

# On Concentric Spherical Codes and Permutation Codes With Multiple Initial Codewords

Ha Q. Nguyen, Vivek K Goyal, and Lav R. Varshney  
 Research Laboratory of Electronics, Massachusetts Institute of Technology

**Abstract**—Permutation codes are a class of structured vector quantizers with a computationally-simple encoding procedure. In this paper, we provide an extension that preserves the computational simplicity but yields improved operational rate-distortion performance. The new class of vector quantizers has a codebook comprising several permutation codes as subcodes. Methods for designing good code parameters are given. One method depends on optimizing the rate allocation in a shape-gain vector quantizer with gain-dependent wrapped spherical shape codebook.

## I. INTRODUCTION

Because of the “sphere hardening” effect, the performance in coding a memoryless Gaussian source can approach the rate-distortion bound even with the added constraint of placing all codewords on a single sphere. Such spherical source coding techniques, however, are not commonly used in practice. Furthermore, the elegant but uncommon technique of permutation source coding—which places all codewords on a single sphere—has asymptotic performance only as good as entropy-constrained *scalar* quantization.

A heuristic explanation for the disappointing performance of permutation codes is as follows. For large block lengths, permutation codes suffer because there are too many permutations ( $n^{-1} \log_2 n!$  grows) and the vanishing fraction that are chosen to meet a rate constraint do not form a good code. For moderate block lengths, sphere hardening does not occur fast enough for a codebook on a sphere to perform well. We improve permutation codes by allowing multiple initial codewords, so codewords are placed on concentric spheres.

We first review in Sec. II the convergence of spherical source codes to the rate-distortion bound and the basic formulation of permutation coding. Sec. III introduces permutation codes with multiple initial codewords and discusses the difficulty of their optimization. One simplification that reduces the design complexity—the use of a single common integer partition for all initial codewords—is discussed in Sec. IV. The use of a common integer partition obviates the issue of allocating rate amongst concentric spheres of codewords. Sec. V returns to the general case, with integer partitions that are not necessarily equal. We develop fixed- and variable-rate generalizations of the wrapped spherical gain-shape vector quantization of Hamkins and Zeger [1] for the purpose of guiding the rate allocation problem. These results may be of independent interest. Concluding comments appear in Sec. VI.

Work supported in part by a Vietnam Education Foundation Fellowship and NSF Grant CCF-0729069.

## II. BASIC FORMULATION AND BACKGROUND

Let  $X \in \mathbb{R}^n$  be a random vector with independent  $\mathcal{N}(0, \sigma^2)$  components. We wish to approximate  $X$  with a *codeword*  $\hat{X}$  drawn from a finite *codebook*  $\mathcal{C}$ . We want small per-component mean-squared error (MSE) distortion  $D = n^{-1} E[\|X - \hat{X}\|^2]$  when the approximation  $\hat{X}$  is represented with  $nR$  bits. In the absence of entropy coding, this means the codebook has size  $2^{nR}$ . For a given codebook, the distortion is minimized when  $\hat{X}$  is chosen to be the codeword closest to  $X$ .

### A. Spherical Codes

In a *spherical (source) code*, all codewords lie on a single sphere in  $\mathbb{R}^n$ . Nearest-neighbor encoding with such a codebook partitions  $\mathbb{R}^n$  into  $2^{nR}$  cells that are infinite polygonal cones with apexes at the origin. In other words, the representations of  $X$  and  $\alpha X$  are the same for any scalar  $\alpha > 0$ . Thus a spherical code essentially ignores  $\|X\|$ , placing all codewords at radius

$$E[\|X\|] = \frac{\sqrt{2\pi\sigma^2}}{\beta(n/2, 1/2)} \approx \sigma\sqrt{n-1/2},$$

while representing  $X/\|X\|$  with  $nR$  bits.

Sakrison [2] first analyzed the performance of spherical codes for memoryless Gaussian sources. Following [1], [2], the distortion can be decomposed as

$$D = \frac{1}{n} E \left[ \left\| (E[\|X\|]/\|X\|)X - \hat{X} \right\|^2 \right] + \frac{1}{n} \text{var}(\|X\|).$$

The first term is the distortion between the projection of  $X$  to the code sphere and its representation on the sphere, and the second term is the distortion incurred from the projection. The second term vanishes as  $n$  increases even though no bits are spent to convey the length of  $X$ . Placing codewords uniformly at random on the sphere controls the first term sufficiently for achieving the rate-distortion bound as  $n \rightarrow \infty$ .

### B. Permutation Codes

In a *permutation (source) code*, all the codewords are related by permutation and thus have equal length. So permutation codes are spherical codes. Permutation codes were introduced by Dunn [3] and subsequently developed by Berger et al. [4]–[6]. For brevity, we limit our discussion to Variant I permutation codes; Variant II could be generalized similarly.

Let  $\mu_1 > \mu_2 > \dots > \mu_K$  be  $K$  ordered real numbers, and let  $n_1, n_2, \dots, n_K$  be positive integers with sum equal to  $n$  (an

integer partition of  $n$ ). The initial codeword of the codebook  $\mathcal{C}$  has the form

$$\hat{x}_{\text{init}} = (\underbrace{\mu_1, \dots, \mu_1}_{n_1}, \underbrace{\mu_2, \dots, \mu_2}_{n_2}, \dots, \underbrace{\mu_K, \dots, \mu_K}_{n_K}), \quad (1)$$

where each  $\mu_i$  appears  $n_i$  times. The codebook is the set of all distinct permutations of  $\hat{x}_{\text{init}}$ . The number of codewords in  $\mathcal{C}$  is thus given by the multinomial coefficient

$$M = \frac{n!}{n_1! n_2! \dots n_K!}. \quad (2)$$

The permutation structure of the codebook enables low-complexity nearest-neighbor encoding [4]: map  $X$  to the codeword  $\hat{X}$  whose components have the same order as  $X$ ; in other words, replace the  $n_1$  largest components of  $X$  with  $\mu_1$ , the  $n_2$  next-largest components of  $X$  with  $\mu_2$ , and so on. The complexity of sorting is  $O(n \log n)$  operations, so the encoding complexity is much lower than with an unstructured source code and only  $O(\log n)$  times higher than scalar quantization.

For i.i.d. sources, each codeword is chosen with equal probability. Consequently, there is no need for entropy coding and the per-letter rate is simply  $R = n^{-1} \log M$ . For a given initial codeword  $\hat{x}_{\text{init}}$ , the per-letter distortion is

$$D = n^{-1} E \left[ \sum_{i=1}^K \sum_{\ell \in I_i} (\xi_\ell - \mu_i)^2 \right], \quad (3)$$

where  $\xi_1 \geq \xi_2 \geq \dots \geq \xi_n$  are the order statistics [7] of random vector  $X$  and  $I_i$ s are the groups of indices generated by the integer partition, i.e.,

$$I_i = \left\{ \left( \sum_{m=1}^{i-1} n_m \right) + 1, \dots, \left( \sum_{m=1}^i n_m \right) \right\}.$$

Thus, given  $(n_1, n_2, \dots, n_K)$ , the optimal initial codeword can be determined easily from the means of the order statistics. Optimization of the integer partition is not easy. The analysis of [5] shows that when  $n$  is large, the optimal integer partition gives performance equal to entropy-constrained scalar quantization (ECSQ) of  $X$ . Performance does not strictly improve with increasing  $n$ ; permutation codes outperform ECSQ for certain combinations of block size and rate [8].

### III. MULTIPLE INITIAL CODEWORDS

We now generalize permutation codes by allowing multiple initial codewords. The resulting codebook is contained in a set of concentric spheres.

#### A. Basic Construction

Let  $J$  be a positive integer. Our generalization is the union of  $J$  permutation codes. Each notation from Sec. II-B is extended with a superscript or subscript  $j \in \{1, 2, \dots, J\}$  that indexes the constituent permutation codes. Thus,  $\mathcal{C}_j$  is the subcodebook of full codebook  $\mathcal{C} = \cup_{j=1}^J \mathcal{C}_j$  consisting of all  $M_j$  distinct permutations of initial vector

$$\hat{x}_{\text{init}}^j = \left( \mu_1^j, \dots, \mu_1^j, \dots, \mu_{K_j}^j, \dots, \mu_{K_j}^j \right), \quad (4)$$

where each  $\mu_i^j$  appears  $n_i^j$  times,  $\mu_1^j > \mu_2^j > \dots > \mu_{K_j}^j$ , and  $\sum_{i=1}^{K_j} n_i^j = n$ . Also,  $\{I_i^j\}_{i=1}^{K_j}$  are sets of indices generated by the  $j$ th integer partition.

Nearest-neighbor encoding of  $X$  with codebook  $\mathcal{C}$  can be accomplished with the following procedure:

- 1) Find the permutation that puts  $X$  in descending order. This single sorting operation finds  $\hat{X}_j \in \mathcal{C}_j$  closest to  $X$  for each  $j$ .

- 2) Encode  $X$  with the nearest codeword amongst  $\{\hat{X}_j\}_{j=1}^J$ .

The first step of the algorithm requires  $O(n \log n) + O(Jn)$  operations; the second step requires  $O(Jn)$  operations. The total complexity of encoding is therefore  $O(n \log n)$ , provided that we keep  $J = O(\log n)$ . In fact, in this rough accounting, the encoding with  $J = O(\log n)$  is as cheap as the encoding for ordinary permutation codes.

For i.i.d. sources, codewords within a subcodebook are approximately equally likely to be chosen, but codewords in different subcodebooks may have very different probabilities. Using entropy coding yields

$$R \approx \frac{1}{n} \left[ H(\{p_j\}_{j=1}^J) + \sum_{j=1}^J p_j \log M_j \right], \quad (5)$$

where  $p_j$  is the probability of choosing subcodebook  $\mathcal{C}_j$ . We omit a parallel development for the case without entropy coding due to space constraints.

The per-letter distortion is now given by

$$D = \frac{1}{n} E \left[ \min_{1 \leq j \leq J} \sum_{i=1}^{K_j} \sum_{\ell \in I_i^j} (\xi_\ell - \mu_i^j)^2 \right], \quad (6)$$

in analogy to (3).

Vector permutation codes are another generalization of permutation codes with improved performance [9]. The encoding procedure, however, requires solving the assignment problem in combinatorial optimization [10] and has complexity  $O(n^2 \sqrt{n} \log n)$ .

#### B. Optimization

In general, finding the best ordinary permutation code requires an exhaustive search over all integer partitions of  $n$ . (Assuming a precomputation of all the order statistic means  $\{E[\xi_\ell]\}_{\ell=1}^n$ , the computation of the distortion for a given integer partition through (3) is simple [4].) The search space can be reduced for certain distributions of  $X$  using [4, Thm. 3], but seeking the optimal code still quickly becomes intractable as  $n$  increases.

Our generalization makes the design problem considerably more difficult. Not only do we need  $J$  integer partitions, but the distortion for a given integer partition is not as easy to compute. Because of the minimization over  $j$  in (6), we lack a simple expression for the distortion in terms of the integer partition and the order statistic means. The relevant means are of conditional order statistics, conditioned on which subcodebook is selected; this depends on all  $J$  integer partitions.

In the remainder of the paper, we consider two ways to reduce the design complexity. In Sec. IV, we fix all subcodebooks to have a common integer partition. Along with reducing the design space, this restriction induces a structure in the full codebook that enables the joint design of  $\{\mu_i^j\}_{j=1}^J$  for any  $i$ . In Sec. V, we take a brief detour into the optimal

rate allocations in a wrapped spherical shape–gain vector quantizer with gain-dependent shape codebook. We use these rate allocations to pick the sizes of subcodebooks  $\{\mathcal{C}_j\}_{j=1}^J$ .

The simplifications presented here still leave high design complexity for large  $n$ . Thus, some simulations use complexity-reducing heuristics including our conjecture that an analogue to [4, Thm. 3] holds. Since our designs are not provably optimal, the improvements from allowing multiple initial codewords could be somewhat larger than we demonstrate.

#### IV. DESIGN WITH COMMON INTEGER PARTITION

In this section, let us assume that the  $J$  integer partitions are equal, i.e., the  $n_i^j$ s have no dependence on  $j$ . The sizes of the subcodebooks are also equal, and dropping unnecessary sub- and superscripts we write the common integer partition as  $\{n_i\}_{i=1}^K$  and the size of a single subcodebook as  $M$ .

The Voronoi regions of the code now have a special geometric structure. Recall that any spherical code partitions  $\mathbb{R}^n$  into infinite polygonal cones. Having a common integer partition implies that each subcodebook induces the same conic Voronoi structure on  $\mathbb{R}^n$ . The full code divides each of the  $M$  cones into  $J$  Voronoi regions. The following theorem precisely maps this to a vector quantization problem.

*Theorem 1:* For common integer partition  $\{n_1, n_2, \dots, n_K\}$ , the initial codewords  $\{(\mu_1^j, \dots, \mu_{n_1}^j, \dots, \mu_{n_2}^j, \dots, \mu_{n_K}^j)\}_{j=1}^J$  are optimal if and only if  $\{\mu^1, \dots, \mu^J\}$  are  $J$  optimal representation points of the vector quantization of  $\xi \in \mathbb{R}^K$ , where

$$\mu^j = \left( \sqrt{n_1} \mu_1^j, \sqrt{n_2} \mu_2^j, \dots, \sqrt{n_K} \mu_K^j \right), \quad 1 \leq j \leq J,$$

$$\bar{\xi} = \left( \frac{1}{\sqrt{n_1}} \sum_{\ell \in I_1} \xi_\ell, \frac{1}{\sqrt{n_2}} \sum_{\ell \in I_2} \xi_\ell, \dots, \frac{1}{\sqrt{n_K}} \sum_{\ell \in I_K} \xi_\ell \right).$$

*Proof:* Rewrite the distortion

$$\begin{aligned} nD &= E \left[ \min_{1 \leq j \leq J} \sum_{i=1}^K \sum_{\ell \in I_i} (\xi_\ell - \mu_i^j)^2 \right] \\ &= E \left[ \min_{1 \leq j \leq J} \sum_{i=1}^K \left( \frac{1}{\sqrt{n_i}} \sum_{\ell \in I_i} \xi_\ell - \sqrt{n_i} \mu_i^j \right)^2 \right] \\ &\quad + E \left[ \sum_{i=1}^K \sum_{\ell \in I_i} (\xi_\ell)^2 \right] - E \left[ \sum_{i=1}^K \left( \frac{1}{\sqrt{n_i}} \sum_{\ell \in I_i} \xi_\ell \right)^2 \right] \\ &= E \left[ \min_{1 \leq j \leq J} \|\bar{\xi} - \mu^j\|^2 \right] + E [\|X\|^2] \\ &\quad - E \left[ \sum_{i=1}^K \left( \frac{1}{\sqrt{n_i}} \sum_{\ell \in I_i} \xi_\ell \right)^2 \right]. \end{aligned} \quad (7)$$

Since the second and third terms of (7) do not depend on  $\{\hat{x}_{\text{init}}^j\}_{j=1}^J$ , minimizing  $D$  is equivalent to minimizing the first term of (7). By definition of a  $K$ -dimensional VQ, that term is minimized if and only if  $\{\mu^1, \dots, \mu^J\}$  are optimal

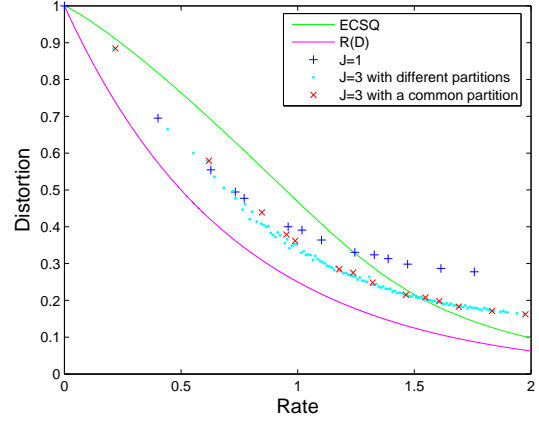


Fig. 1. Operational rate–distortion performance for i.i.d.  $\mathcal{N}(0, 1)$  source with block length  $n = 7$ . Ordinary permutation codes ( $J = 1$ ) are compared with generalized permutation codes with  $J = 3$ . Codes with common integer partitions are designed according to Thm. 1. Codes with different integer partitions are designed with heuristic selection of integer partitions and a Lloyd algorithm. For clarity, amongst approximately-equal rates, only operational points with the lowest distortion are plotted.

representation points of the  $J$ -point VQ of random vector  $\bar{\xi}$ , completing the proof. ■

For any fixed integer partition, it is straightforward to implement the  $J$ -point VQ design inspired by Thm. 1. Fig. 1 compares the performance of an ordinary permutation code ( $J = 1$ ) with permutation codes with  $J = 3$  initial vectors.

#### V. DESIGN WITH DIFFERENT INTEGER PARTITIONS

Suppose now that the integer partitions of subcodebooks can be different. The Voronoi partitioning of  $\mathbb{R}^n$  is much more complicated, lacking the separability discussed in the previous section.<sup>1</sup> Furthermore, the apparent design complexity for the partitions is increased greatly to equal the number of integer partitions raised to the  $J$ th power.

In this section we first outline an algorithm for local optimization of initial vectors with all the integer partitions fixed. Then we address a portion of the integer partition design problem which is the sizing of the subcodebooks. For this, we extend the high-resolution analysis of [1].

##### A. Local Optimization of Initial Vectors

Given  $J$  initial codewords  $\{\hat{x}_{\text{init}}^j\}_{j=1}^J$ , for each  $1 \leq j \leq J$ , let  $R_j \subset \mathbb{R}^n$  denote the quantization region corresponding to codeword  $\hat{x}_{\text{init}}^j$ , and let  $E_j[\cdot]$  denote the expectation conditioned on  $X \in R_j$ . By extension of an argument in [4], the distortion conditioned on  $X \in R_j$  is minimized with

$$\mu_i^j = \frac{1}{n_i^j} \sum_{\ell \in I_i^j} E_j[\xi_\ell], \quad 1 \leq i \leq K_j. \quad (8)$$

Denote the resulting distortion  $D_j$ . Since the total distortion is determined by

$$D = \sum_{j=1}^J \Pr(X \in R_j) D_j,$$

<sup>1</sup>For a related two-dimensional visualization, compare [11, Fig. 3] against [11, Figs. 7–13].

it is minimized if  $D_j$  is minimized for all  $1 \leq j \leq J$ . From the above analysis, a Lloyd algorithm can be developed to design initial codewords as follows:

- 1) Choose an arbitrary initial set of  $J$  representation vectors  $\hat{x}_{\text{init}}^1, \hat{x}_{\text{init}}^2, \dots, \hat{x}_{\text{init}}^J$ .
- 2) For each  $j$ , determine the corresponding quantization region  $R_j$ .
- 3) For each  $j$ ,  $\hat{x}_{\text{init}}^j$  is set to the new value given by (8).
- 4) Repeat steps 2 and 3 until further improvement in MSE is negligible.

This Lloyd algorithm was used to produce the operating points shown in Fig. 1 for generalized permutation codes with different integer partitions. We can see through the figure that common integer partitions can produce almost the same distortion as possibly-different integer partitions for the same rate. However, allowing the integer partitions to be different yields many more rates.

### B. Wrapped Spherical Shape-Gain Vector Quantization

Hamkins and Zeger [1] introduced a type of spherical code for  $\mathbb{R}^n$  where a lattice in  $\mathbb{R}^{n-1}$  is “wrapped” around the code sphere. They applied the wrapped spherical code (WSC) to the shape component in a shape-gain vector quantizer.

We generalize this construction to allow the size of the shape codebook to depend on the gain. Along this line of thinking, in [12, pp. 102–104] Hamkins provided an algorithm to optimize the number of codewords on each sphere. However, neither analytic nor experimental improvement was demonstrated. In contrast, our approach based on high-resolution optimization gives an explicit expression for the improvement in SNR. While our results may be of independent interest, our present purpose is to guide the selection of  $\{M_j\}_{j=1}^J$  in generalized permutation codes.

A *shape-gain* vector quantizer (VQ) decomposes a source vector  $X$  into a *gain*  $g = \|X\|$  and a *shape*  $S = X/g$ , which are quantized to  $\hat{g}$  and  $\hat{S}$ , respectively, and the approximation is  $\hat{X} = \hat{g} \cdot \hat{S}$ . We optimize here a wrapped spherical VQ with gain-dependent shape codebook. The gain codebook,  $\{\hat{g}_1, \hat{g}_2, \dots, \hat{g}_J\}$ , is optimized for the gain pdf, e.g., using the Lloyd algorithm. For each gain codeword  $\hat{g}_j$ , a shape subcodebook is generated by wrapping the sphere packing  $\Lambda \subset \mathbb{R}^{n-1}$  on to  $\Omega_n$ , the unit sphere in  $\mathbb{R}^n$ . The same  $\Lambda$  is used for each  $j$ , but the density (or scaling) of the packing may vary with  $j$ . Thus the normalized second moment  $G(\Lambda)$  applies for each  $j$  while minimum distance  $d_\Lambda^j$  depends on the quantized gain  $\hat{g}_j$ . We denote such sphere packing as  $(\Lambda, d_\Lambda^j)$ .

The per-letter MSE distortion will be

$$\begin{aligned} D &= \frac{1}{n} E \left[ \|X - \hat{g} \hat{S}\|^2 \right] \\ &= \frac{1}{n} E \left[ \|X - \hat{g} S\|^2 \right] + \frac{1}{n} E \left[ \|\hat{g} S - \hat{g} \hat{S}\|^2 \right] \\ &\equiv D_g + D_s, \end{aligned}$$

where the omitted cross term is zero due to the independence

of  $g$  and  $\hat{g}$  from  $S$  [1]. The gain distortion,  $D_g$ , is given by

$$D_g = \frac{1}{n} \int_0^\infty (r - \hat{g}(r))^2 f_g(r) dr,$$

where  $\hat{g}(\cdot)$  is the quantized gain and  $f_g(\cdot)$  is the pdf of  $g$ .

Conditioned on the gain codeword  $\hat{g}_j$  chosen, the shape  $S$  is distributed uniformly on  $\Omega_n$ , which has surface area  $S_n = 2\pi^{n/2}/\Gamma(n/2)$ . Thus, as shown in [1], for asymptotically high shape rate  $R_s$ , the conditional distortion  $E \left[ \|S - \hat{S}\|^2 \mid \hat{g}_j \right]$  is equal to the distortion of the lattice quantizer with codebook  $(\Lambda, d_\Lambda^j)$  for a uniform source in  $\mathbb{R}^{n-1}$ . Thus,

$$E \left[ \|S - \hat{S}\|^2 \mid \hat{g}_j \right] = (n-1)G(\Lambda)V_j(\Lambda)^{2/(n-1)}, \quad (9)$$

where  $V_j(\Lambda)$  is the volume of a Voronoi region of the  $(n-1)$ -dimensional lattice  $(\Lambda, d_\Lambda^j)$ . Therefore, for a given gain codebook  $\{\hat{g}_1, \hat{g}_2, \dots, \hat{g}_J\}$ , the shape distortion  $D_s$  can be approximated by

$$D_s = \frac{1}{n} \sum_{j=1}^J p_j \hat{g}_j^2 E \left[ \|S - \hat{S}\|^2 \mid \hat{g} = \hat{g}_j \right] \quad (10)$$

$$\approx \frac{1}{n} \sum_{j=1}^J p_j \hat{g}_j^2 (n-1)G(\Lambda)V_j(\Lambda)^{2/(n-1)} \quad (11)$$

$$\approx \frac{1}{n} \sum_{j=1}^J p_j \hat{g}_j^2 (n-1)G(\Lambda) (S_n/M_j)^{2/(n-1)} \quad (12)$$

$$= \frac{n-1}{n} G(\Lambda) \left( 2\pi^{n/2}/\Gamma(n/2) \right)^{\frac{2}{n-1}} \cdot \sum_{j=1}^J p_j \hat{g}_j^2 M_j^{\frac{-2}{n-1}},$$

where  $p_j$  in (10) is the probability of  $\hat{g}_j$  being chosen; (11) follows from (9);  $M_j$  in (12) is the number of codewords in the shape subcodebook associated with  $\hat{g}_j$ ; (12) follows from the high-rate assumption and neglecting the overlapping regions.

### C. Rate Allocations

The optimal rate allocation given below will be used as the rate allocation across subcodebooks in our generalized permutation codes.

*Theorem 2:* Let  $X \in \mathbb{R}^n$  be an i.i.d.  $\mathcal{N}(0, \sigma^2)$  vector, and let  $\Lambda$  be a lattice in  $\mathbb{R}^{n-1}$  with normalized second moment  $G(\Lambda)$ . Suppose  $X$  is quantized by a shape-gain VQ at rate  $R = R_g + R_s$  with gain-dependent shape codebook constructed from  $\Lambda$  with different minimum distances. Also, assume that a variable-rate coding follows the quantization. Then, the asymptotic decay of the distortion  $D$  is given by

$$\lim_{R \rightarrow \infty} D 2^{2R} = \frac{n}{(n-1)^{1-1/n}} \cdot C_g^{1/n} C_s^{1-1/n} \quad (13)$$

and is achieved by  $R_s = R_s^*$  and  $R_g = R_g^*$ , where

$$R_s^* = \left( \frac{n-1}{n} \right) \left[ R + \frac{1}{2n} \log \left( \frac{C_s}{C_g} \cdot \frac{1}{n-1} \right) \right], \quad (14)$$

$$R_g^* = \left( \frac{1}{n} \right) \left[ R - \frac{n-1}{2n} \log \left( \frac{C_s}{C_g} \cdot \frac{1}{n-1} \right) \right], \quad (15)$$

$$C_s = \frac{n-1}{n} G(\Lambda) \left( 2\pi^{n/2}/\Gamma(n/2) \right)^{2/(n-1)} \cdot 2\sigma^2 e^{\psi(n/2)}, \quad (16)$$

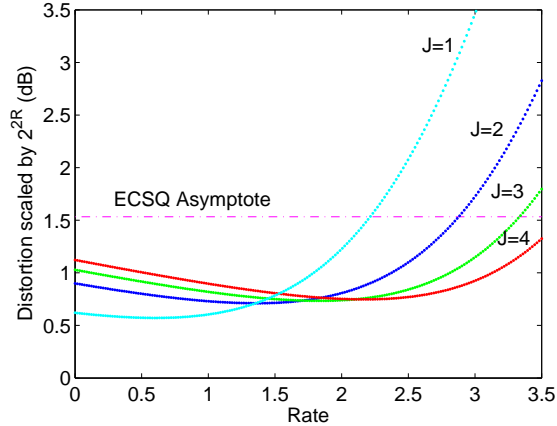


Fig. 2. High-resolution approximation of the rate–distortion performance of WSC with gain-dependent shape codebooks for coding an i.i.d.  $\mathcal{N}(0, 1)$  source with block length  $n = 25$ . The rate allocation computation assumes  $G(\Lambda_{24}) \approx 0.065771$ .

$$C_g = \sigma^2 \cdot \frac{3^{n/2} \Gamma^3(\frac{n+2}{6})}{8n \Gamma(n/2)},$$

and  $\psi(\cdot)$  is the digamma function.

*Proof:* The optimization of  $M_j$ s given the gain codebook can be accomplished using Lagrange multipliers. Then, the final results require limit computations similar to those in the proof of [1, Thm. 1]. Details are omitted for brevity. ■

Note that, in comparison with the asymptotic performance of independent shape–gain encoding given in [1, Thm. 1], the above theorem yields the following SNR improvement:

$$\Delta_{\text{SNR}}(n) = -10(1 - 1/n) \log_{10}(2e^{\psi(n/2)}/n) \quad (\text{dB}). \quad (17)$$

From the theory of the gamma function [13],  $[\psi(n/2) - \ln(n/2)] \rightarrow 0$ , and thus  $\Delta_{\text{SNR}}(n) \rightarrow 0$ , as  $n \rightarrow \infty$ , which is not surprising because of the “sphere hardening” effect.

A similar optimal rate allocation is possible for fixed-rate coding. Fig. 2 illustrates the resulting performance as a function of the rate for several values of  $J$ . As expected, for a fixed block size  $n$ , higher rates require higher values of  $J$  (more concentric spheres) to attain good performance and the best performance is improved by increasing the maximum value for  $J$ .

#### D. Using WSC Rate Allocation for Permutation Codes

In this section we use the optimal rate allocations for WSC to guide the design of generalized permutation codes at a given rate. The rate allocations are used to set target sizes for each subcodebook. Then for each subcodebook  $C_j$ , an integer partition meeting the constraint on  $M_j$  is selected (using heuristics inspired by [4, Thm. 3]). The Lloyd algorithm of Sec. V-A is then used for those integer partitions to compute the actual rate and distortion.

Results for the fixed-rate case are plotted in Fig. 3. This demonstrates that using the rate allocation of WSC with gain-dependent shape codebook actually yields good permutation codes for most of the rates.

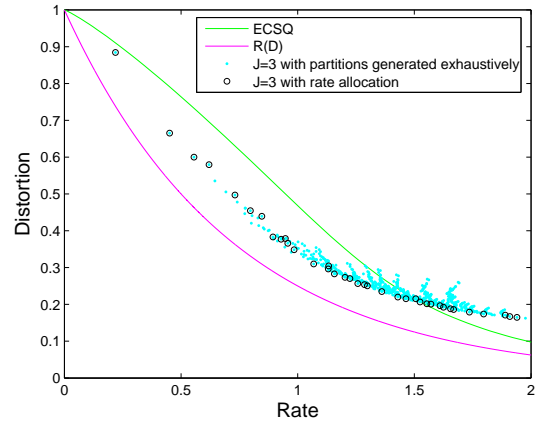


Fig. 3. Operational rate–distortion performance of generalized permutation codes designed with integer partitions guided by the WSC rate allocation and a Lloyd algorithm, in comparison with codes designed with exhaustive search over a heuristic subset of integer partitions. Computation uses i.i.d.  $\mathcal{N}(0, 1)$  source,  $n = 7$ , and  $J = 3$ .

## VI. CONCLUSIONS

We have proposed a generalization of permutation codes in which more than one initial codeword is allowed. This improves rate–distortion performance while adding very little to encoding complexity. However, the design complexity is increased considerably. We introduce two methods to reduce the design complexity: restricting the subcodebooks to share a common integer partition or allocating rates across subcodebooks using high-resolution analysis of wrapped spherical codes. Proving effectiveness of these heuristics is a remaining challenge.

## REFERENCES

- [1] J. Hamkins and K. Zeger, “Gaussian source coding with spherical codes,” *IEEE Trans. Inf. Theory*, vol. 48, pp. 2980–2989, Nov. 2002.
- [2] D. J. Sakrison, “A geometric treatment of the source encoding of a Gaussian random variable,” *IEEE Trans. Inf. Theory*, vol. IT-14, pp. 481–486, May 1968.
- [3] J. G. Dunn, “Coding for continuous sources and channels,” Ph.D. dissertation, Columbia Univ., New York, 1965.
- [4] T. Berger, F. Jelinek, and J. K. Wolf, “Permutation codes for sources,” *IEEE Trans. Inf. Theory*, vol. IT-18, pp. 160–169, Jan. 1972.
- [5] T. Berger, “Optimum quantizers and permutation codes,” *IEEE Trans. Inf. Theory*, vol. IT-18, pp. 759–765, Nov. 1972.
- [6] —, “Minimum entropy quantizers and permutation codes,” *IEEE Trans. Inf. Theory*, vol. IT-18, pp. 149–157, Mar. 1982.
- [7] H. A. David and H. N. Nagaraja, *Order Statistics*, 3rd ed. Wiley-Interscience, 2003.
- [8] V. K. Goyal, S. A. Savari, and W. Wang, “On optimal permutation codes,” *IEEE Trans. Inf. Theory*, vol. 47, pp. 2961–2971, Nov. 2001.
- [9] W. A. Finamore, S. V. B. Bruno, and D. Silva, “Vector permutation encoding for the uniform sources,” in *Proc. Data Compression Conf. (DCC 2004)*, Mar. 2004, p. 539.
- [10] H. W. Kuhn, “The Hungarian method for the assignment problem,” *Nav. Res. Logist. Q.*, vol. 2, pp. 83–97, 1955.
- [11] P. F. Swaszek and J. B. Thomas, “Optimal circularly symmetric quantizers,” *J. Franklin Inst.*, vol. 313, pp. 373–384, Jun. 1982.
- [12] J. Hamkins, “Design and analysis of spherical codes,” Ph.D. dissertation, Univ. Illinois at Urbana-Champaign, Sept. 1996.
- [13] J. L. W. V. Jensen, “An elementary exposition of the theory of the gamma function,” *Ann. Math.*, vol. 17, no. 3, pp. 124–166, Mar. 1916.