Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2009, Article ID 463823, 13 pages
doi:10.1155/2009/463823

Research Article

Progressive Refinement of Beamforming Vectors for High-Resolution Limited Feedback

Robert W. Heath Jr.,[1] Tao Wu,[2] and Anthony C. K. Soong[3]

[1] *Wireless Networking and Communications Group, Department of Electrical and Computer Engineering, The University of Texas at Austin, 1 University Station C0803, Austin, TX 78712-0240, USA*
[2] *Huawei Technologies, 10180 Telesis Court, Suite 365, San Diego, CA 92121, USA*
[3] *Huawei Technologies, 1700 Alma Drive, Suite 500, Plano, TX 75075, USA*

Correspondence should be addressed to Robert W. Heath Jr., rheath@ece.utexas.edu

Received 25 December 2008; Revised 20 April 2009; Accepted 15 June 2009

Recommended by Ana Perez-Neira

Limited feedback enables the practical use of channel state information in multiuser multiple-input multiple-output (MIMO) wireless communication systems. Using the limited feedback concept, channel state information at the receiver is quantized by choosing a representative element from a codebook known to both the receiver and transmitter. Unfortunately, achieving the high resolution required with multiuser MIMO communication is challenging due to the large number of codebook entries required. This paper proposes to use a progressively scaled local codebook to enable high resolution quantization and reconstruction for multiuser MIMO with zero-forcing precoding. Several local codebook designs are proposed including one based on a ring and one based on mutually unbiased bases; both facilitate efficient implementation. Structure in the local codebooks is used to reduce search complexity in the progressive refinement algorithm. Simulation results illustrate sum rate performance as a function of the number of refinements.

1. Introduction

Multiuser multiple-input multiple-output (MIMO) communication systems can use limited feedback of channel state information obtained from the receiver to perform multiuser transmission on the downlink [1]. With limited feedback, channel state information is quantized by choosing a representative element from a codebook known to both the receiver and transmitter. The transmitter uses quantized channel state information to design the transmission strategy, for example to find the zero-forcing beamforming vectors [2, 3]. Because imperfect channel state information is used at the transmitter, multiuser MIMO systems are quantization error limited at high signal-to-noise ratios. Consequently, higher resolution is required than in comparable single user systems [2]. Unfortunately, achieving high-resolution in commercial wireless systems through the use of large codebooks is challenging due to practical requirements like low digital storage, fast codeword search, and variable feedback allocation.

This paper proposes a new codebook design and quantization algorithm that facilitates high-resolution limited feedback beamforming. The key idea is the use of two codebooks: a nonlocal codebook and a local codebook, to implement a progressive refinement beamforming quantization algorithm. The base codebook is designed to be as uniform as possible, using for example a Grassmannian codebook [4]. The local codebook is inspired by recent work on clustered codebooks that are designed to take advantage of correlation or localization in the channel [5, 6]. The local codebook consists of a root vector and a set of vectors that are all "close" to the root vector and yet are far apart from each other. The base codebook is used to generate an initial quantization while successive rotations and shrinking operations applied to the local codebook are used to generate progressively better refinements. The proposed algorithm allows for high-resolution using multiple refinements; it has low-storage requirements since only a base and single local codebook need to be stored; it facilitates fast codeword search

since each step only requires a search over a small local codebook; it can be used with single user and multiple user beamforming; and it allows variable feedback rate allocation by assigning different numbers of refinements to different users.

The main technical contributions of this paper are in the area of local codebook design and in its application for progressive refinement beamforming. We propose a specific construction of a local codebook, called a ring codebook, which consists of a root vector and several nonroot vectors that are equidistant from the root vector. We provide several specific ring constructions for two and four antennas using uniform phase quantization and mutually unbiased bases[7, 8]. We also present an approach for building nonring local codebooks from a general codebook, like a Grassmannian codebook.

Using the local codebook concept we propose an algorithm for progressively refining an initial base quantization through several refinements that involve rotating and shrinking the local codebook based on the previous quantization value at each step. We also propose several low complexity variations of the algorithm. To avoid rotating the local codebook, we propose to rotate the vector to be quantized instead of the whole local codebook, but requiring a derotation operation on the resulting reconstruction. To further reduce complexity, we show how ring codebooks can allow a different rescaling operation where the vector to be quantized is scaled prior to quantization. We suggest an approach to choosing the amount of shrinkage at each codebook step based on numerical optimization. While our approach can be applied to both single user and multiuser MIMO limited feedback scenarios, we focus on the multiuser MIMO case with zero-forcing precoding due to its high-resolution requirement. Simulations illustrate the performance of the proposed refinement algorithms in uncorrelated and correlated Rayleigh fading channels in terms of sum rate for two- and four-user systems.

Local codebooks were first proposed in [5] and later studied in [6] in more detail. That work motivates the utility of local codebooks in single user MIMO for time varying channels and channels with spatial correlation. A successive refinement algorithm for single user MISO beamforming in time-varying channels was considered in [9] and later extended to MISO-OFDM [10]. Local codebooks were considered in [9, 10] but specific constructions beyond a Lloyd-like solution were not studied. Radius selection in [9] was done based on single user MISO performance bounds that do not necessarily correspond to the multiuser MIMO case. Compared with [5, 6, 9, 10] we use the local codebook definition, scaling, and rotations operations but we also propose several local codebook designs, describe how to use local codebooks to implement progressive refinement with low complexity variations, and consider multiuser MIMO communication. Hierarchical quantization was proposed in [11] for time varying channels and was applied to the case of multiuser MIMO. That algorithm uses a hierarchical structured beamforming codebook derived through a smart partitioning operation of a DFT beamforming codebook. The number of levels though is fixed by the base codebook and the entire codebook must be stored unless special structure is exploited. Our approach allows non-DFT codebooks (which are good primarily for line-of-sight channels and uniform linear arrays), allows a variable number of refinement levels, and has structure that permits reduced storage and low search complexity. We provide performance comparisons to show that our approach performs well in a variety of channel conditions.

From the vector quantization perspective, the proposed progressive refinement technique falls within the class of constrained vector quantizers [26, Chapter 12] like tree-structured vector quantizers or residual vector quantizers [12]. Our work is not a straightforward extension of prior work on vector quantization, however, since our quantization is on the Grassmannan manifold [13], involving subspace distortion measures and non-Euclidean distance concepts. Unlike typical work on vector quantization, we use mathematical concepts to build structured codebooks instead of relying on the variations of the Lloyd algorithm to build a codebook from a training set. Exploring deeper connections between our work and structured vector quantizers is an interesting topic of future research.

Organization. In Section 2 we review the multiuser MIMO beamforming system model. In Section 3, we present the concept of progressive refinement using a base and local codebook. Then in Section 4 we define local codebooks and local codebook operations. In Section 5 we present several preferred codebook designs including the general ring codebook, ring codebook from Kerdock codes, and a procedure for deriving a local codebook from a nonlocal codebook. Then in Section 6 we present the progressive refinement algorithm, discussing two approaches to reduce complexity and remarking on the selection of the radius. In Section 7 we present several simulation results for the case of two and four transmit antennas. Finally in Section 8 we draw some conclusions and mention directions for future research.

Notation. Bold lowercase $\mathbf{a}$ is used to denote column vectors, bold uppercase $\mathbf{A}$ is used to denote matrices, nonbold letters a, A are used to denote scalar values, and caligraphic letters $\mathcal{A}$ to denote sets or functions of sets. Using this notation, $|a|$ is the magnitude of a scalar, $\|\mathbf{a}\|$ is the vector 2-norm, $\mathbf{A}^*$ is the conjugate transpose, $\mathbf{A}^T$ is the matrix transpose, $\mathbf{A}^{-1}$ denotes the inverse of a square matrix, $\mathbf{A}^\dagger$ is the Moore-Penrose pseudo inverse, $[\mathbf{A}]_{k,l}$ is the scalar entry of $\mathbf{A}$ in kth row lth column, $[\mathbf{A}]_{:,k}$ is the kth column of matrix $\mathbf{A}$, $[\mathbf{a}]_k$ is the kth entry of $\mathbf{a}$, $|\mathcal{A}|$ is the cardinality of set $\mathcal{A}$, and $:=$ denotes by definition. We use the notation $\mathcal{N}(\mathbf{m}, \mathbf{R})$ to denote a complex circularly symmetric Gaussian random vector with mean $\mathbf{m}$ and covariance $\mathbf{R}$. We use $\mathbb{E}$ to denote expectation.

2. Multiuser Zero-Forcing Beamforming with Limited Feedback

Consider a multiuser MIMO system with limited feedback beamforming. Following prior work we assume that there are $U = N_t$ active users, each with a single receive antenna [2]. We do not consider user scheduling; it is known

that scheduling reduces the required codebook resolution [3]; thus we expect our approach to work seamlessly with scheduling. The received signal at the uth user for discrete-time n is given by

$$y_u[n] = \mathbf{h}_u^T \mathbf{f}_u s_u[n] + \mathbf{h}_u^T \sum_{k \neq u} \mathbf{f}_k s_k[n] + \nu_u[n], \quad (1)$$

where $y_u[n]$ is the scalar received signal, $\mathbf{h}_u^T$ is the $1 \times N_t$ complex channel vector, $\mathbf{f}_u$ is the unit norm transmit beamforming vector, $s_u[n]$ is the complex transmitted symbol, and $\nu_u[n]$ is a realization of an i.i.d. random process with circularly symmetric complex Gaussian distribution $\mathcal{N}(0, N_o)$.

A zero-forcing beamforming system with limited feedback uses quantized channel direction information from each user to derive the beamforming vectors $\{\mathbf{f}_u\}_{u=1}^U$. The feedback channel is generally assumed to be error-free and zero-delay [1]. In prior work, the channel direction is quantized by selecting an element from a codebook $\mathcal{F}$, in this case an ordered set of unit norm vectors. Each user performs quantization by solving

$$\mathcal{Q}(\mathbf{h}_u, \mathcal{F}) = \arg\min_{\mathbf{w} \in \mathcal{F}} d\left(\frac{\mathbf{h}_u}{\|\mathbf{h}_u\|}, \mathbf{w}\right) \quad (2)$$

where $d(\mathbf{a}, \mathbf{b}) := \sqrt{1 - |\mathbf{a}^* \mathbf{b}|^2}$ is the *subspace distance* function for unit norm vector arguments $\mathbf{a}$ and $\mathbf{b}$. This is a proper distance function for points $\mathbf{a}$ and $\mathbf{b}$ on the Grassmann manifold $G(N_t, 1)$, which is the collection of one dimensional subspaces in $\mathbb{C}^{N_t}$. The form of quantization in (2) minimizes the angle between the normalized channel vector $\mathbf{h}_u/\|\mathbf{h}_u\|$ and the entries of the codebook. Under the zero-forcing criterion, the transmit beamforming vectors $\mathbf{f}_u$ is computed from normalized columns of the pseudo inverse of the effective channel $\mathbf{F} = \left[\mathcal{Q}(\mathbf{h}_1, \mathcal{F})^T; Q(\mathbf{h}_2, \mathcal{F})^T; \ldots; Q(\mathbf{h}_U, \mathcal{F})^T \right]^\dagger$.

Implementing the quantization in (2) is challenging because the number of entries in the codebooks $\mathcal{F}$ can be quite large in multiuser systems [2]. For example, to maintain a constant gap from the sum rate in zero-forcing, the size of the codebook in bits $\log_2|\mathcal{F}|$ grows linearly with the signal-to-noise ratio (SNR), measured in dB, and the number of users assuming $N_t = U$ [2].

Commercial wireless systems use codebooks with special structure to implement beamforming vector quantization. Desirable properties of such codebooks for multiuser systems include low digital storage, fast codeword search, high-resolution, and variable feedback allocation. Low digital storage means that either the codebook coefficients can be stored with low precision (saving valuable on-chip RAM) or the codebook can be generated with a simple algorithm. Fast codeword search means that the vector quantization operation can be implemented with lower computational complexity using, for example, fewer mathematical operations or simplified operations like sign flips. High resolution means that large codebook sizes are feasible, for example, codebooks with $|\mathcal{F}| = 2^{12} = 4096$ entries may be required to enable multiuser MIMO operation. Variable feedback allocation means that different codebook sizes can be allocated to different users, based on their operating conditions. Unfortunately, previous codebook designs lack one or more properties that are desirable for practical implementation. This motivates the locally refined search strategy as described in this paper.

3. Progressive Refinement of Beamforming Vectors

To reduce the complexity of codeword search, this paper proposes to progressively refine an initial beamformer quantization using successively smaller local codebooks. The idea is illustrated in Figure 1. The first quantization is performed with a nonlocal base codebook. In the next stage quantization occurs using a local codebook, in this case a ring codebook with the center of the previously chosen codeword. The process repeats with progressively smaller local codebooks. In each step, the previously chosen codeword is used as a center for the next local refinement. We enlarge the effective codebook size by progressively applying a local codebook in a smaller and smaller area. Note that search complexity is reduced: instead of implementing directly the brute force search over $\mathcal{F}$ in (2), our approach employs several searches over multiple smaller sized codebooks.

A block diagram for the proposed multiuser MIMO system with progressive quantization and reconstruction is illustrated in Figure 2. Unlike a conventional limited feedback system, the transmitter and receiver have two codebooks of unit norm vectors: a base codebook denoted $\mathcal{F}$ and a local codebook denoted $\mathcal{S}$. Rather than using multiple local codebooks each with smaller radius, we rotate and scale a single local codebook. This reduces storage requirements and allows us to exploit structure in the local codebook to reduce computational complexity.

The base codebook should be as uniform as possible. This objective is already achieved by codebooks found in literature including Grassmannian codebooks that maximize the minimum subspace distance between vectors [4, 14], DFT codebooks [15, 16], Kerdock/mutually unbiased bases codebooks [7, 8], and others. Variations of these codebooks appear in several commercial wireless systems including IEEE 802.16e wireless system [17], 3GPP LTE systems [18, 19], and 3GPP2 UMB systems [20]. In this paper we assume that a good uniform base codebook is given. For example for our simulations with $N_t = 4$, we use the 6 bit $|\mathcal{F}| = 64$ Grassmannian codebook and the 4 bit $|\mathcal{F}| =$ 163GPP LTE codebooks as a base codebook. Because we have multiple levels of refinement, it is not necessary to choose a large codebook for the initial quantization—codebooks that facilitate low-storage and search complexity can be used at this stage.

The choice of the local codebook and the use of local codebooks to implement progressive beamforming vector refinement are the main subjects of this paper. A formal definition of a local codebook, desirable properties of local codebooks, and the rotation and scaling operations are provided in Section 4. Several preferred local codebooks are identified in Section 5. Finally, the progressive refinement

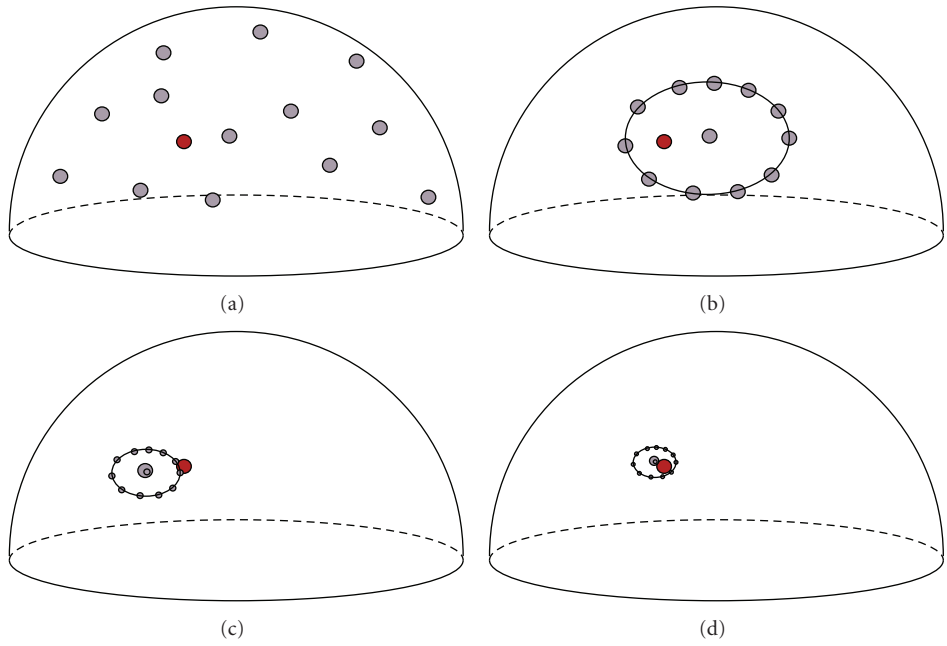

Figure 1: Illustration of progressive quantization with a local codebook, in this case a ring as described in more detail in Section 4. Points on a sphere are used for visualization purposes. (a) Quantization with a base codebook to choose the starting point for progressive refinement. (b) Quantization with a ring codebook centered around the previously chosen codebook point. (c) Next level of refinement with a smaller ring, centered around the previously chosen codebook point. (d) The process repeats with a smaller ring until the desired performance is reached.

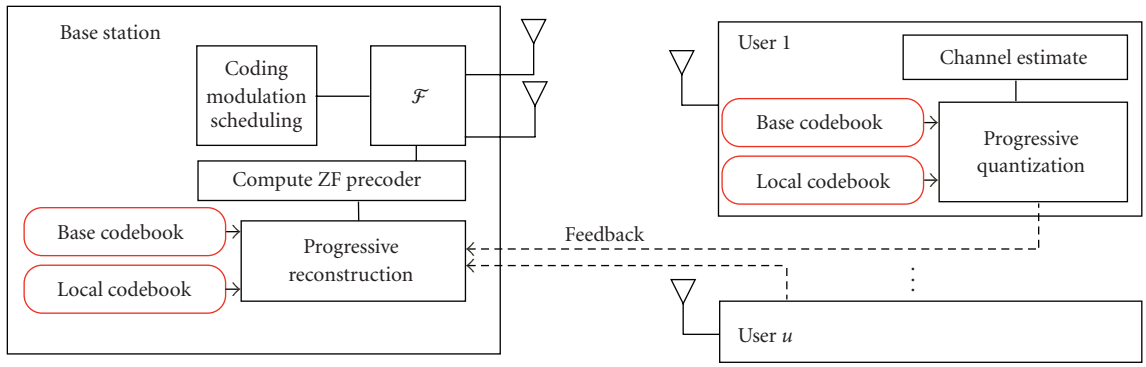

Figure 2: Illustration of a multiuser MIMO system with limited feedback beamforming. Progressive quantization is employed at the receiver while progressive reconstruction is used at the transmitter. A nonlocal base and a local codebook, both known to transmitter and receiver, are used in the progressive quantization and reconstruction.

algorithm and low complexity variations that exploit local codebook structure are described in Section 6.

4. Local Codebook Operations

In this section we define the concept of a local codebook, scaling, and rotation operations.

4.1. Local Codebook Definition. A local codebook is a codebook that consists of a root or centroid vector and several other vectors that are all sufficiently close to a root vector [5, 6]. Let the size of the local codebook be denoted $N_l \geq N_t + 1$. To aid in the definitions of scaling and rotation, all local codebooks are built using the special $N_t \times 1$ root vector: $\mathbf{e}_1 := [1, 0, \ldots, 0]^T$. We define a *local codebook* as follows.

Definition 1. A local codebook with N_l entries has the following properties.

(1) It contains $\mathbf{e}_1$. Let the codebook be $\mathcal{S} = \{\mathbf{e}_1, \mathbf{w}_0, \ldots, \mathbf{w}_{N_l-2}\}$.

(2) All vectors must have a nonzero distance to the root vector $d(\mathbf{e}_1, \mathbf{w}_k) > 0$ for $k = 0, 1, \ldots, N_l - 2$.

(3) No vector can be orthogonal to the root vector $d(\mathbf{e}_1, \mathbf{w}_k) < 1$ for $k = 0, 1, \ldots, N_l - 2$.

Property 1 ensures that the local codebook contains the root vector. The structure of the root vector is used to define scaling and rotation operations. The presence of the root vector also ensures that the codeword used at the previous quantization step is also present, ensuring that distortion is non-increasing with increasing refinements. Property 2 means that no vector is parallel to the root vector. This is to ensure no redundancy and only a single root vector in the codebook. Property 3 ensures there are no orthogonal vectors to the root vector. The reason is that orthogonal vectors cannot be scaled, thus cannot be local.

The radius of the local codebook is used to define a measure of locality.

Definition 2 (codebook radius). The *radius* of a local codebook $\mathcal{S}$ is

$$\gamma_0 := \max_{\mathbf{s}_k \in \mathcal{S}, \mathbf{s}_k \neq \mathbf{e}_1} d(\mathbf{s}_k, \mathbf{e}_1). \tag{3}$$

Note that $\gamma_0 < 1$ from Definition 1. Essentially the radius is the smallest diameter of a ball centered around the root vector that covers all the elements of the local codebook.

Associated with the radius of the local codebook, we also need to define a notion of a covering radius.

Definition 3 (local covering radius). The covering radius of a local codebook $\mathcal{S}$ with radius γ_0 is defined as

$$c_l(\mathcal{S}) := \sup_{\mathbf{s}\ s.t. \|\mathbf{s}\|=1,\, d(\mathbf{s},\mathbf{e}_1) \leq \gamma_0} \min_{\mathbf{s}_k \in \mathcal{S}} d(\mathbf{s}, \mathbf{s}_k). \tag{4}$$

The radius of the codebook captures the overall region occupied by the local codebook while the covering radius captures the minimum radius of a ball that would cover all the Voronoi quantization regions for the codebook, defined in terms of subspace distance, without holes in the interior of the codebook. Note that from geometry it should be clear that $c_l(\mathcal{S}) < \gamma_0$.

An equivalent definition of the covering radius for a nonlocal codebook can also be defined, which we call $c(\mathcal{F})$. The main difference between the covering radius for a nonlocal and local codebook is that the latter is only computed for vectors that lie inside the radius of the codebook. The covering radius of the base codebook provides a bound on the radius of the local codebook. The local covering radius provides a bound on the amount of shrinking required during each stage of the proposed refinement algorithm.

4.2. Scaling a Local Codebook. We use the scaling function defined in [5, 6] to scale the vectors in the local codebook $\mathcal{S}$ to a new radius $\gamma\gamma_0$. Scaling is applied to the canonical local codebook centered around the root $\mathbf{e}_1$.

Definition 4 (scaling function). For $\mathbf{w} \in \mathbb{C}^{N_t \times 1}$ let $\mathbf{w} = \left[r_1 e^{j\theta_1} r_2 e^{j\theta_2} \cdots r_{N_t} e^{j\theta_{N_t}} \right]^T$. Define the vector scaling operation $\mathbf{s} : \mathbb{C}^{N_t \times 1} \times \mathbb{R}[0,1] \mapsto \mathbb{C}^{N_t \times 1}$ as

$$\mathbf{s}(\mathbf{w}, \alpha) = \begin{bmatrix} \sqrt{1 - \alpha^2(1 - r_1^2)}\, e^{j\theta_1} \\ \alpha r_2 e^{j\theta_2} \\ \vdots \\ \alpha r_{N_t} e^{j\theta_{N_t}} \end{bmatrix}. \tag{5}$$

The scaling operation preserves the unit norm property for $\alpha \in [0,1]$, that is, $\|\mathbf{s}(\mathbf{w}, \alpha)\| = 1$.

Definition 5 (scaled codebook). Define the scaled codebook function as

$$S(\mathcal{S}, \gamma) := \{\mathbf{e}_1, \mathbf{s}(\mathbf{w}_1, \gamma), \ldots, \mathbf{s}(\mathbf{w}_{N_l - 2}, \gamma)\}. \tag{6}$$

As established in the following Lemma, the scaling function scales the distance of the nonroot and root vectors by γ. Note that no guarantees are made about scaling of the distance between nonroot vectors.

Lemma 1 (radius of scaled codebook). *The scaling function in Definition 5 satisfies for* $\mathbf{w} \in S(\mathcal{S}, \gamma)$ *and* $\mathbf{w} \neq \mathbf{e}_1$,

$$d(\mathbf{e}_1, \mathbf{s}(\mathbf{w}, \gamma)) = \gamma d(\mathbf{e}_1, \mathbf{w}) = \gamma\gamma_0. \tag{7}$$

Proof. See [5, 6], for example. □

4.3. Rotating a Local Codebook. The codewords surround the generating vector $\mathbf{e}_1$. To perform a local quantization, it will be necessary to define a function that rotates a vector $\mathbf{v}$ to a vector $\mathbf{e}_1$ as well the rotation from $\mathbf{e}_1$ to $\mathbf{v}$. First let us define a unitary transformation from $\mathbf{e}_1$ to $\mathbf{v}$.

Definition 6 (center rotation). Let $\mathbf{U} : \mathbb{C}^{N_t \times 1} \mapsto \mathcal{U}^{N_t \times N_t}$ be the matrix function that determines a unitary matrix that rotates $\mathbf{e}_1$ to $\mathbf{v}$ thus $\mathbf{U}(\mathbf{v})\mathbf{e}_1 = \mathbf{v}$.

There are several ways to compute the rotation matrix using either the singular value decomposition [5, 6] or the complex Householder matrix [9] (as summarized here).

Example 1 (rotation with complex householder matrix [9]). Let $\mathbf{H}_{\text{ouse}} = \mathbf{I} - \mathbf{u}\mathbf{u}^*/\mathbf{u}^*\mathbf{e}_1$ where $\mathbf{u} := \mathbf{e}_1 - \mathbf{v}$ denote the complex Householder matrix [21]. The first column of $\mathbf{H}_{\text{ouse}}$ contains the entries of $\mathbf{v}$ while the remaining columns are orthogonal to $\mathbf{v}$. Further note that $\mathbf{H}_{\text{ouse}}$ is a unitary matrix. Thus if $\mathbf{U}(\mathbf{v}) = \mathbf{H}_{\text{ouse}}$ then $\mathbf{v} = \mathbf{U}(\mathbf{v})\mathbf{e}_1$ as required.

Definition 7 (codebook rotation function). Let the codebook rotation function as the function that applies the rotation $\mathbf{U}(\mathbf{v})$ to each entry of codebook $\mathcal{S}$ as follows:

$$T(\mathcal{S}, \mathbf{v}) = \{\mathbf{U}(\mathbf{v})\mathbf{e}_1, \mathbf{U}(\mathbf{v})\mathbf{w}_0, \mathbf{U}(\mathbf{v})\mathbf{w}_1, \ldots, \mathbf{U}(\mathbf{v})\mathbf{w}_{N_l - 2}\}. \tag{8}$$

The resulting codebook is rotated such that the first entry aligns with $\mathbf{v}$. Note that because of the unitary invariance of the subspace distance function, the rotation operation preserves the distance properties of the local codebook.

5. Preferred Local Codebooks

In this section we propose several local codebook designs and provide a general recipe for constructing local codebooks from a nonlocal codebook. The proposed local codebooks each have different features that make them attractive for progressive refinement including low complexity, reduced storage, or good distance properties.

5.1. Ring Codebook. The ring codebook is constructed from a collection of vectors that are equidistant from the centroid, conceptually illustrated in Figure 1(a). Ring codebooks have mathematical structure that permits certain simplifications in the progressive refinement algorithm. As such, in this section we introduce ring codebooks and discuss some of their mathematical properties.

Definition 8 (ring codebook). A ring codebook with radius $\gamma_0 < 1$ consists of $N_l - 1$ vectors $\{\mathbf{w}_n\}_{n=0}^{N_l-2}$ that are equidistant from the root vector $\mathbf{e}_1$. The nonroot entries of a ring codebook satisfy $d(\mathbf{w}_n, \mathbf{e}_1) = \gamma_0$ for $n = 0, 1, \ldots, N_l - 2$.

Lemma 2. *The first nonroot entry of the vectors of a ring codebook can be chosen to be equal to* $\sqrt{1-\gamma_0^2}$ *without loss of generality.*

Proof. Observe that $d(\mathbf{w}_n, \mathbf{e}_1) = \sqrt{1 - |[\mathbf{w}_n]_1|^2} = \gamma_0$ for $n = 0, 1, \ldots, N_l - 2$ thus $|[\mathbf{w}_n]_1| = \sqrt{1-\gamma_0^2}$ for all n. Since the subspace distance is phase invariant, that is, $d(\mathbf{a}, \mathbf{b}e^{j\theta}) = d(\mathbf{a}, \mathbf{b})$, the first entry $[\mathbf{w}_n]_1$ can be chosen to be real without loss of generality. □

Corollary 1. *The nonroot entries of a ring codebook with radius* γ_0 *can be chosen to have the following form:*

$$\mathbf{w}_k = \begin{bmatrix} \sqrt{1-\gamma_0^2} \\ \gamma_0 \widetilde{\mathbf{w}}_k \end{bmatrix}, \tag{9}$$

where $\widetilde{\mathbf{w}}_k$ *is a* $N_t - 1 \times 1$ *unit norm vector.*

We now summarize some general principles for constructing a ring codebook.

5.1.1. Uniform Phase Ring for $N_t = 2$. With $N_t = 2$, the nonroot codebook vectors can have the form

$$\mathbf{w}_\ell = \begin{bmatrix} \sqrt{1-\gamma_0^2} \\ \gamma_0 e^{j\theta_\ell} \end{bmatrix}. \tag{10}$$

A good ring codebook has elements on the ring that are far apart, in other words $\min_{k,\ell,k\neq\ell} d(\mathbf{w}_k, \mathbf{w}_\ell)$ is as large as possible. For the ring codebook with $N_t = 2$, $d^2(\mathbf{w}_k, \mathbf{w}_\ell) = 1 - |1 - \gamma_0^2 + \gamma_0^2 e^{j(\theta_k - \theta_\ell)}|^2$.

Using a little calculus, it is possible to see that the $N_l - 1$ roots of unity is one solution that maximizes the minimum distance. Thus we propose to take $\theta_\ell = 2\pi\ell/(N_l - 1)$ for $\ell = 0, 1, \ldots, N_l - 2$.

5.1.2. General Principles for Constructing a Ring Codebook for $N_t > 2$. Now let us consider the distance properties of the codewords on the ring to find some design principles for $N_t > 2$ that result in large $\min_{k,\ell,k\neq\ell} d(\mathbf{w}_k, \mathbf{w}_\ell)$. Using the notation in Corollary 1, note that

$$\begin{aligned} d^2(\mathbf{w}_k, \mathbf{w}_\ell) &= 1 - \left|\mathbf{w}_k^* \mathbf{w}_\ell\right|^2 \\ &= 1 - \left|1 - \gamma_0^2 + \gamma_0^2 \widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right|^2 \\ &= 1 - (1-\gamma_0^2)^2 - 2(1-\gamma_0^2)\gamma_0^2 \left|\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right| \\ &\quad \times \cos(\theta_{k,\ell}) - \gamma_0^4 \left|\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right|^2 \end{aligned} \tag{11}$$

where $\theta_{k,\ell} = \text{phase}(\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell)$.

Using the worst case value of $\cos\theta_{k,\ell} = 1$ it follows that

$$\begin{aligned} d^2(\mathbf{w}_k, \mathbf{w}_\ell) &\geq 1 - (1-\gamma_0^2)^2 - 2(1-\gamma_0^2)\gamma_0^2 \left|\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right| \\ &\quad - \gamma_0^4 \left|\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right|^2 \\ &= \gamma_0^4 \left(1 - \left|\widetilde{\mathbf{w}}_k^* \widetilde{\mathbf{w}}_\ell\right|^2\right). \end{aligned} \tag{12}$$

Since $\gamma_0 < 1$, maximizing the minimum absolute correlation maximizes the minimum of the lower bound in (12) over the collection of unit norm vectors $\{\widetilde{\mathbf{w}}_k\}$. This leads us to the following somewhat surprising observation that a Grassmannian codebook [4, 14] with vectors of length $N_t - 1$ can be used to build a ring codebook. Note, however, that the phase of the vectors plays a role in this case since we used the worst case phase to find the lower bound in (12). Suppose that a Grassmannian codebook of vectors with dimension $N_t - 1 \times 1$ is given by $\{\mathbf{g}_n\}_{n=0}^{N_l-2}$. We find that choosing $\widetilde{\mathbf{w}}_n = \mathbf{g}_n e^{j\phi_n}$ with $\phi_n = 2\pi\ell/(N_l - 1)$ tends to "randomize the phase" and give good performance.

One important question when constructing ring codebooks is how large should N_l be? For example, consider Figure 1(b), which shows a uniform phase ring with 11 points on the circle. Suppose that it had many points on the circle. As the number of points are increased, the Voronoi regions of the points on the circle would be narrow, like the spokes on a bicycle wheel; adding more points to the circle would not improve substantially quantization performance. Essentially the question for a fixed feedback size is how to tradeoff between the size of the local codebook and the number of refinements. In our simulation results in Section 7.1, we find that ring codebooks with a moderate number of points give the best performance.

5.2. Ring Codebooks Built from the Kerdock Codebook. Kerdock codebooks are structured beamforming codebooks [7], based on quaternary mutually unbiased bases [22] also known as Kerdock codes [23]. The Kerdock limited feedback codebook consists of the columns of multiple $N_t \times N_t$ unitary matrices $\mathcal{M} = \{\mathbf{M}_k\}_{k=0}^{M}$ that satisfy the mutually unbiased property $|\langle [\mathbf{M}_k]_{:,n}, [\mathbf{M}_\ell]_{:,m} \rangle| = 1/\sqrt{N_t}$ for all $n \in [1, N_t]$ and $m \in [1, N_t]$. We show how to design a ring codebook

with good distance properties from the Kerdock codebook after presenting several facts about collections of mutually unbiased matrices.

Summary 1 (properties of collections of mutually unbiased bases). (1) For a given N_t, at most $M+1$ bases where $M \leq N_t$ can be found that satisfy the mutually unbiased property, with equality when N_t is a prime or a power of a prime [24].

(2) A collection of mutually unbiased bases can be transformed to include the identity matrix. To see this note that if $\{\mathbf{M}_m\}_{m=0}^{M}$ are mutually unbiased bases then so are $\{\mathbf{M}_k^* \mathbf{M}_m\}_{m=0}^{M}$ for any $k = 0, 1, \ldots, M$. We refer to mutually unbiased bases that contain an identity matrix as transformed mutually unbiased bases.

(3) Let $\mathbf{n}$ and $\mathbf{m}$ denote two distinct columns of $\mathbf{M}_k \in \mathcal{M}$. Then $d(\mathbf{m}, \mathbf{n}) = 1$.

(4) Let $\mathbf{n}$ denote a column of $\mathbf{M}_k \in \mathcal{M}$ and $\mathbf{m}$ denote a column of $\mathbf{M}_\ell \in \mathcal{M}$ where $k \neq \ell$. Then $d(\mathbf{m}, \mathbf{n}) = \sqrt{1 - 1/N_t}$.

Definition 9 (kerdock ring codebook). Suppose that $\{\mathbf{M}_m\}_{m=0}^{M}$ are a transformed mutually unbiased bases with $\mathbf{M}_0 = \mathbf{I}$. The Kerdock ring codebook is constructed as

$$\mathcal{K} = \left\{ \mathbf{e}_1, [\mathbf{M}_1]_{:,1}, \ldots, [\mathbf{M}_1]_{:,N_t}, \ldots, [\mathbf{M}_M]_{:,1}, \ldots, [\mathbf{M}_M]_{:,N_t} \right\} \tag{13}$$

and has at most $MN_t + 1$ entries.

In constructing the Kerdock ring codebook, the only column of $\mathbf{M}_0$ present is the first one, $\mathbf{e}_1$, because the other columns are orthogonal to $\mathbf{e}_1$, which is forbidden by Definition 1. The radius of the Kerdock ring codebook is $\gamma_0 = \sqrt{1 - 1/N_t}$. The nonroot vectors satisfy

$$d(\mathbf{m}, \mathbf{n}) = \begin{cases} \gamma_0 & \text{for } \mathbf{m} \text{ and } \mathbf{n} \text{ in different bases} \\ 1 & \text{for } \mathbf{m} \text{ and } \mathbf{n} \text{ in the same basis.} \end{cases} \tag{14}$$

We conclude with some examples.

5.2.1. Kerdock Ring with $N_t = 2$. In this case, $M = N_t$ thus $N_l = 5$. Using the construction from [7], derived from [22], we obtain the codebook

$$\mathcal{S}_{N_t=2} = \left\{ \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ j \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -j \end{bmatrix} \right\}. \tag{15}$$

A further advantage of this codebook is that, to a scaling factor, the entries are plus/minus 1 or plus/minus j, which can be used to simplify computation.

5.2.2. Kerdock Ring with $N_t = 4$. For the case of $N_t = 4$, $M = N_t$ and $N_l = 17$. Using the construction from [7] derived from [25] gives the codebook

$$\begin{aligned} \mathcal{S}_{N_t=4} = \frac{1}{2} \Bigg\{ & \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ j \\ 1 \\ -j \end{bmatrix}, \begin{bmatrix} 1 \\ -j \\ 1 \\ j \end{bmatrix}, \begin{bmatrix} 1 \\ j \\ -1 \\ j \end{bmatrix}, \begin{bmatrix} 1 \\ -j \\ -1 \\ -j \end{bmatrix}, \\ & \begin{bmatrix} 1 \\ j \\ j \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ j \\ -j \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -j \\ -j \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ -j \\ j \\ 1 \end{bmatrix}, \\ & \begin{bmatrix} 1 \\ 1 \\ -j \\ j \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ j \\ -j \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ j \\ j \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ -j \\ -j \end{bmatrix}, \\ & \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix} \Bigg\}. \end{aligned} \tag{16}$$

Like the case of $N_t = 2$, this codebook also has plus/minus 1 or plus/minus j, which can be used to simplify computation.

5.3. General Procedure for Constructing a Local Codebook. While ring codebooks are attractive, and have a computational advantage discussed in the sequel, it will no doubt be of interest to construct other local codebooks either for other values of N_t or nonring codebooks. With this in mind, we present a technique for deriving a local codebook from any given codebook $\mathcal{F}$. This approach can be used to randomly generate a local codebook or to convert a Grassmannian codebook to a local codebook. Suppose that a codebook $\mathcal{F}$ is given. It is desired to construct a local codebook that satisfies all the requirements of Definition 1. This can be performed as follows.

(1) Rotate the codebook to the first entry $\mathbf{f}_0 \in \mathcal{F}$ so that $\mathbf{f}_0$ becomes the root vector $\mathbf{e}_1$. Of course, any entry can be chosen to become the root. Define the codebook

$$\mathcal{G} = \{ \mathbf{U}^*(\mathbf{f}_0)\mathbf{f}_0, \mathbf{U}^*(\mathbf{f}_0)\mathbf{f}_1, \ldots, \mathbf{U}^*(\mathbf{f}_0)\mathbf{f}_{N_l-1} \}. \tag{17}$$

The first entry of the resulting codebook is $\mathbf{e}_1$.

(2) To meet the requirements of Definition 1, remove any vectors that are orthogonal to $\mathbf{e}_1$ as required in Definition 1. Essentially this amounts to removing vectors with a zero in their first entry. This step is only required with special hand designed codes (as we did in constructing the Kerdock Ring code in Definition 9).

The resulting local codebook may not have good distance properties but this construction can be used as an aid in the design of numerical algorithms for finding good local codebooks.

6. Progressive Refinement Algorithms

In this section we explain the progressive refinement algorithm described in Section 3 in more detail. We discuss how symmetry in the distance function and structure in ring codeboks can be used to reduce computation. Finally, we comment on selection of the contraction radius.

6.1. Basic Algorithm. Consider a minimum distance quantization function $Q(\mathbf{h}, \mathcal{F})$ that produces an element of $\mathcal{F}$ from channel $\mathbf{h} = \mathbf{h}_u$ observed by user u. We assume the quantizer implements the function described in (2). Suppose that a total of R refinements are desired. At each refinement level r, let $l(r)$ denote the scaling of the local codebook (scaling is discussed in Section 6.3). Using this notation, the basic progressive refinement algorithm is described as follows.

Algorithm 1 (progressive Refinement). (i) Perform the initial quantization step and let $\mathbf{f}[0] = \mathcal{Q}(\mathbf{h}, \mathcal{F})$.

(ii) Let $\mathbf{c}[1] = \mathbf{f}[0]$ denote the desired centroid for the first refinement.

(iii) Let $r = 1, 2, \ldots, R$ denote the refinement level. For each refinement r:

(a) form the scaled codebook $\mathcal{S}_s = S(\mathcal{S}, l(r))$;
(b) form the rotated codebook $\mathcal{S}_t = T(\mathcal{S}_s, \mathbf{c}[r])$;
(c) let the rth refinement be $\mathbf{f}[r] = \mathcal{Q}(\mathbf{h}, \mathcal{S}_t)$;
(d) update the centroid $\mathbf{c}[r+1] = \mathbf{f}[r]$;

(iv) The final refinement is $\mathbf{f}[R]$.

The basic algorithm requires storing the base and local codebook. The complexity of the base quantization step is due to the search over the entries of $\mathcal{F}$. Each refinement step requires $N_l - 1$ rotation and scaling operations, not to mention a search over N_l entries to perform the quantization. The scaling operations could be avoided by storing multiple codebooks for each scaling, but this increases the memory requirements.

Quantization using progressive refinement is comparable to quantization with an effectively larger codebook. Of course quantizing with the proposed algorithm involves a constrained search so it is not exactly the same as quantizing with the corresponding compound codebook. The effective codebook size assuming R refinement steps is

$$N_{\text{effective}} = |\mathcal{F}||\mathcal{S}|^R \tag{18}$$

and the amount of feedback (assuming independent coding of the base and refinement operations) is $\log_2|\mathcal{F}| + R\log_2|\mathcal{S}|$.

Notice that the amount of feedback depends on the number of refinements R in the algorithm. If users are operating at different SNR levels, it may be desirable to allocate different sized codebooks to each user. This can be performed easily by assigning different numbers of refinement steps to each user.

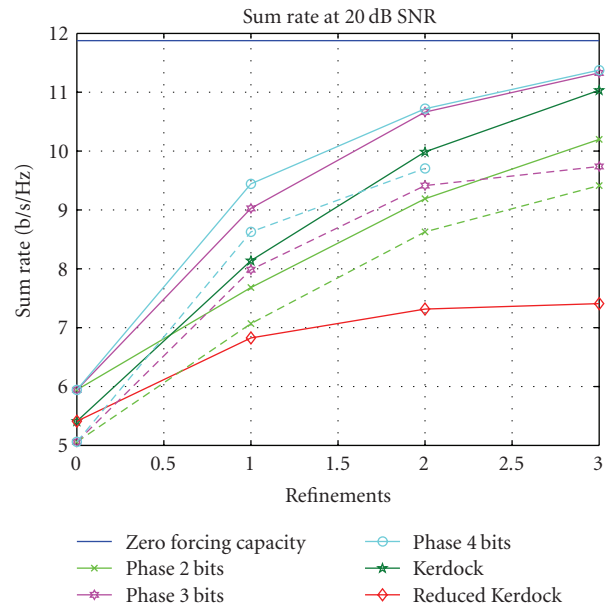

FIGURE 3: Sum rate performance of multiuser MIMO with $N_t = U = 2$ at SNR = 20 dB for different ring codebooks. For comparison the dashed lines correspond to sum rates achieved with random vector quantization with a total codebook size corresponding to the phase base plus refinements.

6.2. Complexity Reduction. In Algorithm 1, the entire local codebook is rotated during each refinement. Reduced complexity, though, is possible by recognizing symmetry in the quantization function. First notice that $d(\mathbf{h}/\|\mathbf{h}\|, \mathbf{U}(\mathbf{c}[r])\mathbf{w}) = d(\mathbf{U}^*(\mathbf{c}[r])\mathbf{h}/\|\mathbf{h}\|, \mathbf{w})$. Thus $\mathcal{Q}(\mathbf{h}, \mathcal{S}_t) = \mathcal{Q}(\mathbf{U}^*(\mathbf{c}[r])\mathbf{h}, \mathcal{S}_s)$. Consequently, *the codebook does not actually have to be rotated*. It suffices to rotate the observation to match the canonical local codebook with root $\mathbf{e}_1$.

Algorithm 2 (progressive refinement with rotated observation). (i) Perform the initial quantization step and let $\mathbf{f}[0] = \mathcal{Q}(\mathbf{h}, \mathcal{F})$.

(ii) Let $\mathbf{c}[0] = \mathbf{f}[0]$ denote the initial centroid.

(iii) Initialize the rotation matrix $\mathbf{V}[0] = \mathbf{I}$.

(iv) Let $r = 1, 2, \ldots, R$ denote the refinement level. For each refinement r:

(a) form the scaled codebook $\mathcal{S}_s = S(\mathcal{S}, l(r))$.
(b) form the rotation matrix $\mathbf{V}[r] = \mathbf{U}(\mathbf{V}[r-1]\mathbf{c}[r-1])$.
(c) update the centroid $\mathbf{c}[r+1] = \mathcal{Q}(\mathbf{V}^*[r]\mathbf{h}, \mathcal{S}_s)$.

(v) the final refinement is $\mathbf{V}[R]\mathbf{c}[R]$.

A main observation about this algorithm is that a rotation matrix needs to be updated based on the previous rotated refinement. In terms of rotation computations, it requires $2R + 1$ rotations versus the $R(N_l - 1)$ rotations required by the basic algorithm.

To further reduce complexity, it would be nice to also avoid rescaling the codebook. The rescaling operation though is more delicate due to nonlinear transformation of r_1 in (5). This can be simplified though for ring codebooks

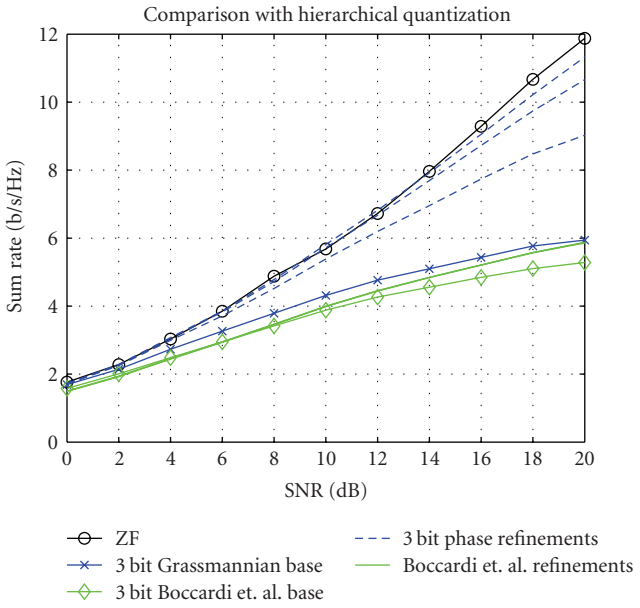

FIGURE 4: Sum rate performance comparison of multiuser MIMO with $N_t = U = 2$ in an uncorrelated Rayleigh channel. The dashed lines correspond to increasing numbers of 3 bit phase refinements while the solid lines with no markers correspond to increasing numbers of refinements for the Boccardi et. al. algorithm (which happen to overlap).

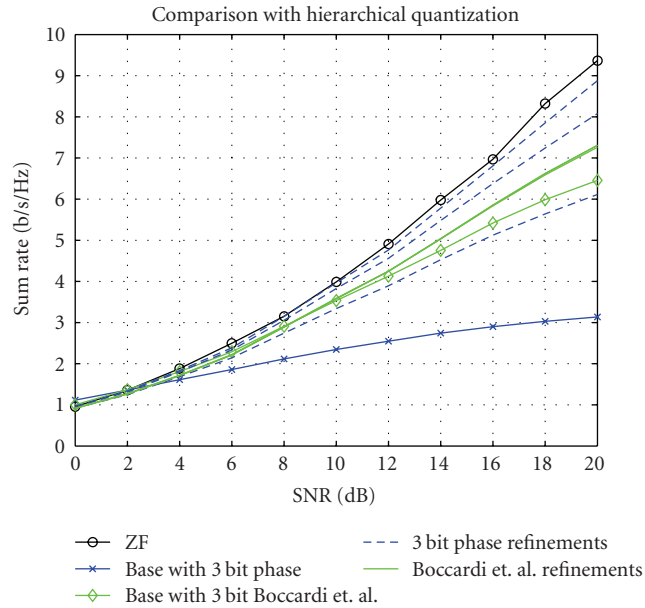

FIGURE 5: Sum rate performance comparison of multiuser MIMO with $N_t = U = 2$ in a spatially correlated channel. The dashed lines correspond to increasing numbers of 3−bit phase refinements while the solid lines with no markers correspond to increasing numbers of refinements for the Boccardi et. al. algorithm (which happen to overlap).

exploiting $[\mathbf{w}_k]_1 = \sqrt{1-\gamma_0^2}$ and the use of a dual scaling function.

Definition 10 (alternate scaling for ring codebooks). For $\mathbf{w} \in \mathbb{C}^{N_t \times 1}$ let $\mathbf{w} = [w_1, \tilde{\mathbf{w}}^T]^T$. Define the vector scaling operation $\mathbf{t} : \mathbb{C}^{N_t \times 1} \times \mathbb{R} \mapsto \mathbb{C}^{N_t \times 1}$ as

$$\mathbf{t}(\mathbf{w}, \alpha) := \begin{bmatrix} \sqrt{\dfrac{1-\alpha^2\gamma_0^2}{1-\gamma_0^2}}\, w_1 \\ \alpha\tilde{\mathbf{w}} \end{bmatrix}. \tag{19}$$

With this revised scaling algorithm we have the following lemma.

Lemma 3. *For the scaling operation in Definition 10,* $\mathbf{w}_n$ *from a local ring codebook, and unit norm* $\mathbf{v}$

$$d(\mathbf{s}(\mathbf{w}_n, \alpha), \mathbf{v}) = d(\mathbf{w}_n, \mathbf{t}(\mathbf{v}, \alpha)). \tag{20}$$

Proof. follows by direct substitution using the ring structure in Lemma 2. □

Using this novel scaling function, a new algorithm is described with even lower complexity, specifically for ring codebooks.

Algorithm 3 (progressive refinement with rotated and scaled observation). (i) Perform the initial quantization step and let $\mathbf{f}[0] = \mathcal{Q}(\mathbf{h}, \mathcal{F})$.

(ii) Let $\mathbf{c}[0] = \mathbf{f}[0]$ denote the initial centroid.

(iii) Initialize the rotation matrix $\mathbf{V}[0] = \mathbf{I}$.

(iv) Let $r = 1, 2, \ldots, R$ denote the refinement level. For each refinement r:

(a) form the scaled input vector $\mathbf{h}_t = \mathbf{t}(\mathbf{h}/\|\mathbf{h}\|, l(r))$.

(b) form the rotation matrix $\mathbf{V}[r] = \mathbf{U}(\mathbf{V}[r-1]\mathbf{c}[r-1])$.

(c) update the centroid $\mathbf{c}[r+1] = \mathcal{Q}(\mathbf{V}^*[r]\mathbf{h}_t, \mathcal{S}_s)$.

(v) the final refinement is $\mathbf{V}[R]\mathbf{c}[R]$.

Algorithm 3 exploits the ring structure of a local codebook to remove the codebook scaling requirement in Algorithm 2, saving $R(N_l - 1)$ scaling operations. Note that the scaling operation does not impact the reconstruction in any way.

Other codebook structures facilitate further complexity reductions. If the complex Householder matrix is used to compute the rotation of a ring codebook, then for $\mathbf{w}_k \neq \mathbf{e}_1 \in \mathcal{S}$

$$\mathbf{H}_{\text{ouse}}(\mathbf{v}) = \mathbf{I} - \frac{1}{1-\sqrt{1-\gamma_0^2}}(\mathbf{e}_1 - \mathbf{w}_k)(\mathbf{e}_1 - \mathbf{w}_k)^* \tag{21}$$

where $\mathbf{e}_1 - \mathbf{w}_k$ can be computed simply by recognizing that the first coefficient is $\sqrt{1-\gamma_0^2} - 1$ so the subtraction is not actually required and the normalization factor is a constant.

The nature of the entries of the codebook can also be used to reduce complexity. For example the quartenary structure of the Kerdock ring codebook can be used to compute the inner product used in the distance function between $\mathbf{h}_t$ and $\mathbf{w} \in \mathcal{W}$ without actually doing any multiplies. These computational advantages motivate the use of ring codebooks in general, and specifically the preferred codebooks that we suggested.

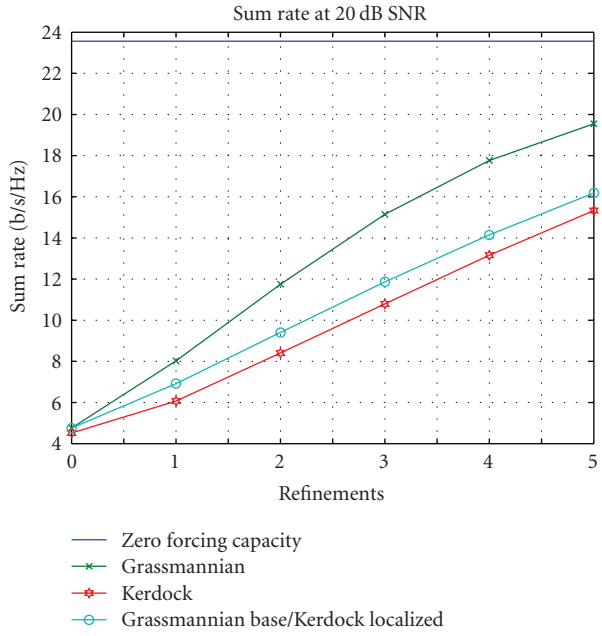

FIGURE 6: Sum rate performance of multiuser MIMO with $N_t = U = 4$ at SNR = 20 dB for the Grassmannian, Kerdock, and Grassmannian/Kerdock ring codebooks.

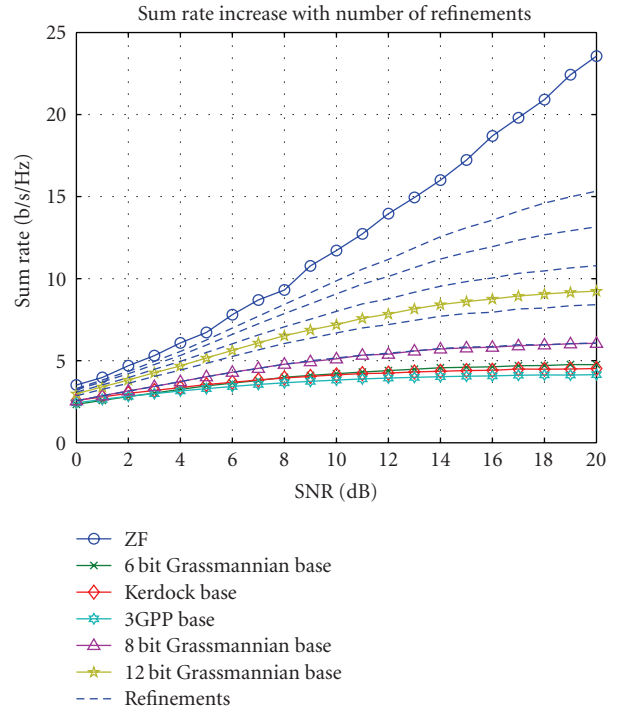

FIGURE 7: Sum rate performance of multiuser MIMO with $N_t = U = 4$ for the $N_l = 64$ Grassmannian base and ring codebook compared to various base codebooks. The sum rate increases with each refinement, with an error floor at higher SNR. The first refinement overlaps the 8 bit Grassmannian base curve.

To summarize, rotation of the local codebook is not required, saving $R(N_l - 1)$ rotation operations. For ring codebooks, scaling of the local codebooks is also not required, saving $R(N_l - 1)$ scaling operations or an equivalent amount of memory, depending on how the scaling is implemented. By avoiding the rotation and scaling operations, the structure in the local codebook can be employed to further reduce hardware implementation complexity.

6.3. Radius Selection. An important question associated with the proposed progressive refinement algorithm is the choice of the scaling radius $l(r)$ during refinement step r. Scaling the radius too aggressively can cause an error floor while not scaling aggressively enough will require an excessive number of refinements to reach a target average distortion. Even more fundamentally, does there exist a sequence of radii $\{l(r)\}$ that reduces quantization error as R grows large? This question is answered in the following theorem.

Proposition 1. *Given a base codebook $\mathcal{F}$ with covering radius $c(\mathcal{F}) < 1$ and a local codebook $\mathcal{S}$ with local covering radius $c_l(\mathcal{S})$, there exists a sequence of radii $\{l(r)\}$ that guarantees the quantization error is decreasing.*

Proof. We provide a sketch of the proof. Consider an observation given by $\mathbf{h}$. Suppose that $\mathbf{h}$ is quantized to $\mathbf{f}_k$ with the base codebook. Now define a ball $B_\delta(\mathbf{x})$ of radius δ and center $\mathbf{x} \in G(N_t, 1)$ using subspace distance. From the definition of covering radius, a ball of radius $\delta \geq c(\mathcal{F})$ with center $\mathbf{f}_k$ covers the Voronoi region of $\mathbf{f}_k$ for the minimum distance quantizer. Thus the maximum error is less than $c(\mathcal{F})$. Suppose that $l(1) = c(\mathcal{F})/\gamma_0$ (the γ_0 is required since the local codebook radius is by default γ_0 but this can be adjusted by an initial scaling). Then the local codebook covers the Voronoi region of $\mathbf{f}_k$. For refinement r let $\mathcal{S}_r$ denote the scaled local codebook $\mathcal{S}_r = S(\mathcal{S}, l(r))$ and let $l(r+1) = c_l(\mathcal{S}_r)$. Since the covering radius of the local codebook is strictly less than the codebook radius at each r, the maximum error is decreasing. □

It follows from Proposition 1 that with appropriate selection of $l(r) \geq c_l(\mathcal{S}_r)$ and $l(r+1) < l(r)$, the maximum quantization error will eventually go to zero since at every step the rescaled local codebook completely covers the Voronoi region from the previous quantization and that all observations in this Voronoi region are inside the radius defined by the next shrunk local codebook. Choosing the smallest possible $l(r)$ ensure the most aggressive refinement and the fastest potential convergence.

Calculating the local covering radius is challenging. For the first refinement, the minimum distance of the base codebook $d_{\min}(\mathcal{F})/2$ is a lower bound for the covering radius while 1 can be taken as an upper bound. For subsequent refinements, the minimum distance of the local codebook $d_{\min}(\mathcal{S}_r)/2$ is a lower bound for the covering radius while γ_r, the radius of $\mathcal{S}_r$, is an upper bound on the covering radius, measured by the distance from the centroid to the furthest quantization point. These bounds provide a range over which to search for an appropriate scaling $l(r)$ for each r, based on $l(r-1)$. Because it is difficult to calculate the

covering radius for either the base or local codebooks exactly, we propose to use a greedy numerical method to optimize the radius at each step.

Given $\{l(1), \ldots, l(r-1)\}$ are already determined, we propose to simulate numerically the sum rate performance through 10000 simulations of an i.i.d. Rayleigh fading channel at a target high SNR (say 20 dB) and choose the best radius. Note that the ad hoc and greedy nature of the radius computation is not a serious deficiency of the algorithm since the sequence of radii $\{l(r)\}$ are computed offline and would be known to both the transmitter and receiver. In fact, such ad hoc calculations are used in the vector quantization in the design of tree-structured vector quantizers [26]. Optimizing using the uncorrelated channel is reasonable since the correlation is not known a priori, though it could be used to dynamically adjust the radius (we do not pursue this due to lack of space).

7. Simulations

In this section we present several simulation results to illustrate the performance of the proposed local codebooks and progressive refinement algorithm. As with related papers on multiuser MIMO [2], we compute the sum rate under the assumption that all users experience the same average SNR E_s/N_o as

$$C\left(\frac{E_s}{N_o}\right) = \sum_{u=1}^{U} \log_2(1 + \mathrm{SINR}_u), \tag{22}$$

where the SINR (signal-to-interference-plus-noise ratio) at the uth user is given by

$$\mathrm{SINR}_u := \frac{(E_s/UN_o)\left|\mathbf{h}_u^T \mathbf{f}_u\right|^2}{1 + (E_s/UN_o)\sum_{k \neq u}\left|\mathbf{h}_u^T \mathbf{f}_k\right|^2}. \tag{23}$$

The interference is a byproduct of quantization error: with quantization the zero-forcing solution does not perfectly cancel interference. The sum rate in (22) is a genie-aided performance measure since it assumes the rate for each user is chosen based on the measured SINR_u. This is realizable assuming that pilots are sent over the chosen beamforming vectors to measure SINR_u as in most commercial wireless systems. Further we assume that $N_t = U$ and that there are 4,000 Monte Carlo simulations for each SNR point. The numerically optimized radius values listed in Table 1 were used for each codebook configuration.

7.1. Two Transmit Antennas and Two Users. First we study the impact of increasing refinements on the sum rate at 20 dB average SNR. We compare the phase ring codebook in Section 5.1.1 with $N_l = 4, 8, 16$ (corresponding to 2, 3, and 4 bits), the Kerdock ring codebook in Section 5.2.1 with $N_l = 5$, a variation where only the vectors from one basis are chosen with $N_l = 3$. We use the $N = 8$ vector Grassmannian codebook [27] for the base codebook for the phase ring while we use the Kerdock codebook for the base codebook with the Kerdock ring. From Figure 3, we see that performance increases with increasing refinement levels. Now if the total feedback size is fixed, what is the right distribution between local codebook size and number of refinements? This is difficult to answer in general. Comparing the performance of the 4 bit local codebook for one refinement and the 2 bit local codebook with two refinements, one refinement with a larger codebook is better than two with a smaller codebook. We do not expect this trend to continue with larger local codebook sizes because there are diminishing returns. For example, with the 3 and 4 bit codebooks have similar performance for larger numbers of refinements. Intuitively this is because the ring becomes more dense and the distance between codebook vectors on the ring become much closer than the radius of the ring. The Kerdock code with $N_l = 5$ outperforms the 2 bit phase codebook and approaches the 3 bit codebook with more refinements. Notice also that the Kerdock codebook needs all the vectors to work efficiently - using only 3 (removing one basis) substantially reduces the performance.

One relevant question is how does progressive refinement compare with using codebooks of fixed dimension but with the same number of feedback bits? Unfortunately, optimized codebooks are not readily available for larger codebooks sizes. Consequently we compare with random vector quantization [28], where performance is averaged over randomly generated codebooks. Random vector quantization has been used in the analysis of multiuser MIMO [2], and is a lower bound on what can be achieved with optimized codebooks. In Figure 3, we plot the sum rate performance of random vector quantization in dashed lines with same feedback size as the corresponding three phase codebooks. For example, the *total* feedback with the $N_l = 4$ three phase codebook is 3 bits for the base quantization, 5 bits for the first refinement, 7 bits for the second refinement, and 9 bits for the final refinement. We compare with random vector quantization with the corresponding codebook dimensions in Figure 3. In each case we see that the phase codebooks outperform random vector quantization for total feedback constraint.

Now we compare the sum rate performance versus SNR of the proposed progressive refinement operation with different numbers of refinements with the hierarchical quantization proposed by Boccardi et al. in [11]. For the Boccardi algorithm, we use a codebook size of 8 to compare with the $N_l = 8$ uniform phase codebook. With these parameters, we require 3 bits per refinement while the Boccardi algorithm actually requires 4 bits (since there are 9 possibilities at each level). In Figure 4, we see that the Boccardi algorithm provides only marginal performance improvement as the number of levels in the hierarchy are increased. The reason for this is that the Boccardi algorithm uses a DFT codebook, which has poor subspace distance properties but has a structure that is better suited for correlated channels.

To demonstrate performance in correlated channels, we consider transmit correlation with a single cluster for each user, truncated Laplacian power azimuth spectrum, uniform linear array, and half-wavelength element spacing [29]. The first user has an angle of departure $\pi/4$ and angle spread $\pi/16$, while the second user has angle of departure of $\pi/2$ and angle spread $\pi/16$. The corresponding

Table 1: Numerically optimized radius values.

Codebook name	Optimized radius values
$N_t = 2$, $N = 4, 8, 16$, uniform phase ring codebook	$\{0.35, 0.18, 0.09\}$
$N_t = 2$, Kerdock ring codebook	$\{0.4, 0.2, 0.1\}$
$N_t = 2$, Kerdock ring w/ one basis	$\{0.45, 0.2, 0.1\}$
$N_t = 4$, Kerdock base, Kerdock ring	$\{0.5, 0.25, 0.2, 0.15, 0.1, 0.075, 0.05\}$
$N_t = 4$, Grassmannian base, Kerdock ring	$\{0.4, 0.25, 0.175, 0.125, 0.09, 0.06\}$
$N_t = 4$, Grassmannian base, Grassmannian local	$\{0.4, 0.25, 0.175, 0.125, 0.09, 0.06\}$

results are illustrated in Figure 5. Notice in this case that the base refinement with the Boccardi algorithm performs much better than the Grassmannian base codebook. The reason is that the channel is highly correlated with a poorly conditioned correlation matrix. The local codebook is able to adapt, achieving the same performance as the base Boccardi algorithm with just one refinement. Subsequent levels of the Hierarchical approach from the Boccardi algorithm do not yield substantial improvements while the progressive approach is able to zoom in on the channel estimation, more closely approaching the unquantized sum capacity.

7.2. Four Transmit Antennas and Four Users. Now we consider the more challenging case of $N_t = U = 4$ under the same simulation assumptions as before. For this case we consider three different scenarios. First we use the full Kerdock codebook with $N = 20$ entries as the base codebook and the Kerdock ring codebook described in Section 5.2.2 for the local codebook. Second we consider the 6 bit Grassmannian codebook [27] for the base codebook paired with the Kerdock ring codebook described in Section 5.2.2. Finally we consider the 6 bit Grassmannian codebook [27] for the base codebook paired with a local codebook derived from the base codebook according to the procedure in Section 5.3. Five refinements were considered in each case with numerically optimized refinement values provided in Table 1. The Kerdock refinements require 5 bits while Grassmannian refinements require 6 bits each. We do not compare with the Boccardi strategy due to the complexity of our implementation of the Boccardi approach.

We compare the performance of the different progressive refinement approaches at an average SNR of 20 dB as a function of increasing refinement levels in Figure 6. The Grassmannian base codebook with Kerdock refinements outperforms the Kerdock base codebook with Kerdock refinements since it starts with a better initial quantization. The Grassmannian codebook with Grassmannian codebook refinements outperforms both cases with Kerdock codebook refinements. In part this is due to the fact that it has a larger size ($N_l = 64$ versus $N_l = 17$) and also since it is more dense. The main penalty is that Grassmannian refinements require higher complexity to compute, since they cannot take advantage of the ring structure to reduce the number of scaling operations.

Now we compare the sum rate performance versus SNR with different fixed sized codebooks in Figure 7. We use the Grassmannian base and local codebooks, since they give the best performance, and compare with the 6 bit Grassmannian codebook, the 3GPP codebook LTE 4 bit codebook [19], an 8 bit near Grassmannian codebook, and a 12 bit near Grassmannian codebook. We see that the 6 bit base codebook and one 6 bit refinement gives approximately the same performance as an 8 bit near-Grassmannian codebook (the lines almost exactly overlap). Three refinements are required to beat the 12 bit Grassmannian codebook, at a penalty of an extra 12 bits. The performance difference is not unexpected—performance penalties are common in the implementation of structured vector quantizers [26] and residual vector quantizers [12]. Nonetheless, the complexity with the proposed progressive refinement algorithm is reduced, requiring in this example $42^6 = 2^8$ searches and some additional scaling and rotation operations instead of a search over a 2^{12} dimension codebook, not to mention the memory savings.

8. Conclusions and Future Work

In this paper we proposed a progressive refinement algorithm that refines an initial quantization from a base codebook using progressively smaller local codebooks to achieve high-resolution quantization of beamforming vectors in multiuser MIMO beamforming systems. We discussed several criteria for designing local codebooks and presented a number of constructions for two and four transmit antennas. Monte Carlo simulations confirm that the proposed algorithms provide a flexible means of increasing quantizer resolution using multiple refinement levels.

There are several directions for future work. While we considered the specific application to multiuser MIMO it should be clear that the algorithm can be extended to single user MIMO by changing the quantization function. Throughout the paper we assumed the channel was static but it is also of interest to use progressive algorithms in time varying channels. Extending the MISO analysis in [9] to our case or the hierarchical algorithm that adjusts the level based on the channel variation in [11] seem to be promising directions of future research. We assumed all the users had the same average SNR, which may not be true in practice. A leverage of our algorithm is that users can be assigned different effective codebook sizes based on their average SNR (smaller codebooks for lower SNRs, bigger codebooks for higher SNRs). Studying sum feedback rate tradeoffs in

this context seems to be promising. Unlike the hierarchical DFT based codebook in [11], the proposed codebook with refinements does not satisfy the constant modulus property, which incurs a peak-to-average power ratio penalty. An interesting topic of future research is to find local codebooks that also have near constant modulus property. Finally, it would be interesting to investigate structured nonring codebooks that retain the complexity reduction properties of ring codebooks.

Acknowlegment

Work done while the first author was consulting with Huawei Technologies.

References

[1] D. J. Love, R. W. Heath Jr., V. K. N. Lau, D. Gesbert, B. D. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 8, pp. 1341–1365, 2008.

[2] N. Jindal, "MIMO broadcast channels with finite-rate feedback," *IEEE Transactions on Information Theory*, vol. 52, no. 11, pp. 5045–5060, 2006.

[3] T. Yoo, N. Jindal, and A. Goldsmith, "Multi-antenna downlink channels with limited feedback and user selection," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 7, pp. 1478–1491, 2007.

[4] D. J. Love, R. W. Heath Jr., and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2735–2747, 2003.

[5] R. Samanta and R. W. Heath Jr., "Codebook adaptation for quantized MIMO beamforming systems," in *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, pp. 376–380, October-November 2005.

[6] V. Raghavan, R. W. Heath Jr., and A. M. Sayeed, "Systematic codebook designs for quantized beamforming in correlated MIMO channels," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 7, pp. 1298–1310, 2007.

[7] T. Inoue and R. W. Heath Jr., "Kerdock codes for limited feedback MIMO systems," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '08)*, pp. 3113–3116, Las Vegas, Nev, USA, March-April 2008.

[8] B. Mondal, T. A. Thomas, and M. Harrison, "Rank-independent codebook design from a quaternary alphabet," in *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, pp. 297–301, Pacific Grove, Calif, USA, November 2007.

[9] L. Liu and H. Jafarkhani, "Novel transmit beamforming schemes for time-selective fading multiantenna systems," *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4767–4781, 2006.

[10] L. Liu and H. Jafarkhani, "Successive transmit beamforming algorithms for multiple-antenna OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1512–1522, 2007.

[11] F. Boccardi, H. Huang, and A. Alexiou, "Hierarchical quantization and its application to multiuser eigenmode transmissions for MIMO broadcast channels with limited feedback," in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, pp. 1–5, Athens, Greece, September 2007.

[12] C. F. Barnes, S. A. Rizvi, and N. M. Nasrabadi, "Advances in residual vector quantization: a review," *IEEE Transactions on Image Processing*, vol. 5, no. 2, pp. 226–262, 1996.

[13] B. Mondal, S. Dutta, and R. W. Heath Jr., "Quantization on the Grassmann manifold," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4208–4216, 2007.

[14] K. K. Mukkavilli, A. Sabharwal, E. Erkip, and B. Aazhang, "On beamforming with finite rate feedback in multiple-antenna systems," *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2562–2579, 2003.

[15] B. M. Hochwald, T. L. Marzetta, T. J. Richardson, W. Sweldens, and R. Urbanke, "Systematic design of unitary space-time constellations," *IEEE Transactions on Information Theory*, vol. 46, no. 6, pp. 1962–1973, 2000.

[16] D. J. Love and R. W. Heath Jr., "Limited feedback unitary precoding for orthogonal space-time block codes," *IEEE Transactions on Signal Processing*, vol. 53, no. 1, pp. 64–73, 2005.

[17] IEEE, "IEEE 802.16e-2005 ammendment".

[18] 3GPP, "3rd Generation Partnership Project physical layer standard," http://www.3gpp.org.

[19] e. R1-072235, Samsung, "Codebook design for 4Tx SU MIMO," *3GPP TSG RAN WG1 49*, Kobe, Japan, May 2007, http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_49/Docs/R1-072235.zip.

[20] 3GPP2, "3rd Generation Partnership Project 2 physical layer standard," http://www.3gpp2.org.

[21] K. L. Chung and W. M. Yan, "The complex householder transform," *IEEE Transactions on Signal Processing*, vol. 45, no. 9, pp. 2374–2376, 1997.

[22] A. R. Calderbank, P. J. Cameron, W. M. Kantor, and J. J. Seidel, "Z_4-Kerdock codes, orthogonal spreads, and extremal euclidean line-sets," *Proceedings of the London Mathematical Society*, vol. 75, no. 2, pp. 436–480, 1997.

[23] A. Kerdock, "Studies of low-rate binary codes," *IEEE Transactions on Information Theory*, vol. 18, no. 2, p. 316, 1972.

[24] A. Klappenecker and M. Roetteler, "Constructions of mutually unbiased bases," *Finite Fields and Applications*, pp. 137–144, 2004.

[25] R. Gow, "Generation of mutually unbiased bases as powers of a unitary matrix in 2-power dimensions," http://arXiv.org/abs/math/0703333v2.

[26] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Springer, New York, NY, USA, 1991.

[27] D. J. Love, "Grassmannian subspace packing," http://cobweb.ecn.purdue.edu/~djlove/grass.html.

[28] W. Santipach and M. L. Honig, "Capacity of a multiple-antenna fading channel with a quantized precoding matrix," *IEEE Transactions on Information Theory*, vol. 55, no. 3, pp. 1218–1234, 2009.

[29] R. Bhagavatula and R. W. Heath Jr., "Computing the receive spatial correlation for a multi-cluster MIMO channel using different array configurations," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '08)*, pp. 3959–3963, 2008.