Generalizations of Permutation Source Codes

by

Ha Quy Nguyen

Submitted to the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2009

Author .....
Department of Electrical Engineering and Computer Science
September 4, 2009

Certified by .....
Vivek K Goyal
Esther and Harold E. Edgerton Associate Professor of Electrical Engineering
Thesis Supervisor

Accepted by .....
Terry P. Orlando
Chairman, Department Committee on Graduate Students

Generalizations of Permutation Source Codes

by

Ha Quy Nguyen

Submitted to the Department of Electrical Engineering and Computer Science
on September 4, 2009, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Permutation source codes are a class of structured vector quantizers with a computationally-simple encoding procedure. In this thesis, we provide two extensions that preserve the computational simplicity but yield improved operational rate–distortion performance. In the first approach, 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.

In the second approach, we introduce frame permutation quantization (FPQ), a new vector quantization technique using finite frames. In FPQ, a vector is encoded using a permutation source code to quantize its frame expansion. This means that the encoding is a partial ordering of the frame expansion coefficients. Compared to ordinary permutation source coding, FPQ produces a greater number of possible quantization rates and a higher maximum rate. Various representations for the partitions induced by FPQ are presented and reconstruction algorithms based on linear programming and quadratic programming are derived. Reconstruction using the canonical dual frame is also studied, and several results relate properties of the analysis frame to whether linear reconstruction techniques provide consistent reconstructions. Simulations for uniform and Gaussian sources show performance improvements over entropy-constrained scalar quantization for certain combinations of vector dimension and coding rate.

Thesis Supervisor: Vivek K Goyal
Title: Esther and Harold E. Edgerton Associate Professor of Electrical Engineering

Acknowledgments

I would like to thank my research advisor, Prof. Vivek Goyal, who led me into this exciting problem of permutation codes and its generalizations, and worked with me closely throughout the past two years. I am indebted to Vivek for his patience, encouragement, insights, and enthusiasm. My adaptation to the new school, the new life, the new research field, and the new ways of thinking could have not been that smooth without his guidance and support. It is no doubt that one of my luckiest things while studying at MIT was working under Vivek's supervision.

I thank Lav Varshney for his collaborations on most of the work in this thesis. It was Lav who initially extended Vivek's ideas of generalizing permutation codes in [1, 2], and got several results. Without his initial work, this thesis could not have been done. His advice helped me tremendously. I also thank other members of the STIR group: Adam, Daniel, Vinith, John, and Aniruddha, who provided criticism along the way.

I especially thank my parents for their unwavering and limitless support, my sister for her ample encouragement whenever I have troubles in the academic life.

My work at MIT was financially supported in part by a Vietnam Education Foundation fellowship, and by National Science Foundation Grant CCF-0729069.

Contents

Chapter 1

Introduction

Quantization (or analog data compression) plays an increasingly important role in the transmission and storage of analog data, due to the limited capacity of available channels and storage media. The simplest form of quantization is the *fixed-rate scalar quantization* (FSQ), in which the real line is partitioned into a fixed number of intervals, each mapped into a corresponding representation point such that the expected distortion is minimized. It is well-known that *entropy-constrained scalar quantization* (ECSQ)—where the distortion is minimized subject to the entropy of the output rate—significantly outperforms the FSQ within 1.53 dB of the rate–distortion limit, with respect to mean-squared-error fidelity criterion [3]. However, the major disadvantage of entropy-coded quantizers is that their variable output rates cause synchronization problems. Along with tackling the buffer overflow problems [4], various fixed-rate coding techniques have been developed to attain the ECSQ performance [5–7]. These coding schemes, however, considerably increase the encoding complexity.

The elegant but uncommon technique of *permutation source coding* (PSC)—which places all codewords on a single sphere—has asymptotic performance as good as ECSQ while enabling a simple encoding. On the other hand, 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. This well-known gap, along with the knowledge that the performance of PSCs does not improve monotonically with increasing vector length [8],

motivates the first part of this thesis.

We propose a generalization of PSCs whereby codewords are the distinct permutations of more than one *initial codeword*. While adding very little to the encoding complexity, this makes the codebook of the vector quantizer lie in the union of concentric spheres rather than in a single sphere. Our use of multiple spheres is similar to the wrapped spherical shape–gain vector quantization of Hamkins and Zeger [9]; one of our results, which may be of independent of interest, is an optimal rate allocation for that technique. Our use of permutations could be replaced by the action of other groups to obtain further generalizations [10].

Optimization of PSCs contains a difficult integer partition design problem. Our generalization makes the design problem more difficult, and our primary focus is on methods for reducing the design complexity. We demonstrate the effectiveness of these methods and improvements over ordinary PSCs through simulations.

The second part of this work incorporates PSCs with redundant representations obtained with frames, which are playing an ever-expanding role in signal processing due to design flexibility and other desirable properties [11, 12]. One such favorable property is robustness to additive noise [13]. This robustness, carried over to quantization noise (without regard to whether it is random or signal-independent), explains the success of both ordinary oversampled analog-to-digital conversion (ADC) and Σ–Δ ADC with the canonical linear reconstruction. But the combination of frame expansions with scalar quantization is considerably more interesting and intricate because boundedness of quantization noise can be exploited in reconstruction [14–22] and frames and quantizers can be designed jointly to obtain favorable performance [23].

This thesis introduces a new use of finite frames in vector quantization: *frame permutation quantization* (FPQ). In FPQ, permutation source coding is applied to a frame expansion of a vector. This means that the vector is represented by a partial ordering of the frame coefficients (Variant I) or by signs of the frame coefficients that are larger than some threshold along with a partial ordering of the absolute values of the significant coefficients (Variant II). FPQ provides a space partitioning that can be combined with additional signal constraints or prior knowledge to generate a

variety of vector quantizers. A simulation-based investigation that uses a probabilistic source model shows that FPQ outperforms PSC for certain combinations of signal dimensions and coding rates. In particular, improving upon PSC at low rates provides quantizers that perform better than entropy-coded scalar quantization (ECSQ) in certain cases [8]. Beyond the explication of the basic ideas in FPQ, the focus of this thesis is on how—in analogy to works cited above—there are several decoding procedures that can sensibly be used with the encoding of FPQ. One is to use the ordinary decoding in PSC for the frame coefficients followed by linear synthesis with the canonical dual; from the perspective of frame theory, this is the natural way to reconstruct. Taking a geometric approach based on *consistency* yields instead optimization-based algorithms. We develop both views and find conditions on the frame used in FPQ that relate to whether the canonical reconstruction is consistent.

1.1 Outline and Contributions

Chapter 2: Background

The chapter provides the requisite background by reviewing source coding preliminaries, especially spherical codes and PSCs, and fundamentals of frame theory. Section 2.1 first discusses spherical codes for memoryless Gaussian sources and the "hardening effect," then turns in to formal definitions of two variants of PSCs. Also, optimal encoding algorithms for PSCs are given, and optimization of the initial codeword is discussed. Section 2.2 reviews different types of frames and their relationship from the perspective of equivalence class. Our contributions in this chapter are derivations of the relationship between *modulated harmonic tight frames* and *zero-sum frames*, and the characterization of *real equiangular tight frames* in the codimension-1 case. Proof of the main result is deferred to Appendix 2.A.

Chapter 3: Concentric Permutation Source Codes

This chapter introduces PSCs 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 Section 3.3. The use of a common integer partition obviates the issue of allocating rate amongst concentric spheres of codewords. Section 3.4 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 [9] for the purpose of guiding the rate allocation problem. These results may be of independent interest.

Chapter 4: Permutation Frame Quantization

The chapter is organized as follows: Before formal introduction to the combination of frame expansions and PSCs, Section 4.1 provides a preview of the geometry of FPQ. This serves both to contrast with ordinary scalar-quantized frame expansions and to see the effect of frame redundancy. Section 4.2 reviews vector quantization and PSCs from the perspective of decoding. Section 4.3 discusses scalar-quantized frame expansions with focus on linear-program-based reconstruction algorithm which can be extended to FPQ. Section 4.4 formally defines FPQ, emphasizing constraints that are implied by the representation and hence must be satisfied for consistent reconstruction. The results on choices of frames in FPQ appear in Section 4.5. These are necessary and sufficient conditions on frames for linear reconstructions to be consistent. Section 4.6 provides simulation results that demonstrate improvement in operational rate–distortion compared to ordinary PSC. Proofs of the main results are given in Appendices 4.A and 4.B. Preliminary results on FPQ were mentioned briefly in [24].

Chapter 5: Closing Remarks

The final chapter recapitulates the main results of the thesis and provides suggestions for future work.

Bibliographical Note

A significant fraction of the thesis is taken verbatim from the following joint works:

- H. Q. Nguyen, V. K Goyal and L. R. Varshney, "On Concentric Spherical Codes and Permutation Codes With Multiple Initial Codewords," in *Proceedings of the IEEE International Symposium on Information Theory*, Seoul, Korea, June 28-July 3, 2009.

- H. Q. Nguyen, L. R. Varshney and V. K Goyal, "Concentric Permutation Source Codes," *IEEE Trans. Commun.*, submitted for publication.

- H. Q. Nguyen, V. K Goyal and L. R. Varshney, "Frame Permutation Quantization," *Appl. Comput. Harmon. Anal.*, submitted for publication.

Chapter 2

Background

2.1 Source Coding Preliminaries

Let $X \in \mathbb{R}^n$ be a random vector with independent 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.

2.1.1 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 αX are the same for any scalar $\alpha > 0$. Thus a spherical code essentially ignores $\|X\|$, placing all codewords at a single radius. The logical code radius is $E[\|X\|]$.

Let us assume that $X \in \mathbb{R}^n$ is now i.i.d. $\mathcal{N}(0, \sigma^2)$. Sakrison [25] first analyzed the performance of spherical codes for memoryless Gaussian sources. Following [9, 25], using the code radius

$$E\left[\|X\|\right] = \frac{\sqrt{2\pi\sigma^2}}{\beta(n/2, 1/2)} \approx \sigma\sqrt{n - 1/2}, \tag{2.1}$$

the distortion can be decomposed as

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

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 $n \to \infty$.

2.1.2 Permutation Source Codes

In a *permutation source code* (PSC), all the codewords are related by permutation and thus have equal length. So PSCs are spherical codes. Permutation source codes were originally introduced as channel codes by Slepian [26], and then applied to a specific source coding problem, through the source encoding-channel decoding duality, by Dunn [27]. Berger et al. [28–30] developed PSCs in much more generality.

Variant I: Let $\mu_1 > \mu_2 > \cdots > \mu_K$ be K ordered real numbers, and let $n_1, n_2, \ldots, n_K$ be positive integers with sum equal to n (an *integer partition* of n). The *initial codeword* of the codebook $\mathcal{C}$ has the form

$$\hat{x}_{\text{init}} = (\underbrace{\mu_1, \ldots, \mu_1}_{n_1}, \underbrace{\mu_2, \ldots, \mu_2}_{n_2}, \ldots, \underbrace{\mu_K, \ldots, \mu_K}_{n_K}), \tag{2.2}$$

where each μ_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

$$L_{\text{I}} = \frac{n!}{n_1!\, n_2! \cdots n_K!}. \tag{2.3a}$$

Variant II: Here codewords are related through permutations and sign changes.

Algorithm 2.1 Variant I Encoding Algorithm

1. Replace the n_1 largest components of X with μ_1.
2. Replace the n_2 next largest components of X with μ_2.

$\vdots$

K. Replace the n_K smallest components of X with μ_K.

Algorithm 2.2 Variant II Encoding Algorithm

1. Replace the n_1 components of X largest in absolute value by either $+\mu_1$ or $-\mu_1$, the sign chosen to agree with that of the component it replaces.
2. Replace the n_2 components of X next largest in absolute value by either $+\mu_2$ or $-\mu_2$, the sign chosen to agree with that of the component it replaces.

$\vdots$

K. Replace the n_K components of X smallest in absolute value by either $+\mu_K$ or $-\mu_K$, the sign chosen to agree with that of the component it replaces.

Let $\mu_1 > \mu_2 > \cdots > \mu_K \geq 0$ be nonnegative real numbers, and let $(n_1, n_2, \ldots, n_K)$ be an integer partition of n. The initial codeword has the same form as in (2.2), and the codebook now consists of all distinct permutations of $\hat{x}_{\text{init}}$ with each possible sign for each nonzero component. The number of codewords in $\mathcal{C}$ is thus given by

$$L_{\text{II}} = 2^h \frac{n!}{n_1!\, n_2! \cdots n_K!}, \tag{2.3b}$$

where $h = n$ if $\mu_K > 0$ and $h = n - n_K$ if $\mu_K = 0$.

The permutation structure of the codebook enables low-complexity nearest-neighbor encoding procedures for both Variants I and II PSCs [28], which are given in Algorithm 2.1 and Algorithm 2.2, respectively.

It is important to note that 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.

The following theorem [28, Thm. 1] formalizes the optimal encoding algorithms for PSCs, and for a general distortion measure.

Theorem 2.1 ([28]). *Consider a block distortion measure of the form*

$$d(x, \hat{x}) = g\left(\sum_{i=1}^{n} f\left(|x_i - \hat{x}_i|\right)\right), \tag{2.4}$$

where $x = (x_1, \ldots, x_n)$, $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_n)$, $g(\cdot)$ *is nondecreasing, and* $f(\cdot)$ *is nonnegative, nondecreasing, and convex for positive arguments. Then optimum encoding of Variant I and Variant II PSCs is accomplished by Algorithm 2.1 and Algorithm 2.2, respectively.*

Proof. See [28, Appendix 1]. □

In this thesis, we only consider the popular mean-squared error (MSE) as the distortion measure, and restrict the sources to be independent and identically distributed (i.i.d.). 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

$$R = n^{-1} \log L. \tag{2.5}$$

Let $\xi_1 \geq \xi_2 \geq \ldots \geq \xi_n$ denote the order statistics of random vector $X = (X_1, \ldots, X_n)$, and $\eta_1 \geq \eta_2 \geq \ldots \geq \eta_n$ denote the order statistics of random vector $|X| \triangleq (|X_1|, \ldots, |X_n|)$.[1] With these notations, for a given initial codeword $\hat{x}_{\text{init}}$, the per-letter distortion of optimally encoded Variant I and Variant II PSCs are respectively given by

$$D_I = n^{-1} E\left[\sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i} (\xi_\ell - \mu_i)^2\right], \tag{2.6}$$

and

$$D_{II} = n^{-1} E\left[\sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i} (\eta_\ell - \mu_i)^2\right], \tag{2.7}$$

[1]Because of the convention of $\mu_i > \mu_{i+1}$, it is natural to index the order statistics in descending order as shown, which is opposite to the ascending convention in the order statistics literature [31].

where $\mathcal{I}_i$s are the groups of indices generated by the integer partition, i.e.,

$$\mathcal{I}_1 = \{1, 2, \ldots, n_1\}, \tag{2.8}$$

$$\mathcal{I}_i = \left\{ \left(\sum_{m=1}^{i-1} n_m \right) + 1, \ldots, \left(\sum_{m=1}^{i} n_m \right) \right\}, \quad i \geq 2. \tag{2.9}$$

Given an integer partition $(n_1, n_2, \ldots, n_K)$, minimization of D_{I} or D_{II} can be done separately for each μ_i, yielding optimal values

$$\mu_i = n_i^{-1} \sum_{\ell \in \mathcal{I}_i} E\left[\xi_\ell\right], \quad \text{for Variant I,} \tag{2.10}$$

and

$$\mu_i = n_i^{-1} \sum_{\ell \in \mathcal{I}_i} E\left[\eta_\ell\right], \quad \text{for Variant II.} \tag{2.11}$$

Overall minimization of D_{I} or D_{II} over the choice of K, $\{n_i\}_{i=1}^K$, and $\{\mu_i\}_{i=1}^K$ subject to a rate constraint is difficult because of the integer constraint of the partition.

The analysis of [29] shows that as n grows large, the integer partition can be designed to give performance equal to optimal entropy-constrained scalar quantization (ECSQ) of X. As discussed in the introduction, it could be deemed disappointing that this vector quantizer performs only as well as the best scalar quantizer. However, it should be noted that the PSC is producing fixed-rate output and hence avoiding the possibility of buffer overflow associated with entropy coding highly nonequiprobable outputs of a quantizer [32].

Heuristically, it seems that for large block lengths, PSCs 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. The technique we introduce is for moderate values of n, for which the second term of (2.1) is not negligible; thus, it is not adequate to place all codewords on a single sphere.

2.2 Frame Definitions and Classifications

The theory of finite-dimensional frames is often developed for a Hilbert space $\mathbb{C}^N$ of complex vectors. In this thesis, we use frame expansions only for quantization using PSCs, which rely on order relations of real numbers. Therefore we limit ourselves to real finite frames. We maintain the Hermitian transpose notation * where a transpose would suffice because this makes several expressions have familiar appearances.

The Hilbert space of interest is $\mathbb{R}^N$ equipped with the standard inner product (dot product),

$$\langle x, y \rangle = x^T y = \sum_{k=1}^{N} x_k y_k,$$

for $x = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N$ and $y = [y_1, y_2, \ldots, y_N]^T \in \mathbb{R}^N$. The norm of a vector x is naturally induced from the inner product,

$$\|x\| = \sqrt{\langle x, x \rangle}.$$

Definition 2.2 ([13]). *A set of N-dimensional vectors, $\Phi = \{\phi_k\}_{k=1}^M \subset \mathbb{R}^N$, is called a* frame *if there exist a lower frame bound, $A > 0$, and an upper frame bound, $B < \infty$, such that*

$$A\|x\|^2 \leq \sum_{k=1}^{M} |\langle x, \phi_k \rangle|^2 \leq B\|x\|^2, \qquad \text{for all } x \in \mathbb{R}^N. \tag{2.12a}$$

The matrix $F \in \mathbb{R}^{M \times N}$ with kth row equal to ϕ_k^* is called the analysis frame operator. *F and Φ will be used interchangeably to refer to a frame. Equivalent to (2.12a) in matrix form is*

$$AI_N \leq F^*F \leq BI_N, \tag{2.12b}$$

where I_N is the $N \times N$ identity matrix.

The lower bound in (2.12) implies that Φ spans $\mathbb{R}^N$; thus a frame must have $M \geq N$. It is therefore reasonable to call the ratio $r = M/N$ the *redundancy* of the frame. A frame is called a *tight frame* (TF) if the frame bounds can be chosen to be equal. A frame is an *equal-norm frame* (ENF) if all of its vectors have the same norm. If an ENF is normalized to have all vectors of unit norm, we call it a *unit-norm*

frame (UNF) (or sometimes *normalized frame* or *uniform frame*). For a unit-norm frame, it is easy to verify that $A \leq r \leq B$. Thus, a unit-norm tight frame (UNTF) must satisfy $A = r = B$ and

$$F^*F = rI_N. \tag{2.13}$$

Naimark's theorem [33] provides an efficient way to characterize the class of equal-norm tight frames (ENTFs): a set of vectors is an ENTF if and only if it is the orthogonal projection (up to a scale factor) of an orthonormal basis of an ambient Hilbert space on to some subspace.[2] As a consequence, deleting the last $(M - N)$ columns of the (normalized) discrete Fourier Transform (DFT) matrix in $\mathbb{C}^{M\times M}$ yields a particular subclass of UNTF called *(complex) harmonic tight frames* (HTFs). One can adapt this derivation to construct *real* HTFs [34], which are always UNTFs, as follows.

Definition 2.3. *The real harmonic tight frame of M vectors in $\mathbb{R}^N$ is defined for even N by*

$$\begin{aligned}\phi^*_{k+1} = \sqrt{\frac{2}{N}}\Bigg[&\cos\frac{k\pi}{M}, \cos\frac{3k\pi}{M}, \ldots, \cos\frac{(N-1)k\pi}{M},\\ &\sin\frac{k\pi}{M}, \sin\frac{3k\pi}{M}, \ldots, \sin\frac{(N-1)k\pi}{M}\Bigg]\end{aligned} \tag{2.14a}$$

and for odd N by

$$\begin{aligned}\phi^*_{k+1} = \sqrt{\frac{2}{N}}\Bigg[&\frac{1}{\sqrt{2}}, \cos\frac{2k\pi}{M}, \cos\frac{4k\pi}{M}, \ldots, \cos\frac{(N-1)k\pi}{M},\\ &\sin\frac{2k\pi}{M}, \sin\frac{4k\pi}{M}, \ldots, \sin\frac{(N-1)k\pi}{M}\Bigg],\end{aligned} \tag{2.14b}$$

where $k = 0, 1, \ldots, M-1$. The modulated harmonic tight frames *are defined by*

$$\psi_k = \gamma(-1)^k\phi_k, \qquad \text{for } k = 1, 2, \ldots, M, \tag{2.15}$$

where $\gamma = 1$ or $\gamma = -1$ (fixed for all k).

[2]The theorem holds for a general separable Hilbert space of possibly infinite dimension.

HTFs can be viewed as the result of a group of orthogonal operators acting on one generating vector [12]. This property has been generalized in [35, 36] under the name *geometrically uniform frames* (GUFs). Note that a GUF is a special case of a group code as developed by Slepian [10, 26]. An interesting connection between PSCs and GUFs is that under certain conditions, a PSC codebook is a GUF with generating vector $\hat{x}_{\text{init}}$ and the generating group action provided by all permutation matrices [37].

Classification of frames is often up to some unitary equivalence. Holmes and Paulsen [38] proposed several types of equivalence relations between frames. In particular, for two frames in $\mathbb{R}^N$, $\Phi = \{\phi_k\}_{k=1}^M$ and $\Psi = \{\psi_k\}_{k=1}^M$, we say Φ and Ψ are

(i) type I equivalent if there is an orthogonal matrix U such that $\psi_k = U\phi_k$ for all k;

(ii) type II equivalent if there is an permutation $\sigma(\cdot)$ on $\{1, 2, \ldots, M\}$ such that $\psi_k = \phi_{\sigma(k)}$ for all k; and

(iii) type III equivalent if there is a *sign function* in k, $\delta(k) = \pm 1$ such that $\psi_k = \delta(k)\phi_k$ for all k.

Two frames are called equivalent if they are equivalent according to one of the three types of equivalence relations above. It is important to note that for $M = N + 1$ there is exactly one equivalence class of UNTFs [34, Thm. 2.6]. Since HTFs are always UNTFs, the following property follows directly from [34, Thm. 2.6].

Proposition 2.4. *Assume that $M = N+1$, and $\Phi = \{\phi_k\}_{k=1}^M \subset \mathbb{R}^N$ is the real HTF. Then every UNTF $\Psi = \{\psi_k\}_{k=1}^M$ can be written as*

$$\psi_k = \delta(k)U\phi_{\sigma(k)}, \qquad \text{for } k = 1, 2, \ldots, M, \tag{2.16}$$

where $\delta(k) = \pm 1$ is some sign function in k, U is some orthogonal matrix, and $\sigma(\cdot)$ is some permutation on the index set $\{1, 2, \ldots, M\}$.

Another important subclass of UNTFs is defined as follows:

Definition 2.5 ([39, 40]). *A UNTF* $\Phi = \{\phi_k\}_{k=1}^M \subset \mathbb{R}^N$ *is called an* equiangular tight frame *(ETF) if there exists a constant* a *such that* $|\langle \phi_\ell, \phi_k \rangle| = a$ *for all* $1 \leq \ell < k \leq M$.

ETFs are sometimes called *optimal Grassmannian frames* or *2-uniform frames.* They prove to have rich application in communications, coding theory, and sparse approximation [38,39,41]. For a general pair (M, N), the existence and constructions of such frames is not fully understood. Partial answers can be found in [40,42,43].

In our analysis of FPQ, we will find that *restricted* ETFs—where the absolute value constraint can be removed from Definition 2.5—play a special role. In matrix view, a restricted ETF satisfies $F^*F = rI_N$ and $FF^* = (1-a)I_M + aJ_M$, where J_M is the all-1s matrix of size $M \times M$. The following proposition specifies the restricted ETFs for the codimension-1 case.

Proposition 2.6. *For* $M = N + 1$*, the family of all restricted ETFs is constituted by the Type I and Type II equivalents of modulated HTFs.*

Proof. See Appendix 2.A. □

The following property of modulated HTFs in the $M = N + 1$ case will be very useful.

Proposition 2.7. *If* $M = N+1$ *then a modulated harmonic tight frame is a zero-sum frame, i.e., each column of the analysis frame operator* F *sums to zero.*

Proof. We only consider the case when N is even; the N odd case is similar. For each $\ell \in \{1, \ldots, N\}$, let ϕ_k^ℓ denote the ℓth component of vector ϕ_k and let $S_\ell = \sum_{k=1}^M \phi_k^\ell$ denote the sum of the entries in column ℓ of matrix F.

For $1 \leq \ell \leq N/2$, using Euler's formula, we have

$$\begin{aligned} S_\ell &= \pm\sqrt{\frac{2}{N}} \sum_{k=0}^{M-1} (-1)^k \cos \frac{(2\ell-1)k\pi}{M} \\ &\propto \sum_{k=0}^{M-1} e^{jk\pi} \left[e^{j(2\ell-1)k\pi/M} + e^{-j(2\ell-1)k\pi/M} \right] \\ &= \sum_{k=0}^{M-1} e^{j\pi((2\ell-1)/M+1)k} + \sum_{k=0}^{M-1} e^{-j\pi((2\ell-1)/M+1)k} \\ &= \frac{1 - e^{j\pi(2\ell+M-1)}}{1 - e^{j\pi((2\ell-1)/M+1)}} + \frac{1 - e^{-j\pi(2\ell+M-1)}}{1 - e^{-j\pi((2\ell-1)/M+1)}} \\ &= 0, \end{aligned} \tag{2.17}$$

where (2.17) follows form the fact that $2\ell + M - 1 = 2\ell + N$ is an even integer.

For $N/2 < \ell \leq N$, we can show that $S_\ell = 0$ similarly, and so the proposition is proved. □

Appendix

2.A Proof of Proposition 2.6

In order to prove Proposition 2.6, we need the following lemmas.

Lemma 2.8. *Assume that $M = N + 1$ and let $W = e^{j2\pi/M}$. Then for all $\alpha \in \mathbb{R}$ we have*

$$\sum_{i=1}^{N/2} W^{\alpha(2i-1)/2} = \frac{(-1)^\alpha - W^{\alpha/2}}{W^\alpha - 1}, \qquad \text{if } N \text{ is even;}$$

and

$$\sum_{i=1}^{(N-1)/2} W^{\alpha i} = \frac{(-1)^\alpha - W^\alpha}{W^\alpha - 1}, \qquad \text{if } N \text{ is odd.}$$

Proof. By noting that $W^{M/2} = -1$, we have the following computations.

N even:

$$\begin{aligned}\sum_{i=1}^{N/2} W^{\alpha(2i-1)/2} &= W^{-\alpha/2} \cdot \sum_{i=1}^{N/2} W^{\alpha i} = W^{-\alpha/2} \cdot \frac{W^{\alpha(N+2)/2} - W^\alpha}{W^\alpha - 1} \\ &= W^{-\alpha/2} \cdot \frac{\left(W^{M/2}\right)^\alpha W^{\alpha/2} - W^\alpha}{W^\alpha - 1} = \frac{(-1)^\alpha - W^{\alpha/2}}{W^\alpha - 1}.\end{aligned}$$

N odd:

$$\sum_{i=1}^{(N-1)/2} W^{\alpha i} = \frac{W^{\alpha(N+1)/2} - W^\alpha}{W^\alpha - 1} = \frac{\left(W^{M/2}\right)^\alpha - W^\alpha}{W^\alpha - 1} = \frac{(-1)^\alpha - W^\alpha}{W^\alpha - 1}.$$

□

Lemma 2.9. *For $M = N+1$, the HTF $\Phi = \{\phi_k\}_{k=1}^M$ satisfies $\langle \phi_k, \phi_\ell \rangle = (-1)^{k-\ell+1}/N$, for all $1 \le k < \ell \le M$.*

Proof. Consider two following cases.

N even: Using Euler's formula, for $k = 0, 1, \ldots, M-1$, ϕ_{k+1}^* can be rewritten as

$$\sqrt{\frac{2}{N}} \left[\frac{W^k + W^{-k}}{2}, \frac{W^{3k} + W^{-3k}}{2}, \ldots, \frac{W^{(N-1)k} + W^{-(N-1)k}}{2}, \right.$$
$$\left. \frac{W^k - W^{-k}}{2j}, \frac{W^{3k} - W^{-3k}}{2j}, \ldots, \frac{W^{(N-1)k} - W^{-(N-1)k}}{2j} \right].$$

For $1 \leq k < \ell \leq M$, let $\alpha = k - \ell$. After some algebraic manipulations we can obtain

$$\begin{aligned} N \cdot \langle \phi_k, \phi_\ell \rangle &= \sum_{i=1}^{N/2} W^{(k-\ell)(2i-1)/2} + \sum_{i=1}^{N/2} W^{(\ell-k)(2i-1)/2} \\ &= \frac{(-1)^\alpha - W^{\alpha/2}}{W^\alpha - 1} + \frac{(-1)^{-\alpha} - W^{-\alpha/2}}{W^{-\alpha} - 1} \qquad (2.18) \\ &= \frac{(-1)^\alpha W^{-\alpha/2} - 1}{W^{\alpha/2} - W^{-\alpha/2}} - \frac{(-1)^\alpha W^{\alpha/2} - 1}{W^{\alpha/2} - W^{-\alpha/2}} \\ &= (-1)^{\alpha+1}, \end{aligned}$$

where (2.18) is obtained using Lemma 2.8.

N odd: Similarly, for $1 \leq k < \ell \leq M$ and $\alpha = k - \ell$, we have

$$\begin{aligned} N \cdot \langle \phi_k, \phi_\ell \rangle &= 1 + \sum_{i=1}^{(N-1)/2} W^{(k-\ell)i} + \sum_{i=1}^{(N-1)/2} W^{(\ell-k)i} \\ &= 1 + \frac{(-1)^\alpha - W^\alpha}{W^\alpha - 1} + \frac{(-1)^{-\alpha} - W^{-\alpha}}{W^{-\alpha} - 1} \qquad (2.19) \\ &= 1 + \frac{(-1)^\alpha W^{-\alpha/2} - W^{\alpha/2}}{W^{\alpha/2} - W^{-\alpha/2}} - \frac{(-1)^\alpha W^{\alpha/2} - W^{-\alpha/2}}{W^{\alpha/2} - W^{-\alpha/2}} \\ &= 1 - (-1)^\alpha - 1 \\ &= (-1)^{\alpha+1}, \end{aligned}$$

where (2.19) is due to Lemma 2.8. □

Proposition 2.6. For a modulated HTF $\Psi = \{\psi_k\}_{k=1}^M$, as defined in Definition 2.3, for all $1 \leq k < \ell \leq M$ we have

$$\begin{aligned} \langle \psi_k, \psi_\ell \rangle &= \langle \gamma(-1)^k \phi_k, \gamma(-1)^\ell \phi_\ell \rangle \\ &= \gamma^2 (-1)^{k+\ell} (-1)^{k-\ell+1}/N \qquad (2.20) \\ &= (-1)^{k+\ell} (-1)^{k-\ell+1}/N \qquad (2.21) \\ &= -1/N, \qquad (2.22) \end{aligned}$$

where (2.20) is due to Lemma 2.9, and (2.21) is true because $|\gamma| = 1$ for all $1 \leq k < \ell \leq M$. Since the inner product is preserved through an orthogonal mapping,

(2.22) is true for Type I and/or Type II equivalences of modulated HTFs as well. The tightness and unit-norm of the HTF are obviously preserved for Type I and/or Type II equivalences. Therefore, the modulated HTFs and their equivalences of Type I and/or Type II are all restricted ETFs.

Conversely, from Proposition 2.4, every restricted ETF $\Psi = \{\psi\}_{k=1}^{M}$, can be represented up to Type I and Type II equivalences as follows:

$$\psi_k = \delta(k)\phi_k, \qquad \text{for all } 1 \leq k \leq M,$$

where $\delta(k) = \pm 1$ is some sign function on k. Thus, the constraint $\langle \psi_k, \psi_\ell \rangle = a$ for some constant a of a restricted ETF is equivalent to

$$\begin{aligned} aN &= N\delta(k)\delta(\ell) \cdot \langle \phi_k, \phi_\ell \rangle \\ &= \delta(k)\delta(\ell)(-1)^{k-\ell+1}, \quad \text{for all } 1 \leq k < \ell \leq M. \end{aligned}$$

Therefore, $\delta(k)\delta(\ell)(-1)^{k-\ell}$ is constant for all $1 \leq k < \ell \leq M$. If we fix k and vary ℓ, it is clear that the sign of $\delta(\ell)$ must be alternatingly changed. Thus, Ψ is one of the two HTFs specified in the proposition, completing the proof. □

Chapter 3

Concentric Permutation Source Codes

In this chapter, we generalize ordinary PSCs by allowing multiple initial codewords. The resulting codebook is contained in a set of concentric spheres. We assume through out the chapter that components of the source vector X are i.i.d. $\mathcal{N}(0, \sigma^2)$

3.1 Basic Construction

Let J be a positive integer. We will define a *concentric permutation source code* (CPSC) with J initial codewords. This is equivalent to having a codebook that is the union of J PSCs. Each notation from Section 2.1.2 is extended with a superscript or subscript $j \in \{1, 2, \ldots, J\}$ that indexes the constituent PSCs. 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

$$\hat{x}_{\text{init}}^j = \left(\mu_1^j, \ldots, \mu_1^j, \ldots, \mu_{K_j}^j, \ldots, \mu_{K_j}^j\right), \tag{3.1}$$

where each μ_i^j appears n_i^j times, $\mu_1^j > \mu_2^j > \cdots > \mu_{K_j}^j$ (all of which are nonnegative for Variant II case), and $\sum_{i=1}^{K_j} n_i^j = n$. Also, $\{\mathcal{I}_i^j\}_{i=1}^{K_j}$ are sets of indices generated by the jth integer partition.

Proposition 3.1. *Nearest-neighbor encoding of X with codebook $\mathcal{C}$ can be accom-*

plished with the following procedure:

1. *For each j, find $\hat{X}_j \in \mathcal{C}_j$ whose components have the same order as X.*

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

Proof. Suppose $X' \in \mathcal{C}$ is an arbitrary codeword. Since $\mathcal{C} = \cup_{j=1}^J \mathcal{C}_j$, there must exist $j_0 \in \{1, 2, \ldots, J\}$ such that $X' \in \mathcal{C}_{j_0}$. We have

$$\begin{aligned} E\left[\|X - \hat{X}\|\right] &\leq E\left[\|X - \hat{X}_{j_0}\|\right] && (3.2) \\ &\leq E\left[\|X - X'\|\right], && (3.3) \end{aligned}$$

where (3.2) follows from the second step of the algorithm, and (3.3) follows from the first step and the optimality of the encoding for ordinary PSCs. Thus, the encoding algorithm above is optimal. □

The first step of the algorithm requires $O(n \log n) + O(Jn)$ operations (sorting components of X and reordering each $\hat{x}_{\text{init}}^j$ according to the index matrix obtained from the sorting); 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 PSCs.

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

$$R \approx \frac{1}{n}\left[H\left(\{p_j\}_{j=1}^J\right) + \sum_{j=1}^J p_j \log M_j\right], \tag{3.4}$$

where $H(\cdot)$ denotes the entropy of a distribution, p_j is the probability of choosing subcodebook $\mathcal{C}_j$, and M_j is the number of codewords in $\mathcal{C}_j$. Without entropy coding, the rate is

$$R = \frac{1}{n} \log\left(\sum_{j=1}^J M_j\right). \tag{3.5}$$

The per-letter distortion for Variant I codes is now given by

$$\begin{aligned} D &= \frac{1}{n} E\left[\min_{1\le j\le J} \|X - \hat{X}_j\|^2\right] \\ &= \frac{1}{n} E\left[\min_{1\le j\le J} \sum_{i=1}^{K_j} \sum_{\ell\in\mathcal{I}_i^j} \left(\xi_\ell - \mu_i^j\right)^2\right], \end{aligned} \tag{3.6}$$

where (3.6) is obtained by rearranging the components of X and $\hat{X}_j$ in descending order. The distortion for Variant II codes has the same form as (3.6) with $\{\xi_\ell\}$ replaced by $\{\eta_\ell\}$.

Vector permutation codes are another generalization of PSCs with improved performance [44]. The encoding procedure, however, requires solving the assignment problem in combinatorial optimization [45] and has complexity $O(n^2\sqrt{n}\log n)$.

3.2 Optimization

In general, finding the best ordinary PSC requires an exhaustive search over all integer partitions of n. (Assuming a precomputation of all the order statistic means, the computation of the distortion for a given integer partition through either (2.6) or (2.7) is simple [28].) The search space can be reduced for certain distributions of X using [28, 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 (3.6), 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 this chapter, we consider two ways to reduce the design complexity. In Section 3.3, 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 Section 3.4, 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 [28, Thm. 3] holds. Since our designs are not provably optimal, the improvements from allowing multiple initial codewords could be somewhat larger than we demonstrate.

3.3 Design with Common Integer Partition

In this section, let us assume that the J integer partitions are equal, i.e., the n_i^js 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.

3.3.1 Common Integer Partitions Give Common Conic Partitions

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 3.2. *For common integer partition $\{n_1, n_2, \ldots, n_K\}$, the initial codewords $\{(\mu_1^j, \ldots, \mu_1^j, \ldots, \mu_K^j, \ldots, \mu_K^j)\}_{j=1}^J$ of Variant I CPSCs are optimal if and only if $\{\mu^1, \ldots, \mu^J\}$ are J optimal representation points of the vector quantization of $\overline{\xi} \in \mathbb{R}^K$, where*

$$\mu^j = \left(\sqrt{n_1}\,\mu_1^j, \sqrt{n_2}\,\mu_2^j, \ldots, \sqrt{n_K}\,\mu_K^j\right), \qquad 1 \le j \le J,$$

$$\bar{\xi} = \left(\frac{1}{\sqrt{n_1}} \sum_{\ell \in \mathcal{I}_1} \xi_\ell, \frac{1}{\sqrt{n_2}} \sum_{\ell \in \mathcal{I}_2} \xi_\ell, \ldots, \frac{1}{\sqrt{n_K}} \sum_{\ell \in \mathcal{I}_K} \xi_\ell \right).$$

Proof. Rewrite the distortion

$$\begin{aligned}
nD &= E\left[\min_{1 \le j \le J} \sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i} (\xi_\ell - \mu_i^j)^2 \right] \\
&= E\left[\min_{1 \le j \le J} \sum_{i=1}^{K} \left(\sum_{\ell \in \mathcal{I}_i} (\xi_\ell)^2 - 2\mu_i^j \sum_{\ell \in \mathcal{I}_i} \xi_\ell + n_i (\mu_i^j)^2 \right) \right] \\
&= E\left[\min_{1 \le j \le J} \sum_{i=1}^{K} \left(\frac{1}{\sqrt{n_i}} \sum_{\ell \in \mathcal{I}_i} \xi_\ell - \sqrt{n_i} \mu_i^j \right)^2 \right] \\
&\quad + E\left[\sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i} (\xi_\ell)^2 \right] - E\left[\sum_{i=1}^{K} \left(\frac{1}{\sqrt{n_i}} \sum_{\ell \in \mathcal{I}_i} \xi_\ell \right)^2 \right] \\
&= E\left[\min_{1 \le j \le J} \| \bar{\xi} - \mu^j \|^2 \right] + E\left[\|X\|^2 \right] \\
&\quad - E\left[\sum_{i=1}^{K} \left(\frac{1}{\sqrt{n_i}} \sum_{\ell \in \mathcal{I}_i} \xi_\ell \right)^2 \right]. \qquad (3.7)
\end{aligned}$$

Since the second and third terms of (3.7) do not depend on $\{\hat{x}_{\text{init}}^j\}_{j=1}^J$, minimizing D is equivalent to minimizing the first term of (3.7). By definition of a K-dimensional VQ, that term is minimized if and only if $\{\mu^1, \ldots, \mu^J\}$ are optimal representation points of the J-point VQ of random vector $\bar{\xi}$, completing the proof. □

Theorem 3.2 can be trivially extended for Variant II codes by simply replacing $\{\xi_\ell\}$ with $\{\eta_\ell\}$. For any fixed integer partition, it is straightforward to implement the J-point VQ design inspired by Theorem 3.2. Figure 3-1 illustrates the design of initial codewords for a given integer partition. Figure 3-2 compares the performance of an ordinary Variant I PSC ($J = 1$) with CPSCs with $J = 3$ initial vectors. For a given integer partition, the distortion of the optimal ordinary PSC is computed using (2.10) and variances of the order statistics (see [28, Eq. (13)]), whereas that of the optimal CPSC is estimated empirically from 500 000 samples generated according to the $\mathcal{N}(0, 1)$ distribution.

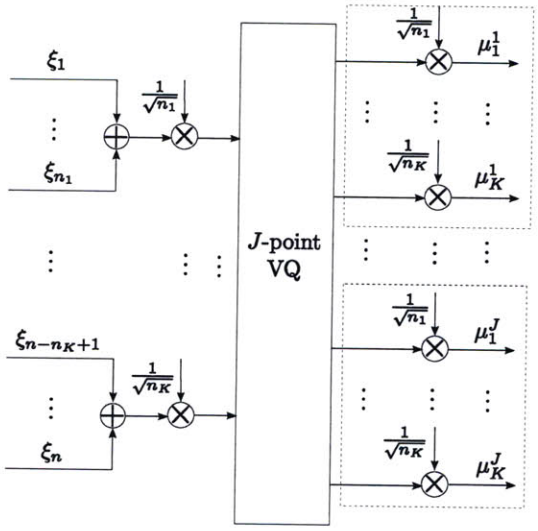


Figure 3-1: Block diagram for initial codewords design using Theorem 3.2. The inputs of the VQ box are K random variables and the outputs are J vectors, each of which has K components grouped within a dashed box.

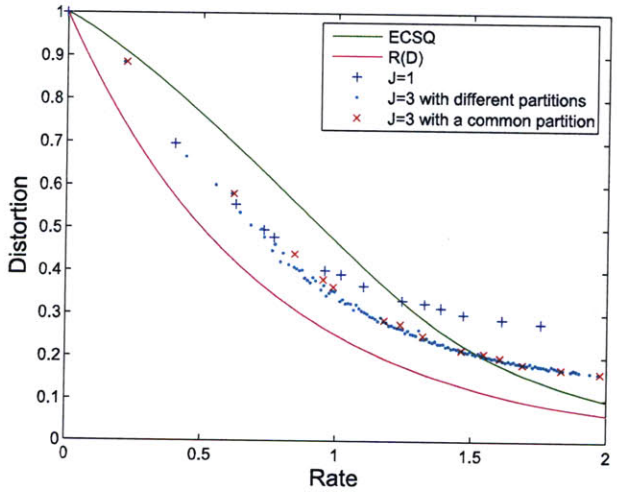


Figure 3-2: Operational rate–distortion performance for fixed-rate coding of i.i.d. $\mathcal{N}(0,1)$ source with block length $n = 7$. Ordinary Variant I PSCs ($J = 1$) are compared with CPSCs with $J = 3$. Codes with common integer partitions are designed according to Theorem 3.2. Codes with different integer partitions are designed with heuristic selection of integer partitions and Algorithm 3.1. For clarity, amongst approximately-equal rates, only operational points with the lowest distortion are plotted.

3.3.2 Optimization of Integer Partition

Although the optimization of integer partitions is not easy even for PSCs, for a certain class of distributions, there is a necessary condition for the optimal integer partition [28, Thm. 3]. The following conjecture is an analogy of that condition.

Conjecture 3.3. *Suppose that $J > 1$ and that $E[\eta_j]$ is a convex function of j, i.e.*

$$E[\eta_{j+2}] - 2E[\eta_{j+1}] + E[\eta_j] \geq 0, \quad 1 \leq j \leq n-2. \tag{3.8}$$

Then the optimum n_i for Variant II codes with J initial codewords increases monotonically with i.

The conjecture is true if one can show that the distortion associated with the integer partition $\{n_1, \ldots, n_m, n_{m+1}, \ldots, n_K\}$, where $n_m > n_{m+1}$, can be decreased by reversing the roles of n_m and n_{m+1}. As a plausibility argument for the conjecture, we will show that the reversing is possible for an additional constraint imposed on the codewords. Before stating a weaker proposition, we want to note that the convexity of $E[\eta_j]$ implies the nonnegativity of the expected value of the following random variable [28, Thm. 3]:

$$\zeta \triangleq \frac{1}{r}\sum_{L+1}^{L+r}\eta_\ell - \frac{2}{q-r}\sum_{L+r+1}^{L+q}\eta_\ell + \frac{1}{r}\sum_{L+q}^{L+q+r}\eta_\ell, \tag{3.9}$$

where $L = n_1 + n_2 + \ldots + n_{m-1}$. On the other hand, $E[\zeta]$ can be written as the difference of two nonnegative terms,

$$\overline{\zeta}_+ \triangleq E[\zeta \mid \zeta \geq 0]$$

and

$$\overline{\zeta}_- \triangleq -E[\zeta \mid \zeta \leq 0].$$

Since $E[\zeta] \geq 0$, it is clear that $\overline{\zeta}_+ \geq \overline{\zeta}_-$. Therefore, the following set is non-empty

$$\Omega_m = \left\{ \left\{\mu_i^j\right\}_{i,j} \text{ s.t. } \frac{\min_j\left(\mu_m^j - \mu_{m+1}^j\right)}{\max_j\left(\mu_m^j - \mu_{m+1}^j\right)} \geq \frac{\overline{\zeta}_-}{\overline{\zeta}_+} \right\}. \tag{3.10}$$

With the notations above, we are now ready to state the proposition.

Proposition 3.4. *Suppose that $J > 1$ and $E[\eta_j]$ is a convex function of j. If $n_m > n_{m+1}$ for some m, and the constraint Ω_m given in (3.10) is imposed on the codewords, then the distortion associated with the partition $\{n_1, \ldots, n_m, n_{m+1}, \ldots, n_K\}$ can be decreased by reversing the roles of n_m and n_{m+1}.*

Proof. See Appendix 3.A. □

A straightforward extension of Conjecture 3.3 for Variant I codes is the following

Conjecture 3.5. *Suppose that $J > 1$, and that $E[\xi_j]$ is convex over $\mathcal{S}_1 \triangleq \{1, 2, \ldots, \lfloor K/2 \rfloor\}$ and concave over $\mathcal{S}_2 \triangleq \{\lfloor K/2 \rfloor + 1, \lfloor K/2 \rfloor + 2, \ldots, K\}$. Then the optimum n_i for Variant I codes with J initial codewords increases monotonically with $i \in \mathcal{S}_1$ and decreases monotonically with $i \in \mathcal{S}_2$.*

It is important to emphasize that the convexity of $E[\eta_j]$ and $E[\xi_j]$ required in the conjectures above is a fairly broad condition satisfied by a certain class of distributions, including Gaussian ones [28, Thm. 4–5]. We will later restrict the integer partitions, while doing simulations for Variant I codes and Gaussian sources, to satisfy Conjecture 3.5.

3.4 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.[1] Furthermore, the apparent design complexity for the integer partitions is increased greatly to equal the number of integer partitions raised to the Jth power, namely $2^{J(n-1)}$.

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

[1] For a related two-dimensional visualization, compare [46, Fig. 3] against [46, Figs. 7–13].

design problem which is the sizing of the subcodebooks. For this, we extend the high-resolution analysis of [9]. For brevity, we limit our discussion to Variant I PSCs; Variant II could be generalized similarly.

3.4.1 Local Optimization of Initial Vectors

Given J initial codewords $\{\hat{x}^j_{\text{init}}\}_{j=1}^J$, for each $1 \leq j \leq J$, let $R_j \subset \mathbb{R}^n$ denote the quantization region corresponding to codeword $\hat{x}^j_{\text{init}}$, and let $E_j[\cdot]$ denote the expectation conditioned on $X \in R_j$. By extension of an argument in [28], the distortion conditioned on $X \in R_j$ is minimized with

$$\mu_i^j = \frac{1}{n_i^j} \sum_{\ell \in \mathcal{I}_i^j} E_j[\xi_\ell], \qquad 1 \leq i \leq K_j. \tag{3.11}$$

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

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

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 given in Algorithm 3.1. This algorithm was used to produce the operating points shown in Figure 3-2 for CPSCs with different integer partitions in which the distortion of a locally optimal code was computed empirically from 500 000 samples generated according to $\mathcal{N}(0, 1)$ distribution. 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.

3.4.2 Wrapped Spherical Shape–Gain Vector Quantization

Hamkins and Zeger [9] 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

Algorithm 3.1 Lloyd Algorithm for Initial Codeword Optimization from Given Integer Partition

1. Choose an arbitrary initial set of J representation vectors $\hat{x}^1_{\text{init}}, \hat{x}^2_{\text{init}}, \ldots, \hat{x}^J_{\text{init}}$.
2. For each j, determine the corresponding quantization region R_j.
3. For each j, $\hat{x}^j_{\text{init}}$ is set to the new value given by (3.11).
4. Repeat steps 2 and 3 until further improvement in MSE is negligible.

(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, Hamkins [47, pp. 102–104] 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 signal-to-noise ratio (SNR). While our results may be of independent interest, our present purpose is to guide the selection of $\{M_j\}_{j=1}^J$ in CPSCs.

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, $\{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_J\}$, is optimized for the gain pdf, e.g., using the scalar Lloyd-Max algorithm [48, 49]. 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 Ω_n, the unit sphere in $\mathbb{R}^n$. The same Λ 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^j_Λ depends on the quantized gain $\hat{g}_j$. We denote such sphere packing as (Λ, d^j_Λ).

The per-letter MSE distortion will be

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

where the omitted cross term is zero due to the independence of g and $\hat{g}$ from S [9]. 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 Ω_n, which has surface area $S_n = 2\pi^{n/2}/\Gamma(n/2)$. Thus, as shown in [9], 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 (Λ, d_Λ^j) for a uniform source in $\mathbb{R}^{n-1}$. Thus,

$$E\left[\|S-\hat{S}\|^2 \mid \hat{g}_j\right] = (n-1)G(\Lambda)V_j(\Lambda)^{2/(n-1)}, \tag{3.12}$$

where $V_j(\Lambda)$ is the volume of a Voronoi region of the $(n-1)$-dimensional lattice (Λ, d_Λ^j). Therefore, for a given gain codebook $\{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_J\}$, the shape distortion

D_s can be approximated by

$$\begin{aligned} D_s &= \frac{1}{n} E\left[\|\hat{g}\, S - \hat{g}\, \hat{S}\|^2\right] \\ &= \frac{1}{n} \sum_{j=1}^{J} p_j\, \hat{g}_j^2 E\left[\|S - \hat{S}\|^2 \mid \hat{g} = \hat{g}_j\right] \qquad (3.13) \\ &\approx \frac{1}{n} \sum_{j=1}^{J} p_j\, \hat{g}_j^2 (n-1) G(\Lambda) V_j(\Lambda)^{2/(n-1)} \qquad (3.14) \\ &\approx \frac{1}{n} \sum_{j=1}^{J} p_j\, \hat{g}_j^2 (n-1) G(\Lambda) \left(S_n / M_j\right)^{2/(n-1)} \qquad (3.15) \\ &= \frac{n-1}{n} G(\Lambda) S_n^{2/(n-1)} \sum_{j=1}^{J} p_j\, \hat{g}_j^2 M_j^{-2/(n-1)} \\ &= C \cdot \sum_{j=1}^{J} p_j\, \hat{g}_j^2 M_j^{\frac{-2}{n-1}}, \qquad (3.16) \end{aligned}$$

where p_j in (3.13) is the probability of $\hat{g}_j$ being chosen; (3.14) follows from (3.12); M_j in (3.15) is the number of codewords in the shape subcodebook associated with $\hat{g}_j$; (3.15) follows from the high-rate assumption and neglecting the overlapping regions; and in (3.16),

$$C \triangleq \frac{n-1}{n} G(\Lambda) \left(2\pi^{n/2} / \Gamma(n/2)\right)^{2/(n-1)}. \qquad (3.17)$$

3.4.3 Rate Allocations

The optimal rate allocation for high-resolution approximation to WSC given below will be used as the rate allocation across subcodebooks in our CPSCs. Before stating the theorem, we need the following lemma.

Lemma 3.6. *If there exist constants C_s and C_g such that*

$$\lim_{R_s \to \infty} D_s \cdot 2^{2(n/(n-1))R_s} = C_s \qquad (3.18)$$

and

$$\lim_{R_g \to \infty} D_g \cdot 2^{2nR_g} = C_g, \qquad (3.19)$$

then if $R = R_s + R_g$ is fixed, the minimum of $D = D_s + D_g$ satisfies

$$\lim_{R\to\infty} D2^{2R} = \frac{n}{(n-1)^{1-1/n}} \cdot C_g^{1/n} C_s^{1-1/n} \tag{3.20}$$

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

$$R_s^* = \left(\frac{n-1}{n}\right)\left[R + \frac{1}{2n}\log\left(\frac{C_s}{C_g}\cdot\frac{1}{n-1}\right)\right], \tag{3.21}$$

$$R_g^* = \left(\frac{1}{n}\right)\left[R - \frac{n-1}{2n}\log\left(\frac{C_s}{C_g}\cdot\frac{1}{n-1}\right)\right]. \tag{3.22}$$

Proof. See [9, Thm. 1]. □

Theorem 3.7. *Let $X \in \mathbb{R}^n$ be an i.i.d. $\mathcal{N}(0,\sigma^2)$ vector, and let Λ be a lattice in $\mathbb{R}^{n-1}$ with normalized second moment $G(\Lambda)$. Suppose X is quantized by an n-dimensional shape-gain VQ at rate $R = R_g + R_s$ with gain-dependent shape codebook constructed from Λ with different minimum distances. Also, assume that a variable-rate coding follows the quantization. Then, the asymptotic decay of the minimum mean-squared error D is given by*

$$\lim_{R\to\infty} D2^{2R} = \frac{n}{(n-1)^{1-1/n}} \cdot C_g^{1/n} C_s^{1-1/n} \tag{3.23}$$

and is achieved by $R_s = R_s^*$ and $R_g = R_g^*$, where R_s^* and $R_g = R_g^*$ are given in (3.21) and (3.22),

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

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

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

Proof. We first minimize D_s for a given gain codebook $\{\hat{g}_j\}_{j=1}^J$. From (3.17), ignoring

the constant C, we must perform the minimization

$$\min_{M_1,\ldots,M_J} \sum_{j=1}^{J} p_j\, \hat{g}_j^2\, M_j^{2/(1-n)}$$
$$\text{subject to} \sum_{j=1}^{J} p_j \log M_j = nR_s. \tag{3.24}$$

Using a Lagrange multiplier to get an unconstrained problem, we obtain the objective function

$$f = \sum_{j=1}^{J} p_j\, \hat{g}_j^2\, M_j^{2/(1-n)} - \lambda \sum_{j=1}^{J} p_j \log M_j. \tag{3.25}$$

Neglecting the integer constraint, we can take the partial derivatives

$$\frac{\partial f}{\partial M_j} = \frac{2}{1-n} p_j\, \hat{g}_j^2 M_j^{(n+1)/(1-n)} - \lambda p_j M_j^{-1}, \quad 1 \le j \le J. \tag{3.26}$$

Setting $\frac{\partial f}{\partial M_j} = 0,\ 1 \le j \le J$, yields

$$M_j = \left[\lambda(1-n)/(2\hat{g}_j^2)\right]^{(1-n)/2}. \tag{3.27}$$

Substituting into the constraint (3.24), we get

$$\sum_{j=1}^{J} p_j \log \left[\lambda(1-n)/(2g_j^2)\right]^{(1-n)/2} = nR_s.$$

Thus,

$$\begin{aligned} [\lambda(1-n)/2]^{(1-n)/2} &= 2^{nR_s - (n-1)\sum_{k=1}^{J} p_k \log \hat{g}_k} \\ &= 2^{nR_s - (n-1)E[\log \hat{g}]}. \end{aligned}$$

Therefore, it follows from (3.27) that the optimal size for jth shape subcodebook for a given gain codebook is

$$M_j = \hat{g}_j^{n-1} \cdot 2^{nR_s^* - (n-1)E[\log \hat{g}]}, \quad 1 \le j \le J. \tag{3.28}$$

The resulting shape distortion is

$$\begin{aligned} D_s &\approx C \cdot \sum_{j=1}^{J} p_j \, \hat{g}_j^2 \left(\hat{g}_j^{n-1} 2^{nR_s^* - (n-1)E[\log \hat{g}]} \right)^{2/(1-n)} \\ &= C \cdot 2^{2E[\log \hat{g}]} \cdot 2^{-2(n/(n-1))R_s^*}, \end{aligned}$$

where C is the same constant as specified in (3.17). Hence,

$$\begin{aligned} \lim_{R \to \infty} D_s \cdot 2^{2(n/(n-1))R_s^*} &= C \cdot \lim_{R_g^* \to \infty} 2^{2E[\log \hat{g}]} \\ &= C \cdot 2^{2E[\log g]} \qquad (3.29) \\ &= C \cdot 2\sigma^2 e^{\psi(n/2)} \qquad (3.30) \\ &= C_s, \qquad (3.31) \end{aligned}$$

where (3.29) follows from the high-rate assumption; and (3.30) follows from computing the expectation $E[\log g]$. On the other hand, it is shown in [9, Thm. 1] that

$$\lim_{R \to \infty} D_g \cdot 2^{2n(R - R_s^*)} = \lim_{R \to \infty} D_g \cdot 2^{2nR_g^*} = C_g. \qquad (3.32)$$

The limits (3.31) and (3.32) now allow us to apply Lemma 3.6 to obtain the desired result. □

Through this theorem we can verify the rate–distortion improvement as compared to independent shape–gain encoding by comparing C_g and C_s in the distortion formula to the analogous quantities in [9, Thm. 1]. C_g remains the same whereas C_s, which plays a more significant role in the distortion formula, is scaled by a factor of $2e^{\psi(n/2)}/n < 1$. In particular, the improvement in signal-to-quantization noise ratio achieved by the WSC with gain-dependent shape codebook is given by

$$\Delta_{\text{SNR}}\,(\text{in dB}) = -10(1 - 1/n)\log_{10}(2e^{\psi(n/2)}/n). \qquad (3.33)$$

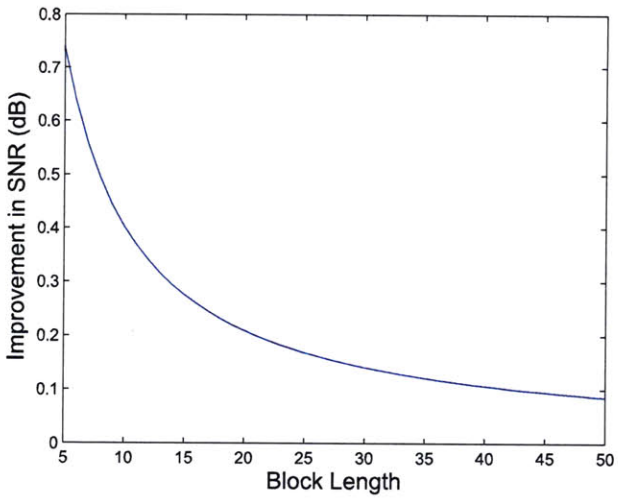


Figure 3-3: Improvement in signal-to-quantization noise ratio of WSC with gain-dependent shape quantizer specified in (3.33), as compared to the asymptotic rate–distortion performance given in [9, Thm. 1]

From the theory of the Gamma function [50, Eq. 29], we know that, for $s \in \mathbb{C}$,

$$\lim_{|s|\to\infty} [\psi(s) - \ln(s)] = 0.$$

It follows that $[\psi(n/2) - \ln(n/2)] \to 0$, and thus $\Delta_{\text{SNR}}(n) \to 0$, as $n \to \infty$; this is not surprising because of the "sphere hardening" effect. This improvement is plotted in Figure 3-3 as a function of block length n in the range between 5 and 50.

A similar optimal rate allocation is possible for fixed-rate coding.

Theorem 3.8. *Let $X \in \mathbb{R}^n$ be an i.i.d. $\mathcal{N}(0, \sigma^2)$ vector, and let Λ be a lattice in $\mathbb{R}^{n-1}$ with normalized second moment $G(\Lambda)$. Suppose X is quantized by an n-dimensional shape–gain VQ at rate R with gain-dependent shape codebook constructed from Λ with different minimum distances. Also, assume that J gain codewords are used and that a fixed-rate coding follows the quantization. Then, the optimal number of codewords in each subcodebook is*

$$M_j = 2^{nR} \cdot \frac{\left(p_j \hat{g}_j^2\right)^{(n-1)/(n+1)}}{\sum_{k=1}^{J} \left(p_k \hat{g}_k^2\right)^{(n-1)/(n+1)}}, \qquad 1 \le j \le J, \tag{3.34}$$

where $\{\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_J\}$ *is the optimal gain codebook. The resulting asymptotic decay of the shape distortion* D_s *is given by*

$$\lim_{R\to\infty} D_s 2^{2(n/(n-1))R} = C \cdot \left[\sum_{j=1}^{J} (p_j \hat{g}_j^2)^{\frac{n-1}{n+1}}\right]^{\frac{n+1}{n-1}}, \tag{3.35}$$

where p_j *is probability of* $\hat{g}_j$ *being chosen and* C *is the same constant as given in (3.17).*

Proof. For a given gain codebook $\{\hat{g}_j\}_{j=1}^{J}$, the optimal subcodebook sizes are given by the optimization

$$\begin{aligned} &\min_{M_1,\ldots,M_J} \sum_{j=1}^{J} p_j\, \hat{g}_j^2\, M_j^{2/(1-n)} \\ &\text{subject to } \sum_{j=1}^{J} M_j = 2^{nR}. \end{aligned} \tag{3.36}$$

Similarly to the variable-rate case, we can use a Lagrange multiplier to obtain an unconstrained optimization with the objective function

$$h = \sum_{j=1}^{J} p_j \hat{g}_j^2 M_j^{2/(1-n)} - \lambda \sum_{j=1}^{J} M_j. \tag{3.37}$$

Again, assuming high rate, we can ignore the integer constraints on M_j to take partial derivatives. Setting them equal to zero, one can obtain

$$M_j = \left[\lambda(1-n)/(2p_j\hat{g}_j^2)\right]^{(1-n)/(n+1)}. \tag{3.38}$$

Substituting into the constraint (3.36) yields

$$\sum_{j=1}^{J} \left[\lambda(1-n)/(2p_j\hat{g}_j^2)\right]^{(1-n)/(n+1)} = 2^{nR}.$$

Hence,

$$\lambda^{(n-1)/(n+1)} = 2^{-nR} \sum_{k=1}^{J} \left(\frac{1-n}{2p_k\hat{g}_k^2}\right)^{(1-n)/(n+1)}. \tag{3.39}$$

Combining (3.39) and (3.38) gives us

$$M_j = \lambda^{(1-n)/(n+1)} \left(\frac{1-n}{2p_j\hat{g}_j^2}\right)^{(1-n)/(n+1)} \tag{3.40}$$

$$= 2^{nR} \frac{\left(p_j\hat{g}_j^2\right)^{(n-1)/(n+1)}}{\sum_{k=1}^{J}\left(p_k\hat{g}_k^2\right)^{(n-1)/(n+1)}}, \quad 1 \le j \le J. \tag{3.41}$$

With the high-rate assumption, the resulting shape distortion will be

$$\begin{aligned} D_s &= C \cdot \sum_{j=1}^{J} p_j\hat{g}_j M_j^{2/(1-n)} \\ &= C \cdot \sum_{j=1}^{J} p_j\hat{g}_j \left[\frac{2^{nR}(p_j\hat{g}_j)^{(n-1)/(n+1)}}{\sum_{k=1}^{J}(p_k\hat{g}_k^2)^{(n-1)/(n+1)}}\right]^{2/(1-n)} \\ &= C \cdot 2^{-2(n/(n-1))R} \left[\sum_{j=1}^{J}(p_j\hat{g}_j^2)^{(n-1)/(n+1)}\right]^{\frac{n+1}{n-1}} \end{aligned} \tag{3.42}$$

where $C = \frac{n-1}{n}G(\Lambda)\left(2\pi^{n/2}/\Gamma(n/2)\right)^{2/(n-1)}$, completing the proof. □

Figure 3-4 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.

3.4.4 Using WSC Rate Allocation for CPSCs

In this section we use the optimal rate allocations for WSC to guide the design of CPSCs at a given rate. The rate allocations are used to set target sizes for each subcodebook. Then for each subcodebook $\mathcal{C}_j$, a partition meeting the constraint on M_j is selected (using heuristics inspired by Conjecture 3.5). Algorithm 3.1 of Section 3.4.1 is then used for those partitions to compute the actual rate and distortion.

For the variable-rate case, Theorem 3.7 provides the key rate allocation step in the design procedure given in Algorithm 3.2. Similarly, Theorem 3.8 leads to the design procedure for the fixed-rate case given in Algorithm 3.3. Each case requires as input

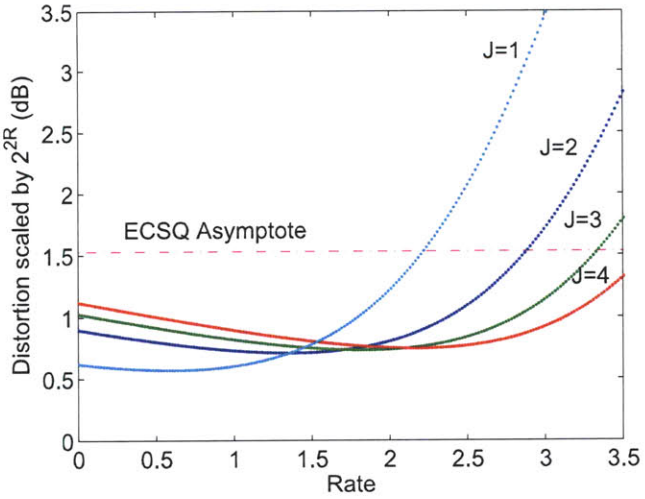


Figure 3-4: High-resolution approximation of the rate–distortion performance of WSC with gain-dependent shape codebooks and fixed-rate coding for an i.i.d. $\mathcal{N}(0,1)$ source with block length $n = 25$.

Algorithm 3.2 Design Algorithm for Variable-Rate Case

1. Compute R_s^* and R_g^* from (3.21) and (3.22), respectively.
2. For $1 \leq j \leq J$, compute M_j from (3.28).
3. For $1 \leq j \leq J$, search through all possible integer partitions of n that satisfy Conjecture 3.5, choosing the one that produces the number of codewords closest to M_j.
4. Run Algorithm 3.1 for the J partitions chosen in step 4 to generate the initial codewords and to compute the actual rate and distortion.

not only the rate R but also the number of initial codewords J.

Results for the fixed-rate case are plotted in Figure 3-5. This demonstrates that using the rate allocation of WSC with gain-dependent shape codebook actually yields good CPSCs for most of the rates. Figure 3-6 demonstrates the improvement that comes with allowing more initial codewords. The distortion is again computed empirically from Gaussian samples. It has a qualitative similarity with Figure 3-4.

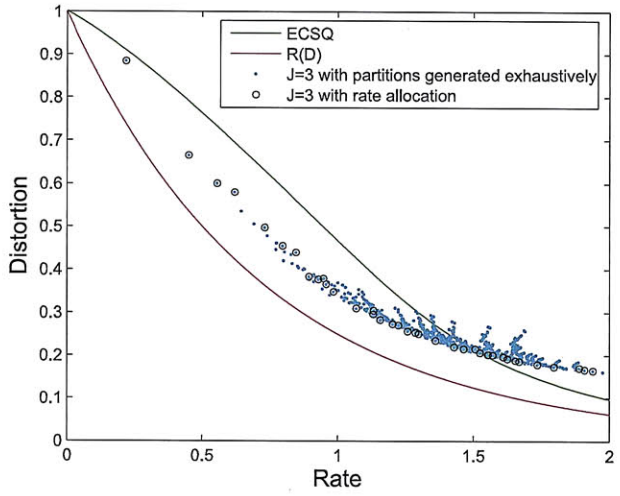


Figure 3-5: Operational rate–distortion performance of fixed-rate CPSCs designed with integer partitions guided by the WSC rate allocation and Algorithm 3.1, in comparison with codes designed with exhaustive search over a heuristic subset of integer partitions. Computation uses i.i.d. $\mathcal{N}(0,1)$ source, $n = 7$, and $J = 3$.

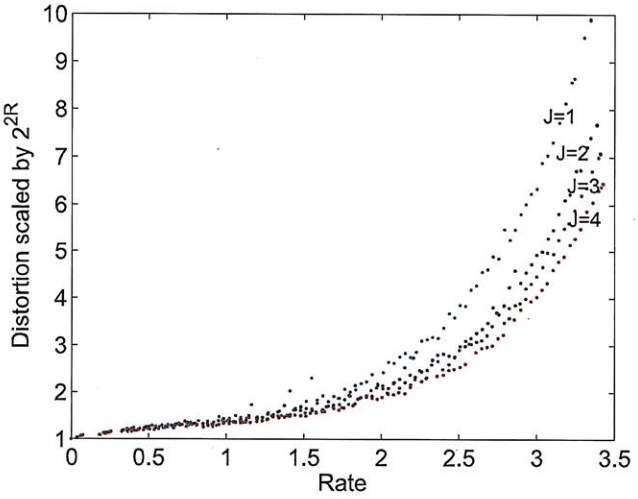


Figure 3-6: Operational rate–distortion performance for fixed-rate coding of i.i.d. $\mathcal{N}(0,1)$ source with block length $n = 25$. CPSCs with different integer partitions are designed using rate allocations from Theorem 3.8 and initial codewords locally optimized by Algorithm 3.1. The rate allocation computation assumes $G(\Lambda_{24}) \approx 0.065771$ [51, p. 61].

Algorithm 3.3 Design Algorithm for Fixed-Rate Case

1. Use the scalar Lloyd-Max algorithm to optimize J gain codewords.
2. For $1 \leq j \leq J$, compute M_j from (3.34)
3. Repeat steps 3 and 4 of Algorithm 3.2.

Appendix

3.A Proof of Proposition 3.4

Proof. Consider a new integer partition $\{n'_1, n'_2, \ldots, n'_K\}$ obtained by swapping n_m and n_{m+1}, i.e.,

$$n'_i = \begin{cases} n_i & i \neq m \text{ or } m+1 \\ n_{m+1} & i = m \\ n_m & i = m+1 \end{cases} \tag{3.43}$$

Let $\{\mathcal{I}'_i\}$ denote groups of indices generated by partition $\{n'_i\}$. Suppose that D is the optimal distortion associated with $\{n_i\}$,

$$D = \frac{1}{n} E\left[\min_{1\leq j\leq J} \sum_{i=1}^{K} \sum_{\ell\in\mathcal{I}_i} \left(\eta_\ell - \mu_i^j\right)^2\right], \tag{3.44}$$

where $\{\mu_i^j\}$ is the optimum of the minimization of the right side over Ω_m. Consider a suboptimal distortion D' associated with $\{n'_i\}$,

$$D' = \frac{1}{n} E\left[\min_{1\leq j\leq J} \sum_{i=1}^{K} \sum_{\ell\in\mathcal{I}'_i} \left(\eta_\ell - \tilde{\mu}_i^j\right)^2\right], \tag{3.45}$$

where $\{\tilde{\mu}_i^j\}$ is constructed from $\{\mu_i^j\}$ as follows, for each j,

$$\tilde{\mu}_i^j = \begin{cases} \mu_i^j & i \neq m \text{ or } m+1 \\ \frac{2n_m\mu_m^j + (n_{m+1}-n_m)\mu_{m+1}^j}{n_m+n_{m+1}} & i = m \\ \frac{(n_m-n_{m+1})\mu_m^j + 2n_{m+1}\mu_{m+1}^j}{n_m+n_{m+1}} & i = m+1. \end{cases} \tag{3.46}$$

Note that, for the above construction, we have

$$\tilde{\mu}_m^j - \tilde{\mu}_{m+1}^j = \mu_m^j - \mu_{m+1}^j, \quad \forall j \in \{1, 2, \ldots, J\}. \tag{3.47}$$

Therefore $\{\tilde{\mu}_i^j\}$ also satisfies Ω_m, and so forms a valid codebook corresponding to partition $\{n_i'\}$. Thus, it will be sufficient if we can show that $D > D'$. On the other hand, it is easy to verify that, for all j,

$$n_{m+1}(\tilde{\mu}_m^j)^2 + n_m(\tilde{\mu}_{m+1}^j)^2 = n_m(\mu_m^j)^2 + n_{m+1}(\mu_{m+1}^j)^2.$$

Hence,

$$\sum_{i=1}^{K} n_i'(\tilde{\mu}_i^j)^2 = \sum_{i=1}^{K} n_i(\mu_i^j)^2 \ , \quad \forall j \tag{3.48}$$

Now consider the difference between D and D'

$$\begin{aligned}
\Delta &= n(D - D') \\
&= E\left[\min_j \sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i} \left(\eta_\ell - \mu_i^j\right)^2 - \min_j \sum_{i=1}^{K} \sum_{\ell \in \mathcal{I}_i'} \left(\eta_\ell - \tilde{\mu}_i^j\right)^2\right] \\
&\geq E\left[\min_j \left\{\sum_{i=1}^{K} \left(n_i(\mu_i^j)^2 - 2\mu_i^j \sum_{\ell \in \mathcal{I}_i} \eta_\ell\right)\right.\right. \\
&\qquad \left.\left. - \sum_{i=1}^{K} \left(n_i'(\tilde{\mu}_i^j)^2 - 2\tilde{\mu}_i^j \sum_{\ell \in \mathcal{I}_i'} \eta_\ell\right)\right\}\right], \qquad (3.49) \\
&= 2E\left[\min_j \left\{\tilde{\mu}_m^j \sum_{\ell=L+1}^{L+r} \eta_\ell + \tilde{\mu}_{m+1}^j \sum_{\ell=L+r+1}^{L+r+q} \eta_\ell\right.\right. \\
&\qquad \left.\left. - \mu_m^j \sum_{\ell=L+1}^{L+q} \eta_\ell - \mu_{m+1}^j \sum_{\ell=L+q+1}^{L+q+r} \eta_\ell\right\}\right], \qquad (3.50)
\end{aligned}$$

where to obtain (3.49), we used the fact that $\min\{f - g\} \leq \min f - \min g$, for arbitrary functions f, g; (3.50) follows from (3.48) in which $q = n_m$, $r = n_{m+1}$, and $L = n_1 + n_2 + \cdots + n_{m-1}$. Now using the formulae of $\tilde{\mu}_m^j$ and $\tilde{\mu}_{m+1}^j$ in (3.46), we

obtain

$$\Delta \geq 2E\left[\min_j \left\{ \frac{(q-r)(\mu_m^j - \mu_{m+1}^j)}{q+r} \sum_{\ell=L+1}^{L+r} \eta_\ell \right.\right.$$
$$- \frac{2r(\mu_m^j - \mu_{m+1}^j)}{q+r} \sum_{\ell=L+r+1}^{L+r+q} \eta_\ell$$
$$\left.\left. + \frac{(q-r)(\mu_m^j - \mu_{m+1}^j)}{q+r} \sum_{\ell=L+q+1}^{L+q+r} \eta_\ell \right\}\right] \tag{3.51}$$
$$= \frac{2r(q-r)}{q+r} E\left[\min_j \left\{(\mu_m^j - \mu_{m+1}^j)\zeta\right\}\right] \tag{3.52}$$
$$= \frac{2r(q-r)}{q+r}\left[\overline{\zeta}_+ \cdot \min_j \left\{\mu_m^j - \mu_{m+1}^j\right\} - \overline{\zeta}_- \cdot \max_j \left\{\mu_m^j - \mu_{m+1}^j\right\}\right]$$
$$\geq 0, \tag{3.53}$$

where ζ in (3.52) is the random variable specified in (3.9), and (3.53) follows from constraint Ω_m and that $q > r$. The nonnegativity of Δ has proved the proposition. $\square$

Chapter 4

Frame Permutation Quantization

This chapter presents the second approach of the thesis in which ordinary PSCs are used to quantize the coefficients of frame expansions of the source vector. Throughout this chapter, we just use small letters to denote random vectors, since most of the analysis focuses on their point-wise samples. Notations for ordinary PSCs given in Chapter 2 still apply for different dimensions, N and M, other than n.

4.1 Preview through $\mathbb{R}^2$ Geometry

Consider the quantization of $x \in \mathbb{R}^N$, where we restrict attention to $N = 2$ in this section but later allow any finite N. The uniform scalar quantization of x partitions $\mathbb{R}^N$ in a trivial way, as shown in Fig. 4-1(a). (An arbitrary segment of the plane is shown.) If over a domain of interest each component is divided into K intervals, a partition with K^N cells is obtained.

A way to increase the number of partition cells without increasing the scalar quantization resolution is to use a frame expansion. A conventional quantized frame expansion is obtained by scalar quantization of $y = Fx$, where $F \in \mathbb{R}^{M \times N}$ with $M \geq N$. Keeping the resolution K fixed, the partition now has K^M cells. An example with $M = 8$ is shown in Fig. 4-1(d). Each frame element φ_k (transpose of row of F) induces a *hyperplane wave partition* [52]: a partition formed by equally-spaced $(N-1)$-dimensional hyperplanes normal to φ_k. The overall partition has M

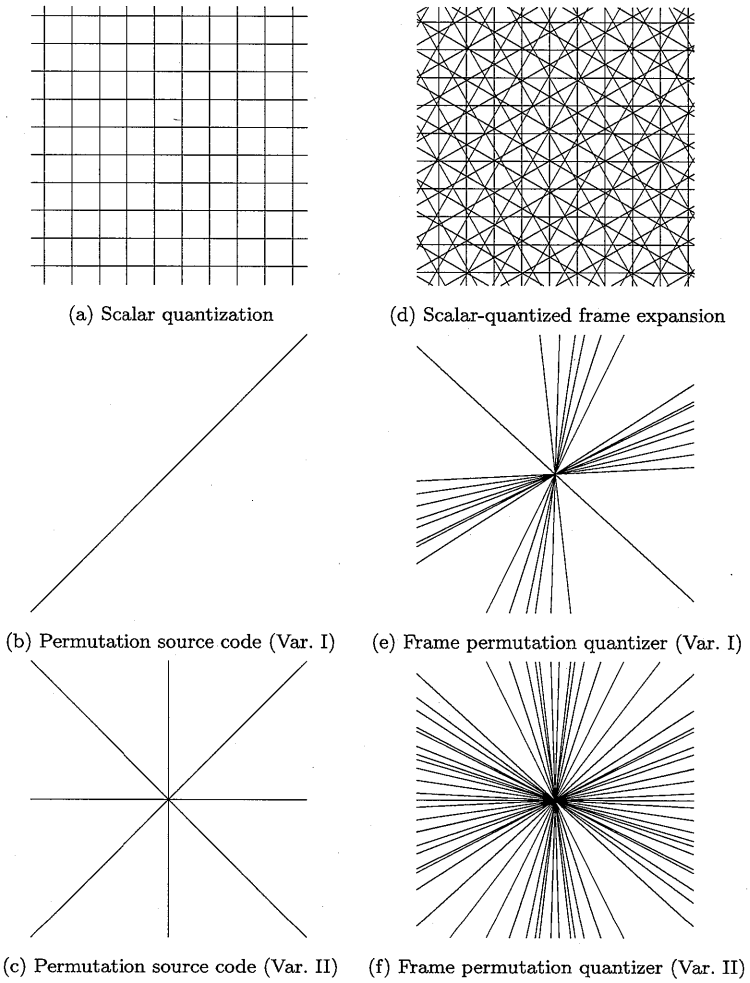


Figure 4-1: Partition diagrams for $x \in \mathbb{R}^2$. (a) Scalar quantization. (b) Permutation source code, Variant I. (c) Permutation source code, Variant II. (Both permutation source codes have $n_1 = n_2 = 1$.) (d) Scalar-quantized frame expansion with $M = 6$ coefficients (real harmonic tight frame). (e) Frame permutation quantizer, Variant I. (f) Frame permutation quantizer, Variant II. (Both frame permutation quantizers have $M = 6$, $m_1 = m_2 = \cdots = m_6 = 1$, and the same random frame.)

hyperplane waves and is spatially uniform. A spatial shift invariance can be ensured formally by the use of subtractively dithered quantizers [53].

A Variant I PSC represents x just by which permutation of the components of x puts the components in descending order. In other words, only whether $x_1 > x_2$ or whether $x_2 > x_1$ is specified.[1] The resulting partition is shown in Fig. 4-1(b). A Variant II PSC specifies (at most) the signs of the components of x_1 and x_2 and whether $|x_1| > |x_2|$ or $|x_2| > |x_1|$. The corresponding partitioning of the plane is shown in Fig. 4-1(c), with the vertical line coming from the sign of x_1, the horizontal line coming from the sign of x_2, and the diagonal lines from $|x_1| \gtrless |x_2|$.

While low-dimensional diagrams are often inadequate in explaining PSC, several key properties are illustrated. The partition cells are (unbounded) convex cones, giving special significance to the origin and a lack of spatial shift invariance. The unboundedness of cells implies that some additional knowledge, such as a bound on $\|x\|$ or a probabilistic distribution on x, is needed to compute good estimates. At first this may seem extremely different from ordinary scalar quantization or scalar-quantized frame expansions, but those techniques also require some prior knowledge to allow the quantizer outputs to be represented with finite numbers of bits. We also see that the dimension N determines the maximum number of cells ($N!$ for Variant I and $2^N N!$ for Variant II); there is no parameter analogous to scalar quantization step size that allows arbitrary control of the resolution.

To get a finer partition without changing the dimension N, we can again employ a frame expansion. With $y = Fx$ as before, PSC of y gives more relative orderings with which to represent x. If φ_j and φ_k are frame elements (transposes of rows of F) then $\langle x, \varphi_j \rangle \gtrless \langle x, \varphi_k \rangle$ is $\langle x, \varphi_j - \varphi_k \rangle \gtrless 0$ by linearity of the inner product, so every pair of frame elements can give a condition on x. An example of a partition obtained with Variant I and $M = 6$ is shown in Fig. 4-1(e). There are many more cells than in Fig. 4-1(b). Similarly, Fig. 4-1(f) shows a Variant II example. The cells are still (unbounded) convex cones. If additional information such as $\|x\|$ or an affine

[1]The boundary case of $x_1 = x_2$ can be handled arbitrarily in practice and safely ignored in the analysis.

subspace constraint (not passing through the origin) is known, x can be specified arbitrarily closely by increasing M.

4.2 Vector Quantization and PSCs Revisited

Recall that a vector quantizer is a mapping from an input $x \in \mathbb{R}^n$ to a *codeword* $\hat{x}$ from a finite *codebook* $\mathcal{C}$. Without loss of generality, a vector quantizer can be seen as a composition of an *encoder*

$$\alpha : \mathbb{R}^n \to \mathcal{I}$$

and a *decoder*

$$\beta : \mathcal{I} \to \mathbb{R}^n,$$

where $\mathcal{I}$ is a finite index set. The encoder partitions $\mathbb{R}^n$ into $|\mathcal{I}|$ regions or *cells* $\{\alpha^{-1}(i)\}_{i \in \mathcal{I}}$, and the decoder assigns a *reproduction value* to each cell. Examples of partitions are given in Fig. 4-1. For the quantizer to output R bits per component, we have $|\mathcal{I}| = 2^{nR}$.

For any codebook (i.e., any β), the encoder α that minimizes $\|x - \hat{x}\|^2$ maps x to the nearest element of the codebook. The partition is thus composed of convex cells. Since the cells are convex, reproduction values are optimally within the corresponding cells—whether to minimize expected distortion, maximum distortion, or any other reasonable cost function. To minimize maximum distortion, reproduction values should be at centers of cells; to minimize expected distortion, they should be at centroids of cells. Reproduction values being within corresponding cells is formalized as *consistency*:

Definition 4.1. *The reconstruction $\hat{x} = \beta(\alpha(x))$ is called a* consistent reconstruction of x *when $\alpha(x) = \alpha(\hat{x})$ (or equivalently $\beta(\alpha(\hat{x})) = \hat{x}$). The decoder β is called* consistent *when $\beta(\alpha(x))$ is a consistent reconstruction of x for all x.*

In practice, the pair (α, β) usually does not minimize any desired distortion criterion for a given codebook size because the optimal mappings are hard to design

and hard to implement [3]. The mappings are commonly designed subject to certain structural constraints, and β may not even be consistent for α [14, 16].

For both historical reasons and to match the conventional approach to vector quantization, PSCs were defined in terms of a codebook structure, and the codebook structure led to an encoding procedure. Note that we may now examine the partitions induced by PSCs separately from the particular codebooks for which they are nearest-neighbor partitions.

The partition induced by a Variant I PSC is completely determined by the integer partition $(n_1, n_2, \ldots, n_K)$. Specifically, the encoding mapping can index the permutation P that places the n_1 largest components of x in the first n_1 positions (without changing the order within those n_1 components), the n_2 next-largest components of x in the next n_2 positions, and so on; the μ_is are actually immaterial. This encoding is placing all source vectors x such that Px is n-descending in the same partition cell, defined as follows.

Definition 4.2. *Given an ordered integer partition $n = (n_1, n_2, \ldots, n_K)$ of N, a vector in $\mathbb{R}^N$ is called* n-descending *if its n_1 largest entries are in the first n_1 positions, its n_2 next-largest components are in the next n_2 positions, etc.*

The property of being n-descending is to be descending up the arbitrariness specified by the integer partition n.

Because this is nearest-neighbor encoding for *some* codebook, the partition cells must be convex. Furthermore, multiplying x by any nonnegative scalar does not affect the encoding, so the cells are convex cones. (This was discussed and illustrated in Section 4.1.) We develop a convenient representation for the partition in Section 4.4.

The situation is only slightly more complicated for Variant II PSCs. The partition is determined by the integer partition $(n_1, n_2, \ldots, n_K)$ and whether or not the signs of the smallest-magnitude components should be encoded (whether $\mu_K = 0$, in the codebook-centric view).

The PSC literature has mostly emphasized the design of PSCs for sources with i.i.d. components. But as developed in Section 4.4, the simple structured encoding of

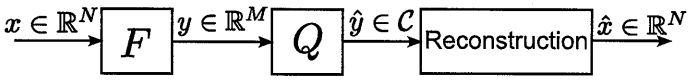


Figure 4-2: Block diagram of reconstruction from quantized frame expansion.

PSCs could be combined with unconventional decoding techniques for other sources. The possible suitability of PSCs for sources with unknown or time varying statistics had been previously observed [28].

4.3 Reconstruction from Frame Expansions

A central use of frames is to formalize the reconstruction of $x \in \mathbb{R}^N$ from the frame expansion $y_k = \langle x, \phi_k \rangle$, $k = 1, 2, \ldots, M$, or estimation of x from degraded versions of the frame expansion. Using the analysis frame operator we have $y = Fx$, and (2.12) implies the existence of at least one linear *synthesis operator* G such that $GF = I_N$. A frame with analysis frame operator G^* is then said to be *dual* to Φ.

The frame condition (2.12) also implies that F^*F is invertible, so the Moore–Penrose inverse (pseudo-inverse) of the frame operator

$$F^\dagger = (F^*F)^{-1}F^*$$

exists and is a valid synthesis operator. Using the pseudo-inverse for reconstruction has several important properties including an optimality for mean-squared error (MSE) under assumptions of uncorrelated zero-mean additive noise and linear synthesis [13, Sect. 3.2]. This follows from the fact that $FF^\dagger$ is an orthogonal projection from $\mathbb{R}^M$ onto the subspace $F(\mathbb{R}^N)$, the range of F. Because of this special role, reconstruction using $F^\dagger$ is called *canonical reconstruction* and the corresponding frame is called the *canonical dual*. In this thesis, we use the term *linear reconstruction* for reconstruction using an arbitrary linear operator.

When y is quantized to $\hat{y}$ (see Figure 4-2), it is possible for the quantization noise $\hat{y} - y$ to have mean zero and uncorrelated components; this occurs with subtractive dithered quantization [53] or under certain asymptotics [54]. In this case,

the optimality of canonical reconstruction holds. However, it should be noted that even with these restrictions, canonical reconstruction is optimal *only amongst linear reconstructions.*

When nonlinear reconstruction is allowed, quantization noise may behave fundamentally differently than other additive noise. The key is that a quantized value gives hard constraints that can be exploited in reconstruction. For example, suppose that $\hat{y}$ is obtained from y by rounding each element to the nearest multiple of a quantization step size Δ. Then knowledge of $\hat{y}_m$ is equivalent to knowing

$$y_k \in [\hat{y}_k - \tfrac{1}{2}\Delta,\ \hat{y}_k + \tfrac{1}{2}\Delta]. \tag{4.1}$$

Geometrically, $\langle x, \phi_k \rangle = \hat{y}_k - \frac{1}{2}\Delta$ and $\langle x, \phi_k \rangle = \hat{y}_k + \frac{1}{2}\Delta$ are hyperplanes perpendicular to ϕ_k, and (4.1) expresses that x must lie between these hyperplanes; this may be visualized as one pair of parallel lines in Fig. 4-1(d). Using the upper and lower bounds on all M components of y, the constraints on x imposed by $\hat{y}$ are readily expressed as [16]

$$\begin{bmatrix} F \\ -F \end{bmatrix} x \leq \begin{bmatrix} \frac{1}{2}\Delta + \hat{y} \\ \frac{1}{2}\Delta - \hat{y} \end{bmatrix}, \tag{4.2}$$

where the inequalities are elementwise. For example, all $2M$ constraints specify a single cell in Fig. 4-1(d). This formulation inspires Algorithm 4.1, which is a modification of [16, Table I] using the principle of maximizing slackness of inequalities that was also implemented in [18]. We will find analogues to (4.2) and Algorithm 4.1 for FPQ.

4.4 Frame Permutation Quantization

With background material on permutation source codes and finite frames in place, we are now prepared to formally introduce frame permutation quantization. FPQ is simply PSC applied to a frame expansion.

Algorithm 4.1 Consistent Reconstruction from Scalar-Quantized Frame Expansion

Inputs: Analysis frame operator F, quantization step size Δ, and quantized frame expansion y

Output: Estimate $\hat{x}$ consistent with $\hat{y}$ and as far from the partition boundaries as possible

1. Let $A = \begin{bmatrix} F & 1_{M\times 1} \\ -F & 1_{M\times 1} \end{bmatrix}$ and $b = \begin{bmatrix} \hat{y} \\ -\hat{y} \end{bmatrix}$.
   (Consistency as in (4.2) is expressed as $A \begin{bmatrix} x \\ 0 \end{bmatrix} \leq b$.)
2. Let $c = \begin{bmatrix} 0_{N\times 1} \\ -1 \end{bmatrix}$.
3. Use a linear programming method to minimize $c^T \begin{bmatrix} x \\ \delta \end{bmatrix}$ subject to $A \begin{bmatrix} x \\ \delta \end{bmatrix} \leq b$. Return the first N components of the minimizer as $\hat{x}$.

4.4.1 Encoder Definition

Definition 4.3. *A* frame permutation quantizer *with analysis frame* $F \in \mathbb{R}^{M\times N}$, *integer partition* $m = (m_1, m_2, \ldots, m_K)$ *and initial codeword* $\hat{y}_{\text{init}}$ *associated with* m *encodes* $x \in \mathbb{R}^N$ *by applying a permutation source code with integer partition* m *and initial codeword* $\hat{y}_{\text{init}}$ *to* Fx.

We sometimes use the triple $(F, m, \hat{y}_{\text{init}})$ to refer to such an FPQ. The two variants of PSCs yield two variants of FPQ.

For Variant I, the result of the encoding can be expressed as a permutation P from the permutation matrices of size M. The permutation is such that PFx is m-descending. For uniqueness in the representation P chosen from the set of permutation matrices, we can specify that the first m_1 components of Py are kept in the same order as they appeared in y, the next m_2 components of Py are kept in the same order as they appeared in y, etc. Then P is in a subset $\mathcal{G}(m)$ of the $M \times M$ permutation matrices and

$$|\mathcal{G}(m)| = \frac{M!}{m_1!\, m_2! \cdots m_K!}. \tag{4.3}$$

For Variant II, we will sidestep the differences between the $\mu_K = 0$ and $\mu_K \neq 0$ cases in Section 2.1.2 by specifying that the signs of the m_K smallest-magnitude components of Fx are not encoded and $m_K = 0$ is allowed. Now the result of encoding can be expressed similarly as $P \in \mathcal{G}(m)$ along with a diagonal matrix V with ± 1 on its diagonal. These matrices are selected such that the elementwise absolute values of $VPFx$ are m-descending and also the first $M - m_K$ entries of $VPFx$ are positive. The last m_K diagonal entries of V do not affect the encoding and are set to $+1$. Thus V is in a subset $\mathcal{Q}(m)$ of the $M \times M$ sign-changing matrices and

$$|\mathcal{Q}(m)| = 2^{M-m_K}. \tag{4.4}$$

The sizes of the sets $\mathcal{G}(m)$ and $\mathcal{G}(m) \times \mathcal{Q}(m)$ are analogous to the codebook sizes in (2.3), and the per-component rates of FPQ are thus defined as

$$R_{\mathrm{I}} = N^{-1} \log_2 \frac{M!}{m_1! \, m_2! \cdots m_K!}, \qquad \text{for Variant I,} \tag{4.5a}$$

and

$$R_{\mathrm{II}} = N^{-1} \left(M - m_K + \log_2 \frac{M!}{m_1! \, m_2! \cdots m_K!} \right), \qquad \text{for Variant II.} \tag{4.5b}$$

4.4.2 Expressing Consistency Constraints

Suppose FPQ encoding of $x \in \mathbb{R}^N$ with frame $F \in \mathbb{R}^{M \times N}$, integer partition $m = (m_1, m_2, \ldots, m_K)$, and initial codeword $\hat{y}_{\text{init}}$ associated with m results in permutation $P \in \mathcal{G}(m)$ (and, in the case of Variant II, $V \in \mathcal{Q}(m)$) as described in Section 4.4.1. We would like to express constraints on x that are specified by $(F, m, \hat{y}_{\text{init}}, P)$ (or $(F, m, \hat{y}_{\text{init}}, P, V)$). This will provide an explanation of the partitions induced by FPQ and lead to reconstruction algorithms in Section 4.4.3.

Knowing that a vector is m-descending is a specification of many inequalities. Recall the definitions of the index sets generated by an integer partition given in (2.8) and (2.9), and use the same notation with n_ks replaced by m_ks. Then z being

m-descending implies that for any $i < j$,

$$z_k \geq z_\ell \qquad \text{for every } k \in \mathcal{I}_i \text{ and } \ell \in \mathcal{I}_j.$$

By transitivity, considering every $i < j$ gives redundant inequalities. Taking only $j = i + 1$, we obtain a full description

$$z_k \geq z_\ell \qquad \text{for every } k \in \mathcal{I}_i \text{ and } \ell \in \mathcal{I}_{i+1} \text{ with } i = 1, 2, \ldots, K-1. \tag{4.6}$$

For one fixed (i, ℓ) pair, (4.6) gives $|\mathcal{I}_i| = m_i$ inequalities, one for each $k \in \mathcal{I}_i$. These inequalities can be gathered into an elementwise matrix inequality as

$$\begin{bmatrix} 0_{m_i \times M_{i-1}} & I_{m_i} & 0_{m_i \times (M - M_i)} \end{bmatrix} z \;\geq\; \begin{bmatrix} 0_{m_i \times (\ell-1)} & 1_{m_i \times 1} & 0_{m_i \times (M-\ell)} \end{bmatrix} z$$

where $M_k = m_1 + m_2 + \cdots + m_k$, or $D^{(m)}_{i,\ell} z \geq 0_{m_i \times 1}$ where

$$D^{(m)}_{i,\ell} = \begin{bmatrix} 0_{m_i \times M_{i-1}} & I_{m_i} & 0_{m_i \times (\ell - M_i - 1)} & -1_{m_i \times 1} & 0_{m_i \times M - \ell} \end{bmatrix} \tag{4.7a}$$

is an $m_i \times M$ differencing matrix. Allowing ℓ to vary across $\mathcal{I}_{i+1}$, we define the $m_i m_{i+1} \times M$ matrix

$$D^{(m)}_i = \begin{bmatrix} D^{(m)}_{i,M_i+1} \\ D^{(m)}_{i,M_i+2} \\ \vdots \\ D^{(m)}_{i,M_i+m_{i+1}} \end{bmatrix} \tag{4.7b}$$

and express all of (4.6) for one fixed i as $D^{(m)}_i z \geq 0_{m_i m_{i+1} \times 1}$.

Continuing our recursion, it only remains to gather the inequalities (4.6) across

$i \in \{1, 2, \ldots, K-1\}$. Let

$$D^{(m)} = \begin{bmatrix} D_1^{(m)} \\ D_2^{(m)} \\ \vdots \\ D_{K-1}^{(m)} \end{bmatrix}, \tag{4.7c}$$

which has

$$L(m) = \sum_{i=1}^{K-1} m_i m_{i+1} \tag{4.8}$$

rows. The property of z being m-descending can be expressed as $D^{(m)}z \geq 0_{L(m)\times M}$. The following example illustrates the form of $D^{(m)}$:

$$D^{((2,3,2))} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}.$$

Notice the important property that each row of $D^{(m)}$ has one 1 entry and one -1 entry with the remaining entries 0. This will be exploited in Section 4.5.

Now we can apply these representations to FPQ.

Variant I: In this case, we know PFx is m-descending. Consistency is thus simply expressed as

$$D^{(m)}PFx \;\geq\; 0. \tag{4.9}$$

Variant II: The second variant has an m-descending property after V has made the signs of the significant frame coefficients (all but last m_K) positive: $D^{(m)}VPFx \geq 0$. In addition, we have the nonnegativity of all of the first $M - m_K$ sorted and sign-changed coefficients. To specify

$$\begin{bmatrix} I_{M-m_K} & 0_{(M-m_K)\times m_K} \end{bmatrix} VPFx \;\geq\; 0_{(M-m_K)\times 1}$$

is redundant with what is expressed with the m-descending property. The added constraints can be applied only to the entries of $VPFx$ with indexes in $\mathcal{I}_{K-1}$ because all the earlier entries are already ensured to be larger. We thus express consistency as

$$\underbrace{\begin{bmatrix} & D^{(m)} & \\ 0_{m_{K-1}\times M_{K-1}} & I_{m_{K-1}} & 0_{m_{K-1}\times m_K} \end{bmatrix}}_{\widetilde{D}^{(m)}} VPFx \;\geq\; 0. \tag{4.10}$$

4.4.3 Consistent Reconstruction Algorithms

The constraints (4.9) and (4.10) both specify unbounded sets, as discussed previously and illustrated in Fig. 4-1(e) and (f). To able to decode FPQs in analogy to Algorithm 4.1, we require some additional constraints. We develop two examples: a source x bounded to $[-\frac{1}{2}, \frac{1}{2}]^N$ (e.g., having an i.i.d. uniform distribution over $[-\frac{1}{2}, \frac{1}{2}]$) or having an i.i.d. standard Gaussian distribution. For the remainder of this section, we consider only Variant I because adjusting for Variant II using (4.10) is easy.

Source Bounded to $[-\frac{1}{2}, \frac{1}{2}]^N$

To impose (4.9) along with $x \in [-\frac{1}{2}, \frac{1}{2}]^N$ is trivial because $x \in [-\frac{1}{2}, \frac{1}{2}]^N$ is decomposable into $2N$ inequality constraints:

$$\begin{bmatrix} I_N \\ -I_N \end{bmatrix} x \;\leq\; \frac{1}{2} \begin{bmatrix} 1_{N\times 1} \\ -1_{N\times 1} \end{bmatrix}.$$

Algorithm 4.2 Reconstruction of Source on $[-\frac{1}{2}, \frac{1}{2}]^N$ for Variant I Frame Permutation Quantization

Inputs: Analysis frame operator F, integer partition m, and FPQ encoding P
Output: Estimate $\hat{x}$ consistent with (F, m, P) and as far from the partition boundaries as possible

1. Let $A = \begin{bmatrix} -D^{(m)}PF & 1_{L(m)\times 1} \\ -I_N & 1_{N\times 1} \\ I_N & 1_{N\times 1} \end{bmatrix}$ and $b = \frac{1}{2}\begin{bmatrix} 0_{L(m)\times 1} \\ 1_{2N\times 1} \end{bmatrix}$, where $D^{(m)}$ is defined in (4.7) and $L(m)$ is defined in (4.8). (Consistency with (4.9) and $x \in [-\frac{1}{2}, \frac{1}{2}]^N$ is expressed as $A\begin{bmatrix} x \\ 0 \end{bmatrix} \leq b$.)
2. Let $c = \begin{bmatrix} 0_{N\times 1} \\ -1 \end{bmatrix}$.
3. Use a linear programming method to minimize $c^T\begin{bmatrix} x \\ \delta \end{bmatrix}$ subject to $A\begin{bmatrix} x \\ \delta \end{bmatrix} \leq b$. Return the first N components of the minimizer as $\hat{x}$.

A linear programming formulation will return some corner of the consistent set, depending on the choice of cost function. The unknown vector x can be augmented with a variable δ which represents the slackness of the inequality constraint with the least slack. Maximizing δ moves the solution away from the boundary of the consistent set (partition cell) as much as possible. Reconstruction using this principle is outlined in Algorithm 4.2.

If the source x is random and the distribution $p(x)$ is known, then one could optimize some criterion explicitly. For example, one could maximize $p(x)$ over the consistent set or compute the centroid of the consistent set with respect to $p(x)$. This would improve upon reconstructions computed with Algorithm 4.2 but presumably increase complexity greatly.

Source with i.i.d. Standard Gaussian Distribution

Suppose x has i.i.d. Gaussian components with mean zero and unit variance. Since the source support is unbounded, something beyond consistency must be used in reconstruction. Here we use a quadratic program to find a good bounded, consistent

estimate and combine this with the average value of $\|x\|$.

The problem with using (4.9) combined with maximization of minimum slackness alone (without any additional boundedness constraints) is that for any purported solution, multiplying by a scalar larger than 1 will increase the slackness of all the constraints. Thus, any solution technique will naturally and correctly have $\|\hat{x}\| \to \infty$. Actually, because the partition cells are convex cones, we should not hope to recover the radial component of x from the partition. Instead, we should only hope to recover a good estimate of $x/\|x\|$.

It estimates the angular component $x/\|x\|$ from the partition, it would be convenient to maximize minimum slackness while also imposing a constraint of $\|\hat{x}\| = 1$. Unfortunately, this is a nonconvex constraint. It can be replaced by $\|\hat{x}\| \leq 1$ because slackness is proportional to $\|\hat{x}\|$. This suggests the optimization

$$\text{maximize } \delta \text{ subject to } \|x\| \leq 1 \text{ and } D^{(m)}PFx \;\geq\; \delta 1_{L(m)\times 1}.$$

Denoting the x at the optimum by $\hat{x}_{\text{ang}}$, we still need to choose the radial component, or length, of $\hat{x}$.

For the $\mathcal{N}(0, I_N)$ source, the mean length is [25]

$$E[\|x\|] = \frac{\sqrt{2\pi}}{\beta(N/2, 1/2)} \approx \sqrt{N - 1/2}.$$

We can combine this with $\hat{x}_{\text{ang}}$ to obtain a reconstruction $\hat{x}$.

We use a slightly different formulation to have a quadratic program in standard form. We combine the radial component constraint with the goal of maximizing slackness to obtain

$$\text{minimize } \tfrac{1}{2}x^T x - \lambda\delta \text{ subject to } -D^{(m)}PFx \;\leq\; -\delta 1_{L(m)\times 1},$$

where λ trades of slackness against the radial component of x. Since the radial component will be replaced with its expectation, the choice of λ is immaterial; it is set to 1 in Algorithm 4.3.

Algorithm 4.3 Reconstruction of $\mathcal{N}(0, I_N)$ Source for Variant I Frame Permutation Quantization

Inputs: Analysis frame operator F, integer partition m, and FPQ encoding P
Output: Estimate $\hat{x}$ consistent with (F, m, P) and as far from the partition boundaries as possible while keeping $\hat{x} = E[\|x\|]$

1. Let $A = \begin{bmatrix} -D^{(m)}PF & 1_{L(m)\times 1} \end{bmatrix}$ and $b = 0_{L(m)\times 1}$, where $D^{(m)}$ is defined in (4.7) and $L(m)$ is defined in (4.8). (Consistency with (4.9) is expressed as $A \begin{bmatrix} x \\ 0 \end{bmatrix} \leq b$.)
2. Let $c = \begin{bmatrix} 0_{N\times 1} \\ -1 \end{bmatrix}$ and $H = \begin{bmatrix} I_N & 0_{N\times 1} \\ 0_{1\times N} & 0 \end{bmatrix}$.
3. Use a quadratic programming method to minimize $\frac{1}{2}\begin{bmatrix} x \\ \delta \end{bmatrix}^T H \begin{bmatrix} x \\ \delta \end{bmatrix} + c^T \begin{bmatrix} x \\ \delta \end{bmatrix}$ subject to $A \begin{bmatrix} x \\ \delta \end{bmatrix} \leq b$. Denote the first N components of the minimizer as $\hat{x}_{\text{ang}}$.
4. Return $(\sqrt{2\pi}/\beta(N/2, 1/2))\, \hat{x}_{\text{ang}}$.

4.5 Conditions on the Choice of Frame

In this section, we provide necessary and sufficient conditions so that a linear reconstruction is also consistent. We first consider a general linear reconstruction, $\hat{x} = R\hat{y}$, where R is some $N \times M$ matrix and $\hat{y}$ is a decoding of the PSC of y. We then restrict attention to canonical reconstruction, where $R = F^{\dagger}$. For each case, we describe all possible choices of a "good" frame F, in the sense of the consistency of the linear reconstruction.

4.5.1 Arbitrary Linear Reconstruction

We begin by introducing some useful terminology.

Definition 4.4. *A matrix is called* column-constant *when each column of the matrix is a constant. The set of all $M \times M$ column-constant matrices is denoted $\mathcal{J}$.*

We now give our main results for arbitrary linear reconstruction combined with FPQ decoding of an estimate of y.

Theorem 4.5. *Suppose $A = FR = aI_M + J$ for some $a \geq 0$ and $J \in \mathcal{J}$. Then the linear reconstruction $\hat{x} = R\hat{y}$ is consistent with Variant I FPQ encoding using frame F, an arbitrary integer partition and an arbitrary Variant I initial codeword associated with it.*

Proof. We start the proof by pointing out two special properties of any matrix $J \in \mathcal{J}$:

(P1) $PJP^{-1} \in \mathcal{J}$ for any permutation matrix P; and

(P2) $D^{(m)}J = 0_{L(m)\times 1}$ for any integer partition m.

(P1) follows from the fact that neither left multiplying by P nor right multiplying by P^{-1} disturbs column-constancy. (P2) is true because each row of $D^{(m)}$ has zero entries except for one 1 and one -1.

Suppose $m = (m_1, m_2, \ldots, m_K)$ is an arbitrary integer partition of M and $\hat{y}_{\text{init}}$ is an arbitrary Variant I initial codeword associated with m. Let P be the Variant I FPQ encoding of x using $(F, m, \hat{y}_{\text{init}})$. We would like to check that $\hat{x} = R\hat{y}$ is consistent with the encoding P. This is verified through the following computation:

$$\begin{aligned}
D^{(m)}PF\hat{x} &= D^{(m)}PFR\hat{y} \\
&= D^{(m)}PFRP^{-1}\hat{y}_{\text{init}} && (4.11)\\
&= D^{(m)}PAP^{-1}\hat{y}_{\text{init}} \\
&= D^{(m)}P\left(aI_M + J\right)P^{-1}\hat{y}_{\text{init}} && (4.12)\\
&= a\,D^{(m)}\,\hat{y}_{\text{init}} + D^{(m)}\hat{J}\hat{y}_{\text{init}} \qquad \text{for some } \hat{J} \in \mathcal{J} && (4.13)\\
&= a\,D^{(m)}\,\hat{y}_{\text{init}} && (4.14)\\
&\geq 0_{L(m)\times 1}, && (4.15)
\end{aligned}$$

where (4.11) uses the conventional decoding of a PSC; (4.12) follows from the hypothesis of the theorem on A; (4.13) follows from (P1); (4.14) follows from (P2); and (4.15) follows from the definition of Variant I initial codewords associated with m, and the nonnegativity of a. This completes the proof. □

The key point of the proof of Theorem 4.5 is showing that the inequality

$$D^{(m)}PAP^{-1}\hat{y}_{\text{init}} \geq 0, \tag{4.16}$$

where $A = FR$, holds for every integer partition m and every initial codeword $\hat{y}_{\text{init}}$ associated with it. It turns out that the form of matrix A given in Theorem 4.5 is the unique form that guarantees that (4.16) holds for every pair $(m, \hat{y}_{\text{init}})$. In other words, the condition on A that is sufficient for every integer partition m and every initial codeword $\hat{y}_{\text{init}}$ associated with it is also a necessary for consistency for every pair $(m, \hat{y}_{\text{init}})$.

Theorem 4.6. *Consider Variant I FPQ using frame F with $M \geq 3$. If linear reconstruction $\hat{x} = R\hat{y}$ is consistent with every integer partition and every Variant I initial codeword associated with it, then matrix $A = FR$ must be of the form $aI_M + J$, where $a \geq 0$ and $J \in \mathcal{J}$.*

Proof. See Section 4.A. □

Similar necessary and sufficient conditions can be derived for linear reconstruction of Variant II FPQs. Since the partition cell associated with a codeword of a Variant II FPQ is much smaller than that of the corresponding Variant I FPQ, we expect the condition for a linear reconstruction to be consistent to be more restrictive than that given in Theorem 4.5 and Theorem 4.6. The following two theorems show that this is indeed the case.

Theorem 4.7. *Suppose $A = FR = aI_M$ for some $a \geq 0$ and $M = N$. Then the linear reconstruction $\hat{x} = R\hat{y}$ is consistent with Variant II FPQ encoding using frame F, an arbitrary integer partition, and an arbitrary Variant II initial codeword associated with it.*

Proof. Suppose that $m = (m_1, m_2, \ldots, m_K)$ is an arbitrary integer partition of M, and $\hat{y}_{\text{init}}$ is an arbitrary Variant II initial codeword associated with it. Let (P, V) be the Variant II FPQ encoding of x using $(F, m, \hat{y}_{\text{init}})$. We would like to check that

$\hat{x} = R\hat{y}$ is consistent with the encoding (P, V). This is verified through the following computation:

$$\begin{aligned}
\widetilde{D}^{(m)}VPF\hat{x} &= \widetilde{D}^{(m)}VPFR\hat{y} \\
&= \widetilde{D}^{(m)}VPFRP^{-1}V^{-1}\hat{y}_{\text{init}} \qquad (4.17) \\
&= \widetilde{D}^{(m)}VPAP^{-1}V^{-1}\hat{y}_{\text{init}} \\
&= \widetilde{D}^{(m)}VPaI_MP^{-1}V^{-1}\hat{y}_{\text{init}} \qquad (4.18) \\
&= a\,\widetilde{D}^{(m)}\,\hat{y}_{\text{init}} \\
&\geq 0_{L(m)\times 1}, \qquad (4.19)
\end{aligned}$$

where (4.17) uses the conventional decoding of a PSC; (4.18) follows from the hypothesis of the theorem on A; and (4.19) follows from the definition Variant II initial codewords associated with m, and the nonnegativity of a. This completes the proof. □

Theorem 4.8. *Consider Variant II FPQ using frame F with $M \geq 3$. If linear reconstruction $\hat{x} = R\hat{y}$ is consistent with every integer partition and every Variant II initial codeword associated with it, then matrix $A = FR$ must be of the form aI_M, where $a \geq 0$ and $M = N$.*

Proof. See Section 4.B. □

The two theorems above show that, if we insist on linear consistent reconstructions for Variant II FPQs, the frame must degenerate into a basis. For nonlinear consistent reconstructions, we could use algorithms analogous to those presented in Section 4.4.3 for an arbitrary frame that is not necessarily a basis.

4.5.2 Canonical Reconstruction

We now restrict the linear reconstruction to use the canonical dual; i.e., R is restricted to be the pseudo-inverse $F^{\dagger} = (F^*F)^{-1}F^*$. The following corollary characterizes the non-trivial frames for which canonical reconstructions are consistent.

Corollary 4.9. *Consider Variant I FPQ using frame F with $M > N$ and $M \geq 3$. For canonical reconstruction to be consistent with every integer partition and every Variant I initial codeword associated with it, it is necessary and sufficient to have $M = N + 1$ and $A = FF^\dagger = I_M - \frac{1}{M} J_M$, where J_M is the $M \times M$ all-1s matrix.*

Proof. Sufficiency follows immediately from Theorem 4.5. The necessary condition of Theorem 4.6 leaves some flexibility in the choice of F that we must eliminate using $R = F^\dagger$.

From Theorem 4.6, it is necessary to have $A = FF^\dagger = aI_M + J$ for some $a \geq 0$ and $J \in \mathcal{J}$. Now, by noting that A is an orthogonal projection operator, we can impose the self-adjointness on matrix A to get

$$aI_M + J = (aI_M + J)^* = aI_M + J^*. \tag{4.20}$$

Thus, $J = J^*$, and it follows that $J = bJ_M$, for some constant b. On the other hand, the idempotence of A gives

$$\begin{aligned} aI_M + bJ_M &= (aI_M + bJ_M)^2 \\ &= a^2 I_M + (2ab + b^2 M) J_M. \end{aligned} \tag{4.21}$$

Since $M > N$, b must be different from zero. Equating the two sides of (4.21) yields $a = 1$ and $b = -1/M$. Hence,

$$A = I_M - \frac{1}{M} J_M. \tag{4.22}$$

From (4.22), note that $\mathrm{tr}(A) = M - 1$. But also,

$$\mathrm{tr}(A) = \mathrm{tr}\left(F(F^*F)^{-1}F^*\right) = \mathrm{tr}\left((F^*F)^{-1}F^*F\right) = \mathrm{tr}(I_N) = N.$$

Thus $M = N + 1$. □

We continue to add more constraints to frame F. Tightness and equal-norm are amongst common requirements in frame design [11]. By imposing tightness and unit

norm on our analysis frame, we can progress a bit further from Corollary 4.9 to derive the form of FF^*.

Corollary 4.10. *Consider Variant I FPQ using unit-norm tight frame F with $M > N$ and $M \geq 3$. For canonical reconstruction to be consistent for every integer partition and every Variant I initial codeword associated with it, it is necessary and sufficient to have $M = N + 1$ and*

$$FF^* = \begin{bmatrix} 1 & -\frac{1}{N} & \cdots & -\frac{1}{N} \\ -\frac{1}{N} & 1 & \cdots & -\frac{1}{N} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{N} & -\frac{1}{N} & \cdots & 1 \end{bmatrix}. \tag{4.23}$$

Proof. Corollary 4.9 asserts that $M = N + 1$ and

$$F(F^*F)^{-1}F^* = \begin{bmatrix} \frac{N}{M} & -\frac{1}{M} & \cdots & -\frac{1}{M} \\ -\frac{1}{M} & \frac{N}{M} & \cdots & -\frac{1}{M} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{M} & -\frac{1}{M} & \cdots & \frac{N}{M} \end{bmatrix}. \tag{4.24}$$

On the other hand, the tightness of a unit-norm frame F implies

$$(F^*F)^{-1} = \frac{N}{M} I_N. \tag{4.25}$$

Combining (4.24) with (4.25), we get (4.23). □

Recall that a UNTF that satisfies (4.23) is a restricted ETF. Therefore Corollary 4.10 together with Proposition 2.6 gives us a complete characterization of UNTFs that are "good" in the sense of canonical reconstruction being consistent.

Corollary 4.11. *Consider Variant I FPQ using unit-norm tight frame F with $M > N$ and $M \geq 3$. For canonical reconstruction to be consistent for every integer partition and every Variant I initial codeword associated with it, it is necessary and sufficient for F to be the modulated HTFs or their Type I or Type II equivalents.*

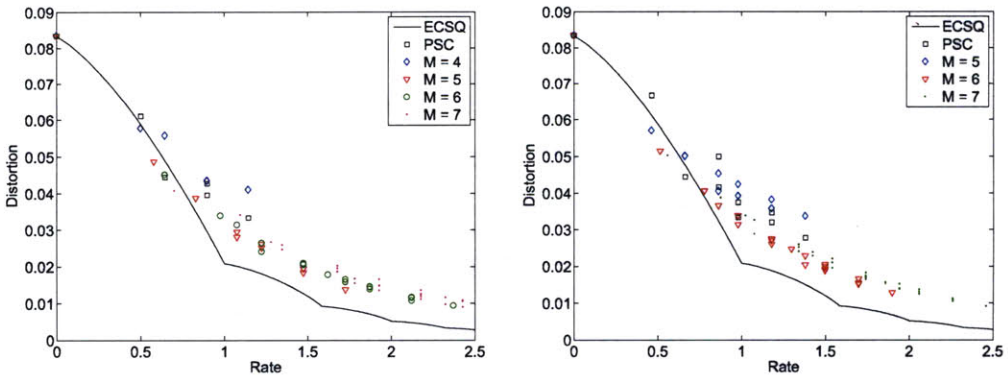
4.6 Simulations

In this section, we provide simulations to demonstrate some properties of FPQ and to demonstrate that FPQ can give attractive performance for certain combinations of signal dimension and rate. All FPQ simulations use modulated harmonic tight frames and are based on implementations of Algorithms 4.2 and 4.3 using MATLAB, with linear programming and quadratic programming provided by the Optimization Toolbox. For every data point shown, the distortion represents a sample mean estimate of $N^{-1}E[\|x-\hat{x}\|^2]$ over at least 10^6 trials. Testing was done with exhaustive enumeration of the relevant integer partitions. This makes the complexity of simulation high, and thus experiments are only shown for small N and M. Recall the encoding complexity of FPQ is low, $O(M \log M)$ operations. The decoding complexity is polynomial in M for either of the algorithms presented explicitly, and in some applications it could be worthwhile to precompute the entire codebook at the decoder. Thus much larger values of N and M than used here may be practical.

4.6.1 Basic Experiments

Uniform source. Let x have i.i.d. components uniformly distributed on $[-\frac{1}{2}, \frac{1}{2}]$. Algorithm 4.2 is clearly well-suited to this source since the support of x is properly specified and reconstructions near the centers of cells is nearly optimal. Fig. 4-3 summarizes the performance of Variant I FPQ for several frames and an enormous number of integer partitions. Also shown are the performances of ordinary PSC and entropy-constrained scalar quantization.

Using $F = I_N$ makes FPQ reduce to ordinary PSC. We see that, consistent with results in [8], PSC is sometimes better than ECSQ. Next notice that FPQ is not identical to PSC when F is square but not the identity matrix. The modulated harmonic frame with $M = N$ provides an orthogonal matrix F. The set of rates obtained with $M = N$ is the same as PSC, but since the source is not rotationally-invariant, the partitions and hence distortions are not the same; the distortion is sometimes better and sometime worse. Increasing M gives more operating points—



(a) $N = 4$ (b) $N = 5$

Figure 4-3: Performance of Variant I FPQ on an i.i.d. uniform($[-\frac{1}{2}, \frac{1}{2}]$) source using modulated harmonic tight frames ranging in size from N to 7. Also shown are the performances of ordinary PSC (equivalent to FPQ with frame $F = I_N$), and entropy-constrained scalar quantization.

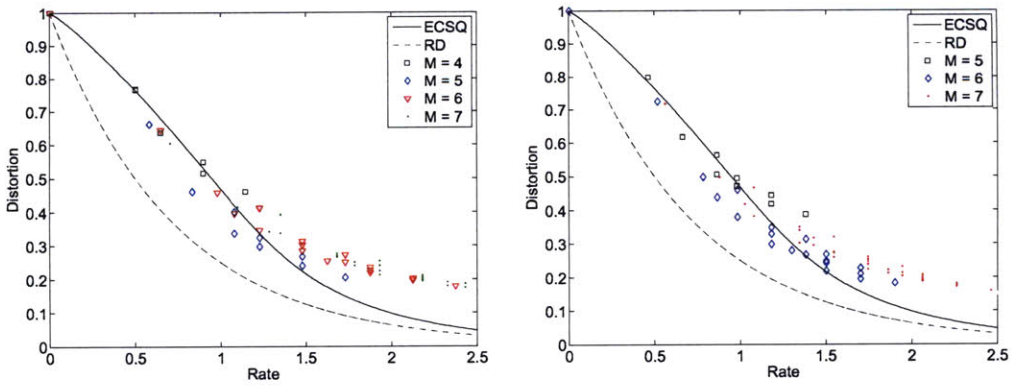
some of which are attractive—and a higher maximum rate. In particular, for both $N = 4$ and $N = 5$, it seems that $M = N + 1$ gives several operating points better than those obtainable with larger or smaller values of M.

Gaussian source. Let x have the $\mathcal{N}(0, I_N)$ distribution. Algorithm 4.3 is designed precisely for this source. Fig. 4-4 summarizes the performance of Variant I FPQ with decoding using Algorithm 4.3. Also shown are the performance of entropy-constrained scalar quantization and the distortion–rate bound.

We have not provided an explicit comparison to ordinary PSC because, due to rotational-invariance of the Gaussian source, FPQ with any orthonormal basis as the frame is identical to PSC. (The modulated harmonic tight frame with $M = N$ is an orthonormal basis.) The trends are similar to those for the uniform source: PSC and FPQ are sometimes better than ECSQ; increasing M gives more operating points and a higher maximum rate; and $M = N + 1$ seems especially attractive.

4.6.2 Variable-Rate Experiments and Discussion

We have posed FPQ as a fixed-rate coding technique. As mentioned in Section 2.1.2, symmetries will often make the outputs of a PSC equally likely, making variable-rate



(a) $N = 4$ (b) $N = 5$

Figure 4-4: Performance of Variant I FPQ on an i.i.d. $\mathcal{N}(0, 1)$ using modulated harmonic tight frames ranging in size from N to 7. Performance of PSC is not shown because it is equivalent to FPQ with $M = N$ for this source. Also plotted are the performance of entropy-constrained scalar quantization and the distortion–rate bound.

coding superfluous. This does not necessarily carry over to FPQ.

In Variant I FPQ with modulated HTFs, when $M > N+1$ the codewords are not only nonequiprobable, some cannot even occur. To see an example of this, consider the case of $(N, M) = (2, 4)$. Then

$$F = \begin{bmatrix} 1 & 0 \\ -\rho & -\rho \\ 0 & 1 \\ \rho & -\rho \end{bmatrix}, \qquad \text{where } \rho = 1/\sqrt{2}.$$

If we choose the integer partition $m = (2, 2)$, we might expect six distinct codewords that are equiprobable for a rotationally-invariant source. The permutation matrices consistent with this integer partition are

$$\left\{ \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \right\}.$$

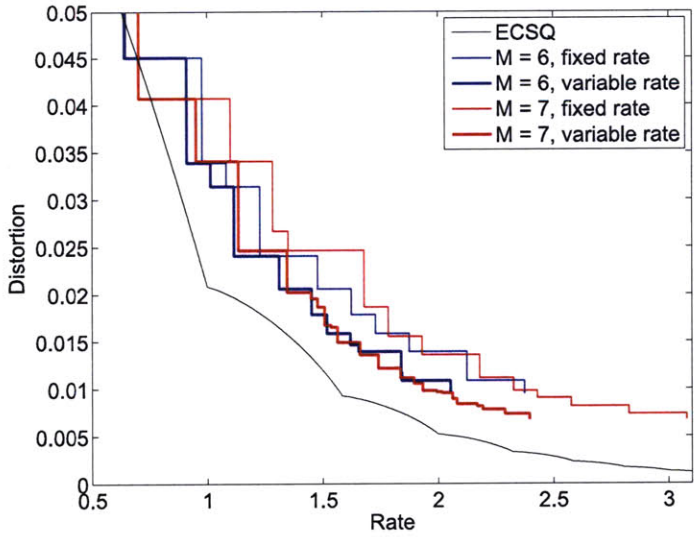
The first and fifth of these occur with probability zero because the corresponding partition cells have zero volume. Let us verify this for the fifth permutation matrix ($P = I_4$). By forming $D^{((2,2))}I_4F$, we see that the fifth cell is described by

$$\begin{bmatrix} 1 & -1 \\ -\rho & -1-\rho \\ 1-\rho & \rho \\ -2\rho & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{4.26}$$

This has no nonzero solutions. (Subtracting the second and third inequalities from the first gives $2\rho x_1 \geq 0$, which combines with the fourth inequality to give $x_1 = 0$. With $x_1 = 0$, the first and third inequalities combine to give $x_2 = 0$.)

While further investigation of the joint design of the integer partition m and frame F—or of the product $D^{(m)}PF$ as P varies over the partitions induced by m—is merited, it is beyond the scope of this thesis. Instead, we have extended our experiments with uniform source to show the potential benefit of using entropy coding to exploit the lack of equiprobable codewords.

Fig. 4-5 summarizes experiments similar to those reported in Figs. 4-3 and 4-4. Each curve in this figure shows, for any given rate R on the horizontal axis, the lowest distortion can be achieved at any rate not exceeding R. The source $x \in \mathbb{R}^4$ has i.i.d. components uniformly distributed on $[-\frac{1}{2}, \frac{1}{2}]$, and Variant I FPQ with modulated harmonic tight frames of sizes $M = 6$ and 7 were used. Performance with rate measured only by (4.3) as before is labeled `fixed rate`. The codewords are highly nonequiprobable at all but the lowest rates. To demonstrate this, we alternatively measure rate by the empirical output entropy and label the performance `variable rate`. Clearly, the rate is significantly reduced by entropy coding at all but the lowest rates.



$N = 4$

Figure 4-5: Performance of Variant I FPQ for fixed- and variable-rate coding of an i.i.d. uniform($[-\frac{1}{2}, \frac{1}{2}]$) source with $N = 4$ using modulated harmonic tight frames of sizes 6 and 7. Also plotted is the performance of entropy-constrained scalar quantization.

Appendices

4.A Proof of Theorem 4.6

The following lemmas are all stated for Variant II initial codewords. They are somewhat stronger than what we need for the proof of Theorem 4.6 because a Variant II initial codeword is automatically a Variant I initial codeword. However, these lemmas will be reused to prove Theorem 4.8 later on.

For convenience, if $\{i_1, \ldots i_k\}$ is a subset of $\{1, 2, \ldots, M\}$ and σ is a permutation on that subset, we simply write

$$P = \begin{pmatrix} i_1 & i_2 & \cdots i_k \\ \sigma(i_1) & \sigma(i_2) & \cdots \sigma(i_k) \end{pmatrix}$$

if Py maps y_{i_ℓ} to $y_{\sigma(i_\ell)}$, $1 \leq \ell \leq k$, and fixes all the other components of vector y.

This notation with round brackets should not be confused with a matrix in which we always use square brackets.

Proofs of the lemmas rely heavily on the key observation that the operator $P(\cdot)\,P^{-1}$ first permutes the columns of the original matrix, then permutes the rows of the resulting matrix by the same manner.

Lemma 4.12. *Assume that $M \geq 3$. If the entries of matrix A satisfy $a_{k,1} \neq a_{\ell,1}$ for some $1 < k < \ell$, then there exists a pair $(P, \hat{y}_{\text{init}})$, where P is a permutation matrix and $\hat{y}_{\text{init}}$ is a Variant II initial codeword associated with some integer partition, such that the inequality (4.16) is violated.*

Proof. Consider the two following cases.

Case 1: If $a_{k,1} < a_{\ell,1}$, choose $P = I_M$, and $\hat{y}_{\text{init}} = (\mu_1, \mu_2, \ldots, \mu_M)$. Consider the following difference:

$$\begin{aligned}
\Delta_{k,\ell} &= \langle (a_{k,j})_j,\, \hat{y}_{\text{init}} \rangle - \langle (a_{\ell,j})_j,\, \hat{y}_{\text{init}} \rangle \\
&= \sum_{j=1}^{M} a_{k,j}\mu_j - \sum_{j=1}^{M} a_{\ell,j}\mu_j \\
&= (a_{k,1} - a_{\ell,1})\mu_1 + \left(\sum_{j=2}^{M} a_{k,j}\mu_j - \sum_{j=2}^{M} a_{\ell,j}\mu_j \right)
\end{aligned}$$

Fix $\mu_2 > \mu_3 > \cdots > \mu_M \geq 0$ and let μ_1 go to $+\infty$. Since $a_{k,1} < a_{\ell,1}$, $\Delta_{k,\ell}$ will go to $-\infty$. Thus, there exist $\mu_1 > \mu_2 > \cdots > \mu_M \geq 0$ such that $\Delta_{k,\ell} < 0$. On the other hand, for $m = (1, 1, \ldots, 1)$, inequality (4.16) requires that $\Delta_{k,\ell} \geq 0$ for all $k < \ell$. Therefore the chosen pair violates inequality (4.16).

Case 2: If $a_{k,1} > a_{\ell,1}$, choose $P = \begin{pmatrix} & k & & \ell & \\ & & \cdots & & \\ & \ell & & k & \end{pmatrix}$. Since $k, \ell \neq 1$, the entries of matrix $A' = PA\,P^{-1}$ will satisfy that $a'_{k,1} < a'_{\ell,1}$. We return to the first case, completing the proof. □

Lemma 4.13. *Assume that $M \geq 3$. If the entries of matrix A satisfy $a_{k,j} \neq a_{\ell,j}$, for any pairwise distinct triple (k, j, ℓ), then there exists a pair $(P, \hat{y}_{\text{init}})$, where P is a permutation matrix and $\hat{y}_{\text{init}}$ is a Variant II initial codeword associated with some integer partition, such that the inequality (4.16) is violated.*

Proof. We first show that there exists some permutation matrix P_1 such that $\tilde{A} = P_1 A P_1^{-1}$ satisfies the hypothesis of Lemma 4.12. Indeed, consider the following cases:

1. If $j = 1$, it is obvious to choose $P_1 = I_M$.
2. If $j > 1$ and $k > 1$, choosing $P_1 = \begin{pmatrix} 1 & j \\ j & 1 \end{pmatrix}$ yields $\tilde{a}_{k,1} = a_{k,j} \neq a_{l,j} = \tilde{a}_{l,1}$, since $k, \ell \notin \{1, j\}$.
3. If $j > 1$ and $k = 1$, choosing $P_1 = \begin{pmatrix} 1 & j \\ j & 1 \end{pmatrix}$ yields $\tilde{a}_{j,1} = a_{k,j} \neq a_{l,j} = \tilde{a}_{l,1}$, since $k = 1$, and $\ell \notin \{1, j\}$. Note that in this case, $j \neq 1$, and so $\tilde{A}$ satisfies the hypothesis of Lemma 4.12.

Now with P_1 chosen as above, according to Lemma 4.12 there exits a pair $(P_2, \hat{y}_{\text{init}})$, where P is a permutation matrix and $\hat{y}_{\text{init}}$ is a Variant II initial codeword associated with some integer partition, such that

$$\begin{aligned} \mathbf{0} \not\leq & \; D^{(m)} P_2 \tilde{A} P_2^{-1} \hat{y}_{\text{init}} \\ = & \; D^{(m)} P_2 \left(P_1 A P_1^{-1} \right) P_2^{-1} \hat{y}_{\text{init}} \\ = & \; D^{(m)} P A P^{-1} \hat{y}_{\text{init}}, \end{aligned}$$

where $P \triangleq P_2 P_1$. Since the product of any two permutation matrices is also a permutation matrix, the pair $(P, \hat{y}_{\text{init}})$ violates the inequality (4.16). □

Lemma 4.14. *Suppose that A is a diagonal matrix. Then the inequality (4.16) holds for every integer partition and every Variant II initial codeword associated with it, only if A is equal to the identity matrix scaled by a nonnegative factor.*

Proof. We first show that there exists some permutation matrix P_1 such that $\tilde{A} = P_1 A P_1^{-1}$ satisfies the hypothesis of Lemma 4.12. Indeed, consider the following cases:

1. If $j = 1$, it is obvious to choose $P_1 = I_M$.
2. If $j > 1$ and $k > 1$, choosing $P_1 = \begin{pmatrix} 1 & j \\ j & 1 \end{pmatrix}$ yields $\tilde{a}_{k,1} = a_{k,j} \neq a_{l,j} = \tilde{a}_{l,1}$, since $k, \ell \notin \{1, j\}$.

3. If $j > 1$ and $k = 1$, choosing $P_1 = \begin{pmatrix} 1 & j \\ j & 1 \end{pmatrix}$ yields $\tilde{a}_{j,1} = a_{k,j} \neq a_{l,j} = \tilde{a}_{l,1}$, since $k = 1$, and $\ell \notin \{1, j\}$. Note that in this case, $j \neq 1$, and so $\tilde{A}$ satisfies the hypothesis of Lemma 4.12.

Now with P_1 chosen as above, according to Lemma 4.12 there exits a pair $(P_2, \hat{y}_{\text{init}})$, where P is a permutation matrix and $\hat{y}_{\text{init}}$ is a Variant II initial codeword associated with some integer partition, such that

$$\begin{aligned} \mathbf{0} \not\leq & \; D^{(m)} P_2 \, \tilde{A} \, P_2^{-1} \, \hat{y}_{\text{init}} \\ = & \; D^{(m)} P_2 \left(P_1 \, A \, P_1^{-1} \right) P_2^{-1} \, \hat{y}_{\text{init}} \\ = & \; D^{(m)} P A \, P^{-1} \, \hat{y}_{\text{init}}, \end{aligned}$$

where $P \triangleq P_2 P_1$. Since the product of any two permutation matrices is also a permutation matrix, the pair $(P, \hat{y}_{\text{init}})$ violates the inequality (4.16). □

Lemma 4.15. *Suppose that A is a diagonal matrix. Then the inequality (4.16) holds for every integer partition and every Variant II initial codeword associated with it, only if A is equal to the identity matrix scaled by a nonnegative factor.*

Proof. Suppose that $A = \text{diag}(a_1, a_2, \ldots, a_M)$. We first show that $a_i \geq 0$ for every i by contradiction.

If $a_1 < 0$, we can choose $P = I_M$ and $\mu_1 > \mu_2 > \cdots > \mu_M \geq 0$, where μ_1 is large enough relative to $\mu_2, \cdots, \mu_M$ to violate inequality (4.16).

If $a_j < 0$ for some $1 < j \leq M$, using $P = \begin{pmatrix} 1 & j \\ j & 1 \end{pmatrix}$ yields $a_1' = a_j < 0$, where a_1' is the first entry on the diagonal of matrix $A' \triangleq P A \, P^{-1}$. Repeating the previous argument, we get the contradiction.

Now we show that if $a_k \neq a_\ell$ for some $1 \leq k < \ell \leq M$, there exists a pair $(P, \hat{y}_{\text{init}})$, where P is a permutation matrix and $\hat{y}_{\text{init}}$ is a Variant II initial codeword associated with some integer partition, such that inequality (4.16) is violated.

Case 1: if $a_k < a_\ell$, choose $P = I_M$ and consider $\hat{y}_{\text{init}} = (\mu_1, \mu_2, \ldots, \mu_M)$, where

$\mu_\ell = \mu_k - \varepsilon$ for some positive ε. Choose μ_k such that

$$\mu_k > \frac{\varepsilon a_\ell}{a_\ell - a_k} \geq 0. \tag{4.27}$$

On the other hand, we can choose ε small enough so that μ_ℓ is positive as well. The other components can therefore be chosen to make $\hat{y}_{\text{init}}$ a Variant II initial codeword associated with integer partition $m = (1, 1, \ldots, 1)$. For the above choice of μ_k we can easily check that $\Delta_{k,\ell} = a_k\mu_k - a_\ell\mu_\ell < 0$, violating inequality (4.16).

Case 2: if $a_k > a_\ell$, choosing $P = \begin{pmatrix} k & \ell \\ \ell & k \end{pmatrix}$ yields

$$PA\,P^{-1} = \text{diag}(a_1, a_2, \ldots, a_\ell, \ldots, a_k, \ldots, a_M).$$

We return to case 1, completing the proof. □

Theorem 4.6. First note that a Variant II initial codeword is always a Variant I initial codeword, therefore, Lemmas 4.12, 4.13, and 4.15 also apply for Variant I initial codewords. From Lemma 4.13, all entries on each column of matrix A are constant except for the one that lies on the diagonal. Thus, A can be written as $A = \tilde{I} + J$, where $\tilde{I} = \text{diag}(a_1, a_2, \ldots, a_M)$, and

$$J = \begin{bmatrix} b_1 & b_2 & \cdots & b_M \\ b_1 & b_2 & \cdots & b_M \\ \vdots & \vdots & & \vdots \\ b_1 & b_2 & \cdots & b_M \end{bmatrix} \in \mathcal{J}.$$

Recall that from properties (P1) and (P2) of J we have

$$D^{(m)} P J P^{-1} = \mathbf{0}, \qquad \text{for any } m.$$

Hence,

$$D^{(m)} P \tilde{I} P^{-1}\, \hat{y}_{\text{init}} \geq \mathbf{0}, \qquad \text{for any } m \text{ and any } \hat{y}_{\text{init}}. \tag{4.28}$$

From (4.28) and Lemma 4.15, we can deduce that $\tilde{I} = aI_M$, for some nonnegative constant a. $\square$

4.B Proof of Theorem 4.8

In order for R to produce consistent reconstructions, we need the following inequality (noting that $V = V^{-1}$ for any $V \in \mathcal{Q}(m)$):

$$\widetilde{D}^{(m)} V P A P^{-1} V \hat{y}_{\text{init}} \geq \mathbf{0}, \qquad \text{for any } V \in \mathcal{Q}(m) \text{ and } P \in \mathcal{G}(m), \tag{4.29}$$

where $A = FR$. We first fix the sign-changing matrix V to be the identity matrix I_M. Then the first $L(m)$ rows of (4.29) exactly form the inequality (4.16). Since Lemmas 4.12, 4.13, and 4.15 are stated for Variant II initial codewords, it follows from Theorem 4.6 that A must be of the form $aI_M + J$, where $a \geq 0$ and $J \in \mathcal{J}$. Substituting in to (4.29), we obtain

$$a\widetilde{D}^{(m)} \hat{y}_{\text{init}} + \widetilde{D}^{(m)} V P J P^{-1} V \hat{y}_{\text{init}} \geq \mathbf{0}, \tag{4.30}$$

Now we show that $J = \mathbf{0}$ by contradiction. Indeed, suppose all entries in column i of J are b_i, for $1 \leq i \leq M$. Consider the following cases:

1. If the first column of J is negative, choose $V = P = I_M$ and $\hat{y}_{\text{init}} = (\mu_1, \mu_2, \ldots, \mu_M)$ associated with partition $m = (1, 1, \ldots, 1)$. Consider the last row of inequality (4.30):

$$b_1\mu_1 + a\mu_{M-1} + \sum_{i=2}^{M} b_i\mu_i \geq 0. \tag{4.31}$$

   Since $M \geq 3$, $M-1 \neq 1$. Therefore the scale associated with μ_1 in the left hand side of inequality (4.31) is $b_1 < 0$. Hence, choosing μ_1 large enough certainly breaks inequality (4.31), and therefore violates inequality (4.30).

2. If the first column of J is positive, choosing $P = I_M$, $V = \text{diag}(-1, 1, 1, \ldots, 1)$ makes the first entry of the $(M-1)$th row of matrix $VPJP^{-1}V$ negative (note that $M - 1 \neq 1$ and the operator $V(\cdot)V$ first changes the signs of columns of

the original matrix and then changes the signs of rows of the resulting matrix by the same manner.) Repeating the argument in the first case we can break the last row of inequality (4.30) by appropriate choice of $\hat{y}_{\text{init}}$.

3. If column ℓ of J, $1 < \ell \leq M$ is different from zero, choosing $P = \begin{pmatrix} 1 & \ell \\ \ell & 1 \end{pmatrix}$ leads us to either case 1 or case 2.

Hence,

$$A = FR = aI_M. \tag{4.32}$$

Equality (4.32) states that the row vectors of F and the column vectors of R forms a biorthogonal basis pair of $\mathbb{R}^N$ within a nonnegative scale factor. Since the number of vectors in each basis can not exceed the dimension of the space, we can deduce $M \leq N$. On the other hand, $M \geq N$ because F is a frame. Thus, $M = N$.

Chapter 5

Closing Remarks

In this work, we have proposed two generalizations of permutation source codes which improve rate–distortion performance while adding very little to encoding complexity. The first generalization, CPSCs, while allowing multiple initial codewords, considerably increases the design complexity. The second generalization, FPQ—in which we combine PSCs with overcomplete representations in a unique coding scheme—arouses reconstruction and frame design problems simultaneously.

Two methods are introduced in the first part of the thesis to reduce the design complexity of CSPCs: restricting the subcodebooks to share a common integer partition and allocating rates across subcodebooks using high-resolution analysis of wrapped spherical codes. For the common integer partition, we have mapped the initial vectors design problem to a vector quantization problem. In order to restrict the searching space of integer partitions, we also attempt to extend a necessary condition for optimal integer partitions of ordinary PSCs to those of concentric PSCs, by proving a weaker proposition with a constraint imposed on the initial codewords. This constraint is somewhat strange, but removing it from the proposition is not easy and requires further work. When the integer partitions are different, for the purpose of guiding PSC rate allocation, we derive fixed- and variable-rate allocations of wrapped spherical codes for memoryless Gaussian source under the assumption of high-resolution. Proving effectiveness of these two heuristic methods, however, is a remaining challenge.

To cope with the issues of FPQ, we exploit canonical reconstructions, as well as consistent reconstructions in the second part of the thesis. Resembling the reconstruction methods for scalar-quantized frame expansions, we provide linear-program-based and quadratic-program-based algorithms to achieve consistent reconstructions for different source distributions. For the frame design problem, we derive a variety of necessary and sufficient conditions on the frames for linear reconstructions in general, and canonical reconstructions in particular, to be consistent. It is because we want to combine advantages of both linear reconstruction, which has low complexity, and consistent reconstruction, which yields better performance. This idea again follows strictly the philosophy of the overall thesis, "low complexity while still attaining good performance." Along the way of describing "good" frames for the combination of canonical and consistent reconstructions, a complete characterization of real restricted equiangular tight frames in the codimension-1 case is given, in the relation with the popular harmonic tight frames. This result might be of independent interest. Although simulations have demonstrated the efficiency of the proposed reconstruction methods and choices of frames, we do not know what frames are "good" when dealing with the two types of reconstructions, linear and consistent, separately. Another issue this work has completely ignored is the joint design of integer partitions and frames. These gaps can be explored in future work. Also, a naïve idea of combining CPSCs and FPQ may yield better performance and is worth trying.

Bibliography

[1] L. R. Varshney, "Permutations and combinations," Dec. 2004, Massachusetts Institute of Technology 6.961 Report (Fall 2004).

[2] ——, "On the flexibility of ties," Jul. 2005, Massachusetts Institute of Technology.

[3] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, no. 6, pp. 2325–2383, Oct. 1998.

[4] N. Farvardin and J. W. Modestino, "Adaptive buffer-instrumented entropy-coded quantizer performance for memoryless sources," *IEEE Trans. Inform. Theory*, vol. 32, pp. 9–22, Jan. 1986.

[5] R. Laroia and N. Farvardin, "A structured fixed-rate vector quantizer derived from a variable-length scalar quantizer-I: Memoryless sources," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 851–867, May 1993.

[6] ——, "A structured fixed-rate vector quantizer derived from a variable-length scalar quantizer-II: Vector sources," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 868–876, May 1993.

[7] A. S. Balamesh, "Block-constrained methods of fixed-rate entropy-coded quantization," Ph.D. dissertation, University of Michigan, Ann Arbor, MI, 1993.

[8] V. K. Goyal, S. A. Savari, and W. Wang, "On optimal permutation codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2961–2971, Nov. 2001.

[9] J. Hamkins and K. Zeger, "Gaussian source coding with spherical codes," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2980–2989, Nov. 2002.

[10] D. Slepian, "Group codes for the Gaussian channel," *Bell Syst. Tech. J.*, vol. 47, pp. 575–602, 1968.

[11] J. Kovačević and A. Chebira, "Life after bases: The advent of frames (Part I)," *IEEE Sig. Process. Mag.*, vol. 24, no. 4, pp. 86–104, Jul. 2007.

[12] ——, "Life after bases: The advent of frames (Part II)," *IEEE Sig. Process. Mag.*, vol. 24, no. 5, pp. 115–125, Sep. 2007.

[13] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA: SIAM, 1992.

[14] N. T. Thao and M. Vetterli, "Reduction of the MSE in R-times oversampled A/D conversion from $O(1/R)$ to $O(1/R^2)$," *IEEE Trans. Signal Process.*, vol. 42, pp. 200–203, Jan. 1994.

[15] ——, "Deterministic analysis of oversampled A/D conversion and decoding improvement based on consistent estimates," *IEEE Trans. Signal Process.*, vol. 42, no. 3, pp. 519–531, Mar. 1994.

[16] V. K. Goyal, M. Vetterli, and N. T. Thao, "Quantized overcomplete expansions in $\mathbb{R}^N$: Analysis, synthesis, and algorithms," *IEEE Trans. Inform. Theory*, vol. 44, no. 1, pp. 16–31, Jan. 1998.

[17] Z. Cvetković, "Resilience properties of redundant expansions under additive noise and quantization," *IEEE Trans. Inform. Theory*, vol. 49, no. 3, pp. 644–656, Mar. 2003.

[18] S. Rangan and V. K. Goyal, "Recursive consistent estimation with bounded noise," *IEEE Trans. Inform. Theory*, vol. 47, no. 1, pp. 457–464, Jan. 2001.

[19] J. J. Benedetto, A. M. Powell, and Ö. Yilmaz, "Sigma–Delta ($\Sigma\Delta$) quantization and finite frames," *IEEE Trans. Inform. Theory*, vol. 52, no. 5, pp. 1990–2005, May 2006.

[20] A. M. Powell, "Estimation algorithms with noisy frame coefficients," in *Proc. Wavelets XII, part of SPIE Optics & Photonics*, vol. 6701, San Diego, CA, Aug. 2007.

[21] B. G. Bodmann and V. I. Paulsen, "Frame paths and error bounds for sigma-delta quantization," *Appl. Comput. Harmon. Anal.*, vol. 22, no. 2, pp. 176–197, Mar. 2007.

[22] B. G. Bodmann and S. P. Lipshitz, "Randomly dithered quantization and sigma–delta noise shaping for finite frames," *Appl. Comput. Harmon. Anal.*, vol. 25, no. 3, pp. 367–380, Nov. 2008.

[23] B. Beferull-Lozano and A. Ortega, "Efficient quantization for overcomplete expansions in $\mathbb{R}^n$," *IEEE Trans. Inform. Theory*, vol. 49, no. 1, pp. 129–150, Jan. 2003.

[24] L. R. Varshney and V. K. Goyal, "Ordered and disordered source coding," in *Proc. UCSD Workshop Inform. Theory & Its Applications*, La Jolla, CA, Feb. 2006.

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

[26] D. Slepian, "Permutation modulation," *Proc. IEEE*, vol. 53, no. 3, pp. 228–236, Mar. 1965.

[27] J. G. Dunn, "Coding for continuous sources and channels," Ph.D. dissertation, Columbia Univ., New York, 1965.

[28] T. Berger, F. Jelinek, and J. K. Wolf, "Permutation codes for sources," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 160–169, Jan. 1972.

[29] T. Berger, "Optimum quantizers and permutation codes," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 759–765, Nov. 1972.

[30] ——, "Minimum entropy quantizers and permutation codes," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 149–157, Mar. 1982.

[31] H. A. David and H. N. Nagaraja, *Order Statistics*, 3rd ed. Wiley-Interscience, 2003.

[32] F. Jelinek, "Buffer overflow in variable length coding of fixed rate sources," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 490–501, May 1968.

[33] D. Han and D. R. Larson, *Frames, Bases and Group Representations.* Providence, RI: American Mathematical Society, 2000.

[34] V. K. Goyal, J. Kovačević, and J. A. Kelner, "Quantized frame expansions with erasures," *Appl. Comput. Harmon. Anal.*, vol. 10, no. 3, pp. 203–233, May 2001.

[35] Y. Eldar and H. Bölcskei, "Geometrically uniform frames," *IEEE Trans. Inform. Theory*, vol. 49, no. 4, pp. 993–1006, Apr. 2003.

[36] J. D. Kolesar, "SD modulation and correlation criteria for the construction of finite frames arising in communication theory," Ph.D. dissertation, University of Maryland, College Park, MD, 2004.

[37] F. Abdelkefi, "Performance of Sigma-Delta quantizations in finite frames," *IEEE Trans. Inform. Theory*, vol. 54, no. 11, pp. 5087–5101, Nov. 2008.

[38] R. B. Holmes and V. I. Paulsen, "Optimal frames for erasures," *Linear Algebra Appl.*, vol. 377, pp. 31–51, 2004.

[39] T. Strohmer and R. W. Heath, "Grassmanian frames with application to coding and communications," *Appl. Comput. Harmon. Anal.*, vol. 14, no. 3, pp. 257–275, 2003.

[40] M. A. Sustik, J. A. Tropp, I. S. Dhillon, and R. W. Heath, "On the existence of equiangular tight frames," *Linear Algebra Appl.*, vol. 426, pp. 619–635, 2007.

[41] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[42] T. Strohmer, "A note on equiangular tight frames," *Linear Algebra Appl.*, vol. 429, pp. 326–330, 2008.

[43] P. G. Casazza, D. Redmond, and J. C. Tremain, "Real equiangular frames," in *Proc. 42th Annu. Conf. Information Sciences and Systems (CISS 2008)*, Mar. 2008, pp. 715–720.

[44] 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.

[45] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Logist. Q.*, vol. 2, no. 1-2, pp. 83–97, Mar. 1955.

[46] P. F. Swaszek and J. B. Thomas, "Optimal circularly symmetric quantizers," *J. Franklin Inst.*, vol. 313, pp. 373–384, Jun. 1982.

[47] J. Hamkins, "Design and analysis of spherical codes," Ph.D. dissertation, Univ. Illinois at Urbana-Champaign, Sept. 1996.

[48] S. P. Lloyd, "Least squares quantization in PCM," Jul. 1957, unpublished Bell Laboratories Technical Note.

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

[50] 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.

[51] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices, and Groups.* New York: Springer-Verlag, 2008.

[52] N. T. Thao and M. Vetterli, "Lower bound on the mean-squared error in oversampled quantization of periodic signals using vector quantization analysis," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 469–479, Mar. 1996.

[53] R. M. Gray and T. G. Stockham, Jr., "Dithered quantizers," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 805–812, May 1993.

[54] H. Viswanathan and R. Zamir, "On the whiteness of high-resolution quantization errors," *IEEE Trans. Inform. Theory*, vol. 47, no. 5, pp. 2029–2038, Jul. 2001.