

Received March 18, 2022, accepted April 14, 2022, date of publication April 25, 2022, date of current version May 4, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3170491

Low-Complexity Decoder for Overloaded Uniquely Decodable Synchronous CDMA

MICHEL KULHANDJIAN¹, (Senior Member, IEEE),

HOVANNES KULHANDJIAN², (Senior Member, IEEE),

CLAUDE D'AMOURS¹, (Member, IEEE), HALIM YANIKOMEROGLU³, (Fellow, IEEE),

DIMITRIS A. PADOS^{4,5}, (Senior Member, IEEE),

AND GURGEN KHACHATRIAN⁶, (Member, IEEE)

¹School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada

²Department of Electrical and Computer Engineering, California State University, Fresno, Fresno, CA 93740, USA

³Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada

⁴Department of Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431, USA

⁵I-SENSE Center, Florida Atlantic University, Boca Raton, FL 33431, USA

⁶College of Science and Engineering, American University of Armenia, Yerevan 0019, Armenia

Corresponding author: Michel Kulhandjian (mkk6@buffalo.edu)

ABSTRACT We consider the problem of designing a low-complexity decoder for antipodal uniquely decodable (UD) /errorless code sets for overloaded synchronous code-division multiple access (CDMA) systems, where the number of signals K_{max}^a is the largest known for the given code length L . In our complexity analysis, we illustrate that compared to maximum-likelihood (ML) decoder, which has an exponential computational complexity for even moderate code lengths, the proposed decoder has a quasi-quadratic computational complexity. Simulation results in terms of bit-error-rate (BER) demonstrate that the performance of the proposed decoder has only a 1 – 2 dB degradation in signal-to-noise ratio (SNR) at a BER of 10^{-3} when compared to ML. Moreover, we derive the proof of the minimum Manhattan distance of such UD codes and we provide the proofs for the propositions; these proofs constitute the foundation of the formal proof for the maximum number users K_{max}^a for $L = 8$.

INDEX TERMS Uniquely decodable (UD) codes, overloaded CDMA, overloaded binary and ternary spreading spreading codes.

NOMENCLATURE

5G	The fifth generation
6G	The sixth generation
AWGN	Additive white Gaussian noise
BER	Bit-error-rate
BPSK	Binary phase-shift keying
CDM	Code-division multiplexing
CDMA	Code-division multiple access
eMBB	Enhanced mobile broadband
FDA	Fast detection algorithm
FDMA	Frequency division multiple access
LDS	Low-density spreading
MAI	Multiple-access interference
MF	Matched filter
MIMO	Multiple-input multiple-output

ML	Maximum-likelihood
MMSE	Minimum mean-square estimation
mMTC	Massive machine-type-communications
mmWave	Millimeter-wave
MO	Multiple-orthogonal
NDA	Noiseless detection algorithm
NOMA	Non-orthogonal multiple access
OMA	Orthogonal multiple access
PDA	Probabilistic data association
PIC	Parallel interference cancellation
PN	Pseudo-noise
SCMA	Sparse code multiple access
SIC	Successive interference cancellation
SNR	Signal-to-noise ratio
TDMA	Time-division multiple access
UD	Uniquely decodable
uRLLC	Ultra-reliable low-latency communications
ZF	Zero-forcing

The associate editor coordinating the review of this manuscript and approving it for publication was Meng-Lin Ku¹.

I. INTRODUCTION

In the last decade, wireless communication services have experienced explosive growth while communication technologies have progressed generation by generation. In the previous generations spanning from 1G to 4G, the multiple access schemes were mostly characterized by orthogonal multiple access (OMA) techniques, where users are assigned orthogonal resources in either frequency, (frequency-division multiple access (FDMA)), time, (time-division multiple access (TDMA)) or code, (code-division multiple access (CDMA)). CDMA [1] was the basic technology for 3G and for some 2G (IS-95) networks. High spectral- and power-efficiency, massive connectivity and low latency are among the requirements for next generation communications and these requirements are expected to increase in the future, as researchers turn their efforts towards sixth generation (6G) wireless communications. Enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (uRLLC) and massive machine-type communication (mMTC) support a suite of compelling applications driving these requirements. Massive multiple-input multiple-output (MIMO), non-orthogonal multiple access (NOMA) and millimeter-wave (mmWave) communications constitute promising techniques of addressing these stringent requirements [2].

Supporting a large number of users communicating over a common channel may not be readily achievable by OMA techniques due to the multiple-access interference (MAI) in rank-deficient systems, where the number of users is greater than that of the resource blocks. To meet the demand of increased bandwidth efficiency in synchronous CDMA, a CDMA concept was introduced in [3], which can support many more users for a given code length compared to traditional CDMA. A number of signature design schemes have been studied for dense spreading in conventional CDMA, where low cross-correlation sequence sets are designed to minimize the overall MAI, which allows more users to simultaneously access the common channel. This in turn results in increased spectral efficiency. Finding suitable spreading codes and decoding schemes for such overloaded systems is a challenging optimization problem. To address these challenges, numerous non-uniquely decodable (non-UD) [3]–[10], and UD [11]–[35] construction based code sets have been proposed. Examples of such non-UD code sets are pseudo-noise spreading (PN) [4], [5], orthogonal/orthogonal CDMA (O/O), [6], [7], PN/orthogonal CDMA (PN/O) [3], multiple-orthogonal CDMA (MO) [8], improved O/O CDMA [9]. Those codes employ two or more sets of orthogonal signal waveforms, which allows the system to accommodate more users than the signature length L . As a consequence of this, a significant level of MAI exists at the output of each user's matched filter due to the non-zero cross-correlation of different signatures.

Low cross-correlation sequence sets might not be the best criterion for very high rank-deficient systems. One important criterion in such rank-deficient systems is for the code set

to be UD. By definition the UD codes are those in which the data of different users can be unambiguously decoded in a noiseless channel using linear recursive decoders [31]. Low-complexity linear decoders were introduced for these UD code sets using either binary $\{0, 1\}$, or antipodal $\{\pm 1\}$, or alternatively ternary $\{0, \pm 1\}$ chips in [27]–[30], [32]–[35]. On the other hand, Lu *et al.* [36] proposed M -ary code sets for the multiple-access adder channel. Various applications have been implementing UD code sets framework such as in multi-way physical-layer network coding conceived in [33], [37].

All of these multiple access concepts were introduced in order to serve a number of excess users beyond the available resources. These multiple access schemes are characterized by NOMA techniques [38] in 5G and beyond wireless communications. Recently, several NOMA solutions have been actively investigated [2], which can be basically divided into two main categories, namely power-domain and code-domain NOMA. A few of the strong contenders of code-domain NOMA are low-density spreading aided CDMA (LDS-CDMA) [39], [40] and sparse code multiple access (SCMA) [41], [42]. LDS-CDMA [39], [40] must generally guarantee to be UD code set [43], which means non-zero Euclidean distance. The design of LDS type matrices offers flexible resource allocation, performs better in terms of handling the MAI that exists in rank-deficient systems and has low-complexity receivers compared to conventional CDMA.

Unlike the dense UD code set, the LDS-CDMA structure can be represented by a factor graph, the classic message passing algorithm (MPA) can be employed for its detection. Due to the reason that UD code sets were originally designed for adder channels, generally a simple noiseless detection is developed. However in practice, the wireless transmission channel exhibits, among other things, selective fading, multipath and the near-far problem, which leads to unequal received power among users. Consequently, if synchronization, channel equalization are compensated, low-complexity detectors can be applied to wireless channels.

Inspired by these attractive features of UD code sets, this paper investigates UD codes for synchronous uplink NOMA. In the case of dispersive fading channels, we can potentially equalize the channel effect by using channel precoding. Therefore, we consider developing a low-complexity detector for UD code sets that are proposed in [31] for overloaded synchronous CDMA systems. It is widely recognized that the complexity of an optimal detector is exponentially proportional to the number of users, which prohibits its practical implementation. Various suboptimal low-complexity detection techniques have been already proposed. These suboptimal approaches can be classified into two categories: linear and non-linear multiuser detectors. Linear multiuser detectors include among others, matched filter (MF), minimum mean-square error (MMSE), and zero-forcing (ZF), etc. In a non-linear subtractive interference cancellation detector the interference is first estimated and then it is subtracted from

the received signal before detection. The cancellation process can be carried out either successively (SIC) [44], or in parallel (PIC) [45]–[47]. In non-linear iterative detectors [48]–[52], probabilistic data association (PDA) [53] aims to suppress the MAI in each iteration in order to improve the overall error performance. Suboptimal polynomial time detectors that are based on the geometric approach are studied in [54], [55].

In general, linear as well as non-linear detectors cannot separate users in overloaded systems even in the case of asymptotically vanishing noise. Therefore, the spreading codes must have property such that decoding can achieve asymptotically zero probability of error multiuser detection when the signal-to-noise (SNR) ratio becomes arbitrary large. The UD class of codes that guarantee “errorless” communication in an ideal (noiseless) synchronous CDMA/code-division multiplexing (CDM) also shows a good performance in the presence of noise.

Finding the overloaded UD class of codes for noiseless channel is directly related to coin-weighing problem, one of the problems that is discussed by Erdős and Rényi [11]. It can be considered as a special case of a general problem where authors in [12], [13], [16], [25], [26] refer to them as detecting matrices. Lindström [15] defines the same problem as the detecting set of vectors. Given an integer $q \geq 2$ and a finite set alphabet $\mathcal{M}$ of rational integers, let $\mathbf{v}_k$ for $1 \leq k \leq K$ be L -dimensional (column) vectors with all components from $\mathcal{M}$ such that the q^K sums

$$\sum_{k=1}^K \mathbf{v}_k \epsilon_k \quad (\epsilon_k = 0, 1, 2, \dots, q-1) \quad (1)$$

are all distinctly unique, then $\{\mathbf{v}_1, \dots, \mathbf{v}_K\}$ are detecting set of vectors. Let $F_q(L)$ be the maximal number of L -dimensional vectors and $f_q(K)$ be the minimal vector length to form a detecting matrix for a given length L and a number of vectors K . The problem of determining $f_q(K)$ as a special case when $q = 2$, $\mathcal{M} = \{0, 1\}$ that can be equivalently expressed as a coin-weighing problem: what is the minimal number of weighings on an accurate scale to determine all false coins in a set of K coins. The choice of coins for a weighing must not depend on results of previous weighings. This problem was first introduced by Söderberg and Shapiro [12] for $K = 5$. The minimal number of weighings, L , has only been found for a few different values of K in [17]. However, Lindström gives an explicit construction of $L \times \gamma(L+1)$ binary (alphabet $\{0, 1\}$) and $L \times \gamma(L) + 1$ antipodal (alphabet $\{\pm 1\}$) detecting matrices [14], where $\gamma(L)^i$ is the number of ones in the binary expansion of all positive integers less than L . He also proved that the lower bound in the case of $\mathcal{M} = \{0, 1\}$ or $\{\pm 1\}$ is

$$\lim_{K \rightarrow \infty} \frac{f_2(K) \log K}{K} = 2. \quad (2)$$

Cantor and Mills [16] constructed a class of $2^i \times (i+2)2^{(i-1)}$ ternary (alphabet $\{0, \pm 1\}$) detecting matrices for $i \in \mathbb{Z}^+$,

ⁱAs an example, $\gamma(8) = 12$.

which implies that in the case of $\mathcal{M} = \{0, \pm 1\}$ the lower bound is

$$\lim_{K \rightarrow \infty} \frac{f_3(K) \log K}{K} \leq 2. \quad (3)$$

In the literature, most of the explicit construction of UD code sets are recursive [18]–[35]. To the best of our knowledge, it is worth mentioning that the maximum number of vectors of the explicit constructions of binary, antipodal and ternary code sets are $K_{max}^b = \gamma(L+1)$, $K_{max}^a = \gamma(L)+1$ and $K_{max}^t = (i+2)2^{(i-1)}$, as shown in Table 1, Table 2 and Table 3, respectively. Several authors have proposed decoders with linear complexity in noiseless scenarios for the explicit construction, where the detecting matrix has the known-to-us maximum number of vectors, K_{max} .

For noisy channels, recently in [28], a class of antipodal code sequences, for overloaded CDM systems with simplified two-stage maximum-likelihood (ML) detection, has been proposed. In addition to that, other overloaded matrices over the ternary alphabet are introduced in [29] with a low-complexity decoding algorithm. Similarly, in [30] the authors propose overloaded code sets over the ternary alphabet that has a twin tree structured cross-correlation hierarchy that can be decoded with a simple multi-stage detector. Yet another construction of ternary codes that increases the number of columns, K , of UD codes for a such length compared to those proposed in [29] and [30] with a low-complexity polynomial time decoder is proposed in [32]. The primary reason for such low-complexity decoders is that the code sets are constructed with a certain criteria, which entails lowering the maximum number of users $K < K_{max}$, as shown in Table 3.

Apart from, binary, antipodal and ternary UD spreading codes higher alphabet k -ary spreading codes were studied by Lu et al. [56] for the multiple-access adder channels.

In this work, for the first time we consider the problem of designing a low-complexity decoder that has a complexity of $\mathcal{O}(LK \log_2(K))$ for UD code sets in [31] having the maximum number of users K_{max}^a . The code sets presented in [31] are also recursive, which make use of a linear map between vector spaces to Galois field extensions. These UD code sets are one possible construction of all possible distinct UD code sets, shown in Table 2. Simulation results in terms of bit-error-rate (BER) demonstrate that the proposed decoder has a degradation of only 1 – 2 dB in SNR compared to the ML decoder at a BER of 10^{-3} .

Our contributions are summarized as follows:

- (1) We present the proofs for the important **Propositions 1-4** that are presented for the first time in [31]. Those new prepositions are actually the broken down versions of the unique decodability (UD) property.
- (2) Moreover, for the first time, based on **Propositions 1-4**, we formally prove that the maximum number of users for the given $L = 8$ is $K_{max}^a = 13$.
- (3) The minimum Manhattan distance is proved to be 4 for the recursive UD code sets in [31].

TABLE 1. Binary codes.

Year	Authors and Publications	Rows	Columns	Decoder	
				Noiseless	AWGN
1963	Söderberg and Shapiro [12]	L	$< \gamma(L+1)$	No	No
1964	Lindström [14]	L	$\gamma(L+1)^\dagger$	No	No
1966	Cantor and Mills [16]	$2^i - 1$	$i2^{(i-1)}$	No	No
1989	Martirossian and Khachatrian [22]	L	$\gamma(L+1)$	Yes	No
2019	Kulhandjian et al. [34]	L	$\gamma(L+1)$	Yes	Yes

[†] Code set constructions that achieve the maximum number of vectors $K_{\max}$ are presented in bold.

TABLE 2. Antipodal codes.

Year	Authors and Publications	Rows	Columns	Decoder	
				Noiseless	AWGN
1964	Lindström [14]	L	$\gamma(L) + 1$	No	No
1987	Khachatrian and Martirossian [21]	L	$\gamma(L) + 1$	No	No
1995	Khachatrian and Martirossian [23]	2^i	$i2^{(i-1)} + 1$	Yes	No
2012	Kulhandjian and Pados [31]	2^i	$i2^{(i-1)} + 1$	Yes	No

TABLE 3. Ternary codes.

Year	Authors and Publications	Rows	Columns	Decoder	
				Noiseless	AWGN
1966	Cantor and Mills [16]	2^i	$(i+2)2^{(i-1)}$	No	No
1979	Chang and Weldon [18]	2^i	$(i+2)2^{(i-1)}$	Yes	No
1982	Ferguson [19]	2^i	$(i+2)2^{(i-1)}$	Yes	No
1984	Chang [20]	2^i	$(i+2)2^{(i-1)}$	No	No
1998	Khachatrian and Martirossian [24]	2^i	$(i+2)2^{(i-1)}$	Yes	No
2012	Mashayekhi and Marvasti [29]	2^i	$2^{(i+1)} - 1$	Yes	Yes
2016	Singh et al. [30]	2^i	$2^{(i+1)} - 2$	Yes	Yes
2018	Kulhandjian et al. [32]	2^i	$2^{(i+1)} + 2^{(i-2)} - 1$	Yes	Yes
2021	Kulhandjian et al. [35]	2^i	$i2^{(i-1)} + 1$	Yes	Yes

- (4) We develop a low-complexity decoder that has a complexity of $\mathcal{O}(LK \log_2(K))$ for our developed UD code sets in [31] having the maximum number of users $K_{\max}^a$.
- (5) We compute the complexity of the deterministic noiseless detection algorithm (NDA) presented in [31] and perform some complexity analysis for the proposed fast (low-complexity) detection algorithm (FDA).

The rest of the paper is organized as follows. The minimum Manhattan distance of code sets in [31] is presented in Section IV, followed by the proofs of the **Propositions 1-4** and the maximum number of $K_{\max}^a$ for the case of $L = 8$ in Sections II and III, respectively. Detailed discussion of the FDA is presented in Section V. The complexity analysis for both NDA and FDA algorithms is presented in Section VI. After illustrating simulation results in Section VII, a few conclusions are drawn in Section VIII.

The following notations are used in this paper. All boldface lower case letters indicate column vectors and upper case letters indicate matrices, $()^T$ denotes transpose operation, mod

denotes the modulo operation, sgn denotes the sign function, and $|\cdot|$ denotes cardinality of the set.

II. PROOF OF PROPOSITIONS

In order to facilitate the development of the proof it is beneficial to present the UD code set in [31] which are constructed as $\mathbf{C} = [\mathbf{H}_L \mathbf{V}_L] \in \{\pm 1\}^{L \times K}$, where $\mathbf{V}_L \in \{\pm 1\}^{L \times (K-L)}$. We recall that the Sylvester-Hadamard matrix of order 2 is $\mathbf{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ and of order 2^{p+1} for $p = 1, 2, \dots$ is $\mathbf{H}_{2^{p+1}} = \begin{bmatrix} \mathbf{H}_{2^p} & \mathbf{H}_{2^p} \\ \mathbf{H}_{2^p} & -\mathbf{H}_{2^p} \end{bmatrix}$. Then, for any $p = 1, 2, \dots$, $\mathbf{H}_{2^p} \mathbf{H}_{2^p} = 2^p \mathbf{I}_{2^p \times 2^p}$, where $\mathbf{I}_{N \times N}$ is the $N \times N$ identity matrix. We introduce the notation $\mathbf{H}_4 = [\mathbf{h}_0, \mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3]$, $[\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3] \triangleq \begin{bmatrix} [-1, 1, 1, 1]^T, [1, -1, 1, 1]^T, [1, 1, -1, 1]^T, [1, 1, 1, -1]^T \end{bmatrix}$, and the negation function $\mathbf{x}^- \triangleq -\mathbf{x}$. We can see that set of vectors $G = \{\mathbf{h}_0, \mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$ together with $\odot$ operator form a finite group $(G, \odot)$. There

exists an isomorphism φ , shown in Table 4 from G to finite additive Abelian group $(\mathbb{F}_{2^4}, +)$ of extended Galois field $\mathbb{F}_{2^4}$, in other words G is isomorphic to $(\mathbb{F}_{2^4}, +)$, $(G, \odot) \cong (\mathbb{F}_{2^4}, +)$. From linear algebra we know that there is an isomorphism from finite additive groups $(\mathbb{F}_{p^n}, +)$ to vector fields $(\mathbb{F}_p^n, +)$ and to $\mathbb{Z}_p^n$, that is $(\mathbb{F}_{p^n}, +) \cong (\mathbb{F}_p^n, +) \cong \mathbb{Z}_p^n$, [57].

Table 4 shows the mapping of the vectors $\mathbf{h}_0, \dots, \mathbf{h}_3$, $\mathbf{a}_0, \dots, \mathbf{a}_3$ and its negated forms to elements in $\mathbb{F}_{2^4}$ with primitive polynomial $\alpha^4 + \alpha + 1 = 0$, where α is the primitive element in extended Galois field $\text{GF}(2^4)$. Notice carefully that operation of the finite group G is $\odot$, whereas the finite additive group $\mathbb{F}_{2^4}$ is $+$.

TABLE 4. Isomorphism $\varphi: G \mapsto \mathbb{F}_{2^4}$.

Antipodal	Polynomial	Power
$\mathbf{h}_0$	0	0
$\mathbf{h}_2$	1	1
$\mathbf{h}_1$	α	α
$\mathbf{h}_0^-$	α^2	α^2
$\mathbf{a}_1$	α^3	α^3
$\mathbf{h}_3$	$\alpha + 1$	α^4
$\mathbf{h}_1^-$	$\alpha^2 + \alpha$	α^5
$\mathbf{a}_1^-$	$\alpha^3 + \alpha^2$	α^6
$\mathbf{a}_2$	$\alpha^3 + \alpha + 1$	α^7
$\mathbf{h}_2^-$	$\alpha^2 + 1$	α^8
$\mathbf{a}_3$	$\alpha^3 + \alpha$	α^9
$\mathbf{h}_3^-$	$\alpha^2 + \alpha + 1$	α^{10}
$\mathbf{a}_3^-$	$\alpha^3 + \alpha^2 + \alpha$	α^{11}
$\mathbf{a}_2^-$	$\alpha^3 + \alpha^2 + \alpha + 1$	α^{12}
$\mathbf{a}_0$	$\alpha^3 + \alpha^2 + 1$	α^{13}
$\mathbf{a}_0^-$	$\alpha^3 + 1$	α^{14}

With the above formulation, the columns of $\mathbf{V}_L$, $\mathbf{F}_\varphi: \{\pm 1\}^{L \times 1} \mapsto \mathbb{F}_{2^4}^{L' \times 1}$, $\mathbf{f}_i = \mathbf{F}_\varphi(\mathbf{v}_i)$ maps the i -th column $\mathbf{v}_i$ to $\mathbf{f}_i$, for $i = 0, \dots, K - L - 1$, where $\mathbf{f}_i = [f_{i,0} f_{i,1}, \dots, f_{i,L'}]^T$, $f_{i,j} \in \{0, 1, \alpha, \dots, \alpha^{14}\}$, $0 \leq j \leq L'$, $L' = L/4 = 2^{p-2}$ for $p \geq 2$. As an example for $\mathbf{V}_8$ the mapped vectors are as two-dimensional vectors $\mathbf{f}_i = [f_{i,0} f_{i,1}]^T$.

In order to prove **Proposition 1-4** that was first presented in [31], we will expose some very interesting claims. One property of the Hadamard matrix is the following, if we replace 1's in Hadamard matrix with 0's, and replace the -1's with 1's we create $L - 1$ Hadamard binary channel codes, since the first column results in zero vector [58]. Let the Hadamard columns $\mathbf{H}_L = [\mathbf{h}_0, \dots, \mathbf{h}_{L-1}]$ be mapped into linear binary codes $\mathbf{B}_L = [\mathbf{b}_0, \dots, \mathbf{b}_{L-1}]$. There is an isomorphism between binary addition and multiplication of $\{\pm 1\}$ elements. Consequently, since binary addition of Hadamard linear codes is a Hadamard code itself then it is equivalent to element-wise multiplication of any Hadamard codes $\mathbf{H}_L$ is also in $\mathbf{H}_L$. Therefore, some properties can be derived. We denote the mapping of $\mathbf{h}_i$ for $i = 0, \dots, L - 1$ vectors into binary $\mathbf{b}_i \in \{0, 1\}^{L \times 1}$ vectors by $\mathbf{b}_i = f(\mathbf{h}_i)$ and inverse $\mathbf{h}_i = f^{-1}(\mathbf{b}_i)$, where $\mathbf{H}_L = [\mathbf{h}_0, \dots, \mathbf{h}_{L-1}]$ and $\mathbf{B}_L = [\mathbf{b}_0, \dots, \mathbf{b}_{L-1}]$. Define the $f(\mathbf{x}) = (1 - \mathbf{x})/2$ and the inverse $f^{-1}(\mathbf{y}) = [(-1)^{y_0}, \dots, (-1)^{y_{L-1}}]$ functions. Then, the j th column corresponds to linear combination of

TABLE 5. Binary addition and $\{\pm 1\}$ multiplication.

Binary	Bipolar
$0 + 0 = 0$	$1 \times 1 = 1$
$0 + 1 = 1$	$1 \times -1 = -1$
$1 + 0 = 1$	$-1 \times 1 = -1$
$1 + 1 = 0$	$-1 \times -1 = 1$

$\mathbf{b}_j = j_1 \mathbf{b}_1 \oplus j_2 \mathbf{b}_2 \oplus j_4 \mathbf{b}_4 \oplus \dots \oplus j_{2^p} \mathbf{b}_{2^p}$, where j has the binary representation $j = j_1 + 2j_2 + 4j_4 + \dots + 2^p j_{2^p}$ [58]. Here $\oplus$ is modular 2 summation and $j_l \in \{0, 1\}$ for $l = \{1, 2, 4, \dots, 2^p\}$. Therefore, the resulting code is linear code $\mathbf{b}_k = \epsilon_0 \mathbf{b}_0 \oplus \epsilon_1 \mathbf{b}_1 \oplus \dots \oplus \epsilon_{L-1} \mathbf{b}_{L-1}$ for all $\epsilon_i \in \{0, 1\}$, where $\mathbf{b}_k \in \mathbf{B}_L$. Let us look at the problem and assume that $\text{Null}(\mathbf{C}) \in \mathcal{Z}$, where $\mathcal{Z} \in \{0, \pm 1\}^{K \times 1}$ excluding trivial case $\{0\}^{K \times 1}$ and let $\mathbf{z} \in \mathcal{Z}$. Then the nullspace of $\mathbf{C}$ can be formulated as such

$$\mathbf{C}\mathbf{z} = \mathbf{0} \quad (4a)$$

$$[\mathbf{H}_L \mathbf{V}] \mathbf{z} = \mathbf{0}_L \quad (4b)$$

$$\mathbf{H}_L \mathbf{z}_1 + \mathbf{V}_L \mathbf{z}_2 = \mathbf{0} \quad (4c)$$

$$\mathbf{H}_L \mathbf{z}_1 = -\mathbf{V}_L \mathbf{z}_2. \quad (4d)$$

where $\mathbf{z} = [\mathbf{z}_1^T \mathbf{z}_2^T]^T$, $\mathbf{z}_1 \in \{0, \pm 1\}^{L \times 1}$, $\mathbf{z}_2 \in \{0, \pm 1\}^{(K-L) \times 1}$ and $\mathbf{0} = \{0\}^{L \times 1}$. If we assume $\mathbf{z}_1 = \mathbf{0}$ and for some $\mathbf{z}_2$ (4) is true, then (4) can be expressed in terms of $\mathbf{V}_L$ only as

$$\mathbf{V}_L \mathbf{z}_2 = \mathbf{0}. \quad (5)$$

Claim 1: Element-wise multiplication of $\mathbf{H}_L$'s columns $\mathbf{h}_j = \epsilon_0 \mathbf{h}_0 \odot \epsilon_1 \mathbf{h}_1 \odot \dots \odot \epsilon_{L-1} \mathbf{h}_{L-1}$ for all possible $0 \leq i \leq L - 1$, $\epsilon_i \in \{0, 1\}$ is true, where $\mathbf{h}_j \in \mathbf{H}_L$.

Claim 2: If $\mathbf{V}_L$ does not intersect $\mathbf{0}$ over the non-trivial $\{0, \pm 1\}$ combinations and (4) is true, then $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{K-L-1} \mathbf{v}_{K-L-1} = \alpha \mathbf{h}_j$ must be true for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq K - L - 1$, $\alpha \in \{\pm 1\}$. However, if $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{K-L-1} \mathbf{v}_{K-L-1} = \alpha \mathbf{h}_j$ for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq K - L - 1$, $\alpha \in \{\pm 1\}$ is true then it is not necessary that (4) is also true.

Claim 3: If (5) is true, then $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{K-L-1} \mathbf{v}_{K-L-1} = \pm \mathbf{1}$ must be true for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq K - L - 1$ and $\mathbf{1}$ which is all one vector of dimension $L \times 1$. However, if $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{K-L-1} \mathbf{v}_{K-L-1} = \pm \mathbf{1}$ for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq K - L - 1$ is true then it is not necessary that (5) is also true.

Claim 4: If $\mathbf{V}_L$ does not satisfy (4) and (5) then multiplying any column of $\mathbf{V}_L$ by -1 will still not satisfy (4) and (5). Authors in [31] showed by exhaustive search that in case when $L = 4$ then $K_{\max}^a = 5$ and all the possible candidates are $[-1, 1, 1, 1]^T$, $[1, -1, 1, 1]^T$, $[1, 1, -1, 1]^T$, $[1, 1, 1, -1]^T$ and their negatives according to the Claim 4. Furthermore, we continue to present additional claims. Let $\mathbf{v}_i \in \{\pm 1\}^4$ for $0 \leq i \leq N - 1$ then,

Claim 5: If $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{N-1} \mathbf{v}_{N-1} = \alpha \mathbf{h}_j$ and $\alpha \in \{\pm 1\}$, $\mathbf{h}_j \in \mathbf{H}_4$ then $\forall N \in \mathbb{N}$, $[\mathbf{v}_0, \dots, \mathbf{v}_{N-1}] \mathbf{A}_v \mathbf{1} = \mathbf{H}_4 \mathbf{A}_H \mathbf{1}$ is true for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq N - 1$,

$N' \in \mathbb{N}$ is not necessarily equal to N , $\mathbf{A}_v \in \{0, \pm 1\}^{N \times N}$, $\mathbf{A}_H \in \{0, \pm 1\}^{4 \times N'}$, which have maximum one ± 1 entry in each column and 0's elsewhere.

Claim 6: If $0 = N \bmod 2$ and $\beta_0 \mathbf{v}_0 \odot \beta_1 \mathbf{v}_1 \odot \dots \odot \beta_{N-1} \mathbf{v}_{N-1} = \pm 1$ then $[\mathbf{v}_0, \dots, \mathbf{v}_{N-1}] \mathbf{A}_v \mathbf{1} = \mathbf{0}$ is true for some $\beta_i \in \{0, 1\}$, $0 \leq i \leq N-1$, $\mathbf{A}_v \in \{0, \pm 1\}^{N \times N}$, which have maximum one ± 1 entry in each column and 0's elsewhere, except when $N = 4$ and $\mathbf{v}_0 = [-1, 1, 1, 1]^T$, $\mathbf{v}_1 = [1, -1, 1, 1]^T$, $\mathbf{v}_2 = [1, 1, -1, 1]^T$, $\mathbf{v}_3 = [1, 1, 1, -1]^T$.

The generalization of **Claims 1-6** formalizes the **Propositions 1-4** and hence we presented the proofs.

III. PROOF $K_{max}^a = 13$ FOR $L = 8$

For the case when $L = 8$, we prove that the maximum number of columns we can append to $\mathbf{H}_8$ is actually $K_{max}^a - L = 5$. Note that all proposed UD or another words one-to-one matrix constructions in literature are $\mathbf{C} \subset \mathcal{C}$, where $\mathcal{C}$ is all possible antipodal UD code sets for a given L . In order to prove for the maximum number of possible vectors K_{max}^a , we should look at all possible $\mathbf{V}$ and count how many structure of $\mathbf{V}$ hits any of the forbidden lattice points $\mathbf{H}_m \mathbf{z}_1$ and how many does not. If for a given k column, we count the number of $\mathbf{V}$ that hits any forbidden lattice points is equal to the total number of possible $\mathbf{V}$ vector set, then we know that maximum number of columns of $\mathbf{V}$ that does not hit any forbidden lattice points should be smaller than k .

First, we transform antipodal vectors into polynomials with integer coefficients $\mathbb{Z}[x]$, $\mathbf{F} : \{\pm 1\}^{m \times 1} \mapsto \mathbb{Z}[x]$. Those polynomials represent the row location and number of -1 s or $+1$ s in any antipodal $\mathbf{v} \in \{\pm 1\}^{m \times 1}$ with dimension m . Let the polynomial be

$$G(x) = a_0 x^0 + a_1 x^1 + a_2 x^2 + \dots + a_{m-1} x^{m-1}, \quad (6)$$

where $a_i \in \mathbb{Z}$. Additions of vector $\mathbf{v}$ in vector space is equivalent to the addition of $\mathbb{Z}[x]$ in polynomial space. Each antipodal vectors, $\mathbf{v}_j \in \{\pm 1\}^{m \times 1}$, where $0 \leq j \leq 2^m - 1$, is mapped to the corresponding polynomials $G_n^j(x)$ and $G_p^j(x)$ to represent the $+1$ and -1 of the $\mathbf{v}_j$. As an example, for $m = 4$, the antipodal vector, $\mathbf{v}_{10} = [1, -1, 1, -1]^T$, is mapped to $G_n^{10}(x) = x^1 + x^3$ and $G_p^{10}(x) = x^0 + x^2$ polynomials. Observe that for any antipodal vector $\mathbf{v}_j$, the addition of polynomials, $G_n^j(x) + G_p^j(x) = x^0 + x^1 + \dots + x^{m-1}$. In order to visualize polynomial additions in 2-dimensional Euclidean space we can further transform $\mathbb{Z}[x]$ vector space into $\Lambda \subseteq \mathbb{Z}^2$ integer lattice, $\mathbf{H} : \mathbb{Z}[x] \mapsto \mathbb{Z}^2$. Let us define functions $\sigma_n^j(m) = G_n^j(m)$, $\sigma_p^j(m) = G_p^j(m)$, which are the evaluations of polynomials $G_n^j(x)$ and $G_p^j(x)$ at m , where m is the dimension of vector $\mathbf{v}_j$. By setting the x-axis and y-axis to be $G_n^j(x)$ and $G_p^j(x)$, we can build $\Lambda \subseteq \mathbb{Z}^2$ lattice points, since evaluations $\sigma_n^j(m)$ and $\sigma_p^j(m)$ for each antipodal $\mathbf{v}_j$ vectors are integers. Taking the above example of antipodal vectors having dimension of $m = 4$ the equivalent integer lattice points is shown in Table 6.

Let us define sublattices $\Lambda_H \subseteq \Lambda$ and $\Lambda_{V'} \subseteq \Lambda$ constructed by vectors of $[\mathbf{H}_4 - \mathbf{H}_4]$ and $\mathbf{V}'$, where

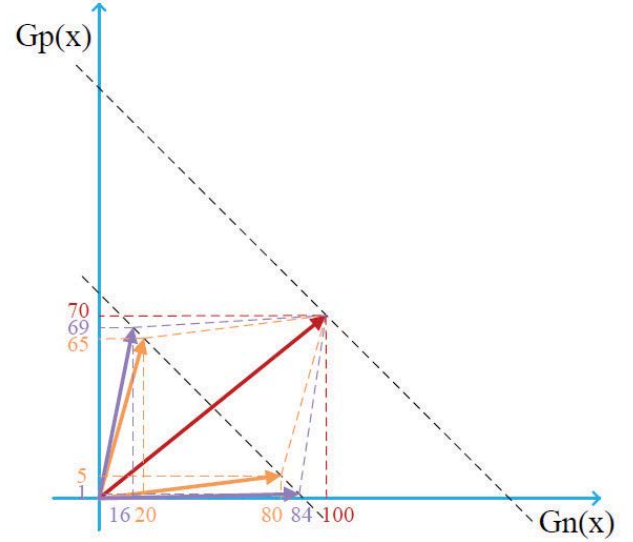


FIGURE 1. Addition of lattice points in $\Lambda \subseteq \mathbb{Z}^2$, $m = 4$.

$\mathbf{V}' \in \{\pm 1\}^{4 \times 8}$, $\mathbf{v}_i' \notin [\mathbf{H}_4 - \mathbf{H}_4]$ for $1 \leq i \leq 8$. As a reminder, for the antipodal UD code set the goal is to construct $\mathbf{V}'$ with the maximum number of columns $K_{max}^a - L$ such that $\Lambda_H \cap \Lambda_{V'} = \emptyset$.

Fig. 1 demonstrates a hit, in other words $\Lambda_H \cap \Lambda_{V'} \neq \emptyset$ for $k = 2$ and $\mathbf{V}' = [\mathbf{v}_4, \mathbf{v}_{14}] = [[1, 1, -1, 1]^T [1, -1, -1, -1]^T]$, which is equivalent to $[16, 69]^T$ and $[84, 1]^T$ in $\Lambda_{V'}$. If we construct the third and the fourth columns to be $\mathbf{v}_6 = [1, -1, -1, 1]^T$ and $\mathbf{v}_{12} = [1, 1, -1, -1]^T$, equivalently $[20, 65]^T$ and $[80, 5]^T$ in Λ_H of $\mathbf{H}_4$ adding them we get $[2, 0, -2, 0]^T$, which is shown as a lattice point $[100, 70]^T$ in Fig. 1. Clearly, addition of $[16, 69]^T$ and $[84, 1]^T$ lattice points also results $[100, 70]^T$, hence, this is why intersection of two sublattices is not an empty set. Therefore, it can be shown that all possible $\binom{8}{2}$ combination of two columns of $\mathbf{V}'$ will at least hit a lattice points of Λ_H , which means that $\Lambda_H \cap \Lambda_{V'} \neq \emptyset$ for $k = 2$. Hence, $|\mathbf{V}'| = 1$ for the case of $m = 4$. Using this approach, we can construct sublattices Λ_H and $\Lambda_{V'}$ for the case of $m = 8$ and find the upper bound of k such that $\Lambda_H \cap \Lambda_{V'} = \emptyset$. This creates a framework to search the solution using geometric combinatorics (e.g. Minkowski sum, Minkowski geometry of numbers [59]), partition and decomposition of Λ into equivalence classes, formed by sublattices and its' cosets to prove the K_{max}^a we can have for a given m and how to generate those vectors in $\mathbf{V}'$.

Polynomials of $G_n^j(x)$ and $G_p^j(x)$ can also be represented by their exponents in the above example $G_n^{10}(x) \rightarrow E_n^{10} = \{1, 3\}$ and $G_p^{10}(x) \rightarrow E_p^{10} = \{0, 2\}$. Therefore, our problem of avoiding non-trivial combinations of $\mathbf{V}$ to hit any forbidden lattice points becomes

$$\bigcup_{\mathbf{h}_i \in \mathbf{H}_m} E_i^t \neq \bigcup_{\mathbf{v}_i \in \mathbf{V}} E_i^t \quad i, j \in \{p, n\}, \quad (7)$$

TABLE 6. Lattice points for all antipodal vectors $\mathbf{v}$ when $m = 4$.

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$G_n^j(m)$	0	1	4	5	16	17	20	21	64	65	68	69	80	81	84	85
$G_p^j(m)$	85	84	81	80	69	68	65	64	21	20	17	16	5	4	1	0

where $\uplus$ denotes the multiset union, element-wise sums of each multisets and $1 \geq t \geq 2^m - 1$ integer value corresponds to antipodal vector. The objective is to avoid $\mathbf{V} \uplus_{\mathbf{v}_i \in \mathbf{V}} E_i^t$ sums of multisets E_p or E_n of $\mathbf{V}$ to hit any forbidden $\uplus_{\mathbf{h}_i \in \mathbf{H}_m} E_i^t$ sum of multisets E_p or E_n of $\mathbf{H}_m$. Obviously, there are 3^m forbidden multisets of $\mathbf{H}_m$ and we do not want $3^k - 1$ multisets of $\mathbf{V}$ excluding trivial case hit any of the forbidden multisets. We can use multiset partition theories and study bipolar vectors by their E_p and E_n representations and prove the maximal number of k .

Note that in our matrix construction design of the $\mathbf{v}_i$'s that they are distinct and not equal to any of columns of $\mathbf{H}_m$ or $-\mathbf{H}_m$. If, however, any of $\mathbf{v}_i \in \pm \mathbf{H}_m$, then the multiset of $\mathbf{v}_i$ hits the forbidden multiset of $\mathbf{H}_m$. Such $\mathbf{v}_i$ vectors can never be included in vector set of UD codes. Additionally, the $\mathbf{v}_i$ s can be replaced by $-\mathbf{v}$ without violating the uniquely decodability property (4). Since all possible combinations of vectors including $-\mathbf{v}$ of $\mathbf{V}$ do not hit any of the forbidden multisets. In other words, if $[\mathbf{H}_m \mathbf{V}] \in \mathcal{C}$ so is any number of columns of $\mathbf{V}$ or $\mathbf{H}_m$ multiplied by -1 .

One way to approach this problem is to classify all $\mathbf{v}$ bipolar vectors into groups then use inclusion-exclusion principle. There are 2^m number of $\mathbf{v}$'s that consist of $|E_n| = i$, where $i = \{0, 1, \dots, m\}$ with all the combinations makes $\sum_{i=0}^m \binom{m}{i} = 2^m$.

Let us divide the total number of m -dimensional bipolar vectors into two classes; $\mathbf{v} \in \mathcal{B}_m^+$ if $0 \leq |E_n| < m/2$, $|E_n| = m/2$ if $|E_p| = \{1, 2, \dots\}$ excluding m $\mathbf{h}_i$'s columns from $\mathbf{H}_m$ and $\mathbf{v} \in \mathcal{B}_m^-$ if $m/2 < |E_n| \leq m$, $|E_n| = m/2$ if $|E_n| = \{1, 2, \dots\}$ excluding m $-\mathbf{h}_i$'s columns from $-\mathbf{H}_m$. It's clear that if any $\mathbf{v} \in \mathcal{B}_m^+$ then $-\mathbf{v} \in \mathcal{B}_m^-$. This narrows for our design to consider only distinct vector sets and the total number of such distinct $\mathbf{v}_i$'s to be considered in our $\mathbf{V}$ design is $\sum_{i=0}^{m/2} \binom{m}{i} - m = 2^{m-1} - m$. We need to construct from distinct vectors $\mathbf{v} \in \mathcal{B}_m^+$ such that they do not hit any forbidden multisets.

Therefore, the total possible number of $\mathbf{V}$ sets with k columns is $\binom{2^{m-1}-m}{k}$. Out of this total number of $\mathbf{V}$ sets with k columns only some satisfy UD or one-to-one condition when appended to $\mathbf{H}_m$. If all the possible $\mathbf{V}$'s do not satisfy one-to-one condition that means k exceeds the maximum number of columns that can be added to $\mathbf{H}_m$. Hence, we want to count how many combinations of $\mathbf{v} \in \mathcal{B}_m^+$ with k columns hit the forbidden multisets. If the number of combination is equal the total number of $\mathbf{V}$ sets then we know that k is not the maximum. Counting that number will help us to prove the maximum number of columns k .

We classify $\mathcal{B}_m^+$ into different groups, so that any combinations of vectors in similar group hits the forbidden multisets

and therefore, such $\mathbf{v}$ that belong to the same group must be avoided in our design.

In our example, for $m = 8$ and $k = 2$, we classify $\mathcal{B}_8^+$ into groups and count how many $\mathbf{V}$'s do not satisfy the UD condition or one-to-one condition out of $\binom{2^{8-1}-8}{2} = \binom{120}{2} = 7140$ possible vector sets. By looking at all possible combinations of $\mathbf{V} = [\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$, $|E_n^{j_1}|, |E_n^{j_2}| \in \{1, \dots, 8\}$ we only consider $(|E_n^{j_1}|, |E_n^{j_2}|) = \{(1, 3), (2, 2), (3, 3), (4, 4)\}$. Since to hit $[\mathbf{h}_i \pm \mathbf{h}_i]$, where $2 \leq i \leq 8$, the vectors $[\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ must have $(|E_n^{j_1}|, |E_n^{j_2}|) = (1, 3)$ and $(|E_n^{j_1}|, |E_n^{j_2}|) = (2, 2)$. We can easily show that if $[\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ hit forbidden multisets with $(|E_n^{j_1}|, |E_n^{j_2}|) = (1, 3)$ and $(|E_n^{j_1}|, |E_n^{j_2}|) = (2, 2)$ then $[-\mathbf{v}_{j_1} - \mathbf{v}_{j_2}]$ with $(|E_n^{j_1}|, |E_n^{j_2}|) = (7, 5)$ and $(|E_n^{j_1}|, |E_n^{j_2}|) = (6, 6)$ hit $[-\mathbf{h}_i \mp \mathbf{h}_i]$ and $l \neq i$. Also, if $[\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ hit forbidden multisets with $(|E_n^{j_1}|, |E_n^{j_2}|) = (2, 2)$ then $[-\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ and $[\mathbf{v}_{j_1} - \mathbf{v}_{j_2}]$ with $(|E_n^{j_1}|, |E_n^{j_2}|) = (2, 6)$ hit $[\pm \mathbf{h}_i \pm \mathbf{h}_p]$ and $i \neq l, p \neq i$. Hitting $[\pm \mathbf{h}_i \pm \mathbf{h}_j]$, where $2 \leq i \neq j \leq 8$ multisets then vectors must have $(|E_n^{j_1}|, |E_n^{j_2}|) = (3, 5)$ and $(|E_n^{j_1}|, |E_n^{j_2}|) = (4, 4)$. Since we only work with distinct vectors $\mathcal{B}_8^+$ then $(|E_n^{j_1}|, |E_n^{j_2}|) = (3, 5)$ that has $[\mathbf{v}_{j_1} - \mathbf{v}_{j_2}]$ is equivalent to $[\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ with $(|E_n^{j_1}|, |E_n^{j_2}|) = (3, 3)$.

Now, let us classify $(|E_n^{j_1}|, |E_n^{j_2}|) = (1, 3), (|E_n^{j_1}|, |E_n^{j_2}|) = (2, 2), (|E_n^{j_1}|, |E_n^{j_2}|) = (3, 3), (|E_n^{j_1}|, |E_n^{j_2}|) = (4, 4)$ into $\mathcal{A}_i$'s, $\mathcal{D}_i$'s, $\mathcal{F}_i$'s and $\mathcal{G}_i$'s groups. We know that the total number of $\mathbf{v}$ s with $|E_n| = 1$ is $\binom{8}{1} = 8$, $|E_n| = 2$ is $\binom{8}{2} = 28$, $|E_n| = 3$ is $\binom{8}{3} = 56$ and $|E_n| = 4$ is $\binom{8}{4} = 70$.

Here is how we divide 56 of $|E_n| = 3$ and 8 of $|E_n| = 1$ into $\mathcal{A}_i$'s, $1 \leq i \leq 8$ groups.

$$\mathcal{A}_1 = \begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{bmatrix},$$

$$\mathcal{A}_2 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \end{bmatrix},$$

$$\mathcal{A}_3 = \begin{bmatrix} 1 \\ 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \end{bmatrix},$$

$$\mathcal{A}_4 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \end{bmatrix},$$

$$\mathcal{A}_5 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \end{bmatrix},$$

$$\mathcal{A}_6 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \end{bmatrix},$$

$$\mathcal{A}_7 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \end{bmatrix},$$

$$\mathcal{A}_8 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

equivalently, we can then write them in multiset form as

$$\begin{aligned} \mathcal{A}_1 &= \begin{aligned} &E_p = \{1, 2, 3, 4, 5, 6, 7\} \quad E_n = \{0\} \\ &E_p = \{0, 4, 5, 6, 7\} \quad E_n = \{1, 2, 3\} \\ &E_p = \{0, 2, 3, 6, 7\} \quad E_n = \{1, 4, 5\} \\ &E_p = \{0, 2, 3, 4, 5\} \quad E_n = \{1, 6, 7\} \\ &E_p = \{0, 1, 3, 5, 7\} \quad E_n = \{2, 4, 6\} \\ &E_p = \{0, 1, 3, 4, 6\} \quad E_n = \{2, 5, 7\} \\ &E_p = \{0, 1, 2, 5, 6\} \quad E_n = \{3, 4, 7\} \\ &E_p = \{0, 1, 2, 4, 7\} \quad E_n = \{3, 5, 6\} \end{aligned} \\ \mathcal{A}_2 &= \begin{aligned} &E_p = \{0, 2, 3, 4, 5, 6, 7\} \quad E_n = \{1\} \\ &E_p = \{1, 4, 5, 6, 7\} \quad E_n = \{0, 2, 3\} \\ &E_p = \{1, 2, 3, 6, 7\} \quad E_n = \{0, 4, 5\} \\ &E_p = \{1, 2, 3, 4, 5\} \quad E_n = \{0, 6, 7\} \\ &E_p = \{0, 1, 3, 5, 6\} \quad E_n = \{2, 4, 7\} \\ &E_p = \{0, 1, 3, 4, 7\} \quad E_n = \{2, 5, 6\} \\ &E_p = \{0, 1, 2, 5, 7\} \quad E_n = \{3, 4, 6\} \\ &E_p = \{0, 1, 2, 4, 6\} \quad E_n = \{3, 5, 7\} \end{aligned} \\ \mathcal{A}_3 &= \begin{aligned} &E_p = \{0, 1, 3, 4, 5, 6, 7\} \quad E_n = \{2\} \\ &E_p = \{2, 4, 5, 6, 7\} \quad E_n = \{0, 1, 3\} \\ &E_p = \{1, 2, 3, 5, 7\} \quad E_n = \{0, 4, 6\} \\ &E_p = \{1, 2, 3, 4, 6\} \quad E_n = \{0, 5, 7\} \\ &E_p = \{0, 2, 3, 5, 6\} \quad E_n = \{1, 4, 7\} \\ &E_p = \{0, 2, 3, 4, 7\} \quad E_n = \{1, 5, 6\} \\ &E_p = \{0, 1, 2, 6, 7\} \quad E_n = \{3, 4, 5\} \\ &E_p = \{0, 1, 2, 4, 5\} \quad E_n = \{3, 6, 7\} \end{aligned} \\ \mathcal{A}_4 &= \begin{aligned} &E_p = \{0, 1, 2, 4, 5, 6, 7\} \quad E_n = \{3\} \\ &E_p = \{1, 4, 5, 6, 7\} \quad E_n = \{0, 1, 2\} \\ &E_p = \{1, 2, 3, 6, 7\} \quad E_n = \{0, 4, 7\} \\ &E_p = \{1, 2, 3, 4, 5\} \quad E_n = \{0, 5, 6\} \\ &E_p = \{0, 1, 3, 5, 6\} \quad E_n = \{1, 4, 6\} \\ &E_p = \{0, 1, 3, 4, 7\} \quad E_n = \{1, 5, 7\} \\ &E_p = \{0, 1, 2, 5, 7\} \quad E_n = \{2, 4, 5\} \\ &E_p = \{0, 1, 2, 4, 6\} \quad E_n = \{2, 6, 7\} \end{aligned} \\ \mathcal{A}_5 &= \begin{aligned} &E_p = \{0, 1, 2, 3, 5, 6, 7\} \quad E_n = \{4\} \\ &E_p = \{2, 3, 4, 6, 7\} \quad E_n = \{0, 1, 5\} \\ &E_p = \{1, 3, 4, 5, 7\} \quad E_n = \{0, 2, 6\} \\ &E_p = \{1, 2, 4, 5, 6\} \quad E_n = \{0, 3, 7\} \\ &E_p = \{0, 3, 4, 5, 6\} \quad E_n = \{1, 2, 7\} \\ &E_p = \{0, 2, 4, 5, 7\} \quad E_n = \{1, 3, 6\} \\ &E_p = \{0, 1, 4, 6, 7\} \quad E_n = \{2, 3, 5\} \\ &E_p = \{0, 1, 2, 3, 4\} \quad E_n = \{5, 6, 7\} \end{aligned} \\ \mathcal{A}_6 &= \begin{aligned} &E_p = \{0, 1, 2, 3, 4, 6, 7\} \quad E_n = \{5\} \\ &E_p = \{2, 3, 5, 6, 7\} \quad E_n = \{0, 1, 4\} \\ &E_p = \{1, 3, 4, 5, 6\} \quad E_n = \{0, 2, 7\} \\ &E_p = \{1, 2, 4, 5, 7\} \quad E_n = \{0, 3, 6\} \\ &E_p = \{0, 3, 4, 5, 7\} \quad E_n = \{1, 2, 6\} \\ &E_p = \{0, 2, 4, 5, 6\} \quad E_n = \{1, 3, 7\} \\ &E_p = \{0, 1, 5, 6, 7\} \quad E_n = \{2, 3, 4\} \\ &E_p = \{0, 1, 2, 3, 5\} \quad E_n = \{4, 6, 7\} \end{aligned} \end{aligned}$$

$$\begin{aligned}
& E_p = \{0, 1, 2, 3, 4, 5, 7\} \quad E_n = \{6\} \\
& E_p = \{2, 3, 4, 5, 6\} \quad E_n = \{0, 1, 7\} \\
& E_p = \{1, 3, 5, 6, 7\} \quad E_n = \{0, 2, 4\} \\
\mathcal{A}_7 = & \begin{aligned} & E_p = \{1, 2, 4, 6, 7\} \quad E_n = \{0, 3, 5\} \\ & E_p = \{0, 3, 4, 6, 7\} \quad E_n = \{1, 2, 5\} \\ & E_p = \{0, 2, 5, 6, 7\} \quad E_n = \{1, 3, 4\} \\ & E_p = \{0, 1, 4, 5, 6\} \quad E_n = \{2, 3, 7\} \\ & E_p = \{0, 1, 2, 3, 6\} \quad E_n = \{4, 5, 7\} \end{aligned} \\
& E_p = \{0, 1, 2, 3, 4, 6, 7\} \quad E_n = \{7\} \\
& E_p = \{2, 3, 4, 5, 7\} \quad E_n = \{0, 1, 6\} \\
& E_p = \{1, 3, 4, 6, 7\} \quad E_n = \{0, 2, 5\} \\
\mathcal{A}_8 = & \begin{aligned} & E_p = \{1, 2, 5, 6, 7\} \quad E_n = \{0, 3, 4\} \\ & E_p = \{0, 3, 5, 6, 7\} \quad E_n = \{1, 2, 4\} \\ & E_p = \{0, 2, 4, 6, 7\} \quad E_n = \{1, 3, 5\} \\ & E_p = \{0, 1, 4, 5, 7\} \quad E_n = \{2, 3, 6\} \\ & E_p = \{0, 1, 2, 3, 7\} \quad E_n = \{4, 5, 6\} \end{aligned}
\end{aligned}$$

We can prove that

$$E_i^1 + E_i^t = E_j^{t_1} + E_j^{t_2} \quad i, j \in \{p, n\}, \quad (8)$$

holds only if $t_1 = \{t_1 | \mathbf{a}_{t_1} \in \mathcal{A}_h\}$, $t_2 = \{t_2 | \mathbf{a}_{t_2} \in \mathcal{A}_h\}$, $t_1 \neq t_2$, $\mathbf{h}_t \in \mathbf{H}_8 | t \in \{2, \dots, 8\}$ and $h \in \{1, \dots, 8\}$ and does not hold if $t_1 = \{t_1 | \mathbf{a}_{t_1} \in \mathcal{A}_{h_1}\}$, and $t_2 = \{t_2 | \mathbf{a}_{t_2} \in \mathcal{A}_{h_2}\}$, where $h_1 \neq h_2$. Therefore, we can conclude that we can append two vectors $\mathbf{v}_1 \in \mathcal{A}_j$ and $\mathbf{v}_2 \in \mathcal{A}_i$ to $\mathbf{H}_m$, which do not hit any forbidden multisets by choosing any two vectors from different groups $1 \leq i \neq j \leq 8$.

Here is how we divide 28 of $|E_n| = 2$ into $\mathcal{D}_i$'s, $1 \leq i \leq 7$ groups.

$$\begin{aligned}
\mathcal{D}_1 = & \begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \end{bmatrix}, \quad \mathcal{D}_2 = \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}, \\
\mathcal{D}_3 = & \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \end{bmatrix}, \quad \mathcal{D}_4 = \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \end{bmatrix}, \\
\mathcal{D}_5 = & \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \end{bmatrix}, \quad \mathcal{D}_6 = \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix},
\end{aligned}$$

$$\mathcal{D}_7 = \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}$$

equivalently, we can then write in multiset form as

$$\begin{aligned}
& E_p = \{2, 3, 4, 5, 6, 7\} \quad E_n = \{0, 1\} \\
\mathcal{D}_1 = & \begin{aligned} & E_p = \{0, 1, 4, 5, 6, 7\} \quad E_n = \{2, 3\} \\ & E_p = \{0, 1, 2, 3, 6, 7\} \quad E_n = \{4, 5\} \\ & E_p = \{0, 1, 2, 3, 4, 5\} \quad E_n = \{6, 7\} \end{aligned} \\
& E_p = \{1, 3, 4, 5, 6, 7\} \quad E_n = \{0, 2\} \\
\mathcal{D}_2 = & \begin{aligned} & E_p = \{0, 2, 4, 5, 6, 7\} \quad E_n = \{1, 3\} \\ & E_p = \{0, 1, 2, 3, 5, 7\} \quad E_n = \{4, 6\} \\ & E_p = \{0, 1, 2, 3, 4, 6\} \quad E_n = \{5, 7\} \end{aligned} \\
& E_p = \{1, 2, 4, 5, 6, 7\} \quad E_n = \{0, 3\} \\
\mathcal{D}_3 = & \begin{aligned} & E_p = \{0, 3, 4, 5, 6, 7\} \quad E_n = \{1, 2\} \\ & E_p = \{0, 1, 2, 3, 5, 6\} \quad E_n = \{4, 7\} \\ & E_p = \{0, 1, 2, 3, 4, 7\} \quad E_n = \{5, 6\} \end{aligned} \\
& E_p = \{1, 2, 3, 5, 6, 7\} \quad E_n = \{0, 4\} \\
\mathcal{D}_4 = & \begin{aligned} & E_p = \{0, 2, 3, 4, 6, 7\} \quad E_n = \{1, 5\} \\ & E_p = \{0, 1, 3, 4, 5, 7\} \quad E_n = \{2, 6\} \\ & E_p = \{0, 1, 2, 4, 5, 6\} \quad E_n = \{3, 7\} \end{aligned} \\
& E_p = \{1, 2, 3, 4, 6, 7\} \quad E_n = \{0, 5\} \\
\mathcal{D}_5 = & \begin{aligned} & E_p = \{0, 2, 3, 5, 6, 7\} \quad E_n = \{1, 4\} \\ & E_p = \{0, 1, 3, 4, 5, 6\} \quad E_n = \{2, 7\} \\ & E_p = \{0, 1, 2, 4, 5, 7\} \quad E_n = \{3, 6\} \end{aligned} \\
& E_p = \{1, 2, 3, 4, 5, 7\} \quad E_n = \{0, 6\} \\
\mathcal{D}_6 = & \begin{aligned} & E_p = \{0, 2, 3, 4, 5, 6\} \quad E_n = \{1, 7\} \\ & E_p = \{0, 1, 3, 5, 6, 7\} \quad E_n = \{2, 4\} \\ & E_p = \{0, 1, 2, 4, 6, 7\} \quad E_n = \{3, 5\} \end{aligned} \\
& E_p = \{1, 2, 3, 4, 5, 6\} \quad E_n = \{0, 7\} \\
\mathcal{D}_7 = & \begin{aligned} & E_p = \{0, 2, 3, 4, 5, 7\} \quad E_n = \{1, 6\} \\ & E_p = \{0, 1, 3, 4, 6, 7\} \quad E_n = \{2, 5\} \\ & E_p = \{0, 1, 2, 5, 6, 7\} \quad E_n = \{3, 4\} \end{aligned}
\end{aligned}$$

It can be proved that

$$E_i^1 + E_i^t = E_j^{t_1} + E_j^{t_2} \quad i, j \in \{p, n\}, \quad (9)$$

and

$$E_i^{t'} + E_i^{t''} = E_j^{t_1} + E_j^{t_2} \quad i, j \in \{p, n\}, \quad (10)$$

holds only if $t_1 = \{t_1 | \mathbf{d}_{t_1} \in \mathcal{D}_h\}$, $t_2 = \{t_2 | \mathbf{d}_{t_2} \in \mathcal{D}_h\}$, $t_1 \neq t_2$, where $\mathbf{h}_t, \mathbf{h}_{t'}, \mathbf{h}_{t''} \in \mathbf{H}_8 | t, t', t'' \in \{2, \dots, 8\}$ and $h \in \{1, \dots, 7\}$ and does not hold if $t_1 = \{t_1 | \mathbf{d}_{t_1} \in \mathcal{D}_{h_1}\}$, and $t_2 = \{t_2 | \mathbf{d}_{t_2} \in \mathcal{D}_{h_2}\}$, where $h_1 \neq h_2$. Therefore, we can conclude that we can append two vectors $\mathbf{v}_1 \in \mathcal{D}_j$ and $\mathbf{v}_2 \in \mathcal{D}_i$ to $\mathbf{H}_m$, which do not hit any forbidden multisets by choosing any two vectors from different groups $1 \leq i \neq j \leq 7$.

Dividing 56 of $|E_n| = 3$ into $\mathcal{F}_i$, $1 \leq i \leq 8$ groups is equivalent exactly $\mathcal{A}_i$'s. Similarly, we can prove that

$$E_i^{t'} + E_i^{t''} = E_j^{t_1} + E_j^{t_2} \quad i, j \in \{p, n\}, \quad (11)$$

holds only if $t_1 = \{t_1 | \mathbf{f}_{t_1} \in \mathcal{F}_h\}$, $t_2 = \{t_2 | \mathbf{f}_{t_2} \in \mathcal{F}_h\}$, $t_1 \neq t_2$, where $\mathbf{h}_{t'}, \mathbf{h}_{t''} \in \mathcal{H}_8 | t', t'' \in \{2, \dots, 8\}$ and $h \in \{1, \dots, 8\}$ and does not hold if $t_1 = \{t_1 | \mathbf{f}_{t_1} \in \mathcal{F}_{h_1}\}$, and $t_2 = \{t_2 | \mathbf{f}_{t_2} \in \mathcal{F}_{h_2}\}$, where $h_1 \neq h_2$. Therefore, we can conclude that we can append two vectors $\mathbf{v}_1 \in \mathcal{F}_j$ and $\mathbf{v}_2 \in \mathcal{F}_i$ to $\mathbf{H}_m$, which do not hit any forbidden multisets by choosing any two vectors from different groups $1 \leq i \neq j \leq 8$.

Here is how we divide $70/2 - 7 = 28$ of $|E_n| = 4$ into $\mathcal{G}_i$'s, $1 \leq i \leq 7$ groups.

$$\begin{aligned} \mathcal{G}_1 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 \end{bmatrix}, \quad \mathcal{G}_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 \end{bmatrix}, \\ \mathcal{G}_3 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \quad \mathcal{G}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \\ \mathcal{G}_5 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \quad \mathcal{G}_6 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \\ \mathcal{G}_7 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \end{aligned}$$

equivalently, we can then write in multiset form as

$$\begin{aligned} \mathcal{G}_1 &= \begin{aligned} E_p &= \{0, 4, 5, 6\} \quad E_n = \{1, 2, 3, 7\} \\ E_p &= \{0, 2, 3, 6\} \quad E_n = \{1, 4, 5, 7\} \\ E_p &= \{0, 1, 3, 5\} \quad E_n = \{2, 4, 6, 7\} \\ E_p &= \{0, 1, 2, 4\} \quad E_n = \{3, 5, 6, 7\} \end{aligned} \\ \mathcal{G}_2 &= \begin{aligned} E_p &= \{0, 4, 5, 7\} \quad E_n = \{1, 2, 3, 6\} \\ E_p &= \{0, 2, 3, 7\} \quad E_n = \{1, 4, 5, 6\} \\ E_p &= \{0, 1, 3, 4\} \quad E_n = \{2, 5, 6, 7\} \\ E_p &= \{0, 1, 2, 5\} \quad E_n = \{3, 4, 6, 7\} \end{aligned} \end{aligned}$$

$$\begin{aligned} \mathcal{G}_3 &= \begin{aligned} E_p &= \{0, 4, 6, 7\} \quad E_n = \{1, 2, 3, 5\} \\ E_p &= \{0, 2, 3, 4\} \quad E_n = \{1, 5, 6, 7\} \\ E_p &= \{0, 1, 3, 7\} \quad E_n = \{2, 4, 5, 6\} \\ E_p &= \{0, 1, 5, 6\} \quad E_n = \{2, 3, 4, 7\} \end{aligned} \\ \mathcal{G}_4 &= \begin{aligned} E_p &= \{0, 5, 6, 7\} \quad E_n = \{1, 2, 3, 4\} \\ E_p &= \{0, 2, 3, 5\} \quad E_n = \{1, 4, 6, 7\} \\ E_p &= \{0, 1, 3, 6\} \quad E_n = \{2, 4, 5, 7\} \\ E_p &= \{0, 1, 2, 7\} \quad E_n = \{3, 4, 5, 6\} \end{aligned} \\ \mathcal{G}_5 &= \begin{aligned} E_p &= \{0, 2, 6, 7\} \quad E_n = \{1, 3, 4, 5\} \\ E_p &= \{0, 2, 4, 5\} \quad E_n = \{1, 3, 6, 7\} \\ E_p &= \{0, 1, 5, 7\} \quad E_n = \{2, 3, 4, 6\} \\ E_p &= \{0, 1, 4, 6\} \quad E_n = \{2, 3, 5, 7\} \end{aligned} \\ \mathcal{G}_6 &= \begin{aligned} E_p &= \{0, 3, 6, 7\} \quad E_n = \{1, 2, 4, 5\} \\ E_p &= \{0, 3, 4, 5\} \quad E_n = \{1, 2, 6, 7\} \\ E_p &= \{0, 1, 5, 6\} \quad E_n = \{2, 3, 4, 7\} \\ E_p &= \{0, 1, 4, 7\} \quad E_n = \{2, 3, 5, 6\} \end{aligned} \\ \mathcal{G}_7 &= \begin{aligned} E_p &= \{0, 3, 5, 7\} \quad E_n = \{1, 2, 4, 6\} \\ E_p &= \{0, 3, 4, 6\} \quad E_n = \{1, 2, 5, 7\} \\ E_p &= \{0, 2, 5, 6\} \quad E_n = \{1, 3, 4, 7\} \\ E_p &= \{0, 2, 4, 7\} \quad E_n = \{1, 3, 5, 6\} \end{aligned} \end{aligned}$$

It can be proved that

$$E_i^{t'} + E_i^{t''} = E_j^{t_1} + E_j^{t_2} \quad i, j \in \{p, n\}, \quad (12)$$

holds only if $t_1 = \{t_1 | \mathbf{g}_{t_1} \in \mathcal{G}_h\}$, $t_2 = \{t_2 | \mathbf{g}_{t_2} \in \mathcal{G}_h\}$, $t_1 \neq t_2$, where $\mathbf{h}_{t'}, \mathbf{h}_{t''} \in \mathbf{H}_8 | t', t'' \in \{2, \dots, 8\}$ and $h \in \{1, \dots, 7\}$ and does not hold if $t_1 = \{t_1 | \mathbf{g}_{t_1} \in \mathcal{G}_{h_1}\}$, and $t_2 = \{t_2 | \mathbf{g}_{t_2} \in \mathcal{G}_{h_2}\}$, where $h_1 \neq h_2$. Therefore, we can conclude that we can append two vectors $\mathbf{v}_1 \in \mathcal{G}_j$ and $\mathbf{v}_2 \in \mathcal{G}_i$ to $\mathbf{H}_m$, which do not hit any forbidden multisets by choosing any two vectors from different groups $1 \leq i \neq j \leq 7$. Note that $\mathcal{A}_i$'s group can be constructed from $\mathcal{A}_i$'s and $\mathcal{D}_i$'s such as $\mathcal{A}_1 = \{\mathcal{A}_2 + \mathcal{D}_1, \mathcal{A}_3 + \mathcal{D}_2, \mathcal{A}_4 + \mathcal{D}_3, \mathcal{A}_5 + \mathcal{D}_4, \mathcal{A}_6 + \mathcal{D}_5, \mathcal{A}_7 + \mathcal{D}_6, \mathcal{A}_8 + \mathcal{D}_7\}$, $\mathcal{A}_2 = \{\mathcal{A}_1 + \mathcal{D}_1, \mathcal{A}_3 + \mathcal{D}_3, \mathcal{A}_4 + \mathcal{D}_2, \mathcal{A}_5 + \mathcal{D}_5, \mathcal{A}_6 + \mathcal{D}_4, \mathcal{A}_7 + \mathcal{D}_7, \mathcal{A}_8 + \mathcal{D}_6\}$, $\mathcal{A}_3 = \{\mathcal{A}_1 + \mathcal{D}_2, \mathcal{A}_2 + \mathcal{D}_3, \mathcal{A}_4 + \mathcal{D}_1, \mathcal{A}_5 + \mathcal{D}_6, \mathcal{A}_6 + \mathcal{D}_7, \mathcal{A}_7 + \mathcal{D}_4, \mathcal{A}_8 + \mathcal{D}_5\}$, $\mathcal{A}_4 = \{\mathcal{A}_1 + \mathcal{D}_3, \mathcal{A}_2 + \mathcal{D}_2, \mathcal{A}_3 + \mathcal{D}_1, \mathcal{A}_5 + \mathcal{D}_7, \mathcal{A}_6 + \mathcal{D}_6, \mathcal{A}_7 + \mathcal{D}_5, \mathcal{A}_8 + \mathcal{D}_4\}$, $\mathcal{A}_5 = \{\mathcal{A}_1 + \mathcal{D}_4, \mathcal{A}_2 + \mathcal{D}_5, \mathcal{A}_3 + \mathcal{D}_6, \mathcal{A}_4 + \mathcal{D}_7, \mathcal{A}_6 + \mathcal{D}_1, \mathcal{A}_7 + \mathcal{D}_2, \mathcal{A}_8 + \mathcal{D}_3\}$, $\mathcal{A}_6 = \{\mathcal{A}_1 + \mathcal{D}_5, \mathcal{A}_2 + \mathcal{D}_4, \mathcal{A}_3 + \mathcal{D}_7, \mathcal{A}_4 + \mathcal{D}_6, \mathcal{A}_5 + \mathcal{D}_1, \mathcal{A}_7 + \mathcal{D}_3, \mathcal{A}_8 + \mathcal{D}_2\}$, $\mathcal{A}_7 = \{\mathcal{A}_1 + \mathcal{D}_6, \mathcal{A}_2 + \mathcal{D}_7, \mathcal{A}_3 + \mathcal{D}_4, \mathcal{A}_4 + \mathcal{D}_5, \mathcal{A}_5 + \mathcal{D}_2, \mathcal{A}_6 + \mathcal{D}_3, \mathcal{A}_8 + \mathcal{D}_1\}$, $\mathcal{A}_8 = \{\mathcal{A}_1 + \mathcal{D}_7, \mathcal{A}_2 + \mathcal{D}_6, \mathcal{A}_3 + \mathcal{D}_5, \mathcal{A}_4 + \mathcal{D}_4, \mathcal{A}_5 + \mathcal{D}_3, \mathcal{A}_6 + \mathcal{D}_2, \mathcal{A}_7 + \mathcal{D}_1\}$. Also the $\mathcal{G}_i$'s group can be constructed from different combinations of $\mathcal{A}_i$'s or $\mathcal{D}_i$'s groups. For example, $\mathcal{G}_1 = \{\mathcal{A}_1 + \mathcal{A}_8, \mathcal{A}_2 + \mathcal{A}_7, \mathcal{A}_3 + \mathcal{A}_6, \mathcal{A}_4 + \mathcal{A}_5\}$, $\mathcal{G}_2 = \{\mathcal{A}_1 + \mathcal{A}_7, \mathcal{A}_2 + \mathcal{A}_8, \mathcal{A}_3 + \mathcal{A}_5, \mathcal{A}_4 + \mathcal{A}_6\}$, $\mathcal{G}_3 = \{\mathcal{A}_1 + \mathcal{A}_6, \mathcal{A}_2 + \mathcal{A}_5, \mathcal{A}_3 + \mathcal{A}_8, \mathcal{A}_4 + \mathcal{A}_7\}$, $\mathcal{G}_4 = \{\mathcal{A}_1 + \mathcal{A}_5, \mathcal{A}_2 + \mathcal{A}_6, \mathcal{A}_3 + \mathcal{A}_7, \mathcal{A}_4 + \mathcal{A}_8\}$, $\mathcal{G}_5 = \{\mathcal{A}_1 + \mathcal{A}_4, \mathcal{A}_2 + \mathcal{A}_3, \mathcal{A}_5 + \mathcal{A}_8, \mathcal{A}_6 + \mathcal{A}_7\}$, $\mathcal{G}_6 = \{\mathcal{A}_1 + \mathcal{A}_3, \mathcal{A}_2 + \mathcal{A}_4, \mathcal{A}_5 + \mathcal{A}_7, \mathcal{A}_6 + \mathcal{A}_8\}$, $\mathcal{G}_7 = \{\mathcal{A}_1 + \mathcal{A}_2, \mathcal{A}_3 + \mathcal{A}_4, \mathcal{A}_5 + \mathcal{A}_6, \mathcal{A}_7 + \mathcal{A}_8\}$ and $\mathcal{G}_1 = \{\mathcal{D}_1 + \mathcal{D}_6, \mathcal{D}_2 + \mathcal{D}_5, \mathcal{D}_3 + \mathcal{D}_4\}$, $\mathcal{G}_2 = \{\mathcal{D}_1 + \mathcal{D}_7, \mathcal{D}_2 + \mathcal{D}_4, \mathcal{D}_3 + \mathcal{D}_5\}$, $\mathcal{G}_3 = \{\mathcal{D}_1 + \mathcal{D}_4, \mathcal{D}_2 + \mathcal{D}_7, \mathcal{D}_3 + \mathcal{D}_6\}$, $\mathcal{G}_4 = \{\mathcal{D}_1 + \mathcal{D}_5, \mathcal{D}_2 + \mathcal{D}_6, \mathcal{D}_3 + \mathcal{D}_7\}$, $\mathcal{G}_5 = \{\mathcal{D}_1 + \mathcal{D}_2, \mathcal{D}_4 + \mathcal{D}_7, \mathcal{D}_5 + \mathcal{D}_6\}$, $\mathcal{G}_6 = \{\mathcal{D}_1 + \mathcal{D}_3, \mathcal{D}_4 + \mathcal{D}_6, \mathcal{D}_5 + \mathcal{D}_7\}$ and $\mathcal{G}_7 = \{\mathcal{D}_2 + \mathcal{D}_3, \mathcal{D}_4 + \mathcal{D}_5, \mathcal{D}_6 + \mathcal{D}_7\}$.

Now, let us count how many vector sets $\mathbf{V} = [\mathbf{v}_{j_1} \mathbf{v}_{j_2}]$ hit forbidden multisets. We can easily count after classifying $\mathcal{B}_8^+$ into groups as discussed above. Hence, the total number of vector sets that hit forbidden matrices are

$$\binom{8}{1} \times 7 + \binom{4}{2} \times 7 + \binom{7}{2} \times 8 + \binom{4}{2} \times 7 \\ = 56 + 42 + 168 + 42 = 308.$$

There are no other different combinations of two vectors in $\mathcal{B}_8^+$ that can hit the forbidden multisets. We find that our computed number of combination that hits the forbidden multiset $308 < 7140$ is less than the total number of two vectors combination sets, then we can claim that the maximum number of vectors that can be added to $\mathbf{H}_8$ is $(K_{max}^a - L) \geq 2$. This method of classifying $\mathcal{B}_8^+$ into groups not only helps us to prove the maximum number of vectors but also on how to construct such vector sets that possess unique decodability property (4).

Similar computation can be carried out for the cases $k = 3, 4, 5$ and still claim that the number of computation is less than the total number of $k = 3, 4, 5$ vectors combination sets. As an example, we present two of such combinations below for the case of $k = 5$,

$$\mathbf{V}_1 = \begin{bmatrix} -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$\mathbf{V}_2 = \begin{bmatrix} -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$\mathbf{V}_1 = \begin{matrix} E_p = \{1, 2, 3, 4, 5, 6, 7\} & E_n = \{0\} \\ E_p = \{0, 2, 3, 4, 5, 6, 7\} & E_n = \{1\} \\ E_p = \{0, 1, 3, 4, 5, 6, 7\} & E_n = \{2\} \\ E_p = \{1, 2, 3, 4, 5, 6, 7\} & E_n = \{0, 4\} \\ E_p = \{0, 5, 6, 7\} & E_n = \{1, 2, 3, 4\} \\ E_p = \{1, 2, 3, 4, 5, 6, 7\} & E_n = \{0\} \\ E_p = \{0, 2, 3, 4, 5, 6, 7\} & E_n = \{1\} \end{matrix},$$

$$\mathbf{V}_2 = \begin{matrix} E_p = \{0, 1, 3, 4, 5, 6, 7\} & E_n = \{2\} \\ E_p = \{0, 1, 3, 4, 6, 7\} & E_n = \{2, 5\} \\ E_p = \{2, 5, 6, 7\} & E_n = \{0, 1, 3, 4\}. \end{matrix}$$

In both construction examples $\mathbf{V}_1$ and $\mathbf{V}_2$, we cannot find any 2 vector combinations that belong to the same $\mathcal{A}_i$'s, $\mathcal{D}_i$'s, $\mathcal{F}_i$'s and $\mathcal{G}_i$'s. Therefore, all possible combinations of 2 vectors do not hit any of forbidden multisets.

However, once we add any other vector $\mathbf{v}_6 \in \mathcal{B}_8^+$ to the above sets some of the combinations of resulting vector

sets hit one of the forbidden multisets. This means that if $k = 6$ we compute the number of combinations that hit the forbidden multisets is exactly equal to $\binom{2^{8-1}-8}{6} = \binom{120}{6} = 3, 652, 745, 460$. Hence, the maximum number of vectors $\mathbf{v}_i$ that can be appended to $\mathbf{H}_8$ is $f_2(8) = 5$. The proof is complete. Next, we also show the different groups of such combinations.

Among all possible constructions, here we show that we can only have these group combinations $(\mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{G})$, $(\mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{D}, \mathcal{D})$, $(\mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{D}, \mathcal{G})$, $(\mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{G}, \mathcal{G})$, $(\mathcal{A}, \mathcal{D}, \mathcal{D}, \mathcal{D}, \mathcal{D})$, $(\mathcal{A}, \mathcal{D}, \mathcal{D}, \mathcal{D}, \mathcal{G})$, $(\mathcal{A}, \mathcal{D}, \mathcal{D}, \mathcal{G}, \mathcal{G})$ and $(\mathcal{A}, \mathcal{D}, \mathcal{G}, \mathcal{G}, \mathcal{G})$. We take each combinations and using the rules developed in [31], we show in Appendix B that we cannot add any more columns from any groups.

IV. MINIMUM DISTANCE OF CODE SETS

The Manhattan distance [60] equivalently (ℓ_1) -norm of two L -dimensional vectors $\mathbf{y}_i$ and $\mathbf{y}_j$ for $i \neq j$ is defined as

$$d_L(\mathbf{y}_i, \mathbf{y}_j) = \sum_{t=1}^L |y_{i,t} - y_{j,t}|, \quad (13)$$

where $|\cdot|$ denotes complex amplitude. Then the general minimum Manhattan distance of received vectors for a given antipodal code set can be formulated by

$$d_{min}(\mathbf{C}) = \underset{\substack{\mathbf{x}_i, \mathbf{x}_j \in \{\pm 1\}^{K \times 1} \\ \mathbf{y}_i = \mathbf{C}\mathbf{x}_i, \mathbf{y}_j = \mathbf{C}\mathbf{x}_j}}{\operatorname{argmin}} d_L(\mathbf{y}_i, \mathbf{y}_j). \quad (14)$$

Theorem 1: Let $\mathcal{C} \in \{\pm 1\}^{L \times K}$ represent the set of all antipodal matrices constructed by distinctⁱⁱ columns, then the minimum distance of the code set, $\delta(\mathcal{C})$, is equal to 4, where

$$\delta(\mathcal{C}) = \underset{\mathbf{C}' \in \mathcal{C}}{\operatorname{argmin}} d_{min}(\mathbf{C}'). \quad (15)$$

Proof: Let assume that $\delta(\mathcal{C}) = d_{min}(\mathbf{C}) = d_L(\mathbf{y}_n, \mathbf{y}_m)$, where $\mathbf{y}_n = \mathbf{C}\mathbf{x}_n$, $\mathbf{y}_m = \mathbf{C}\mathbf{x}_m$, $\mathbf{y}_n, \mathbf{y}_m \in \mathcal{N}^{L \times 1}$, $\mathcal{N} \in \{\pm K, \pm(K-2), \dots\}$, and $\mathbf{x}_n, \mathbf{x}_m \in \{\pm 1\}^{K \times 1}$. The minimum value obtained when the difference vector $\mathbf{y} = \mathbf{y}_n - \mathbf{y}_m = \mathbf{C}(\mathbf{x}_n - \mathbf{x}_m) = \mathbf{C}\bar{\mathbf{x}}$ has only one non-zero element $y_c \neq 0$, $y_{n,c} \neq y_{m,c}$, and $L-1$ zeros $y_i = 0$, $y_{n,i} = y_{m,i}$ for $i \neq c$ where $\bar{\mathbf{x}} \in \{0, \pm 2\}^{K \times 1}$. In order to achieve only one non-zero element of $\mathbf{y}$ then $\mathbf{c}^i \bar{\mathbf{x}} = 0$ for $\forall i \in \{1, \dots, L\}$ except $i \neq c$ where $\mathbf{c}^i$ is the i -th row of matrix $\mathbf{C}$. One obvious remark is that the number of ± 2 elements in $\bar{\mathbf{x}}$ must be even for the inner product, $\mathbf{c}^i \bar{\mathbf{x}}$, to be zero. One possible option is to have all zeros but two non-zero elements in $\bar{\mathbf{x}}$. In this case the matrix $\mathbf{C}$ must have at least two columns $\mathbf{c}_j$ and $\mathbf{c}_k$ with $c_{j,i} = c_{k,i}$ or $c_{j,i} = -c_{k,i}$ for $\forall i \in \{1, \dots, L\}$ except $i \neq c$ where $1 \leq j \neq k \leq K$ to result in a vector $\mathbf{y}$ having only one non-zero element. In other words, the difference of $\mathbf{c}_j$ and $\mathbf{c}_k$ is either in one or $L-1$ elements. Therefore, $y_{n,c} - y_{m,c} = \pm 2(c_{j,c} - c_{k,c})$ where $c_{j,c} - c_{k,c}$ results in $d_{min}(\mathbf{C}) = |y_{n,c} - y_{m,c}| = 4$. $\square$

ⁱⁱNot only the columns required to be distinct but we assume any column multiplication with minus one should result distinct columns as well.

Now that we proved that $\delta(\mathcal{C}) = 4$, we will try to find $d_{\min}(\mathbf{C})$ of our proposed UD code sets $\mathbf{C} \in \mathcal{C} \subset \mathcal{E}$, where $\mathcal{C} \in \{\pm 1\}^{L \times K}$ is the set of all the antipodal UD code sets. Let us follow the option of having all zeros but two non-zero elements in $\bar{\mathbf{x}}$. In the case of $L = 4$ the columns $\mathbf{c}_1$ and $\mathbf{c}_5$ differ in the first element only. If $\bar{\mathbf{x}} = [2, 0, 0, 0, -2]^T$ then the difference vector is $\mathbf{y} = [4, 0, 0, 0]^T$. Note that even if we substitute the $\mathbf{c}_1$ by $-\mathbf{c}_1$ we can still obtain same $\mathbf{y}$ with $\bar{\mathbf{x}} = [2, 0, 0, 0, 2]^T$. Based on our construction in [31], we look at the case of $L = 8$. Observe that all the elements of the columns 9-th and 12-th, $[\alpha^{13}, 0]^T$ and $[\alpha^{13}, \alpha^{13}]^T$ of the $\mathbf{C}$ are equal except the 5-th element in which they differ. If we select $x_{n,9} \neq x_{m,9}$, $x_{n,12} \neq x_{m,12}$, and $x_{n,i} = x_{m,i}$ for all $i \notin \{9, 12\}$ then $y_{n,5} = 2$ and $y_{m,5} = -2$ or $y_{n,5} = -2$ and $y_{m,5} = 2$, which will result in $d_L(\mathbf{y}_n, \mathbf{y}_m) = 4$. With this specific observation together with the Theorem 1, we conclude that $d_{\min}(\mathbf{C}) = 4$. From this observation, we learn that if any two columns differ at one element in a UD code set we assure that $d_{\min}(\mathbf{C}) = 4$. Similarly, for $L = 16$ columns 17-th and 27-th, $[\alpha^{13}, 0, \alpha^{13}, 0]^T$ and $[\alpha^{13}, 0, 0, 0]^T$ differ in one element only. Due to our recursive construction in [31] for $L = 2^p$, where $p = 5, 6, \dots$ columns $p2^{(p-2)}+3$ and $(p-1)2^{(p-1)}+3$ differ in one element only. Therefore, all the UD code set generated in [31] has $d_{\min}(\mathbf{C}) = 4$.

V. FAST DECODING ALGORITHM IN AWGN

The recursive linear NDA decoder, discussed in [31], is not suitable for the noisy transmission channel. Let the received vector in the presence of noise be mathematically formulated as

$$\mathbf{y} = A\mathbf{C}\mathbf{x} + \mathbf{n} \quad (16)$$

$$= \sum_{k=1}^K A\mathbf{c}_k x_k + \mathbf{n}, \quad (17)$$

where A denotes the amplitude, $\mathbf{c}_k \in \{\pm 1\}^{L \times 1}$ are signatures for $1 \leq k \leq K$, $\mathbf{x} \in \{\pm 1\}^{K \times 1}$ is user data and $\mathbf{n}$ is the Additive white Gaussian noise (AWGN) channel noise vector with a variance of σ^2 .

The objective of the receiver is as follows; recover the user data $\hat{\mathbf{x}}$ given the received vector $\mathbf{y}$ in (16) and $\mathbf{C}$, so that the mean square error $\mathbb{E}\{\|\mathbf{x} - \hat{\mathbf{x}}\|^2\}$ is minimized. The ML solution is given by

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \in \{\pm 1\}^{K \times 1}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{C}\mathbf{x}\|^2. \quad (18)$$

It is widely recognized that obtaining the ML solution is generally NP-hard [61].

Our detection problem, where the overloaded signature matrix has UD structure, can be solved efficiently if there is a function that maps $\mathbf{y} \mapsto \hat{\mathbf{y}} \in \Lambda \subset \mathcal{N}^{L \times 1}$, where Λ is a $\mathbb{Z}$ -module with rank L . Therefore, it is equivalent to finding the closest vector point in a lattice Λ ,

such that

$$\hat{\mathbf{y}} = \underset{\mathbf{y}' \in \mathcal{N}^{L \times 1}}{\operatorname{argmin}} d_L(\mathbf{y}, \mathbf{y}'). \quad (19)$$

Gaining the knowledge of $\hat{\mathbf{y}}$, one of the points in Λ generated by $\mathbf{C}$, we can obtain $\hat{\mathbf{x}}$ unambiguously (uniquely) by applying the NDA [31], since $\mathbf{C}$ satisfies the unique decodability criteria. However, there is no known polynomial algorithm to solve for $\hat{\mathbf{y}}$ from a given $\mathbf{y}$.

Without loss of generality, we design our low-complexity decoding algorithm for code sets that are generated by the seed matrix $\mathbf{V}_8$ which is given by:

$$\mathbf{V}_8 = \begin{bmatrix} \alpha^{14} & 1 & \alpha & \alpha^3 & \alpha^3 \\ 0 & 0 & 0 & \alpha^3 & \alpha^6 \end{bmatrix}. \quad (20)$$

However, we do not necessarily imply that our proposed decoder cannot be applied to other recursive UD code sets such as in [14], [21], [23]. A slight modification may be required depending on a given $\mathbf{C}$ matrix. We present our proposed low-complexity FDA for $\mathbf{C}_L$, $L \in 2^i$, where $i \in \{2, 3, \dots\}$, which is portrayed in Table 7. To quickly summarize, the FDA estimates iteratively the number and positions of -1 s in $\hat{\mathbf{x}}$ from the received values in $\mathbf{y}$. Those estimates are updated when the FDA compares each value of the vector $\mathbf{y}$ against quantized levels. Those quantized levels are computed based on the information from the previous rows and the current values of the vector $\mathbf{y}$.

Therefore, the FDA attempts to find the closest lattice point that is generated by matrix $\mathbf{C}$ from the estimate $\hat{\mathbf{x}}$ to the received vector $\mathbf{y} \in \mathbb{R}^{L \times 1}$. This is achieved by performing quantization on each row of vector $\mathbf{y}$ to obtain $\mathbf{z} \in \mathcal{N}^{L \times 1}$ such that $\mathbf{z}$ is a valid lattice point. In order to demonstrate how the FDA works, it is beneficial to describe each function in detail. The quantizer $\mathbf{Q} : \mathbb{R} \mapsto \mathcal{N}$, $z_1 = \mathbf{Q}(y, -K, K, 2)$ maps a received real value $y \in \mathbb{R}$ to one of the constellation in $\{\pm K, \pm(K-2), \dots\}$ as follows,

$$\mathbf{Q}(y, t^-, t^+, s) = \begin{cases} t^- & \text{if } y \leq t^- + s/2 \\ t^- + s(i' - 1) & \text{if } t^- + s(i' - 3/2) < y \dots \\ \dots \leq t^- + s(i' - 1/2) \\ t^+ & \text{if } y > t^+ - s/2 \end{cases} \quad (21)$$

where t^- , t^+ and s are the input parameters for the minimum, maximum, step-size values and $i' \in \mathbb{N}$ is the internal value that decides the quantization level, respectively. Furthermore, let integer n and vector $\mathbf{m}$ denote the number of -1 s and locations, where the -1 s occur in $\hat{\mathbf{x}}$, respectively. Mathematically, n and $\mathbf{m}$ can be expressed as

$$n = (K - \hat{\mathbf{x}}^T \mathbf{1})/2, \quad (22)$$

and

$$\mathbf{m} = (\mathbf{1} - \hat{\mathbf{x}})/2. \quad (23)$$

$$\text{iii } \mathcal{A} \setminus \mathcal{B} = \{x : x \in \mathcal{A} \text{ AND } x \notin \mathcal{B}\}.$$

Merge, meP function

Input: $dP', \mathbf{m}, n, K, r_c, m_{LR}, mP$

```

1:  $dPart \leftarrow \{\emptyset\}, \mathcal{B} \leftarrow mP(r_c), p' \leftarrow 1$ 
2: for  $p \leftarrow 1$  to  $len(dP')$ 
3:    $A_L \leftarrow dP'(p, 3), f \leftarrow true$ 
4:   if  $A_L \neq 0$ 
5:      $dPart(p', 1:2) \leftarrow [0, A_L], \mathcal{L}_0 \leftarrow dP'(p, 5:4+A_L)$ 
6:      $\mathcal{L}_1 \leftarrow \mathcal{L}_0 \cap \mathcal{B}(1), \mathcal{R}_1 \leftarrow \mathcal{L}_0 \cap \mathcal{B}(2)$ 
7:      $dPart(p', 3:4+A_L) \leftarrow [|\mathcal{L}_1|, |\mathcal{R}_1|, \mathcal{L}_1, \mathcal{R}_1]$ 
8:      $r \leftarrow r_c, f \leftarrow false$ 
9:     if  $\mathcal{L}_0 = mP(r-1, 1)$ 
10:       $dPart(p', 1) \leftarrow m_{LR}(r-1, 3), f \leftarrow true$ 
11:     else
12:       while  $f = false$  AND  $r > 1$ 
13:          $\mathcal{I}_L \leftarrow \{i | \mathbf{m}(i) = -1, i \in \mathcal{L}_0\}$ 
14:         if  $\mathcal{I}_L = \emptyset$ 
15:            $dPart(p', 1) \leftarrow \sum_{i \in \mathcal{L}_0} \mathbf{m}(i), f \leftarrow true$ 
16:         else
17:            $\mathcal{L}'_0 \leftarrow [1:K] \setminus \mathcal{L}_0$ 
18:            $\mathcal{I}_{L'} \leftarrow \{i | \mathbf{m}(i) = -1, i \in \mathcal{L}'_0\}$ 
19:           if  $\mathcal{I}_{L'} = \emptyset$ 
20:              $dPart(p', 1) \leftarrow n - \sum_{i \in \mathcal{L}'_0} \mathbf{m}(i), f \leftarrow true$ 
21:           else
22:              $r \leftarrow r - 1$ 
23:              $\mathbf{m} \leftarrow uMissing(m_{LR}(r, 3), mP(r, 1), \mathbf{m})$ 
24:           if  $f = false$ 
25:             break
26:            $p' \leftarrow p' + 1$ 
27:          $A_R \leftarrow dP'(p, 4)$ 
28:         if  $A_R \neq 0$  AND  $f = true$ 
29:            $dPart(p', 1:2) \leftarrow [0, A_R],$ 
30:            $\mathcal{L}_0 \leftarrow dP'(p, 5+A_L:4+A_L+A_R)$ 
31:            $\mathcal{L}_2 \leftarrow \mathcal{L}_0 \cap \mathcal{B}(1), \mathcal{R}_2 \leftarrow \mathcal{L}_0 \cap \mathcal{B}(2)$ 
32:            $dPart(p', 3:4+A_R) \leftarrow [|\mathcal{L}_2|, |\mathcal{R}_2|, \mathcal{L}_2, \mathcal{R}_2]$ 
33:            $r \leftarrow r_c, f \leftarrow false$ 
34:           if  $\mathcal{L}_0 = mP(r-1, 2)$ 
35:              $dPart(p', 1) \leftarrow m_{LR}(r-1, 4), f \leftarrow true$ 
36:           else
37:             while  $f = false$  AND  $r > 1$ 
38:                $\mathcal{I}_R \leftarrow \{i | \mathbf{m}(i) = -1, i \in \mathcal{L}_0\}$ 
39:               if  $\mathcal{I}_R = \emptyset$ 
40:                  $dPart(p', 1) \leftarrow \sum_{i \in \mathcal{L}_0} \mathbf{m}(i), f \leftarrow true$ 
41:               else
42:                  $\mathcal{L}'_0 \leftarrow [1:K] \setminus \mathcal{L}_0$ 
43:                  $\mathcal{I}_{L'} \leftarrow \{i | \mathbf{m}(i) = -1, i \in \mathcal{L}'_0\}$ 
44:                 if  $\mathcal{I}_{L'} = \emptyset$ 
45:                    $dPart(p', 1) \leftarrow n - \sum_{i \in \mathcal{L}'_0} \mathbf{m}(i), f \leftarrow true$ 
46:                 else
47:                    $r \leftarrow r - 1$ 
48:                    $\mathbf{m} \leftarrow uMissing(m_{LR}(r, 4), mP(r, 2), \mathbf{m})$ 
49:                 if  $f = false$ 
50:                   break
51:                  $p' \leftarrow p' + 1$ 
52:             if  $f = false$ 
53:                $tP \leftarrow [m'_{LR}, K, m_{LR}(r_c-1, 1), m_{LR}(r_c-1, 2), mP(r_c-1)]$ 
54:                $[dPart, \mathbf{m}] \leftarrow meP(tP, \mathbf{m}, n, K, r_c, m_{LR}, mP)$ 

```

Output: $dPart, \mathbf{m}$

For the trivial case when $z_1 = K$ or $z_1 = -K$ the algorithm outputs the decision vector $\hat{\mathbf{x}}$ without proceeding to the next steps. Otherwise, it will initialize the vector $\mathbf{m} \leftarrow \mathbf{1}_K$, index $r_c \leftarrow 1$ and $n \leftarrow (K - z_1)/2$. Then the value of n is recorded in the table m_{LR} at the row $r_c = 1$ and the column n_L . The table m_{LR} keeps track of l_L, l_R, n_L and n_R values for each row and l_L and l_R are defined as the number of +1s and -1s of the row of code set $\mathbf{C}$, n_L and n_R are the number of -1s of the estimated $\hat{\mathbf{x}}$ that corresponds to locations of +1 and -1 of each row of the code set.

$^{iv}i_d$ is the lowest index that $\mathbf{t}_d(i_d) \neq 0$.

TABLE 7. Fast decoder algorithm (FDA).

FDA			
Input: $\mathbf{y}, K$			
1:	$z_1 \leftarrow Q(y_1, -K, K, 2)$		
2:	if $ z_1 = K, \hat{\mathbf{x}} \leftarrow \text{sgn}(z_1)\mathbf{1}_K$		
3:	else		
4:	$\mathbf{m} \leftarrow -\mathbf{1}_K, r_c \leftarrow 1, n \leftarrow (K - z_1)/2$		
5:	$m_{LR}(r_c, 3) \leftarrow n$		
6:	$dP(r_c) \leftarrow [n, K, m_{LR}(r_c, 1), m_{LR}(r_c, 2), mP(r_c)]$		
7:	$c_{AL} \leftarrow \mathbf{0}, \mathbf{z} \leftarrow \mathbf{0}, s_I \leftarrow 1, c_T \leftarrow 1$		
8:	while $s_I = 1$ AND $c_T < N_c$		
9:	$s_I \leftarrow 0$		
10:	while $r_c < L, r_c \leftarrow r_c + 1$		
11:	$[dP(r_c), \mathbf{m}] \leftarrow meP(dP(r_c-1), \mathbf{m}, n, K, r_c, m_{LR}, mP)$		
12:	$A^- \leftarrow \min T(dP(r_c)), A^+ \leftarrow \max T(dP(r_c))$		
13:	$\mathbf{z}(r_c) \leftarrow Q(y', A^-, A^+, 4)$		
14:	$m_{LR}(r_c, 3) \leftarrow (2n - \mathbf{z}(r_c) -$		
14:	$(m_{LR}(r_c, 2) - m_{LR}(r_c, 1)))/4$		
15:	$m_{LR}(r_c, 4) \leftarrow n - m_{LR}(r_c, 3)$		
16:	$\mathbf{m} \leftarrow uM(\mathbf{m}, m_{LR}, r_c, mP)$		
17:	$\mathbf{m} \leftarrow f_c(\mathbf{m}, m_{LR}, n), \mathbf{t}_d \leftarrow \mathbf{z} - \mathbf{C}(-2\mathbf{m} + \mathbf{1})$		
18:	if $\mathbf{t}_d \notin \mathbf{0}, s_I \leftarrow 1, r_c \leftarrow i_d^{iv}$		
19:	$c_{AL}(r_c + 1) \leftarrow c_{AL}(r_c + 1) + 1$		
20:	$c_T \leftarrow c_T + 1$		
21:	$\hat{\mathbf{x}} \leftarrow -2\mathbf{m} + \mathbf{1}$		
Output: $\hat{\mathbf{x}}$			

For example, m_{LR} table for the code set in Fig. 6 can be constructed as

l_L	l_R	n_L	n_R
5	0	n	.
2	3	.	.
3	2	.	.
3	2	.	.

where the last two columns presented as dots are filled up in the further steps of the algorithm. Whereas $mP(r_c)$ is the actual column indices of +1s and -1s in the row r_c . In the case of code set in Fig. 6, the matrix mP is defined as

r_c	$index_L$	$index_R$
1	$\{\emptyset\}$	$\{1, 2, 3, 4, 5\}$
2	$\{2, 4, 5\}$	$\{1, 3\}$
3	$\{3, 4\}$	$\{1, 2, 5\}$
4	$\{2, 3\}$	$\{1, 4, 5\}$

The algorithm proceeds by partitioning each row r_c and saving the estimated number of -1s of vector $\hat{\mathbf{x}}$, n' , the partition size, K' , the number of +1s and -1s, $l_{L'}$ and $l_{R'}$ of the specific partition in $dP(r_c)$. In Step 7, the adaptive parameter c_{AL} , stopping iteration flag s_I , and the number of repetition count c_T (e.g., algorithm can repeat steps from 10 to 16 up to a maximum of N_c times, since it directly depends on the variance of the noise) are initialized. At each row r_c , $dP(r_c)$ gets updated by calling $meP()$ function. The function $meP(dP(r_c-1), \mathbf{m}, n, K, r_c, m_{LR}, mP)$ scans each partition of the row r_c with updated values and if it finds one or more partitions completely identified the exact locations of -1s, hence will skip partitioning further. A^- and A^+ are the minimum and maximum values calculated for a given partitions at each row.

MinT function**Input:** dP

- 1: $A^- \leftarrow 0$
- 2: **for** $i \leftarrow 1$ to $\text{len}(dP)$
- 3: $A^- \leftarrow A^- + \text{minF}(dP(i, 1), dP(i, 3), dP(i, 2))$

Output: A^- **MaxT function****Input:** dP

- 1: $A^+ \leftarrow 0$
- 2: **for** $i \leftarrow 1$ to $\text{len}(dP)$
- 3: $A^+ \leftarrow A^+ + \text{maxF}(dP(i, 1), dP(i, 4), dP(i, 2))$

Output: A^+

where $\text{minF}(n, L, K) = 2|n - L| - K$ and $\text{maxF}(n, R, K) = -2|n - R| + K$. In line 13 of FDA, we define $y' = y(r_c) + 2\text{sgn}(y(r_c) - z(r_c))c_{AL}(r_c)$, where c_{AL} is an integer vector that is incremented in Step 19 by one if estimated $\mathbf{z}$ is not one of the lattice vertices generated by $\mathbf{C}$. In line 18, we scan the rows from 1 to L to find the first r_c , where $\mathbf{z}$ differ from estimated lattice vertex. The function $\text{uM}(\mathbf{m}, m_{LR}, r_c, mP)$ updates $\mathbf{m}$ with the given updated parameters as follows

uM function**Input:** $\mathbf{m}, m_{LR}, r, mP$

- 1: **if** $m_{LR}(r, 3) = 0$
- 2: $\mathbf{m}(mP(r, 1)) \leftarrow 0$
- 3: **elseif** $m_{LR}(r, 3) = m_{LR}(r, 1)$
- 4: $\mathbf{m}(mP(r, 1)) \leftarrow 1$
- 5: **else**
- 6: $\mathbf{m} \leftarrow \text{uMissing}(m_{LR}(r, 3), mP(r, 1), \mathbf{m})$
- 7: **if** $m_{LR}(r, 4) = 0$
- 8: $\mathbf{m}(mP(r, 2)) \leftarrow 0$
- 9: **elseif** $m_{LR}(r, 4) = m_{LR}(r, 2)$
- 10: $\mathbf{m}(mP(r, 2)) \leftarrow 1$
- 11: **else**
- 12: $\mathbf{m} \leftarrow \text{uMissing}(m_{LR}(r, 4), mP(r, 2), \mathbf{m})$

Output: $\mathbf{m}$

where the function uMissing in lines 6 and 12 of uM are defined as

uMissing function**Input:** $n', \mathcal{A}, \mathbf{m}$

- 1: $\mathcal{I}_L \leftarrow \{i | \mathbf{m}(i) = -1, i \in \mathcal{A}\}$
- 2: **if** $|\mathcal{I}_L| = 1$
- 3: $\mathbf{m}(\mathcal{I}_L) \leftarrow n' - \sum_{i \neq \mathcal{I}_L} \mathbf{m}(i)$

Output: $\mathbf{m}$

In line 17 of the FDA, the function $\mathbf{f}_c(\mathbf{m}, m_{LR}, n)$ finds all the locations of -1 s in $\mathbf{m}$ based on the UD structure of $\mathbf{C}$ codes. As an example, in case of code set presented in Fig. 6, the function $\mathbf{f}_c$ is expressed as

where $\mathcal{A}_5 = \{2, 3, 4\}$. In case the information in $\mathbf{m}$ does not correspond to $\mathbf{z}$, which is verified in line 18 of FDA it then sets r_c to the row, where the discrepancy happened and re-runs from Step 10 again until it finds $\mathbf{m}$ that correspondence to $\mathbf{z}$.

For the case of Rayleigh fading channel instead of AWGN in (16) the proposed FDA is still applicable to perform detection. However, the channel coefficients for each user k should

 $\mathbf{f}_c$ function, $L = 4$ **Input:** $\mathbf{m}, m_{LR}, n$

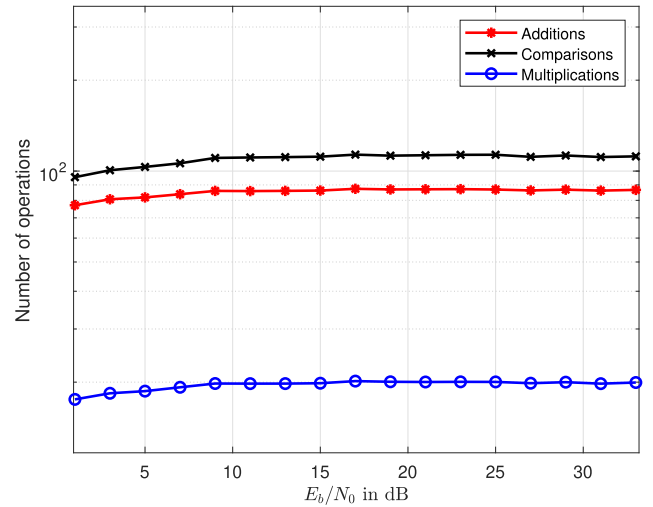
- 1: $n' \leftarrow \sum_{i \in \mathcal{A}_5} m_{LR}(i, 4)$
- 2: $\mathbf{m}(5) \leftarrow \text{mod}(n', 2)$
- 3: $m' \leftarrow (n' - \mathbf{m}(5))/2$
- 4: $\mathbf{m}(1) \leftarrow n - m' - \mathbf{m}(5)$
- 5: $\mathbf{m}(2) \leftarrow m_{LR}(3, 3) - \mathbf{m}(1) - \mathbf{m}(5)$
- 6: $\mathbf{m}(3) \leftarrow m_{LR}(2, 3) - \mathbf{m}(1)$
- 7: $\mathbf{m}(4) \leftarrow m_{LR}(2, 4) - \mathbf{m}(2) - \mathbf{m}(5)$

Output: $\mathbf{m}$

be known at the receiver side. For the frequency-selective fading channels, we can employ transmitter precoding scheme to overcome multipath channel effect as proposed by Fantuz and D'Amours, which is detailed in [62]. Briefly, this transmit precoding scheme exploits the knowledge of the channel impulse response for transforming the multipath channel into a single-path non-dispersive channel, which is equivalent over non-dispersive Rayleigh fading channel model.

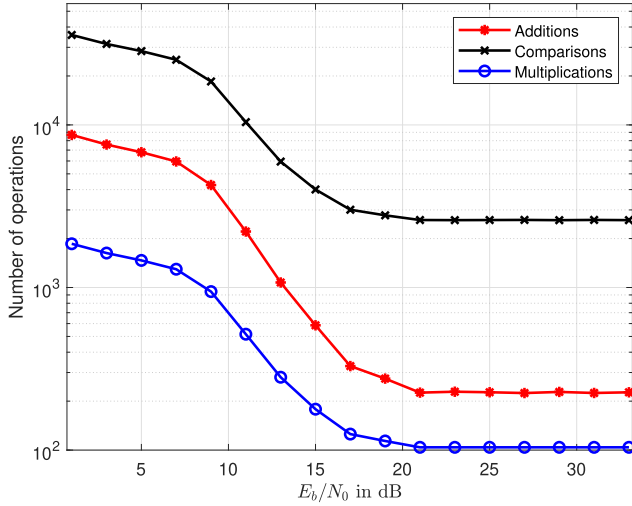
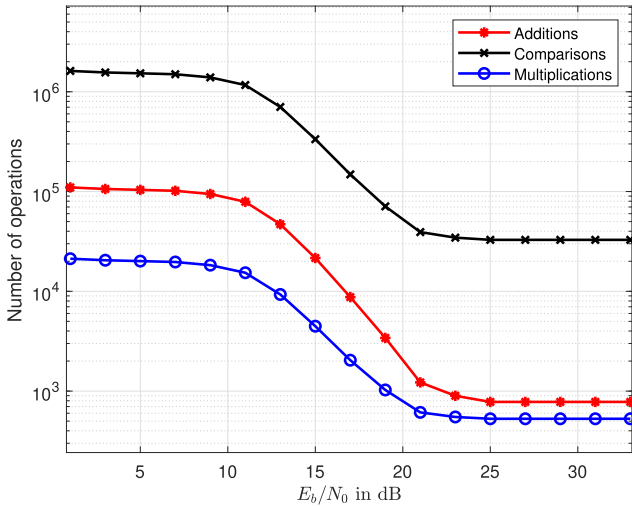
VI. COMPLEXITY ANALYSIS

In this section, we discuss the complexity analysis of the proposed NDA and FDA algorithms. The NDA decoder for the noiseless transmission channels, discussed in [31], deciphers the data of all users at the receiver side in a recursive manner. At each step, it performs additions, comparisons and multiplications to decipher the bits of the users.

**FIGURE 2.** Complexity for $L = 4$.

The NDA is deterministic with an exact number of execution steps. After $L/8$ number of execution steps it calls the NDA algorithm recursively, using two smaller vectors composed of the upper and lower $L/2$ elements of the received vector. For that reason to compute the complexity of the algorithm we first break the algorithm into two blocks B_1 and B_2 from steps 1 to 8 and 9 to 12.

We denote N_1 and N_2 to represent the total number of additions, comparisons and multiplications ($N_{add}^i, N_{comp}^i, N_{mult}^i$) of block B_i at each recursive calls. As an example

FIGURE 3. Complexity for $L = 8$.FIGURE 4. Complexity for $L = 16$.

$N_1 = (N_{add}^1, N_{comp}^1, N_{mult}^1)$, where $N_{add}^1 = 10$, $N_{comp}^1 = 3$ and $N_{mult}^1 = 0$ since there is 3, 3 and 4 additions in steps 2, 4 and 6, and 1 comparison in steps 3, 5 and 8, respectively. Similarly, when $L = 16$ the $N_2 = (N_{add}^2, N_{comp}^2, N_{mult}^2)$, where $N_{add}^2 = (3 + L)L/2 + 4 = 156$, $N_{comp}^2 = 1$ and $N_{mult}^2 = L^2/2 = 128$, respectively. For the trivial case when $L = 4$ we have $N_4 = (N_{add}^4, N_{comp}^4, N_{mult}^4)$, where $N_{add}^4 = 4$, $N_{comp}^4 = 4$ and $N_{mult}^4 = 4$.

The complexity when $L = 8$ is $N_1 + 2N_4$ since the NDA will not execute block B_2 . In the case when $L = 16$, the complexity of NDA is $4N_1 + N_2 + 4N_4$. It follows that when $L > 8$ the complexity of the recursive NDA can be represented as such

$$(i - 2)2^{(i-3)}N_1 + ((i - 4)2^{(i-3)} + 1)N_2 + 2^{(i-2)}N_4, \quad (24)$$

where $L = 2^i$ for $i \in \{4, 5, \dots\}$. Based on this calculation, we can conclude that the complexity of NDA algorithm is $\mathcal{O}(L \log_2(L))$.

TABLE 8. Complexity comparison of the detectors.

Algorithms	Complexity	Main procedures
NDA	$\mathcal{O}(L \log_2(L))$	Comparisons
FDA	$\mathcal{O}(LK \log_2(K))$	Comparisons
MMSE-PIC	$\mathcal{O}(LK^2)$	Inversion + multiplication
Slab-sphere	$\mathcal{O}(LK^2)$	Multiplication + comparisons
PDA	$\mathcal{O}(L^2K^2)$	Inversion + multiplication
ML	$\mathcal{O}(2^K)$	Multiplication + addition

As one would normally expect, the complexity of the decoder in noisy channels is much higher than in the noiseless channels. However, the complexity of the proposed FDA decoder in Section V is not more complex than the NDA in terms of big $\mathcal{O}$ notation. It is important to state that the proposed FDA requires neither matrix inversion nor decomposition, instead, only additions, comparisons and multiplications. The algorithm goes through each row of the received vector to decode one or more users. The best case scenario of our FDA would be to satisfy the condition in Step 2 with the complexity of maximum of K comparisons, $N_{comp} = K$. We should note that unlike the deterministic NDA, FDA has an element of probability in the execution steps. Therefore, to compute the overall complexity of the FDA we need to consider the worst case scenario. In Step 11 the complexity of meP can be shown to have $6K$ additions and $(1 + \log_2(K))6K + 2$ comparisons. The complexity of steps 12 and 13 is 6 additions and K comparisons. Similarly, the complexity of steps 14 and 15 is 4 additions. The complexity of uM method is $2K$ additions and $2K + 6$ comparisons. Those execution steps from 11 to 16 are repeated L times. Finally, in Step 17 the complexity is $(2 + K)L + \mathcal{O}(K)$ additions and LK multiplications. Considering all the components, the total complexity can be shown to be $N_{add} = 9LK + 12L + \mathcal{O}(LK)$, $N_{comp} = 6LK \log_2(K) + 9LK + 8L$ and $N_{mult} = LK$. Therefore, we can conclude by looking at the higher order terms of comparisons, N_{comp} , since it is higher than additions and multiplications, hence the overall complexity is $\mathcal{O}(LK \log_2(K))$. We note that in both NDA and FDA algorithms we do not consider assignments in our complexity computations. We can see that the complexity of FDA is comparably larger than NDA but much lower than the MMSE-PIC, Slab-sphere, PDA and ML decoders, which have complexities of $\mathcal{O}(LK^2)$, $\mathcal{O}(LK^2)$, $\mathcal{O}(L^2K^2)$, and $\mathcal{O}(2^K)$, respectively, as shown in Table 8. The exact computation for the cases $L = 4$, $L = 8$ and $L = 16$ are shown in Table 9. The FDA verifies in Step 18 if the L -dimensional lattice point that is generated by the estimated vector $\mathbf{m}$ is the same as the vector $\mathbf{z}$ or not. If that condition is not true the algorithm will adjust c_{AL} value by increasing it by one and starts repeating steps from 11 to 16. It is obvious that the FDA will make an exact number of execution steps if no noise vector, $\mathbf{n}$, is present and the number of times it will repeat

[illegible]

FIGURE 5. UD code set **C** with $L = 16$ and $K = 33$.

TABLE 9. Complexity of decoders.

Decoder	Complexity	(4×5)	(8×13)	(16×33)
FDA	Additions	248	1, 136	5, 472
	Comparisons	491	3, 309	20, 861
	Multiplications	20	104	528
PDA	Inver. + add.	400	10, 816	278, 784
ML	Mult. + add.	2^5	2^{13}	2^{33}

$$\mathbf{C}_{4 \times 5} = \begin{bmatrix} + & + & + & + & + \\ + & - & + & - & - \\ + & + & - & - & + \\ + & - & - & + & + \end{bmatrix}$$

FIGURE 6. UD code set **C** with $L = 4$ and $K = 5$.

$$\mathbf{C}_{8 \times 13} = \begin{bmatrix} + & + & + & + & + & + & + & + & + & + & + & + & + \\ + & + & - & - & + & - & - & - & - & - & + & - & + \\ + & - & - & + & + & - & - & - & - & - & - & + & + \\ + & + & + & + & - & - & - & + & + & + & + & + & - \\ + & + & - & - & - & + & - & + & + & + & + & - & + \\ + & - & - & + & - & + & - & + & + & + & + & + & - \end{bmatrix}$$

FIGURE 7. UD code set **C** with $L = 8$ and $K = 13$.

the steps from 11 to 16 will only depend on the variance of noise, σ^2 .

Therefore, we demonstrate the number of additions, comparisons and multiplications in Figs. 2, 3 and 4 by varying σ^2 in terms of E_b/N_0 in our simulations. For the case of $L = 4$ in Fig. 2, the number of additions, comparisons and multiplications does not depend on variance due to small overload factor $5/4$. However, for $L = 8$ and $L = 16$ it stays high for up to 10dB in E_b/N_0 . Then it drops to constant numbers at around 20dB in E_b/N_0 , which is shown in Figs. 3 and 4, respectively.

VII. SIMULATION RESULTS

In this section, we evaluate the performance of the proposed antipodal UD code sequences generated by the seed matrix (20), which are portrayed in Figs. 6, 7 and 5. In our simulations, we compare the FDA with the MMSE-PIC [40], slab-sphere [63], PDA [53] and ML detectors for binary phase-shift keying (BPSK) modulation, which are characterized in Figs. 8, 9 and 10.

The BER performance of UD code sets are averaged over the different users for $C_{4 \times 5}$, $C_{8 \times 13}$ and $C_{16 \times 33}$, respectively.

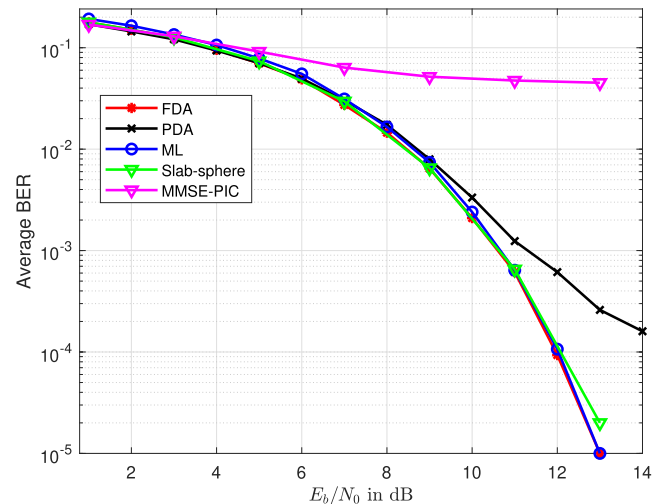
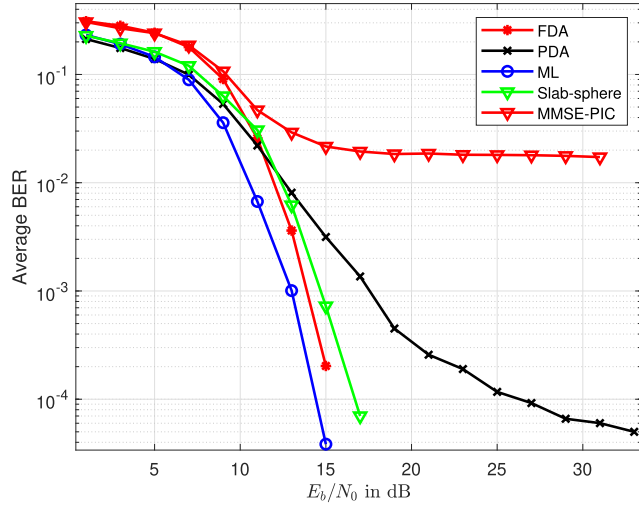
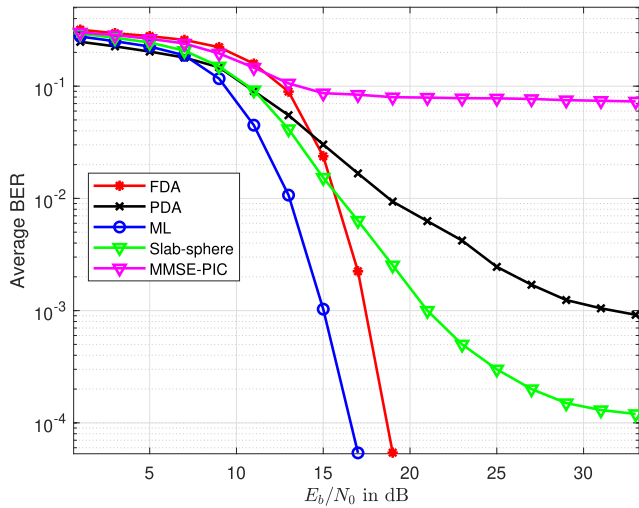


FIGURE 8. UD code set $C_{4 \times 5}$.

The performance of the proposed FDA is comparable to that of ML, as shown in Fig. 8.

For the larger values of L_s , the FDA has slightly inferior performance in terms of BER compared to ML. However, in practice the E_b/N_o at the BER of 10^{-3} is considered to be the operating threshold since we can apply channel encoding to achieve even as low as 10^{-6} BER. At the BER of 10^{-3} the FDA achieves 1 dB and 4 dB gain compared to the Slab-sphere, 4 dB and 15 dB gain compared the PDA, as shown in Figs. 9 and 10. The significant increase of the BER performance gap between the FDA and Slab-sphere, PDA detectors for greater values of L_s is due to error floor experienced by the Slab-sphere and the PDA detectors. The reason for this is since for larger values of L the overload factor increases exponentially. This in turn results in degradation of the linear separability of the overall system.

The linear separability criterion for linear detectors such as PDA have the property of vanishing BER as the channel noise goes to zero. Notice that we omitted the discussion of MMSE-PIC detector due to the overall poor BER performance of MMSE-PIC detector in comparison to other detectors. The Table 8 shows the computational complexity of all the detectors. Even though the performance of the proposed

FIGURE 9. UD code set $\mathbf{C}_{8 \times 13}$.FIGURE 10. UD code set $\mathbf{C}_{16 \times 33}$.

FDA is slightly worse than the ML detector, it has much lower complexity compared to ML.

VIII. CONCLUSION

In this paper, we introduced a novel fast (low-complexity) decoder algorithm (FDA) for antipodal uniquely decodable (UD) code sets. The proposed algorithm has a much lower computational complexity compared to the maximum likelihood (ML) decoder whose complexity may be prohibitive for even moderate code lengths. Simulation results show that the performance of the proposed decoder is almost as good as that of the ML decoder with only a 1 – 2 dB SNR degradation at a BER of 10^{-3} . Moreover, we proved the minimum Manhattan distance of UD codes proposed in [31] and a number of propositions which collectively served to be the foundation of the formal proof for the maximum number of users, K_{max}^a , for the case of $L = 8$.

In our future research, we will conceive multiuser detection for higher-order constellations for transmission over dispersive fading channels.

APPENDIX A

PROOF OF THE CONVERSION FROM $\mathbf{C}^a$ TO $\mathbf{C}^b$

The proof of the conversion from the antipodal overloaded UD code sets to binary UD code sets, coined as optical CDMA code sets in [27], are presented next.

Theorem 2: *If there is an antipodal UD code set $\mathbf{C}^a \in \{\pm 1\}^{L \times K}$, then there is an equivalent binary UD code set $\mathbf{C}^b \in \{0, 1\}^{L \times K}$.*

Proof: Suppose there is an antipodal UD code set $\mathbf{C}_{L \times K}^a$. By corollary, if multiplying each row or column by -1 we can assume that the entries of the first row of $\mathbf{C}_{L \times K}^a$ are all 1s. Let the conversion to the binary matrix $\mathbf{C}_{L \times K}^b = (\mathbf{C}_{L \times K}^a + \mathbf{J})/2$, where $\mathbf{J}$ is the $L \times K$ all-one matrix. It is clear that $\mathbf{C}_{L \times K}^b \in \{0, 1\}^{L \times K}$, therefore, we now need to prove the following $\text{Null}^v(\mathbf{C}_{L \times K}^b) \cap \{0, 1\}^{K \times 1} = \{0\}^{K \times 1}$ statement. Assume that $\mathbf{C}_{L \times K}^b \mathbf{z} = \mathbf{0}_L$, which yields to $(\mathbf{C}_{L \times K}^a + \mathbf{J})\mathbf{z} = \mathbf{0}_L$ and thus, $\mathbf{C}_{L \times K}^a \mathbf{z} = -\mathbf{J}\mathbf{z}$, where $\mathbf{z} \in \{0, 1\}^{K \times 1}$. Since the entries of the first row of $\mathbf{C}_{L \times K}^a$ as well as the matrix $\mathbf{J}$ are all 1s, the first entry of $\mathbf{C}_{L \times K}^a \mathbf{z}$ must be equal to the first entry of $-\mathbf{J}\mathbf{z}$. It is only possible if the first entry of $-\mathbf{J}\mathbf{z}$ is 0. Thus, $-\mathbf{J}\mathbf{z} = \mathbf{0}_L$ which leads to $\mathbf{z} = \mathbf{0}_L$. As a consequence of UD code set of $\mathbf{C}_{L \times K}^a$ that satisfies unique decodability condition (4) the expression $\mathbf{C}_{L \times K}^a \mathbf{z} = \mathbf{0}_L$ implies that $\mathbf{z} = \mathbf{0}_L$. $\square$

However, in general it is not necessarily true that any binary (optical) code sets $\mathbf{C}^b$ of any size can be converted to UD code sets $\mathbf{C}^a$. The reason is because for a given UD code set $\mathbf{C}^b$ with K_{max}^b , the following is true $K_{max}^b \geq K_{max}^a$ since $K_{max}^b = \gamma(L + 1)$ and $K_{max}^a = \gamma(L) + 1$ and obviously $\gamma(L + 1) \geq \gamma(L) + 1$. It is equality, $K_{max}^b = K_{max}^a$, if $L = 2^i$ where $i \in \{1, 2, \dots\}$. Therefore, for a given UD code set $\mathbf{C}^b$ with K_{max}^b , if $L \neq 2^i$ we have $K_{max}^b > K_{max}^a$ and there is no equivalent UD code set $\mathbf{C}^a$.

APPENDIX B

PROOF OF ALL POSSIBLE COMBINATIONS

Combination 1 ($\mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{G}$): This group combination is such that distinct $(\mathcal{A}, \mathcal{A}, \mathcal{A})$ produce $\binom{3}{2} = 3$ distinct $\mathcal{D}$'s different from existing $\mathcal{D}$, and $\mathcal{G}$'s by $\mathcal{A} + \mathcal{A} = \mathcal{D}/\mathcal{G}$ rule. Different $\mathcal{A}$ is also produced by $\mathcal{A} + \mathcal{D} = \mathcal{A}$ from existing ones. With this new $\mathcal{A}$ it can produce another distinct $\binom{3}{2} = 3$ $\mathcal{G}$'s. Total of six $\mathcal{G}$'s are produced with $(\mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{D})$ and existing $\mathcal{G}$ we have seven $\mathcal{G}$'s. This completes all seven different $\mathcal{G}$'s from each seven groups, hence we cannot add any other $\mathcal{G}$ to the existing combination. So three $\mathcal{A}$'s with $\mathcal{D}$ produce six distinct $\mathcal{A}$'s and the $\mathcal{A} + \mathcal{A} + \mathcal{A} = \mathcal{A}$, $\mathcal{A} + \mathcal{A} + \mathcal{A} + \mathcal{D} = \mathcal{A}$ produce another two distinct $\mathcal{A}$'s. Therefore, we cannot add any more $\mathcal{A}$ to the existing combinations. Now three $\mathcal{A}$'s will produce three distinct $\mathcal{D}$'s, and three $\mathcal{A}$'s with $\mathcal{D}$, $\mathcal{A} + \mathcal{D} = \mathcal{A}$, produce another three distinct $\mathcal{D}$'s, plus the $\mathcal{D}$ that is in the combination that makes a total of seven $\mathcal{D}$'s. Therefore, we cannot add any more $\mathcal{D}$'s to the combination. Therefore, we proved that we cannot add any more $\mathcal{A}$, $\mathcal{D}$ and $\mathcal{G}$ to the $(\mathcal{A}, \mathcal{A}, \mathcal{A}, \mathcal{D}, \mathcal{G})$.

^vNull() represents the nullspace of a matrix.

$\mathcal{D} + \mathcal{G} = \mathcal{D}$, $\mathcal{D} + \mathcal{G} + \mathcal{G} = \mathcal{D}$, $\mathcal{D} + \mathcal{G} + \mathcal{G} + \mathcal{G} = \mathcal{D}$ rules and with two existing $\mathcal{D}$'s that makes a total of seven $\mathcal{D}$'s, hence we cannot add any more $\mathcal{D}$ to the combination. Similarly, for the case of $\mathcal{G}$ the combination produces five distinct $\mathcal{G}$'s by $\mathcal{G} + \mathcal{G} = \mathcal{G}$, $\mathcal{G} + \mathcal{G} + \mathcal{G} = \mathcal{G}$ rules and with the existing $\mathcal{G}$ that makes a total of seven $\mathcal{G}$'s, hence we cannot add any more $\mathcal{G}$ to the combination. Therefore, we proved that we cannot add any more $\mathcal{A}$, $\mathcal{D}$ and $\mathcal{G}$ to the $(\mathcal{A}, \mathcal{D}, \mathcal{G}, \mathcal{G}, \mathcal{G})$.

REFERENCES

- [1] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*, C. Sammut and G. I. Webb, Eds. Redwood City, CA, USA: Addison-Wesley, 1995.
- [2] L. Dai, B. Wang, Z. Ding, Z. Wang, S. Chen, and L. Hanzo, "A survey of non-orthogonal multiple access for 5G," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2294–2323, 3rd Quart., 2018.
- [3] H. Sari, F. Vanhaverbeke, and M. Moeneclaey, "Extending the capacity of multiple access channels," *IEEE Commun. Mag.*, vol. 38, no. 1, pp. 74–82, Jan. 2000.
- [4] A. J. Grant and P. D. Alexander, "Random sequence multisets for synchronous code-division multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2832–2836, Nov. 1998.
- [5] S. Verdú and S. Shamai (Shitz), "Spectral efficiency of CDMA with random spreading," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 622–640, Feb. 1999.
- [6] F. Vanhaverbeke, M. Moeneclaey, and H. Sari, "DS/CDMA with two sets of orthogonal spreading sequences and iterative detection," *IEEE Commun. Lett.*, vol. 4, no. 9, pp. 289–291, Sep. 2000.
- [7] H. Sari, F. Vanhaverbeke, and M. Moeneclaey, "Multiple access using two sets of orthogonal signal waveforms," *IEEE Commun. Lett.*, vol. 4, no. 1, pp. 4–6, Jan. 2000.
- [8] F. Vanhaverbeke, M. Moeneclaey, and H. Sari, "Increasing CDMA capacity using multiple orthogonal spreading sequence sets and successive interference cancellation," in *Proc. IEEE Int. Conf. Commun.*, New York, NY, USA, vol. 3, Apr. 2002, pp. 1516–1520.
- [9] F. Vanhaverbeke and M. Moeneclaey, "An improved OCDMA/OCDMA scheme based on displaced orthogonal user sets," *IEEE Commun. Lett.*, vol. 8, no. 5, pp. 265–267, May 2004.
- [10] M. Li, S. N. Batalama, D. A. Pados, and J. D. Matyjas, "Minimum total-squared-correlation quaternary signature sets: New bounds and optimal designs," *IEEE Trans. Commun.*, vol. 57, no. 12, pp. 3662–3671, Dec. 2009.
- [11] R. Erdos and A. Rényi, "On two problems of information theory," *Publications Math. Inst. Hung. Acad. Sci.*, vol. 8, pp. 241–254, Aug. 1963.
- [12] S. Söderberg and H. S. Shapiro, "A combinatorial detection problem," *Amer. Math. Monthly*, vol. 70, no. 10, pp. 1066–1070, 1963.
- [13] D. G. Cantor, "Determining a set from the cardinalities of its intersections with other sets," *Can. J. Math.*, vol. 16, pp. 94–97, Feb. 1964.
- [14] B. Lindström, "On a combinatorial detection problem," *Publications Math. Inst. Hung. Acad. Sci.*, vol. 9, pp. 195–207, Sep. 1964.
- [15] B. Lindström, "On a combinatorial problem in number theory," *Can. Math. Bull.*, vol. 8, no. 4, pp. 477–490, Jun. 1965.
- [16] D. G. Cantor and W. H. Mills, "Determination of a subset from certain combinatorial properties," *Can. J. Math.*, vol. 18, pp. 42–48, Feb. 1966.
- [17] P. Erdos, "On a combinatorial problem in number theory," *Amer. Math. Monthly*, vol. 77, no. 6, pp. 619–621, Jul. 1970.
- [18] S.-C. Chang and E. Weldon, "Coding for T-user multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 25, no. 6, pp. 684–691, Nov. 1979.
- [19] T. Ferguson, "Generalized T-user codes for multiple-access channels (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 28, no. 5, pp. 775–778, Sep. 1982.
- [20] S.-C. Chang, "Further results on coding for T-user multiple-access channels (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 30, no. 2, pp. 411–415, Mar. 1984.
- [21] G. H. Khachatrian and S. S. Martirosian, "Codes for T-user noiseless adder channel," *Prob. Control Inf. Theory*, vol. 16, no. 3, pp. 187–192, 1987.
- [22] S. S. Martirosian and G. H. Khachatrian, "Construction of signature codes and the coin weighing problem," *Problemy Peredachi Inf.*, vol. 25, no. 4, pp. 96–97, Sep. 1989.
- [23] G. H. Khachatrian and S. S. Martirosian, "A new approach to the design of codes for synchronous-CDMA systems," *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 1503–1506, Sep. 1995.
- [24] G. H. Khachatrian and S. S. Martirosian, "Code construction for the T-user noiseless adder channel," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1953–1957, Sep. 1998.
- [25] Y.-W. Wu and S.-C. Chang, "Coding scheme for synchronous-CDMA systems," *IEEE Trans. Inf. Theory*, vol. 43, no. 3, pp. 1064–1067, May 1997.
- [26] W. H. Mow, "Recursive constructions of detecting matrices for multiuser coding: A unifying approach," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 93–98, Jan. 2009.
- [27] P. Pad, F. Marvasti, K. Alishahi, and S. Akbari, "Further results on coding for T-user multiple-access channels (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 55, no. 6, pp. 2705–2715, Jun. 2009.
- [28] M. Li and Q. Liu, "Fast code design for overloaded code-division multiplexing systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 447–452, Jan. 2016.
- [29] O. Mashayekhi and F. Marvasti, "Uniquely decodable codes with fast decoder for overloaded synchronous CDMA systems," *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3145–3149, Nov. 2012.
- [30] A. Singh, A. Amini, P. Singh, and F. Marvasti, "Set of uniquely decodable codes for overloaded synchronous CDMA," *IET Commun.*, vol. 10, no. 10, pp. 1236–1245, Jul. 2016.
- [31] M. Kulhandjian and D. A. Pados, "Uniquely decodable code-division via augmented Sylvester-Hadamard matrices," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Paris, France, Apr. 2012, pp. 359–363.
- [32] M. Kulhandjian, C. D'Amours, and H. Kulhandjian, "Uniquely decodable ternary codes for synchronous CDMA systems," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Bologna, Italy, Sep. 2018, pp. 1–6.
- [33] M. Kulhandjian, C. D'Amours, and H. Kulhandjian, "Multiway physical-layer network coding via uniquely decodable codes," *Wireless Commun. Mobile Comput.*, vol. 2018, Jun. 2018, Art. no. 2034870.
- [34] M. Kulhandjian, H. Kulhandjian, C. D'Amours, H. Yanikomeroglu, and G. Khachatrian, "Fast decoder for overloaded uniquely decodable synchronous optical CDMA," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Marrakech, Morocco, Apr. 2019, pp. 1–7.
- [35] M. Kulhandjian, H. Kulhandjian, and C. D'Amours, "Uniquely decodable ternary codes via augmented Sylvester-Hadamard matrices," in *Proc. IEEE Int. Black Sea Conf. Commun. Netw. (BlackSeaCom)*, May 2021, pp. 1–5.
- [36] S. Lu, W. Hou, J. Cheng, and H. Kamabe, "Multi-user UD K-ary codes recursively constructed from short-length multiary codes for multiple-access adder channel," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France, Jul. 2019, pp. 425–429.
- [37] Q.-Y. Yu, Y.-T. Li, W.-X. Meng, and W. Xiang, "Uniquely decodable codes for physical-layer network coding in wireless cooperative communications," *IEEE Syst. J.*, vol. 13, no. 4, pp. 3956–3967, Dec. 2019.
- [38] L. Dai, B. Wang, Y. Yuan, S. Han, I. Chih-Lin, and Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74–81, Sep. 2015.
- [39] R. Hoshyar, F. P. Wathan, and R. Tafazolli, "Novel low-density signature for synchronous CDMA systems over AWGN channel," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1616–1626, Apr. 2008.
- [40] M. Kulhandjian, H. Kulhandjian, C. D'Amours, and L. Hanzo, "Low-density spreading codes for NOMA systems and a Gaussian separability-based design," *IEEE Access*, vol. 9, pp. 33963–33993, 2021.
- [41] H. Nikopour and H. Baligh, "Sparse code multiple access," in *Proc. IEEE 24th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, London, U.K., Sep. 2013, pp. 332–336.
- [42] M. Kulhandjian and C. D'Amours, "Design of permutation-based sparse code multiple access system," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Montreal, QC, Canada, Oct. 2017, pp. 1–6.
- [43] J. van de Beek and B. M. Popovic, "Multiple access with low-density signatures," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Honolulu, HI, USA, Nov./Dec. 2009, pp. 1–6.
- [44] M. Kobayashi, J. Boutros, and G. Caire, "Successive interference cancellation with SISO decoding and EM channel estimation," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 8, pp. 1450–1460, Aug. 2001.
- [45] M. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 509–519, Apr. 1990.

- [46] G. Xue, J. Weng, T. Le-Ngoc, and S. Tahar, "Adaptive multistage parallel interference cancellation for CDMA," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 10, pp. 1815–1827, Oct. 1999.
- [47] D. Guo, L. K. Rasmussen, S. Sun, and T. J. Lim, "A matrix-algebraic approach to linear parallel interference cancellation in CDMA," *IEEE Trans. Commun.*, vol. 48, no. 1, pp. 152–161, Jan. 2000.
- [48] M. C. Reed, C. B. Schlegel, P. D. Alexander, and J. A. Asenstorfer, "Iterative multiuser detection for CDMA with FEC: Near-single-user performance," *IEEE Trans. Commun.*, vol. 46, no. 12, pp. 1693–1699, Dec. 1998.
- [49] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.
- [50] S. Morosi, R. Fantacci, E. D. Re, and A. Chiassai, "Design of turbo-MUD receivers for overloaded CDMA systems by density evolution technique," *IEEE Trans. Wireless Commun.*, vol. 6, no. 10, pp. 3552–3557, Oct. 2007.
- [51] P. Kumar and S. Chakrabarti, "An analytical model of iterative interference cancellation receiver for orthogonal/orthogonal overloaded DS-CDMA system," *Int. J. Wireless Inf. Netw.*, vol. 17, nos. 1–2, pp. 64–72, Jun. 2010.
- [52] S. Sasipriya and C. S. Ravichandran, "An analytical model of iterative interference cancellation receiver for orthogonal/orthogonal overloaded DS-CDMA system," *Telecommun. Syst.*, vol. 56, no. 4, pp. 509–518, Aug. 2014.
- [53] G. Romano, F. Palmieri, and P. K. Willett, "Soft iterative decoding for overloaded CDMA," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Philadelphia, PA, U.S.A., vol. 3, Mar. 2005, pp. 733–736.
- [54] A. Nafkha, C. Roland, and E. Boutillon, "A near-optimal multiuser detector for MC-CDMA systems using geometrical approach," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Philadelphia, PA, USA, vol. 3, Mar. 2005, pp. 877–880.
- [55] G. Mangani and A. K. Chaturvedi, "Application of computational geometry to multiuser detection in CDMA," *IEEE Trans. Commun.*, vol. 54, no. 2, pp. 204–207, Feb. 2006.
- [56] S. Lu, W. Hou, J. Cheng, and H. Kamabe, "A new kind of nonbinary uniquely decodable codes with arbitrary code length for multiple-access adder channel," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Guangzhou, China, Nov. 2018, pp. 1–5.
- [57] D. S. Dummit and R. M. Foote, *Abstract Algebra*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [58] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Hoboken, NJ, USA: Wiley, 2005.
- [59] C. Olds, A. Lax, and G. Davidoff, *The Geometry of Numbers*, 1st ed. Washington, DC, USA: Mathematical Association of America, 2000.
- [60] S. Craw, "Manhattan distance," in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA, USA: Springer, 2010, p. 639.
- [61] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 35, no. 1, pp. 123–136, Jan. 1989.
- [62] M. Fantuz and C. D'Amours, "Low complexity PIC-MMSE detector for LDS systems," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, Edmonton, AB, Canada, May 2019, pp. 1–4.
- [63] K. K. Wong, A. Paulraj, and R. D. Murch, "Efficient high-performance decoding for overloaded MIMO antenna systems," *IEEE Trans. Wireless Commun.*, vol. 6, no. 5, pp. 1833–1843, May 2007.



MICHEL KULHANDJIAN (Senior Member, IEEE) received the B.S. degree (*summa cum laude*) in electronics engineering and computer science (minor) from The American University in Cairo (AUC), in 2005, and the M.S. and Ph.D. degrees in electrical engineering from The State University of New York at Buffalo, in 2007 and 2012, respectively.

He was employed at Alcatel-Lucent, Ottawa, ON, in 2012. In the same year, he was appointed as a Research Associate at EION Inc. He is currently a Research Scientist at the School of Electrical Engineering and Computer Science, University of Ottawa. He is also employed as a Senior Embedded Software Engineer at L3Harris Technologies. He has received the Natural Science and Engineering Research Council of Canada (NSERC) Industrial Research and Development Fellowship (IRDF). His research interests include wireless multiple access

communications, adaptive coded modulation, waveform design for overloaded code-division multiplexing applications, channel coding, space-time coding, adaptive multiuser detection, statistical signal processing, covert communications, spread-spectrum steganography, and steganalysis. He actively serves as a member for the Technical Program Committee (TPC) of IEEE WCNC, IEEE GLOBECOM, IEEE ICC, and IEEE VTC. He has served as a Guest Editor for *Journal of Sensor and Actuator Networks*.



HOVANNES KULHANDJIAN (Senior Member, IEEE) received the B.S. degree (*magna cum laude*) in electronics engineering from The American University in Cairo, Cairo, Egypt, in 2008, and the M.S. and Ph.D. degrees in electrical engineering from The State University of New York at Buffalo, Buffalo, NY, USA, in 2010 and 2014, respectively. From December 2014 to July 2015, he was an Associate Research Engineer with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, California State University, Fresno, Fresno, CA, USA. His current research interests include wireless communications and networking, with applications to underwater acoustic communications, visible light communications, and applied machine learning. He actively serves as a member for the Technical Program Committee for ACM and IEEE conferences, such as IEEE GLOBECOM, ICC, UComms, PIMRC, WD, ACM WUWNet, and among others. He has also served as the Session Co-Chair for IEEE UComms 2020 and the Session Chair for ACM WUWNet 2019. He has served as a Guest Editor for IEEE ACCESS—Special Section Journal on Underwater Wireless Communications and Networking.



CLAUDE D'AMOURS (Member, IEEE) received the B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical engineering from the University of Ottawa, in 1990, 1992, and 1995, respectively. In 1992, he was employed as a Systems Engineer at Calian Communications Ltd. In 1995, he joined the Communications Research Centre, Ottawa, ON, Canada, as a Systems Engineer. Later in 1995, he joined the Department of Electrical and Computer Engineering, Royal Military College of Canada, Kingston, ON, Canada, as an Assistant Professor. He joined the School of Information Technology and Engineering (SITE), which has since been renamed as the School of Electrical Engineering and Computer Science (EECS), University of Ottawa, as an Assistant Professor, in 1999. From 2007 to 2011, he has worked as the Vice Dean of undergraduate studies with the Faculty of Engineering and has been working as the Director of the School of EECS, University of Ottawa, since 2013. His research interests include physical layer technologies for wireless communications systems, notably in multiple access techniques and interference cancellation.



HALIM YANIKOMEROGLU (Fellow, IEEE) is currently a Professor with the Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada. His research group has made substantial contributions to 4G and 5G wireless technologies. His group's current focus is the aerial and satellite networks for the 6G and beyond-6G era. His extensive collaboration with industry resulted in 39 granted patents. He is a fellow of the Engineering Institute of Canada (EIC) and the Canadian Academy of Engineering (CAE). He has received several awards for his research, teaching, and service. He is a Distinguished Speaker for both IEEE Communications Society and IEEE Vehicular Technology Society.



DIMITRIS A. PADOS (Senior Member, IEEE) received the Diploma degree in computer science and engineering (five-year program) from the University of Patras, Greece, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, USA.

From 1997 to 2017, he was with the Department of Electrical Engineering, The State University of New York at Buffalo, as an Assistant Professor, an Associate Professor, a Professor, and a Clifford

C. Furnas Chair Professor of electrical engineering. He also worked as the Associate Chair and was appointed as the Chair of the Department of Electrical Engineering. He was a University Faculty Senator (elected) four times and worked at the Faculty Senate Executive Committee for two terms. In 2017, he joined Florida Atlantic University, Boca Raton, FL, USA, as the Schmidt Eminent Scholar Professor of engineering and computer science and a fellow of the Institute for Sensing and Embedded Network Systems Engineering (I-SENSE). He is the Founding Director of the FAU Center for Connected Autonomy and Artificial Intelligence (<https://ca-ai.fau.edu/>). He has served as a Principal Investigator on federal grants (NSF and DoD) of about \$17M and has been author/coauthor of 230 journal and conference proceedings articles in predominantly IEEE venues. Notable technical contributions from his team include small-sample-support adaptive filtering (auxiliary-vector filters), optimal total-squared-correlation multiple-access code sets (Karystinos-Pados bounds and designs), optimal spread-spectrum data hiding, L1-norm principal-component analysis (optimal algorithms for exact L1-norm PCA), and robust localization in extreme environments (L1-norm feature extraction from complex-valued data).

Dr. Pados is a member of the IEEE Communications, IEEE Signal Processing, IEEE Information Theory, and IEEE Computational Intelligence Societies. Articles that he has coauthored with his students received the 2001 IEEE International Conference on Telecommunications Best Paper Award, the 2003 IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS Outstanding Paper Award, the 2010 IEEE International

Communications Conference (ICC) Best Paper Award in signal processing for communications, the 2013 International Symposium on Wireless Communication Systems Best Paper Award in physical layer communications and signal processing, the Best of IEEE GLOBECOM 2014-Top 50 Papers Distinction, the Best Paper in the 2016 IEEE International Conference on Multimedia Big Data, and the Paper Distinctions at the International Workshop on Antenna Technology (iWAT) 2019 and IEEE/MTS Oceans 2020. He was a recipient of the 2009 SUNY-Wide Chancellor's Award for Excellence in Teaching and the 2011 University at Buffalo Exceptional Scholar-Sustained Achievement Award. He was presented with the 2021 Florida Atlantic Research and Development Authority Distinguished Researcher Award, Boca Raton, FL, USA. He has served as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS and the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.



GURGEN KHACHATRIAN (Member, IEEE) is currently a well-known and highly regarded Applied Scientist, a Scholar, and a Technologist in the field of error control coding and cryptography. For his career achievements in these fields, he was elected as a Full Member of the Armenian National Academy of Sciences, in 1996, which is the highest honor awarded to scientists in Armenia. He worked as a Chief Cryptographer at Cylink Corporation (Nasdaq: CYLK), Sunnyvale, CA, USA, from 1999 to 2001. From 2001 to 2007, he worked as a Chief

Scientist at Quantum Digital Solutions Corporation, Santa Monica, CA, USA. He was a Professor at the American University of Armenia (AUA), from 2011 to 2018. Since 2019, he has been a Chief Cryptography Officer at Quantum Digital Solutions Corporation (QDSC), Marina Del Rey, CA, USA; and a Professor on leave from AUA.

...