6: Soft decision quantizer with rate 1/3 Turbo-code with $M = 2$ designed with the system of equation and expected value methods.

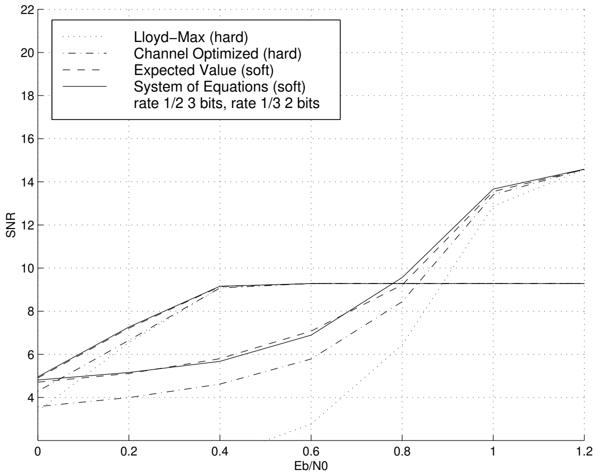
code rate 1/4 with 3 bit quantizer.

3.4 Channel Mismatch

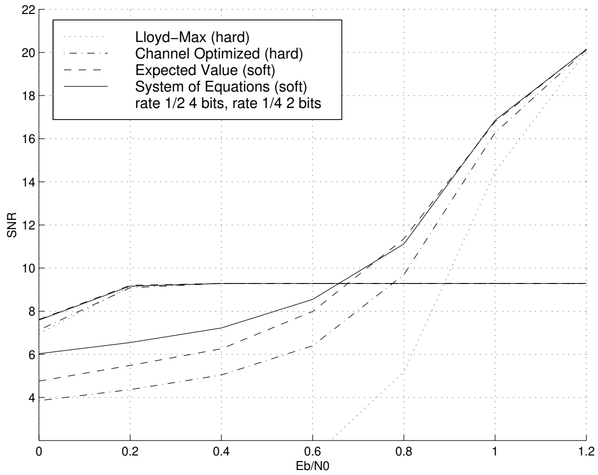
All the quantizers presented in this work operate under the assumption that the channel noise level is known and does not change during the transmission. In practice, the channel noise level fluctuates over time. Fig. 8 shows the effect of such mismatch for the 4 bit quantizers. The curve labeled *fully optimized* corresponds to the case that the quantizer and reconstructor are designed to match the operating channel noise level. The curve labeled *not optimized* corresponds to the case that the quantizer is designed for $E_b/N_0 = 0.8$, and operated at different channel noise levels.

In another variation, the quantizer is designed for a fixed noise level, however at the last step the decoder is recalculated for the actual noise level. This results in a class of quantizers with the same encoder and different decoders. This situation is shown with the curve labeled *reconstructor optimized*, where the quantizer has been designed for $E_b/N_0 = 0.8$. This situation is realistic because a soft output channel decoder (including a Turbo decoder) usually assumes knowledge of the channel noise variance, and consequently, the reconstructor structure can be adjusted accordingly.

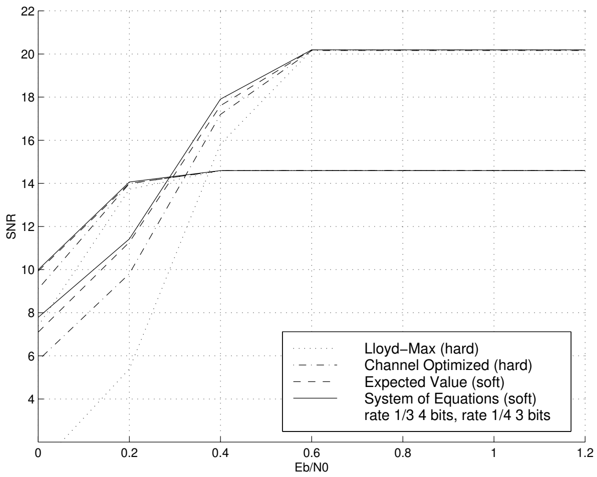
Fig. 8(a) shows the channel mismatch for the 5 bit Channel Optimized quantizer with a rate 1/3 Turbo-code. At $E_b/N_0 = 0.2$, the mismatch results in 4.2 dB degradation in the sample signal-to-noise ratio from the fully optimized case, and the adaptive reconstructor reduces this degradation to 3.4 dB.



(a) Rate 1/2, 3 bit and rate 1/3, 2 bit quantizer



(b) Rate 1/2, 4 bit and rate 1/4, 2 bit quantizer



(c) Rate 1/3, 4 bit and rate 1/4, 3 bit quantizer

Figure 7: Spectral efficiency comparisons.

Fig. 8(b) shows the channel mismatch for the 5 bit soft decision quantizer designed with expected value. The sensitivity to the mismatch at $E_b/N_0 = 0.2$ is 4.0 dB, however since the reconstructor design in Eq. 29 is independent of the noise level, no further gain can be achieved with reconstructor optimization. Fig. 8(c) shows the channel mismatch for the 5 bit soft decision quantizer designed with the system of equations. The distortion at $E_b/N_0 = 0.2$ is 4.5 dB below the fully optimized case, however 0.3 dB better than the soft decision quantizer designed with expected value. Optimizing the reconstructor improves the distortion by 0.3 dB.

4 Summary

We have discussed a new method of combined source-channel coding using Turbo-codes, or more generally a channel code with reliability information. A system is presented that transmits a discrete time, continuous amplitude signal over a noisy channel. The transmitted samples are encoded using a fixed rate scalar quantizer, and Turbo-code channel coding is used for error correction. The system performance is improved by using a soft reconstruction of samples using the reliability information available at the channel decoder. The quantizer and the reconstructor are jointly optimized using an iterative algorithm to minimize the end-to-end distortion. The performance of the presented quantizers has been shown to be up to 1.0 dB better than the conventional channel optimized scalar quantizer. The effect of the trade-off between the rates allocated to the source and channel coders, as well as, the effect of the channel mismatch are studied. It is shown that the increase in distortion due to a channel mismatch can be reduced from approximately 3.5 dB to approximately 1.5 dB with an adaptive receiver.

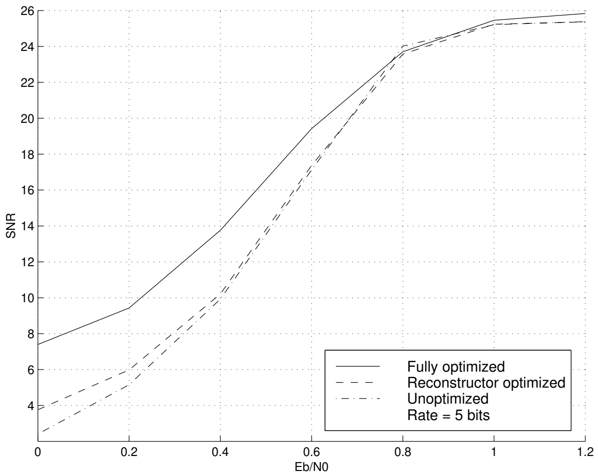
References

[1] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions On Information Theory*, vol. 20, no. 2, March 1974, pp. 284–287.

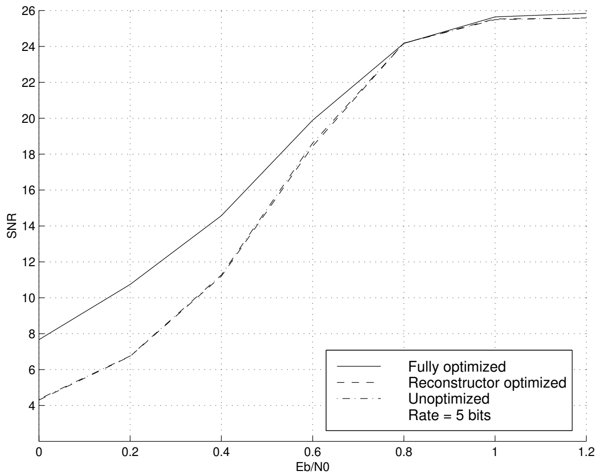
[2] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding: Turbo codes(1). In *Proceedings of the 1993 International Conference on Communication*, pp. 1064–1070, Geneva, Switzerland, May 1993.

[3] J. Max, "Quantizing for minimum distortion," *IEEE Transactions On Information Theory*, March 1960, pp. 7–12.

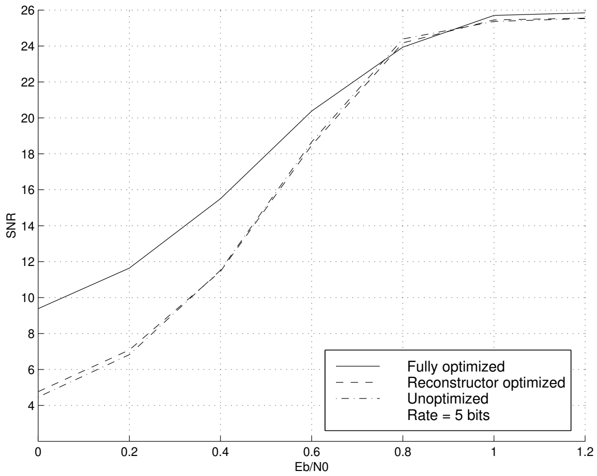
[4] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions On Information Theory*, vol. IT-28, no. 2, March 1982, pp. 129–137.



(a) Hard Decision Channel optimized



(b) Soft decision (Expected Value)



(c) Soft decision (System of Equations)

Figure 8: Channel mismatch for the Channel optimized and the Soft decision (expected value and system of equations) quantizers with rate 1/3 Turbo-code with $M = 2$.

[5] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions On Communication*, vol. COM-28, January 1980, pp. 84–94.

[6] A. J. Kurntenbach and P. A. Wintz, "Quantizing for noisy channels," *IEEE Transactions on Communciation Technology*, vol. COM-17, no. 2, April 1969, pp. 291–302.

[7] N. Rydbeck and C. W. Sundberg, "Analysis of digital errors in nonlinear PCM systems," *IEEE Transactions on Communications*, vol. COM-24, January 1976, pp. 59–65.

[8] N. Farvardin and V. A. Vaishampayan, "Optimal quantizer design for noisy channels: An approach to combined source-channel coding," *IEEE Transactions On Information Theory*, vol. 33, no. 6, November 1987, pp. 827–838.

[9] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Transactions On Information Theory*, vol. 36, no. 4, July 1990, pp. 799–809.

[10] N. Farvardin and V. A. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," *IEEE Transactions On Information Theory*, vol. 37, no. 1, January 1991, pp. 155–160.

[11] V. A. Vaishampayan and N. Farvardin, "Joint design of block source codes and modulation signal sets," *IEEE Transactions On Information Theory*, vol. 38, no. 4, July 1992, pp. 1230–1248.

[12] F. I. Alajaji and N. C. Phamdo, "Soft-decision COVQ for rayleigh-fading channels," *IEEE Communications Letters*, vol. 2, no. 6, June 1998, pp. 162–164.

[13] N. Phamdo and F. Alajaji, "Soft-decision demodulation design for COVQ over white, colored, and isi gaussian channels," *IEEE Transactions On Communication*, vol. 48, no. 9, September 2000, pp. 1499–1506.

[14] G-C. Zhu and F. I. Alajaji, "Soft-decision covq for turbo-coded awgn and rayleigh fading channels," *IEEE Communications Letters*, vol. 5, no. 6, June 2001, pp. 257–259.

[15] M. Skoglund and P. Hedelin. Vector quantization over a noisy channel using soft decision decoding. In *Proceedings of IEEE International Conference On Acoustics, Speech, and Signal Processing*, pp. 605–608, Adelaide, Australia, April 1994.

[16] M. Skoglund and P. Hedelin, "Hadamard-based soft decoding for vector quantization over noisy channels," *IEEE Transactions On Information Theory*, vol. IT-45, no. 2, March 1999, pp. 515–532.

[17] M. Skoglund. A soft decoder vector quantizer for a rayleigh fading channel - application to image transmission. In *Proceedings of IEEE International Conference On Acoustics, Speech, and Signal Processing*, pp. 2507–2510, Detroit, Michigan, USA, 1995.

[18] T. Ottosson and M. Skoglund. Soft multiuser decoding for vector quantization over rayleigh fading CDMA channels. In *Proceedings of IEEE Symposium on Spread Spectrum Techniques and Applications*, pp. 22–25, Mainz, Germany, 1996.

[19] T. Ottosson and M. Skoglund, "Soft multiuser decoding for vector quantization over a cdma channel," *IEEE Transactions On Information Theory*, vol. IT-46, no. 3, March 1998, pp. 327–337.

[20] D. Miller and K. Rose, "Combined source-channel vector quantization using deterministic annealing," *IEEE Transactions On Information Theory*, vol. 38, July 1994, pp. 1249–1257.

[21] S. Gadkari and K. Rose, "Robust vector quantizer design by noisy channel relaxation," *IEEE Transactions On Information Theory*, vol. 47, no. 8, August 1999, pp. 1113–1116.

[22] K. Rose, "deterministic annealing for clustering, compression, classification, regression and related optimization problems," *Proceedings of the IEEE*, vol. 86, no. 11, November 1998, pp. 2210–2239.

[23] K-P. Ho, "Soft-decoding vector quantizer using reliability information from turbo codes," *IEEE Communications Letters*, vol. 3, no. 7, July 1999, pp. 208–210.

[24] J. He, D. J. Costello, Y. Huang, and R. L. Stevenson. On the application of turbo codes to the robust transmission of compressed images. In *IEEE International Conference on Image Processing*, pp. 559–562, Santa Barbara, California, USA, October 1997.

[25] Z. Peng, Y. Huang, D. J. Costello, and R. L. Stevenson. On the application of turbo codes to the robust transmission of compressed images. In *The Proceedings of the Conference on Information Sciences and Systems*, pp. 330–335, Princeton, NJ, USA, March 1998.

[26] W. B. Pennerbaker and J. L. Mithchell, *JPEG: Still Image Compression Standard.* Van Nostrand Rheinhold, New York, NY, 1993.

[27] J. Bakus and A. K. Khandani. Combined source-channel coding using turbo-codes. In *The Proceedings of the Conference on Information Sciences and Systems*, pp. 309–313, Baltimore, Mariland, USA, 1997.

[28] J. Bakus and A. K. Khandani, "Combined source-channel coding using turbo-codes," *Electronics Letters*, vol. 33, no. 13, September 1997, pp. 1613–1614.

[29] S. Dolinar and D. Divsalar, "Weight distributions for turbo codes using random and nonrandom permutations," *Jet Propulsion Laboratory, Pasadena, California*, vol. TDA42-122, August 1995, pp. 56–65.