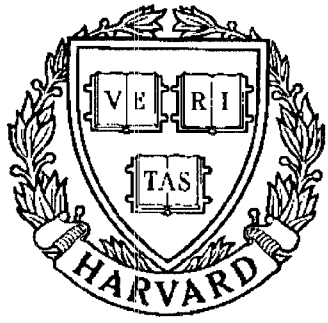
TECHNICAL RESEARCH REPORT

SYSTEMS RESEARCH CENTER



Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

On Optimal Shaping of Multidimensional Constellations - An Alternative Approach to Lattice-Bounded (Voronoi) Constellations

by R. Laroia

TR 92-1

On Optimal Shaping of Multidimensional Constellations — An Alternative Approach to Lattice-Bounded (Voronoi) Constellations[†]

Rajiv Laroia
Electrical Engineering Department and
Systems Research Center
University of Maryland
College Park, Maryland 20742

Abstract

A scheme for the optimal shaping of multidimensional constellations is proposed. This scheme uses some of the ideas from a type of structured vector quantizer originally proposed for the quantization of memoryless sources, and results in N-sphere shaping of N-dimensional cubic lattice based constellations. Its implementation complexity is very reasonable. Because N-sphere shaping is optimal in N dimensions, shaping gains higher than those of N-dimensional Voronoi constellations can be realized. Optimal shaping for a large N however has the undesirable effect of increasing the size and the peak-to-average power ratio of the constituent 2D constellation, thus limiting its usefulness in practical implementation over QAM modems. It is shown that the proposed scheme alleviates this problem by achieving optimal constellation shapes for a given limit on the constellation expansion ratio or the peak-to-average power ratio of the constituent 2D constellation. Finally, compatibility with trellis-coded modulation is demonstrated for the realization of both shaping and coding gain, giving this scheme a distinct edge over lattice-bounded constellations.

Index Terms: Multidimensional constellations; SVQ shaping; Optimal Shaping; Voronoi constellations; Trellis coding.

[†] This work was supported in part by National Science Foundation grants NSFD MIP-86-57311 and NSFD CDR-85-00108.

I. Introduction

The problem of data transmission is the dual of the quantization problem and this duality is formally described in [1]. Advances in transmission theory have therefore resulted in useful insight and new quantization techniques (and vice versa). One such example is the use of the idea behind trellis-coded modulation [2],[3] for the quantization of data [4],[5]. Another example is the quantization of memoryless sources using lattice- and trellis-bounded codebooks [6], a topic extensively studied in the context of modulation. This paper applies some of the ideas from a structured vector quantizer developed for memoryless sources [7] to the optimal shaping of multidimensional constellations.

Shaping of multidimensional constellations has been studied in detail in [8] and [9]. While [8] contains an excellent introduction to the problem and discusses cross and generalized cross constellations, [9] covers lattice-bounded constellations. Most of the terminology in this paper is adopted from [8]. Some of the relevant terms defined there are now briefly described.

A constellation C generally consists of a set of points on an N-dimensional lattice Λ that are enclosed within a finite region $\mathcal{R}$. The simplest N-dimensional constellation consists of all the points on a cubic lattice enclosed within an N-cube. This is the baseline system and the performance of all the more complex constellations is measured in terms of gains over this constellation. There are two kinds of gains that can be achieved over the baseline system. The first is obtained by using a more densely packed N-dimensional lattice than the N-dimensional cubic lattice and is called the *coding gain* γ_c. The second is the *shaping gain* γ_s that results from using a more spherical bounding region $\mathcal{R}$ than an N-cube. When the region $\mathcal{R}$ is big and encloses a large number of lattice points, the distribution of points in $\mathcal{R}$ can be approximated by a continuous uniform distribution over $\mathcal{R}$. This is the *continuous approximation* and, unless mentioned otherwise, in this paper we assume that it holds. Under the continuous approximation, the coding gain is decoupled from the shaping gain and both can be realized independently. The topic of coding gain is addressed in [10]. Here we focus on the shaping gain which is defined as the ratio of the average energy of a baseline constellation with the same number of points as the given constellation, to the average energy of the given constellation. Under the continuous approximation, this is the same as the inverse ratio of the average energy of the constellation region $\mathcal{R}$ to the average energy of the region bounded by an N-cube of the

same volume as $\mathcal{R}$; this is also referred to as the shaping gain of the region $\mathcal{R}$. The region that has the smallest average energy for a given volume is an N-sphere. For a given N therefore, the shaping gain is bounded by the shaping gain of an N-sphere. The maximum possible shaping gain is 1.423 (1.53 dB) which is the limit of the N-sphere shaping gain as N becomes large. The shaping gains of the generalized cross constellations described in [8] are limited to about 0.4 dB while the Voronoi constellations of known lattices [9] can achieve gains as high as 1.1 dB.

A *constituent* 2D *constellation* of a given N-dimensional (throughout this paper we assume an even N) constellation C is the set of all values that a given two-dimensional symbol takes as the N-dimensional signal points range through C. The constellation C is termed as a 2D-*symmetric constellation* if it has the same constituent 2D constellation for all possible pairs of dimensions. In this case we say that *the* constituent 2D constellation of C is C_2. For constellations that are not 2D-symmetric, C_2 is taken as the union of all the different constituent 2D constellations. Some attributes of the constituent 2D constellations are important from the view point of implementation over QAM (quadrature amplitude modulation) modems and are described next.

The *shaping constellation expansion ratio* CER_2 of the constituent 2D constellation C_2 of C is defined as the ratio of the size $|\mathrm{C}_2|$ of C_2 to the size $|\mathrm{C}|^{2/N}$ of the constituent 2D constellation of a baseline N-cube bounded constellation containing the same number of points as in C. Therefore $\mathrm{CER}_2 = |\mathrm{C}_2|/|\mathrm{C}|^{2/N}$ and is lower-bounded by unity. The *peak-to-average power ratio* (PAR_2) of C_2 is defined as the ratio of the squared-distance of the farthest point(s) in C_2 from the origin, to the average energy of points in C normalized to two dimensions (assuming all points in C are equally probable). The PAR_2 of the baseline N-cube bounded constellation is 3. For implementation on QAM modems it is desirable to have both, a small PAR_2 and a small CER_2 [8],[10].

The Voronoi region $\bar{\mathcal{R}}_V(\Lambda) \in \mathbb{R}^N$ of an N-dimensional lattice Λ is defined as the set of points in $\mathbb{R}^N$ that are at least as close (in the Euclidean distance sense) to the origin as to any other point in the lattice. The conventional approach of bounding the constellation by the Voronoi region of a lattice is based on the fact that the Voronoi regions of some N-dimensional lattices can approximate an N-sphere (especially for large N). In [11] it is shown that algorithmic indexing or labeling of the points in such constellations can be performed (look-up tables are generally impractical for a large N) leading to mod-

est complexity encoding/decoding algorithms. Although for large N the lattice-bounded constellations can achieve a significant portion of the maximum possible shaping gain of 1.53 dB, this usually comes at the cost of an equally significant increase in CER_2 and PAR_2. Bounds obtained in [8] (see Figures 2 and 3 of this paper) show that it is at least in principle possible to realize the same shaping gain as some of the best known lattices with a significantly smaller CER_2 and PAR_2.

We now describe the contribution of this paper. A structured vector quantizer was introduced by Laroia and Farvardin in [7] (also see [12] and [13]). The structure of the codebook of this structured vector quantizer (SVQ) is derived from a variable-length scalar quantizer. Here, we borrow some ideas from the SVQ for the shaping of multidimensional constellations. Of particular interest in this context are the codevector encoding and decoding algorithms of the SVQ. These algorithms index (label) each vector of the codebook of an N-dimensional SVQ with a unique Nr-bit binary number c (codeword), where r is the rate of the SVQ in bits/sample. We show that optimal N-sphere shaping of an N-dimensional cubic lattice based constellation is possible by first representing such a constellation as the codebook of an SVQ and then using the encoding and decoding algorithms of the SVQ to index the constellation points. The complexity of implementation of this scheme is very reasonable. The optimal (N-sphere) shaping however comes at the cost of a large CER_2 and PAR_2 both of which are undesirable for a QAM modem based implementation. We show here that it is possible to use the SVQ ideas for the optimal shaping of cubic lattice based constellations given a constraint on their maximum CER_2 or PAR_2. The Voronoi constellations perform rather poorly in this respect, giving the present approach an advantage. Compatibility of this approach with trellis-coded modulation is also demonstrated making it possible to realize optimal or near optimal shaping gain and significant coding gain while maintaining a small CER_2 and PAR_2.

In the next section we begin with a brief description of the SVQ. The coding/decoding algorithms of the SVQ are also given and N-sphere shaping of constellations is discussed. Section III describes the optimal shaping with a CER_2 constraint. Compatibility with trellis-coded modulation is demonstrated in Section IV and some characteristics of constellations shaped using the present approach are discussed in Section V.

II. SVQ Shaping of Constellations

For the rest of this paper the shaping of constellations based on the structured vector quantizer codebook is referred to as SVQ shaping. This section deals with N-sphere SVQ shaping. We start with a brief description of the SVQ; more details can be found in [7],[13].

2.1 The structured vector quantizer

The SVQ is a special kind of vector quantizer (VQ) in which the codebook structure is derived from a variable-length scalar quantizer $\mathcal{S}$. Let $\mathcal{Q} \equiv \{q_1, q_2, ..., q_n\}$ be the set of n quantization levels of $\mathcal{S}$ and $\mathcal{L} \equiv \{\ell_1, \ell_2, ..., \ell_n\}$ be the corresponding set of integer lengths, where ℓ_i; $i \in J_n \equiv \{1, 2, ..., n\}$ is the length associated with the quantization level q_i. These lengths can be arbitrary positive integers and $\mathcal{L}$ does not have to be the set of codeword lengths of a uniquely decodable code such as Huffman code. Now $\mathcal{Q}^N$, the N-fold cartesian product of $\mathcal{Q}$ with itself, is a grid of n^N points in $\mathbb{R}^N$. The total length of a point (N-tuple) in this grid is defined as the sum of the lengths of its N components. The codebook $\mathcal{Z}$ of an N-dimensional SVQ $\mathcal{V}$ derived from $\mathcal{S} \equiv (\mathcal{Q}, \mathcal{L})$ is a subset of $\mathcal{Q}^N$ consisting of only those points that have a total length no greater than an integer threshold L. The threshold L is chosen such that the codebook $\mathcal{Z}$ contains (at most) 2^{Nr} of the n^N total points in $\mathcal{Q}^N$, where r is the desired rate (in bits/sample) of the SVQ $\mathcal{V}$. This is formally described by the following definition.

Definition: An N-dimensional SVQ $\mathcal{V}$ derived from a variable-length n-level scalar quantizer $\mathcal{S} \equiv (\mathcal{Q}, \mathcal{L})$ is a VQ with a codebook $\mathcal{Z}$ given as,

$$\mathcal{Z} = \{\mathbf{z} \equiv (z_1, z_2, \ldots, z_N) \in \mathcal{Q}^N \ : \ \ell_{f(z_1)} + \ell_{f(z_2)} + \cdots + \ell_{f(z_N)} \leq L\},$$

where the index function $f : \mathcal{Q} \rightarrow J_n$ is defined as,

$$f(q_i) = i, \ i \in J_n \ .$$

For a rate r bits/sample SVQ, the threshold L is selected as the largest integer such that the cardinality of $\mathcal{Z}$ is no greater than 2^{Nr}.

With this structure of the SVQ codebook, there exist fast and efficient algorithms for codebook search and codevector encoding/decoding. Given a point in $\mathbb{R}^N$, codebook search amounts to determining the vector in the codebook that is closest, in some distance

measure, to the given point. This algorithm is not very relevant to the present discussion and is not described here. Of primary interest here are the threshold determination and the encoding/decoding algorithms of the SVQ and these are described next. The encoding/decoding algorithms label each vector in the SVQ codebook with a unique Nr-bit binary number.

Determination of the threshold L: For a given $\mathcal{L}$ and a desired per sample rate r, the threshold L can be obtained by counting the grid-points (starting with the ones that have the smallest total length) until there are 2^{Nr} points and then taking the largest total length for which all grid-points of that length are included in this collection.

Let M_i^j represent the number of distinct i-vectors $(v_1, v_2, ..., v_i) \in \mathcal{Q}^i$ such that their total length $\ell_{f(v_1)} + \ell_{f(v_2)} + \cdots + \ell_{f(v_i)} = j$. Then M_i^j satisfies the following recursive equation $\forall i \in J_N$:

$$M_i^j = \sum_{k=1}^{n} M_{i-1}^{j-\ell_k},$$

where $M_i^j = 0$ for $j < 0$ and $M_0^0 = 1$. The M_i^j can hence be determined by solving these equations. An algorithm similar in structure to the Viterbi algorithm will do this job. Define C_N^j as the number of N-vectors in $\mathcal{Q}^N$ that have a total length no greater than j. Clearly, $C_N^j = \sum_{k=1}^{j} M_N^k$. The threshold L is now given as, $L = \max\{j : C_N^j \leq 2^{Nr}\}$.

The inequality in the last equation ensures that there are no more than 2^{Nr} codevectors for a given rate r. In this paper however, we will use this algorithm in the data transmission context and for this purpose we modify the last equation to $L = \min\{j : C_N^j \geq 2^{Nr}\}$. This ensures that when we design a rate r bits/sample constellation, there are at least 2^{Nr} points in the constellation. The M_i^j; $\forall i \in J_{N-1}$; $\forall j \in J_L$, and C_N^j; $\forall j \in J_L$, that are a byproduct of threshold determination, can be stored in memory for use in the encoding/decoding algorithms that follow.

Encoding of codevectors: The codebook $\mathcal{Z}$ consists of 2^{Nr} codevectors. The encoder is a mapping which assigns a unique Nr-bit binary number or codeword to each of these codevectors. There are several possible ways to do this — the following algorithm implements one such mapping. This algorithm is a little different but essentially equivalent to the one in [7] and is better suited to the present task.

To each codevector $\mathbf{z} = (z_1, z_2, ..., z_N) \in \mathcal{Z}$ assign an N-digit base n number $\mathcal{M}(\mathbf{z}) = (f(z_1) - 1, f(z_2) - 1, ..., f(z_N) - 1) = \sum_{k=1}^{N} n^{N-k}(f(z_k) - 1)$. Clearly, $\mathcal{M}(\mathbf{z}_1) = \mathcal{M}(\mathbf{z}_2) \Leftrightarrow$

$\mathbf{z}_1 = \mathbf{z}_2$. All the codevectors are now ordered according to the following two rules— 1.) a codevector $\mathbf{z}_1$ is 'smaller than' a codevector $\mathbf{z}_2$ (i.e., $\mathbf{z}_1 < \mathbf{z}_2$) if $T(\mathbf{z}_1) < T(\mathbf{z}_2)$, where $T(\mathbf{z})$ denotes the total length of $\mathbf{z}$; and 2.) if $T(\mathbf{z}_1) = T(\mathbf{z}_2)$ then $\mathbf{z}_1 < \mathbf{z}_2$ if $\mathcal{M}(\mathbf{z}_1) < \mathcal{M}(\mathbf{z}_2)$. The encoder function $E : \mathcal{Z} \rightarrow \{0\} \cup J_{2^{Nr}-1}$ is defined as the number of vectors in $\mathcal{Z}$ that are smaller than the given codevector, i.e.,

$$\begin{aligned} E(\mathbf{z}) &= \sum_{\substack{\mathbf{w} \in \mathcal{Z} \\ T(\mathbf{w}) = T(\mathbf{z})}} I_{\mathbf{z},\mathbf{w}} + C_N^{T(\mathbf{z})-1}, \\ &= \mathcal{E}(\mathbf{z}) + C_N^{T(\mathbf{z})-1}, \end{aligned}$$

where $I_{\mathbf{z},\mathbf{w}} = 0$ if $\mathcal{M}(\mathbf{w}) \geq \mathcal{M}(\mathbf{z})$; 1 otherwise, and $C_N^0 = 0$. The function $\mathcal{E}(\mathbf{z})$ in the above equation gives the total number of length $T(\mathbf{z})$ codevectors that are smaller than $\mathbf{z}$. We can further write $\mathcal{E}(\mathbf{z}) = \sum_{k=1}^{N} \mathcal{E}_k(\mathbf{z})$, where $\mathcal{E}_k(\mathbf{z})$ is the number of length $T(\mathbf{z})$ codevectors $\mathbf{w}$ such that the base n representations of $\mathcal{M}(\mathbf{z})$ and $\mathcal{M}(\mathbf{w})$ are identical in the $k-1$ most significant digits, while the k^{th} digit of $\mathcal{M}(\mathbf{w})$ is smaller than that of $\mathcal{M}(\mathbf{z})$. It is simple to see that $\mathcal{E}_k(\mathbf{z}) = \sum_{j=1}^{f(z_k)-1} M_{N-k}^{T(\mathbf{z})-L_{k-1}-\ell_j}$; $k \in J_N$, where $L_0 = 0$ and $L_i = \sum_{j=1}^{i} \ell_{f(z_j)}$; $i \in J_N$. The encoder function can hence be written as,

$$E(\mathbf{z}) = \sum_{k=1}^{N} \sum_{j=1}^{f(z_k)-1} M_{N-k}^{T(\mathbf{z})-L_{k-1}-\ell_j} + C_N^{T(\mathbf{z})-1}.$$

Given that M_i^j and C_N^J; $\forall i \in J_{N-1}$; $\forall j \in J_L$, are computed once and stored in the memory, this equation shows that the total number of additions required (per dimension) for the encoding operation is upper-bounded by n (in fact it is approximately upper-bounded by n' $(< n)$, where n' is the number of distinct length values in $\mathcal{L}$). It should be noted though that each of C_N^j and M_i^j can be up to Nr bits long. An important feature of this algorithm (the usefulness of which will be discussed in Section V) is that the codevectors with the largest length are assigned the largest codewords. This is not the case with the encoding algorithm of [7].

Decoding of codevectors: The decoder function $E^{-1} : \{0\} \cup J_{2^{Nr}-1} \rightarrow \mathcal{Z}$, is the inverse of the encoder and assigns a unique codevector to every Nr-bit binary codeword. The decoding can be performed as follows. Given a codeword c, first compare it with the C_N^j; $j \in J_L$

and determine the length $T(\mathbf{z})$ of the codevector $\mathbf{z}$ corresponding to c. This is trivial. Let $c' = c - C_N^{T(\mathbf{z})-1}$. The decoding is now done one dimension at a time starting with z_1. The algorithm is straight-forward. Compare c' with the total number $M_{N-1}^{T(\mathbf{z})-\ell_{f(q_1)}}$ of codevectors beginning with q_1 that have length equal to $T(\mathbf{z})$. If $c' < M_{N-1}^{T(\mathbf{z})-\ell_{f(q_1)}}$, then $z_1 = q_1$; else compare $c' - M_{N-1}^{T(\mathbf{z})-\ell_{f(q_1)}}$ with the number of codevectors $M_{N-1}^{T(\mathbf{z})-\ell_{f(q_2)}}$ beginning with q_2 and so on until z_1 is determined. Now the problem reduces to an equivalent $(N-1)$-dimensional problem which can similarly be handled. This algorithm when implemented efficiently requires at most n' additions (subtractions) per dimension of the codevector.

The implementation complexities of the encoding and decoding operations are hence very reasonable even for relatively large n and N.

2.2 SVQ shaping

Consider the quantization of N-vectors from a stationary memoryless unit variance Gaussian source. By the asymptotic equipartition principle (AEP), for a large N, these N-vectors with a high probability lie inside an N-sphere of squared-radius N. To quantize this source therefore, one can use an N-dimensional VQ that has a codebook consisting of all points on some lattice that are enclosed inside an N-sphere. This is the rationale behind lattice-bounded lattice codebooks [6] where the boundary lattice has a Voronoi region that approximates an N-sphere and this region is used to shape the codebook boundary. We now show that by a suitable choice of $\mathcal{Q}, \mathcal{L}$ and L, the SVQ can also be used for the N-sphere shaping of the codebook. We shall restrict our attention to codebooks based on integer cubic lattices scaled by a positive real α and hence $\mathcal{Q} = \{-k\alpha, ..., -\alpha, 0, \alpha, ..., k\alpha\}$ for some suitably large k. If we define $\ell_i = (q_i/\alpha)^2$; $\forall\ |i| \leq k$, then the N-dimensional SVQ codebook consists of all points of the cubic lattice that lie inside and on an N-sphere of squared-radius $\alpha^2 L$. The encoding algorithm of the SVQ can be used to assign unique codewords to each vector in the codebook. In fact, any set of points in $\mathbb{R}^N$ that can be specified by some choice of $\mathcal{Q}, \mathcal{L}$ and L can be encoded using the SVQ encoding algorithm.

In the framework of signal transmission, the constellation is analogous to the quantizer codebook. As mentioned before, the N-dimensional constellation that minimizes the average power for a given number of points on a lattice is bounded by an N-sphere. In

the continuous approximation, this statement holds irrespective of the type of lattice. To achieve an appreciable shaping gain, the constellation dimension should be large implying that algorithmic indexing of constellation points is necessary. The conventional approach to the indexing problem is to shape the constellations using Voronoi regions of lattices because fast encoding/decoding of points in such constellations can be performed by an algorithm due to Conway and Sloane [11]. Our approach is to exploit the similarity between an optimally shaped constellation and the SVQ codebook for a Gaussian source and use the SVQ encoding/decoding algorithms for indexing the constellation points.

The optimally shaped N-dimensional cubic lattice based constellation can be specified as the codebook of an N-dimensional SVQ for $\mathcal{Q} = \{-k\alpha, ..., -\alpha, 0, \alpha, ..., k\alpha\}$, $\ell_i = (q_i/\alpha)^2$ and an appropriate L determined (using the algorithm given earlier in this section) for the required number of points on the constellation. From the continuous approximation, L is (approximately) the squared-radius of an N-sphere enclosing 2^{Nr} lattice-points and is linear in N. This makes the computational complexities of the SVQ encoding and decoding algorithms for these constellations linear in N and their storage complexities quadratic in N. These constellations can hence be implemented (with a reasonable complexity) for large N making it possible to realize almost all of the maximum shaping gain of 1.53 dB. If complexity does cause a problem, it can be reduced by dividing all the lengths in $\mathcal{L}$ by some $\rho > 1$ and then quantizing them to the nearest integers. This will also reduce the threshold L resulting in reduced complexity encoding and decoding algorithms. The shaping in this case will obviously be sub-optimal.

Fig. 1 gives the block diagram of the transmitter and receiver for an N-sphere SVQ-shaped cubic lattice based constellation. The transmitter takes Nr bits from the input stream and uses an SVQ decoder to convert these bits to a constellation point (N-vector) which is transmitted. At the receiver, the channel output is first quantized, using a bank of N scalar quantizers (or $N/2$ successive uses of a pair of quantizers), to the nearest point on the cubic lattice. This will give back the transmitted constellation point (assuming channel noise does not cause an error) which is converted to an Nr-bit binary stream using the SVQ encoder.

Although optimal shaping gains are realized by SVQ shaping, the discussion so far has been limited to cubic lattices which offer no coding gain. More densely packed lattices (or trellis codes) can result in significant coding gains. The SVQ shaping can be combined with

trellis coding to realize both shaping and coding gains. Indexing of constellation points in this case is performed using the encoding/decoding algorithms of the trellis-based finite-state SVQ [5],[13]. We postpone further discussion on this until Section IV.

III. Shaping and Constellation Expansion Ratio

While N-sphere shaping achieves the best shaping gains, it also results in large CER_2 and PAR_2. Table 1 gives the shaping gain γ_s, CER_2 and PAR_2 of N-spheres for various N (also see [8]). For large N, the probability of occurrence of the points in the constituent 2D constellation is not uniform but is close to a 2-dimensional Gaussian distribution even when the constellation points themselves are equally probable. Channel capacity arguments also show that the optimal distribution of points in the constituent 2D constellation is the 2-dimensional Gaussian distribution. Because of this the points of the constituent 2D constellation that occur most frequently are the ones that are close to the origin, hence the average energy (per two dimensions) of this constellation is small and this is the reason for its large shaping gain. On the other hand, the points on the periphery of the constituent 2D constellation are very improbable suggesting that even if these points are removed (resulting in a smaller CER_2) good shaping gains should still be possible. Bounds on the shaping gain for a given CER_2 and PAR_2 derived in [8] (see Fig. 2 and 3 of this paper) show that it is indeed possible to get large shaping gains at considerably smaller values of CER_2 and PAR_2 than those required for N-spheres. Soon it will be shown that these bounds can indeed be achieved (asymptotically in N) by SVQ shaping.

A subtle point worth observing is the following. The definition of the constituent 2D constellation given in [8] (and in Section I) is a little too restrictive. For a QAM modem based implementation, all that is really required is that the N dimensions be partitioned into a set of $N/2$ pairs of dimensions and the constituent constellation along any pair in this set, if they are all the same (if not, take the largest of such constituent constellations), can be taken as the constituent 2D constellation. This is less restrictive than requiring that the constituent 2D constellation be the same for all possible pairs of dimensions. Although it is not mentioned, the bounds on shaping gain derived in [8] are asymptotically achievable only for a constraint on the expansion ratio CER_2 of this less restrictive definition of the constituent 2D constellation. For the rest of this paper we use these new definitions of the constituent 2D constellation C_2 and the constellation expansion ratio CER_2.

To demonstrate that optimal SVQ shaping is possible for a given (constraint on) CER_2, we first consider the simpler problem of optimal SVQ shaping for a given CER_1, which is the constellation expansion ratio of the constituent 1D constellation C_1 (the constituent 1D constellation is defined as the largest of the constituent constellations along each dimension). This will be generalized to the 2D case.

3.1 Optimal shaping for a given CER_1

The problem is to determine the shaping region of a rate r (bits per dimension) N-dimensional cubic lattice based constellation, that maximizes the shaping gain for $1 \leq \mathrm{CER}_1 \leq \beta$. For this rate the baseline constellation is bounded by an N-cube of side $(2^r - 1)\alpha$ and has 2^r points in its constituent 1D constellation. The desired constellation has at most $\delta = \beta 2^r$ points in its constituent 1D constellation and is a subset of points inside an outer N-cube of side $\beta(2^r - 1)\alpha$. It is obvious that the best way to choose a given (small) number of points inside the outer N-cube while minimizing the average energy is to choose them inside an N-sphere (centered at the origin) of appropriate radius. As the number of points to be chosen increases, the radius of the N-sphere (that contains them) also increases until the sphere begins to intersect with the outer N-cube. When that happens, only the points that lie inside the intersection of the N-sphere interior and the outer N-cube interior must be included in the collection. The size of the N-sphere can be increased if necessary to accommodate a total of 2^{Nr} points in the required constellation. This procedure of choosing the constellation points ensures that the points closest to the origin (minimum energy) that satisfy the outer N-cube constraint are chosen first. Hence the resulting constellation has the smallest possible average power for the required number of points. The shaping region of this constellation will be the intersection of an N-sphere interior and an N-cube interior. It might happen that for a small N (and large β) the N-sphere is entirely contained inside the N-cube. The resulting constellation then is bounded by the N-sphere and the actual $\mathrm{CER}_1 \leq \beta$ (equality holds only if the diameter of the N-sphere is equal to the side of the N-cube).

The set of points in the optimally shaped constellation described above (assuming that δ is an odd integer) can be specified as the codebook of an N-dimensional SVQ with $\mathcal{Q} = \{-(\delta-1)\alpha/2,\ -(\delta-3)\alpha/2, ..., -\alpha, 0, \alpha, ..., (\delta-1)\alpha/2\}$; $\ell_i = (q_i/\alpha)^2$ and L such that the constellation has (at least) 2^{Nr} points. The transmitter/receiver block diagram for this

case is also described by Fig. 1. As in [8], it can be reasoned that the above constellations also give the best trade-off between the shaping gain γ_s and the PAR_1.

Table 2 gives the shaping gain γ_s for various values of CER_1 (and the corresponding PAR_1) for two different N. These values were obtained using Monte-Carlo simulations. For a given $\text{CER}_1 = \beta$ of the constituent 1D constellation, the CER_2 is upper-bounded by β^2 and the PAR_2 is upper-bounded by PAR_1 (for large N these quantities become equal to their bounds). Table 2 also gives these (upper-bound) values for the corresponding constituent 2D constellation. For these values of CER_2 and PAR_2 however, higher shaping gains are possible as shown next.

3.2 Optimal shaping for a given CER_2

The optimal shaping solution in this case is the generalization of the 1D solution described above. It is shown in [8] that the required optimally shaped N-dimensional constellation of 2^{Nr} points (under the $\text{CER}_2 \leq \beta$ constraint) should have a circular constituent 2D constellation $\mathcal{C}$ with $\beta 2^{2r}$ points. The required constellation is hence constrained to be a subset of points enclosed by $\mathcal{C}^{N/2}$, which is the $N/2$-fold cartesian product of $\mathcal{C}$ with itself. Proceeding as in the 1D case, we choose the points in the intersection of the interiors of $\mathcal{C}^{N/2}$ and an N-sphere of appropriate radius so that the constellation has 2^{Nr} points. This ensures that we pick the 2^{Nr} minimum energy points that satisfy the CER_2 constraint (lie inside $\mathcal{C}^{N/2}$). Hence the average power of this constellation is the minimum for the given size. The points of this constellation can also be described as the codebook of an SVQ but this is a little more involved than in the corresponding 1D case. The scalar quantization levels of the SVQ are replaced by the $\delta = \beta 2^{2r}$ points (2-tuples) $\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_\delta$ of the constituent 2D constellation, where $\mathbf{q}_i = (q_{1i}, q_{2i})$. Therefore $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_\delta\}$. The length ℓ_i of the 2-tuple $\mathbf{q}_i$ is the normalized squared-distance from the origin, i.e., $\ell_i = (1/\alpha^2)({q_{1i}}^2 + {q_{2i}}^2)$. The threshold L is once again chosen such that the $N/2$-dimensional SVQ codebook consists of (at least) 2^{Nr} points. The SVQ encoding/decoding algorithms will therefore index the constellation points. As shown in [8], this optimally shaped constellation (under the $\text{CER}_2 \leq \beta$ constraint) also represents the best trade-off between shaping gain and PAR_2.

Table 3 gives the optimal γ_s (obtained by Monte-Carlo simulations) for various values of CER_2 and the corresponding PAR_2. These results are also plotted in Fig. 2 and 3 along

with the asymptotic bounds (in N) and demonstrate that large shaping gains are indeed possible with a small CER_2.

IV. Compatibility With Trellis-Coding

We have so far discussed the optimal shaping of cubic lattice based N-dimensional constellations. These constellations have a large shaping gain but offer no coding gain. Since maximum achievable coding gains (up to 6 dB) are significantly larger than the maximum shaping gain of 1.53 dB, the present approach will be useful only if it allows constellations based on lattices or (trellis codes) denser than the cubic lattice. It was shown in [2] that trellis codes can be constructed from a redundant cubic lattice (that has a higher density of points than required) to achieve significant coding gains. The SVQ shaping scheme is indeed compatible with trellis coding and the two can be combined. This makes it possible to have SVQ-shaped trellis-coded modulation that has near optimal shaping gains and offers significant coding gains. This scheme is the dual of the trellis-based finite-state SVQ (TB-FS-SVQ) [5]. In the TB-FS-SVQ, the SVQ is used for shaping the codebook according to the probability distribution of the source, and trellis coding is used to realize 'coding' gain over a cubic lattice. In the case of quantization however, this coding gain is bounded by 1.53 dB and the shaping gain is usually larger.

The SVQ-shaped trellis-coded modulation scheme is now described. The description is quite general and any trellis that satisfies the two properties given below can be used. Other than these two properties, the scheme does not place a constraint on the trellis. Details of trellis design, the resulting coding gain and trellis-decoding (Viterbi algorithm) are not too important here and can be found in [10]. Some familiarity with conventional trellis-coded modulation is assumed.

The two properties we require the trellises to possess are:

(1) The trellis partitions the given constituent 2D constellation A_0 into two subsets B_0 and B_1 (the case for a k subset partition is similar) such that the subsets have the same number (density) of points. Further, it is required that each point in B_0 can be paired with a point in B_1 that has the same energy. This last requirement is not for the trellis but the constituent 2D constellation A_0. Let the set of pairs be denoted by $\mathcal{P} \equiv \{\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_n\}$, where $\mathbf{P}_i \equiv (\mathbf{p}_{1i}, \mathbf{p}_{2i})$; $i \in J_n$, with $\mathbf{p}_{1i} \in \text{B}_0$ and $\mathbf{p}_{2i} \in \text{B}_1$. Also, to every pair $\mathbf{P}_i$; $i \in J_n$, assign a length ℓ_i which is the (normalized to an integer)

squared-distance of point $\mathbf{p}_{1i}$ (or $\mathbf{p}_{2i}$) from the origin, and let $\mathcal{L} \equiv \{\ell_1, \ell_2, ..., \ell_n\}$. Fig. 4 shows an example of a 2D constellation, its partitions and the pairs.

(2) All outgoing transitions from each state of the trellis are either associated with all points in B_0 or in B_1 but not both. In other words, the allowed points of A_0 in any given trellis state either constitute B_0 or B_1. This implies that given the initial state s_0 of the trellis, a sequence of pairs $\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_{N/2}$, where $\mathbf{D}_i \in \mathcal{P}$, can be uniquely decoded using the trellis into a sequence of points $\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N/2}$, where $\mathbf{d}_i \in \mathrm{A}_0$.

Define an $N/2$-dimensional SVQ with (the quantization levels replaced by) the set of pairs $\mathcal{P}$, the set lengths $\mathcal{L}$, and a threshold L. The codebook $\mathcal{Z}$ of this SVQ contains all $N/2$-tuples of 'pairs' $(\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_{N/2}) \in \mathcal{P}^{N/2}$ which have a total length no greater than L. As usual, L is chosen such that the SVQ has (at least) 2^{Nr} of the $N/2$-tuples. Each of these $N/2$-tuples can be indexed by a unique Nr-bit binary number using the SVQ encoding/decoding algorithms. We call the codebook $\mathcal{Z}$ the primary codebook or the primary constellation, and the pair $N/2$-tuples the primary codevectors. As mentioned above, if the initial state of the trellis is known, each of the primary codevectors can be decoded into a unique sequence of $N/2$ points in A_0. The 2^{Nr} primary codevectors, for a given initial state $s_i \in S$ (where S is the set of K trellis states), hence correspond to a set $\mathcal{Z}_{s_i}$ of 2^{Nr} different $N/2$-tuples in ${\mathrm{A}_0}^{N/2}$. Of the total of $n^{N/2}$ $N/2$-tuples in ${\mathrm{A}_0}^{N/2}$ that are consistent with the trellis, $\mathcal{Z}_{s_i}$ consists of only those 2^{Nr} that have the smallest energy. The set $\mathcal{Z}_{s_i}$ is called the secondary codebook or the secondary constellation associated with the state s_i. Its constituent $N/2$-tuples (in ${\mathrm{A}_0}^{N/2}$) are the secondary codevectors. For a K-state trellis there are K secondary constellations. Which one of these K secondary constellations will be used for the next transmission (of Nr bits) is determined by the (final) trellis state at the end of the previous transmission.

Fig. 5 gives the block diagrams of the transmitter/receiver for the SVQ-shaped trellis-coded modulation scheme. First consider the transmitter. Assume that at the beginning the trellis state is known, say s_0. The first block of Nr bits to be transmitted is mapped, using the SVQ decoder, into a codevector in the primary codebook $\mathcal{Z}$. The trellis (and the initial state s_0) is used to convert this primary codevector into a secondary constellation point $\mathbf{z}$ in $\mathcal{Z}_{s_0}$. The final state of the trellis at the end of this conversion is its initial state for the next transmission. The secondary constellation point $\mathbf{z}$ is transmitted as a sequence of $N/2$ points in A_0.

The receiver receives a noise affected sequence of $N/2$ points in A_0. Just as in the receiver of the conventional trellis-coded modulation scheme, a Viterbi (trellis) decoder is first used to recover the transmitted sequence of points in A_0. Assuming an error free recovery, the $N/2$-tuple (secondary constellation point) in ${A_0}^{N/2}$ is trivially converted to a primary codevector in the SVQ codebook $\mathcal{Z}$, which is in turn decoded into an Nr-bit binary stream using the SVQ encoding algorithm.

The scheme described above is very general and can be split into the shaping part and the trellis coding/decoding part. The implementation of the shaping part, consisting of the SVQ decoder (in the transmitter) and the SVQ encoder (in the receiver), is independent of the details of the trellis.

Although the SVQ-shaped trellis-coded modulator is a dual of the trellis-based finite-state SVQ, it is much simpler to implement than the TB-FS-SVQ as no codebook search is required. This is because the receiver only performs Viterbi decoding (no codebook search) and the reconstructed sequence is assumed to be error free. However, in the above scheme there is one type of errors that are easily detectable. It corresponds to the case when the reconstructed 'primary codevector' has a total length greater than the threshold L. This is called a constellation overload error. Clearly, this can only be the result of a channel error. This kind of errors are most likely to result from the outermost points in the shaping region of the primary constellation. A simple way to deal with them is to let them cause bit-errors. Often, the knowledge that an error has occurred is useful even if the error cannot be corrected. A better but computationally more expensive way to deal with overload errors is to use the codebook search algorithm of the TB-FS-SVQ in place of the simpler Viterbi trellis decoder. The codebook search will map the received data onto the closest 'allowed' (within the shaping region) secondary constellation point rather than the closest point on the trellis code. This does not guarantee error correction but reduces the probability of such errors. This gain is the result of the fact that the outermost constellation points in the shaping region have fewer neighbors than the inside points. Considering that for a large N, most of the constellation points are close to the boundary of the shaping region, codebook search might cause a small but significant reduction in the overall error probability. Any such conclusion however, can only be drawn after further investigation. Note that codebook search only increases the complexity of the receiver and not the transmitter, and is required only when such an error is detected. For

Voronoi constellations such errors are not easy to detect and hence no such error correction possibility exists.

V. Characteristics of SVQ Shaping

Various desirable constellation characteristics, specially for implementation over QAM modems, are discussed in [8]. It would be useful to consider the SVQ-shaped constellations in the light of some of these characteristics. It has already been shown that SVQ shaping results in optimal shaping gains for a given CER_2 or PAR_2. Its implementation complexity is very reasonable and it is compatible with trellis coding. Now we briefly consider a few of their other characteristics.

Phase symmetry: For QAM implementations, the constituent 2D constellation should be invariant to as many phase rotations as possible. This enables the carrier phase tracker at the receiver to converge quickly. Differential encoding is used to ensure that the resulting phase ambiguity (due to rotational invariance) does not lead to errors. Since optimal SVQ-shaped constellations are bounded by N-spheres which are invariant under any phase rotation, the phase symmetry of their constituent 2D constellations is determined only by the lattice on which they are based. A cubic lattice based constellation is invariant to $\pi/2$ phase rotations. Even for optimally shaped constellations under the $\mathrm{CER}_2 \leq \beta$ constraint, the constituent 2D constellation is bounded by a circle and the phase symmetry is dictated only by the underlying lattice.

Scalability: Scaling of SVQ-shaped constellations is trivially possible by scaling the threshold L. Scaling the threshold by σ, assuming that the continuous approximation holds, scales the constellation size by $\sigma^{N/2}$.

Opportunistic secondary channels: For any value of the threshold L, it is usually not possible to have exactly 2^{Nr} points in the constellation. In this case L is taken as the minimum value of the threshold for which there are at least 2^{Nr} constellation points. Since there are usually more than the required number of points available on the constellation, one approach is to keep only the ones that are labeled from 0 to $2^{Nr}-1$ and not use the rest. The SVQ encoding algorithm of Section II ensures that the points not used are those with the maximum energy (boundary points). Another approach is to associate some labels to more than one points on the boundary and choose any one of these points when such a label is to be transmitted. This allows the possibility of sending some additional infor-

mation over the channel without increasing the average power. This secondary channel which is probabilistic in nature is called an opportunistic secondary channel and can be used to transmit low rate control data over the channel. The SVQ encoding algorithm as described in this paper labels the boundary constellation points with consecutive numbers (codewords) thus making it easy to identify these points and use the opportunistic secondary channel.

VI. Conclusions

The concept of SVQ shaping was introduced in this paper. Optimal (N-sphere) SVQ shaping results in higher shaping gains than those of Voronoi constellations based on known N-dimensional lattices. For a given CER_2 (or PAR_2) SVQ shaping results in optimal shaping. This is useful for implementation over QAM modems as significant shaping gains can be achieved even for a small constellation expansion. In contrast, Voronoi constellations result in a much larger constellation expansion and peak power for the same shaping gain. The SVQ-shaped constellations have a very reasonable implementation complexity. They allow for the possibility of opportunistic secondary channels, and there constituent 2D constellations are invariant to $\pi/2$ phase rotations (assuming cubic lattice based constellations). Compatibility of SVQ shaping with trellis coding was also demonstrated, resulting in the SVQ-shaped trellis-coded modulation scheme which can realize both shaping and coding gains.

References

1. M. V. Eyuboglu and G. D. Forney, Jr., "Lattice and Trellis Quantization with Lattice- and Trellis-Bounded Codebooks – Part I: High-Rate Theory for Memoryless Sources," submitted to *IEEE Trans. Inform. Theory*, Dec 1990.
2. G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals," *IEEE Trans. Inform. Theory*, Vol. IT-28, pp. 55-67, Jan. 1982.
3. G. D. Forney, Jr., "Coset Codes – Part I: Introduction and Geometrical Classification," *IEEE Trans. Inform. Theory*, Vol. 34, No. 5, pp. 1123-1151, Sept. 1988.
4. M. W. Marcellin and T. R. Fischer, "Trellis Coded Quantization of Memoryless and Gauss-Markov Sources," *IEEE Trans. Commun.*, Vol. 38, No. 1, pp. 82-93, Jan. 1990.

5. R. Laroia and N. Farvardin, "Trellis-Based Finite-State Structured Vector Quantization of Memoryless Sources," in preparation for submission to *IEEE Trans. Inform. Theory.*
6. M. V. Eyuboglu and A. S. Balamesh, "Lattice and Trellis Quantization with Lattice- and Trellis-Bounded Codebooks – Construction and Implementation for Memoryless Sources," submitted to *IEEE Trans. Inform, Theory*, Oct. 1991.
7. R. Laroia and N. Farvardin, "A Structured Fixed-Rate Vector Quantizer Derived from a Variable-Length Scalar Quantizer," submitted to *IEEE Trans. Inform. Theory*, Aug. 1991.
8. G. D. Forney, Jr. and L. F. Wei, "Multidimensional Constellations – Part I: Introduction, Figures of Merit, and Generalized Cross Constellations," *IEEE J. Selcct. Area Commun.*, Vol. 7, No. 6, pp. 877-892, Aug. 1989.
9. G. D. Forney, Jr., "Multidimensional Constellations – Part II: Voronoi Constellations," *IEEE J. Select. Area Commun.*, Vol. 7, No. 6, pp. 941-958, Aug. 1989.
10. G. D. Forney, Jr. *et al.*, "Efficient Modulation for Band Limited Channels," *IEEE J. Select. Area Commun.*, Vol. SAC-2, pp. 632-647, Sept. 1984.
11. J. H. Conway and N. J. A. Sloane, "A Fast Encoding Method for Lattice Codes and Quantizers," *IEEE Trans. Inform. Theory,* Vol. IT-29, pp. 106-109, 1985.
12. R. Laroia and N. Farvardin, "Extension of the Fixed-Rate Structured Vector Quantizer to Vector Sources," in preparation for submission to *IEEE Trans. Inform. Theory.*
13. R. Laroia, *Design and Analysis of a Structured Fixed-Rate Vector Quantizer Derived from Variable-length Scalar Quantizers,* Ph.D. Dissertation, Electrical Engineering Department and Systems Research Center, Univ. of Maryland, College Park, in preparation.

N	γ_s (dB)	CER_2	PAR_2
2	0.2	1.0	2.0
4	0.46	1.41	3.0
8	0.73	2.21	5.0
16	0.98	3.76	9.0
32	1.17	6.80	17.0
64	1.31	12.79	33.0

Table 1: Shaping gains γ_s, CER_2 and PAR_2 of N-spheres.

γ_s (dB)	CER_1	PAR_1	CER_2	PAR_2
$N = 32$				
0.00	1.00	3.00	1.00	3.00
0.69	1.06	3.95	1.12	3.95
0.79	1.08	4.22	1.17	4.22
0.86	1.11	4.48	1.23	4.48
0.97	1.16	5.02	1.35	5.02
1.08	1.26	6.13	1.59	6.13
1.15	1.50	8.81	2.25	8.81
1.17	2.01	16.81	4.04	16.81
1.17	2.47	24.10	6.10	24.10
$N = 64$				
0.00	1.00	3.00	1.00	3.00
1.04	1.12	4.83	1.26	4.83
1.09	1.15	5.09	1.32	5.09
1.17	1.20	5.67	1.44	5.67
1.24	1.29	6.67	1.66	6.67
1.27	1.39	7.76	1.93	7.76
1.29	1.50	9.12	2.25	9.12
1.30	1.65	11.08	2.72	11.08
1.30	1.74	12.24	3.03	12.24
1.30	2.01	16.35	4.04	16.35
1.30	2.47	24.70	6.10	24.70

Table 2: The shaping gain γ_s for a given CER_1. The corresponding values of PAR_1, CER_2 and PAR_2 are also given.

γ_s (dB)	CER_2	PAR_2
$N = 32$		
0.00	1.00	2.00
0.54	1.05	2.27
0.80	1.14	2.62
0.94	1.24	2.95
1.02	1.36	3.28
1.09	1.54	3.79
1.13	1.75	4.34
1.15	1.99	4.96
1.17	2.47	6.17
1.17	3.06	7.67
$N = 64$		
0.00	1.00	2.00
0.64	1.05	2.33
0.81	1.10	2.53
0.92	1.14	2.71
1.06	1.24	3.04
1.22	1.51	3.82
1.29	2.01	5.16
1.30	2.53	6.52
1.30	2.99	7.72

Table 3: The shaping gain γ_s for a given CER_2. The corresponding value of PAR_2 is also given.

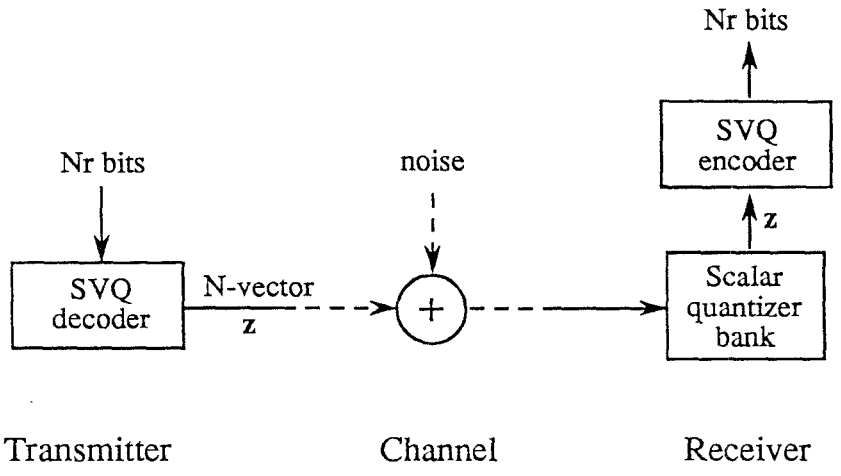


Fig. 1: Transmitter/receiver for the SVQ-shaped cubic lattice based constellations.

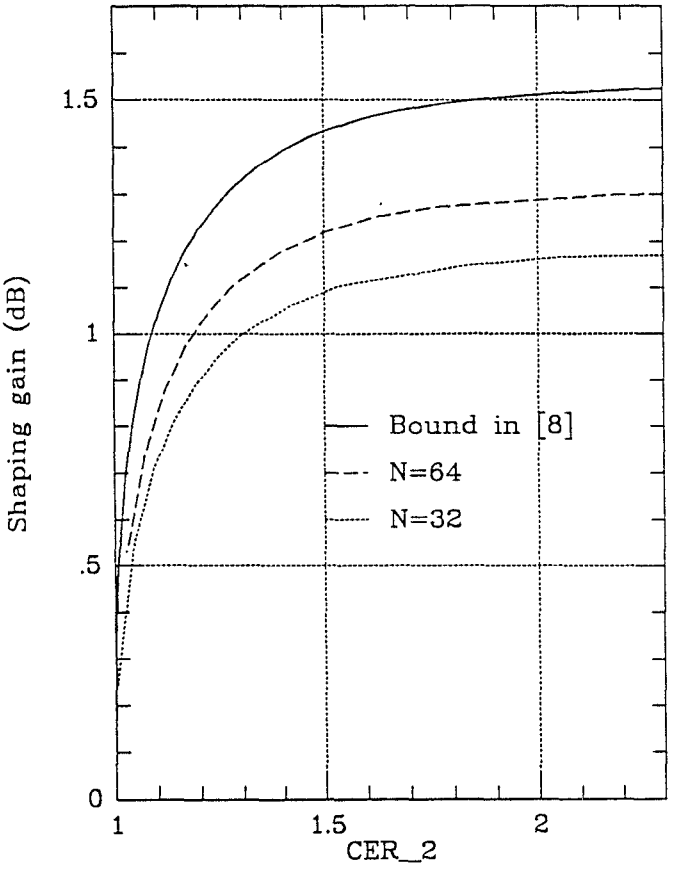


Fig. 2: Shaping gain γ_s as a function of CER_2 for SVQ-shaping.

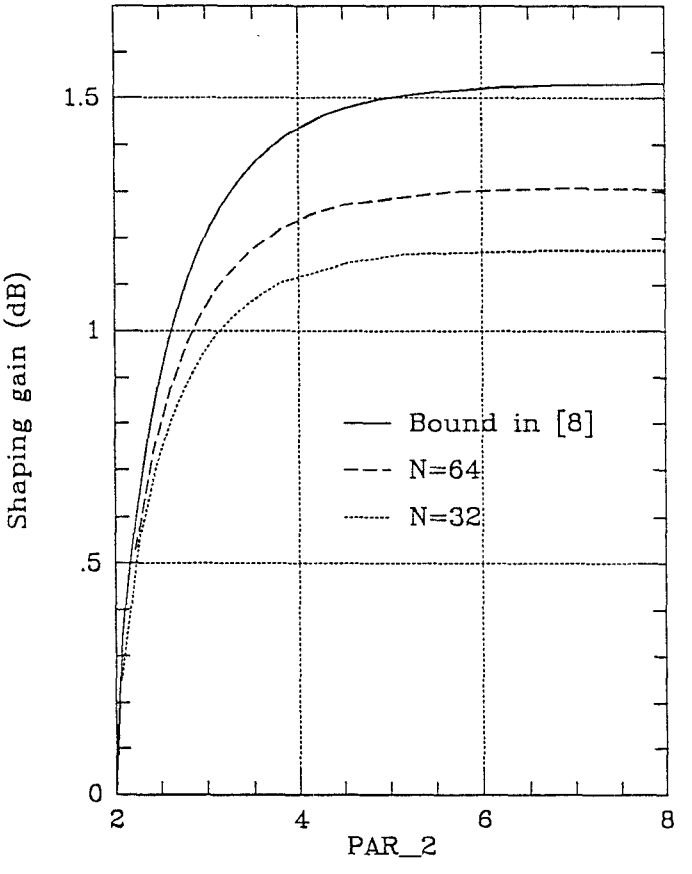


Fig. 3: Shaping gain γ_s as a function of PAR_2 for SVQ-shaping.

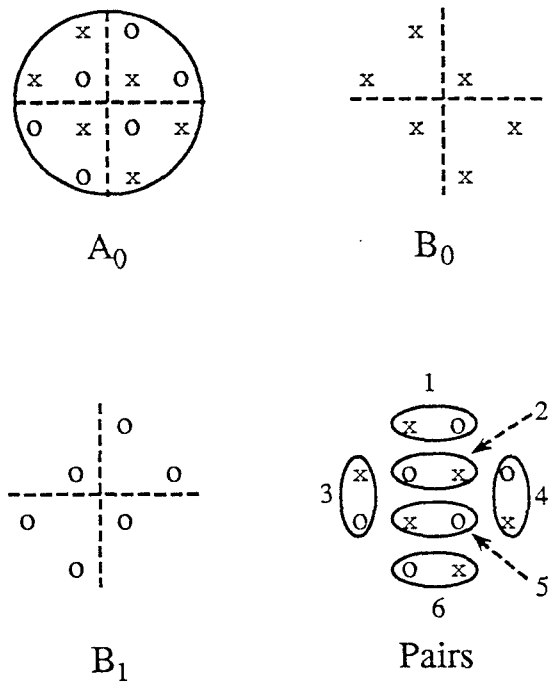


Fig. 4: A 2D constellation A_0, partitions B_0 and B_1, and six pairs of points.

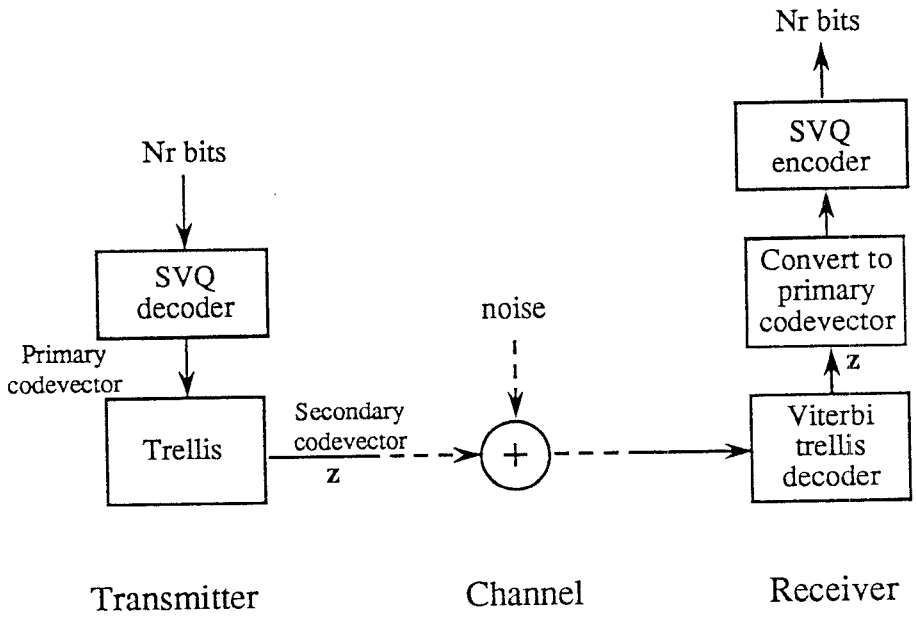


Fig. 5: Transmitter/receiver for the SVQ-shaped trellis-coded modulation scheme.

List of Tables

List of Figures