Transactions Letters

Robust Vector Quantizer Design by Noisy Channel Relaxation

Shrinivas Gadkari, *Member, IEEE,* and Kenneth Rose, *Member, IEEE*

***Abstract*— This letter proposes a method to design a vector quantizer (VQ) for robust performance under noisy channel conditions. By re-optimizing the quantizer at progressively lower levels of assumed channel noise, the design is less susceptible to poor local optima. The method is applied to: 1) channel-optimized VQ design; and 2) index assignment for a source-optimized VQ. For both problems, we demonstrate substantial performance improvements over commonly used techniques.**

***Index Terms*—Index assignment, source-channel coding, vector quantizer.**

I. Introduction

THIS work is concerned with vector quantizer (VQ) design for transmission over noisy channels (NC's) and, in particular, binary symmetric channels. This channel model is useful in communication scenarios where a binary modulation scheme is used to transmit data over a Gaussian channel. For example, a wireless communication channel can be modeled as a Gaussian channel with time-varying signal-to-noise ratio (SNR). If binary modulation is used, the channel may be conveniently modeled as a binary symmetric channel with time-varying bit-error rate (BER).

Two main approaches exist for the problem of robust VQ design: channel-optimized VQ (COVQ) [1], [4] and index assignment (IA) [2]–[4]. COVQ directly optimizes the VQ for a specific channel condition. While it may provide optimal performance at the prespecified channel condition, its performance is suboptimal under (almost) clean channel conditions, which typically prevail most of the time in deep fading wireless channels. IA employs a source-optimized VQ (VQ designed assuming a noiseless channel), followed by a judicious assignment of binary indexes to codevectors. It offers uncompromised performance under clean channel conditions, but underperforms COVQ at the prescribed channel noise.

The common design algorithm for COVQ is the "noisy channel generalized Lloyd algorithm" (NC-GLA) [1], [4]. A known IA design method is the binary switching algorithm (BSA) [2]. Both NC-GLA and BSA are local descent algorithms and suffer from the problem of shallow local minimum traps. This problem is the main motivation for the proposed noisy channel relaxation (NCR) approach, which is applicable to the design of both COVQ and IA. Experimental results demonstrate the superior performance of the resulting robust VQ's, which is obtained at the cost of a modest increase in design complexity.

The main ideas described in this letter were presented in part in [5]. Historical credit is due to an earlier independent work [7] which included a suggestion to vary the level of channel noise during VQ design by self-organizing feature maps.

II. Channel-Optimized VQ Design

This section briefly reviews the COVQ structure and basic design approach. The problem is to design a COVQ of n bits per vector given a vector source represented by a training set $\mathcal{T} = \{\boldsymbol{x}\} \subset \mathcal{R}^k$ and a binary symmetric channel with BER ϵ_{ch}. It is convenient to decompose the problem into two parts: encoder and decoder design.

The COVQ encoder is specified by a partition of the input space $\mathcal{R}^k$ into disjoint encoding regions $R_0, R_1, \cdots, R_{2^n-1}$. Given source vector $\boldsymbol{x}$, index $I(\boldsymbol{x})$ is transmitted and indicates the encoding region, i.e., $\boldsymbol{x} \in R_{I(\boldsymbol{x})}$. The decoder receives some index J and produces the corresponding codevector y_J from its codebook C. The overall squared error distortion for the training set is

$$D = \frac{1}{|\mathcal{T}|} \sum_{\boldsymbol{x}\in\mathcal{T}} \sum_{J} p_{J|I(\boldsymbol{x})} \|\boldsymbol{x} - \boldsymbol{y}_J\|^2 \tag{1}$$

where $p_{J|I}$ denotes the probability of receiving index J when index I is transmitted, and $|\mathcal{T}|$ is the size of $\mathcal{T}$.

Given a fixed encoder $I(\boldsymbol{x})$, the optimal decoder is given by the NC centroid rule which specifies the codevectors

$$\boldsymbol{y}_J = \frac{\displaystyle\sum_{\boldsymbol{x}\in\mathcal{T}} \boldsymbol{x} p_{J|I(\boldsymbol{x})}}{\displaystyle\sum_{\boldsymbol{x}\in\mathcal{T}} p_{J|I(\boldsymbol{x})}}. \tag{2}$$

Paper approved by B. Girod, the Editor for Image Processing of the IEEE Communications Society. Manuscript received June 2, 1997; revised March 22, 1998 and January 11, 1999. This work was supported in part by the National Science Foundation under Grant NCR-9314335, the University of California MICRO Program, ACT Networks, Inc., Advanced Computer Communications, Cisco Systems, Inc., DSP Group, Inc., DSP Software Engineering, Inc., Fujitsu Laboratories of America, Inc., General Electric Company, Hughes Electronics Corporation, Intel Corporation, Nokia Mobile Phones, Qualcomm, Inc., Rockwell International Corporation, and Texas Instruments Incorporated. This paper was presented in part at the IEEE International Conference on Acoustics, Speech, and Signal Processing, Atlanta, GA, May 1996.

S. Gadkari is with Conexant Systems, Inc., Newport Beach, CA 92658-8902 USA.

K. Rose is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106-9560 USA (e-mail: rose@ece.ucsb.edu).

Publisher Item Identifier S 0090-6778(99)06285-6.

Given a fixed decoder codebook C, the optimal encoder is given by the NC nearest neighbor rule (see [6], [8])

$$I(\boldsymbol{x}) = \arg\min_{L} \{d(\boldsymbol{x}, \boldsymbol{u}_L) + \theta_L^2\} \tag{3}$$

where $\boldsymbol{u}_L = \Sigma_J \; p_{J|L}\boldsymbol{y}_J$, $\theta_L^2 = \{\Sigma_J \; p_{J|L}\|\boldsymbol{y}_J\|^2\} - \|\boldsymbol{u}_L\|^2$, and where $d(\cdot, \cdot)$ denotes the squared Euclidean distance. A conceptual simplification of the COVQ encoding rule (3) is obtained by considering the $k+1$-dimensional "augmented" vectors $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, 0)$ and prototypes $\tilde{\boldsymbol{u}}_I = (\boldsymbol{u}_I, \theta_I)$. The encoding rule (3) is equivalent to the familiar nearest neighbor encoding rule in the "augmented" space

$$I(\boldsymbol{x}) = I(\tilde{\boldsymbol{x}}) = \arg\min_{L} \{d(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{u}}_L)\}. \tag{4}$$

Let the ith component of $\tilde{\boldsymbol{x}}$ be denoted by $\tilde{x}_i$. It is easy to see that the source vectors lie in the hyperplane $\tilde{x}_{k+1} = 0$. Since (4) is the ordinary nearest neighbor rule, the prototypes $\{\tilde{\boldsymbol{u}}_I\}$ define a Voronoi partition of the augmented space. The intersection of the Voronoi cells with the hyperplane $\tilde{x}_{k+1} = 0$ determines the encoding regions $\{R_i\}$. It is possible that not all the Voronoi regions intersect the hyperplane $\tilde{x}_{d+1} = 0$. If the Voronoi region corresponding to index L does not intersect the hyperplane $\tilde{x}_{k+1} = 0$, then the encoding region R_L is empty, and index L is never transmitted. The system is, in effect, performing some form of "error correction" by not transmitting certain indexes. Also note that the encoding complexity of the COVQ is at most equal to the complexity of a full-search $k+1$-dimensional VQ [6]. We note that the geometric properties of the encoder partition are not only theoretically appealing but useful, as the number of nonempty encoding regions can be used to effectively control the relaxation schedule for our design method.

The NC-GLA method for COVQ design involves alternating between application of the NC nearest neighbor rule (3) to repartition the training set $\mathcal{T}$ and the NC centroid rule (2) to update the entries of the codebook C.

III. Noisy Channel Relaxation

At low BER ϵ_{ch}, the probability of multiple bit errors per index is negligible. Hence, the transition probability $p_{J|I}$ is only significant for a small Hamming neighborhood about index I. This implies that during optimization, codevectors only have very localized influence and, hence, a poor indexing initialization almost invariably leads to a poor local minimum. It is, however, different in the case of high values of ϵ_{ch}, where parameters tend to have a global influence that enables rearranging the codevectors according to their indexes, thereby matching proximity of codewords (in the Hamming sense) with proximity of codevectors (in the Euclidean sense). In other words, at high values of the channel noise, the COVQ design solution is much less sensitive to initialization. This is supported by experimental data which shows that at high values of ϵ_{ch}, different initializations produce very similar solutions, and the effect of poor initializations is largely mitigated. Fig. 1 shows a typical example of the similar solutions produced by different initializations when the channel is very noisy. In fact, the two depicted solutions correspond to an equivalent "error correcting code" as all the codewords of one solution can be obtained from the other by adding the fixed word (0011). Note that such a shift conserves all Hamming distances.

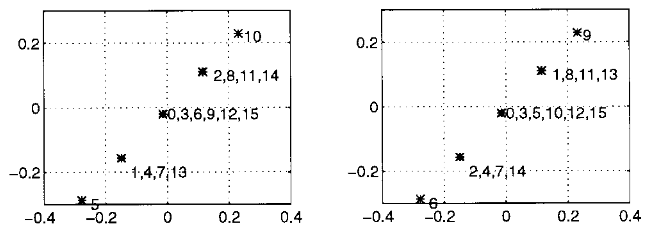

Fig. 1. This plot depicts the codevectors along with their binary code words (represented as integers) obtained for two different initializations using NC-GLA for a binary symmetric channel with transition probability $\epsilon_{\text{ch}} = 0.4$. Note the striking similarity in the two solutions due to high value of the channel BER. The two-dimensional training data was generated by a Gauss–Markov source with correlation coefficient of 0.9.

TABLE I
A Comparison of Results Obtained Using NCR and NC-GLA. The Source Is Gauss–Markov with Correlation Coefficient ρ and the Codebook Size Is 256. The Values Denote the SNR Calculated on a Test Set for the Corresponding COVQ

ϵ_{ch}	dim $=4$, $\rho=0.0$		dim $=4$, $\rho=0.8$	
	NC-GLA	NCR	NC-GLA	NCR
0.05	5.8	5.8	7.6	7.9
0.03	6.8	6.9	8.9	9.1
0.02	7.3	7.6	9.6	9.9
0.01	7.6	8.5	10.2	10.9
0.005	8.4	9.2	11.1	11.7
0.001	9.6	9.8	12.8	12.9
	dim $=8$, $\rho=0.0$		dim $=8$, $\rho=0.8$	
	NC-GLA	NCR	NC-GLA	NCR
0.05	3.2	3.2	5.9	5.9
0.03	3.6	3.8	6.7	6.8
0.02	3.8	4.1	7.0	7.3
0.01	4.1	4.6	7.6	8.0
0.005	4.5	4.9	7.9	8.4
0.001	5.0	5.1	8.7	8.9

We use the observation to avoid many shallow local minima that trap the VQ design procedure. Instead of applying the NC-GLA with the prescribed value of channel BER ϵ_{ch}, we start the iterations at a very high level of channel BER ϵ and gradually reduce the value of ϵ to ϵ_{ch}. The solutions obtained at high ϵ, which are less sensitive to initialization, act as effective initial conditions for iterations with lower values of ϵ. We refer to such a gradual reduction in the value of ϵ as noisy channel relaxation. Observe that noisy channel relaxation (NCR) is similar in spirit to the idea of deterministic annealing [9] and most particularly in its application to COVQ design [10]. However, NCR is not annealing in a strict sense as stochastic equilibrium is not maintained.

In Table I, we compare the performance of COVQ design by NC-GLA with that obtained by NCR for the case of Gauss–Markov sources. NC-GLA was initialized using the standard splitting initialization of GLA [11]. A VQ designed with this initialization is known to be naturally robust to channel errors [4]. To eliminate the effect of arbitrary or random initialization, NCR was also initialized with the same splitting initialization. The early iterations were performed

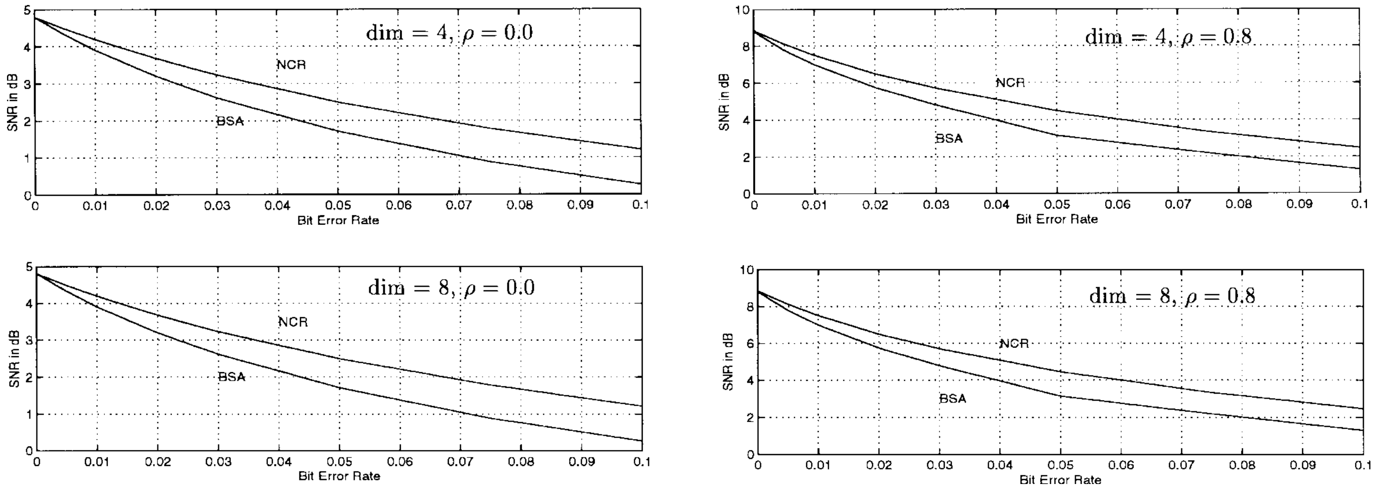

Fig. 2. A comparison of the IA performance obtained from BSA and NCR. The results are given for a Gauss–Markov source with correlation coefficient ρ. The VQ codebook size is 256.

with a value of $\epsilon = 0.45$. At high values of ϵ, the number of nonempty encoding regions is significantly smaller than the target size of the VQ. We use the number of nonempty encoding regions to control the relaxation schedule. If the number of nonempty encoding regions in the mth iteration increases by more than four over that in iteration $m-2$, we use $\epsilon_{m+1} = \epsilon_m$, else we use $\epsilon_{m+1} = (\epsilon_m/1.05)$. When ϵ_m reaches the target (prescribed) channel BER ϵ_{ch}, the procedure is stopped. For both the NC-GLA and NCR designs, we used training sets and test sets of size 24 000 each. Note that the precise heuristic of the relaxation schedule is only given here for completeness. In fact, the method is relatively insensitive to the choice of schedule.

The results demonstrate that COVQ design by NCR achieves performance gains of up to 0.9 dB over standard NC-GLA. It should be noted that these improvements in performance were achieved at the cost of manageable increase in computational complexity. More precisely, the additional computation comes from the extra iterations required by NCR as the assumed channel noise is lowered. In our experiments, the NCR complexity was two to three times that of standard NC-GLA.

As an alternative to NCR, one may employ a very computation-intensive technique such as simulated annealing to determine an IA initialization for the NC-GLA algorithm. Our purpose here is to demonstrate that substantial performance gains can be achieved by NCR without recourse to the excessive computational complexity of simulated annealing.

IV. Index Assignment

Since the COVQ is optimized for a specific noisy channel (NC) condition, its performance under clean channel conditions is suboptimal. This may be a considerable disadvantage in applications where clean channel conditions prevail for a large fraction of the time. IA is an alternate NC VQ technique which avoids this performance loss. A source-optimized VQ, which is designed for operation under clean channel conditions, is employed. Robustness to channel errors is achieved by judicious assignment of binary indexes to the VQ codevectors.

Consider a source-optimized VQ specified by the codebook $C = \{\boldsymbol{y}_J\}$. Given a source vector $\boldsymbol{x} \in \mathcal{R}^k$, the VQ encoder determines the index I by performing the nearest neighbor search: $I(\boldsymbol{x}) = \arg\ \min_L \|\boldsymbol{x} - \boldsymbol{y}_L\|^2$. The decoder receives index J and produces codevector vector $\boldsymbol{y}_J$. The overall distortion can be decoupled into quantization distortion D_q and channel distortion D_c

$$D = \overbrace{\frac{1}{|\mathcal{T}|}\sum_{\boldsymbol{x})i\in\mathcal{T}} \|\boldsymbol{x}_i - \boldsymbol{y}_I\|^2}^{D_q} + \overbrace{\frac{1}{|\mathcal{T}|}\sum_{\boldsymbol{x}_i\in\mathcal{T}}\sum_{J} p_{J|I}\|\boldsymbol{y}_I - \boldsymbol{y}_J\|^2}^{D_c}. \tag{5}$$

IA consists of selecting a representative value for the channel BER ϵ_r for design purposes and assigning indexes to codevectors so as to minimize the channel distortion $D_c(\epsilon_r)$. The resulting VQ offers robustness over a wide range of channel conditions.

The above IA optimization problem is NP-hard (i.e., it is at least as hard as a known class of optimization problems for which no polynomial time solution is believed to exist). Hence, the global minimum is generally elusive. There exist several greedy and local descent techniques to perform IA. A widely used local descent technique is the BSA [2]. It consists of a sequence of switching tests wherein the indexes of two codevectors are switched and the change in the channel distortion $D_c(\epsilon_r)$ is evaluated. If the distortion decreases, the switch is "accepted," else it is "rejected" and the system defaults back to its previous state.

The local descent nature of BSA makes the quality of the IA solution highly sensitive to the initial IA. To overcome this drawback, we propose to adapt the NCR approach to the problem of IA.

NCR for IA: In contrast to the BSA approach which directly assigns indexes to the codevectors of a given source-optimized VQ, we design a VQ with prewired indexing. More specifically, we apply NC-GLA at gradually diminishing levels of the assumed channel BER ϵ. Here, however, instead of stopping at a specific expected level of channel BER, we reduce the value of ϵ all the way to zero. Thus, the VQ that we ultimately obtain is optimized for the noiseless channel, i.e., it is a source-optimized VQ. Moreover, the design procedure that optimizes the VQ at progressively lower noise levels assures that the correspondence between codewords and codevectors at high levels of channel noise is preserved and provides good IA at the noiseless limit.

We substantiate the performance of NCR for IA by experimental comparison with the performance of BSA. The results are given for the case of four- and eight-dimensional VQ's with codebook size of 256, which were designed for Gauss–Markov sources. The standard BSA was initialized with a VQ designed by GLA with the splitting initialization (see e.g., [4] for details on this setup). For the case of design by NCR, the initialization and the relaxation schedule employed was as specified in Section III. The number of empty encoding regions decreases with ϵ. When this number reaches zero, we speed up the update rule to $\epsilon_{m+1} = (\epsilon_m/1.2)$. The final iterations used $\epsilon = 0$. The resulting IA was fine-tuned using the BSA (for final local convergence). The performance over a test set was evaluated for a variety of channel conditions and is depicted in Fig. 2. Training and test sets of size 24 000 each were employed to design and test the performance of BSA and NCR schemes. It is evident that NCR generates superior IA. We observe substantial performance gains in the range of 0.7–1.1 dB at moderate to high levels of channel BER.

It is important to consider the complexity of the competing IA methods. The design by NCR requires three to four times the computational load of GLA. Our experiments indicate that the computational complexity of BSA itself is comparable to the complexity of GLA (see also [2]). However, when we employ BSA for fine tuning, the VQ already has good indexing that reduces the convergence time of BSA by a large factor. Based on experimental observation (and as expected from the above considerations), we found that the overall complexity of IA by NCR is about two to three times that of standard BSA design. Since the entire design is performed offline, we conclude that the benefits outweigh the moderate increase in the computational load.

V. Conclusions

This letter is concerned with VQ design for transmission over a noisy channel. The NCR approach for VQ design was presented. It was shown that for both channel-optimized VQ design and IA problems, NCR yields superior performance relative to known design algorithms. Typical performance gains are in the neighborhood of 1 dB. Although the experimental results focus on simple application examples, the technique can be used to build robustness into any signal coding scheme utilizing VQ as the quantization block.

References

[1] H. Kumazawa, M. Kasahara, and T. Namekawa, "A construction of vector quantizers for noisy channels," *Electron. Eng. Jpn.*, vol. 67-B, no. 4, pp. 39–47, 1984.

[2] K. Zeger and A. Gersho, "Pseudo-gray coding," *IEEE Trans. Commun.*, vol. 38, pp. 2147–2158, Dec. 1990.

[3] J. DeMarca and N. Jayant, "An algorithm for assigning binary indices to the codevectors of a multi-dimensional quantizer," in *Proc. Int. Conf. Communications*, 1987, pp. 1128–1132.

[4] N. Farvardin, "A study of vector quantization for noisy channels," *IEEE Trans. Inform. Theory*, vol. 36, pp. 799–809, July 1990.

[5] S. Gadkari and K. Rose, "Noisy channel relaxation for VQ design," in *Proc. ICASSP*, 1996, pp. 2048–2051.

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

[7] P. Knagenhjelm, "A recursive design method for robust vector quantization," in *Proc. Int. Conf. Signal Processing Applications and Technology*, Boston, MA, Nov. 1992, pp. 948–954.

[8] P. Hedelin, P. Knagenhjelm, and M. Skoglund, "Theory for transmission of vector quantized data," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds. Amsterdam, The Netherlands: Elsevier, 1995, ch. 10.

[9] K. Rose, E. Gurewitz, and G. C. Fox, "Vector quantization by deterministic annealing," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1249–1257, July 1992.

[10] D. Miller and K. Rose, "Combined source-channel vector quantization using deterministic annealing," *IEEE Trans. Commun.*, vol. 42, pp. 347–356, Feb. 1994.

[11] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, Jan. 1980.