A Pyramid Vector Quantizer

THOMAS R. FISCHER, MEMBER, IEEE

Abstract—The geometric properties of a memoryless Laplacian source are presented and used to establish a source coding theorem. Motivated by this geometric structure, a pyramid vector quantizer (PVQ) is developed for arbitrary vector dimension. The PVQ is based on the cubic lattice points that lie on the surface of an L-dimensional pyramid and has simple encoding and decoding algorithms. A product code version of the PVQ is developed and generalized to apply to a variety of sources. Analytical expressions are derived for the PVQ mean square error (mse), and simulation results are presented for PVQ encoding of several memoryless sources. For large rate and dimension, PVQ encoding of memoryless Laplacian, gamma, and Gaussian sources provides mse improvements of 5.64, 8.40, and 2.39 dB, respectively, over the corresponding optimum scalar quantizer. Although suboptimum in a rate-distortion sense, because the PVQ can encode large-dimensional vectors, it offers significant reduction in mse distortion compared with the optimum Lloyd–Max scalar quantizer, and provides an attractive alternative to currently available vector quantizers.

I. INTRODUCTION

THE DATA COMPRESSION or vector quantization of a continuous-valued source has a rich and varied history, dating to the original work of Shannon [1]. An algorithm for the design of scalar quantizers based on the necessary conditions for optimality is well-known [2], [3], and the resulting Lloyd–Max quantizer is routinely used in data compression applications. Despite the relative simplicity of the optimum scalar quantizer design, the performance fails to achieve a distortion close to the rate-distortion bound, particularly for sources with memory. For certain memoryless sources, however, including entropy coding in the design process has yielded very good performing scalar quantizers [4].

In an attempt to obtain quantization performance closer to that promised by rate-distortion theory, there has recently been considerable interest in vector quantization. In this case the average statistical properties of a block or vector of data values can be used to advantage. A theoretical basis for asymptotically optimum vector quantization has been provided by Zador [5] and Gersho [6], [7], and asymptotic vector quantizer (VQ) performance bounds were developed by Yamada, Tazaki, and Gray [8]. The necessary conditions for an optimum scalar quantizer have been generalized to the vector case, and Linde, Buzo, and Gray (LBG) [9] developed an iterative vector quantizer design algorithm based on a training sequence. This LBG algorithm has been rigorously studied [10] and applied to the vector quantization of a number of sources (see, e.g., [11]–[16]). Although the LBG approach is quite general, it suffers from two basic drawbacks. First, the training sequence approach is essentially a Monte Carlo method and requires considerable computer time for the design of each vector quantizer. Second and more importantly, the design that is produced has no general structure, and the implementation of the VQ design requires considerable computation. In the worst case the implementation requires a distance calculation between the input vector and every output vector, so that the best representation vector can be determined. Such a brute force search for the nearest representation point becomes infeasible for modest rates and dimensions.

The principal alternative to quantizers based on the LBG algorithm is the lattice quantizer [7], [17]–[20]. By using a regular set of points in space, lattice quantizers offer the potential of rapid encoding and decoding algorithms [18], [19]. Sayood, Gibson, and Rost [20] have used a lattice quantizer in transform image coding, but so far lattice quantizers have been primarily oriented toward memoryless uniform sources.

The purpose of the present paper is twofold; first, the geometric properties of a Laplacian source are developed and used to prove a source coding theorem; second, motivated by the geometric approach, an instrumentable vector quantizer is developed for arbitrary vector dimension. The general approach is strongly influenced by Sakrison's [21] clever development of the geometric properties of a Gaussian source. As noted by a perceptive reviewer, the source coding theorem presented in Section III basically replaces the Gaussian source model, mean square error (mse) criterion, and representation sphere of Sakrison's treatment, with a Laplacian source model, mean absolute error (mae) criterion, and representation pyramid. That such an extension of Sakrison's approach is possible, is interesting in its own right, and a general geometric formulation for source coding is presented elsewhere [22]. Unfortunately, as in [21], the proof of the source coding theorem is based on a random coding argument, which is not instrumentable [23]. Unlike Sakrison's treatment of the Gaussian source, however, the geometric approach is then used to develop an implementable vector quantizer, termed a pyramid vector quantizer (PVQ). The PVQ uses codewords corresponding to points in the cubic (i.e., all integer) lattice, that also lie

Manuscript received November 1, 1983; revised December 20, 1984. This work was supported in part by the Air Force Office of Scientific Research, under Grant AFOSR 84-0003. A preliminary version of this paper was presented at the 21st Annual Allerton Conference on Communication, Control, and Computing, October 5–7, 1983.

The author is with the Telecommunications and Control Systems Laboratory, Department of Electrical Engineering, Texas A & M University, College Station, TX 77843, U.S.A.

IEEE Log Number 8608405.

on a particular pyramid. As such, the PVQ is a type of lattice quantizer, but interestingly, not one based on a uniformly distributed source.

The paper is organized as follows. Sections II and III develop the geometric properties of a memoryless Laplacian source and prove a source coding theorem, verifying the asymptotic optimality (in a rate-distortion sense) of the geometric approach. In Section IV the PVQ is developed and generalized to a product code [24] version appropriate for moderate sizes of vector dimension. Analytical expressions approximating the large-dimensional mse performance of the PVQ and product code PVQ are then developed through use of Shannon's entropy power [1]. For the product code PVQ, the optimum rate allocation is determined for the "gain" and "shape" (see [24]) code books. In Section V the PVQ is generalized to apply to a wide variety of other sources, with memoryless Gaussian and gamma sources treated in detail. Section VI provides the results of Monte Carlo simulations of PVQ and product code PVQ performance for Laplacian, Gaussian, and gamma sources, at a variety of rates and dimensions. These results are compared with the (large-dimensional) performance expressions, the rate-distortion bounds, optimum scalar quantizer performance, the Sayood, Gibson, and Rost VQ, and several LBG algorithm-based results in the literature. It is demonstrated that because the PVQ can take advantage of large block sizes, for memoryless sources the PVQ and product code PVQ offer an attractive alternative to scalar quantizers and LBG algorithm-based vector quantizers.

II. The Geometric Structure of a Laplacian Source

Let X_i be a sequence of independent and identically distributed (i.i.d.) Laplacian random variables with probability density function

$$p_X(x_i) = \frac{\lambda}{2} e^{-\lambda |x_i|}. \tag{1}$$

The sequence X_i is assembled into vectors $\boldsymbol{X}$ of length L, with a resulting density

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \prod_{i=1}^{L} p_X(x_i).$$

A contour of constant probability density is specified by the condition

$$r \triangleq \sum_{i=1}^{L} |X_i| = \|\boldsymbol{X}\|_1 = \text{constant}, \tag{2}$$

where $\|\boldsymbol{X}\|_1$ is the familiar l_1 norm. Let $S(L, K)$ be defined as

$$S(L, K) = \left\{ \boldsymbol{X}:\ \sum_{i=1}^{L} |X_i| = K \right\}. \tag{3}$$

Geometrically, $S(L, K)$ is the surface of a hyperpyramid in L-dimensional space. The pyramid $S(L, K)$ encloses an L-dimensional volume $V(L, K)$ and has an L-dimensional surface area (($L-1$)-dimensional volume), $A(L, K)$. This volume and surface area can be calculated to be [31, p. 620]

$$V(L, K) = \frac{2^L K^L}{\Gamma(L+1)}$$

$$A(L, K) = \frac{2^L \sqrt{L} K^{L-1}}{\Gamma(L)} \tag{4}$$

where $\Gamma(L) = (L-1)!$.

The scalar random variable r may be thought of as a radial parameter (in the l_1 sense) that indexes a particular contour of constant density $f_{\boldsymbol{X}}(\boldsymbol{x})$ or, equivalently, the pyramid $S(L, r)$. The probability density function for r is easily calculated (using the moment generating function) as

$$p_r(r) = \frac{\lambda^L r^{L-1} e^{-\lambda r}}{\Gamma(L)}.$$

Defining

$$\rho = \frac{r}{L}$$

as the per dimension l_1 norm of $\boldsymbol{X}$, it is easily verified that

$$E[r] = \frac{L}{\lambda}$$

$$\text{var}[r] = \frac{L}{\lambda^2}$$

$$E[\rho] = \frac{1}{\lambda}$$

$$\text{var}[\rho] = \frac{1}{L\lambda^2}. \tag{5}$$

With respect to the (per dimension) l_1 distance measure, it is clear that for large L, vector $\boldsymbol{X}$ becomes highly localized around the particular contour of constant density indexed by $r = L/\lambda$. That is, the per dimension l_1 distance between $\boldsymbol{X}$ and a point on pyramid $S(L, L/\lambda)$ goes to zero as L becomes arbitrarily large. More generally, for any of the norms

$$\|\boldsymbol{X}\|_\nu = \left(\sum_{i=1}^{L} |X_i|^\nu \right)^{1/\nu}, \qquad \nu \geq 1, \tag{6}$$

the per dimension (norm) distance between $\boldsymbol{X}$ and the closest point on pyramid $S(L, L/\lambda)$ goes to zero as dimension L gets large. This may be argued intuitively based on the differential entropy of the source, as follows.

The vector $\boldsymbol{X}$ has entropy per degree of freedom [1],

$$h = -\frac{1}{L} \int \cdots \int f_{\boldsymbol{X}}(\boldsymbol{x}) \log f_{\boldsymbol{X}}(\boldsymbol{x})\, d\boldsymbol{x}$$

$$= \log\left(\frac{2e}{\lambda}\right).$$

Further, it was observed by Shannon [1] that

$$\frac{1}{L} \log f_{\boldsymbol{X}}(\boldsymbol{x}) \overset{\text{a.s.}}{\to} -h \text{ as } L \to \infty. \tag{7}$$

Substituting (1) and (2) into (7) yields

$$\frac{\lambda}{L} r \to 1,$$

so that the important pyramid for source coding is (from (3)) $S(L, L/\lambda)$.

For each vector X let $\hat{X}$ be a closest vector on $S(L, L/\lambda)$, in the norm sense of (6). Then, asymptotic in dimension, negligible distortion is introduced by approximating X by $\hat{X}$. This is summarized by the following theorem.

Theorem 1: For each i.i.d. Laplacian vector X, let $\hat{X} \in S(L, L/\lambda)$ be a vector closest to X (in the $\|X - \hat{X}\|_\nu$ sense). Then, for arbitrary $\epsilon > 0, \delta > 0$, and any fixed α satisfying $0 < \alpha < \infty$, for large enough dimension L,

$$\frac{1}{L}\|X - \hat{X}\|_\nu^\alpha < \epsilon$$

for all X except on a set of total probability δ. Thus, for each X a corresponding $\hat{X} \in S(L, L/\lambda)$ exists such that

$$\frac{1}{L}\|X - \hat{X}\|_\nu^\alpha \xrightarrow{p} 0.$$

Proof: See Appendix I.

The norm distance described in the theorem can be computed explicitly if $\nu = 1$ (and $\alpha = 1$). In this case the mae distance is computed (with $\hat{X} = LE[|X|]X/\|X\|_1$) as

$$E\left\{\frac{1}{L}\|X - \hat{X}\|_1\right\} = E\{|\rho - E(\rho)|\} = \frac{2L^L e^{-L}}{\lambda\Gamma(L+1)}$$

which, using Stirling's approximation for large L, is

$$E\left\{\frac{1}{L}\|X - \hat{X}\|_1\right\} \approx \frac{2}{\lambda\sqrt{2\pi L}}.$$

For $\nu = 2$ the distortion measure is mse and the vector $\hat{X}$ can be computed from the well-known projection theorem. Let vector $S(X)$ have components

$$S_i = \begin{cases} 1, & \text{if } X_i > 0 \\ 0, & \text{if } X_i = 0, \\ -1, & \text{if } X_i < 0 \end{cases} \quad \text{if } \|X\|_1 \geq L/\lambda, \quad (8a)$$

or

$$S_i = \begin{cases} 1, & \text{if } X_i \geq 0 \\ -1, & \text{if } X_i < 0 \end{cases}, \quad \text{if } \|X\|_1 < L/\lambda. \quad (8b)$$

The two cases correspond to X being either on or outside (8a) or inside (8b) the pyramid, respectively. Using the projection theorem, the vector on the pyramid $S(L, L/\lambda)$ that is closest to X can be computed as

$$\hat{X} = X - \left[\left(X, \frac{S}{\|S\|_2}\right) - \frac{L\|S\|_2}{\lambda\|S\|_1}\right]\frac{S}{\|S\|_2} \quad (9)$$

where $(\cdot, \cdot)$ is the usual inner product and provided

$$S(X) = S(\hat{X}). \quad (10)$$

If application of (9) yields a vector $\hat{X}$ that does not satisfy (10) (corresponding to X being "above" a corner or edge of the pyramid), then $\hat{X} \notin S(L, L/\lambda)$ and all components of X that fail (10) should be set to zero and (8) and (9) reapplied. This condition (10) can easily be checked using a computer algorithm. Setting the indicated components of X to zero allows (9) to generate the closest point on the pyramid.

For the norm-based difference distortion measures of (6), nothing is lost (for large L) by first representing X by an appropriate point on the pyramid and then quantizing this point with an optimum vector quantizer. Selection of the optimum vector quantizer for points on the pyramid will naturally depend on the particular distortion measure. The essential concept is that only a single contour of constant probability density is important for designing the optimum source code. Further (somewhat obviously), since the probability density is constant along this important geometric surface, quantizer representation regions should tend to be distributed uniformly on the surface of the pyramid. This intuitive observation is formalized in the following section.

III. A Source Coding Theorem

For the norm-based distortion measures of (6), an asymptotically (in dimension) optimum source code or vector quantizer for a memoryless Laplacian source can be designed based solely on the pyramid $S(L, L/\lambda)$. That is, from Theorem 1 it follows that an optimum source code can be decomposed into the two steps of 1) finding a point on $S(L, L/\lambda)$ that is closest to X in the distortion sense, and 2) quantizing this point with an optimum vector quantizer. Actual construction of the vector quantizer depends, however, on the geometric structure of the hyperpyramid.

Based on Sakrison's development of an optimum source code for a memoryless Gaussian source and an mse distortion criterion [21], a source encoding procedure for an i.i.d. Laplacian vector X and the mae distortion measure is as follows.

1) Form

$$X = X \cdot \frac{E[\|X\|_1]}{\|X\|_1}.$$

2) Let a rate $R \geq 0$ be selected and choose 2^{RL} points at random on $S(L, L/\lambda)$ according to a uniform distribution. About each point place a representation region of geometric shape $S^*(L, \epsilon)$ defined by

$$S^*(L, \epsilon) = \{x: \|x\|_1 \leq \epsilon\}.$$

Adjust the location of each region (i.e., adjust the centroid) so that the region both contains the original point and has maximum area of intersection with pyramid $S(L, L/\lambda)$. Let the centroid of region i be denoted as y_i.

3) If X falls in the ith region, then X is encoded as y_i.

The performance of the source code generated by the construction of 1)–3) is summarized by the following theorem.

Theorem 2: Let $D(R) = E\{(1/L)\|X - y\|_1\}$ be the mae distortion resulting from the construction of 1)–3). Then, for arbitrary $\delta > 0$ and $R \geq 0$, a positive integer L_δ exists such that for $L > L_\delta$

$$D^*(R) \leq D(R) < (1 + \delta)D^*(R),$$

where $D^*(R) = 2^{-R}/\lambda$ is the distortion-rate function for an i.i.d. Laplacian source.

Proof: See Appendix II.

In the case of the mean-square error performance criterion, a construction analogous to 1)–3) (using (9) for $\hat{X}$ and selecting the representation regions as spheres instead of pyramids) can be shown to provide asymptotically (in L) optimum performance in the small distortion case. For either performance measure, however, the construction is based on a random assignment of points to $S(L, L/\lambda)$ and leads to designs which are extremely difficult to implement. An alternative design approach is to look for points on $S(L, L/\lambda)$ that have useful structural properties. Although such an approach does not necessarily lead to designs that achieve optimum (rate-distortion) performance, the implementation can be quite simple and provide performance significantly better than that of the optimum scalar quantizer. This approach is developed in the following sections.

IV. A Pyramid Vector Quantizer

A PVQ can be constructed based on a subset of the points in the cubic lattice (that is, the set of all vectors with integer components). As such, the PVQ is a type of lattice quantizer but, significantly, a lattice quantizer that is not based on a uniform source pdf.

The construction of the PVQ will proceed in three distinct steps. First, the number of cubic lattice points on the pyramid is evaluated. Second, an encoding/decoding algorithm is developed, and the resulting source code is denoted the pyramid vector quantizer. The PVQ codewords are scaled versions of the lattice points that lie on a specified pyramid. Enumeration encoding and decoding algorithms are provided for representing the pyramid codewords as binary codewords for transmission or storage. Although asymptotic in dimension it is sufficient to base an optimum VQ design on a single ($S(L, L/\lambda)$) pyramid, for moderate sizes of dimension significant distortion may be introduced in approximating X by $\hat{X}$. As a final modification to the PVQ design, concentric pyramids are used to generate a hybrid scalar-vector quantizer in product code [24] form. In this design a fraction of the total bit rate is allocated to the scalar quantization of the radial random variable r with the remaining bit rate allocated to the PVQ for a normalized pyramid. The VQ construction is carried out for the mse distortion measure but may be readily extended to any of the norm-based distortion criteria of (6).

A. Integer Points on the Pyramid

If pyramid $S(L, L/\lambda)$ has $L/\lambda = K$ where K is a positive integer, then it is possible to calculate the number of vectors with integer components that lie on $S(L, K)$. Let $N(L, K)$ be the number of such vectors, defined explicitly as

$$N(L,K) = \left\{ \begin{array}{l} \text{the number of vectors } x \text{ such that} \\ \sum_{i=1}^{L} |x_i| = K, \quad \text{and} \quad x_i \text{ an integer, for } i = 1,\cdots, L \end{array} \right\}. \tag{11}$$

For $K = 1$, only one of the x_i in (11) is nonzero, with value either 1 or -1. Hence $N(L,1) = 2L$. Conversely, if $L = 1$, then x_i in (11) is either K or $-K$, so that $N(1, K) = 2$. Using a combinatorial argument, $N(L, K)$ is shown in Appendix III to satisfy the recursive formula

$$N(L,K) = N(L-1,K) + N(L-1,K-1) + N(L,K-1),$$

for $L \geq 2$ and $K \geq 2$. Since $N(1, K)$ and $N(L,1)$ are known, it is simple to compute $N(L, K)$. The integer coordinate points on $S(3,4)$ are illustrated in Fig. 1.

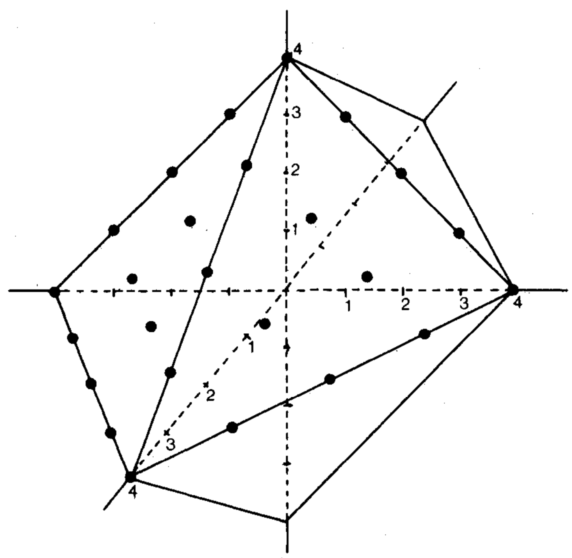

Fig. 1. Integer coordinate points on $S(3,4)$.

B. An Encoding Algorithm

To encode L-dimensional vectors at a specified rate per dimension R, the largest value of K should be found such that

$$N(L,K) \leq 2^{RL}. \tag{12}$$

The $N(L, K)$ points of $S(L, K)$ tend to be uniformly spread over the pyramid surface and make a natural set of representation points for the source code. The basic encoding algorithm is as follows.

1) Form $\hat{X} \in S(L, L/\lambda)$ from (9).

2) Scale $\hat{X}$ by $K\lambda/L$ to obtain a corresponding point, $\tilde{X}$, on $S(L, K)$.

3) Find the closest of the $N(L, K)$ integer component vectors on $S(L, K)$ to $\tilde{X}$. This may be done componentwise as follows.

a) Round each component of $\tilde{X}$ to the nearest integer; call the resulting vector $\tilde{y}$.

b) Compute $\|\tilde{y}\|_1$. If $\|\tilde{y}\|_1 = K$, then the closest point on $S(L, K)$ has been found. If $\|\tilde{y}\|_1 < K$, then increase by one in magnitude the $K - \|\tilde{y}\|_1$ components of $\tilde{y}$ that both contribute the largest distortion and were previously rounded down. If $\|\tilde{y}\|_1 > K$, then decrease by one in magnitude the $\|\tilde{y}\|_1 - K$ nonzero components of $\tilde{y}$ that contribute the most distortion and were previously rounded up. In the case of a tie for which component is to be modified, choose either one arbitrarily.

Let the resulting integer component vector be denoted $\hat{y}$.

4) The vector quantizer representation point is $y = \alpha\hat{y}/K$, where α is a scale parameter selected to minimize the distortion.

Selecting parameter α as

$$\alpha = \|\hat{X}\|_1 = \frac{L}{\lambda} \tag{13}$$

provides quantizer output points on pyramid $S(L, L/\lambda)$. The VQ representation regions should be adjusted, however, to provide the maximum volume of intersection with the pyramid surface. For the mse criterion the appropriate representation region is the sphere, while for the mae criterion the representation region is the pyramid. (The VQ encoding truncates these shapes to account for neighboring output points.) If one of the scaled $N(L, K)$ points of $S(L, K)$ lies in the middle of a face of the pyramid, then this point should be used as the representation sphere centroid. This follows because the pyramid face acts as an $(L-1)$-dimensional hyperplane slicing through the sphere, and the maximum volume of intersection occurs if the hyperplane passes through the sphere centroid. For low rates, however, K may be much less than L so that all of the $N(L, K)$ points lie along edges of the pyramid. The representation sphere centroid should then be located somewhat inside the pyramid. That is, the scaling factor α should be chosen as

$$\alpha = \gamma\frac{L}{\lambda} \tag{14}$$

where $\gamma \leq 1$ is a parameter that can be adjusted experimentally.

The source encoding algorithm is quite easy to implement, since it involves only rounding of the vector components and (if $\|\tilde{y}\|_1 \neq K$) a search over the vector components for the dimensions contributing the largest distortion. Implementation of the algorithm requires addition, multiplication, comparison, and rounding operations. The number of multiplications is a standard measure of implementation complexity, and steps 1)–4) of the algorithm show that no more than $4L$ multiplications are required. In contrast, the LBG algorithm has a design complexity that grows exponentially with the rate-dimension product (for a training sequence length that is proportional to the number of VQ output vectors), and a full-search encoding complexity that also depends exponentially on the rate-dimension product.

The performance of the basic pyramid VQ can be improved in two ways. First, the encoding algorithm of 1)–4) may be improved (for low rates) by inserting a step 1′) prior to 1).

1′) Use a threshold comparison on the components of X so that

$$X_i' = \begin{cases} X_i, & \text{if } |X_i| > \text{threshold} \\ 0, & \text{if } |X_i| \leq \text{threshold} \end{cases}, \tag{15}$$

for $i = 1, \cdots, L$. The modified X' is then used in 1)–4). Modification 1′) is justified by observing that if $|X_i|$ is small, then it will likely be rounded to zero in step 3) (for low rates and hence small K). Setting small X_i to zero in step 1′) reduces the size of the normalizing factor $\|\hat{X}\|_1$ used in step 2). Since K is substantially smaller than L for low rates, many of the components of the possible $\hat{y}$ are zero. Although the algorithm (step 3)) correctly finds the closest integer coordinate point on the pyramid, the VQ output is scaled by α, and the *scaled* point may no longer be closest to the original. The thresholding reduces this effect.

Using step 1′) requires that a new value of α be selected in step 4). For a high rate VQ, parameter α should be selected as

$$\alpha = E\left[\|\hat{X}\|_1\right] = L\left(T + \frac{1}{\lambda}\right)e^{-\lambda T}, \tag{16}$$

where T = threshold used in (15). For low rates, α may be selected as

$$\alpha = \gamma L\left(T + \frac{1}{\lambda}\right)e^{-\lambda T}, \tag{17}$$

again with $\gamma \leq 1$. In practice, the threshold parameter may be adjusted experimentally for a given R and L. Obviously, however, the threshold parameter T should be no larger than the value that satisfies

$$E\left[X^2 \,\middle|\, |X| \leq T\right] = D$$

where D is the mse to be obtained by the quantizer.

The second way in which the PVQ performance can be improved is related to the distortion introduced in approximating X by $\hat{X}$. Although for large dimension this distortion (per dimension) is negligible, for moderate sizes of dimension the distortion may be significant. Improved PVQ performance may be obtained by using concentric pyramids as the basis for the VQ output points. This approach is developed in Section IV-E. First, however, we develop a binary codeword representation for the PVQ codewords and an asymptotic approximation for the mse of the PVQ.

C. Binary Codeword Representation

For the PVQ to be useful in a digital communication or storage system, the $N(L, K)$ pyramid codewords corresponding to the lattice points on $S(L, K)$ must be uniquely represented as binary sequences of RL bits. Since the number of pyramid codewords can be counted, there exists an enumeration (encoding) procedure that assigns to each lattice point a unique integer in $\{0,\cdots, N(L, K) - 1\}$, which can be represented in the usual binary format (as a binary codeword) by RL bits. Inverting this procedure (enumeration decoding) assigns to each binary codeword the corresponding pyramid (lattice point) codeword. Obviously, if the rate-dimension product is small, then the enumeration encoding/decoding can be accomplished with a simple table lookup operation. For larger rate-dimension products the memory requirements of a table lookup become prohibitive, and efficient enumeration encoding and decoding algorithms are necessary.

A general enumeration algorithm may be based on partitioning the pyramid lattice points as follows. If $x^T = (x_1,\cdots, x_L)$, then for $x_1 = 0$, $N(L - 1, K)$ pyramid codewords are possible, corresponding to all permissible combinations of $(x_2,\cdots, x_L)$. Index these as $\{0,\cdots, N(L - 1, K) - 1\}$. If $x_1 = 1$, there are $N(L - 1, K - 1)$ possible pyramid codewords, and index these as $\{N(L - 1, K),\cdots, N(L - 1, K) + N(L - 1, K - 1) - 1\}$. If $x_1 = -1$, there are also $N(L - 1, K - 1)$ possible combinations of $(x_2,\cdots, x_L)$, and so index these codewords as $\{N(L - 1, K) + N(L - 1, K - 1),\cdots, N(L - 1, K) + 2N(L - 1, K - 1) - 1\}$. If $x_1 = 2$, there are $N(L - 1, K - 2)$ possible codewords, and these are indexed as $\{N(L - 1, K) + 2N(L - 1, K - 1),\cdots, N(L - 1, K) + 2N(L - 1, K - 1) + N(L - 1, K - 2) - 1\}$. The partitioning continues until $x_1 = K$, with index $N(L, K) - 2$, and $x_1 = -K$, with the final index $N(L, K) - 1$.

Next, for each value of x_1 the respective range of indices is further partitioned based on x_2. For example, assume that $x_1 = 0$. Then for $x_2 = 0$ there are $N(L - 2, K)$ pyramid codewords corresponding to all permissible values of $(x_3,\cdots, x_L)$. These codewords are indexed by $\{0,\cdots, N(L - 2, K) - 1\}$. If $x_2 = 1$, then the indices are $\{N(L - 2, K),\cdots, N(L - 2, K) + N(L - 2, K - 1) - 1\}$, etc. The partitioning continues through the vector component x_L, at which point there is a one-to-one correspondence between the pyramid codewords and the integers $\{0,\cdots, N(L, K) - 1\}$. For any specified pyramid codeword the corresponding index can be determined, represented as an RL bit binary sequence, and transmitted or stored. To decode, the binary sequence is interpreted as an integer and the pyramid codeword values are extracted sequentially, beginning with x_1. The PVQ enumeration encoding and decoding procedures can be summarized by the following algorithms.

Encoding Algorithm:

0) Set index $b = 0$, $i = 1$, $k = K$, $l = L$. Define $N(l, 0) = 1$ for all $l \geq 0$, $N(0, k) = 0$ for all $k \geq 1$, and

$$\operatorname{sgn}(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x = 0. \\ -1, & \text{if } x < 0 \end{cases}$$

1)
if $x_i = 0$, then $b = b + 0$,

if $|x_i| = 1$, then $b = b + N(l - 1, k)$

$$+ \left[\frac{1 - \operatorname{sgn}(x_i)}{2}\right] N(l - 1, k - 1),$$

if $|x_i| > 1$, then $b = b + N(l - 1, k)$

$$+ 2 \sum_{j=1}^{|x_i|-1} N(l - 1, k - j)$$

$$+ \left[\frac{1 - \operatorname{sgn}(x_i)}{2}\right] N(l - 1, k - |x_i|).$$

2) Replace

$$k \leftarrow k - |x_i|$$
$$l \leftarrow l - 1$$
$$i \leftarrow i + 1.$$

If $(k = 0)$, then stop and transmit b; otherwise, return to 1).

Decoding Algorithm:

0) Assume that the binary codeword is interpreted as an integer $b \in \{0,\cdots, N(L, K) - 1\}$ and is to be decoded as $\hat{x}^T = (\hat{x}_1,\cdots, \hat{x}_L)$. Set $\hat{x} = 0$; $i = 1$; $xb = 0$; $k = K$; $l = L$. Define $N(l, 0)$, $N(0, k)$ as in the encoding algorithm.

1) If $(b = xb)$, then $\hat{x}_i = 0$; go to 5).

2) If $(b < xb + N(l - 1, k))$, then $\hat{x}_i = 0$; go to 4). Otherwise, $xb = xb + N(l - 1, k)$; set $j = 1$.

3) If $(b < xb + 2N(l - 1, k - j))$, then

$$\hat{x}_i = \begin{cases} j, & \text{if } xb \leq b < xb + N(l - 1, k - j) \\ -j, & \text{if } b \geq xb + N(l - 1, k - j) \end{cases}.$$

Otherwise, $xb = xb + 2N(l - 1, k - j)$; $j = j + 1$; go to 3).

4) $k = k - |\hat{x}_i|$; $l = l - 1$; $i = i + 1$. If $(k > 0)$, go to 1).

5) If $(k > 0)$, then $\hat{x}_L = k - |\hat{x}_i|$. Vector $\hat{x}$ is the pyramid codeword.

The algorithms are most simply implemented if the values for $N(l, k)$ are stored in memory for $l = 1,\cdots, L$ and $k = 1,\cdots, K$. Since K is roughly proportional to the product RL, the memory requirement is proportional to RL^2. Both the enumeration encoding and decoding algorithms require at most one loop through the algorithm for each PVQ vector component.

Example: Using the pyramid $S(4,2)$, binary encode the PVQ codeword $x = (0,1,0,-1)$. By direct computation $N(4,2) = 32$, $N(3,2) = 18$, $N(2,2) = 8$.

0) Set $b = 0$; $i = 1$; $k = 2$.
1) $x_1 = 0$, so $b = 0$.
2) $i = 2$, $k = 2$.

Repeat.
1) $|x_2| = 1$, so $b = 0 + N(2,2) + [(1 - \text{sgn}(1))/2]N(2,1) = 8$.
2) $i = 3$, $k = 1$.

Repeat.
1) $x_3 = 0$, so $b = b + 0 = 8$.
2) $i = 4$, $k = 1$.

Repeat.
1) $x_4 = -1$, so $b = 8 + N(0,1) + [(1 - \text{sgn}(-1))/2]N(0,0) = 9$.
2) $i = 5$, $k = 0$, so stop with $b = 9$.

D. Asymptotic Performance

Let $D_{\text{PVQ}}(R)$ be the normalized mse for pyramid quantization at rate R. This distortion is certainly lower bounded by the distortion in the Shannon lower bound (SLB) [23] to the rate-distortion function,

$$D_{\text{PVQ}}(R) \geq D_{\text{SLB}}(R) = \frac{2e}{\pi\lambda^2}2^{-2R}. \tag{18}$$

However, a better approximation is possible. Since the PVQ output vectors are a regular subset of points in the cubic lattice, the normalized mse per quantization cell can asymptotically (in dimension) approach a value no smaller than 1/12. (Interestingly, this is also true for several other lattices (such as A_n and D_n) as well [17], and so basing a PVQ on such lattices would provide similar asymptotic performance.) This normalized mse is precisely that obtained from optimum scalar quantization of a uniformly distributed source, or equivalently, from optimum scalar quantization of each component (i.e., rectangular vector quantization) of a vector of i.i.d. uniform random variables. The resulting mse per dimension is then

$$\text{mse}_{\text{uniform}} = \sigma_u^2 2^{-2R}. \tag{19}$$

It remains to relate σ_u^2 to the parameters of the Laplacian source, and this can be done using Shannon's notion of entropy power [1]. All vectors $\hat{X}$ to be PVQ encoded have "sample differential entropy" of exactly $\log 2e/\lambda$. That is,

$$-\frac{1}{L}\log f_X(\hat{X}) = \log\frac{2e}{\lambda} \tag{20}$$

for each $\hat{X}$. For an i.i.d. uniform source, say Y, every realization satisfies

$$-\frac{1}{L}\log f_Y(Y) = \log 2\sqrt{3}\,\sigma_u, \tag{21}$$

since the density, $f_Y(y)$, is constant. Hence equating the entropies of (20) and (21) (equivalently, measuring the volume of the pyramid with a hypercube), the equivalent variance is

$$\sigma_u^2 = \frac{e^2}{3\lambda^2} \tag{22}$$

so that, using (19), the PVQ distortion is approximated as

$$D_{\text{PVQ}}(R) \simeq \frac{e^2}{3\lambda^2}2^{-2R} = \frac{e^2}{3}(E[|X|])^2 2^{-2R} \tag{23}$$

for large enough dimension. Implicit in the development of (23) are the assumptions that the PVQ lattice points are distributed roughly uniformly on the pyramid and that the distribution of $\hat{X}$ is roughly uniform on the pyramid surface.

For large rates the PVQ performance can be compared with the (Shannon lower bound to the) rate-distortion function in another way. To PVQ encode a memoryless source and achieve an mse distortion D requires, from (23), a rate

$$R_{\text{PVQ}}(D) = \frac{1}{2}\log_2\frac{e^2}{3\lambda^2 D}.$$

Comparing this with the rate R_{SLB} required in the Shannon lower bound to the rate-distortion function (18) indicates that the PVQ requires an increased rate of

$$R_{\text{PVQ}} - R_{\text{SLB}} = \frac{1}{2}\log_2\frac{e\pi}{6} \sim 0.255 \quad \text{bits.}$$

This is precisely the rate increase required for optimum scalar quantization with entropy coding [4], [28]. Significantly, however, the PVQ does not require the variable length codewords typically used in entropy coding which can cause synchronization problems with noisy channels. Further, the entropy coding approach can only be improved by using vector, instead of scalar, quantization. The PVQ performance can be improved by using a better lattice. For example, if the PVQ codewords were taken from the Leech lattice, then replacing the cubic lattice value for mse per quantization cell of 1/12 with the known [17] Leech lattice value of $0.06577\cdots$ (i.e., multiply (23) by $12 \times 0.06577\cdots$) yields an expected PVQ distortion of

$$D_{\text{LeechPVQ}} \sim \frac{0.263 \times e^2}{\lambda^2}2^{-2R}.$$

This would reduce the required rate to only 0.084 bits above the rate-distortion bound.

If the thresholding described in the previous section is included in the encoding algorithm, then the pyramid $S(L, L/\lambda)$ is modified to $S(L, L[T + 1/\lambda]e^{-\lambda T})$. This follows since, if X' is the source vector after thresholding, namely,

$$X_i' = \begin{cases} X_i, & \text{if } |X_i| > T \\ 0, & \text{if } |X_i| \leq T, \end{cases}$$

then $E[\|X'\|_1] = L[T + 1/\lambda]e^{-\lambda T}$. Parameter σ_u^2 must then be modified, so that by replacing $1/\lambda$ with $(T +$

$1/\lambda)e^{-\lambda T}$, (22) becomes

$$\sigma_u^2 = \frac{e^2}{3}\left(T + \frac{1}{\lambda}\right)^2 e^{-2T\lambda}. \quad (24)$$

The thresholding operation introduces the additional distortion (per dimension)

$$E\left[X^2 \mid |X| \le T\right] \cdot P_r(|X| \le T),$$

so that the overall mse distortion for the PVQ with thresholding is approximated as

$$D_{\mathrm{PVQ}T}(R) \simeq \frac{e^2}{3}\left(T + \frac{1}{\lambda}\right)^2 e^{-2\lambda T} 2^{-2R} + \frac{2}{\lambda^2} - \left(T^2 + \frac{2T}{\lambda} + \frac{2}{\lambda^2}\right)e^{-\lambda T}. \quad (25)$$

The threshold value T can be selected to minimize (25), and this minimizing value is easily computed from the necessary condition

$$\left(1 + \frac{1}{\lambda T}\right)e^{-\lambda T} = \frac{3}{2e^2}2^{2R}. \quad (26)$$

Equations (25) and (26) must be approached with some caution. Since the thresholded vector X' has components that can assume a value $x' = 0$ with positive probability $1 - e^{-\lambda T}$, the entropy power argument breaks down. For the optimizing T in (26), $D_{\mathrm{PVQ}T}(R)$ can actually drop below the Shannon lower bound for very small rates ($R < 1$). For larger rates, however, (25) provides a useful approximation to the reduction in mse that can be achieved through thresholding. The thresholding operation is particularly useful at rate $R = 1$, and this is borne out by the Monte Carlo simulations presented in Section VI.

E. A Product Code PVQ

For moderate sizes of dimension, significant distortion may be introduced in approximating X by $\hat{X} \in S(L, L/\lambda)$. Equivalently, the relative variance of r in (2) may be appreciable. A general VQ design can be based on concentric pyramids, with the number and location of output vectors on each pyramid selected to minimize the average distortion. To simplify design complexity, a product code [24] PVQ is designed with identical relative orientation of output vectors on each pyramid, and the pyramids are indexed by quantized versions of r. That is, a single pyramid VQ is designed for normalized vector $\hat{X}$ (with $2^{R_p L}$ output points), and a scalar quantizer designed for r (with 2^{R_r} output levels). The (PVQ) average rate per dimension R_p and the scalar quantizer rate R_r must be selected to minimize the overall distortion but are constrained to satisfy

$$R_p L + R_r = RL. \quad (27)$$

If $\hat{Y} = \mathrm{PVQ}(\hat{X})$ is the pyramid quantizer output and $\hat{r} = Q(r)$ the scalar quantizer output, then the product code output is $Y = \hat{r}\hat{Y}$.

Since the PVQ was specified in the previous section, it remains to design an optimum quantizer for r. The mse encountered in representing X by $\hat{X} \in S(L, L/\lambda)$ is approximated (using (8) and (9)) as

$$\frac{1}{L}\|X - \hat{X}\|_2^2 \simeq \frac{1}{L}\left\|\left[\left(X, \frac{S}{\|S\|_2}\right) - \frac{L\|S\|_2}{\lambda\|S\|_1}\right]\frac{S}{\|S\|_2}\right\|_2^2. \quad (28)$$

The approximation in (28) is exact if $\hat{X}$ is the orthogonal projection of X onto the appropriate face of the pyramid. Assuming that the set of vectors X for which (28) is not exact contribute negligible additional distortion, then disregarding this unimportant set, and noting that from (8) $\|S\|_1 = L$ and $\|S\|_2 = \sqrt{L}$, the expected value of (28) becomes

$$E\left\{\frac{1}{L}\|X - X\|_2^2\right\} \simeq E\left\{\frac{1}{L}\left|\frac{\sum_{i=1}^{L}|X_i|}{\sqrt{L}} - \frac{\sqrt{L}}{\lambda}\right|^2\right\} = \frac{1}{L}\operatorname{var}\left[|X_i|\right]. \quad (29)$$

From the central limit theorem, the random variable inside the absolute value signs in (29) converges in distribution to a Gaussian random variable. Provided the approximation in (29) is valid, it then follows that for large dimension it is appropriate to scalar quantize r with the Max [3] (Gaussian) quantizer. Numerical experience has shown that the Max quantizer is appropriate for small dimension as well.

Rates R_p and R_r can be selected in an optimum way by approximating the PVQ performance by (23) or (25) and by approximating the scalar quantizer distortion by the Huang and Schultheiss [25] model for the Max quantizer. Explicitly, the Max quantizer distortion is modeled as

$$D_{\mathrm{Max}}(R_r) = \sigma^2 K(R_r) 2^{-2R_r}, \quad (30)$$

with $K(R_r) \ge 1$ and $K(R_r)$ selected to fit the Max mse data [3].

Using (3), (25), and (30), the overall product code PVQ (PCPVQ) distortion is

$$D_{\mathrm{PCPVQ}}(R) = D_{\mathrm{PVQ}T}(R_p) + \frac{1}{L\lambda^2}K(R_r)2^{-2R_r}, \quad (31)$$

and rates R_p and R_r must be selected to minimize (31) subject to (27). This optimum rate allocation is easily computed as

$$R_p = \frac{L}{L+1}R - \frac{1}{2(L+1)}G(R_p),$$
$$R_r = \frac{L}{L+1}\left(R + \frac{1}{2}G(R_p)\right) \quad (32)$$

where

$$G(R_p) = \log_2\left\{\frac{3e^{2\lambda T}}{e^2(1+T\lambda)^2} \cdot \left[\frac{K'(L(R - R_p))}{2L\ln 2} + K(L(R - R_p))\right]\right\}$$

and $K'(L(R - R_p)) = dK(L(R - R_p))/dR_p$.

V. Other Sources

The basic PVQ approach may be extended to other memoryless sources (or stationary sources with memory, but treated as memoryless) by modifying parameter λ according to

$$\frac{1}{\lambda} = E[|X_i|]. \tag{33}$$

That is, for an arbitrary source (with finite variance) that satisfies the ergodic property

$$\frac{\|X\|_1}{L} \to E[|X_i|],$$

the pyramid $S(L, L/\lambda)$, with $1/\lambda$ as in (33), is appropriate for source coding. Further, provided that the approximation in (28) is valid, (29) demonstrates that asymptotic in dimension there is negligible distortion introduced in approximating X by the vector $\hat{X} \in S(L, L/\lambda)$ that is closest to X. Significantly, however, the distribution of $\hat{X}$ on $S(L, L/\lambda)$ is generally nonuniform, so that finding a good codeword assignment for $\hat{X}$ may be extremely difficult. Despite this difficulty, if (29) is valid, then it still follows that for a wide variety of memoryless sources any product code based on the pyramid structure should treat $\|X\|_1/\sqrt{L}$ as a Gaussian random variable (for large L), and consequently, Max quantization is still appropriate for radial parameter r.

Regardless of the validity of (28) and (29), for a class of distortion criteria (including the mse distortion measure), simply scaling X to a point on the pyramid (rather than finding the closest point) is a sufficient first step in constructing a PVQ source code. This is established by the following theorem.

Theorem 3: Let X_n be a stationary ergodic source satisfying $E[|X_n|^\nu] < \infty$ for some fixed value of ν, $1 \le \nu < \infty$, and

$$\lim_{L\to\infty} \frac{1}{L}\sum_{i=1}^{L} |X_i|^\beta = E\left[|X_1|^\beta\right]$$

with probability 1 for $\beta = 1, \nu$. For any α satisfying $0 < \alpha \le \nu$, if $\hat{X} = LE[|X|]X/\|X\|_1$, then

$$\frac{1}{L}\|X - \hat{X}\|_\nu^\alpha \to 0 \quad \text{as} \quad L \to \infty.$$

Proof: Let $\hat{X} = XLE[|X|]/\|X\|_1$. Then

$$\begin{aligned}\frac{1}{L}\|X - \hat{X}\|_\nu^\alpha &= \frac{1}{L}\left\| \frac{\|X\|_1 \cdot \hat{X}}{LE[|X|]} - \hat{X}\right\|_\nu^\alpha \\ &= \frac{1}{E[|X|]^\alpha}\left|\frac{\|X\|_1}{L} - E[|X|]\right|^\alpha \cdot \frac{1}{L}\left(\sum_{i=1}^{L}|\hat{X}_i|^\nu\right)^{\alpha/\nu} \\ &\le \frac{1}{E[|X|]^\alpha}\left|\frac{\|X\|_1}{L} - E[|X|]\right|^\alpha \left(\frac{1}{L}\sum_{i=1}^{L}|\hat{X}_i|^\nu\right)^{\alpha/\nu}.\end{aligned}$$

The latter term converges to $E[|X|^\nu]^{\alpha/\nu} < \infty$, the first term is clearly bounded, and by the ergodic assumption the middle term converges to zero as L increases.

The development of Section IV is now extended to several sources of interest. The product code PVQ may then be easily determined by minimizing the equivalent of (31).

A. Gaussian

For the Gaussian source with variance σ^2,

$$\frac{1}{\lambda} = E[|X|] = \sigma\sqrt{\frac{2}{\pi}}$$

or, with thresholding,

$$\begin{aligned}\frac{1}{\lambda} &= E\left[|X|\,\middle|\,|X| > T\right] \cdot P_r[|X| > T] \\ &= \sigma\sqrt{\frac{2}{\pi}}\exp\left\{\frac{-T^2}{2\sigma^2}\right\}.\end{aligned}$$

The PVQ distortion of (22) is then extended to

$$\begin{aligned}D_{\text{PVQ}T}(R) \simeq{} & \frac{2e^2\sigma^2}{3\pi}e^{-T^2/\sigma^2}2^{-2R} \\ & + \sigma^2 P_r(|X| \le T) - \sqrt{\frac{2}{\pi}}\sigma T e^{-T^2/2\sigma^2}, \end{aligned} \tag{34}$$

and the minimizing threshold value satisfies

$$\sqrt{\frac{2}{\pi}}\frac{2e^2}{3}\exp\left\{\frac{-T^2}{2\sigma^2}\right\}2^{-2R} - \frac{T}{\sigma} = 0. \tag{35}$$

As before, if $T > 0$, then (34) is not very accurate for very small rates but is reasonably accurate for $R \ge 1$.

An alternative approach to VQ design, particularly for a Gaussian source, would be to design a spherical VQ (SVQ) with output points corresponding to the lattice points that also lie on a sphere in L-dimensional space. Sloane's sphere packing results [27] would be particularly useful for such a design. If the cubic lattice (or the A_n or D_n lattices) were used for the design, then by replicating the analysis of (18)–(23), for large dimension the expected distortion of the SVQ would be

$$D_{\text{SVQ}}(R) \simeq \frac{2\pi e}{12}\sigma^2 2^{-2R}. \tag{36}$$

Comparing (36) to (34) with $T = 0$, the spherical approach is superior to the pyramid approach by a factor of $\pi^2/4e$ (about 0.42 dB). Note, however, that (36) will remain the same for SVQ encoding of *any* other source (of the same variance). In particular, for the Laplacian source, comparing (36) to (23), the PVQ approach is superior by a factor of e/π (about 0.63 dB).

B. Uniform

Let X be uniformly distributed on $(-\Delta/2, \Delta/2)$, with variance $\sigma^2 = \Delta^2/12$. Then $E[|X|] = \Delta/4$, and (23) becomes

$$D_{\mathrm{PVQ}}(R) = \frac{e^2}{4}\sigma^2 2^{-2R}. \quad (37)$$

With thresholding, analogous to (25) the distortion is

$$D_{\mathrm{PVQ}T}(R) = \frac{e^2}{e}\left[\frac{\sqrt{e}\,\sigma}{2} - \frac{T^2}{2\sqrt{3}\,\sigma}\right]^2 2^{-2R} + \frac{T^3}{3\sqrt{3}\,\sigma} \quad (38)$$

where the minimizing value of T satisfies

$$T = \frac{e^2}{3}\left[\sqrt{3}\,\sigma - \frac{T^2}{\sqrt{3}\,\sigma}\right]2^{-2R}.$$

From (37) the PVQ obviously provides mse distortion considerably larger than that of the optimum scalar quantizer and so is an inappropriate choice for encoding a uniform source. The difficulty is that the distribution of $\hat{X}$ is decidedly nonuniform on the pyramid. Comparing (37) with (36), the SVQ approach is superior (but also considerably worse than optimum scalar quantization) by a factor of $3e/2\pi$ (about 1.1 dB).

C. Gamma

The memoryless gamma source has density [28]

$$p_X(x) = \frac{3^{\frac{1}{4}}}{\sqrt{8\pi\sigma|x|}}\exp\left[-\frac{\sqrt{3}|x|}{2\sigma}\right]$$

and mean absolute value

$$\frac{1}{\lambda} = E[|X|] = \frac{\sigma}{\sqrt{3}}.$$

Hence even without the thresholding operation, (23) becomes

$$D_{\mathrm{PVQ}}(R) = \frac{e^2}{9}\sigma^2 2^{-2R}, \quad (39)$$

which is significantly better than the distortion of the optimum scalar quantizer. Analogous to (25) and (26), the optimum threshold parameter can be selected numerically.

As with the Gaussian source, comparing the PVQ performance with that of an SVQ (compare (39) with (36)), the PVQ is superior by a factor of $2e/3\pi$ (about 2.4 dB).

D. Generalized Gaussian

The generalized Gaussian density is [26]

$$p_X(x) = C_1 e^{-C_2|x|^\nu}, \qquad \nu > 0$$

where

$$C_1 = \nu\eta(\sigma,\nu)/2\Gamma(1/\nu)$$

$$C_2 = [\eta(\sigma,\nu)]^\nu$$

$$\eta(\sigma,\nu) = \frac{1}{\sigma}\left[\frac{\Gamma(3/\nu)}{\Gamma(1/\nu)}\right]^{1/2}$$

$$\mathrm{var}(X) = \sigma^2.$$

The entropy is easily computed to be

$$h = C_2 E[|X|^\nu] - \ln C_1,$$

and the mean absolute value is

$$E[|X|] = \frac{2C_1\Gamma(2/\nu)}{\nu C_2^{2/\nu}}.$$

For large rate and dimension the expected mse distortion for PVQ encoding is then

$$D_{\mathrm{PVQ}}(R,\nu) = \frac{e^2}{3}\left[\frac{2C_1\Gamma(2/\nu)}{\nu C_2^{2/\nu}}\right]^2 2^{-2R}$$

or, in terms of rate,

$$R_{\mathrm{PVQ}}(D,\nu) = \frac{1}{2}\log_2\frac{e^2}{3D}\left[\frac{2C_1\Gamma(2/\nu)}{\nu C_2^{2/\nu}}\right]^2. \quad (40)$$

The Shannon lower bound for the memoryless generalized Gaussian source is given by

$$R_{\mathrm{SLB}}(D,\nu) = h - \tfrac{1}{2}\log 2\pi e D. \quad (41)$$

Comparing R_{SLB} to R_{PVQ} indicates that for a given (small) distortion the PVQ requires a rate $R_{\mathrm{PVQ}} = R_{\mathrm{SLB}} + \Delta R_{\mathrm{PVQ}}$ larger than the minimum rate of the Shannon lower bound. Specifically,

$$\Delta R_{\mathrm{PVQ}} = \frac{1}{2}\log_2\frac{2\pi e^3 c_1^2}{3e^{2C_2\rho}}\left(\frac{2C_1\Gamma(2/\nu)}{\nu C_2^{2/\nu}}\right)^2 \quad (42)$$

where $\rho = E[|X|^\nu]$ and is plotted in Fig. 2 as a function of ν. For $\nu = 1$ the generalized Gaussian density coincides with the Laplacian density and $\Delta R_{\mathrm{PVQ}} = 0.255$ bits, as was shown earlier. For $\nu \neq 1$ the PVQ requires (for the same distortion) a slightly larger rate than does scalar quantization with entropy coding but does not require variable length codewords.

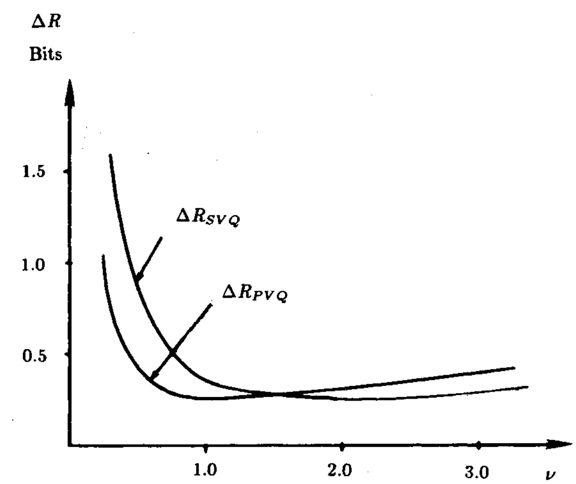

Fig. 2. ΔR_{PVQ}, ΔR_{SVQ} versus ν for generalized Gaussian source.

Equations (40)–(42) can be repeated for the (uniform lattice) SVQ, and the resulting ΔR_{SVQ} is also shown in Fig. 2. For large values of parameter ν the value of ΔR_{SVQ} is smaller than ΔR_{PVQ}, but by no more than 0.188 bits (corresponding to $\nu \to \infty$ and the uniform density). For small ν the PVQ is superior. Interestingly, in transform coding both speech [32] and image [33] sources have transform coefficients that are well-modeled by Laplacian or gamma densities. Fig. 2 then implies that the PVQ is perhaps a better choice than the SVQ for the transform coding of such sources.

VI. PERFORMANCE

The performance of the PVQ was evaluated by Monte Carlo simulation for memoryless Laplacian, gamma, and Gaussian sources. The average mse was computed for a block of 1000 random (i.i.d., zero-mean, and unit variance component) vectors generated by a random number generator and then averaged over 100 blocks of such vectors. The mse performance is summarized in Tables I–III and Figs. 3 and 4.

The Monte Carlo simulations verified the analytical expressions approximating the PVQ performance that were obtained in Sections IV and V. However, the PVQ encoding of the gamma source was better than anticipated (from (39)) for the low rates simulated. For the gamma source, the nonuniformity of the distribution of $\hat{X}$ on the pyramid surface is apparently well matched to the PVQ encoding procedure. As evidenced by Fig. 4, the PVQ performance is quite close to the rate distortion bound and significantly better than the mse of the optimum scalar quantizer. In the high rate case, comparing (39) to the known optimum scalar quantizer [28] indicates that the PVQ provides an improvement of 8.40 dB.

For the Laplacian source, the Monte Carlo simulations yielded mse values very close to those expected from the asymptotic performance expression (23). At a rate of 1 bit/dimension, the thresholding is very important in reducing the mse. However, for larger rates (3 bits/dimension or above), thresholding offers negligible improvement in performance. For high rates, comparing (23) to the optimum scalar quantizer [28], the PVQ provides a 5.64-dB improvement in mse performance.

TABLE I
MONTE CARLO SIMULATIONS OF THE PYRAMID VECTOR QUANTIZER MSE PERFORMANCE FOR A ZERO-MEAN UNIT-VARIANCE MEMORYLESS LAPLACIAN SOURCE ENCODED AT AN AVERAGE RATE OF R BITS / DIMENSION[a]

	Mean Square Error		
	$R = 1$ ($T = 0.75$, $\gamma = 0.73$)	$R = 2$ ($T = 0.4$, $\gamma = 0.94$)	$R = 3$ ($T = 0.17$, $\gamma = 0.98$)
PVQ ($L = 16$)	0.310 ($K = 4$, $N = 0$)	0.0907 ($K = 10$, $N = 3$)	0.0233 ($K = 22$, $N = 4$)
PVQ ($L = 32$)	0.295 ($K = 7$, $N = 2$)	0.0778 ($K = 21$, $N = 3$)	0.0210 ($K = 45$, $N = 4$)
PVQ ($L = 48$)	0.283 ($K = 11$, $N = 0$)	0.0773 ($K = 31$, $N = 3$)	0.0202 ($K = 68$, $N = 4$)
PVQ ($L = 64$)	0.281 ($K = 14$, $N = 2$)	0.0773 ($K = 41$, $N = 3$)	0.0198 ($K = 91$, $N = 4$)
Distortion-rate bound [30]	0.2178	0.0542	0.0136
Optimum scalar quantizer [29]	0.5	0.1765	0.0514
$D_{PVQ}(R)$	0.3079	0.07697	0.01924
$D_{PVQT}(R)$	0.2464	0.0750	0.01921

[a]For each product code PVQ, the parameters are L, K, N, T, and γ, where L is the dimension, K the pyramid index in (11), N the number of scalar (Max) quantizer bits for radial parameter r, T the threshold in (15), and γ the scale factor in (14).

TABLE II
MONTE CARLO SIMULATIONS OF THE PYRAMID VECTOR QUANTIZER MSE PERFORMANCE FOR A ZERO-MEAN UNIT-VARIANCE MEMORYLESS GAMMA SOURCE ENCODED AT AN AVERAGE RATE OF R BITS / DIMENSION[a]

	Mean Square Error		
	$R = 1$ ($T = 0.45$, $\gamma = 0.82$)	$R = 2$ ($T = 0.2$, $\gamma = 0.92$)	$R = 3$ ($T = 0$, $\gamma = 0.98$)
PVQ ($L = 16$)	0.218 ($K = 3$, $N = 3$)	0.0503 ($K = 10$, $N = 3$)	0.0130 ($K = 22$, $N = 4$)
PVQ ($L = 32$)	0.170 ($K = 7$, $N = 2$)	0.0397 ($K = 21$, $N = 3$)	0.0105 ($K = 45$, $N = 4$)
PVQ ($L = 48$)	0.157 ($K = 10$, $N = 3$)	0.0374 ($K = 31$, $N = 3$)	0.00988 ($K = 68$, $N = 4$)
PVQ ($L = 64$)	0.148 ($K = 14$, $N = 2$)	0.0370 ($K = 41$, $N = 3$)	0.00959 ($K = 91$, $N = 4$)
Distortion-rate bound [30]	0.140	0.030	0.00693
Optimum scalar quantizer [30]	0.665	0.233	0.0713
$D_{PVQ}(R)$	0.205	0.0513	0.0128

[a]The product code parameters L, K, N, T, and γ are as in Table I.

TABLE III
MONTE CARLO SIMULATIONS OF THE PYRAMID VECTOR QUANTIZER MSE PERFORMANCE FOR A ZERO-MEAN UNIT-VARIANCE MEMORYLESS GAUSSIAN SOURCE ENCODED AT AN AVERAGE RATE OF R BITS / DIMENSION[a]

	Mean Square Error		
	$R = 1$ ($T = 0.75$, $\gamma = 0.64$)	$R = 2$ ($T = 0.28$, $\gamma = 0.92$)	$R = 3$ ($T = 0$, $\gamma = 0.98$)
PVQ ($L = 16$)	0.358 ($L = 15$, $K = 4$, $N = 0$)	0.111 ($K = 11$, $N = 1$)	0.0257 ($K = 25$, $N = 2$)
PVQ ($L = 32$)	0.353 ($L = 24$, $K = 6$, $N = 0$)	0.107 ($K = 21$, $N = 3$)	0.0263 ($K = 47$, $N = 2$)
PVQ ($L = 48$)	0.355 ($L = 47$, $K = 11$, $N = 0$)	0.104 ($K = 32$, $N = 1$)	0.0251 ($K = 71$, $N = 2$)
PVQ ($L = 64$)	0.357 ($L = 61$, $K = 14$, $N = 0$)	0.1048 ($K = 43$, $N = 0$)	0.0250 ($K = 94$, $N = 2$)
Distortion-rate bound [1]	0.25	0.0625	0.016525
Optimum scalar quantizer [3]	0.3634	0.1175	0.03454
$D_{PVQ}(R)$	0.392	0.0980	0.0245
$D_{PVQT}(R)$	0.301	0.0960	0.0244

[a]The product code parameters L, K, N, T, and γ are as in Table I.

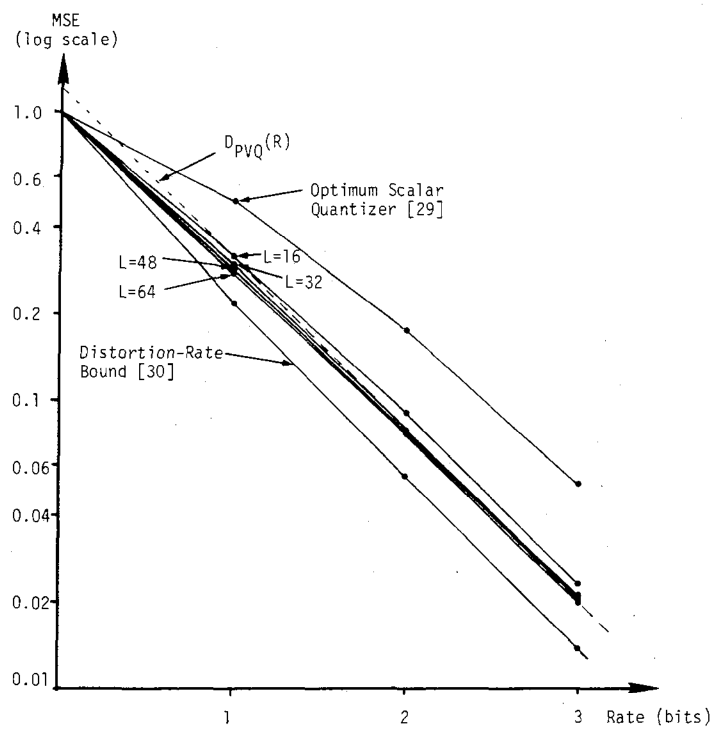

Fig. 3. Normalized mse versus rate for L-dimensional PVQ encoding of memoryless Laplacian source.

For the Gaussian source, the PVQ offers little improvement over the optimum scalar quantizer at a rate of 1 bit/sample, but more significant reduction in mse for larger rates. For large rates, the Max scalar quantizer provides an mse distortion that is a factor of 2.72 larger than the rate distortion function [25], while for large dimension the PVQ provides an mse (from (34) with $T = 0$) of $2e^2/3\pi \simeq 1.568$ times the rate-distortion function. Thus for large rates the PVQ provides an improvement of 2.39 dB over the optimum scalar quantizer.

In Fischer and Dicharry [16], LBG [9] algorithm-based VQ performance is reported for memoryless Gaussian, Laplacian, and gamma sources. The results in [16] are consistent with those of Tables I–III, but the former results were limited (because of the computational burden of the LBG method) to either a rate of 1 bit/dimension and six or fewer dimensions, or two dimensions and 5 bits or less. For Laplacian and gamma sources, the 1-bit/dimension six-dimensional VQ's of [16] provide mse distortions only slightly (less than ten percent) larger than those of the 16-dimensional PVQ. Although the PVQ is based on the simple (and asymptotically suboptimum) cubic lattice, as evidence by Figs. 3 and 4 the advantage of using large dimensions is considerable.

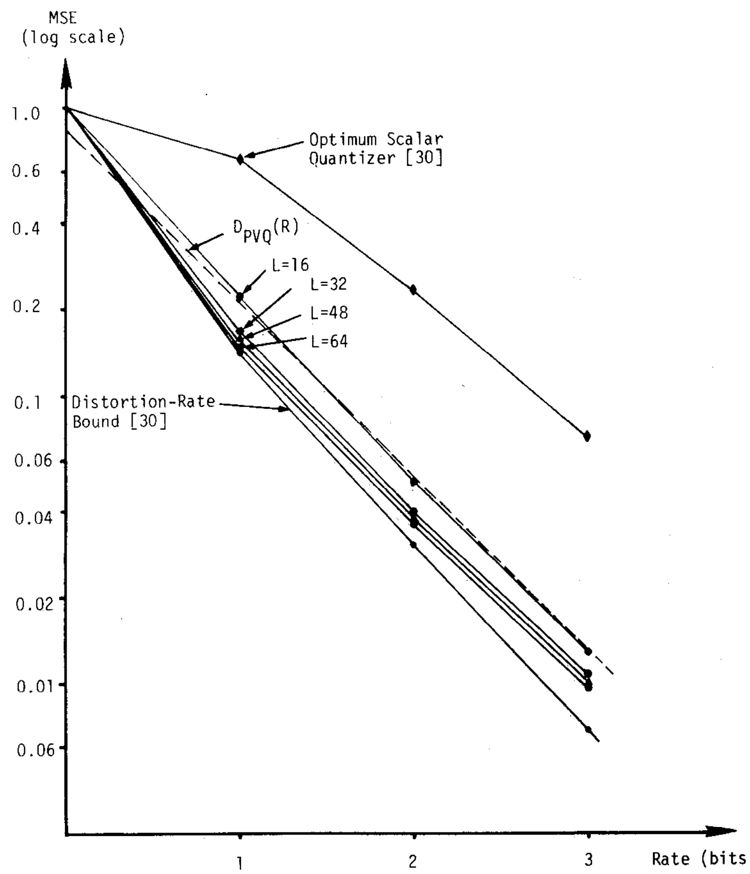

Fig. 4. Normalized mse versus rate for L-dimensional PVQ encoding of memoryless source with gamma pdf.

In Sayood, Gibson, and Rost (SGR) [20], four-dimensional A_n^* lattice quantizers were designed for rate 1 and 2 bits/dimension encoding of memoryless Laplacian and gamma sources. The PVQ mse performance in Tables I and II is uniformly better than the results in [20], due to the larger dimensions considered. For example, in [20] a 1-bit/dimension Laplacian VQ provided an SNR of 3.97 dB, while for the PVQ the SNR is 5.09 dB (for $L = 16$) or 5.51 dB (for $L = 64$). From the results in [17] it is clear that for moderate sizes of dimension (say, $L \sim 16$) the A_n^* lattice is superior to the uniform lattice, so that if the SGR quantizer could be extended to 16 dimensions the performance should be slightly better than that of the PVQ.

The SGR lattice quantizer (and many other lattice quantizers) differs from the PVQ in two important respects. First, the SGR approach uses a lattice centered at the origin and adjusts the lattice density (i.e., radially scales all the lattice points) to "match" the probability density function (pdf) of the source to be encoded. The PVQ is based on Shannon's notion of a region of high probability [1] and the sphere hardening property [34] and assigns the (lattice point) codewords to a specific surface (the pyramid) in L-dimensional space, with no codewords either inside or outside this surface. The product code PVQ uses a radial parameter (the l_1 norm) to radially scale the codewords on a nominal pyramid, and this is different (and simpler) than classifying points in a lattice as lying on concentric (pyramid) shells. Second, the SGR approach provides a general encoding algorithm based on the symmetry of the lattice, but for large code book sizes does not address either how the lattice can be effectively truncated to maintain a fixed coding rate, nor how the lattice codewords can be enumerated to form binary codewords for digital transmission. The PVQ approach solved these problems be finding an explicit formula for computing the number of lattice points on a pyramid and by developing simple enumeration encoding and decoding algorithms for representing each lattice point codeword as a binary codeword.

In Abut *et al.* [12], LBG algorithm-based vector quantization is reported for autoregressive (AR) sources with an i.i.d. Laplacian driving process. These results are not directly comparable with the present PVQ performance because of the source AR structure. For example, at a rate of 1 bit/dimension and scalar quantization, the SNR in [12, Fig. 1] is approximately 3.8 dB. However, the optimum 1-bit scalar quantizer for a memoryless Laplacian source is known [29] to provide an SNR value of only 3.0 dB. Similarly, for two dimensions, [12] has an SNR value of about 4.3 dB, while the best known two-dimensional 2-bit Laplacian VQ [16] provides an SNR of 3.67 dB. Basically, the AR model used in [12] causes the resulting source to be rather unlike the Laplacian driving process (and presumably "more Gaussian"), so that the VQ performance is better than that for the corresponding memoryless Laplacian source. Interestingly, by comparing [12] and [16] for the Laplacian source, and [11] and [9] for the Gaussian source, it appears that as the vector dimension increases, the mse provided by VQ's for sources with memory approaches the rate-distortion bound at a faster rate than does the mse provided by VQ's for the corresponding memoryless source.

Most of the LBG algorithm-based VQ's discussed in the literature [9], [11]–[16] consider a rate-dimension product of less than eight, with product code VQ's [24] extending the rate-dimension product to 12. For large rate-dimension products, the LBG method is computationally prohibitive because of the exponential growth in required VQ design computations (for a training sequence length proportional to the number of VQ output vectors) and the complexity of encoding. The encoding complexity can be reduced significantly by using tree search techniques, but the resulting VQ performance is generally somewhat degraded [15]. Particularly for 2 or 3 bits per dimension encoding (and larger rates), the PVQ offers significant advantages over the LBG algorithm VQ's in simplifying both the VQ design process and the encoding complexity.

VII. Conclusion

By extending the geometric approach of Sakrison [21], the geometric properties of i.i.d. Laplacian source were developed and used to prove a source coding theorem. Analogous to entropy coding of discrete ensembles, the geometric approach attempts to locate VQ output vectors solely in the region of L-dimensional space of high probability. For the memoryless Laplacian source this implies placing the VQ output vectors near the surface of an L-dimensional hyperpyramid. This pyramid is indexed by the differential entropy of the source, or, equivalently, by the mean absolute value of the source.

Motivated by the source coding theorem and geometric structure, a pyramid vector quantizer was developed. The PVQ is based on the cubic lattice points that also lie on the appropriate pyramid. A simple PVQ encoding algorithm was derived that is implementable for arbitrary vector dimension. The PVQ was generalized to a product code version and analytical expressions developed for both the PVQ and product code PVQ to approximate the large-dimensional mse performance. The pyramid vector quantizer's mse performance was evaluated by Monte Carlo simulation and compared with the analytical performance expressions and several VQ results in the literature. Particularly, for rates of 2 bits/dimension or larger, because the PVQ can take advantage of encoding large-dimensional vectors (dimension sizes currently beyond the computational feasibility of the LBG [9] method), the PVQ and product code PVQ offer attractive alternatives to the LBG algorithm-based vector quantizers for the memoryless sources considered.

Acknowledgment

The author would like to thank Prof. J. D. Gibson for many useful discussions, Prof. P. F. Dahm for a critical reading of the manuscript, and the anonymous reviewers for their helpful comments.

Appendix I
Proof of Theorem 1

Let $\epsilon_\alpha \triangleq \epsilon^{1/\alpha}$, so that

$$\frac{1}{L}\|X - \hat{X}\|_\nu < \epsilon_\alpha \Leftrightarrow \frac{1}{L}\|X - \hat{X}\|_\nu^\alpha < \epsilon. \tag{A1}$$

Certainly, (A1) is satisfied if $|X_i - \hat{X}_i| < \epsilon_\alpha$ for each $i = 1,\cdots, L$. For each vector X define set Υ as

$$\Upsilon = \{\hat{x}\colon |X_i - \hat{x}_i| < \epsilon_\alpha, \quad \text{for } i = 1,\cdots, L\}.$$

Since $p_X(x_i)$ is positive for $-\infty < x_i < \infty$, define α_i^S and α_i^I, $i = 1,\cdots, L$, as

$$\alpha_i^S = \min\left(\sup_{\{\hat{x}_i:\, |X_i - \hat{x}_i| < \epsilon_\alpha\}} \frac{p_X(\hat{x}_i)}{p_X(X_i)}, \beta \right)$$

and

$$\alpha_i^I = \max\left(\inf_{\{\hat{x}_i:\, |X_i - \hat{x}_i| < \epsilon_\alpha\}} \frac{p_X(\hat{x}_i)}{p_X(X_i)}, \frac{1}{\beta} \right)$$

for some fixed real number β satisfying $1 < \beta < \infty$. Clearly,

$$0 < \frac{1}{\beta} \le \alpha_i^I \le 1 \le \alpha_i^S \le \beta < \infty,$$

so that $E[\alpha_i^I]$, $E[\alpha_i^S]$, var $[\alpha_i^I]$, and var $[\alpha_i^S]$ are bounded.

From the continuity of $p_X(\cdot)$, it follows that as $\hat{x}_i$ varies over the set $\{\hat{x}_i\colon |X_i - \hat{x}_i| < \epsilon_\alpha\}$, variable $\alpha_i = p_X(\hat{x}_i)/p_X(X_i)$ can assume any value in the range (α_i^I, α_i^S). For each $i = 1,\cdots, L$, an acceptable $\hat{x}_i$ exists such that $p(\hat{x}_i)$ can assume any value in the range

$$\alpha_i^I p(X_i) < p(\hat{x}_i) < \alpha_i^S p(X_i).$$

Hence an $\hat{X} \in \Upsilon$ exists such that $(1/L)\log f_X(x)$ can assume any value in the range

$$\frac{\log f_X(X)}{L} - \frac{1}{L}\sum_{i=1}^{L} \log\left(\frac{1}{\alpha_i^I}\right) < \frac{\log f_X(\hat{X})}{L} < \frac{\log f_X(X)}{L} + \frac{1}{L}\sum_{i=1}^{L} \log\left(\alpha_i^S\right).$$

Asymptotically, both sums converge as

$$\frac{1}{L}\sum_{i=1}^{L} \log\frac{1}{\alpha_i^I} \xrightarrow{\text{a.s.}} E\left[\log\frac{1}{\alpha_i^I}\right] > 0 \tag{A2}$$

and

$$\frac{1}{L}\sum_{i=1}^{L} \log\alpha_i^S \xrightarrow{\text{a.s.}} E\left[\log\alpha_i^S\right] > 0,$$

so that there exists a positive integer, say L_1, such that if $L > L_1$, then each sum is within ϵ_1 of its limit for all X, except on a set of total probability Δ_1. Further, choose L_1 large enough so that ϵ_2, defined as

$$\epsilon_2 = \min\left(E\left[\log\frac{1}{\alpha_i^I}\right], E\left[\log\alpha_i^S\right]\right) - \epsilon_1$$

is positive. It follows that $\hat{X} \in \Upsilon$ exists such that $(1/L)\log f_X(\hat{X})$ can assume any value in

$$\left(\frac{\log f_X(X)}{L} - \epsilon_2, \frac{\log f_X(X)}{L} + \epsilon_2\right).$$

However, for any $\epsilon_2 > 0$ and $\Delta_2 > 0$, the convergence in (7) implies that a positive integer L_2 exists such that if $L > L_2$, then

$$\frac{\log f_X(X)}{L} - \epsilon_2 < -h(x) < \frac{\log f_X(X)}{L} + \epsilon_2$$

except on a set of total probability Δ_2. Hence (for $L > \max(L_1, L_2)$) for all X (except on a set of probability $\delta \le \Delta_1 + \Delta_2$), an $\hat{X} \in \Upsilon$ (so that $(1/L)\|X - \hat{X}\|_\nu < \epsilon_\alpha$) exists such that $(1/L)\log f_X(\hat{X}) = -h(x)$, implying $\hat{X} \in S(L, L/\lambda)$.

Appendix II
Proof of Theorem II

The proof parallels the argument of Sakrison [21]. Using the source code construction of 1)–3) in Section III, the mae (per dimension) may be upper-bounded as

$$E\left[\frac{1}{L}\|X - y\|_1\right] \le \frac{\epsilon}{L}(1 - P(\epsilon)) + \frac{2}{\lambda}P(\epsilon) + \frac{1}{L}E[\|X - \hat{X}\|_1]$$

where $P(\epsilon)$ is the probability that $\hat{X}$ is not within one of the representation regions. The middle term follows since $\hat{X}$ is certainly within a distance $2L/\lambda$ from some representation point. The last term has already been shown to asymptotically decrease to zero and so is dropped.

Probability $P(\epsilon)$ may be calculated as

$$P(\epsilon) = \prod_{i=1}^{2^{RL}}\left[1 - \frac{\text{area of intersection of region } i \text{ and } S(L, L/\lambda)}{\text{total area of } S(L, L/\lambda)}\right],$$

or, since the points are selected at random in 2)

$$P(\epsilon) = \left[1 - \frac{\text{area of intersection}}{\text{area of } S(L, L/\lambda)}\right]^{2^{RL}}. \quad \text{(A3)}$$

Since $S^*(L, \epsilon)$ is the same pyramid shape as $S(L, L/\lambda)$, the smallest area of maximum intersection between $S^*(L, \epsilon)$ and $S(L, L/\lambda)$ occurs if a single face of $S(L, L/\lambda)$ acts as a hyperplane slicing through the centroid of $S^*(L, \epsilon)$. This can be seen by the following argument. The pyramid $S(L, L/\lambda)$ has 2^L faces, $2L$ vertices, and at each vertex 2^{L-1} faces meet. The vertices may be covered by $2L$ representation pyramids, each centered at a point with a single nonzero coordinate of magnitude $L/\lambda - \epsilon$. Any randomly chosen point on the pyramid surface within an l_1 norm distance of 2ϵ from a vertex may then be represented with such a region, and the resulting area of intersection between the representation pyramid and $S(L, L/\lambda)$ is $A(L, \epsilon)/2$ (for $\epsilon < L/\lambda$). Any other point selected at random on $S(L, L/\lambda)$ must be at least an l_1 norm distance of 2ϵ from a vertex, and thus within 2ϵ of at most two faces. Such a point can then be covered by an appropriately placed representation pyramid that intersects with only one face of $S(L, L/\lambda)$. Since this latter area of intersection is smaller than $A(L, \epsilon)/2$, the area of intersection in (A3) is then bounded from below by the $(L-1)$-dimensional volume of the intersection of $S^*(L, \epsilon)$ and an $(L-1)$-dimensional hyperplane parallel to a face of $S^*(L, \epsilon)$ and passing through the centroid. By direct calculation for $2 \le L \le 6$, we determine a bound for this volume as

$$\text{area of intersection} \ge \frac{1}{\sqrt{L}} V(L-1, \epsilon), \quad \text{(A4)}$$

where equality occurs if $L = 2$ and the bound becomes looser as L increases. Using (4) and (A4) in (A3) yields the bound for $P(\epsilon)$

$$P(\epsilon) \le \left[1 - \frac{\dfrac{2^{L-1}\epsilon^{L-1}}{\sqrt{L}\,\Gamma(L)}}{\dfrac{2^L\sqrt{L}\left(\dfrac{L}{\lambda}\right)^{L-1}}{\Gamma(L)}}\right]^{2^{RL}}. \quad \text{(A5)}$$

The rate-distortion function for an i.i.d. Laplacian source is well-known [23] to be

$$R(D) = \log\frac{1/\lambda}{D},$$

so that

$$D = \frac{1}{\lambda}2^{-R}. \quad \text{(A6)}$$

The distortion in L dimensions ϵ may then be expressed as

$$\epsilon = \frac{L(1+\delta)}{\lambda}2^{-R} \quad \text{(A7)}$$

for some $\delta > 0$. Substituting (A7) into (A5) yields

$$P(\epsilon) \le \left[1 - \frac{(1+\delta)^{L-1}}{2L}\cdot 2^{-R(L-1)}\right]^{2^{RL}}. \quad \text{(A8)}$$

For large L, (A6) becomes approximately

$$P(\epsilon) \simeq \exp\left[-\frac{(1+\delta)^{L-1}2^R}{2L}\right],$$

so that for arbitrary $\delta > 0$ the probability that $\hat{X}$ is not within an mae distance of ϵ from some representation point y_i goes to zero as dimension L increases. Hence the encoding scheme of 1)–3) is asymptotically (in L) optimum and achieves a distortion arbitrarily close to the rate-distortion bound.

Appendix III

The number of integer coordinate vectors on pyramid $S(L, K)$ is denoted $N(L, K)$ and derived as follows. All of the $N(L, K)$ possible vectors on $S(L, K)$ are of the form

$$x = \begin{cases} i \\ x_2 \\ \vdots \\ x_L \end{cases}, \qquad \text{for } i = 0, \pm 1, \cdots, \pm K.$$

The portion of x corresponding to $(x_2, \cdots, x_L)$ has $N(L-1, K - |i|)$ possible permutations. This implies

$$N(L, K) = 2(N(L-1, K-1) + N(L-1, K-2) + \cdots + N(L-1, 1) + 1) + N(L-1, K). \quad \text{(A9)}$$

Similarly,

$$N(L, K-1) = 2(N(L-1, K-2) + \cdots + N(L-1, 1) + 1) + N(L-1, K-1). \quad \text{(A10)}$$

Combining (A9) and (A10) yields the desired result,

$$N(L, K) = N(L, K-1) + N(L-1, K-1) + N(L-1, K).$$

References

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

[2] S. P. Lloyd, "Least squares quantization in PCM," unpublished memorandum, Bell Laboratories, 1957; see *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 129–137, 1982.

[3] J. Max, "Quantizing for minimum distortion," *IRE Trans. Inform. Theory*, vol. IT-6, pp. 7–12, Mar. 1960.

[4] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 485–497, 1984.

[5] P. L. Zador, "Asymptotic quantization error of continuous signals and quantization dimension," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 139–149, Mar. 1982. (The paper first appeared as an unpublished Bell Laboratories memorandum in 1966.)

[6] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 373–380, July 1979.

[7] ——, "On the structure of vector quantizers," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 157–166, Mar. 1982.

[8] Y. Yamada, S. Tazaki, and R. M. Gray, "Asymptotic performance of block quantizers with difference distortion measures," *IEEE Trans. Inform. Theory*, vol. IT-26, pp. 6–14, Jan. 1980.

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

[10] R. M. Gray, J. C. Kieffer, and Y. Linde, "Locally optimal block quantizer design," *Inform. Contr.*, vol. 45, pp. 178–198, 1980.

[11] R. M. Gray and Y. Linde, "Vector quantizers and predictive quantizers for Gauss-Markov sources," *IEEE Trans. Commun.*, vol. COM-30, pp. 381–389, Feb. 1982.

[12] H. Abut, R. M. Gray, and G. Rebolledo, "Vector quantization of speech and speech-like waveforms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 423–435, June 1982.

[13] V. Cuperman and A. Gersho, "Adaptive differential vector coding of speech," in *Proc. Globecom '82*, Nov. 1982, pp. 1092–1096.

[14] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, vol. 1, pp. 4–29, Apr. 1984.

[15] A. Gersho and V. Cuperman, "Vector quantization: A pattern-matching technique for speech coding," *IEEE Commun. Mag.*, vol. 21, pp. 15–21, Dec. 1983.

[16] T. R. Fischer and R. M. Dicharry, "Vector quantizer design for memoryless Gaussian, gamma, and Laplacian sources," *IEEE Trans. Commun.*, vol. COM-32, pp. 1065–1069, Sep. 1984.

[17] J. H. Conway and N. J. A. Sloane, "Voronoi regions of lattices, second moments of polytopes, and quantization," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 211–226, Mar. 1982.

[18] ——, "Fast quantizing and decoding algorithms for lattice quantizers and codes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 227–232, Mar. 1982.

[19] ——, "A fast encoding method for lattice codes and quantizers," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 820–824, Nov. 1983.

[20] K. Sayood, J. D. Gibson, and M. C. Rost, "An algorithm for uniform vector quantizer design," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 805–814, Nov. 1984.

[21] D. J. Sakrison, "A geometric treatment of the source encoding of a Gaussian random variable," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 481–486, May 1968.

[22] T. R. Fischer, "Geometric source coding and vector quantization," *Abstracts of Papers*, Int. Symp. Inform. Theory, Brighton, England, June 24–28, 1985.

[23] T. Berger, *Rate-Distortion Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[24] M. J. Sabin and R. M. Gray, "Product code vector quantizers for waveform and voice coding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 474–488, June 1984.

[25] J. J. Y. Huang and P. M. Schultheiss, "Block quantization of correlated Gaussian random variables," *IEEE Trans. Commun. Syst.*, vol. CS-11, pp. 289–296, Sept. 1963.

[26] W. Mauersberger, "Experimental results on the performance of mismatched quantizers," *IEEE Trans. Inform. Theory*, vol. IT-25, July 1979, pp. 381–386.

[27] N. J. A. Sloane, "Tables of sphere packing and spherical codes," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 327–338, May 1981.

[28] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[29] M. D. Paez and T. H. Glisson, "Minimum mean-squared-error quantization in speech PCM and DPCM systems," *IEEE Trans. Commun.*, vol. COM-20, pp. 225–230, Apr. 1972.

[30] P. Noll and R. Zelinski, "Bounds on quantizer performance in the low bit-rate region," *IEEE Trans. Commun.*, vol. COM-26, Feb. 1978, pp. 300–304.

[31] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*. New York: Academic, 1980.

[32] R. Zelinski and P. Noll, "Adaptive transform coding of speech signals," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-25, pp. 299–309, Aug. 1977.

[33] R. C. Reininger and J. D. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Trans. Commun.*, vol. COM-31, pp. 835–839, June 1983.

[34] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering*. New York: Wiley, 1965.