Quantization Based on a Novel Sample-Adaptive Product Quantizer (SAPQ)

Dong Sik Kim, *Member, IEEE*, and Ness B. Shroff, *Member, IEEE*

***Abstract*— In this paper, we propose a novel feedforward adaptive quantization scheme called the sample-adaptive product quantizer (SAPQ). This is a structurally constrained vector quantizer that uses *unions of product codebooks*. SAPQ is based on a concept of adaptive quantization to the varying samples of the source and is very different from traditional adaptation techniques for nonstationary sources. SAPQ quantizes each source sample using a sequence of quantizers. Even when using scalar quantization in SAPQ, we can achieve performance comparable to vector quantization (with the complexity still close to that of scalar quantization). We also show that important lattice-based vector quantizers can be constructed using scalar quantization in SAPQ. We mathematically analyze SAPQ and propose a simple algorithm to implement it. We numerically study SAPQ for independent and identically distributed Gaussian and Laplacian sources. Through our numerical study, we find that SAPQ using scalar quantizers achieves typical gains of 1–3 dB in distortion over the Lloyd–Max quantizer. We also show that SAPQ can be used in conjunction with vector quantizers to further improve the gains.**

***Index Terms*— Feedforward adaptive quantization, structurally constrained vector quantization, vector quantizer.**

I. Introduction

Block *source coding* or *vector quantization* is a mapping from $\mathbb{R}^k$ into a finite subset called a *codebook*. It is well known, from the *Source Coding Theorem*, that the average distortion of a vector quantizer (VQ) on a random vector can be decreased as the block (vector) size k gets large. The average distortion can be made to approach the distortion given by the corresponding *rate distortion function*.

We consider a sequence of random vectors $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ as the discrete-time source to be quantized, where $\boldsymbol{X}_i := (X_{1i}, \cdots, X_{ki})$ is a random vector in $\mathbb{R}^k$ and m is the *sample size* or the *adaptation period*. Suppose that $E\|\boldsymbol{X}_i\|^r < \infty$, for $i = 1, \cdots, m$, where $\|\cdot\|^r$ denotes the rth power of the l_2 norm to be used for the distortion measure. Let $\mathcal{C}_n$ denote the class of sets that take n points from $\mathbb{R}^k$, and let the sets in $\mathcal{C}_n$ be called "n-level codebooks," where each such codebook has n codewords. The quantization of $\boldsymbol{X}_i$ is the mapping of a sequence of observations of $\boldsymbol{X}_i$ to a sequence of points of C $(\in \mathcal{C}_n)$ according to a mapping called the quantizer. The average distortion achieved when a random vector $\boldsymbol{X}_i$ is quantized by a codebook C $(\in \mathcal{C}_n)$ is given by

$$D_{\mathrm{VQ}} := E\left\{\frac{1}{k} \min_{\boldsymbol{y}\in C} \|\boldsymbol{X}_i - \boldsymbol{y}\|^r\right\}. \tag{1}$$

In this k-dimensional VQ, the bit-rate (defined as bits per source point in $\mathbb{R}$) required is $R = (\log_2 n)/k$. In this paper, we focus only on block coding schemes that are based on *fixed-length coding* [20].

Let an observation of $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ be denoted by $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$, where ω is a sample point of the underlying sample space Ω; we call this observation a *sample*. Suppose that the codebooks C_i are $C_i \in \mathcal{C}_{n_i}$, for $i = 1, \cdots, m$, where n_i are positive integers. If we quantize this sample by applying the k-dimensional VQ using codebooks C_i to each $\boldsymbol{X}_i^\omega$ independently, then the overall average distortion D_{PQ} is given by

$$D_{\mathrm{PQ}} := E\left\{\frac{1}{km} \sum_{i=1}^{m} \min_{\boldsymbol{y}\in C_i} \|\boldsymbol{X}_i - \boldsymbol{y}\|^r\right\}. \tag{2}$$

We call this quantization scheme the *product quantizer* (PQ), since the quantizer is a mapping from $\mathbb{R}^{km}$ to the Cartesian product set $C_1 \times \cdots \times C_m$. The size of the product set or PQ codebook is $\prod_{i=1}^m n_i$. Hence, the total bit rate R is given by

$$R = \frac{1}{km} \log_2 \prod_{i=1}^{m} n_i. \tag{3}$$

A specific example of PQ for $k = 1$ (scalar quantization) is the pulse-code modulation (PCM) scheme, where a codebook is applied independently and identically to each random variable.

However, even if the source is independently distributed, independently quantizing each random vector of $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ is just one of many possible coding schemes. For a given bit rate, a natural question to ask is whether there exists a coding scheme that yields an average distortion that is less than the average distortion achieved by PQ. From the source coding theorem, the following method is well known for outperforming PQ. If we represent a sample of $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ by a single index taking $\prod_{i=1}^m n_i$ values (in other words, if we use the km-dimensional VQ), then we can achieve a lower distortion, for the same bit rate, than with PQ. The encoding complexity of the km-dimensional VQ is, however, extremely high, especially at high bit-rates. In order to circumvent the encoding complexity of the traditional

Manuscript received February 1, 1998; revised June 1, 1999.
The authors are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285 USA.
Communicated by R. Laroia, Associate Editor for Source Coding.
Publisher Item Identifier S 0018-9448(99)07679-8.

VQ, various modifications of the VQ structure have been studied [13]. Examples of VQ techniques used to reduce this complexity are: tree-structured VQ (TSVQ), the classified VQ, the product VQ, the multistage VQ, and the lattice VQ. However, in the above schemes (except for product VQ), since the quantizer still has the km-dimensional VQ structure, the complexity of implementing the encoder is high. Further, note that the performance of these techniques will fall between that the performance of the k-dimensional VQ and the performance of the km-dimensional VQ, due to the modifications made in order to reduce the encoding complexity. In the case of the product VQ, the complexity is significantly reduced at the cost of performance.

Based on the above discussion, it would be interesting to see if one could develop a coding scheme for the source $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ which has the same structure as a k-dimensional VQ but could significantly improve upon the performance of PQ in k dimensions. In this paper, we propose such a quantizer. We call the quantizer a *sample-adaptive product quantizer* (SAPQ).

SAPQ is based on a new concept of adaptation to each sample $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$. SAPQ quantizes each sample $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$ using a sequence of m codebooks and periodically replaces the sequence of m codebooks from a finite set of codebooks. Note that, even for a memoryless stationary source, the empirical distribution function (d.f.) [4] that is constructed using $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$ is substantially different from F, the d.f. of $\boldsymbol{X}_i$. Hence, even for such a source, in SAPQ, different codebooks could be chosen from sample to sample. *It is important to note that SAPQ is very different from the traditional adaptive coding schemes that produce increased gains by replacing the quantizer depending on the varying statistical characteristics of a nonstationary source* [18, pp. 188–210]. SAPQ is a *structurally constrained VQ* in km-dimensional space. SAPQ uses codebooks that are *unions of product codebooks*. Hence, compared to PQ, this allows us more control over the point density of the codebook [15]. Therefore, we expect that the performance of SAPQ will be higher than that of PQ. A pictorial example will be shown in Section VI. We will also show that using the scalar quantizer version of SAPQ $(k = 1)$ we can describe a number of important lattice-based vector quantizers. We will formally study this coding scheme and propose a simple algorithm to implement it. Further, we will show via numerical analyses that our coding scheme significantly outperforms PQ and achieves VQ-level performance, even for $k = 1$.

This paper is organized as follows. In Section II, we describe SAPQ and provide some mathematical definitions. In Section III, we provide some mathematical observations which allow us to better understand the design principles of SAPQ. In Section IV, the relationship between SAPQ and several root lattices is investigated and an asymptotic performance bound is provided. A simple codebook design algorithm for SAPQ is provided in Section V. In Section VI, we conduct a numerical study using several synthetic samples, and compare SAPQ with current coding schemes, such as VQ, TSVQ, and the trellis-coded quantization (TCQ). In Section VII, we conclude the paper and discuss future research. We also provide an appendix (Appendix D) in which we describe in more detail the encoding complexity of SAPQ.

II. Sample-Adaptive Product Quantizer (SAPQ)

In this section, we describe our proposed adaptive quantization scheme. For every sample, SAPQ employs codebook sequences from a previously designed set of 2^η codebook sequences available at both the encoder and the decoder, where η is a nonnegative integer. In SAPQ, it is important to note that the codebook sequence can be changed adaptively for each sample that contains m vectors. Let $C_{i,j} (\subset \mathbb{R}^k)$ denote the ith codebook for each $\boldsymbol{X}_i$, where $j \in \{1, 2, \cdots, 2^\eta\}$. Assume that the samples of $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ sequentially enter the encoder. The adaptive scheme quantizes each sample $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$ using the codebooks $C_{1,j}, \cdots, C_{m,j}$ to form the 2^η candidates of the *m-codebook sample distances* defined by

$$\frac{1}{km} \sum_{i=1}^{m} \min_{\boldsymbol{y} \in C_{i,j}} \|\boldsymbol{X}_i^\omega - \boldsymbol{y}\|^r, \qquad \text{for } j = 1, 2, \cdots, 2^\eta. \quad (4)$$

Here, we suppose that $C_{i,j} \in \mathcal{C}_{n_i'}$, for $j = 1, 2, \cdots, 2^\eta$, where $n_i' \in \mathbb{N}$. The distance in (4) is a random variable defined on the underlying sample space if $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$ is replaced with the random vector $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$. Note that, for a fixed j, in order to quantize the m random vectors, a sequence of m codebooks $C_{1,j}, \cdots, C_{m,j}$ are employed as shown in the m-codebook sample distance. For each sample, the adaptive scheme finds a codebook sequence, from a finite class of codebook sequences, that yields the minimum distance given by (4). The resultant distortion of SAPQ, given by taking expectations in (4), is

$$D_{m\text{-SAPQ}} := E\left\{ \min_j \frac{1}{km} \sum_{i=1}^{m} \min_{\boldsymbol{y} \in C_{i,j}} \|\boldsymbol{X}_i - \boldsymbol{y}\|^r \right\}. \quad (5)$$

We call the SAPQ in (5), which is based on the m-codebook sample distance, *m-SAPQ*. Note that SAPQ encoding is optimal for a given codebook.

In SAPQ, for each sample, the encoder transmits bits, for the codebook index with m quantized element indices, in the form of a feedforward adaptive scheme. This makes it possible to replace different codebook sequences for each sample of $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$. In other words, the encoder quantizes m vectors of a sample $\boldsymbol{X}_1^\omega, \cdots, \boldsymbol{X}_m^\omega$ using a codebook sequence of size m from 2^η codebook sequences and replaces the codebook sequence for each sample. Therefore, the total bit rate in m-SAPQ is given by

$$R = \frac{1}{km} \log_2 \prod_{i=1}^{m} n_i' + \frac{\eta}{km} \quad (6)$$

where η are the additional bits required in our scheme as side bits to indicate which codebook sequence is employed.

Note that m-SAPQ requires at most $m2^\eta$ different codebooks from $\mathcal{C}_{n_i'}$. Hence, if m is large, the decoder needs a large memory for the codebooks and the codebook design complexity may be high. In order to reduce the required number of codebooks, one possibility is to use the 1-*codebook* *sample distance* defined by

$$\frac{1}{km}\sum_{i=1}^{m}\min_{\boldsymbol{y}\in C_j}\|\boldsymbol{X}_i^{\omega}-\boldsymbol{y}\|^r, \qquad \text{for } j=1,2,\cdots,2^{\eta}. \quad (7)$$

Note that for each distance, we use only one codebook C_j. Here we assume that $C_j \in \mathcal{C}_{n'}$ for all j, and $n' \in \mathbb{N}$. The resultant distortion of this simplified SAPQ is given by

$$D_{1\text{-SAPQ}} := E\left\{\min_j \frac{1}{km}\sum_{i=1}^{m}\min_{\boldsymbol{y}\in C_j}\|\boldsymbol{X}_i-\boldsymbol{y}\|^r\right\}. \quad (8)$$

We call this SAPQ, which is based on the 1-codebook sample distance, 1-*SAPQ*. Note that the bit rate for 1-SAPQ is given by $R=(\log_2\ n')/k+\eta/km$.

III. Theoretical Observations on SAPQ

A. Structurally Constrained Quantization in km-Dimensional Space

Let us assume that we are using m-SAPQ, when our underlying quantization space is k-dimensional. Then the principle of m-SAPQ can be explained in terms of a km-dimensional VQ. To do that we first describe the distortion for a general km-dimensional VQ for source $\boldsymbol{X}_1,\cdots,\boldsymbol{X}_m$. Here, we call the quantity km the *quantizer dimension*.

Representing the source as a sequence of random variables $\boldsymbol{X}_1,\cdots,\boldsymbol{X}_m$, let km-tuples $\boldsymbol{X} := (X_{11}, X_{12},\cdots,X_{km})$ denote a random vector in $\mathbb{R}^{km}$. If $\boldsymbol{C}$ is a subset of $\mathbb{R}^{km}$ with $|\boldsymbol{C}|=\nu$, where ν is a positive integer, then the average distortion yielded by using a km-dimensional VQ for $\boldsymbol{X}$ is $E\{(1/km)\min_{\xi\in\boldsymbol{C}}\|\boldsymbol{X}-\xi\|^r\}$, where the bit rate is $R=(\log_2\nu)/km$. In this km-dimensional VQ, the codebook $\boldsymbol{C}$ can be any finite subset of $\mathbb{R}^{km}$. In a similar manner to the km-dimensional VQ case, we now describe PQ and m-SAPQ as quantizers in $\mathbb{R}^{km}$. If $\boldsymbol{C}_{\text{PQ}}$ denotes the km-dimensional codebook of PQ, then $\boldsymbol{C}_{\text{PQ}}$ is the Cartesian products of codebooks defined as $\boldsymbol{C}_{\text{PQ}} := C_1\times\cdots\times C_m$, where $C_i\in\mathcal{C}_{n_i}$ and $n_i\in\mathbb{N}$. The average distortion of PQ in (2) can then be rewritten as

$$D_{\text{PQ}} = E\left\{\frac{1}{km}\min_{\xi\in\boldsymbol{C}_{\text{PQ}}}\|\boldsymbol{X}-\xi\|^r\right\}.$$

Let $\boldsymbol{C}_{m\text{-SAPQ}}\in\mathbb{R}^{km}$ denote the m-SAPQ codebook, then $\boldsymbol{C}_{m\text{-SAPQ}}$ is in the form of unions of the product codebooks, i.e.,

$$\boldsymbol{C}_{m\text{-SAPQ}} := \bigcup_{j=1}^{2^{\eta}}(C_{1,j}\times\cdots\times C_{m,j}) \quad (9)$$

where $C_{i,j}\in\mathcal{C}_{n_i'}$ and $n_i'\in\mathbb{N}$. The average distortion of m-SAPQ in (5) can then be rewritten as

$$D_{m\text{-SAPQ}} = E\left\{\frac{1}{km}\min_{\xi\in\boldsymbol{C}_{m\text{-SAPQ}}}\|\boldsymbol{X}-\xi\|^r\right\}.$$

By observing the codebooks in $\mathbb{R}^{km}$, it is clear that the SAPQ distortion can be made less than or equal to that of PQ. However, the SAPQ distortion is at least as large as that of the minimal km-dimensional VQ distortion. In other words, the performance of m-SAPQ lies between that of the k-dimensional VQ and the km-dimensional VQ.

In m-SAPQ, if $n_i'=1$, for all i, and $\eta=\log_2\nu$, then the m-SAPQ distortion can equal the distortion of the km-dimensional VQ. Note that we can decrease the quantizer distortion in this case by increasing the sample size m such that for large enough m the quantizer distortion would eventually converge to the theoretically obtainable minimum distortion. For the case when $n_i'>1$, for some i, and $\eta>0$, the performance of m-SAPQ can be worse than the km-dimensional VQ but better than any PQ. Note that, in this case as well, we can obtain further gains with m-SAPQ, by increasing the sample size m, since this would result in a larger block size (km) for quantization. Of course, if $n_i'=n_i$, for all i, and the side information $\eta=0$, then m-SAPQ is equivalent to the PQ; and we cannot expect any gain by increasing the sample size m. This tells us that it is important to correctly design m-SAPQ by choosing appropriate parameters.

The quantization scheme 1-SAPQ, described earlier by (7) and (8), is a special case of m-SAPQ. Hence, in PQ, if we use the same codebook $C\in\mathcal{C}_n$, $n\in\mathbb{N}$ for all random vectors $\boldsymbol{X}_1,\cdots,\boldsymbol{X}_m$, then there exists a 1-SAPQ with distortion less than or equal to that of PQ, at the same bit rate as PQ. Note that the km-dimensional codebook of 1-SAPQ has the form

$$\boldsymbol{C}_{1\text{-SAPQ}} := \bigcup_{j=1}^{2^{\eta}}(C_j)^m$$

where $C_j\in\mathcal{C}_{n'}$. It is obvious that m-SAPQ is always better than (or equal to) 1-SAPQ. However, depending on the input source and appropriate values of n' and η for a given bit rate R, the performance of 1-SAPQ can be made comparable to that of m-SAPQ. We observe the performance of 1-SAPQ by regarding 1-SAPQ as an adaptive quantizer in k-dimensional space. This will be shown in the next section.

B. Feedforward Adaptive Quantization in k-Dimensional Space

If the d.f.'s of the random vectors $\boldsymbol{X}_1,\cdots,\boldsymbol{X}_m$ are identical, then it seems that employing the 1-codebook sample distance in (7) may be good enough to quantize the random vectors. In this way, the required number of codebooks is reduced from $m2^{\eta}$ in m-SAPQ to 2^{η} in 1-SAPQ. Hence, it may be advantageous to use 1-SAPQ if the coding scheme yields a sufficient gain over PQ. However, unlike in the m-SAPQ case, simply increasing the sample size m does not always guarantee performance gain in the 1-SAPQ case. We will discuss this after Proposition 2. For the large codebook case, however, 1-SAPQ is asymptotically better than PQ for uniformly distributed data (this will be shown based on the root lattice analysis in Section IV). We will also provide numerical results in Section VI to show that under appropriate design parameters 1-SAPQ can significantly outperform PQ.

In the following discussion we will provide some asymptotic results which will help us better understand how to design an efficient sample adaptive coding scheme using 1-SAPQ. For our analysis, we define a new constant $\beta := m/n'$. We call β the *sample ratio* since it is the ratio of the sample size to the maximum number of available codewords for the quantization of a sample in the 1-SAPQ case. In this section we assume that the random vectors $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ are independent and identically distributed (i.i.d.). Let F be defined as the d.f. of X_1. Using F for a codebook C, the VQ distortion can be rewritten as

$$D_{\mathrm{VQ}} = \frac{1}{k} \int \min_{\boldsymbol{y} \in C} \|\boldsymbol{x} - \boldsymbol{y}\|^r \, dF(\boldsymbol{x}). \tag{10}$$

The quantity $\inf_{C \in \mathcal{C}_n} D_{\mathrm{VQ}}$ is called (n-*level*) F-*optimal distortion* and a codebook C^* that yields the F-optimal distortion is called an (n-level) F-*optimal codebook*, if C^* exists. The corresponding quantizer, when $k = 1$, is called the Lloyd–Max quantizer [27], [22]. The optimal PQ is then given by $\inf D_{\mathrm{PQ}} = \inf_{C \in \mathcal{C}_n} D_{\mathrm{VQ}}$, where the codebooks are $|C_i| = n$, for all i.

Designing an optimal SAPQ for given parameters m, n', and η is important in practical applications, since the parameters are related to the encoding complexity of SAPQ. We now simplify the 1-SAPQ design problem to finding codebooks C_j that satisfy

$$D^o_{1\text{-SAPQ}}(n', 2^\eta) := \inf_{C_j \in \mathcal{C}_{n'}} D^o_{1\text{-SAPQ}} \tag{11}$$

for given m, n', and η. Here, $|C_j| = n'$ for all j. Note that $D^o_{1\text{-SAPQ}}(n', 2^\eta)$ could be even worse than PQ for some n' and η. We consider an optimal m-SAPQ design problem in a similar manner as in (11). For given parameters m, n', and η, define the optimal m-SAPQ distortion as follows:

$$D^o_{m\text{-SAPQ}}(n', 2^\eta) := \inf_{C_{i,j} \in \mathcal{C}_{n'}} D_{m\text{-SAPQ}} \tag{12}$$

where $|C_{i,j}| = n'$ for all i and j. It is obvious that

$$D^o_{m\text{-SAPQ}}(n', 2^\eta) \le D^o_{1\text{-SAPQ}}(n', 2^\eta)$$

and

$$D^o_{m\text{-SAPQ}}(n', 2^\eta) \le \inf D_{\mathrm{PQ}}$$

if $(n')^m 2^\eta = n^m$.

In order to provide a lower bound on $D^o_{1\text{-SAPQ}}(n', 2^\eta)$ of 1-SAPQ, we first describe the *empirically optimal distortion* introduced in [30]. We define the empirical distortion as follows. Rewrite the 1-codebook sample distance in (7) as a form of empirical distortion $\delta(C, F^\omega_m)$ as

$$\begin{aligned} \delta(C, F^\omega_m) &:= \frac{1}{k} \int \min_{\boldsymbol{y} \in C} \|\boldsymbol{x} - \boldsymbol{y}\|^r \, dF^\omega_m(\boldsymbol{x}) \\ &= \frac{1}{km} \sum_{i=1}^{m} \min_{\boldsymbol{y} \in C} \|\boldsymbol{X}^\omega_i - \boldsymbol{y}\|^r \end{aligned} \tag{13}$$

for a codebook $C \in \mathcal{C}_{n'}$. In (13), F^ω_m *is the empirical d.f. constructed by placing equal masses at each of the* m *vectors of the sample* $\boldsymbol{X}^\omega_1, \cdots, \boldsymbol{X}^\omega_m$ [4, p. 268]. The empirical d.f. that is constructed for each sample is quite different from F and also quite different from sample to sample with high probability (especially if m is chosen to be small). This allows us to tailor an appropriate codebook for a given sample by choosing a codebook from an optimally designed finite class of codebooks, where each such codebook minimizes the 1-codebook sample distance in (7) within the class. For different codebooks C_j, for $j = 1, 2, \cdots, 2^\eta$, let a codebook class $\mathcal{C}^\eta_{n'}$ denote the class of C_j, i.e., $\mathcal{C}^\eta_{n'} := \{C_j\}$ with $|\mathcal{C}^\eta_{n'}| = 2^\eta$. Using the notation $\delta(C, F^\omega_m)$ defined in (13), we can rewrite the distortion of 1-SAPQ in (8) as

$$D^o_{1\text{-SAPQ}} = E\left\{ \inf_{C \in \mathcal{C}^\eta_{n'}} \delta(C, F_m) \right\}. \tag{14}$$

In (14), let us replace the codebook class $\mathcal{C}^\eta_{n'}$ with $\mathcal{C}_{n'}$. Then, the 1-SAPQ distortion is changed to the empirically optimal distortion defined by

$$E\left\{ \inf_{C \in \mathcal{C}_{n'}} \delta(C, F_m) \right\} \tag{15}$$

which is always less than (14). We call this distortion the F_m-*optimal distortion*. If $\beta = 1$, i.e., the codebook size n' is equal to the sample size m, then for all sample points $\omega \in \Omega$, $\delta(C^\omega, F^\omega_m) = 0$ holds if codebook $C^\omega = \{\boldsymbol{X}^\omega_1, \cdots, \boldsymbol{X}^\omega_{n'}\}$ is chosen for each F^ω_m. Hence, the F_m-optimal distortion is obviously equal to zero. In the special case when $n' = 1$ and $\beta \ge 1$, we obtain the well-known relation

$$E\left\{ \inf_{C \in \mathcal{C}_{n'}} \delta(C, F_m) \right\} = (m-1)/m \cdot \mathrm{Var}\,(\boldsymbol{X}_1)$$

which implies that the mean distortion is a biased estimator of the variance. However, for the $\beta > 1$ case, an explicit derivation of (15) is usually difficult in general. It will be shown that this F_m-optimal distortion is the infimum of the sample-adaptive distortion of 1-SAPQ, for all η.

Proposition 1: For an increasing sequence $\mathcal{C}^1_{n'} \subset \mathcal{C}^2_{n'} \subset \cdots$, the sequence $(E\{\inf_{C \in \mathcal{C}^\eta_{n'}} \delta(C, F_m)\})_\eta$ is monotonically decreasing, i.e.,

$$E\{\inf_{C \in \mathcal{C}^\eta_{n'}} \delta(C, F_m)\} \downarrow E\{\inf_{C \in \mathcal{C}_{n'}} \delta(C, F_m)\}, \qquad \text{as } \eta \to \infty \tag{16}$$

for any positive integers m and n'.

Proof of Proposition 1: See Appendix A for the proof. □

From Proposition 1, for given k, m, n', and η, we have a lower bound to the 1-SAPQ distortion as

$$\max\left\{ D^o_{m\text{-SAPQ}}(n', 2^\eta), E\left\{ \inf_{C \in \mathcal{C}_{n'}} \delta(C, F_m) \right\} \right\} \le D^o_{1\text{-SAPQ}}(n', 2^\eta). \tag{17}$$

From (17), it is clear that the distortion of 1-SAPQ is bounded by the F_m-optimal distortion. We now observe the quantity $D^o_{1\text{-SAPQ}}(n', 2^\eta)$ by investigating its relationship with the F_m-optimal distortion. In [28] and [1], the consistency problem of the F_m-optimal distortion is investigated based on the convergence of the probability measure in metric space, and the Glivenko–Cantelli theorem. From these investigations we obtain an asymptotic result (as the sample size m gets large).

Proposition 2: Suppose that there exists an n'-level F-optimal codebook. Then

$$D^o_{1\text{-SAPQ}}(n', 2^\eta) \to \inf_{C \in \mathcal{C}_{n'}} E\left\{\frac{1}{k} \min_{\boldsymbol{y} \in C} \|\boldsymbol{X}_1 - \boldsymbol{y}\|^r\right\}, \quad \text{as } m \to \infty \tag{18}$$

for fixed integer n'.

Proof of Proposition 2: See Appendix B for the proof. □

The above proposition tells us that as the sample size m increases to infinity compared to the codebook size n', $D^o_{1\text{-SAPQ}}(n', 2^\eta)$ simply converges to the n'-level F-optimal distortion, which is greater than the n-level F-optimal distortion, inf D_{PQ}. Note that, since $n > n'$

$$\inf D_{\text{PQ}} \le \inf_{C \in \mathcal{C}_{n'}} E\left\{(1/k) \min_{\boldsymbol{y} \in C} \|\boldsymbol{X}_i - \boldsymbol{y}\|^r\right\}.$$

Further, for a fixed n' and an increasing η at a fixed ratio η/m, the bit rate is also fixed, and $D^o_{1\text{-SAPQ}}(n', 2^\eta)$ still converges to the n'-level F-optimal distortion from Proposition 2, even if there is constant side information η/m. In other words, if we increase m for fixed n' (increase the sample ratio β), then the gain decreases and the 1-SAPQ performance eventually becomes worse than PQ, independently of η. This is an expected result since it implies that the adaptation occurs over larger and larger time intervals compared to the codebook size. However, as will be demonstrated in Section VI, increasing m while keeping the sample ratio $\beta = m/n'$ small (i.e., increasing the codebook size n' as well), will allow for very efficient adaptation, and performance of 1-SAPQ that is virtually identical to the optimal m-SAPQ case, $D^o_{m\text{-SAPQ}}(n', 2^\eta)$.

IV. SAPQ and Root Lattices

In this section, we provide further intuition as to why we expect SAPQ to outperform PQ. We do this by observing the Voronoi partitions of SAPQ and PQ as km-dimensional VQ. It is obvious that all the Voronoi regions that are generated by the product codebook of PQ are rectangular in $\mathbb{R}^{km}$. However, in SAPQ it is possible to make a Voronoi partition such that each Voronoi region yields lower average distortion than the rectangular region for a given volume of the Voronoi region. Thus we expect that for a given bit-rate, SAPQ will achieve a smaller average distortion than by using PQ. Also note that the product codebooks of SAPQ can be efficiently assigned to the joint d.f. of $\boldsymbol{X}$, even if the random vectors are dependent.

The lattice VQ [14] is a uniform quantizer whose output is a truncated root lattice [17]. An m-dimensional lattice is defined as a set of points in $\mathbb{R}^{m'}$

$$\Lambda_m := \{\boldsymbol{x} | \boldsymbol{x} = U\boldsymbol{p}, \boldsymbol{p} \in \mathbb{Z}^m\} \tag{19}$$

where the $m' \times m$ matrix $U = (\boldsymbol{u}_1 \cdots \boldsymbol{u}_m)$ is a generator matrix for Λ, $\boldsymbol{u}_i (\in \mathbb{R}^{m'})$ are linearly independent vectors, and $m \le m'$. Here $\boldsymbol{p}$ is written as a column vector. Let the Voronoi regions that are constructed by the lattice Λ_m have the shape of some polytope $\mathcal{P}$ with centroid $\boldsymbol{x}_o$. Then $G(\Lambda_m)$, the *normalized second moment of* $\mathcal{P}$, is defined as

$$G(\Lambda_m) := \frac{1}{m} \frac{\int_{\mathcal{P}} \|\boldsymbol{x} - \boldsymbol{x}_o\|^2 \, d\boldsymbol{x}}{\left(\int_{\mathcal{P}} d\boldsymbol{x}\right)^{(m+2)/m}}. \tag{20}$$

The quantity $G(\Lambda_m)$ determines the performance of a lattice VQ using the mean-square distortion measure, i.e., $r = 2$ in $\|\cdot\|^r$, as the metric of performance [12], [6]. Conway and Sloane have calculated the second moments of various lattices that yield values close to $\inf_{\Lambda_m} G(\Lambda_m)$ for various dimensions, where the infimum is taken over all m-dimensional lattices [6], [7, Table I]. For example, the hexagonal lattice, which is equivalent to the lattice A_2 in [6], is the optimal lattice in two dimensions. In the three-dimensional case, the $D_3^\perp$ lattice (or, equivalently, the lattice $A_3^\perp$) is a body-centered cubic lattice and optimal in three dimensions [3]. Furthermore, Conway and Sloane have found that

$$\inf_{\Lambda_2} G(\Lambda_2) = G(A_2) \cong 0.0802$$

and

$$\inf_{\Lambda_3} G(\Lambda_3) = G(D_3^\perp) \cong 0.0785$$

and they have also conjectured a lower bound for $\inf_{\Lambda_m} G(\Lambda_m)$ [8]. For the definitions of these lattices, see [6].

A. Lattice VQ and SAPQ

It turns out that vector quantization based on many different root lattices can easily be constructed by scalar quantization in SAPQ, i.e., SAPQ with $k = 1$. For sets $\Lambda^{i,j} \subset \mathbb{R}$, $i = 1, \cdots, m$, $j = 1, 2, \cdots, 2^\eta$, assume that $\text{card}(\Lambda^{i,j})$ is the same as the cardinality of the integer set and $\boldsymbol{L}_m^\eta$ is the class of all sets that have the unions of the Cartesian product form $\cup_{j=1}^{2^\eta} (\Lambda^{1,j} \times \cdots \times \Lambda^{m,j})$. Note that $(\Lambda^{1,j} \times \cdots \times \Lambda^{m,j})$ is a coset of a rectangular lattice and that the *product codebook of m-SAPQ is a subset of* $(\Lambda^{1,j} \times \cdots \times \Lambda^{m,j})$. Hence, a truncated set of elements in $\boldsymbol{L}_m^\eta$ can be implemented using m-SAPQ with $k = 1$ and η. (For each j, if $\Lambda^{i,j}$ are the same for all i, then we can use 1-SAPQ.)

The first type of lattice that we investigate is the A_m lattice. Let $A_m^\perp$ denote the dual lattice [6] of A_m. Then A_1 and $A_1^\perp$ are equivalent to $\mathbb{Z}$, i.e., $A_1 \equiv A_1^\perp \equiv \mathbb{Z}$. $A_2 (\equiv A_2^\perp)$ is the hexagonal lattice and can be generated by the basis vectors $\boldsymbol{u}_1 = (1, -1, 0)$ and $\boldsymbol{u}_2 = (1, 0, -1)$. Since $\boldsymbol{u}_1$ and $(2\boldsymbol{u}_2 - \boldsymbol{u}_1)$ are orthogonal, $\boldsymbol{u}_1$ and $2\boldsymbol{u}_2$ generate a rotated rectangular lattice Λ. Thus

$$A_2 = \Lambda \cup (\boldsymbol{u}_2 + \Lambda) \tag{21}$$

and it follows that there exists a lattice A satisfying $A_2 \equiv A \in \boldsymbol{L}_2^1$. In the three-dimensional case, A_3 is the faced-centered cubic lattice and has three basis vectors: $(1, -1, 0, 0)$, $(1, 0, -1, 0)$, and $(1, 0, 0, -1)$ [29]. Applying the same idea as in the A_2 case, there is a lattice A such that $A_3 \equiv A \in \boldsymbol{L}_3^1$. To summarize, the lattice VQ based on A_2 and A_3 can be described by m-SAPQ ($m = 2$ and 3, respectively), with $\eta = 1$. By increasing η we can also describe A_m for larger values of m.

Another important type of lattice is the D_m $(m \geq 2)$ lattice. We can represent D_m as the union of 2^{m-1} cosets of rectangular lattices, and since one rectangular lattice corresponds to a product codebook in SAPQ, we obtain $2^{m-1} = 2^\eta$. Hence, the side information required in this case is $\eta = m - 1$ which corresponds to the number of 1's in the diagonal of the D_m generator matrix. This also means that $D_m \in \boldsymbol{L}_m^{m-1}$ can be implemented by using m-SAPQ. For $m \geq 2$, $D_m^\perp$ is the dual of the lattice D_m, and it is clear that $D_m^\perp \in \boldsymbol{L}_m^1$. This implies that 1-SAPQ can construct the $D_m^\perp$ lattice with only $\eta = 1$. Further, it also tells us that 1-SAPQ with $\eta = 1$ can construct the optimal lattice in three dimensions, since $D_3^\perp \equiv A_3^\perp$. Furthermore, in a similar way to the D_m case, we obtain $E_7 \in \boldsymbol{L}_7^3$ and $E_8 \in \boldsymbol{L}_8^4$. Hence, these lattices can also be implemented by m-SAPQ.

B. Asymptotic Lattice Bound

Using the normalized second moment $G(\Lambda_{km})$ of a lattice Λ_{km} in $\mathbb{R}^{km}$, we can obtain an asymptotic upper bound on the m-SAPQ distortion for any vector dimension k. Define the optimal km-dimensional VQ distortion $D^o(\nu)$ as

$$D^o(\nu) := \inf_{\boldsymbol{C},\, |\boldsymbol{C}|=\nu} E\left\{ \frac{1}{km} \min_{\xi \in \boldsymbol{C}} \|\boldsymbol{X} - \xi\|^r \right\}. \quad (22)$$

We now show that the optimal m-SAPQ distortion can achieve the optimal km-dimensional VQ in an asymptotic sense.

Theorem 1: Suppose that the source

$$\boldsymbol{X} = (X_{11}, X_{12}, \cdots, X_{km})$$

has a joint density function f with $E\{\|\boldsymbol{X}\|^{2+\epsilon}\} < \infty$ for some $\epsilon > 0$, and f is bounded on $\mathbb{R}^{km}$. Then, for an increasing sequence $(n_\eta)_\eta$ such that $n_\eta / 2^{(\eta/km)} \to \alpha$, where α is a positive constant,

$$\limsup_{\eta\to\infty} [(n_\eta)^{km} 2^\eta]^{2/km} D^o_{m\text{-SAPQ}}(n_\eta, 2^\eta) \leq G(\Lambda_{km}) \|f\|_{km/(km+2)} \quad (23)$$

where the functional $\|\cdot\|_{km/(km+2)}$ is given by

$$\|f\|_{km/(km+2)} := \left[\int f^{km/(km+2)}(\boldsymbol{x})\, d\boldsymbol{x} \right]^{(km+2)/km}. \quad (24)$$

Proof of Theorem 1: The proof of Theorem 1 is given in Appendix C. □

Note that, in the m-SAPQ of Theorem 1, the size of the codebook $\boldsymbol{C}_{m\text{-SAPQ}}$ is $(n_\eta)^{km} 2^\eta$. From Theorem 1, we can obtain the asymptotic result

$$\limsup_{\eta\to\infty} [(n_\eta)^{km} 2^\eta]^{2/km} D^o_{m\text{-SAPQ}}(n_\eta, 2^\eta) \leq J_{km} \|f\|_{km/(km+2)} \quad (25)$$

where $J_{km} := \inf_{\Lambda_{km}} G(\Lambda_{km})$. It is clear that the optimal km-dimensional VQ is such that

$$\limsup_{\nu\to\infty} \nu^{2/km} D^o(\nu) \leq J_{km} \|f\|_{km/(km+2)}.$$

From [31] and [5], we know that the sequence on the left-hand side converges under certain conditions. Further, from a well-known conjecture (that is generally believed to be correct), it has been hypothesized that the asymptotically optimal quantizer is a function of J_{km} [12]. More specifically

$$\lim_{\nu\to\infty} \nu^{2/km} D^o(\nu) = J_{km} \|f\|_{km/(km+2)}.$$

Therefore, if this conjecture were true (as is typically assumed), then m-SAPQ could achieve the asymptotically optimal km-dimensional VQ performance [24], [10]. Based on the asymptotic result given by Theorem 1, we can discuss the achievable performance of SAPQ. As shown in (25), since SAPQ can achieve the performance $J_{km}\|f\|_{km/(km+2)}$, the asymptotic performance of the optimal km-dimensional VQ, the advantages of SAPQ over PQ are the same as the advantage of VQ over scalar quantization [24], [25].

V. Codebook Design for Nonuniform SAPQ

For given k, m, and n', as shown in (11), the design problem of 1-SAPQ is to find an optimal codebook that achieves the distortion $D^o_{1\text{-SAPQ}}(n', 2^\eta)$. However, finding such an optimal codebook is not easy for the nonuniformly distributed inputs. In order to find (sub)optimal codebooks, we have developed a clustering algorithm that uses a large number of samples as a *training sequence* (TS). But this TS size is still substantially less than that of traditional VQ or modified schemes. (In a straightforward manner, this algorithm can be extended to designing the m-SAPQ codebook.) Let $\boldsymbol{x}_{1,\ell}, \cdots, \boldsymbol{x}_{m,\ell}$ denote the ℓth training sample in a given TS that has M samples, where a sample has m *training vectors*. The first part of our algorithm quantizes m training vectors in each sample using 2^η different codebooks and then selects a codebook that yields the minimal distance (given in (7)) for the sample. The second part of the algorithm updates the codebooks using the partitioned TS in the quantization process of the first part. These two parts are then iteratively applied to the given TS. The clustering algorithm is described below.

Clustering Algorithm (1-SAPQ)

0. Initialization $(\gamma = 0)$: Given sample size m, codebook size n', side bits η, distortion threshold $\epsilon \geq 0$, initial codebook $\boldsymbol{C}_0$, and TS $((\boldsymbol{x}_{1,\ell}, \cdots, \boldsymbol{x}_{m,\ell}))_{\ell=1}^M$, set $D_{-1} = \infty$.
1. Given codebook $\boldsymbol{C}_\gamma = \cup_{j=1}^{2^\eta} (C_j)^m$, where $C_j \in \mathcal{C}_{n'}$, find $2^\eta n'$ partitions of each training vector in the TS for the corresponding $2^\eta n'$ codewords, where each training vector's codeword is determined by the following quantization:

$$d_\ell := \min_{j \in \{1,2,\cdots,2^\eta\}} \frac{1}{m} \sum_{i=1}^{m} \min_{\boldsymbol{y} \in C_j} \|\boldsymbol{x}_{j,\ell} - \boldsymbol{y}\|^r, \quad \text{for } \ell = 1, \cdots, M. \quad (26)$$

Next, we compute the average distortion D_γ for the γth iteration, given by

$$D_\gamma := \frac{1}{M} \sum_{\ell=1}^{M} d_\ell. \quad (27)$$

2. If $(D_{\gamma-1}-D_\gamma)/D_\gamma \le \epsilon$, stop. $\boldsymbol{C}_\gamma$ is the final codebook. Otherwise continue.
3. Compute centroids for each of the $2^\eta n'$ partitions and replace the codewords in $\boldsymbol{C}_\gamma$ by the new $2^\eta n'$ centroids. Increase γ by 1. Go to Step 1.

It can be shown using similar techniques as in the case of the generalized Lloyd algorithm (GLA) or the K-means algorithm [2] that D_γ is a decreasing sequence. Thus D_γ converges to a (local) minimum, which depends on the initial codebook $\boldsymbol{C}_0$. In the traditional scalar quantizer design problem, we can obtain the global optimum if the input source has a log-concave density as in the case of the Gaussian source [19]. However, even for Gaussian sources, in the case of SAPQ, convergence to the global optimum is not guaranteed. Therefore, it is especially important to choose an appropriate initial codebook. We next outline a "split method" to determine the codebook $\boldsymbol{C}_0$. For the generation of an initial codebook $\boldsymbol{C}_0$ from the split method, we need a start codebook that is denoted by $\boldsymbol{C}_0^0$ in $\mathbb{R}^{km}$. The start codebook $\boldsymbol{C}_0^0 = (C_1^0)^m$ contains codebook C_1^0 that belongs to $\mathcal{C}_{n'}$, where C_1^0 is a k-dimensional VQ codebook designed by the current VQ design algorithm.

Initial Codebook Guess (Split Method)

0. Initialization ($\gamma = 0$): Given sample size m, codebook size n', side bits η, split constant $\overline{\varepsilon} = (\varepsilon, \cdots, \varepsilon) \in \mathbb{R}^k$, where $\varepsilon \ge 0$, start codebook $\boldsymbol{C}_0^0 \subset \mathbb{R}^{km}$, and TS $((\boldsymbol{x}_{1,\ell}, \cdots, \boldsymbol{x}_{m,\ell}))_{\ell=1}^M$.
1. If $\gamma \ge \eta$, stop. $\boldsymbol{C}_0^\gamma$ is the initial codebook $\boldsymbol{C}_0$ for the clustering algorithm. Otherwise continue.
2. Increase γ by 1. Construct a new codebook $\boldsymbol{C}_0^\gamma = \cup_{j=1}^{2^\gamma}(C_j^\gamma)^m$ by doubling the number of codebooks from the cosets of codebooks in $\boldsymbol{C}_0^{\gamma-1} = \cup_{j=1}^{2^{\gamma-1}}(C_j^{\gamma-1})^m$ as follows:

$$C_j^\gamma = -\overline{\varepsilon} + C_j^{\gamma-1} \quad \text{and} \quad C_{2^{\gamma-1}+j}^\gamma = \overline{\varepsilon} + C_j^{\gamma-1} \tag{28}$$

for $j = 1, 2, \cdots, 2^{\gamma-1}$.

3. Given $\boldsymbol{C}_0^\gamma$, find $2^\gamma n'$ partitions of training vectors according to the quantization

$$\min_{j\in\{1,2,\cdots,2^\gamma\}} \frac{1}{m}\sum_{i=1}^m \min_{\boldsymbol{y}\in C_j^\gamma} \|\boldsymbol{x}_{i,\ell} - \boldsymbol{y}\|^r, \quad \text{for } \ell = 1, \cdots, M. \tag{29}$$

Compute the centroids for each of the $2^\gamma n'$ partitions and replace the codewords in $\boldsymbol{C}_0^\gamma$ by the new $2^\gamma n'$ centroids. Go to Step 1.

In this *Split Method*, Step 2 doubles the number of codebooks in the km-dimensional codebook $\boldsymbol{C}_{1\text{-SAPQ}}$ by adding to and subtracting from each element of the previous codewords a small constant $\overline{\varepsilon}$. This doubling scheme is based on a principle that the all-product codebooks in 1-SAPQ must be symmetric with respect to the line $\boldsymbol{x}_1 = \boldsymbol{x}_2 = \cdots = \boldsymbol{x}_m$ in km-dimensional space. However, even though the split method is simple, and provides good results (as shown in Section VI) in order to find better optima especially for m-SAPQ, it is necessary to continue research on the initial codebook guess.

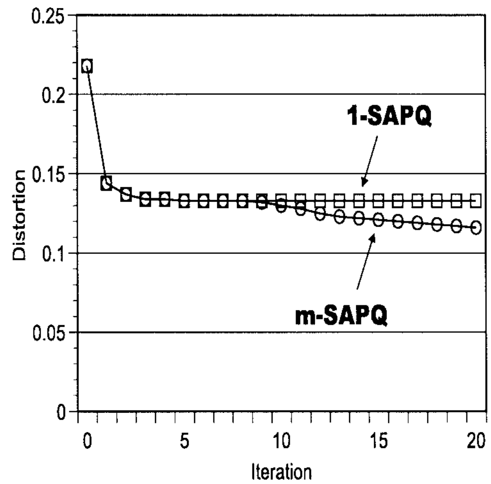
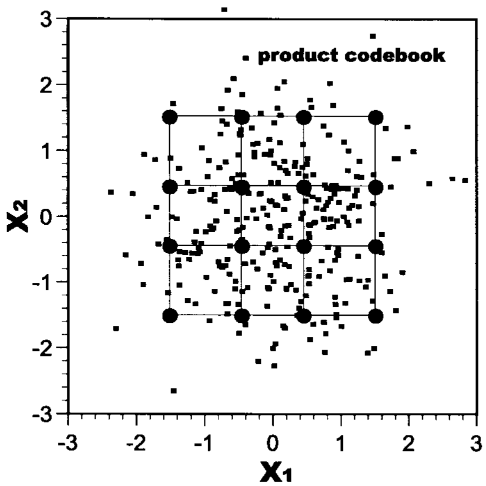


Fig. 1. Sample-adaptive distortion D_γ with respect to iteration i in the algorithm at $R = 2$. (Gaussian i.i.d. with unit variance, $k = 1$, $m = 2$, $n = 2$, and $\eta = 2$.)

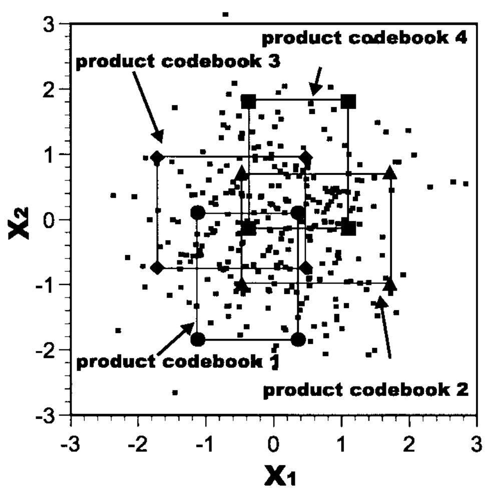
VI. Simulation Results and Discussions

For our experimental results, we use synthetic data that represents a Gaussian (or Laplacian) i.i.d. signal for all elements of the source $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$. In other words, all the random variables in $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ are i.i.d. In our numerical study, to ensure a good codebook design for the underlying distribution, we have used more than 5000 training vectors per codeword. The mean-square distortion measure is employed (i.e., $r = 2$ in (1)). In Fig. 1 we illustrate how the algorithm converges to a (local) minimum distortion in designing the m-SAPQ of (12) and the 1-SAPQ of (11) (here $m = 2$, $n' = 2$, and $\eta = 2$), where the split constant is $\varepsilon = 0.001$. Since the initial codebook is made of the Lloyd–Max quantizer at $n' = 2$, the starting distortion D_0 in (27) is less than 0.363, that of the Lloyd–Max quantizer. The distortion sequence of D_γ monotonically decreases with each iteration in both the m-SAPQ and 1-SAPQ cases. In the m-SAPQ case in Fig. 1, the distortions for the first several iterations follow that of 1-SAPQ, since the *Initial Codebook Guess*, which is for 1-SAPQ, is also employed for m-SAPQ.

For understanding the principle of SAPQ, the product codebooks of PQ and m-SAPQ are shown in Fig. 2(a) and (b), respectively, for the $k = 1$ case. In the PQ case, since the 16 $(= n^m)$ codewords in two-dimensional space $(m = 2)$ are the elements of the product codebook $C \times C$, the Voronoi regions are rectangular from the codebooks as shown in Fig. 2(a). The product codebooks in the m-SAPQ case include the PQ case and further, can make nonrectangular Voronoi regions, which yield lower distortion than the PQ case, by arranging the codewords as shown in Fig. 2(b). In contrast, in the 1-SAPQ case, all the product codebooks must be symmetric with respect to the line $\boldsymbol{X}_2 = \boldsymbol{X}_1$ as shown in Fig. 3. This means that the product codebooks of 1-SAPQ cannot include all the product codebooks of PQ for given n' and η, and that the sample-adaptive distortion of 1-SAPQ is not always guaranteed to be less than the PQ case. In the examples of Figs. 2 and 3, PQ yields − 9.30 dB of distortion at the bit rate of 2. m-SAPQ decreases the distortion to −



(a)



(b)

Fig. 2. The product codebooks of PQ and m-SAPQ in the quantizer dimension 2. (Gaussian i.i.d. with unit variance, $k = 1$, $m = 2$, and $R = 2$.) (a) Product codebook in PQ. (Distortion $= -$ 9.30 dB at $n = 4$.) (b) Product codebooks in m-SAPQ. (Distortion $= -$ 9.56 dB at $n' = 2$ and $\eta = 2$.)

9.56 dB but, 1-SAPQ increases the distortion to $-$ 8.75 dB. However, we will show that 1-SAPQ can significantly do better than PQ when appropriate parameters n' and β are chosen.

In Fig. 4(a) and (b), for bit rates of 2 and 3, respectively, we compare the distortions of m-SAPQ, 1-SAPQ, and PQ for increasing values of the sample size m at fixed values of n' (or, equivalently, for increasing values of the sample ratio β). As expected, m-SAPQ always yields better results than PQ and 1-SAPQ. In the m-SAPQ case, increasing m for a fixed value of n yields more gain over the PQ case. As mentioned before, the 1-SAPQ distortion could be worse than PQ, which is again illustrated in Fig. 4(a). However, by appropriately choosing m and n' we can obtain better results than in the PQ case as shown in Fig. 4(b). It should be pointed out here that even for the case of Fig. 4(b), the gains in distortion flatten out with increasing m, and, in fact, for a large enough value of m will cause the distortion to eventually increase and converge to that of the four-level Lloyd–Max quantizer (i.e., to $-$ 9.30 dB for

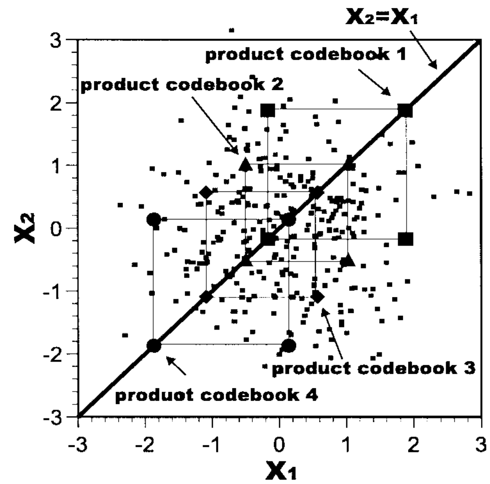


Fig. 3. The product codebooks of 1-SAPQ in the quantizer dimension 2. (Distortion $= -$ 8.75 dB for Gaussian i.i.d. with unit variance at $k = 1$, $m = 2$, $n' = 2$, $R = 2$, and $\eta = 2$.)

a bit rate of 3). Therefore, to obtain gains in 1-SAPQ for a given bit rate, it is important to use as large a value for m (and n') as possible, while keeping the sample ratio β small (note that since increasing n increases the total bit rate, this implies that for a given bit rate the side information η/m should be accordingly decreased).

We know that m-SAPQ is always better than 1-SAPQ. However, observe Fig. 5(a) and (b). The signal-to-noise ratio (SNR) of 1-SAPQ is nearly the same as that of m-SAPQ, especially for $n' \geq 4$. In fact, through extensive simulation studies, we have found that for fixed values of β and the bit rate, increasing n' results in each of the m codebooks of a codebook sequence in m-SAPQ to approach a single codebook (i.e., to become equal to one another). Therefore, for a relatively large n' (compared to η) and a fixed value of β, it is advantageous to use 1-SAPQ, since its performance will closely approximate that of m-SAPQ. Using these design guidelines allows us to reduce the memory requirement by a factor of m (from m-SAPQ to 1-SAPQ) without major compromise on performance. Furthermore, for large codebook-size cases, the 1-SAPQ design problem is nearly independent of the sample size m. In other words, if we design a 1-SAPQ codebook for $m = 16$, then this codebook also works well for $m = 4$ cases; an example of this is shown in Table I. The designed 1-SAPQ codebooks are cross-verified using TS and the validating sequence (VS) [21]. In the next few simulation studies, we will show results under the above design guidelines, and hence show comparisons only between PQ and 1-SAPQ.

A. Distortions for Different Parameters

In Fig. 6 we depict the 1-SAPQ results obtained by simulations for various parameters β and η, when $k = 1$ and the codebook size $n' = 16$. In Fig. 6(a) and (b) we plot the distortion (dB) versus the bit rate for Gaussian and Laplacian i.i.d. sources, respectively. In each case, we plot curves for $\eta = 1, \cdots, 4$ and compare the performance of 1-SAPQ to PQ (the Lloyd–Max quantizer). In all the cases

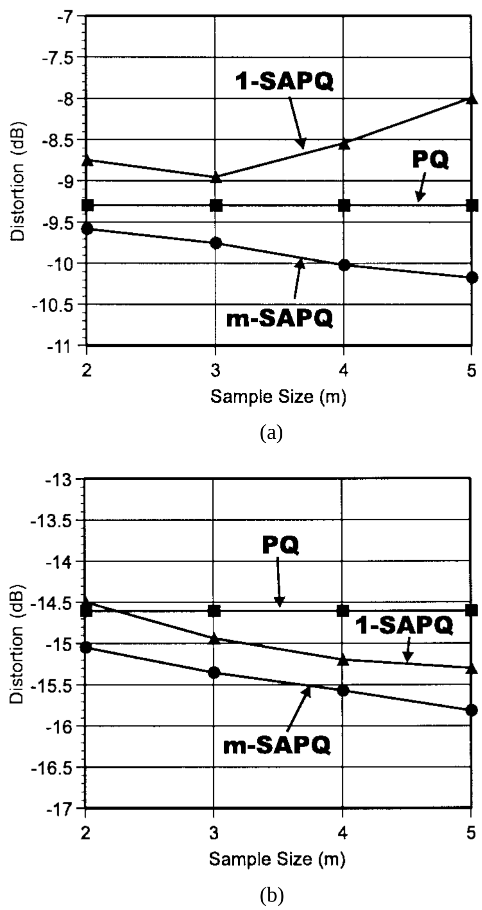


Fig. 4. Distortions (dB) versus sample size m for PQ, m-SAPQ, and 1-SAPQ. (Gaussian i.i.d. with unit variance, $k=1$, and $\eta = m$.) (a) $R=2$, $n'=2$ for SAPQ, and $n=4$ for PQ. (b) $R=3$, $n'=4$ for SAPQ, and $n=8$ for PQ.

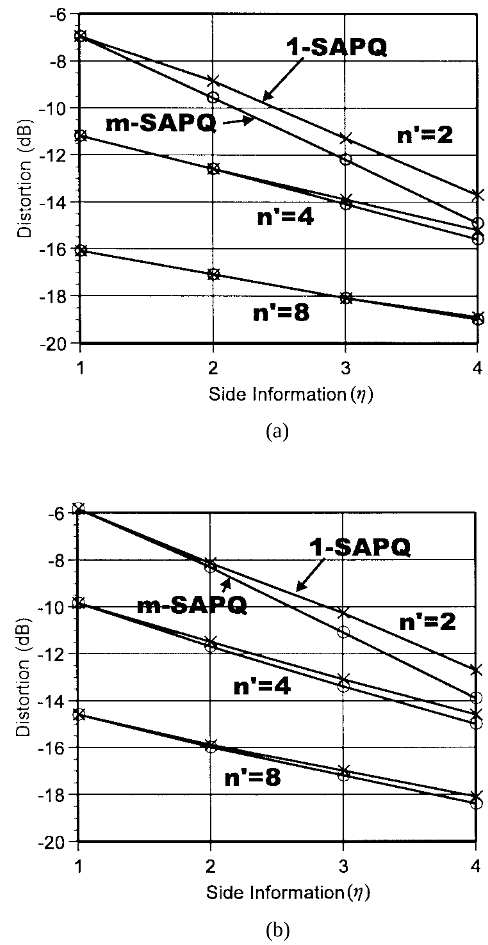


Fig. 5. Distortions (dB) of m-SAPQ and 1-SAPQ for different values of η $(\beta = 1)$. (a) Gaussian i.i.d. with unit variance. (b) Laplacian i.i.d. with unit variance.

of Fig. 6(a) and (b), 1-SAPQ can be seen to significantly outperform PQ. For example, for $\beta = 1$ and $\eta = 4$ in Fig. 6(a) (i.e., at a bit rate of 4.25), the distortion curve of 1-SAPQ shows a 1.8-dB improvement over the distortion curve of PQ. As shown in Fig. 6, depending on the parameters β and η, we can obtain different bit rates and performance from 1-SAPQ. Hence, how to choose the parameters is dependent on a tradeoff between performance and complexity. For example, in Fig. 6(a), focus on the bit rate of 4.125. We have three different 1-SAPQ with different combinations of parameters. As β increases, we can obtain more gain, but the encoding complexity increases, since the required number of multiplications is $2^{\eta+1}\log_2 n'$, where $\eta = 2\beta$ (see Appendix D).

In Fig. 7, the performance benefits of using 1-SAPQ are also demonstrated in the VQ case $(k=2)$. Using the k-dimensional VQ $(k>1)$ in 1-SAPQ, we can obtain a gain especially for low bit rates, and exploit the VQ gain if there exists high correlation inside the k-dimensional random vector [10]. As we can see in both Figs. 6 and 7, increasing η provides a greater improvement in the quantizer performance than is obtained by increasing the codebook size n'. In other words, the 1-SAPQ distortion curve for each sample ratio β always shows a higher slope than the PQ distortion cases. For the $\beta = 1$ case at $k=2$, the quantizer dimension is now increased

TABLE I
1-SAPQ Design and Sample Size m

Bit-Rate R	Data	1-SAPQ Designed at $m=4$	$m=16$	
4.25	TS	-22.60	-22.58	≈ -22.6
$(m=4)$	VS	-22.62	-22.62	
4.0625	TS	-21.41	-21.43	≈ -21.4
$(m=16)$	VS	-21.42	-21.44	

to $km=64$ (from the case of $km=16$ in the earlier example with $\beta=1$ in Fig. 6). Hence there is more room for improving the gain, by increasing η. However, if β and η were kept the same, then the performance gain from 1-SAPQ over PQ for $k=2$ would be less than the performance gain from 1-SAPQ over PQ for $k=1$. This is because the $k=2$ case requires more side bits η to obtain comparable gain. Hence, if we were to increase the side bits η, the gain in the higher dimensional 1-SAPQ over an equal dimension PQ would be comparable to the scalar 1-SAPQ over the scalar PQ. For example, in the lattice discussion in Section IV, in order to implement good lattices at higher dimensions, we need more side bits η. On the other hand, it should be noted that if the quantizer dimension

TABLE II
DISTORTION (dB) AND COMPLEXITY COMPARISON OF SAPQ

Quan. Dim. (km)	Quantizer		$R = 1.5$	$R = 2.5$	$R = 3.5$	$R = 4.5$	Formula
2	VQ	Codebook Size (ν)	8	32	128	512	
		Distortion (dB)	-6.94	-12.4	-18.1	-23.8	
		Multiplications[1]	8	32	128	512	ν
		Memory[2]	16	64	256	1024	$m\nu$
2	TSVQ (Breadth = 2)	Depth (d)	3	5	7	9	
		Distortion (dB)	-6.23	-11.7	-17.5	-23.3	
		Multiplications	6	10	14	18	$2d$
		Memory	28	124	508	2044	$2(2^d - 1)m$
2	1-SAPQ $(\eta = 1)$	Codebook Size (n')	2	4	8	16	
		Distortion (dB)	-6.91 (-6.92)[3]	-12.3	-17.9	-23.7	
		Multiplications	4 (4)	8	12	16	$2^{\eta+1} \log_2 n'$
		Memory	4 (8)	8	16	32	$2^\eta n'$
4	1-SAPQ $(\eta = 2)$	Codebook Size (n')	2	4	8	16	
		Distortion (dB)	-6.83 (-7.08)	-12.6	-18.3	-24.2	
		Multiplications	8 (8)	16	24	32	$2^{\eta+1} \log_2 n'$
		Memory	8 (32)	16	32	64	$2^\eta n'$
8	1-SAPQ $(\eta = 4)$	Codebook Size (n')	2	4	8	16	
		Distortion (dB)	-6.31 (-7.51)	-12.8	-18.8	-24.7	
		Multiplications	32 (32)	64	96	128	$2^{\eta+1} \log_2 n'$
		Memory	32 (256)	64	128	256	$2^\eta n'$

[1] Number of multiplications per scalar value in encoding.

[2] Number of scalar values for codewords.

[3] m-SAPQ.

km of the two-dimensional 1-SAPQ and one-dimensional 1-SAPQ are kept the same, then the gain of the two-dimensional 1-SAPQ over the two-dimensional PQ would be less than the gain from the one-dimensional 1-SAPQ over one-dimensional PQ. The reason is that the two-dimensional PQ would have a lower distortion than the one-dimensional PQ, but since km is the same, the one-dimensional and two-dimensional 1-SAPQ are bounded from below by the same km-dimensional VQ distortion. Hence, to use a higher dimension 1-SAPQ it makes sense to increase the adaptation period m (hence, the value of km) accordingly.

B. Performance Comparison and Trellis-Coded Quantization (TCQ)

In Table II, VQ and TSVQ are compared with several SAPQ's in terms of their distortions (dB), number of multiplications for encoding, and required memory for the codewords. In this table, VQ and TSVQ are designed using GLA, where in the TSVQ, the breadth is equal to 2 [13]. For the quantizer dimension 2 (Quan. Dim. (km) is equal to 2 in Table II), the VQ distortions shown are the lowest ones for each bit rate. However, the required multiplications and memory for VQ are very large compared to those of 1-SAPQ. In the TSVQ case, the multiplications are reduced, but TSVQ requires even more memory than the traditional VQ case. On the other hand, the 1-SAPQ scheme can obtain a comparable level of performance and requires only a relatively small number of multiplications and memory. In the 1-SAPQ case, we can further increase the gains by increasing m to $m = 4, 8$ (this corresponds to the quantizer dimension of 4 and 8 in the table), if this is done carefully. For example, see Table II, columns $R = 2.5, 3.5$, and 4.5. However, for column $R = 1.5$, increasing m actually increases the distortion. This is because in the $R = 1.5$ case, the value of β is relatively large ($\beta = 2$ for $m = 4$ and $\beta = 4$ for $m = 8$). This observation also corresponds to Fig. 4(a) and (b), earlier. In this case ($R = 1.5$), the m-SAPQ can achieve gains over VQ, TSVQ, and 1-SAPQ (See Table II). Similar simulations are performed on a Laplacian i.i.d. source, and summarized in Table III. Further, in Table IV, 1-SAPQ is compared with PQ, and km-dimensional VQ and TSVQ at a bit rate of 0.5, where $k \neq 1$. In the Gaussian d.f. case, the 1-SAPQ distortions are very close to those of the km-dimensional VQ.

In Fig. 8, 1-SAPQ is compared with the different coding schemes, TCQ [26] and entropy-coded quantization [18], [11]. TCQ uses the Viterbi algorithm to encode memoryless sources,

TABLE III
DISTORTION (dB) COMPARISON OF SAPQ

Quan. Dim. (km)	Quantizer	$R = 1.5$	$R = 2.5$	$R = 3.5$	$R = 4.5$
2	VQ	-6.30	-11.5	-17.2	-22.9
2	TSVQ (Breadth = 2)	-6.24	-11.0	-16.3	-22.1
2	1-SAPQ ($\eta = 1$)	-5.84 (-5.85)[1]	-10.8	-16.3	-22.0
4	1-SAPQ ($\eta = 2$)	-6.37 (-6.38)	-11.5	-17.2	-23.0
8	1-SAPQ ($\eta = 4$)	-6.05 (-7.08)	-12.4	-18.1	-23.9

[1] m-SAPQ.

TABLE IV
DISTORTION (dB) COMPARISON BETWEEN 1-SAPQ, PQ, km-DIMENSIONAL VQ

Quan. Dim. (km)	Quantizer	Gaussian i.i.d.	Laplacian i.i.d.	Multiplications	Memory
8	VQ ($\nu = 16$)	-2.22	-2.69	16	128
	TSVQ (Breadth=2, $d = 4$)	-1.68	-1.98	8	240
	1-SAPQ ($k = 4$, $m = 2$, $\eta = 2$)	-2.20	-2.06	8	32
	PQ ($k = 4$, $m = 2$, $n = 4$)	-1.89	-1.69	4	16
12	VQ ($\nu = 64$)	-2.32	-2.73	64	768
	TSVQ (Breadth=2, $d = 6$)	-1.70	-2.01	12	1512
	1-SAPQ ($k = 4$, $m = 3$, $\eta = 3$)	-2.24	-2.50	16	64
	1-SAPQ ($k = 6$, $m = 2$, $\eta = 2$)	-2.27	-2.59	16	96
	PQ ($k = 4$, $m = 3$, $n = 4$)	-1.89	-1.69	4	16
	PQ ($k = 6$, $m = 2$, $n = 8$)	-2.07	-2.10	8	48

and uses a sliding-block decoder to reconstruct the quantized signal. TCQ can also be regarded as a structurally constrained VQ of length kL where L is the searching depth in the trellis of TCQ [18]. In the 1-SAPQ's of Fig. 8, the quantizer dimensions km are equal to 16, and in TCQ, $k = 1$, $L = 1000$, and the number of states in the trellis is 16. We found that the 1-SAPQ distortions can be very close to that of TCQ when km in 1-SAPQ and the trellis size is equal. For bit rates around 2–2.5 in Fig. 8, if $k = 1$, since the sample ratio is large ($\beta = m/n' = 4$), we cannot achieve good gains from 1-SAPQ. However, when $k = 2$, the sample ratio decreases to $\beta = 0.5$, hence we can obtain better gains than in the $k = 1$ case. (In Fig. 8, only the $k = 2$ case is depicted above bit rate 2. Note that the 1-SAPQ with $k = 1$ and $n' = 8$ has $\beta = 2$ and the 1-SAPQ with $k = 1$ and $n' = 16$ has $\beta = 1$.) The entropy coded quantization has variable-length outputs, hence the coding scheme suffers from error propagation, data loss, and buffer control problems. TCQ also has an error propagation problem within the trellis size, and requires a long search depth L in the Viterbi algorithm [26]. However, in SAPQ there is no such error propagation, unless the side bits are corrupted. Fortunately, even if the side bits were corrupted, the produced error would be relatively small, since the designed codebooks are very similar to each other. Further, unlike the long search depth in TCQ, by using relatively short data in SAPQ, e.g., $km = 16$, we can get large gains. If we were to combine SAPQ with entropy coders, we can further improve upon these gains. As can be observed in Fig. 8, SAPQ with entropy coding achives better gains than TCQ with entropy coding. Note that in Fig. 8, the curve 1-SAPQ Ent. corresponds to 1-SAPQ distortion with respect to the entropy H_{SAPQ} defined by

$$H_{\mathrm{SAPQ}} := -\frac{1}{k} \sum_{j=1}^{2^\eta} \Pr\{C_j \text{ used for } \boldsymbol{X}\} \sum_{\ell=1}^{n'} P_{j\ell} \log_2 P_{j\ell} + \frac{\eta}{km} \quad (30)$$

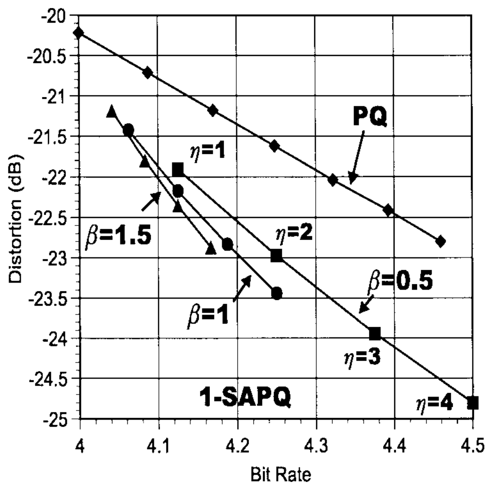
where

$$P_{j\ell} := \frac{1}{m} \sum_{i=1}^{m} \Pr\{\ell\text{th codeword used for } \boldsymbol{X}_i | C_j \text{ used for } \boldsymbol{X}\}.$$

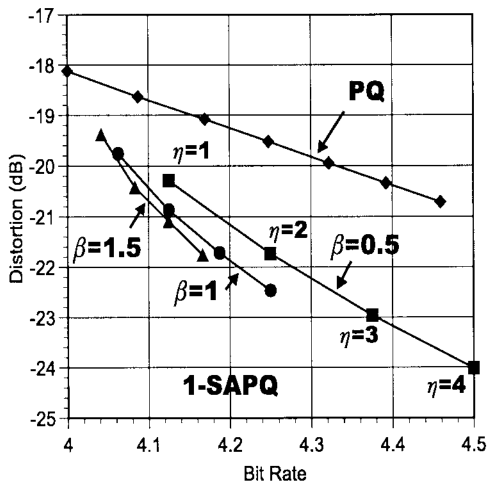
Also note that in the literature, TCQ schemes have been modified so that they can be applied to Markov sources based on predictive coding schemes [13], [26]. In a similar way, the proposed SAPQ scheme can also be modified for quantizing Markov sources. Furthermore, SAPQ can easily be applied to quantization of nonindependent and/or nonidentically distributed signals by using different codebooks for each random vector $\boldsymbol{X}_i$ as illustrated by m-SAPQ.

VII. CONCLUSION

In this paper we have proposed a novel coding scheme called the *sample-adaptive product quantizer*. This quantizer is



(a)



(b)

Fig. 6. Distortions (dB) of 1-SAPQ at $n' = 16$. (The results are obtained by varying η for each β. PQ uses the Lloyd–Max quantizers.) (a) Gaussian i.i.d. with unit variance. (b) Laplacian i.i.d. with unit variance.

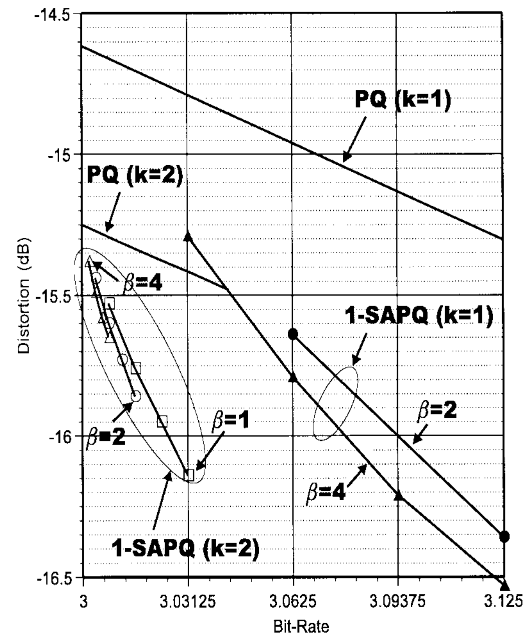


Fig. 7. Distortions (dB) of 1-SAPQ for a Gaussian i.i.d. source with unit variance. (The results are obtained by varying η for each β. The codebook sizes are $n' = 8$ and $n' = 64$ for $k = 1$ and $k = 2$, respectively. PQ uses the Lloyd–Max quantizers.)

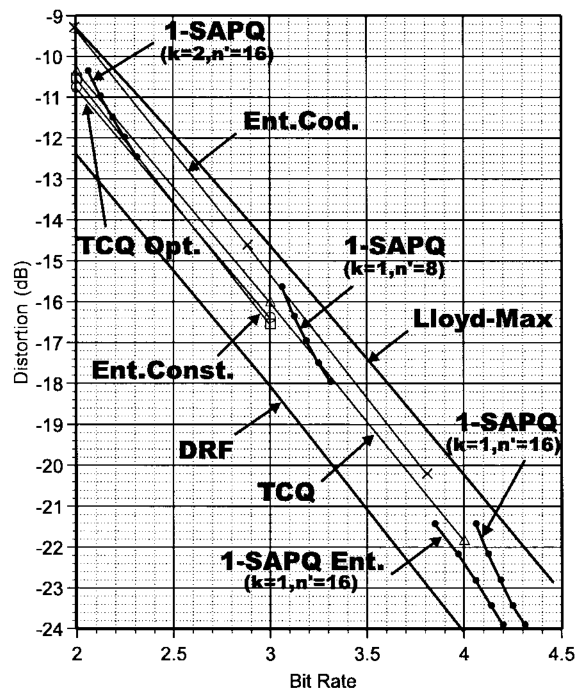


Fig. 8. Distortions (dB) for various coding schemes for a Gaussian i.i.d. source with unit variance. (DRF: distortion rate function, TCQ Opt: TCQ using optimized alphabets [34], TCQ: TCQ using rate-$(R+1)$ Lloyd–Max [34], Ent. Const.: entropy contraint optimum [25], Ent. Cod.: entropy-coded quantization, Lloyd–Max with Huffman [8], 1-SAPQ Ent.: 1-SAPQ with respect to its entropy. 1-SAPQ results are obtained by varying $\eta = 1, \cdots, 5$, for each k and n', and trellis size in TCQ and quantizer dimension km in 1-SAPQ are equal to 16.)

in the form of a structurally constrained vector quantizer. The main principle of SAPQ is using *unions of product codebooks* in km dimensions, in order to obtain gains while reducing the complexity of quantizer. SAPQ uses more than one codebook and, adapting to each sample, selects an appropriate codebook from the previously designed codebooks available at both the encoder and decoder. Based on this new concept, we have proposed two coding schemes; m-SAPQ, which requires $m2^{\eta}$ codebooks, and 1-SAPQ, a simplified version of m-SAPQ which requires 2^{η} codebooks. The performance of m-SAPQ is always better than 1-SAPQ. However, under appropriate design guidelines, the performance of 1-SAPQ can be made to closely approximate that of m-SAPQ. The quantization of both schemes is based on k-dimensional VQ's. One can obtain comparable performance to SAPQ in the scalar quantization case, by using a κ-dimensional VQ $(1 < \kappa < m)$. However, for the same performance, the encoding complexity of the κ-dimensional VQ would be significantly higher. We also show that a number of important lattice-based vector quantizers can be constructed using scalar quantization in SAPQ.

In order to implement SAPQ, we have proposed and simulated a simple iterative algorithm for the design of the codebooks. Through numerical studies, we have found that, for scalar quantization, the sample adaptive coding scheme significantly outperforms the Lloyd–Max-based quantizer for i.i.d. sources, with typical gains in distortion being between 1 and 3 dB. In general, we also show that based on the k-dimensional VQ structure, the performance of SAPQ can be made comparable to a $k\kappa$-dimensional VQ.

Since most commercial data compression systems for video and audio are based on scalar quantization, scalar prediction, and PQ, applying SAPQ with $k=1$ to those systems has the potential for significant performance gains over the state-of-the-art, while maintaining the same basic scalar structure of the quantizer.

Appendix A
Proof of Proposition 1

Let $\mathcal{C}'_{n'}$ denote the direct limit of the increasing sequence $(\mathcal{C}^{\eta}_{n'})_{\eta}$, i.e., $\mathcal{C}'_{n'} := \lim \mathrm{dir}_{\ell\to\infty} \mathcal{C}^{\eta}_{n'}$. Note that $\mathrm{card}\,(\mathcal{C}^{\eta}_{n'})$, the cardinality of $\mathcal{C}^{\eta}_{n'}$, is equal to 2^{η}. Given n' and m, since the sequence $(\inf_{C\in\mathcal{C}^{\eta}_{n'}} \delta(C, F^{\omega}_m))_{\eta}$ is monotonically decreasing and bounded

$$\lim_{\eta\to\infty} \inf_{C\in\mathcal{C}^{\eta}_{n'}} \delta(C, F^{\omega}_m) = \inf_{C\in\mathcal{C}'_{n'}} \delta(C, F^{\omega}_m), \qquad \text{for every } \omega. \tag{A.1}$$

Also every subsequence of $(\mathcal{C}^{\eta}_{n'})_{\eta}$ satisfies

$$\mathrm{card}\,(\mathcal{C}^{\eta}_{n'}) \to \mathrm{card}\,(\mathcal{C}'_{n'}) = \mathrm{card}\,(\mathcal{C}_{n'})$$

where $\mathrm{card}\,(\mathcal{C}_{n'})$ is the cardinal number of the power set of the natural number set, i.e., the cardinal number of the continuum. Thus since $\mathcal{C}'_{n'}$ contains all the possible codebooks, it follows that

$$\inf_{C\in\mathcal{C}'_{n'}} \delta(C, F^{\omega}_m) = \inf_{C\in\mathcal{C}_{n'}} \delta(C, F^{\omega}_m), \qquad \text{for every } \omega. \tag{A.2}$$

Therefore, from the *Dominated Convergence Theorem* [16, p. 110], Proposition 1 follows.

Appendix B
Proof of Proposition 2

It is clear that

$$\inf_{C\in\mathcal{C}_{n'}} \delta(C, F^{\omega}_m) \le \inf_{C\in\mathcal{C}^{\eta}_{n'}} \delta(C, F^{\omega}_m) \le \delta(C^*, F^{\omega}_m) \tag{B.1}$$

holds for every $\omega \in \Omega$, when $\mathcal{C}^{\eta}_{n'}$ includes the F-optimal codebook $C^*(\in \mathcal{C}_{n'})$ (if it exists). From [28, Consistency Theorem] and [1, Theorem 1]

$$\lim_{m\to\infty} \inf_{C\in\mathcal{C}_{n'}} \delta(C, F_m) = \inf_{C\in\mathcal{C}_{n'}} E\left\{\frac{1}{k} \min_{\boldsymbol{y}\in C} \|\boldsymbol{X}_1 - \boldsymbol{y}\|^2\right\} \quad \text{almost surely.} \tag{B.2}$$

It follows that since $\delta(C^*, F_m)$ converges to the F-optimal distortion almost surely from the *Strong Law of Large Numbers* [23, p. 204], the sequence of the sample-adaptive distortions converges to the F-optimal distortion almost surely for fixed n' and η, and Proposition 2 follows.

Appendix C
Proof of Theorem 1

Consider a cube $U = ([-a/2, a/2])^{\kappa} (\subset \mathbb{R}^{\kappa})$ for a vector dimension κ and a positive constant a. For $\zeta = 1, 2, \cdots$, let a sequence (ϕ_{ζ}) be $\phi_{\zeta} := \lfloor \alpha\zeta^{1/\kappa} \rfloor$, where $\lfloor c \rfloor$, $c \in \mathbb{R}$, is the largest integer less than or equal to c, where α is a positive constant. First, divide the cube U into $(\phi_{\zeta})^{\kappa}$ subcubes. Consider a sequence of equivalent lattices of Λ_{κ}. By truncating the equivalent lattice into one of the subcubes, we can design a Λ_{κ} lattice VQ for the subcube, which satisfies

$$(\phi_{\zeta})^{\kappa}\zeta \cdot \mu(H_{\zeta}) \ge \mu(U) \tag{C.1}$$

and

$$\lim_{\zeta\to\infty} (\phi_{\zeta})^{\kappa}\zeta \cdot \mu(H_{\zeta}) = \mu(U) \tag{C.2}$$

where the lattice VQ has less than or equal to ζ codewords. Here $(H_{\zeta})_{\zeta}$ is a polytope sequence corresponding to the sequence of equivalent lattices, and μ is the Lebesgue measure. If we design the same lattice VQ for all subcubes, then the quantizer for cube U can be regarded as a variation of m-SAPQ that has the codebooks $C_{i,j}$, for $i = 1, \cdots, \kappa$ and $j = 1, \cdots, \overline{\zeta}$, where $\overline{\zeta} \le \zeta$, $|C_{i,j}| = n' = \phi_{\zeta}$, $m = \kappa$, and $k = 1$. Hence we have

$$\begin{aligned}
&[(\phi_{\zeta})^{\kappa}\zeta]^{2/\kappa} \int\cdots\int_U \min_{j\in\{1,\cdots,\overline{\zeta}\}} \frac{1}{\kappa}\sum_{i=1}^{\kappa} \\
&\qquad \cdot \min_{y\in C_{i,j}} (x_i - y)^2 \frac{1}{\mu(U)} dx_1\cdots dx_{\kappa} \\
&\quad \le [(\phi_{\zeta})^{\kappa}\zeta]^{2/\kappa} \frac{1}{\kappa} \sum_{\ell=1}^{(\phi_{\zeta})^{\kappa}} \sum_{j=1}^{\zeta} \int_{H_{\zeta}} \|\boldsymbol{x} - \boldsymbol{y}_{\zeta}\|^2 \frac{1}{\mu(U)} d\boldsymbol{x} \\
&\quad \le G(\Lambda_{\kappa})[(\phi_{\zeta})^{\kappa}\zeta\mu(H_{\zeta})]^{1/\rho} \frac{1}{\mu(U)}
\end{aligned} \tag{C.3}$$

where $\rho := \kappa/(\kappa+2)$ and $\boldsymbol{y}_{\zeta}$ is the centroid of the polytope H_{ζ}. It follows from (C.2) that

$$\limsup_{\zeta\to\infty} [(\phi_{\zeta})^{\kappa}\zeta]^{2/\kappa} \int_U \min_{j\in\{1,\cdots,\overline{\zeta}\}} \frac{1}{\kappa}\sum_{i=1}^{\kappa} \min_{y\in C_{i,j}} (x_i - y)^2 \frac{1}{\mu(U)} d\boldsymbol{x} \le G(\Lambda_{\kappa})\,[\mu(U)]^{2/\kappa}. \tag{C.4}$$

Now consider a density function g with compact support. Assume that g is bounded on $\mathbb{R}^{\kappa}$. Let B $(\in \mathbb{R}^{\kappa})$ be a cube that contains the support of g and is defined by $B := ([a, b])^{\kappa}$, where a and b are finite. Consider a partition of B into $2^{\kappa q}$ cubes B_{ℓ} such that

$$\mu(B_{\ell}) = [(b-a)/2^q]^{\kappa} =: v, \qquad \ell = 1, \cdots, 2^{\kappa q}.$$

Define a simple function g_q as

$$g_q(\boldsymbol{x}) := \sum_{\ell=1}^{2^{\kappa q}} p_{\ell} I_{B_{\ell}}(\boldsymbol{x})$$

where $p_{\ell} := \sup_{\boldsymbol{x}\in B_{\ell}} g(\boldsymbol{x})$ and $I_{B_{\ell}}(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in B_{\ell}$, and $I_{B_{\ell}}(\boldsymbol{x}) = 0$ otherwise. Then since the sequence $(g_q(\boldsymbol{x}))_q$ is monotonic and $\lim_{q\to\infty} g_q(\boldsymbol{x}) = g(\boldsymbol{x})$ a.e., it follows that $\int g_q(\boldsymbol{x})\, d\boldsymbol{x} \to 1$ [16, p. 112]. Let $\phi_\zeta := \lfloor \alpha\zeta^{1/\kappa} \rfloor$ and

$$\zeta_\ell := \left\lfloor \zeta \cdot \frac{(p_\ell)^\rho}{\sum_{j=1}^{2^{\kappa q}} (p_j)^\rho} \right\rfloor. \tag{C.5}$$

Note that $\phi_\zeta/\zeta^{(1/\kappa q)} \to \alpha$ and

$$\zeta_\ell/\zeta \to (p_\ell)^\rho \Bigg/ \sum_{j=1}^{2^{\kappa q}} (p_j)^\rho$$

as $\zeta \to \infty$. Using ϕ_ζ and ζ_ℓ make codebooks $C_{i,j}^\ell$ in the same manner as in (C.3). Then, from [16, p. 96], the quantizer distortion of a variation of m-SAPQ for g, $D^s_{m\text{-SAPQ}}(n', \zeta, g)$, where $k = 1$, $m = \kappa$, and $|C_{i,j}| = n' = \phi_\zeta$, for $i = 1, \cdots, \kappa$ and $j = 1, \cdots, \zeta$, satisfies the relation

$$\begin{aligned} &D^s_{m\text{-SAPQ}}(\phi_\zeta, \zeta, g) \\ &:= \inf_{C_{i,j} \in \mathcal{C}_{\phi_\zeta}} \int \min_{j\in\{1,\cdots,\zeta\}} \frac{1}{\kappa} \sum_{i=1}^{\kappa} \min_{y \in C_{i,j}} (x_i - y)^2 g(\boldsymbol{x})\, d\boldsymbol{x} \\ &\le \inf_{C_{i,j} \in \mathcal{C}_{\phi_\zeta}} \int \min_{j\in\{1,\cdots,\zeta\}} \frac{1}{\kappa} \sum_{i=1}^{\kappa} \min_{y \in C_{i,j}} (x_i - y)^2 g_q(\boldsymbol{x})\, d\boldsymbol{x} \\ &\le \sum_{\ell=1}^{2^{\kappa q}} \left[\int_{B_\ell} \min_{j\in\{1,\cdots,\overline{\zeta}_\ell\}} \frac{1}{\kappa} \sum_{i=1}^{\kappa} \min_{y \in C_{i,j}^\ell} (x_i - y)^2 \frac{1}{v} d\boldsymbol{x} \right] p_\ell v \end{aligned} \tag{C.6}$$

where $\overline{\zeta}_\ell \le \zeta_\ell$. From (C.4) we obtain

$$\begin{aligned} &\limsup_{\zeta\to\infty} [(\phi_\zeta)^\kappa \zeta]^{2/\kappa} \sum_{\ell=1}^{2^{\kappa q}} \\ &\quad \cdot \left[\int_{B_\ell} \min_{j\in\{1,\cdots,\overline{\zeta}_\ell\}} \frac{1}{\kappa} \sum_{i=1}^{\kappa} \min_{y \in C_{i,j}^\ell} (x_i - y)^2 \frac{1}{v} d\boldsymbol{x} \right] p_\ell v \\ &\le \limsup_{\zeta\to\infty} [(\phi_\zeta)^\kappa \zeta]^{2/\kappa} \sum_{\ell=1}^{2^{\kappa q}} \left[\frac{1}{(\phi_\zeta)^\kappa \zeta_\ell} \right]^{2/\kappa} G(\Lambda_\kappa) v^{2/\kappa} \cdot p_\ell v \\ &= G(\Lambda_\kappa) \|g_q\|_\rho. \end{aligned} \tag{C.7}$$

Hence, for a subsequence $(n_\eta)_\eta$ of $(\phi_\zeta)_\zeta$, we obtain

$$\begin{aligned} &\limsup_{q\to\infty} \limsup_{\eta\to\infty} [(n_\eta)^\kappa 2^\eta]^{2/\kappa} D^s_{m\text{-SAPQ}}(n_\eta, 2^\eta, g) \\ &\qquad \le \lim_{q\to\infty} G(\Lambda_\kappa) \|g_q\|_\rho = G(\Lambda_\kappa) \|g\|_\rho. \end{aligned} \tag{C.8}$$

We now consider a general density function f. Consider an increasing sequence of cubes $B^1 \subset B^2 \subset \cdots \subset B^s$. For a constant $0 < \lambda < 1$, assign $(1-\lambda)(n_\eta)^\kappa 2^\eta$ points to the cube B^s and $\lambda (n_\eta)^\kappa 2^\eta$ points to $\bar{B}^s$, which is the complement of B^s. Then from the assumption and [5, Theorem 2], there exists a sequence of codebooks $\boldsymbol{C}_\eta (|\boldsymbol{C}_\eta| = (n_\eta)^\kappa 2^\eta)$ such that

$$\lim_{s\to\infty} \lim_{\eta\to\infty} [(n_\eta)^\kappa 2^\eta]^{2/\kappa} \int_{\overline{B}^s} \min_{\boldsymbol{y} \in \boldsymbol{C}_\eta} \|\boldsymbol{x} - \boldsymbol{y}\|^2 f(\boldsymbol{x})\, d\boldsymbol{x} = 0. \tag{C.9}$$

Hence, by letting $\lambda \to 0$, we obtain

$$\limsup_{\eta\to\infty} [(n_\eta)^\kappa 2^\eta]^{2/\kappa} D^s_{m\text{-SAPQ}}(n_\eta, 2^\eta, f) \le G(\Lambda_\kappa) \|f\|_\rho. \tag{C.10}$$

Let $\kappa = km$, and since

$$D^o_{m\text{-SAPQ}}(n_\eta, 2^\eta) \le D^s_{m\text{-SAPQ}}(n_\eta, 2^\eta, f)$$

for $k \ge 1$, the theorem holds.

Appendix D
Encoding Complexity of 1-SAPQ

In this appendix, we analyze in detail the encoding complexity of 1-SAPQ. Consider the encoding process of 1-SAPQ for the mean-squared error distortion measure (i.e., $r = 2$). Now, (8) without the expectation, can be rewritten as

$$\frac{1}{km} \min_j \left[\sum_{i=1}^{m} (\min_{\boldsymbol{y} \in C_j} \|\boldsymbol{x}_i - \boldsymbol{y}\|^2) \right] \tag{D.1}$$

for given m input vectors $\boldsymbol{x}_i \in \mathbb{R}^k$, where $|C_j| = n'$ for all j. We first expand $\|\boldsymbol{x}_i - \boldsymbol{y}\|^2$ to obtain

$$\|\boldsymbol{x}_i - \boldsymbol{y}\|^2 = \boldsymbol{x}_i^T \boldsymbol{x}_i - 2\left(\boldsymbol{x}_i^T \boldsymbol{y} - \frac{\boldsymbol{y}^T \boldsymbol{y}}{2} \right). \tag{D.2}$$

In (D.2), the $\boldsymbol{x}_i^T \boldsymbol{x}_i$ term is independent of the codewords to be compared, while the $\psi := \boldsymbol{y}^T \boldsymbol{y}/2$ term is given by the codebook. Therefore, for an input $\boldsymbol{x}_i$, we need to calculate $\boldsymbol{x}_i^T \boldsymbol{y}$ and add the ψ that was previously obtained for a given C_j. Note that this set of operations requires k additions and k multiplications. We then find the codeword index that minimizes $\boldsymbol{x}_i^T \boldsymbol{y} + \psi$. In order to compare two constants, one addition (subtraction) is required. Hence, $(k+1)n'$ additions and kn' multiplications are required for finding the codeword index in the term enclosed by parentheses in (D.1).

Since for each value of i in the sum given by (D.1), we need $(k+1)n'$ additions and kn' multiplications, we need a total of $(k+1)mn'$ additions and kmn' multiplications to calculate the m-codebook sample distance in the term enclosed by brackets in (D.1).

Now we focus on searching through the index j in (D.1). We have 2^η different code sequences. Thus we require $(k+1)mn'2^\eta$ additions, and $kmn'2^\eta$ multiplications to calculate the distances, and 2^η additions for the comparison of 2^η distances. In summary, in order to perform the 1-SAPQ encoding, we need a total of $(1 + 1/k)n' + 2^\eta/km$ additions per scalar input and $n'2^\eta$ multiplications per scalar input. The number of multiplications can be rewritten as $2^{kR + (1-1/m)\eta}$, R is the 1-SAPQ bit rate given by $R = (1/k)\log_2 n' + \eta/km$. Consequently, as R increases, the encoding complexity of 1-SAPQ is $\mathcal{O}(2^{(1-1/m)\eta} 2^{Rk})$. However, for a comparable performance of $k\kappa$-dimensional VQ at the same bit rate R, where $1 < \kappa < m$, the encoding complexity is $\mathcal{O}(2^{Rk\kappa})$ [9].

For the special case of 1-SAPQ when $k = 1$, the codeword searching process can be conducted based on the *tree-structured search*. Hence, in 1-SAPQ, the required number of multiplications are reduced to $2^{\eta+1}\log_2 n'$, i.e., $\mathcal{O}(2^{\eta+1} R)$.

Acknowledgment

The authors wish to thank Prof. S. Y. Jang of the Department of Mathematics at the University of Ulsan, Korea, for helpful comments.

REFERENCES

[1] E. A. Abaya and G. L. Wise, "Convergence of vector quantizers with applications to optimal quantization," *SIAM J. Appl. Math.*, vol. 44, pp. 183–189, 1984.

[2] M. R. Anderberg, *Clustering Analysis for Applications*. New York: Academic, 1973.

[3] E. S. Barnes and N. J. A. Sloane, "The optimal lattice quantizer in three dimensions," *SIAM J. Alg. Discr. Methods*, vol. 4, no. 1, pp. 30–41, Mar. 1983.

[4] P. Billingsley, *Probability and Measure*, 3rd. ed. New York: Wiley, 1995.

[5] J. A. Bucklew and G. L. Wise, "Multidimensional asymptotic quantization theory with rth power distortion measures," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 239–247, Mar. 1982.

[6] 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.

[7] ——, "On the voronoi regions of certain lattices," *SIAM J. Alg. Discr. Methods*, vol. 5, no. 3, pp. 294–305, Sept. 1984.

[8] ——, "A lower bound on the average error of vector quantizers," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 106–109, Jan. 1985.

[9] G. A. Davison, P. R. Cappellow, and A. Gersho, "Systolic architectures for vector quantization," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1651–1664, Oct. 1988.

[10] M. Effros, "Optimal modeling for complex system design," *IEEE Signal Processing Mag.*, vol. 15, pp. 51–73, Nov. 1998.

[11] N. Favardin 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, May 1984.

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

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

[14] J. D. Gibson and K. Sayood, "Lattice Quantization," *Adv. Electronics Electron Phys.* vol. 72, pp. 259–330, 1988.

[15] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2325–2383, Oct. 1998.

[16] P. R. Halmos, *Measure Theory*. New York: Springer-Verlag, 1974.

[17] J. E. Humphreys, *Introduction to Lie Algebras and Representation Theory*. New York: Springer-Verlag, 1972.

[18] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[19] J. C. Kieffer, "Uniqueness of locally optimal quantizer for log-concave density and convex error weighting function," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 42–47, Jan. 1983.

[20] ——, "A survey of the theory of source coding with a fidelity criterion," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1473–1490, Sept. 1993.

[21] D. S. Kim, T. Kim, and S. U. Lee, "On testing trained vector quantizer codebooks," *IEEE Trans. Image Processing*, vol. 6, pp. 398–406, Mar. 1997.

[22] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 128–137, Mar. 1982.

[23] M. Loéve, *Probability Theory I*, 4th. ed. New York: Springer-Verlag, 1977.

[24] T. D. Lookabaugh and R. M. Gray, "High-resolution quantization theory and the vector quantizer advantage," *IEEE Trans. Inform. Theory*, vol. 35, pp. 1020–1033, Sept. 1989.

[25] J. Makhoul, S. Roucos, and H. Gish, "Vector quantization in speech coding," *Proc. IEEE*, vol. 73, pp. 1551–1588, Nov. 1985.

[26] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of memoryless and Gauss-Markov source," *IEEE Trans. Commun.*, vol. 38, pp. 82–93, Jan. 1990.

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

[28] D. Pollard, "Strong consistency of k-means clustering," *Ann. Statist.*, vol. 9 no. 1, pp. 135–140, 1981.

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

[30] V. Vapnik *Estimation of Dependences Based on Empirical Data*. New York: Springer-Verlag, 1982.

[31] P. Zador, "Asymptotic quantization error of continuous signals and the quantization dimension," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 139–149, Mar. 1982.