Channel Optimized Sample Adaptive Product Quantization*

Zahir Raza, Fady Alajaji and *Tamás Linder*

Mathematics and Engineering

Department of Mathematics and Statistics

Queen's University

Kingston, ON, Canada, K7L 3N6

E-mail: zahir,fady,linder@mast.queensu.ca

Abstract

Channel optimized vector quantization (COVQ), as a joint source-channel coding scheme, has proven to perform well in compressing a source and making the resulting quantizer robust to channel noise. Unfortunately like its counterpart in the noiseless channel case, the vector quantizer (VQ), the COVQ encoding complexity is inherently high. Sample adaptive product quantization was recently introduced by Kim and Shroff to reduce the complexity of the VQ while achieving comparable distortions, even for moderate quantization dimensions. In this paper, we investigate the SAPQ for the case of noisy channels and employ the joint source-channel approach of optimizing the quantizer design by taking into account both source and channel statistics. It is shown that, like its counterpart in the noiseless case, the channel optimized SAPQ achieves comparable performance results to the COVQ (within 0.2-1.0 dB), while maintaining considerably lower encoding complexity (half of that of COVQ) and storage requirements.

1 Introduction

Recently, Kim and Shroff introduced in [8, 9] a constrained vector quantizer structure called the sample adaptive product quantizer (SAPQ) that achieves a comparable performance to the vector quantizer (VQ) [13] while maintaining a lower encoding complexity (refer also to [3, 4, 11, 12, 14] for previous related work). Yet, as with most data compression schemes that solely remove source redundancy, the compressed source tends to be more sensitive to channel noise. Traditionally, tandem source-channel coding was used to achieve reliable transmission of information by separately designing the source and channel codes. It is however known that when there are delay and complexity constraints, it is more advantageous to employ joint source-channel coding where the source and channel codes are designed in cohesion (e.g., [1], [2], [5]-[7],[10], [15]-[17]).

There are three main approaches to joint source-channel coding: the unequal error protection approach, the zero-redundancy channel coding approach, and the combined

* This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

source-channel coding approach. In this paper, we focus on the third approach, where both channel noise and source statistics are included in the design of the source coders. VQ's designed in such a way are labeled channel optimized vector quantizers (COVQ's). COVQ has received a considerable amount of attention due to its improvement in performance over VQ in the presence of channel noise (e.g., [7, 15]). However, COVQ incurs high encoding complexity. In this work, we study the design of SAPQ for noisy memoryless channels, or channel optimized SAPQ (CO-SAPQ), in order to find a less complex alternative to COVQ. Countering [8], we will design and implement two types of channel optimized SAPQ's, namely the COm-SAPQ and the CO1-SAPQ.

2 Preliminaries

The channel considered in this paper is the binary symmetric channel (BSC), and the distortion measure used is the mean squared distortion: $d(\underline{x},\underline{y}) = \|\underline{x}-\underline{y}\|^2$. Let N be any integer to define $J_N = \{1,\ldots,N\}$, $P(l|i)$ as the probability of receiving index l given that index i was sent over the BSC, and $p(\underline{x})$ as the probability density function (p.d.f.) of the source vector $\underline{x}$.

A (k,N) COVQ consists of an encoder ε, a decoder g and their corresponding codebook $C = \{\underline{c}_l\}_{l=1}^N$. The k dimensional source vector $\underline{x}$ is encoded by the encoder ε into an index i. The encoding is such that:

$$\varepsilon(\underline{x}) = i \quad \text{if} \quad \underline{x} \in S_i, \quad \text{where} \quad i \in J_N \text{ and } S_i = \text{ encoding region of } i.$$

The binary expansion of index i is transmitted through a BSC, one bit at a time, and is received as l at the receiver, where $l \in J_N$. The received index is reproduced at the decoder using the codebook $C = \{\underline{c}_l\}_{l=1}^N$. The decoding function g is such that $g(l) = \underline{c}_l$; it is a one-to-one mapping from the indices to the reproduction levels $\underline{c}_l$. The end-to-end distortion is described by:

$$D = \sum_{i=1}^{N}\sum_{l=1}^{N}\int_{S_i} P(l|i)d(\underline{x},\underline{c}_l)p(\underline{x})d\underline{x}. \tag{1}$$

It is shown in [7] that the necessary conditions to minimizing (1) are:

$$S_i = \left\{\underline{x} : \sum_{l=1}^{N} P(l|i)d(\underline{x},\underline{c}_l) \leq \sum_{l=1}^{N} P(l|\gamma)d(\underline{x},\underline{c}_l) \quad \forall\gamma \in J_N\right\} \tag{2}$$

given the codebook $C = \{\underline{c}_l\}_{l=1}^N$, and:

$$\underline{c}_l = \frac{\sum_{i=1}^{N} P(l|i)\int_{S_i}\underline{x}p(\underline{x})d\underline{x}}{\sum_{i=1}^{N} P(l|i)\int_{S_i}p(\underline{x})d\underline{x}} \tag{3}$$

given the encoding regions $\{S_i\}_{i=1}^N$.

In this paper, we also use a (k,m,N) channel optimized product quantizer (COPQ), which is merely a bank of COVQ's. The COPQ can be viewed as the counterpart of the product quantizer (PQ) for noiseless channel quantization [8]. The rate of both schemes is:

$$R = \frac{\log_2 N}{k} \quad \text{bits/source sample}\ . \tag{4}$$

3 CO-SAPQ System Model

Figure 1 depicts the basic structure of a (k,m,N,η) COm-SAPQ. The COm-SAPQ consists of 2^η product encoders $\{\mathrm{PE}_j\}_{j=1}^{2^\eta}$. Each PE_j consists of m parallel encoders $\{\varepsilon_{s,j}\}_{s=1}^{m}$, that have m corresponding decoder functions $\{g_{s,j}\}_{s=1}^{m}$ at the decoder. Adhering the set of encoders $\{\varepsilon_{s,j}\}$ and decoder functions $\{g_{s,j}\}$ is the codebook $\mathbf{C}_j$. The codebook $\mathbf{C}_j$ is constructed by the product of m codebooks $\mathbf{C}_j = C_{1,j} \times \ldots \times C_{m,j}$. Copies of the input source vector $\underline{\mathbf{x}} = (\underline{x}_1, \ldots, \underline{x}_m)$ are encoded by each PE_j to produce an index vector $\underline{I}_j$ as follows:

$$\mathrm{PE}_j(\underline{\mathbf{x}}) = (\varepsilon_{1,j}(\underline{x}_1), \ldots, \varepsilon_{m,j}(\underline{x}_m)) = \underline{I}_j, \text{ where } \varepsilon_{s,j}(\underline{x}_s) \in J_N, \underline{x}_s \in \mathbb{R}^k \text{ and } \underline{I}_j \in J_N^m.$$

Each index vector $\underline{I}_j$ has a distortion associated to it, and the index vector ($\underline{I} = (i_1, \ldots, i_m)$; $i_s \in J_N$) with the minimum distortion is transmitted over the channel along with the index (j^*) of the PE_{j^*} that produced the index vector $\underline{I}$. Index vector $\underline{I}$ and index j^* are transmitted over the channel, and received as $\underline{L}$ ($\underline{L} \in J_N^m$) and j' ($j' \in J_{2^\eta}$). The decoder decodes $\underline{L} = (l_1, \ldots, l_m)$, where $l_s \in J_N$, using j' to indicate which set of decoder functions to use; i.e., set $\{g_{s,j'}\}_{s=1}^{m}$ as follows:

$$\mathrm{Decoder}(\underline{L}, j') = (g_{1,j'}(l_1), \ldots, g_{m,j'}(l_m)) = (\underline{c}_{l_1}^{[1,j']}, \ldots, \underline{c}_{l_m}^{[m,j']}) = \underline{\mathbf{c}}_{\underline{L}}^{[j']},$$

where $\underline{c}_l^{[s,j']} \in C_{s,j'}$ for $l \in J_N$, and $\underline{\mathbf{c}}_{\underline{L}}^{[j']} \in \mathbf{C}_{j'}$ for $\underline{L} \in J_N^m$.

The (k,m,N,η) CO1-SAPQ is similar to the COm-SAPQ except that the PE's are replaced by repeated product encoders RPE. A RPE_j repeats the same encoder ε_j function throughout the m encoding blocks. The encoder ε_j has a decoder function g_j corresponding to it at the receiver. The pair of encoders ε_j and decoders g_j are associated by a single codebook C_j. Now the codebook of RPE_j is $\mathbf{C}_j = C_j \times \ldots \times C_j$, where the product is taken m times. The encoding and decoding for CO1-SAPQ are similar to the COm-SAPQ's:

$$\mathrm{RPE}_j(\underline{\mathbf{x}}) = (\varepsilon_j(\underline{x}_1), \ldots, \varepsilon_j(\underline{x}_m)) = \underline{I}_j, \text{ where } \varepsilon_j(\underline{x}_s) \in J_N, \underline{x}_s \in \mathbb{R}^k \text{ and } \underline{I}_j \in J_N^m,$$

and

$$\mathrm{Decoder}(\underline{L}, j') = (g_{j'}(l_1), \ldots, g_{j'}(l_m)) = (\underline{c}_{l_1}^{[j']}, \ldots, \underline{c}_{l_m}^{[j']}) = \underline{\mathbf{c}}_{\underline{L}}^{[j']},$$

where $\underline{c}_l^{[j']} \in C_{j'}$ for $l \in J_N$, and $\underline{\mathbf{c}}_{\underline{L}}^{[j']} \in \mathbf{C}_{j'}$ for $\underline{L} \in J_N^m$. The rate of both schemes is

$$R = \frac{\log_2 N}{k} + \frac{\eta}{km} \quad \text{bits/source sample.} \tag{5}$$

4 Necessary Conditions for Optimality

To simplify notation define: $v_s(\underline{I})$ = the s^{th} index component of $\underline{I}$; $v_s : J_N^m \to J_N$, and: $u_s(\underline{\mathbf{x}}) = \underline{x}_s$; $s = 1, \ldots, m$; $u_s : \mathbb{R}^{km} \to \mathbb{R}^k$.

Distortion: Let $\mathbf{S}_{\underline{I}}^{[j^*]}$ be the encoding region for index vector $\underline{I}$ and index j^*: $\mathbf{S}_{\underline{I}}^{[j^*]} = \{\underline{\mathbf{x}} : \text{COm-SAPQ Encoder}(\underline{\mathbf{x}}) = (\underline{I}, j^*)\}$. When a source sample $\underline{\mathbf{x}}$ is encoded into the index vector $\underline{I}$ and index j^*, at the receiver we can potentially receive any $\underline{L} \in J_N^m$ and

$j' \in J_{2^\eta}$, and then the end reproduction of $\underline{\mathbf{x}}$ becomes $\underline{\mathbf{c}}_{\underline{L}}^{[j']}$. The mean squared end-to-end distortion of such a system is:

$$\mathrm{D}_{\mathit{COm\text{-}SAPQ}} = \sum_{j^*=1}^{2^\eta} \sum_{\underline{I} \in J_N^m} \int_{\mathbf{S}_{\underline{I}}^{[j^*]}} \sum_{j'=1}^{2^\eta} \sum_{\underline{L} \in J_N^m} P(j'|j^*) P(\underline{L}|\underline{I}) \sum_{s=1}^{m} \|u_s(\underline{\mathbf{x}}) - \underline{c}_{v_s(\underline{L})}^{[s,j']}\|^2 p(\underline{\mathbf{x}}) d\underline{\mathbf{x}}. \quad (6)$$

Optimal Encoding: Given codebooks $\{\mathbf{C}_j\}_{j=1}^{2^\eta}$, we optimally encode a source sample $\underline{\mathbf{x}}$ into an index vector $\underline{I}$ and index j^* using a (k,m,N,η) COm-SAPQ. This optimization is done to minimize the distortion (6). Note that there are two optimizations: one to minimize the distortion over all index vectors $\underline{Z} \in J_N^m$, and the other over indices all $j \in J_{2^\eta}$. The structure of the COm-SAPQ allows the former to be done first, followed by the latter. So first the distortion is optimized over all $\underline{Z} \in J_N^m$, this is done by the PE's:

$$\underline{I}_j = \mathrm{PE}_j(\underline{\mathbf{x}}) = \arg\min_{\underline{Z} \in J_N^m} \sum_{j'=1}^{2^\eta} \sum_{\underline{L} \in J_N^m} P(j'|j) P(\underline{L}|\underline{Z}) \sum_{s=1}^{m} \|u_s(\underline{\mathbf{x}}) - \underline{c}_{v_s(\underline{L})}^{[s,j']}\|^2. \quad (7)$$

When the source $\underline{\mathbf{x}}$ is encoded by PE_j the distortion is:

$$D_j(\underline{\mathbf{x}}) = \min_{\underline{Z} \in J_N^m} \sum_{j'=1}^{2^\eta} \sum_{\underline{L} \in J_N^m} P(j'|j) P(\underline{L}|\underline{Z}) \sum_{s=1}^{m} \|u_s(\underline{\mathbf{x}}) - \underline{c}_{v_s(\underline{L})}^{[s,j']}\|^2.$$

The optimum index j^* chooses the PE_j with the minimum distortion and the index vector $\underline{I}$: $j^* = \arg\min_{j \in J_{2^\eta}} D_j(\underline{\mathbf{x}})$ so $\underline{I} = PE_{j^*}(\underline{\mathbf{x}})$. This gives us our optimal encoding regions:

$$\mathbf{S}_{\underline{I}}^{[j^*]} = \left\{ \underline{\mathbf{x}} : j^* = \arg\min_{j \in J_{2^\eta}} D_j(\underline{\mathbf{x}}) \text{ and } \underline{I} = \underline{I}_{j^*} = \mathrm{PE}_{j^*}(\underline{\mathbf{x}}) \right\}.$$

Optimal Decoding: Assume we are given the $2^\eta N^m$ encoding regions $\{\mathbf{S}_{\underline{I}}^{[j^*]}\}$. We can obtain the centroids by taking the partial derivatives of (6) with respect to $\underline{c}_l^{[s,j']}$ and setting it to zero. Note that with a BSC, at the receiver the output of the s^{th} block $\underline{c}_l^{[s,j']}$ depends only on the s^{th} input to the channel; i.e. $v_s(\underline{I})$. Thus, the centroids are given by:

$$\underline{c}_l^{[s,j']} = \frac{\sum_{j^*=1}^{2^\eta} \sum_{i=1}^{N} P(j'|j^*) P(l|i) \int_{S_i^{[s,j^*]}} u_s(\underline{\mathbf{x}}) p(\underline{\mathbf{x}}) d\underline{\mathbf{x}}}{\sum_{j^*=1}^{2^\eta} \sum_{i=1}^{N} P(j'|j^*) P(l|i) \int_{S_i^{[s,j^*]}} p(\underline{\mathbf{x}}) d\underline{\mathbf{x}}}, \quad (8)$$

where

$$S_i^{[s,j^*]} = \bigcup_{\underline{I}: v_s(\underline{I}) = i} \mathbf{S}_{\underline{I}}^{[j^*]}. \quad (9)$$

Note that we have $2^\eta N^m$ encoding regions $\{\mathbf{S}_{\underline{I}}^{[j^*]}\}$ but only $2^\eta Nm$ partition cells $\{S_i^{[s,j^*]}\}$. Because of the structural constraint we have only $2^\eta Nm$ codewords $\{\underline{c}_l^{[s,j']}\}$, but when encoding a source sample $\underline{\mathbf{x}}$ we have a choice of $2^\eta N^m$ codewords $\{\underline{\mathbf{c}}_{\underline{L}}^{[j^*]}\}$.

The encoding regions and centroids of the CO1-SAPQ can be derived in a fashion similar to the above.

5 Encoding Simplifications

Simplifications to the encoding can be made using similar procedures as in Section IV of [7]. This preprocessing is made to simplify the calculation of $D_j(\underline{\mathbf{x}})$. For the COm-SAPQ, the simplifications are facilitated by introducing the following:

$$\underline{y}_{\gamma,j}^{[s]} = \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma)\underline{c}_l^{[s,j']} \quad \text{and} \quad \alpha_{\gamma,j}^{[s]} = \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma)\|\underline{c}_l^{[s,j']}\|^2. \tag{10}$$

This simplifies the product encoder operation (7) to be:

$$\mathrm{PE}_j(\underline{\mathbf{x}}) = \underline{I}_j = \arg\min_{\underline{I}\in J_N^m} \sum_{s=1}^{m} \alpha_{v_s(\underline{I}),j}^{[s]} - 2\langle u_s(\underline{\mathbf{x}}), \underline{y}_{v_s(\underline{I}),j}^{[s]}\rangle. \tag{11}$$

where $\langle \underline{x}, \underline{y}\rangle$ is the inner product over $\mathbb{R}^k$. Note also that by cancelling the sum of the $\|u_s(\underline{\mathbf{x}})\|^2$ over s in the expansion of $D_j(\underline{\mathbf{x}})$ we get:

$$j^* = \arg\min_j D_j(\underline{\mathbf{x}}) = \arg\min_j \left\{ \min_{\underline{I}\in J_N^m} \sum_{s=1}^{m} \alpha_{v_s(\underline{I}),j}^{[s]} - 2\langle u_s(\underline{\mathbf{x}}), \underline{y}_{v_s(\underline{I}),j}^{[s]}\rangle \right\}. \tag{12}$$

In other words, to encode $\underline{\mathbf{x}}$ by a (k,m,N,η) COm-SAPQ, there are $2^\eta Nm$ k-dimensional vectors $\underline{y}_{\gamma,j}^{[s]}$ and $2^\eta Nm$ scalars $\alpha_{\gamma,j}^{[s]}$ to be calculated prior to encoding. For the CO1-SAPQ the simplifications are similar :

$$\underline{y}_{\gamma,j} = \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma)\underline{c}_l^{[j']} \quad \text{and} \quad \alpha_{\gamma,j} = \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma)\|\underline{c}_l^{[j']}\|^2. \tag{13}$$

A (k,m,N,η) CO1-SAPQ requires $2^\eta N$ k-dimensional vector $\underline{y}_{\gamma,j}$ and $2^\eta N$ scalars $\alpha_{\gamma,j}$ to be calculated prior to encoding.

6 Design Algorithm for CO-SAPQ

The design algorithm for the COm-SAPQ is next described. The specialization to CO1-SAPQ can be easily deduced.

1. Set parameters k, m, N, η, the design BSC error crossover probability ϵ_d, the stopping threshold δ, the splitting constant k-dimensional vector $\underline{\epsilon} = (\epsilon, \ldots, \epsilon)$, the maximum number of iterations $Maxiter$, and M the total number of training vectors $\{\underline{\mathbf{x}}_f = (\underline{x}_{1,f}, \ldots, \underline{x}_{m,f})\}_{f=1}^M$. Initialize $\tau = 1$, $\rho = 0$, and the initial set of codebooks $\mathbf{C}_1^{(0)} = C_{1,1}^{(0)} \times \cdots \times C_{m,1}^{(0)}$.

2. If $\tau \geq 2^\eta$ stop; otherwise split the codebooks using $C_{s,j}^{(\rho)} = C_{s,j}^{(\rho)} - \underline{\epsilon}$ and $C_{s,j+\tau}^{(\rho)} = C_{s,j}^{(\rho)} + \underline{\epsilon}$ for $s = 1, \ldots, m$ and $j = 1, \ldots, \tau$, then increment $\tau = \tau * 2$ and set $\rho = 0$. At this point we have τ sets of codebooks $\{\mathbf{C}_j^{(\rho)}\}_{j=1}^\tau$.

3. Calculate the τNm vectors $\underline{y}_{\gamma,j'}^{[s],(\rho)}$ and values $\alpha_{\gamma,j'}^{[s],(\rho)}$ as in (10), using the codebooks $\{\mathbf{C}_j^{(\rho)}\}_{j=1}^\tau$. For each $\underline{\mathbf{x}}_f$, encode $\underline{\mathbf{x}}_f$ with each $\{\mathrm{PE}_j^{(\rho)}\}_{j=1}^\tau$ as in (11). This will give us the set of index vectors $\{\underline{I}_j\}_{j=1}^\tau$. The $\mathrm{PE}_{j^*}^{(\rho)}$ that produces the index vector $\underline{I}$ with minimum distortion is chosen using (12).

4. Once $\underline{I}$ of $\underline{\mathbf{x}}_f$ is found, $\underline{\mathbf{x}}_f$ can be put into the appropriate partition cells $S_i^{[s,j^*],(\rho)}$ in (9). Each vector $\underline{\mathbf{x}}_f$ should belong to m partition cells. The resulting distortion is:
$$D^{(\rho)}[\underline{\mathbf{x}}_f,\tau] = \sum_{j'=1}^{\tau} \sum_{\underline{L}\in J_N^m} P(j'|j^*)P(\underline{L}|\underline{I}) \sum_{s=1}^{m} \|\underline{x}_{s,f} - \underline{c}_{vs(\underline{L})}^{[s,j'],(\rho)}\|^2.$$

5. Repeat Steps 3 and 4 for $f = 1,\ldots,M$. Then calculate the centroids, using the discrete (8), and update the set of codebooks to $\{\mathbf{C}_j^{(\rho+1)}\}_{j=1}^{\tau}$ using the new centroids. Finally calculate the overall distortion: $\mathcal{D}^{(\rho)}[\tau] = \frac{1}{kmM}\sum_{f=1}^{M} D^{(\rho)}[\underline{\mathbf{x}}_f,\tau]$.

6. Check $\frac{\mathcal{D}^{(\rho-1)}[\tau]-\mathcal{D}^{(\rho)}[\tau]}{\mathcal{D}^{(\rho)}[\tau]} \leq \delta$ or $\rho \geq Maxiter$, if so then go to Step 2; otherwise set $\rho = \rho + 1$ and go to Step 3.

This algorithm assumes an initial set of codebooks $\mathbf{C}_1^{(0)}$ for the (k,m,N,η) COm-SAPQ which is obtained from a (k,m,N) COPQ designed for the same ϵ_d. For the (k,m,N,η) CO1-SAPQ the algorithm starts off with only one codebook which can be obtained from a (k,N) COVQ, again designed with the same ϵ_d.

7 Numerical Results

In this section, we present numerical results on the performance, encoding complexity and storage requirements of the COVQ, COPQ, COm-SAPQ and CO1-SAPQ. The COVQ was produced in accordance to the algorithms provided in [7].

In Tables 1 and 2, the signal-to-distortion ratio (SDR) performance of each system is provided at various rates and crossover probabilities ϵ_d for the case of a memoryless Gaussian source and a Gauss-Markov source with correlation coefficient $\rho = 0.9$, respectively. 200,000 training source samples were used. The criteria used for comparing the complexity are the encoding and storage (for both encoding and decoding) requirements. The goal is to find a channel optimized quantizer that achieves acceptable performances while maintaining low levels of encoding complexity and storage requirements, at the same rate. The encoding complexity is measured as the number of multiplications required to encode a source per scalar source input, and the storage requirement is measured as the total number of scalar values required to implement the quantizer [8]. The storage requirements include the storage of the vectors $\underline{y}_{\gamma,j}^{[s]}$ and values $\alpha_{\gamma,j}^{[s]}$ of (10) for the COm-SAPQ, and the storage of y_i and α_i of (9) and (10) in [7] for the COVQ. These values are imperative for the off-line implementation of the quantizers.

Table 1, which provides results for Gaussian memoryless sources, shows the performance of the COm-SAPQ to be comparable to that of the COVQ, within 0.2 dB, while maintaining lower encoding complexities (half that of the COVQ) and storage requirements. In Table 2, where Gauss-Markov sources are considered, we remark that the COm-SAPQ and the CO1-SAPQ of lower encoding complexities, storage requirements and dimension km , perform 0.3-1.0 dB worse than the COVQ, but still outperforms the COPQ. However, when the dimension km of the CO1-SAPQ is increased, we get an improvement in performance. For example, for R= 3.0 comparing CO1-SAPQ for $km = 4$ with COVQ with $k = 2$, we note that the CO1-SAPQ performs at least 0.3 dB better at low channel noise levels and more than 0.06 dB better at high noise levels.

However, increasing the dimension km of the CO1-SAPQ causes the encoding complexity to increase to that of the COVQ; but it still requires less storage requirements than for the COVQ.

The quantizers designed above were also tested using a validating sequence of 200,000 memoryless Gaussian and Gauss-Markov samples, and a simulated BSC with crossover probability ϵ. Perfectly matched channel conditions were assumed ($\epsilon = \epsilon_d$) in the testing and the results were found to be within 0.01-0.02 dB of those tabulated in Tables 1 and 2.

8 Conclusion

In this paper, we designed and implemented channel optimized sample adaptive product quantizers (COm-SAPQ and CO1-SAPQ) for the efficient compression and reliable transmission of Gaussian sources over BSC's. We also compared the performances of the COm-SAPQ/CO1-SAPQ schemes against those of the COVQ and the COPQ. The performance of the COm-SAPQ was found to be comparable to that of the COVQ (within 0.2 dB) when the source is memoryless Gaussian, with a reduction factor of 1/2 in encoding complexity. For Gauss-Markov sources, an improvement over the COVQ can be made by the CO1-SAPQ for the same encoding complexity, but with the advantage of lower storage requirements. Improvements in the choice of an initial codebook for the COm-SAPQ still needs to be made in order to improve its SDR for Gauss-Markov sources.

References

[1] F. Alajaji, N. Phamdo and T. Fuja, "Channel codes that exploit the residual redundancy in CELP-encoded speech," *IEEE Trans. Speech Audio Processing,* Vol. 4, pp. 325–336, Sept. 1996.

[2] E. Ayanoglu and R. M. Gray, "The design of joint source and channel trellis waveform coders," *IEEE Trans. Inform. Theory,* Vol. 33, pp. 855–865, Nov. 1987.

[3] M. Effros and P. A. Chou, "Weighted universal bit allocation: Optimal multiple quantization matrix coding," *Proc. IEEE ICASSP,* pp. 2343–2346, Detroit, May 1995.

[4] M. Effros, P. A. Chou and R. M. Gray, "Universal image compression," *IEEE Trans. Image Processing,* Vol. 8, pp. 1317–1329, Oct. 1999.

[5] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Trans. Inform. Theory,* Vol. 36, pp. 799–808, Jul. 1990.

[6] 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.

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

[8] D. S. Kim and N. B. Shroff, "Quantization based on a novel sample-adaptive product quantizer (SAPQ)," *IEEE Trans. Inform. Theory,* Vol. 45, pp. 2306–2320, Nov. 1999.

[9] D. S. Kim and N. B. Shroff, "Sample-adaptive product quantization: Asymptotic analysis and examples," *IEEE Trans. Signal Processing,* Vol. 48, pp. 2937–2947, Oct. 2000.

[10] A. Kurtenbach and C. W. Sundberg, "Quantizing for noisy channels," *IEEE Trans. Commun. Technol.,* Vol. 17, pp. 291–302, Apr. 1969.

[11] R. Laroia and N. Farvardin, "A structured fixed-rate vector quantizer derived from a variable-length scalar quantizer: Part I – Memoryless sources," *IEEE Trans. Inform. Theory,* Vol. 39, pp. 851–867, May 1993.

[12] R. Laroia and N. Farvardin, "A structured fixed-rate vector quantizer derived from a variable-length scalar quantizer: Part II – Vector sources," *IEEE Trans. Inform. Theory,* Vol. 39, pp. 868–876, May 1993.

[13] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Inform. Theory,* Vol. 28, pp. 84–95, Jan. 1980.

[14] A. Ortega and M. Vetterli, "Adaptive scalar quantization without side information," *IEEE Trans. Image Processing,* Vol. 6, pp. 665–676, May 1997.

[15] N. Phamdo, N. Farvardin and T. Moriya, "A unified approach to tree structured multi-stage vector quantization for noisy channels," *IEEE Trans. Inform. Theory,* Vol. 39, pp. 835–850, May 1993.

[16] N. Phamdo and F. Alajaji, "Soft-decision demodulation design for COVQ over white, colored and ISI Gaussian channels," *IEEE Trans. Commun.*, Vol. 48, pp. 1499–1506, Sept. 2000.

[17] M. Skoglund and P. Hedelin, "Hadamard-based soft-decoding for vector quantization over noisy channels," *IEEE Trans. Inform. Theory*, Vol. 45, pp. 515–532, Mar. 1999.

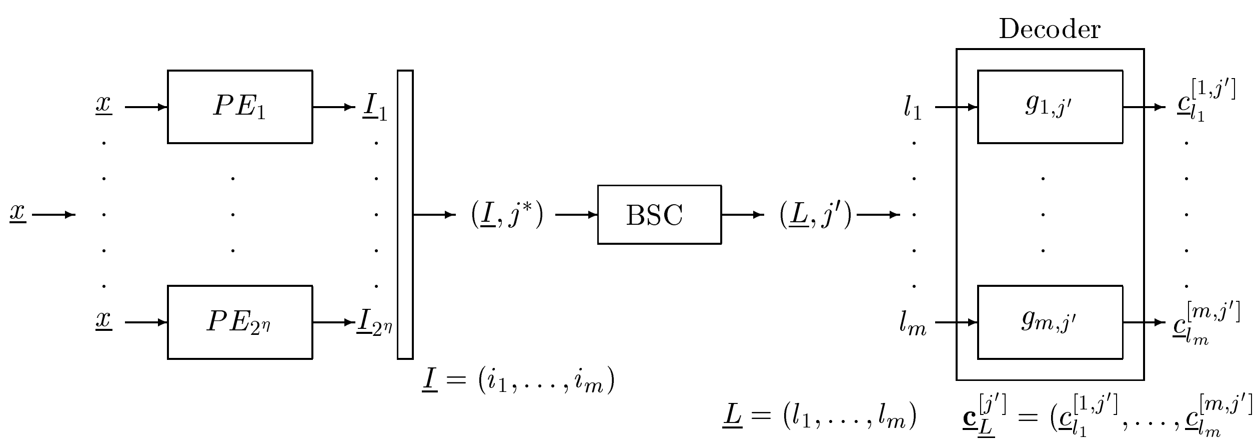

Figure 1: Channel optimized m-SAPQ model.

R	km	Quantizer	ϵ_d 0.000	0.005	0.010	0.050	0.100	0.150	complexity	storage
3.0	2	COVQ $k=2, N=64$	15.23	12.19	11.07	7.35	5.11	3.78	64	320
		COPQ $k=1, N=8$	14.57	12.00	10.47	5.62	4.65	3.48	8	48
		COm-SAPQ $k=1, N=2, \eta=4$	15.07	12.38	11.04	7.20	5.14	3.75	32	192
		CO1-SAPQ $k=1, N=2, \eta=4$	13.59	11.54	10.29	6.48	4.35	3.13	32	96
	4	CO1-SAPQ $k=1, N=4, \eta=4$	15.19	12.41	10.92	6.61	4.47	3.18	64	192
2.0	2	COVQ $k=2, N=16$	9.64	8.71	8.02	5.52	3.82	2.71	16	80
		COPQ $k=1, N=4$	9.27	8.50	7.86	4.85	3.04	1.99	4	24
		COm-SAPQ $k=1, N=2, \eta=2$	9.51	8.71	8.10	5.48	3.86	2.79	8	48
		CO1-SAPQ $k=1, N=2, \eta=2$	8.72	8.04	7.50	5.16	3.61	2.50	8	24
	3	CO1-SAPQ $k=1, N=2, \eta=3$	8.92	8.16	7.58	5.06	3.45	2.45	16	48
1.0	4	COVQ $k=4, N=16$	4.66	4.44	4.24	3.14	2.26	1.61	16	144
		COPQ $k=2, N=4$	4.38	4.16	3.96	2.72	1.75	1.14	4	40
		COm-SAPQ $k=2, N=2, \eta=2$	4.47	4.28	4.09	3.13	2.26	1.61	8	80
		CO1-SAPQ $k=2, N=2, \eta=2$	4.41	4.15	3.95	2.81	1.96	1.44	8	40
	6	CO1-SAPQ $k=2, N=2, \eta=3$	4.53	4.25	4.01	2.73	1.95	1.39	16	80

Table 1: SDR (in dB); 200,000 memoryless Gaussian training samples; Rate R in bits/source sample; ϵ_d is the design BSC crossover probability.

R	km	Quantizer	ϵ_d						complexity	storage
			0.000	0.005	0.010	0.050	0.100	0.150		
3.0	2	COVQ $k=2, N=64$	19.03	14.50	13.58	9.36	6.80	5.13	64	320
		COPQ $k=1, N=8$	14.57	12.00	10.47	5.60	4.63	3.46	8	48
		COm-SAPQ $k=1, N=2, \eta=4$	16.62	14.22	12.75	8.43	6.07	4.62	32	192
		CO1-SAPQ $k=1, N=2, \eta=4$	17.24	14.02	12.65	8.68	6.25	4.58	32	96
	4	CO1-SAPQ $k=1, N=4, \eta=4$	19.72	15.31	13.85	9.51	6.88	5.19	64	192
2.0	2	COVQ $k=2, N=16$	13.54	11.39	10.04	7.27	5.27	3.82	16	80
		COPQ $k=1, N=4$	9.28	8.50	7.85	4.83	3.02	1.96	4	24
		COm-SAPQ $k=1, N=2, \eta=2$	12.50	10.76	9.71	6.26	4.53	3.37	8	48
		CO1-SAPQ $k=1, N=2, \eta=2$	12.50	10.76	9.71	6.26	4.53	3.37	8	24
	3	CO1-SAPQ $k=1, N=2, \eta=3$	13.95	11.81	10.68	7.08	5.21	3.93	16	48
1.0	4	COVQ $k=4, N=16$	10.20	9.15	8.36	6.24	4.64	3.42	16	144
		COPQ $k=2, N=4$	7.89	4.10	3.93	2.94	2.11	1.50	4	40
		COm-SAPQ $k=2, N=2, \eta=2$	9.66	8.78	8.17	5.63	4.08	3.10	8	80
		CO1-SAPQ $k=2, N=2, \eta=2$	9.52	8.63	8.01	5.55	4.11	3.09	8	40
	6	CO1-SAPQ $k=2, N=2, \eta=3$	9.99	9.11	8.51	6.09	4.58	3.50	16	48

Table 2: SDR (in dB); 200,000 Gauss Markov training samples with correlation coefficient $\rho = 0.9$; Rate R in bits/source sample; ϵ_d is the design BSC crossover probability.