Pllaha, Tefjol; Heikkila, Elias; Calderbank, Robert; Tirkkonen, Olav

Low-Complexity Grassmannian Quantization Based on Binary Chirps

Published in:
2022 IEEE Wireless Communications and Networking Conference, WCNC 2022

DOI:
10.1109/WCNC51071.2022.9771694

Published: 16/05/2022

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Low-Complexity Grassmannian Quantization Based on Binary Chirps

Tefjol Pllaha *, Elias Heikkilä †, §, Robert Calderbank ‡, Olav Tirkkonen †
* University of Nebraska, Lincoln, NE, USA. E-mail: tefjol.pllaha@unl.edu
† Aalto University, Helsinki, Finland. E-mails: {elias.heikkila, olav.tirkkonen}@aalto.fi
‡ Duke University, Durham, NC, USA. E-mail: robert.calderbank@duke.edu
§ Nordic Semiconductor, Espoo, Finland. Email: elias.heikkilae@nordicsemi.no

***Abstract*—We consider autocorrelation-based low-complexity decoders for identifying Binary Chirp codewords from noisy signals in $N = 2^m$ dimensions. The underlying algebraic structure enables dimensionality reduction from N complex to m binary dimensions, which can be used to reduce decoding complexity, when decoding is successively performed in the m binary dimensions. Existing low-complexity decoders suffer from poor performance in scenarios with strong noise. This is problematic especially in a vector quantization scenario, where quantization noise power cannot be controlled in the system. We construct two improvements to existing algorithms; a geometrically inspired algorithm based on successive projections, and an algorithm based on adaptive decoding order selection. When combined with a breadth-first list decoder, these algorithms make it possible to approach the performance of exhaustive search with low complexity.**

I. Introduction

Subspaces of unit norm complex valued vectors modulo overall phase rotations are of interest in many information processing tasks, ranging from non-coherent wireless communication [1], [2], [3] and activity detection [4] to vector quantization of Channel State Information (CSI) [5], [6], [7], [8] and deep learning [9], while also being the fundamental objects of interest in finite dimensional quantum mechanics [10]. For communication [2], [3], quantization [5], [7], [8] and quantum coding [11] scenarios, codebook based approaches are essential.

Binary Chirps (BCs) [12] are codebooks of complex unit norm vectors in $N = 2^m$ dimensions. BCs have been explicitly used for CSI quantization in both multiuser [6] and single user [13] scenarios. In a communication scenario, transmitting codewords describing subspaces is justified in a non-coherent setting, when the channel is unknown to the receiver, so that information cannot be transmitted with the overall phase or amplitude of the signal. In this context, BCs, a.k.a. Reed-Muller sequences, have been recently used both for unsourced [2] and massive [14] random access.

The algebraic structure of BCs enables low-complexity decoding algorithms with complexity $\mathcal{O}(m(m+1)N)$. These algorithms are based on sequential search in the m binary dimensions. Unfortunately, in scenarios with strong noise, the low-complexity algorithms are prone to error. Thus, if BCs were to be used in, e.g., vector quantization, where noise is very strong, the low-complexity decoding algorithms of prior art cannot be used.

It is the objective of this paper to improve the low-complexity decoding algorithms such that they can cope with strong noise. We first develop a geometric successive projection algorithm, which reduces the noise contribution when decoding proceeds sequentially in the m binary dimensions. We also devise an algorithm to select the order to search over the binary dimensions. Finally, to increase the probability of finding the correct combination in the sequential search, we develop a breadth first tree search [15] algorithm to address BC decoding.

II. System Model

We consider a signal model of the form

$$\mathbf{z} = \mathbf{w} + \mathbf{n}\,, \tag{1}$$

where the codeword $\mathbf{w}$ is an $N = 2^m$-dimensional unit norm complex vector, while $\mathbf{n}$ represents noise. The model covers both a quantization and a non-coherent communication scenario. The codewords come from a Binary Chirp codebook, discussed below. The phase and amplitude of $\mathbf{w}$, and thus of $\mathbf{z}$ are irrelevant. The codewords thus live in complex projective space $\mathbb{CP}^{N-1}$, or equivalently, they represent complex Grassmannian lines living in $\mathcal{G}_{\mathbb{C}}(N,1)$. In a communication scenario, a typical signal model would be $\mathbf{z} = h\,\mathbf{w} + \mathbf{n}$, where $h \in \mathbb{C}$ is a random complex number. As we concentrate on non-coherent communication, and detecting a single user, h is suppressed here. In a communication scenario, the noise $\mathbf{n}$ would typically be modeled as Additive White Gaussian Noise (AWGN), and $\mathbf{z}$ would not be unit norm. In a quantization scenario, $\mathbf{z}$ is a unit norm complex vector, and quantization noise $\mathbf{n}$ would be uniformly distributed on a Voronoi cell.

The relevant metric in $\mathcal{G}_{\mathbb{C}}(N,1)$ is the *chordal distance* $d_{\mathrm{c}}(\mathbf{u}_1,\mathbf{u}_2) = \sqrt{1-|\mathbf{u}_1^{\mathrm{H}}\,\mathbf{u}_2|^2}$, and the quality of a Grassmannian codebook $\mathcal{C} \subset \mathcal{G}_{\mathbb{C}}(N,1)$ is measured by the *minimum chordal distance*.

The objective of a decoder, both in the communication and quantization scenario, is to find the closest codeword to $\mathbf{z}$:

$$\mathbf{w}_0 = \underset{\mathbf{w}\in\mathcal{C}}{\operatorname{argmin}}\, d_c(\mathbf{z},\mathbf{w})\,. \tag{2}$$

The closest codeword can be found by exhaustive search, which performed blindly, is of order $\mathcal{O}(N\times|\mathcal{C}|)$. For binary chirps, the complexity would be $\mathcal{O}(\sqrt{N}^{5+\log_2 N})$. For real-time operations, a low-complexity decoding algorithm is a must.

III. Binary Chirps

Binary Chirps (BCs) [12] are unit norm vectors in $\mathbb{C}^N$ with many desirable algebraic and geometric features. They are constructed in terms of binary objects in $m = \log_2 N$

dimensions as follows. For an $m \times m$ binary symmetric matrix $\mathbf{S} \in \mathrm{Sym}(m;2)$ and a binary vector $\mathbf{b} \in \mathbb{F}_2^m$, a BC is defined as

$$\mathbf{w}_{\mathbf{S},\mathbf{b}} = \frac{1}{\sqrt{N}} \left[i^{\mathbf{v}\mathbf{S}\mathbf{v}^{\mathsf{T}} + 2\mathbf{v}\mathbf{b}^{\mathsf{T}}} \right]_{\mathbf{v} \in \mathbb{F}_2^m} \tag{3}$$

We see that BCs are made of two parts: a *mask sequence* [7] $\mathbf{m}_{\mathbf{S}} = [i^{\mathbf{v}\mathbf{S}\mathbf{v}^{\mathsf{T}}}]$ corresponding to the quadratic form $\mathcal{Q}_{\mathbf{S}} : \mathbf{v} \to \mathbf{v}\mathbf{S}\mathbf{v}^{\mathsf{T}}$ and a Hadamard sequence $\mathbf{h}_{\mathbf{b}} = [(-1)^{\mathbf{v}\mathbf{b}^{\mathsf{T}}}]$. Overall, the exponents in (3) are evaluations (modulo 4) of degree 2 polynomials in m variables, and therefore the collection of BCs is just an *exponentiated* second order Reed-Muller code.

The algebraic properties of BCs arise from their connection to the Heisenberg-Weyl group $\mathcal{HW}_N$ in N dimensions, generated by the $N \times N$ hermitian Pauli matrices $\mathbf{E}(\mathbf{x},\mathbf{y})$, which are parametrized in terms of two m-dimensional binary vectors $\mathbf{x},\mathbf{y} \in \mathbb{F}_2^m$, see the Appendix. A binary symmetric matrix $\mathbf{S}$ defines a maximal commutative subgroup of $\mathcal{HW}_N$:

$$\mathcal{S}_{\mathbf{S}} = \{\mathbf{E}(\mathbf{x},\mathbf{x}\mathbf{S}) \mid \mathbf{x} \in \mathbb{F}_2^m\}. \tag{4}$$

The subgroup $\mathcal{S}_{\mathbf{S}}$ can be generated by the m Pauli matrices $\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)$ for $r = 1,\ldots,m$, where $\{\mathbf{e}_1,\ldots,\mathbf{e}_m\}$ is the standard basis of $\mathbb{F}_2^m$, and $\mathbf{s}_r$ is row r of $\mathbf{S}$. A BC parametrized by a symmetric $\mathbf{S}$ is a simultaneous eigenvector of $\mathcal{S}_{\mathbf{S}}$, and thus of these m Pauli matrices. This is the main feature that enables a low-complexity decoding algorithm.

The codebook of binary chirps $\mathcal{C}_{\mathrm{chirp}}$ has excellent distance properties. If $\mathbf{w}_1$ and $\mathbf{w}_2$ are BCs with different mask sequences, then

$$d_{\mathrm{c}}(\mathbf{w}_1,\mathbf{w}_2) \geq \sqrt{1-2^{-r}}, \tag{5}$$

where $r = \mathrm{rank}\,(\mathbf{S}_1 - \mathbf{S}_2)$ [12], from which we see that $d_{\mathrm{c}}(\mathcal{C}_{\mathrm{chirp}}) = 1/\sqrt{2}$. On the other hand, since a BC is parametrized by a symmetric matrix and a binary vector, it follows that $|\mathcal{C}_{\mathrm{chirp}}| = 2^{m(m+3)/2}$. In [16], [17] we expand $\mathcal{C}_{\mathrm{chirp}}$ to a codebook of size approximately 2.4 times larger, while keeping the minimum chordal distance intact. On the other hand, the codebook $\mathcal{C}_{\mathrm{chirp}}$ contains a maximal set of mutually unbiased bases with minimum chordal distance $\sqrt{(N-1)/N}$, corresponding to the Kerdock symmetric matrices [18].

IV. Low-Complexity Decoding of Binary Chirps

Exploiting the underlying algebraic structure, one can decode (1) using the algorithm of Howard et al. [12], which is of order $\mathcal{O}\left(N\,(\log_2 N)^2\right)$.

By definition, $|\mathbf{w}^{\mathrm{H}}\mathbf{E}(\mathbf{x},\mathbf{y})\mathbf{w}| \leq 1$ for any Pauli matrix $\mathbf{E}(\mathbf{x},\mathbf{y})$, and it would be maximized for the stabilizer $\mathcal{S}_{\mathbf{S}}$ of $\mathbf{w}$. We define an autocorrelation sequence related to shift $\mathbf{e}_r, r = 1,\ldots,m$ as

$$\begin{aligned} f_r(\mathbf{y}) &= \mathbf{z}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{z} \\ &= \mathbf{w}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{w} + 2\Re\left(\mathbf{w}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{n}\right) + \mathbf{n}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{n}. \end{aligned} \tag{6}$$

These are elements of N-dimensional vectors $\mathbf{f}_r$, indexed by the m-dimensional binary vector $\mathbf{y} \in \mathbb{F}_2^m$. The Pauli matrix $\mathbf{E}(\mathbf{e}_r,\mathbf{y})$ is given in (28). Here, there is a wanted signal component, a cross-term between wanted signal and noise, and a noise component. If there were no noise, the largest absolute value of $\mathbf{f}_r$ would directly single out the row $\mathbf{s}_r$ in $\mathbf{S}$, corresponding to the $\mathbf{E}(\mathbf{e}_r,\mathbf{y})$ which has $\mathbf{w}$ as an eigenvector. Thus, from $|\mathbf{f}_r|$, $r = 1,\ldots,m$, one may find $\mathbf{S}$. Comparing to cross-correlations, the noise contributions in the autocorrelation sequences are larger. Direct computation of cross correlations would, however, have exhaustive search complexity.

Computing the autocorrelation sequences (6) directly would consume $\mathcal{O}(mN^2)$ operations. In [12], complexity is reduced by computing each of the vectors $|\mathbf{f}_r|$ with an N-dimensional Hadamard transform, consuming $\mathcal{O}(mN)$ operations. This will be detailed in Section V below.

If the noise is strong, the algorithm as described may not output a symmetric matrix. To enforce a symmetric output, once rows $1,\ldots,r$ of $\mathbf{S}$ are found, the search for row $r+1$ has to be performed on an $(m-r)$-dimensional subspace commensurate with the previously reconstructed rows.

With $\mathbf{S}$ in hand, the algorithm of [12], continues to estimating $\mathbf{b}$ by *dechirping*. Essentially, the $\mathbf{b}$ generating the Hadamard sequence $\mathbf{h}_{\mathbf{b}} = [(-1)^{\mathbf{v}\mathbf{b}^{\mathsf{T}}}]$ is found by performing a Hadamard transform. The overall decoding complexity thus becomes $\mathcal{O}\big((m+1)mN\big)$. This will also be detailed in Section V.

If the noise is strong, the algorithm rarely finds the closest codeword. In a communication scenario, the signal-to-noise ratio can be controlled by, e.g., using more energy to transmit the codewords. In a quantization scenario, we cannot affect the level of quantization noise—noise is uniformly distributed near and far from the codewords.

Our prime goal is to develop improved low-complexity decoding algorithms for BCs that solve (2) with high probability even in scenario with strong noise.

V. Geometry of Binary Chirps

In [16] we provide a novel direction to understand BC geometry. This leads to an improved understanding of the decoding problem, and reveals possibilities for improving decoders.

We think of the mask sequence $\mathbf{m}_{\mathbf{S}}$ as a diagonal unitary matrix $\mathbf{G}_D(\mathbf{S})$ in the unitary group $\mathbb{U}(N)$. Such diagonal matrix has the desirable property that it fixes the Heisenberg-Weyl group under conjugation [19]:

$$\mathbf{G}_D(\mathbf{S})\mathbf{E}(\mathbf{x},\mathbf{y})\mathbf{G}_D(\mathbf{S})^{\mathrm{H}} = \pm\mathbf{E}(\mathbf{x},\mathbf{y}+\mathbf{x}\mathbf{S}). \tag{7}$$

Unitary matrices that fix the Heisenberg-Weyl group under conjugation, that is, the normalizer of $\mathcal{HW}_N$ in $\mathbb{U}(N)$ form the so-called *Clifford group* [11].

We denote the standard *computational basis* of $\mathbb{C}^2$ as $|0\rangle = [1,0]^{\mathsf{T}}$ and $|1\rangle = [0,1]^{\mathsf{T}}$. The corresponding row-vectors are denoted by $\langle 0|, \langle 1|$. Basis vectors in $\mathbb{C}^N \cong (\mathbb{C}^2)^{\otimes m}$ can then be expressed in terms of vectors $\mathbf{v} = (v_1,\ldots,v_m) \in \mathbb{F}_2^m$ of m bits. Defining $|\mathbf{v}\rangle := |v_1\rangle \otimes \cdots \otimes |v_m\rangle$, the set $\{|\mathbf{v}\rangle \mid \mathbf{v} \in \mathbb{F}_2^m\}$ is the standard basis of column vectors in $\mathbb{C}^N$, i.e., the columns of the N-dimensional identity matrix.

The Hadamard sequence is then a column of the Hadamard matrix, which can be written in terms of a binary sum as

$$\mathbf{H}_N = \frac{1}{\sqrt{N}} \sum_{\mathbf{u},\mathbf{v} \in \mathbb{F}_2^m} (-1)^{\mathbf{u}\mathbf{v}^{\mathsf{T}}} |\mathbf{u}\rangle\langle\mathbf{v}|\,, \tag{8}$$

which also fixes $\mathcal{HW}_N$ by conjugation:

$$\mathbf{H}_N\mathbf{E}(\mathbf{x},\mathbf{y})\mathbf{H}_N^{\mathrm{H}} = i^{-\mathbf{x}\mathbf{y}^{\mathsf{T}}}\mathbf{E}(\mathbf{y},\mathbf{x}), \tag{9}$$

and thus belongs to the Clifford group. Combining all together, a BC $\mathbf{w}_{\mathbf{S},\mathbf{b}}$ is the $\mathbf{b}$th column of the matrix $\mathbf{G}_D(\mathbf{S})\mathbf{H}_N$. Next, the $\mathbf{b}$th column of $\mathbf{H}_N$ is $\mathbf{w}_{\mathbf{0},\mathbf{b}}$, and for all $\mathbf{x} \in \mathbb{F}_2^m$ we have

$$\begin{aligned}\mathbf{E}(\mathbf{x},\mathbf{0})\mathbf{w}_{\mathbf{0},\mathbf{b}} &= \frac{1}{\sqrt{N}} \sum_{\mathbf{u}\in\mathbb{F}_2^m} |\mathbf{u}+\mathbf{x}\rangle\langle\mathbf{u}| \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{\mathbf{v}\mathbf{b}^\mathsf{T}} |\mathbf{v}\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{\mathbf{v}\mathbf{b}^\mathsf{T}} |\mathbf{v}+\mathbf{x}\rangle \;=\; (-1)^{\mathbf{x}\mathbf{b}^\mathsf{T}} \mathbf{w}_{\mathbf{0},\mathbf{b}}. \end{aligned} \tag{10}$$

That is, $\mathbf{w}_{\mathbf{0},\mathbf{b}}$ is an eigenvector of $\mathbf{E}(\mathbf{x},\mathbf{0})$ with eigenvalue $(-1)^{\mathbf{x}\mathbf{b}^\mathsf{T}}$. Combining this with (7), we have that the BC $\mathbf{w}_{\mathbf{S},\mathbf{b}}$ is an eigenvector of $\mathbf{E}(\mathbf{x},\mathbf{x}\mathbf{S})$ for all $\mathbf{x} \in \mathbb{F}_2^m$ with corresponding eigenvalues ± 1. Using (30) and (31) we see that the set (4) is a collection of N commuting matrices closed under multiplication, that is, a *maximal stabilizer*. As such, these matrices can be simultaneously diagonalized, and interestingly, $\mathbf{G}_D(\mathbf{S})\mathbf{H}_N$ does the job. The collection of chirps with the same mask sequence $\{\mathbf{w}_{\mathbf{S},\mathbf{b}} \mid \mathbf{b} \in \mathbb{F}_2^m\}$ is thus the common eigenspace of $\mathcal{S}_\mathbf{S}$. Since $\mathcal{S}_\mathbf{S}$ is a group, it is sufficient to work with a generating set. We use the canonical choice $\{\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r) \mid r = 1,\ldots,m\}$, and refer the reader to [16], [17] for more details.

Using this, we can compute autocorrelations of the codewords $\mathbf{w}$ of (3), needed in decoding. For each $r = 1,\ldots,m$, consider

$$\mathbf{w}^\mathrm{H}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{w} = i^{\mathbf{e}_r\mathbf{y}^\mathsf{T}} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{\mathbf{y}\mathbf{v}^\mathsf{T}} \mathbf{w}(\mathbf{v})\overline{\mathbf{w}(\mathbf{v}+\mathbf{e}_r)} \tag{11}$$

$$= \frac{i^\alpha}{N} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{(\mathbf{e}_r\mathbf{S}+\mathbf{y})\mathbf{v}^\mathsf{T}}. \tag{12}$$

for all $\mathbf{y} \in \mathbb{F}_2^m$. We used (28) for the first equality. The exponent α in (12) can be explicitly computed but for our purposes is irrelevant. The largest value is achieved when $\mathbf{e}_r\mathbf{S}+\mathbf{y} = \mathbf{0}$, that is, $\mathbf{y} = \mathbf{s}_r$—the rth row/column of $\mathbf{S}$.

Recalling (8), it directly follows from the first equality that the vector $\mathbf{f}_r$ with elements $f_r(\mathbf{y}) = \mathbf{z}^\mathrm{H}\mathbf{E}(\mathbf{e}_r,\mathbf{y})\mathbf{z}$ (6) indexed by the binary vector $\mathbf{y}$, can be computed as

$$\mathbf{f}_r = i^{\mathbf{e}_r\mathbf{y}^\mathsf{T}} \mathbf{H}_N \left(\mathbf{z} \odot \; \mathbf{E}(\mathbf{e}_r,\mathbf{0})\,\mathbf{z}^*\right) \;, \tag{13}$$

where $\odot$ is the elementwise product. The vector $\mathbf{f}$ of autocorrelations can thus be computed with one N-dimensional Hadamard transform.

The *dechirping* operation of [12] to find $\mathbf{b}$ while knowing $\mathbf{S}$ works similarly. We compute

$$\mathbf{w}^\mathrm{H}\mathbf{E}(\mathbf{0},\mathbf{y})\mathbf{w}_{\mathbf{S},\mathbf{0}} = \frac{1}{N} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{(\mathbf{y}+\mathbf{b})\mathbf{v}^\mathsf{T}} \tag{14}$$

for all $\mathbf{y} \in \mathbb{F}_2^m$. This has a maximum at $\mathbf{y} = \mathbf{b}$.

We can now compute the eigenvalues of $\mathbf{w}_{\mathbf{S},\mathbf{b}}$ w.r.t the commuting Pauli matrices defined by the rows $\mathbf{s}_r$ of $\mathbf{S}$ as

$$\begin{aligned}\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)\mathbf{w}_{\mathbf{S},\mathbf{b}} &= \frac{i^{s_{r,r}}}{\sqrt{N}} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{\mathbf{v}\mathbf{s}_r^\mathsf{T}} i^{\mathbf{v}\mathbf{S}\mathbf{v}^\mathsf{T}+2\mathbf{b}\mathbf{v}^\mathsf{T}} |\mathbf{v}+\mathbf{e}_r\rangle \\ &= \frac{i^{s_{r,r}}}{\sqrt{N}} \sum_{\mathbf{v}\in\mathbb{F}_2^m} (-1)^{(\mathbf{e}_r+\mathbf{v})\mathbf{s}_r^\mathsf{T}} i^{(\mathbf{e}_r+\mathbf{v})\mathbf{S}(\mathbf{e}_r+\mathbf{v})^\mathsf{T}+2\mathbf{b}(\mathbf{e}_r+\mathbf{v})^\mathsf{T}} |\mathbf{v}\rangle \\ &= \frac{(-1)^{b_r}}{\sqrt{N}} \sum_{\mathbf{v}\in\mathbb{F}_2^m} i^{\mathbf{v}\mathbf{S}\mathbf{v}^\mathsf{T}+2\mathbf{b}\mathbf{v}^\mathsf{T}} |\mathbf{v}\rangle \;=\; (-1)^{b_r}\mathbf{w}_{\mathbf{S},\mathbf{b}} \;, \end{aligned} \tag{15}$$

where the rth diagonal entry of $\mathbf{S}$ is $s_{r,r} = \mathbf{e}_r\mathbf{s}_r^\mathsf{T} = \mathbf{e}_r\mathbf{S}\mathbf{e}_r^\mathsf{T}$ and the rth entry of $\mathbf{b}$ is $b_r = \mathbf{e}_r\mathbf{b}^\mathsf{T}$. Thus $\mathbf{w}_{\mathbf{S},\mathbf{b}}$ is an eigenvector of $\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)$ with eigenvalue $(-1)^{b_r}$ for all $r = 1,\ldots,m$.

VI. Improved Decoding Algorithms

Throughout the process, we will decode a BC $\mathbf{w}$ by finding a maximal set of commuting Paulis $\mathbf{E}$ such that $\mathbf{z}$ is close to a common eigenvector of these; $\mathbf{E}\,\mathbf{w} = (-1)^\varepsilon\mathbf{w}$. As discussed above, in a noiseless situation we need to find only a generating set $\{\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)\}_{r=1}^m$ for which

$$\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)\mathbf{w} = (-1)^{b_r}\mathbf{w}\,, \tag{16}$$

see (15). The rows of $\mathbf{S}$ are then given by $\mathbf{s}_r$, while the elements in $\mathbf{b}$ are b_r. The m signs $(-1)^{b_r}$ completely determine the column vector $\mathbf{b}$, and thus dechirping is automatically taken care of.

As in [12], we estimate parametrizing symmetric matrix $\mathbf{S}$ one row at a time. The generic problem at hand is to determine row r of $\mathbf{S}$ and element $b_r = \varepsilon$, conditioned on a set $\mathcal{R}$ of previously determined rows. Let $\widehat{\mathbf{S}}$ be the partial symmetric matrix determined by the rows in $\mathcal{R}$. The search space for $\mathbf{E}(\mathbf{e}_r,\mathbf{y})$ is then restricted to

$$\mathcal{Y}_r = \{\mathbf{y} \in \mathbb{F}_2^m \mid y_i = \widehat{s}_{r,i} \text{ for all } i \in \mathcal{R}\}\,, \tag{17}$$

commensurate with $\widehat{\mathbf{S}}$. This will guarantee $\mathbf{S}$ to be symmetric and decoding within the codebook. We are thus searching for a Pauli $\mathbf{E} = \mathbf{E}(\mathbf{e}_r,\mathbf{y})$ with $\mathbf{y} \in \mathcal{Y}_r$ to be commensurate with previously determined Paulis, and a sign $\sigma = (-1)^\varepsilon$, such that $\mathbf{E}\,\mathbf{w} = \sigma\,\mathbf{w}$.

A. Decoding Metric & Half-space Projections

Pauli matrices are in one-to-one correspondence to a set of half-space projection operators, see (29). Thus, from an estimated $\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)$, a projection operator

$$\mathbf{\Pi}_{r,\varepsilon} = \frac{1}{2}\left(\mathbf{I}_N + (-1)^\varepsilon\mathbf{E}(\mathbf{e}_r,\mathbf{s}_r)\right), \tag{18}$$

can be constructed, for which we have

$$\mathbf{\Pi}_{r,\varepsilon}\mathbf{w}_{\mathbf{S},\mathbf{b}} = \begin{cases}\mathbf{w}_{\mathbf{S},\mathbf{b}}, & \text{if } \varepsilon = b_r,\\ \mathbf{0}, & \text{if } \varepsilon \neq b_r.\end{cases} \tag{19}$$

This operator projects to an $N/2$-dimensional subspace.

For decoding, we consider autocorrelation sequences (6), retaining the sign information. These can be reinterpreted in terms of the projectors (19). For any Pauli matrix $\mathbf{E}(\mathbf{e}_r,\mathbf{y})$ and binary ε, we can construct a projector of the form (19). The length of $\mathbf{z}$ projected with this operator is

$$\|\mathbf{\Pi}_{r,\varepsilon}\mathbf{z}\|^2 = \mathbf{z}^\mathrm{H}\mathbf{\Pi}_{r,\varepsilon}\mathbf{z} = \frac{1}{2}\Big(\mathbf{z}^\mathrm{H}\mathbf{z} + (-1)^\varepsilon f_r(\mathbf{y})\Big). \tag{20}$$

We thus see that finding the largest absolute value of the autocorrelation (6), together with its sign, directly finds the half-space to which the projection of $\mathbf{z}$ is largest.

Algorithm 1 Decoding row r in $\mathbf{S}$.

Input: Vector $\mathbf{z}$ and metric $\mu = \|\mathbf{z}\|^2$, partial estimate $\widehat{\mathbf{S}}$, Vector of sign-bits $\mathbf{b}$, rows estimated $\mathcal{R}$, Row to estimate r.

1. Compute $f_r(\mathbf{y}) = \mathbf{z}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r, \mathbf{y})\,\mathbf{z}$ for $\mathbf{y} \in \mathbb{F}_2^m$ using (6).
2. Find search set $\mathcal{Y}_r \subset \mathbb{F}_2^m$ from $\mathcal{R}$, $\widehat{\mathbf{S}}$, and r using (17).
3. Find ordered set of $\mathbf{y}_k \in \mathcal{Y}_r$ with K largest $|f_r(\mathbf{y})|$.
4. **for** $k = 1, \ldots, K$
5. Estimate $\widehat{\mathbf{S}}_k$ is $\widehat{\mathbf{S}}$ with row r given by $\mathbf{y}_k$.
6. Metric is $\mu_k = \frac{1}{2}\left(\mu + |f_r(\mathbf{y}_k)|\right)$.
7. Sign is $\sigma_k = \operatorname{sign} f(\mathbf{y}_k)$.
8. Sign-bit vector $\mathbf{b}_k$ is $\mathbf{b}$ with $b_r = (1 + \sigma_k)/2$.
9. Projector is $\mathbf{\Pi}_k = \frac{1}{2}\left(\mathbf{I} + \sigma_k \mathbf{E}(\mathbf{e}_r, \mathbf{y}_k)\right)$.
10. Projected vector is $\mathbf{z}_k = \mathbf{\Pi}_k\,\mathbf{z}$.
11. **end**

Output: List of K partial estimates $\widehat{\mathbf{S}}_k$, projected vectors $\mathbf{z}_k$, sign vectors $\mathbf{b}_k$ and metrics μ_k.

B. Decoder with Consecutive Projectors

We proceed in a sequential manner, finding $\mathbf{S}$ and $\mathbf{b}$ row-by-row, defining a projector $\mathbf{\Pi}_r$ for each row. If we have found the correct $\mathbf{\Pi}_r$, projecting $\mathbf{z}$ with it, results in

$$\mathbf{\Pi}_r\mathbf{z} = \mathbf{w} + \mathbf{\Pi}_r\mathbf{n}\,, \tag{21}$$

as $\mathbf{\Pi}_r$ does not affect $\mathbf{w}$. Such a projection, however, directly reduces the expected noise energy. Assume that the noise is circularly symmetric and i.i.d across dimensions. This means that the noise probability distribution is unitarily invariant: $p(\mathbf{n}) = p(\mathbf{U}\mathbf{n})$ for any unitary $\mathbf{U}$. This assumption holds for circularly symmetric AWGN, and approximately for the problem of quantizing vectors to the BC codebook, when the quantization source is uniformly distributed. Then, the expected noise energy is halved for any half-space projection:

$$\begin{aligned}\mathcal{E}\left\{\|\mathbf{\Pi}_r\mathbf{n}\|^2\right\} &= \int p(\mathbf{n})\ \mathbf{n}^{\mathrm{H}}\mathbf{\Pi}_r\,\mathbf{n}\ d\mathbf{n}d\mathbf{n}^* \\ &= \int p(\mathbf{n})\ \mathbf{n}^{\mathrm{H}}\mathbf{U}^{\mathrm{H}}\mathbf{I}_{N/2,N}\mathbf{U}\,\mathbf{n}\ d\mathbf{n}d\mathbf{n}^* = \tfrac{1}{2}\mathcal{E}\left\{\|\mathbf{n}\|^2\right\}\end{aligned}$$

where $\mathbf{I}_{N/2,N}$ is the identity with the latter half-diagonal 0, representing the eigenvalues of the projector $\mathbf{\Pi}_r$, and $\mathbf{U}$ are the eigenvectors of the projector. For this reason, once a candidate row and sign is found, we project the input with the corresponding projector, *before proceeding with decoding the next row*. This directly reduces the noise components in (6) when decoding the consequent rows.

The resulting algorithm is summarized as Algorithm 1. To be prepared for list decoding, discussed below, the algorithm is formulated as keeping track of K strongest estimates $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ for row r in the search space $\mathcal{Y}_r$. The dominant complexity of this algorithm is in the computation of $f_r(\mathbf{y})$, which is done with a Hadamard transform, as in (13). The projections on Step 10 can be computed in $\mathcal{O}(N)$. Note that both terms in (20) are already pre-computed, thus this step does not contribute any added complexity. The overall complexity of decoding one row in $\mathbf{S}$ with the algorithm is thus $\mathcal{O}\big((m + K)N\big)$, and with $K = 1$, the complexity is $\mathcal{O}\big(m(m+1)N\big)$, which coincides with the complexity of [12].

Algorithm 2 Selecting decoding order.

Input: Vector $\mathbf{z}$.

1. **for** $r = 1, \ldots, m$
2. Compute $f_r(\mathbf{y}) = \mathbf{z}^{\mathrm{H}}\mathbf{E}(\mathbf{e}_r, \mathbf{y})\,\mathbf{z}$ for $\mathbf{y} \in \mathbb{F}_2^m$ using (6).
3. $\mu_r = \max |f(\mathbf{y})|$.
4. **end**
5. Sort $\{\mu_r\}_{r=1}^m$ in descending order. The resulting ordering is $\mathcal{O}$.

Output: Decoding order $\mathcal{O}$.

Algorithm 1 defines a function

$$\mathcal{N} \equiv \left\{\left(\widehat{\mathbf{S}}_k, \mathbf{z}_k, \mathbf{b}_k, \mu_k\right)\right\}_{k=1}^{K} = \mathtt{DecodeRow}_K\left(\widehat{\mathbf{S}}, \mathbf{z}, \mathbf{b}, \mu, \mathcal{R}, r\right)\,, \tag{22}$$

which gives a set with K elements consisting of a partial estimate, a projected input, and the corresponding metric. The Howard et al. algorithm, in essence, calls the function (22), for rows $r = 1, \ldots, m$ with $K = 1$, but does not execute Step 10 of Algorithm 1. Then it proceeds with dechirping, which in our case is not necessary.

C. Decoding Order Selection

The Howard et al. algorithm finds the rows of the parametrizing symmetric matrix in a canonical order, starting from first to last. The first row is most prone to errors, since the search space is the largest. With Algorithm 1, the first decoded row is equally prone to errors. When proceeding to other rows, Algorithm 1 reduces noise power, and eases decoding. Recall that the projector is applied after a row (and sign) is found, which directly reduces the noise affecting the next rows.

In reduced complexity sequential search algorithms, the search order is often adaptively chosen, a prominent example being Ordered Successive Interference Cancellation (OSIC) [20]. In the problem at hand, the wanted signal is corrupted by random noise. It thus makes sense to choose the decoding order to be one that has the smallest expected noise contribution. For this, inspired by OSIC, we choose decoding order as in Algorithm 2.

D. Breadth-First List Decoding

Finally, to increase decoding probability in a sequential search, keeping multiple candidate states alive during the sequential search process improves performance. For this, we apply Breadth-First Search List Decoding (BFS-LD) [15]. BFS has been widely used in multi-antenna communication receivers [21], [22]. One could also perform a Depth-First Search as suggested in [2], but here we use breadth-first due to its deterministic complexity.

The breadth-first search method applied is summarized in Algorithm 3. It keeps K alternatives alive in the search tree. At each stage $i = 2, \ldots m$, when a new row is decoded, KK' candidate branch states are produced by calling Algorithm 1 K times. The K branches with the highest decoding metric are chosen as starting points for the next round. As the search space is reduced according to (17), when more rows have been

Algorithm 3 Breadth-First list decoding of $\mathbf{S}$.

Input: Vector $\mathbf{z}$, decoding order $\mathcal{O}$.
Initialize: Set of previously decoded rows $\mathcal{R} = \varnothing$,
candidate list $\mathcal{C} = \{(\mathbf{0}_m, \mathbf{z}, 0, \|\mathbf{z}\|^2)\}$.
1. **for** $i = 1, \ldots, m$
2. The row to decode is $r = \mathcal{O}(i)$
3. Initialize empty new candidate list $\mathcal{N} = \varnothing$.
4. Number of branches $K' = \min(K, 2^{m-i+1})$.
5. **for** each $(\widehat{\mathbf{S}}_c, \mathbf{z}_c, \mathbf{b}_c, \mu_c) \in \mathcal{C}$
6. Compute new candidates using (22):
$\mathcal{N}_c = \texttt{DecodeRow}_{K'}(\widehat{\mathbf{S}}_c, \mathbf{z}_c, \mathbf{b}_c, \mu_c, \mathcal{R}, r)$.
7. $\mathcal{N} = \mathcal{N} \cup \mathcal{N}_c$.
8. **end**
9. **if** $i < m$
10. $\mathcal{C}$ is K-element subset of $\mathcal{N}$ with largest μ_j.
11. Update decoded rows $\mathcal{R} = \mathcal{R} \cup \{r\}$.
12. **else**
13. Find element j in $\mathcal{N}$ with largest μ_j.
14. **end**
15. **end**
Output: Estimates $\widehat{\mathbf{S}} = \widehat{\mathbf{S}}_j$ and $\mathbf{b} = \mathbf{b}_j$, metric $\mu = \mu_j$.

estimated, it may be that K is larger than the search space. In Step 4, the list size is accordingly chosen as K'.

Note that when calling Algorithm 1 in Step 6 of Algorithm 3, the output decoding metric is directly the length of

$$\tilde{\mathbf{z}} = \left(\prod_{r \in \mathcal{R}} \mathbf{\Pi}_{\mathbf{s}_r, b_r} \right) \mathbf{z} \tag{23}$$

with $\mathcal{R}$ the updated set from Step 11. That is, the metric is the norm of the projection of $\mathbf{z}$ to the subspace determined by the candidate rows $\widehat{\mathbf{s}}_r$ and the sign-bits b_r for $r \in \mathcal{R}$.

To compute complexity, we define the parameters

$$k = \begin{cases} \lceil \log_2 K \rceil, & \text{if } K \leq N, \\ m, & \text{else }, \end{cases}$$

$\widetilde{K} = 2^k - 2$ and $\bar{K} = \min(K, N)$. The overall decoding complexity of BFS-LD with projections is then

$$\mathcal{O}\Big(\big(m + \bar{K} + (m^2 - m)K + (m - k)K^2 + K\widetilde{K}\big) N \Big).$$

The BFS-LD algorithm can be used with or without projections in Algorithm 1, as well as with or without order selection as in Algorithm 2. This gives us four candidate algorithms for each K. However, if the algorithm is used without projections, the metric applied is only a heuristic. For BFS-LD to improve the performance of an algorithm without projections, the halving in the metric computation in Step 6 of Algorithm 1 has to be removed. Then, the contribution of each row was the same weight in the sum. Also, if projections are not used, the output of Algorithm 3 should not be selected based on the heuristic metric. Once all rows are decoded, the corresponding BSs can be constructed. Instead of selecting the largest metric, the vector having the largest inner product with $\mathbf{z}$ is chosen. This increases the complexity of BFS-LD without projections with a factor of $\mathcal{O}(2KN)$ per row of $\mathbf{S}$.

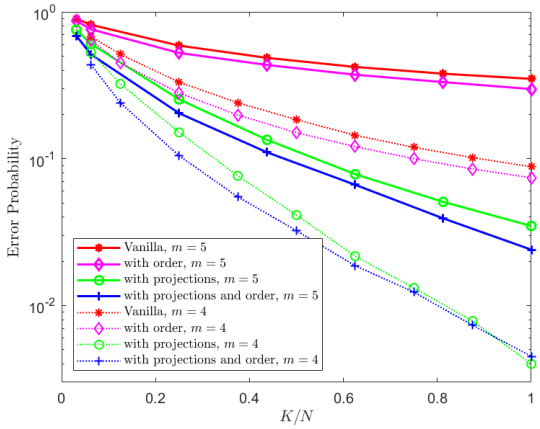

Fig. 1. Decoding error of the vanilla algorithm [12] of Howard et al. compared to the improved algorithms of this paper, in a quantization setting in $N = 16$ and 32 dimensions.

Note that Algorithm 1 performs exhaustive search over the new alternatives emerging on row r, given the selections done for earlier rows. Thus vanilla decoding with $K = 1$ performs exhaustive search over the first row. If $K = N$, all alternatives from the first row are retained when going for the second, leading to exhaustive search over the two first rows.

VII. Performance Analysis

We concentrate on the vector quantization problem, as it has a most unfriendly noise distribution. Geometric analysis is intractable, due to the complex structure of the high-dimensional Voronoi cells. Accordingly, we analyze the performance of the considered algorithms with Monte Carlo integration.

The decoding error probability is of generic interest. It is indicative of algorithm performance in communication scenarios with heavy noise, in addition to the simulated quantization scenario. We simulate performance for $m = 4, N = 16$ and $m = 5, N = 32$. These are of practical interest for vector quantization of Frequency Division Duplexing massive MIMO communication, where first the antenna dimensionality is reduced by identifying an eigenspace, followed by vector quantization in a lower-dimensional subspace [8].

The error probabilities of the different decoders are plotted in Figure 1. We have four basic decoders: the vanilla algorithm of [12], the consecutive projection algorithm, the order selection algorithm, and an algorithm with both order selection and consecutive projections. All four are combined with Breadth-First Search with $K = 1, \ldots, N$, and plotted against K/N. Successive projections and order selection provide considerable gain for small K, and these gains are almost additive. For $m = 4$, ordering becomes counterproductive when approaching $K = N$. This is understandable as $K = N$ performs exhaustive search over the two first rows, which given the ordering principle are the easiest to decode. For larger m more rows remain to be decoded in an non-exhaustive manner for $K = N$, and ordering still provides gain.

VIII. Conclusions

We have considered low-complexity autocorrelation decoders for identifying Binary Chirp codewords from noisy signals,

based on sequential search over the m binary dimensions describing the N-dimensional codeword. We have considered a geometrically inspired projection decoder that reduces noise for subsequent decoder stages, an ordering function which selects the order in which binary dimensions are considered, and breadth-first list decoding. These improvements make it possible to approach low error probability with manageable complexity. Comparing to [12], the performance improvements are achieved without added complexity for a given breadth K of the search. Moreover, the decoder with consecutive projections is directly amenable to finding higher dimensional subspaces.

Note that the use of the constructed BC decoding algorithms based on dimensionality reduction from N to $\log_2 N$ dimensions is not limited to the BC codebook. These algorithms can be used as modules when decoding a generic Grassmannian codebook in $N = 2^m$ dimensions. Assume a codebook $\mathcal{C}'$ with larger cardinality than the BC codebook. A low-complexity decoding algorithm can be constructed by dividing $\mathcal{C}'$ into cells centered at BC codewords $\mathbf{w}$. One first decodes in the BC codebook, using the low-complexity decoders. Next, one can decode within the identified cell of $\mathcal{C}'$.

In future work we shall explore the geometry of the Voronoi cells in the quantization problem, and closed form performance analysis. We shall also expand this work to quantization of points in Euclidean space to Barnes-Wall lattice points.

ACKNOWLEDGMENT

This work was funded by the Academy of Finland (grants 319484, 334539).

APPENDIX

A. The Heisenberg-Weyl Group

Fix $m \in \mathbb{N}$ and put $N = 2^m$. The *Pauli matrices* are

$$\mathbf{I}_2, \quad \sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad \sigma_y = i\sigma_x\sigma_z. \tag{24}$$

For $\mathbf{a}, \mathbf{b} \in \mathbb{F}_2^m$ put $\mathbf{D}(\mathbf{x}, \mathbf{y}) := \sigma_x^{x_1}\sigma_z^{y_1} \otimes \cdots \otimes \sigma_x^{x_m}\sigma_z^{y_m}$. Directly by definition we have

$$\mathbf{D}(\mathbf{x}, \mathbf{0})|\mathbf{v}\rangle = |\mathbf{v} + \mathbf{a}\rangle \text{ and } \mathbf{D}(\mathbf{0}, \mathbf{y})|\mathbf{v}\rangle = (-1)^{\mathbf{y}\mathbf{v}^\mathsf{T}}|\mathbf{v}\rangle, \tag{25}$$

and thus, the former is a permutation matrix whereas the latter is a diagonal matrix. Then

$$\mathbf{D}(\mathbf{a}, \mathbf{b})\mathbf{D}(\mathbf{x}, \mathbf{y}) = (-1)^{\mathbf{b}\mathbf{x}^\mathsf{T}}\mathbf{D}(\mathbf{a} + \mathbf{x}, \mathbf{b} + \mathbf{y}). \tag{26}$$

Thanks to (26), the set

$$\mathcal{HW}_N := \{i^k\mathbf{D}(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}, \mathbf{y} \in \mathbb{F}_2^m, k = 0, 1, 2, 3\} \tag{27}$$

is a subgroup of $\mathbb{U}(N)$, the *Heisenberg-Weyl* group. Its elements are Pauli matrices. The Hermitian Pauli matrices are of form $\mathbf{E}(\mathbf{x}, \mathbf{y}) := i^{\mathbf{x}\mathbf{y}^\mathsf{T}}\mathbf{D}(\mathbf{x}, \mathbf{y})$, where the exponent is taken modulo 4. Combining this with (25) and (26) we obtain

$$\mathbf{E}(\mathbf{x}, \mathbf{y}) = i^{\mathbf{x}\mathbf{y}^\mathsf{T}} \sum_{\mathbf{v} \in \mathbb{F}_2^m} (-1)^{\mathbf{y}\mathbf{v}^\mathsf{T}}|\mathbf{v} + \mathbf{x}\rangle\langle\mathbf{v}|. \tag{28}$$

Being Hermitian, $\mathbf{E}(\mathbf{x}, \mathbf{y})$ has eigenvalues ± 1. Moreover, for $\varepsilon \in \{0, 1\}$, we have the projectors

$$\mathbf{\Pi}_\varepsilon := \frac{1}{2}\left(\mathbf{I}_N + (-1)^\varepsilon\mathbf{E}(\mathbf{x}, \mathbf{y})\right) \tag{29}$$

onto the $(-1)^\varepsilon$-eigenspaces of $\mathbf{E}(\mathbf{x}, \mathbf{y})$. From (26) we have

$$\mathbf{D}(\mathbf{a}, \mathbf{b})\mathbf{D}(\mathbf{x}, \mathbf{y}) = (-1)^{\mathbf{b}\mathbf{x}^\mathsf{T} + \mathbf{a}\mathbf{y}^\mathsf{T}}\mathbf{D}(\mathbf{x}, \mathbf{y})\mathbf{D}(\mathbf{a}, \mathbf{b}). \tag{30}$$

In turn, $\mathbf{D}(\mathbf{a}, \mathbf{b})$ and $\mathbf{D}(\mathbf{c}, \mathbf{d})$ commute iff

$$\langle\, \mathbf{a}, \mathbf{b} \,|\, \mathbf{x}, \mathbf{y} \,\rangle_\mathrm{s} := \mathbf{b}\mathbf{x}^\mathsf{T} + \mathbf{a}\mathbf{y}^\mathsf{T} = 0. \tag{31}$$

The above defines a *symplectic inner product* on $\mathbb{F}_2^{2m}$, which as we see, captures the commutativity of the Pauli matrices [11].

REFERENCES

[1] L. Zheng and D. Tse, "Information theoretic limits for non-coherent multi-antenna communications," in *Proc. IEEE WCNC*, vol. 1, 2000, pp. 18–22.

[2] R. Calderbank and A. Thompson, "CHIRRUP: a practical algorithm for unsourced multiple access," *arXiv preprint arXiv:1811.00879*, 2018. [Online]. Available: https://arxiv.org/pdf/1811.00879.pdf

[3] M. Soleymani and H. Mahdavifar, "Analog subspace coding: A new approach to coding for non-coherent wireless networks," in *Proc. IEEE ISIT*, 2020, pp. 31–36.

[4] Z. Chen, F. Sohrabi, and W. Yu, "Sparse activity detection for massive connectivity," *IEEE Trans. Sign. Proc.*, vol. 66, no. 7, pp. 1890–1904, 2018.

[5] D. Love., R. Heath, Jr., and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Trans. Inf. Th.*, vol. 49, no. 10, pp. 2735–2747, Oct. 2003.

[6] A. Ashikhmin and R. Gopalan, "Grassmannian packings for efficient quantization in MIMO broadcast systems," in *Proc. IEEE ISIT*, 2007, pp. 1811–1815.

[7] R. A. Pitaval and Y. Qin, "Grassmannian frames in composite dimensions by exponentiating quadratic forms," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 13–18.

[8] R. Vehkalahti, J. Liao, T. Pllaha, W. Han, and O. Tirkkonen, "CSI quantization for FDD massive MIMO communication," in *IEEE Vehicular Technology Conference*, VTC-Spring, 2021.

[9] J. Zhang, G. Zhu, R. W. Heath Jr., and K. Huang, "Grassmannian learning: Embedding geometry awareness in shallow and deep learning," preprint arxiv: 1808.02229v2, Aug. 2018.

[10] E. Merzbacher, *Quantum Mechanics, 3rd Edition*. Wiley, 1998.

[11] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction and orthogonal geometry," *Phys. Rev. Lett.*, vol. 78, no. 3, pp. 405–408, 1997.

[12] S. D. Howard, A. R. Calderbank, and S. J. Searle, "A fast reconstruction algorithm for deterministic compressive sensing using second order Reed-Muller codes," in *Conference on Information Sciences and Systems*, March 2008, pp. 11–15.

[13] T. Inoue and R. W. Heath, "Kerdock codes for limited feedback precoded mimo systems," *IEEE Trans. Sign. Proc.*, vol. 57, no. 9, pp. 3711–3716, 2009.

[14] P. Yang, D. Guo, and H. Yang, "Massive access in multi-cell wireless networks using reed-muller codes," preprint arxiv: 2003.11568, Mar. 2020.

[15] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms (2nd ed.)*. McGraw-Hill, 2001.

[16] T. Pllaha, O. Tirkkonen, and R. Calderbank, "Reconstruction of multi-user binary subspace chirps," in *2020 IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 531–536.

[17] ——, "Binary subspace chirps," submitted to Transactions on Information Theory. arXiv:2102.12384, February 2021.

[18] R. Calderbank, S. Howard, and S. Jafarpour, "Construction of a large class of matrices satisfying a statistical isometry property," in *IEEE Journal of Selected Topics in Signal Processing, Special Issue on Compressive Sensing*, vol. 4, no. 2, 2010, pp. 358–374.

[19] N. Rengaswamy, R. Calderbank, S. Kadhe, and H. D. Pfister, "Logical clifford synthesis for stabilizer codes," *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–17, 2020.

[20] X. Li, H. Huang, A. Lozano, and G. Foschini, "Reduced-complexity detection algorithms for systems using multi-element arrays," in *Proc. IEEE GLOBECOM*, vol. 2, 2000, pp. 1072–1076 vol.2.

[21] K. J. Kim and R. Iltis, "Joint detection and channel estimation algorithms for QS-CDMA signals over time-varying channels," *IEEE Trans. Comm.*, vol. 50, no. 5, pp. 845–855, 2002.

[22] S.-J. Choi, S.-J. Shim, Y.-H. You, J. Cha, and H.-K. Song, "Novel mimo detection with improved complexity for near-ML detection in MIMO-OFDM systems," *IEEE Access*, vol. 7, pp. 60 389–60 398, 2019.