THE CHANNEL-OPTIMIZED MULTIPLE-DESCRIPTION SCALAR QUANTIZER

Stephen D. Voran

Institute for Telecommunication Sciences, 325 Broadway, Boulder, CO 80305
svoran@its.bldrdoc.gov, 303-497-3839

ABSTRACT

Multiple-description coding is one way to gain robustness against lossy channels. We extend the multiple-description scalar quantizer (MDSQ) to a channel-optimized MDSQ (COMDSQ) that minimizes mean-squared error for a given channel environment. We discuss necessary and sufficient conditions for the optimality of M-channel COMDSQ's and provide procedures for the design of 2-channel COMDSQ's. We provide example results including audio files from waveform and transform coders that employ COMDSQ's.

1. INTRODUCTION

It is often necessary to transmit signals over lossy channels. Important examples of lossy channels include noisy and fading radio channels and congested packet data networks. Error correcting codes can be used to gain robustness to channel losses. Multiple-description coding (MDC) offers another way to gain robustness. The theory of MDC is set out in [1] and [2] and an example audio MDC application can be found in [3]. In MDC an encoder forms multiple partial descriptions of a signal and these descriptions are sent over different physical or virtual channels. If all descriptions arrive at the decoder intact, a higher-quality reconstruction is possible. If channel losses cause any of the descriptions to be lost, then a lower-quality reconstruction of the original signal is still possible.

Increasing the number of descriptions and the number of channels increases the robustness to the failure of any single channel or group of channels. Consider a set of identical independent channels each with failure probability μ. If a signal is sent by single-description coding the entire description will be lost with probability μ. If the signal is decomposed into M different partial descriptions and sent on M different channels, the probability of losing the entire description drops to μ^M. If the situation is such that the loss of the entire description is to be avoided at all costs, then one would pick the largest practical value of M.

One approach to MDC is multiple description quantization (MDQ). In MDQ, a sample is quantized and mapped to a set of codewords. If all codewords arrive at the decoder, the net effect is higher-resolution quantization. If any of the codewords are lost, the net effect is lower-resolution quantization. A multiple description scalar quantizer (MDSQ) was described by Vaishampayan in [4]. This work was extended and applied in [5]-[12].

In this paper we extend the MDSQ to the channel-optimized MDSQ (COMDSQ). The COMDSQ minimizes mean-squared

error (MSE) for a given channel environment. We discuss necessary and sufficient conditions for the optimality of M-channel COMDSQ's. We present procedures for designing 2-channel COMDSQ's and show example results. Finally, we apply COMDSQ's to waveform and transform audio coding and provide example results.

2. *M*-CHANNEL COMDSQ THEORY

Figure 1 summarizes the operation of the M-channel, b-bit COMDSQ. The input scalar variable x has probability density function $f_x(x)$. The quantizer Q uses the thresholds $\{t_i\}_{i=1}^{N+1}$ to partition the support of $f_x(x)$ into $N=2^b$ disjoint quantization cells $\{C_i\}_{i=1}^N$. The quantization rule is

$$t_i \le x < t_{i+1} \Rightarrow x \in C_i, \ i = 1 \text{ to } N. \quad (1)$$

A total of b bits are used to point to the appropriate cell, and a subset of those bits, b_l are sent into the lth channel. The inverse quantizer Q^{-1} uses all received bits to select one representation point r_k^j from a predetermined set of representation points. The selected representation point becomes the output $\hat{x}$.
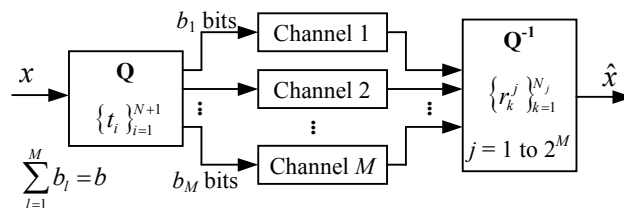

Figure 1. Block diagram of the M-channel, b-bit COMDSQ.

We assume that Q does not know the state of the channels, but Q^{-1} does know which channels have delivered their bits and which channels have not. Each channel can either deliver all its bits or deliver none of its bits, so the group of M channels will always be in one of 2^M possible states. Let $\log_2(N_j)$ be the number of bits received by Q^{-1} when the channels are in state j. Then Q^{-1} can choose between N_j distinct representation points $\{r_k^j\}_{k=1}^{N_j}$ when the channels are in state j. We define state $j=1$ to be the state where all channels deliver and $j=2^M$ to be the state where no channel delivers.

When the channels are in state $j>1$, one or more channels have failed to deliver their bits, N_j is less than N, and at Q^{-1} there is ambiguity regarding which quantization cell x is in. The exact ambiguities associated with each state depend on both the binary representation chosen to express the cell numbers (cell coding) and the chosen assignment of the b bits to the M channels

(channel packing). The compound effect of these two choices can be described by an index assignment or by an ambiguity function $a_j(i)$ that maps $\{1, 2, ..., N\}$ to $\{1, 2, ..., N_j\}$ when the channels are in state j. When the channels are in state j, if the received bits allow for differentiation of two quantizer cells C_u and C_v, then $a_j(u) \neq a_j(v)$. If the received bits do not allow for differentiation of C_u and C_v, then $a_j(u) = a_j(v)$.

We now present the M-channel, b-bit COMDSQ design problem. Let α_j be the probability that the channels are in state j, $j=1$ to 2^M. Then given the situation described in Figure 1, select an ambiguity function, a set of thresholds $\{t_i\}_{i=1}^{N+1}$, and 2^M sets of representation points $\{r_k^j\}_{k=1}^{N_j}$, $j=1$ to 2^M to minimize the mean-squared quantization error ε^2,

$$\varepsilon^2 = \sum_{j=1}^{2^M} \alpha_j \varepsilon_j^{\,2}, \tag{2}$$

where $\varepsilon_j^{\,2}$ is the mean-squared quantization error when the channels are in state j:

$$\varepsilon_j^{\,2} = \sum_{i=1}^{N} \int_{t_i}^{t_{i+1}} f_x(x)\left(x - r_{a_j(i)}^j\right)^2 dx. \tag{3}$$

The joint optimization of ambiguity function (or index assignment), thresholds, and representation points remains an open question. However, once an ambiguity function (or index assignment) has been picked, it is possible to find thresholds and representation points that minimize ε^2.

Good index assignments minimize the additional quantization noise power (ambiguity noise power) that results when one or more channels fail to deliver. Examples of good index assignments are given in [4] and [13]. The ambiguity noise power is directly related to the index assignment spread. When the input data is in quantizer cell C_i and the channels are in state j, the index assignment spread is given by $\max_k \left\{ r_k^j : a_j(k) = i \right\} - \min_k \left\{ r_k^j : a_j(k) = i \right\} + 1$. In [4] it is demonstrated that spread is an asymptotically good measure of ambiguity noise power. We use the index assignments in [13] because they attain the minimum spread that is theoretically possible. We are not aware of any other MDSQ design work using these index assignments.

Necessary conditions for the minimization of ε^2 with respect to the thresholds and representation points can be found by forcing

$$\frac{\partial \varepsilon^2}{\partial t_i} = 0, \; i = 2 \text{ to } N, \tag{4}$$

and

$$\frac{\partial \varepsilon^2}{\partial r_i^j} = 0, \; i = 1 \text{ to } N_j, \; j = 1 \text{ to } 2^M. \tag{5}$$

The thresholds t_1 and t_{N+1} are determined by the support of $f_x(x)$.

Equation (4) yields

$$t_i = \frac{\sum_{j=1}^{2^M} \alpha_j \left[\left(r_{a_j(i+1)}^j\right)^2 - \left(r_{a_j(i)}^j\right)^2 \right]}{2 \sum_{j=1}^{2^M} \alpha_j \left[r_{a_j(i+1)}^j - r_{a_j(i)}^j \right]}, \; i = 2 \text{ to } N, \tag{6}$$

and equation (5) gives

$$r_i^j = \frac{\sum_{k:a_j(k)=i} \int_{t_k}^{t_{k+1}} x f_x(x) dx}{\sum_{k:a_j(k)=i} \int_{t_k}^{t_{k+1}} f_x(x) dx}, \; i = 1 \text{ to } N_j, \; j = 1 \text{ to } 2^M. \tag{7}$$

Equation (7) says that when the channels are in state j, the ith representation point is located at the conditional mean of x, given that x is in one of the cells associated (via the ambiguity function) with the ith representation point. As expected, when $\alpha_1 = 1$, (6) and (7) reduce to the conventional Lloyd-Max Quantizer (LMQ) design equations [14],[15]. As in the LMQ case, (6) and (7) do not provide a closed form solution, but they do suggest iterative quantizer design procedures. We start with uniformly distributed thresholds and alternately use (7) to update the representation points and (6) to update the thresholds until both equations are satisfied simultaneously.

We use (7) in (3) to get an expression for ε^2 that is independent of the representation points. This expression for ε^2 depends only on the thresholds (once a data probability density function and an ambiguity function are chosen). We then form the N-1 by N-1 Hessian matrix H with elements h_{ij},

$$h_{ij} = \frac{\partial^2}{\partial t_i \partial t_j}\left(\varepsilon^2\right), \; i, j = 2 \text{ to } N. \tag{8}$$

Additional details on calculating the COMDSQ Hessian matrix are provided in Appendix A.

When the Hessian is non-negative definite, then equations (6) and (7) are both necessary and sufficient for a local minimum in ε^2. We have used (6) and (7) to design an assortment of 2-channel COMDSQ's for uniform, Gaussian, Laplacian, Rayleigh, and chi-squared distributions with b=2, 4, 6, and 8. In all cases we found the resulting Hessian to be non-negative definite, indicating that in these cases, (6) and (7) are both necessary and sufficient for a local minimum in ε^2.

3. TWO-CHANNEL COMDSQ RESULTS

We have designed COMDSQ's for the balanced 2-channel case, $b_1 = b_2 = \frac{b}{2}$. Unlike other 2-channel MDSQ work, these quantizers are *channel optimized*. We model the channels by extracting alternate samples from a single Gilbert channel model [16]. The Gilbert model can be tuned to approximate a variety of real channel loss processes, including Internet packet loss [17]. Our procedure of extracting alternate samples is analogous to the time-division multiplexing or packet interleaving procedures used to create two virtual channels from one physical Gilbert channel. The Gilbert channel is characterized by a 2-state Markov model with the transition probabilities $p = P$(sample i+1 lost | sample i received) and $q = P$(sample i+1 lost | sample i lost). We require $0 < p \leq 1$ and $0 \leq q < 1$. The average loss rate for the physical and virtual channels is

$$\mu = \frac{p}{p+1-q}, \; 0 < \mu < 1. \tag{9}$$

As a pair, the two channels will be in one of four possible states. The probabilities of these states are:

$$\alpha_1 = P(\text{no loss}) = 1 - \mu(2 - q),$$
$$\alpha_2 = \alpha_3 = P(\text{loss in exactly one channel}) = \mu(1 - q), \quad (10)$$
$$\alpha_4 = P(\text{loss in both channels}) = \mu\, q.$$

State 4 corresponds to the loss of all bits and thus cannot affect the design of the COMDSQ. Since (6) is invariant to a common scaling of all α values, we can divide all α values by α_2, leaving the COMDSQ design problem parameterized by the single parameter β:

$$\beta = \frac{\alpha_1}{\alpha_2} = \frac{1 - \mu(2 - q)}{\mu(1 - q)}, \quad 0 < \beta < \infty. \quad (11)$$

When $p=q$, the physical channel becomes memoryless, the two virtual channels become independent, and

$$\beta = \frac{1 - \mu}{\mu}. \quad (12)$$

In the 2-channel COMDSQ design procedure outlined below, we assume the physical channel is memoryless and move μ through the range (0,1). This moves β through its entire range. Thus, in this particular case, the memoryless channel assumption is not a restriction. One can use (11) to calculate β for arbitrary μ and q (or equivalently p and q) and then use (12) to find the corresponding value of "memoryless μ."

Figure 2 shows a typical evolution of COMDSQ thresholds as μ increases. When μ=0 we have the conventional *b*-bit LMQ design problem, and this is our starting place. After iterating to satisfy (6) and (7) we increase μ slightly and iterate to satisfy (6) and (7) again. As μ increases, some cells begin to shrink. Eventually the thresholds defining certain cells converge, reducing the cell to an empty one. This indicates that the cell has become a liability and should be removed from the COMDSQ design. (In Figure 2, the termination of t_i indicates that the cell C_i has been removed.) Each quantization cell reduces the granular quantization noise when all bits are received but each cell also increases distortion due to ambiguity when bits are not delivered (ambiguity noise). As the probability of channel failures increases, the importance of granular quantization noise decreases, the importance of ambiguity noise increases, and these two trends eventually tip the balance as to whether a given cell is an overall asset or liability. When the thresholds of a cell converge we remove that cell from the COMDSQ design and iterate until (6) and (7) are satisfied. We then increase μ slightly and continue on. As μ approaches 1, the COMDSQ reduces to a pair of identical $\frac{b}{2}$ - bit LMQ's. This is intuitive; when the probability that all *b* bits will be delivered is very small, and the probability that $\frac{b}{2}$ bits will be delivered is slightly larger, then the best strategy is to send two identical $\frac{b}{2}$ - bit representations of *x*.

For a given channel environment, the COMDSQ minimizes MSE with respect to the thresholds and the representation points. The cell elimination procedure described above is heuristic and intuitive but not necessarily optimal. For smaller values of *b*, searches over all of the approximately $2^{\left(2^b\right)}$ possible cell configurations at fixed μ are practical. For $b = 2$ and 4, these exhaustive searches have shown that the cell elimination procedure described above is very close to optimal. We are not

aware of any other MDSQ work that addresses the selection of an optimal set of quantization cells for a given channel situation.
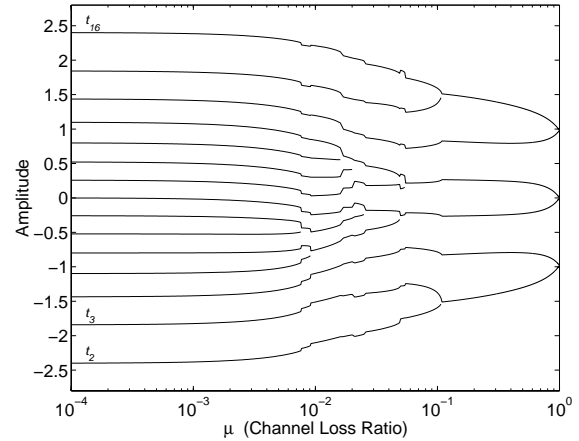

Figure 2. Example threshold evolution: 4-bit, 2-channel COMDSQ for Gaussian data.

Figure 3 shows a signal-to-quantization noise ratio (SQR) example for the 8-bit, 2-channel COMDSQ designed for the uniform distribution. For comparison, the SQR's for a single 8-bit LMQ and a pair of 4-bit LMQ's are shown as well. Note that the COMDSQ SQR approaches the former as $\mu \to 0$ and the latter as $\mu \to 1$. Between these extremes the COMDSQ outperforms both of these alternatives. The maximal improvement is 10.0 dB. (This increases with *b*.) The location of this maximal improvement is $\mu = 0.0035$ (this decreases with *b)*. For consistency we assume that the COMDSQ and the 8-bit LMQ use the same cell coding and channel packing and hence have the same ambiguity functions.
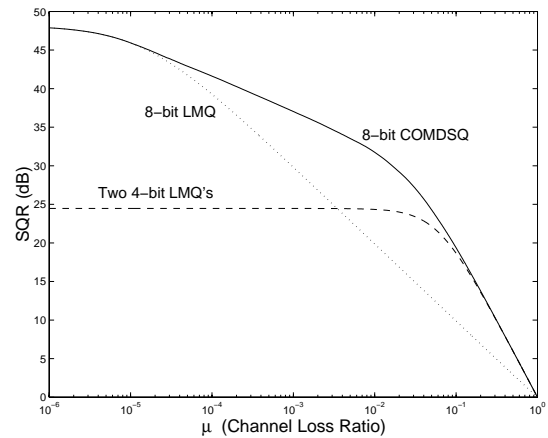

Figure 3. Example signal-to-quantization noise ratios: 8-bit, 2-channel COMDSQ and LMQ's for uniform data.

The COMDSQ is fairly robust to channel mismatch. In Figure 3 the performance of the COMDSQ has been optimized for the channels at every value of μ. For this particular case we can divide the entire μ range into just four intervals and design a single COMDSQ for each interval (design at $\mu = 10^{-5}$, 10^{-4}, 10^{-3}, and 10^{-2}). Across the entire μ range, the SQR of the resulting

"coarsely optimized" COMDSQ falls below the SQR of the "finely optimized" COMDSQ by no more than 1 dB. We have observed similar robustness to channel mismatch in each of the COMDSQ's we have designed.

We have also developed an alternative procedure for approximately determining the optimal number of cells for a given 2-channel environment. We start with the case of a 2-channel MDSQ with uniform cell widths operating on input data that is uniformly distributed on the interval [0,1]. Each channel carries $\frac{b}{2}$ bits, a maximum of $N=2^b$ cells can be represented, and the number of cells used by the quantizer is n, where

$$2^{\frac{b}{2}} = \sqrt{N} \le n \le N = 2^b .$$

When both channels are working the quantization error is just the granular quantization noise

$$\varepsilon_1^2 = \frac{1}{12n^2} . \tag{13}$$

When a single channel fails, ambiguity noise is added to this granular noise. (We make the very reasonable assumption that granular noise and ambiguity noise are independent.) When channel 1 fails, and the input data falls into the quantizer cell C_i, the ambiguity noise is $r_i^1 - r_{a_2(i)}^2$. When channel 2 fails it is $r_i^1 - r_{a_3(i)}^3$. We have empirically determined that for the index assignments given in [13], and the uniform data and cell-width assumptions given above, the ambiguity noise variance is closely modeled by,

$$\sigma_a^2 = 0.0431 \left(\frac{n}{N} \right)^{2.18} . \tag{14}$$

When $64 \le N$, this model fits well when either channel fails. Thus we have

$$\varepsilon_2^2 = \varepsilon_3^2 = \frac{1}{12n^2} + 0.0431 \left(\frac{n}{N} \right)^{2.18} \tag{15}$$

and

$$\varepsilon^2 = \sum_{j=1}^4 \alpha_j \varepsilon_j^2 = \left(\alpha_1 + \alpha_2 + \alpha_3 \right) \frac{1}{12n^2}$$
$$+ \left(\alpha_2 + \alpha_3 \right) 0.0431 \left(\frac{n}{N} \right)^{2.18} + \alpha_4 \frac{1}{12} . \tag{16}$$

We can find a necessary and sufficient condition for a minimum in (16) by taking first and second derivatives. The result is

$$n = 1.15 \left(\sqrt{N} \right)^{1.04} \left[\frac{\left(\alpha_1 + \alpha_2 + \alpha_3 \right)}{\left(\alpha_1 + \alpha_2 \right)} \right]^{0.239} . \tag{17}$$

Result (17) gives the number of quantization cells to use in order to minimize total quantization error (granular quantization noise plus ambiguity noise) under the conditions described above. Figure 4 shows (17) limited to a maximum value of N vs. μ. This figure also includes empirical results for COMDSQ's designed as described above for the uniform, Gaussian and Rayleigh data distributions with $b=6$. Result (17)

generally forms a lower bound for the empirical results and is closest to the uniform results since they most nearly conform to the uniform data and cell-width assumptions behind (17). However, (17) is not much farther from the results for Gaussian or Rayleigh data, even though they clearly violate those assumptions. Given this observation, we suggest that (17) might be used with some caution for COMDSQ design for data from arbitrary distributions.
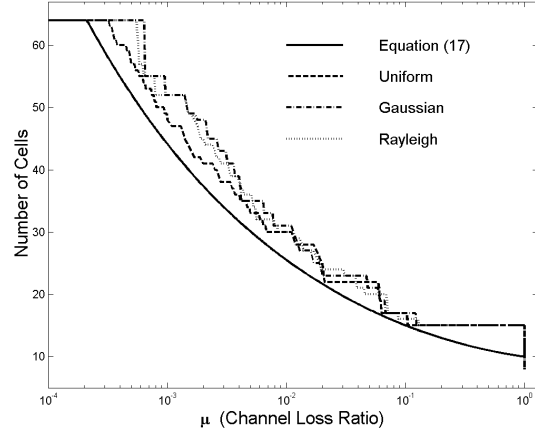

Figure 4. Number of quantizer cells vs. channel loss ratio for equation (17) and for three 6-bit COMDSQ's designed using the procedure given above.

Result (17) leads to an alternate COMDSQ design procedure. In this procedure, we start with N cells and slowly move μ from 0 to 1. Whenever (17) indicates that $n<N$, we select and remove a cell. There are several ways to select a cell for removal. We can remove the cell that holds the smallest probability mass, or we can remove the cell that reduces the mean-squared spread the most. (The mean-squared spread is directly related to the ambiguity noise variance.) The second approach has worked best in the cases we have tried. One might also select a cell for removal by stepping along the diagonals of the index assignment table defined by $a_j(i)$ as described in [4].

4. AUDIO CODING EXAMPLES

We built simple waveform and transform coders to demonstrate COMDSQ's and quantization effects. The waveform coder uses the μ-law compressor and expander functions defined in [18] in conjunction with a 10-bit, 2-channel COMDSQ designed for the uniform distribution. For comparison purposes, we also used a uniform-distribution 10-bit LMQ with the same ambiguity function. Table 1 defines the contents of example audio files from the waveform coder. All audio files and a table that links to them are available in the file voran_demo.zip included in the workshop proceedings. These files are also available at *www.its.bldrdoc.gov/home/programs/audio/comdsq/demo.htm*.

Quantizer	Both Channels	Channel 1	Channel 2
Lloyd-Max	WLb.wav	WL1.wav	WL2.wav
COMDSQ $\mu=2\times10^{-5}$	WC1b.wav	WC11.wav	WC12.wav
COMDSQ $\mu=1\times10^{-3}$	WC2b.wav	WC21.wav	WC22.wav

Table 1. COMDSQ waveform coder examples.

The transform coder generates modified discrete cosine transform (MDCT) coefficients that are compressed by a power law (0.75) and then quantized by COMDSQ's at 0, 2, 4, 6, 8, or 10 bits/sample according to a perception-based bit-allocation algorithm. For comparison purposes, we also used LMQ's with the same bit allocation and ambiguity function. All quantizers were designed for the uniform distribution. Table 2 defines the contents of example audio files from the transform coder. For these examples we used a fixed MDCT length of 256 samples (5.8 ms) and a 50% window overlap.

Quantizer	Both Channels	Channel 1	Channel 2
Lloyd-Max	XLb.wav	XL1.wav	XL2.wav
COMDSQ $\mu=1\times10^{-3}$	XC1b.wav	XC11.wav	XC12.wav
COMDSQ $\mu=3\times10^{-2}$	XC2b.wav	XC21.wav	XC22.wav

Table 2. COMDSQ transform coder examples.

It is clear that the LMQ would not be used in channel environments where only half the bits might arrive, unless additional layers of protection or redundancy were in place. The 2-channel COMDSQ has much better robustness to the loss of either channel. On the other hand, since it generally has fewer cells, the COMDSQ often induces more quantization noise than the LMQ when both channels are received. As μ gets larger, the COMDSQ 2-channel performance continues to drop, its single-channel performance continues to improve, and the two converge as $\mu \rightarrow 1$. Increasing μ increases the redundancy between the two channels and thus lets us trade robustness against channel failures for increased quantization noise when there is no channel failure.

Given estimates of the channel parameters p and q we can design 2-channel COMDSQ's that minimize MSE for those channels. But minimizing MSE is not necessarily equivalent to maximizing perceived audio quality. Subjective and objective assessment of audio quality with significant temporal variations remain open research areas, and auditory perception, audio content, and numerous coding parameters will influence the optimal 2-channel COMDSQ operating point.

APPENDIX A
COMDSQ HESSIAN DERIVATION

The N-1 by N-1 Hessian matrix H contains the elements h_{kl},

$$h_{kl} = \frac{\partial^2}{\partial t_k \partial t_l}\left(\varepsilon^2\right), \quad t,l = 2 \text{ to } N. \tag{A.1}$$

To calculate these elements, we first write the mean-squared quantization error in terms of the thresholds only:

$$\varepsilon^2 = \sum_{j=1}^{2^M} \alpha_j \varepsilon_j^2, \tag{A.2}$$

where

$$\varepsilon_j^2 = \sum_{i=1}^{N} \int_{t_i}^{t_{i+1}} \varphi\left(r_{a_j(i)}^j(t_1,t_2,\ldots,t_{N+1}),x\right)dx \tag{A.3}$$

and

$$\varphi\left(r_{a_j(i)}^j(t_1,t_2,\ldots,t_{N+1}),x\right) = f_x(x)\left(x - r_{a_j(i)}^j(t_1,t_2,\ldots,t_{N+1})\right)^2. \tag{A.4}$$

For notational simplicity, from this point forward we will not explicitly show the functional dependence of representation points $\left(r_{a_j(i)}^j\right)$ on the quantization thresholds.

Next we find several partial derivatives necessary for (A.1). Each of these results is valid for $k,l = 2$ to N. We use $\delta(i,j)$ to denote the Kronicker delta.

$$\frac{\partial}{\partial t_l} r_{a_j(i)}^j = f_x(t_l)\left[\delta\left(a_j(l-1),a_j(i)\right) - \delta\left(a_j(l),a_j(i)\right)\right] \times$$

$$\left[\frac{\left(t_l - r_{a_j(i)}^j\right)}{\displaystyle\sum_{m:\,a_j(m)=a_j(i)} \int_{t_m}^{t_{m+1}} f_x(x)dx}\right] \tag{A.5}$$

$$\frac{\partial^2}{\partial t_k \partial t_l} r_{a_j(i)}^j = \delta(k,l)\frac{\left(\frac{\partial f_x(x)}{\partial x}\right)}{f_x(x)}\Bigg|_{x=t_l} \times \frac{\partial}{\partial t_l} r_{a_j(i)}^j$$

$$+ \left\{\sum_{m:\,a_j(m)=a_j(i)} \int_{t_m}^{t_{m+1}} f_x(x)dx\right\}^{-1} \times \tag{A.6}$$

$$\left\{\left[\delta\left(a_j(l-1),a_j(i)\right) - \delta\left(a_j(l),a_j(i)\right)\right]f_x(t_l)\left(\delta(k,l) - \frac{\partial}{\partial t_k} r_{a_j(i)}^j\right)\right.$$

$$\left. - \left[\delta\left(a_j(k-1),a_j(i)\right) - \delta\left(a_j(k),a_j(i)\right)\right]f_x(t_k)\frac{\partial}{\partial t_l} r_{a_j(i)}^j\right\}$$

$$\frac{\partial}{\partial t_l}\varphi\left(r^j_{a_j(i)},x\right)=2\left(r^j_{a_j(i)}-x\right)f_x(x)\frac{\partial}{\partial t_l}r^j_{a_j(i)} \qquad (A.7)$$

$$\int_{t_i}^{t_{i+1}}\frac{\partial^2}{\partial t_k\partial t_l}\varphi\left(r^j_{a_j(i)},x\right)dx=$$

$$2\left[\frac{\partial}{\partial t_k}r^j_{a_j(i)}\frac{\partial}{\partial t_l}r^j_{a_j(i)}+r^j_{a_j(i)}\frac{\partial^2}{\partial t_k\partial t_l}r^j_{a_j(i)}\right]\int_{t_i}^{t_{i+1}}f_x(x)dx \quad (A.8)$$

$$-2\frac{\partial^2}{\partial t_k\partial t_l}r^j_{a_j(i)}\int_{t_i}^{t_{i+1}}x f_x(x)dx$$

$$\frac{\partial}{\partial t_k}\varphi\left(r^j_{a_j(i)},t_l\right)=2\left(r^j_{a_j(i)}-t_l\right)f_x(t_l)\frac{\partial}{\partial t_k}r^j_{a_j(i)}$$

$$+\delta(k,l)\left\{\left(\frac{\partial f_x(x)}{\partial x}\right)\Bigg|_{x=t_l}\times\left(r^j_{a_j(i)}-t_l\right)^2\right. \qquad (A.9)$$

$$\left.+2\left(t_l-r^j_{a_j(i)}\right)f_x(t_l)\right\}$$

Finally, we calculate (A.1) as

$$h_{kl}=\frac{\partial^2}{\partial t_k\partial t_l}\left(\varepsilon^2\right)=\sum_{j=1}^{2^M}\alpha_j\frac{\partial^2}{\partial t_k\partial t_l}\varepsilon_j^2,$$

$$\frac{\partial^2}{\partial t_k\partial t_l}\varepsilon_j^2=\sum_{i=1}^{N}\int_{t_i}^{t_{i+1}}\frac{\partial^2}{\partial t_k\partial t_l}\varphi\left(r^j_{a_j(i)},x\right)dx$$

$$+\delta(k,i+1)\frac{\partial}{\partial t_l}\varphi\left(r^j_{a_j(i)},x\right)\Bigg|_{x=t_{i+1}}$$

$$-\delta(k,i)\frac{\partial}{\partial t_l}\varphi\left(r^j_{a_j(i)},x\right)\Bigg|_{x=t_{i+1}} \qquad (A.10)$$

$$+\delta(l,i+1)\frac{\partial}{\partial t_k}\varphi\left(r^j_{a_j(i)},t_{i+1}\right)$$

$$-\delta(l,i+1)\frac{\partial}{\partial t_k}\varphi\left(r^j_{a_j(i)},t_{i+1}\right).$$

Note that (A.10) gives five terms for each entry in the Hessian matrix. The first term can be evaluated using (A.8), (A.5), and (A.6). The second and third terms are similar to each other and can be evaluated using (A.7) and (A.5). The final two terms are similar and can be evaluated using (A.9) and (A.5).

5.   REFERENCES

[1] El Gammal, A.A., and Cover, T.M., "Achievable Rates for Multiple Descriptions," *IEEE Trans. Inform. Theory,* vol. IT-28, pp. 851-857, Nov. 1982.

[2] Ozarow, L., "On a Source Coding Problem with Two Channels and Three Receivers," *Bell Syst. Tech. J.*, vol. 59, pp. 1909-1921, Dec. 1980.

[3] Arean, R., Kovačević, J., and Goyal, V.K., "Multiple Description Perceptual Audio Coding with Correlating Transforms," *IEEE Trans. on SAP.,* vol. 8, pp. 140-145, Mar. 2000.

[4] Vaishampayan, V.A., "Design of Multiple Description Scalar Quantizers," *IEEE Trans. Inform. Theory,* vol. 39, pp. 821-834, May 1993.

[5] Vaishampayan, V.A., and Domaszewicz, M., "Design of Entropy - Constrained   Multiple - Description   Scalar Quantizers," *IEEE Trans. Inform. Theory,* vol. 40, pp. 245-250, Jan. 1994.

[6] Buzi, L., Vaishampayan, V.A., and Laroia, R., "Design and Asymptotic Performance of a Structured Multiple Description Vector Quantizer," *Proc. 1994 IEEE Int. Symp. on Inform. Theory,* p. 459, 1994.

[7] Batllo, J.C., and Vaishampayan, V.A., "Multiple Description Transform Codes with an Application to Packetized Speech," *Proc. 1994 IEEE Int. Symp. on Inform. Theory,* p. 458, 1994.

[8] Yang, S. and Vaishampayan, V.A., "Low-Delay Communication for Rayleigh Fading Channels: An Application of the Multiple Description Quantizer," *IEEE Trans. on Comm.,* vol. 43, pp. 2771-2783, Nov. 1995.

[9] Vaishampayan, V.A., Batllo, J.C., and Calderbank, A.R., "On Reducing Granular Distortion in Multiple Description Quantization," *Proc. 1998 IEEE Int. Symp. on Inform. Theory,* p. 98, 1998.

[10] Servetto, S.D., Vaishampayan, V.A., and Sloane, N.J.A., "Multiple Description Lattice Vector Quantization," *Proc. 1999 IEEE Data Compression Conf.,* pp. 13-22, 1999.

[11] Fleming, M. and Effros, M., "Generalized Multiple Description Vector Quantization," *Proc. 1999 IEEE Data Compression Conf.,* pp. 3-12, 1999.

[12] Kelner, J., Goyal, V.K, Effros, M., and Kovačević, J., "Multiple Description Lattice Vector Quantization: Variations and Extensions," *Proc. 2000 IEEE Data Compression Conf.,* pp. 480-489, 2000.

[13] Berger-Wolf, T.Y., and Reingold, E.M., "Optimal Multichannel Communication under Failure," *Proc. Tenth Annual ACM-SIAM Symp. on Discrete Algorithms,* pp. S858-S859, 1999.

[14] Lloyd, S.P., "Least Squares Quantization in PCM," unpublished Bell Lab. Tech. Note., portions presented at the Institute of Mathematical Statistics Meeting, Atlantic City, NJ, Sept. 1957. Also, *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 129–137, Mar. 1982.

[15] Max, J., "Quantizing for Minimum Distortion," *IRE Trans. Inform. Theory,* vol. IT-6, pp. 7-12, March 1960.

[16] Gilbert, E.N., "Capacity of a Burst-Noise Channel," *Bell Syst. Tech. J.,* vol. 39, pp. 1253-1265, Sep. 1960.

[17] Bolot, J.C., Fosse-Parisis, S., and Towsley, D., "Adaptive FEC-Based Error Control for Internet Telephony," Proc. IEEE Infocom `99, pp. 1453-1460, Mar. 1999.

[18] ITU-T Recommendation G.711, "Pulse Code Modulation (PCM) of Voice Frequencies," Geneva, 1988.