Design of Sample Adaptive Product Quantizers for Noisy Channels

Zahir Raza, Fady Alajaji, *Senior Member, IEEE*, and Tamás Linder, *Senior Member, IEEE*

***Abstract*—Channel-optimized vector quantization (COVQ) has proven to be an effective joint source-channel coding technique that makes the underlying quantizer robust to channel noise. Unfortunately, COVQ retains the high encoding complexity of the standard vector quantizer (VQ) for medium-to-high quantization dimensions and moderate-to-good channel conditions. A technique called sample adaptive product quantization (SAPQ) was recently introduced by Kim and Shroff to reduce the complexity of the VQ while achieving comparable distortions. In this letter, we generalize the design of SAPQ for the case of memoryless noisy channels by optimizing the quantizer with respect to both source and channel statistics. Numerical results demonstrate that the channel-optimized SAPQ (COSAPQ) achieves comparable performance to the COVQ (within 0.2 dB), while maintaining considerably lower encoding complexity (up to half of that of COVQ) and storage requirements. Robustness of the COSAPQ system against channel mismatch is also examined.**

***Index Terms*—Channel-optimized quantization, encoding/storage complexity, joint source-channel coding, structurally constrained vector quantization.**

I. Introduction

In RECENT works [7], [8], Kim and Shroff introduced a constrained vector quantizer (VQ) structure, which they called the sample adaptive product quantizer (SAPQ), that achieves a comparable performance to the unconstrained VQ [6] while maintaining a lower encoding complexity.[1] However, like most data-compression schemes that solely remove source redundancy, the compressed source tends to be sensitive to channel noise. The traditional approach uses tandem source-channel coding 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 jointly designed or coordinated source and channel codes (e.g., [1], [3]–[5], [9], [11], [12], [14]). A VQ-based joint source-channel coding system that exploits both source and channel statistics is called a channel-optimized vector quantizer (COVQ). COVQ has received considerable attention due to its improvement in performance over VQ in the presence of channel noise (e.g., [4] and [5]). However, COVQ incurs relatively high encoding complexity, particularly when the channel conditions are moderate or good. In this letter, we study the design of SAPQ for noisy memoryless channels, or channel-optimized SAPQ (COSAPQ), in order to find a less complex alternative to COVQ.

The rest of this letter is organized as follows. We introduce our COSAPQ system in Section II, and we establish its necessary conditions for optimal encoding and decoding in Section III. We briefly discuss the system encoding complexity and storage requirements in Section IV. In Section V, we evaluate the performance and complexity of the COSAPQ system and compare it with the COVQ and other systems. We also present performance results under mismatched channel conditions. Finally, we provide conclusions in Section VI.

II. COSAPQ System Description

The general structure of the proposed COSAPQ system, which is a generalization of SAPQ [7], [8] for noisy channels, is depicted in Fig. 1. As in the case of COVQ, the objective of this system is to convey a random source described by a probability density function (pdf) over a binary symmetric channel (BSC) with bit-error rate (BER) $\epsilon \in (0,1)$, and reproduce it at the receiver with the aim of minimizing the overall expected mean square error distortion.

For every source vector to be quantized, a (k,m,N,η) COSAPQ employs a codebook from a previously designed set of 2^η km-dimensional codebooks $\{\mathbf{C}_j\}_{j=1}^{2^\eta}$, available at both encoder and decoder. Each of the 2^η codebooks is that of a product quantizer (PQ) (e.g., see [6] and [7]), and the choice of the particular codebook used for encoding depends on the source vector. Hence, when transmitting the indexes representing the codevectors or reconstruction vectors, the encoder must also transmit an overhead index that indicates the codebook used for that source vector. Thus, the codebook of a (k,m,N,η) COSAPQ is a union of the 2^η PQ codebooks. As for the case of SAPQ (for noiseless channels) in [7] and [8], we distinguish between two types of COSAPQ: COm-SAPQ and CO1-SAPQ.

Each codebook $\mathbf{C}_j$ of COm-SAPQ is a product of m codebooks: $\mathbf{C}_j = \mathrm{C}_{1,j} \times \cdots \times \mathrm{C}_{m,j}$ where $\mathrm{C}_{s,j}$ is a collection of N k-dimensional codevectors $\{\underline{c}_l^{[s,j]}\}_{l=1}^{N}$.[2] As shown in Fig. 1, the COm-SAPQ encoder consists of 2^η vector functions called product encoders $\{\mathrm{PE}_j\}_{j=1}^{2^\eta}$. Copies of the source vector $\underline{\mathbf{x}} = (\underline{\mathrm{x}}_1, \cdots, \underline{\mathrm{x}}_m) \in \mathbb{R}^{km}$ are encoded by each PE_j producing index

Paper approved by M. Skoglund, the Editor for Source/Channel Coding of the IEEE Communications Society. Manuscript received October 17, 2003; revised November 7, 2004. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada. This paper was presented in part at the 39th Annual Allerton Conference on Communication, Control and Computing, Monticello, IL, October 2001.

Z. Raza is with T-Mobile, Bellevue, WA 98006 USA (e-mail: Zahir.Raza@T-Mobile.com).

F. Alajaji is with the Department of Mathematics and Statistics, Queen's University, Kingston, ON K7L 3N6 Canada (e-mail: fady@mast.queensu.ca).

T. Linder is with the Department of Mathematics and Statistics, Queen's University, Kingston, ON K7L 3N6 Canada and also with the Informatics Laboratory, Computer and Automation Research Institute of the Hungarian Academy of Sciences, Budapest, Hungary (e-mail: linder@mast.queensu.ca).

Digital Object Identifier 10.1109/TCOMM.2005.844938

[1]For previous related work on adaptive quantization, see [2], [10], and the references in [7] and [8].

[2]In the original definition of m-SAPQ [7], the codebooks $\mathrm{C}_{s,j}$ are allowed to have different sizes n_s'. Our formulation introduces the restriction that $n_s' = N$ for all $s = 1, \ldots, m$.

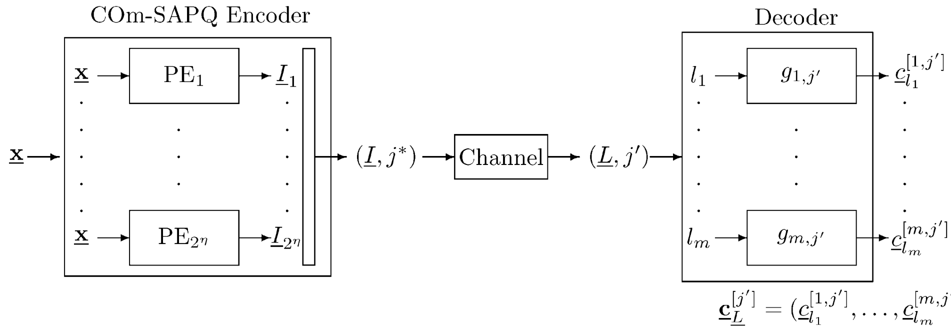

Fig. 1. Block diagram of a (k,m,N,η) COm-SAPQ system, where $\underline{\mathbf{x}} = (\underline{\mathrm{x}}_1,\ldots,\underline{\mathrm{x}}_m) \in \mathbb{R}^{km}$, $\underline{I} = (i_1,\ldots,i_m) \in J_N^m$, $\underline{L} = (l_1,\ldots,l_m) \in J_N^m$, and j^*, $j' \in J_{2^\eta}$.

vector $\underline{I}_j : \mathrm{PE}_j(\underline{\mathbf{x}}) = \underline{I}_j \in J_N^m = \{1,\ldots,N\}^m$. Each index vector in the set $\{\underline{I}_j\}_{j=1}^{2^\eta}$ has an associated distortion, and the index vector $\underline{I}$ with minimum distortion is transmitted over the channel along with index j^*, representing PE_{j^*} that produced $\underline{I}$. Due to channel noise, potentially corrupted versions of $\underline{I}$ and j, $\underline{L} = (l_1,\ldots,l_m) \in J_N^m$ and $j' \in J_{2^\eta} = \{1,\cdots,2^\eta\}$, are received at the decoder. The decoder decodes $\underline{L}$ using decoding function $G_{j'}$ from the set $\{\mathrm{G}_j\}_{j=1}^{2^\eta}$ as follows: $\mathrm{G}_{j'}(\underline{L}) = (g_{1,j'}(l_1), g_{2,j'}(l_2),\cdots,g_{m,j'}(l_m)) = (\underline{c}_{l_1}^{[1,j']},\ldots,\underline{c}_{l_m}^{[m,j']}) = \underline{\mathbf{c}}_{\underline{L}}^{[j']}$, where $\underline{c}_l^{[s,j']} \in \mathrm{C}_{s,j'}$ for $l \in J_N$, and $\underline{\mathbf{c}}_{\underline{L}}^{[j']} \in \mathbf{C}_{j'}$.

The CO1-SAPQ is a simplified version of COm-SAPQ in that each codebook $\mathbf{C}_j$ is an m-fold product of the same codebook: $\mathbf{C}_j = \mathrm{C}_j \times \cdots \times \mathrm{C}_j$. The rate of both the (k,m,N,η) CO1-SAPQ and the (k,m,N,η) COm-SAPQ is given by

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

III. Optimality Conditions for System Design

We next establish the necessary encoding and decoding conditions for optimality that can be used in an iterative design algorithm (as the generalized Lloyd–Max algorithm in COVQ) that attempts to minimize the overall distortion. Let $\mathbf{S}_{\underline{Z}}^{[j]}$ be the encoding region for index vector $\underline{Z}$ and index $j : \mathbf{S}_{\underline{Z}}^{[j]} = \{\underline{\mathbf{x}} : \text{encoder}(\underline{\mathbf{x}}) = (\underline{Z}, j)\}$. Then the average mean-squared end-to-end distortion can be written as

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

where $p(\underline{\mathbf{x}})$ is the source pdf with $\underline{\mathbf{x}} \in \mathbb{R}^{km}$, $P(a|b)$ is the channel transition probability of receiving index a given that index b was sent (which can be easily expressed in terms of the BSC BER ϵ, assuming that each index is sent using a natural binary assignment), $\mathrm{v}_s(\underline{I}) =$ the sth index component of $\underline{I}$ ($\mathrm{v}_s : J_N^m \to J_N$), and $\mathrm{u}_s(\underline{\mathbf{x}}) = \underline{\mathrm{x}}_s$; $s = 1,\ldots,m$ ($\underline{\mathrm{u}}_s : \mathbb{R}^{km} \to \mathbb{R}^k$).

A. Optimal Encoding

Given that codebooks $\{\mathbf{C}_j\}_{j=1}^{2^\eta}$ are fixed, the optimal encoding of a source sample $\underline{\mathbf{x}}$ into $(\underline{I}, j^*)$ is achieved via two optimization steps: one to minimize the distortion over all index vectors $\underline{Z} \in J_N^m$, and the other over all indexes $j \in J_{2^\eta}$. The first optimization over all $\underline{Z} \in J_N^m$ is obtained via the PEs

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

When the source $\underline{\mathbf{x}}$ is encoded by PE_j into $\underline{I}_j$, the associated distortion is given by

$$D_j(\underline{\mathbf{x}}) = \min_{\underline{Z}\in J_N^m} \sum_{j'=1}^{2^\eta} \sum_{\underline{L}\in J_N^m} P(j'|j)P(\underline{L}|\underline{Z}) \sum_{s=1}^{m} \left\| u_s(\underline{\mathbf{x}}) - \underline{c}_{v_s(\underline{L})}^{[s,j']} \right\|^2. \quad (3)$$

The optimum index j^* then chooses the index $\underline{I}$ with the smallest associated distortion, resulting in the following optimal encoding region:

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

Encoding simplifications can be obtained using a similar approach as in [4, Sec. IV]. If we define

$$\begin{aligned} \underline{y}_{s,j}(\gamma) &= \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma)\underline{c}_l^{[s,j']} \\ \alpha_{s,j}(\gamma) &= \sum_{j'=1}^{2^\eta} \sum_{l=1}^{N} P(j'|j)P(l|\gamma) \left\| \underline{c}_l^{[s,j']} \right\|^2 \end{aligned} \quad (4)$$

then the PE operation (2) can be written in the following simpler expression:

$$\begin{aligned} \mathrm{PE}_j(\underline{\mathbf{x}}) &= \underline{I}_j \\ &= \arg\min_{\underline{I}\in J_N^m} \sum_{s=1}^{m} \alpha_{s,j}\left(v_s(\underline{I})\right) - 2\left\langle u_s(\underline{\mathbf{x}}), \underline{y}_{s,j}\left(v_s(\underline{I})\right) \right\rangle \end{aligned} \quad (5)$$

where $\langle \underline{\mathrm{x}}, \underline{\mathrm{y}} \rangle$ is the standard inner product in $\mathbb{R}^k$. In other words, the encoding of $\underline{\mathbf{x}}$ by a COm-SAPQ requires the prior calculation of $2^\eta mN$ k-dimensional vectors $\underline{y}_{\gamma,j}^{[s]}$ and $2^\eta mN$ scalars $\alpha_{\gamma,j}^{[s]}$.

TABLE I

SDR PERFORMANCE IN DECIBELS, ENCODING COMPLEXITY (IN PARENTHESES) AND STORAGE REQUIREMENT OF COVQ, COPQ, COm-SAPQ, AND CO1-SAPQ WITH RATE R AND DIMENSION km. ALL SYSTEMS WERE DESIGNED FOR A BSC WITH BER ϵ_d USING 200 000 MEMORYLESS GAUSSIAN TRAINING SAMPLES

			ϵ_d						
R	km	Quantizer	0.000	0.005	0.010	0.050	0.100	0.150	storage
3.0	2	COVQ	15.23	12.19	11.07	7.35	5.11	3.78	320
		$k=2$, $N=64$	(64)	(64)	(57)	(30)	(27)	(25)	
	2	COPQ	14.57	12.00	10.47	5.62	4.65	3.48	48
		$k=1$, $N=8$	(8)	(8)	(8)	(6)	(6)	(6)	
	2	COm-SAPQ	15.07	12.38	11.04	7.20	5.14	3.75	192
		$k=1$, $N=2$, $\eta=4$	(32)	(32)	(32)	(23)	(16)	(14)	
	2	CO1-SAPQ	13.59	11.54	10.29	6.48	4.35	3.13	96
		$k=1$, $N=2$, $\eta=4$	(32)	(32)	(32)	(22)	(18)	(12)	
	4	CO1-SAPQ	15.19	12.41	10.92	6.61	4.47	3.18	192
		$k=1$, $N=4$, $\eta=4$	(64)	(64)	(52)	(30)	(26)	(17)	
2.0	2	COVQ	9.64	8.71	8.02	5.52	3.82	2.71	80
		$k=2$, $N=16$	(16)	(16)	(16)	(13)	(13)	(13)	
	2	COPQ	9.27	8.50	7.86	4.85	3.04	1.99	24
		$k=1$, $N=4$	(4)	(4)	(4)	(4)	(4)	(4)	
	2	COm-SAPQ	9.51	8.71	8.10	5.48	3.86	2.79	48
		$k=1$, $N=2$, $\eta=2$	(8)	(8)	(8)	(8)	(8)	(8)	
	2	CO1-SAPQ	8.72	8.04	7.50	5.16	3.61	2.50	24
		$k=1$, $N=2$, $\eta=2$	(8)	(8)	(8)	(8)	(8)	(8)	
	3	CO1-SAPQ	8.92	8.16	7.58	5.06	3.45	2.45	48
		$k=1$, $N=2$, $\eta=3$	(16)	(16)	(16)	(16)	(16)	(16)	
1.0	4	COVQ	4.66	4.44	4.24	3.14	2.26	1.61	144
		$k=4$, $N=16$	(16)	(16)	(16)	(16)	(16)	(16)	
	4	COPQ	4.38	4.16	3.96	2.72	1.75	1.14	40
		$k=2$, $N=4$	(4)	(4)	(4)	(4)	(4)	(4)	
	4	COm-SAPQ	4.47	4.28	4.09	3.13	2.26	1.61	80
		$k=2$, $N=2$, $\eta=2$	(8)	(8)	(8)	(8)	(8)	(8)	
	4	CO1-SAPQ	4.41	4.15	3.95	2.81	1.96	1.44	40
		$k=2$, $N=2$, $\eta=2$	(8)	(8)	(8)	(8)	(8)	(8)	
	6	CO1-SAPQ	4.53	4.25	4.01	2.73	1.95	1.39	80
		$k=2$, $N=2$, $\eta=3$	(16)	(16)	(16)	(16)	(16)	(16)	

B. Optimal Decoding

Given that the $2^\eta N^m$ encoding regions $\{\mathbf{S}_{\underline{I}}^{[j^*]}\}$ are fixed, we can determine the optimal centroids via standard minimum mean-squared estimation based on the distortion expression in (1). This results in the following centroids:

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

$$\text{where}\quad S_i^{[s,j^*]} = \bigcup_{\underline{I}:v_s(\underline{I})=i} \mathbf{S}_{\underline{I}}^{[j^*]}. \quad (6)$$

IV. ENCODING COMPLEXITY AND STORAGE REQUIREMENT

We next briefly discuss the encoding complexity and the storage requirement of a (k,m,N,η) COm-SAPQ system.

A. Encoding Complexity

As in [7], [8], and [13], the encoding complexity is defined as the number of multiplications needed to encode one source sample. It is important to point out that unlike conventional VQ systems (designed for noiseless channels), the encoding complexity of VQ schemes for noisy channels (such as COVQ and COSAPQ) is strongly dependent on the design channel BER. Indeed, when the design BER is very high, the encoding complexity of such channel-optimized schemes can be considerably low, since a large number of the encoding cells become empty (e.g., see [5]). Thus, we evaluate the COSAPQ encoding complexity in terms of the number of nonempty encoding cells. More specifically, the encoding complexity of a (k,m,N,η) COm-SAPQ is given by

$$\sum_{j=1}^{2^\eta}\sum_{s=1}^{m}\frac{kN_{sj}}{km} = \sum_{j=1}^{2^\eta}\sum_{s=1}^{m}\frac{N_{sj}}{m} \quad (7)$$

where N_{sj} denotes the number of nonempty encoding cells in codebook $\mathrm{C}_{s,j}$, which consists of N k-dimensional codevectors. Similarly, the encoding complexity of a (k,m,N,η) CO1-SAPQ is given by

$$\sum_{j=1}^{2^\eta}\frac{kmN_j}{km} = \sum_{j=1}^{2^\eta} N_j \quad (8)$$

where N_j denotes the number of nonempty encoding cells in codebook C_j.

B. Storage Requirement

The storage requirement is defined as the total number of scalars needed to implement the quantizer operations in both encoding and decoding stages [7], [8], [13]. At the encoder, we need to store the sets $\{\underline{y}^{[s]}_{\gamma,j}\}$ and $\{\alpha^{[s]}_{\gamma,j}\}$ given in (4). We need $k2^{\eta}mN$ scalars for $\{\underline{y}^{[s]}_{\gamma,j}\}$ and $2^{\eta}mN$ scalars for $\{\alpha^{[s]}_{\gamma,j}\}$. Hence, at the encoder, we require $k2^{\eta}mN + 2^{\eta}mN$ scalars in total. At the decoder, we need to store the entire COSAPQ codebook, which requires $k2^{\eta}mN$ scalars. Therefore, the total storage requirement for a (k, m, N, η) COm-SAPQ is given by

$$2k2^{\eta}mN + 2^{\eta}mN. \tag{9}$$

For a (k, m, N, η) CO1-SAPQ, the total storage requirement is analogously given by

$$2k2^{\eta}N + 2^{\eta}N. \tag{10}$$

V. Numerical Results

We next evaluate the performance in terms of signal-to-distortion ratio (SDR), encoding complexity [via (7) and (8)] and storage requirement [via (9) and (10)] of our COSAPQ system for the compression and transmission of two (unit variance) sources: a memoryless Gaussian source and a Gauss–Markov source with correlation coefficient 0.9. Comparisons with the following competing quantization systems are also provided: COVQ and channel-optimized product quantizer (COPQ).[3] For a (k, N) COVQ, the encoding complexity is given by the number of nonempty encoding cells N^*, and the storage requirement is given by $2kN + N$. The encoding complexity of a (k, m, N) COPQ is given by $(1/m)\sum_{i=1}^{m} N_i$, where N_i is the number of nonempty encoding cells in the ith codebook, while its storage requirement is given by $2kmN + mN$.

The optimality conditions derived in Section III were used in a generalized Lloyd–Max algorithm to design the COm-SAPQ and CO1-SAPQ codes. Details of the algorithm are available in [13]. For the COm-SAPQ system, the algorithm was initialized by a COPQ codebook with the same design parameters, while the CO1-SAPQ system was initialized with a COVQ codebook. We employed 200 000 training samples for the codes design and 200 000 test samples for the simulations. In all cases, the simulation and training results were in agreement.

Table I shows the performance (both SDR, encoding complexity and storage) of COVQ, COPQ, COm-SAPQ, and CO1-SAPQ at various rates R and design channel BERs ϵ_d for memoryless Gaussian sources. We remark that COm-SAPQ performs within 0.2 dB of COVQ of the same rate, while enjoying an encoding complexity that is 23% to 50% smaller than that of COVQ (note that for $R = 1$, the encoding complexity reduction is by 50% for all design BERs) and a storage requirement that is lower by a factor of 40% to 44%. Furthermore, it is observed in [13] that for Gauss–Markov sources, a gain of up to 0.7 dB over COVQ can be attained by a CO1-SAPQ with the same complexity, while keeping a lower storage requirement (by 40% to 67%).[4] This gain is due to the fact that the CO1-SAPQ can employ a higher dimension km while keeping the same encoding complexity as COVQ. Table II compares channel-optimized quantizers with quantizers designed for noiseless channels. The Linde, Buzo and Gray VQ (LBGVQ) system is the same as in [5] that employs simulated annealing for index assignment. We note that COSAPQ offers significant gains over SAPQ, particularly at high BERs.

So far, we assumed that the quantizers' encoder and decoder are perfectly matched to the channel conditions; however, in many practical situations, the channel characteristics may be time varying. We next test the robustness of both COSAPQ and COVQ systems when they are designed for a fixed channel BER ϵ_d, while the actual or true BER is $\epsilon_a \neq \epsilon_d$. Mismatch simulation results are presented in Fig. 2 for $R = 2.0$, $\epsilon_d = 0.05$, and

TABLE II
SDR Performance in Decibels of COVQ, COPQ, COm-SAPQ, and CO1-SAPQ Versus LBGVQ (With Simulated Annealing), PQ, m-SAPQ, and 1-SAPQ With Rate $R = 3.0$, Using 200 000 Gauss–Markov Testing Samples (With Correlation Coefficient 0.9) and a Simulated BSC With Actual BER ϵ_a (Here, the Design BER ϵ_d Is Matched to ϵ_a; i.e., $\epsilon_d = \epsilon_a$)

	ϵ_a					
Quantizer	0.000	0.005	0.010	0.050	0.100	0.150
LBGVQ (+ sim. annl.) $k=2, N=64$	19.01	13.63	11.07	5.29	2.37	0.71
PQ $k=1, m=2, N=8$	14.64	12.01	10.39	5.01	2.25	0.57
m-SAPQ $k=1, m=2, N=2, \eta=4$	16.65	12.82	10.74	4.85	2.09	0.48
1-SAPQ $k=1, m=4, N=4, \eta=4$	19.71	13.42	10.98	4.55	1.69	0.05
COVQ $k=2, N=64$	19.01	14.62	13.62	9.48	6.85	5.20
COPQ $k=1, m=2, N=8$	14.60	12.02	10.50	5.57	4.60	3.45
CO-m-SAPQ $k=1, m=2, N=2, \eta=4$	16.64	14.30	12.75	8.45	6.10	4.66
CO-1-SAPQ $k=1, m=4, N=4, \eta=4$	19.71	15.27	13.94	9.52	6.91	5.21

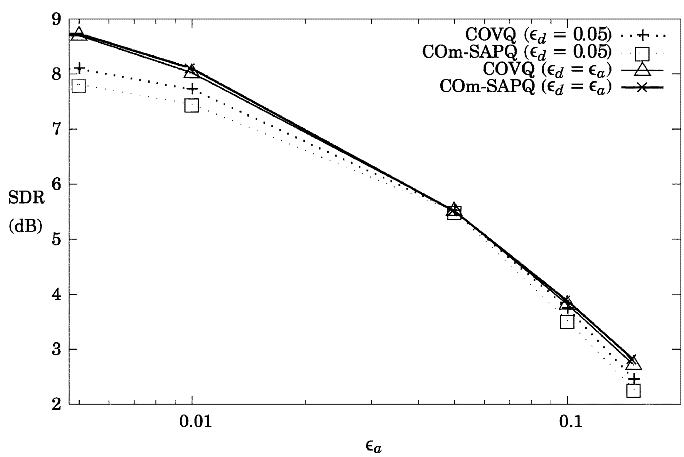

Fig. 2. SDR performance in decibels of (2,16) COVQ and (1,2,2,2) COm-SAPQ at rate $R = 2.0$ and BSC design BER $\epsilon_d = 0.05$, using 200 000 memoryless Gaussian testing samples and a simulated BSC with actual BER ϵ_a.

[3]A (k, m, N) COPQ is a (k, m, N) PQ optimized for noisy channels; its overall codebook is a product of m codebooks, each consisting of N k-dimensional codevectors. Refer to [13] for a detailed description of this system and its design algorithm.

[4]Compare, for example, (2,64) COVQ with (1,4,4,4) CO1-SAPQ in Table II for low values of BERs.

memoryless Gaussian sources. We observe that COm-SAPQ performs within 0.3 dB from COVQ for $\epsilon_a \leq 0.01$; as ϵ_a increases, COm-SAPQ is as robust as COVQ. For Gauss–Markov sources, gains of up to 0.2 dB were attained by CO1-SAPQ over COVQ under mismatch BER conditions [13].[5]

VI. Conclusion

We introduced COSAPQ systems (COm-SAPQ and CO1-SAPQ) for the efficient compression and reliable transmission of Gaussian sources over BSCs. Numerical results show that for memoryless Gaussian sources, performance within 0.2 dB of that of COVQ can be attained by COm-SAPQ of the same rate with up to half the encoding complexity and lower storage requirements. For Gauss–Markov sources, CO1-SAPQ with dimension km outperforms COVQ of dimension k by up to 0.7 dB for the same rate and encoding complexity, while having a lower storage requirement. Finally, COSAPQ was shown to be as robust as COVQ under mismatched channel conditions.

Acknowledgment

The authors are grateful to the editor, the anonymous reviewers, and F. Behnamfar for their constructive and helpful comments.

[5]Additional mismatch results for various rates and design BERs are available in [13].

References

[1] E. Ayanoglu and R. M. Gray, "The design of joint source and channel trellis waveform coders," *IEEE Trans. Inf. Theory*, vol. IT-33, pp. 855–865, Nov. 1987.

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

[3] J. G. Dunham and R. M. Gray, "Joint source and noisy channel trellis encoding," *IEEE Trans. Inf. Theory*, vol. IT-27, pp. 516–519, Jul. 1981.

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

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

[6] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1992.

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

[8] ——, "Sample-adaptive product quantization: Asymptotic analysis and examples," *IEEE Trans. Signal Process.*, vol. 48, pp. 2937–2947, Oct. 2000.

[9] H. Kumazawa, M. Kasahara, and T. Namekawa, "A construction of vector quantizers for noisy channels," *Electron. Eng. Japan*, vol. 67-B, pp. 39–47, Jan. 1984.

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

[11] N. Phamdo, F. Alajaji, and N. Farvardin, "Quantization of memoryless and Gauss–Markov sources over binary Markov channels," *IEEE Trans. Commun.*, vol. 45, pp. 668–675, Jun. 1997.

[12] 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, Sep. 2000.

[13] Z. Raza, "Sample adaptive product quantization for memoryless noisy channels," Master's thesis, Dept. Math. Statist., Queen's Univ., Kingston, ON, Canada, Nov. 2002.

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