LP Decodable Permutation Codes based on Linearly Constrained Permutation Matrices

Tadashi Wadayama and Manabu Hagiwara

Abstract

A set of linearly constrained permutation matrices are proposed for constructing a class of permutation codes. Making use of linear constraints imposed on the permutation matrices, we can formulate a minimum Euclidian distance decoding problem for the proposed class of permutation codes as a linear programming (LP) problem. The main feature of this class of permutation codes, called *LP decodable permutation codes*, is this LP decodability. It is demonstrated that the LP decoding performance of the proposed class of permutation codes is characterized by the vertices of the code polytope of the code. Two types of linear constraints are discussed; one is structured constraints and another is random constraints. The structured constraints such as pure involution lead to an efficient encoding algorithm. On the other hand, the random constraints enable us to use probabilistic methods for analyzing several code properties such as the average cardinality and the average weight distribution.

Index Terms: permutation codes, linear programming, polytope, decoding, error correction

I. INTRODUCTION

The class of linear codes defined over a finite field is ubiquitously employed in digital equipments for achieving reliable communication and storage systems. For example, the class of codes includes practically important codes such as Reed-Solomon codes, BCH codes, and LDPC codes. The linearity of codes enables us to use efficient encoding and decoding algorithms based on their linear algebraic properties.

On the other hand, there are some classes of nonlinear codes which are interesting from both theoretical and practical points of view. The class of *permutation codes* is such a class of nonlinear codes.

The origin of permutation codes dates back to the 1960s. Slepian [17] proposed a class of simple permutation codes, which is referred to as *permutation modulation*, and efficient soft decoding algorithms for these codes. The variant I code [17] is obtained by applying all the permutations to the initial vector

$$(\overbrace{\mu_1, \mu_1 \ldots, \mu_1}^{n_1} \overbrace{\mu_2, \ldots, \mu_2}^{n_2} \cdots \overbrace{\mu_k, \mu_k \ldots, \mu_k}^{n_k}),$$

where μ_i is a real value and $n = n_1 + \cdots + n_k$. This research has been extended and investigated by a number of researchers. Biglieri and Elia [19], Karlof [18], Ingemarsson [20] studied optimization of the initial vector of the permutation modulation. Berger et al. [21] discussed applications of permutation codes to source coding problems.

There is another thread of researches on a class of permutation codes of length n whose codeword contains exactly n-distinct symbols; i.e., any codeword can be obtained by applying a permutation to an initial vector, e.g., $(0, 1, \ldots, n - 1)$.

Some fundamental properties of such permutation codes were discussed in Blake et al. [1], and Frankl and Deza [8]. Vinck [13] [14] proposed applications of permutation codes for power-line communication and this triggered subsequent works on

T. Wadayama is with Nagoya Institute of Technology, Nagoya City, Aichi, 466-8555, JAPAN. (e-mail:wadayama@nitech.ac.jp). M. Hagiwara is with National Institute of Advanced Industrial Science and Technology, Central 2, 1-1-1 Umezono, Tsukuba City, Ibaraki, 305-8568, JAPAN (email: hagiwara.hagiwara@aist.go.jp). A part of this work will be presented at International Symposium on Information Theory, 2011. The initial version of this work has been included in e-preprint server arXiv since Nov. 2010 (identificator:*arXiv:1011.6441*).

permutation codes. Wadayama and Vinck [16] presented a multi-level construction of permutation codes with large minimum Hamming distance. A number of constructions for permutation codes have been developed, including the construction given in [4] [6]. Especially, the idea of a distance-preserving map due to Vinck and Ferreira [15] had influence on the study of permutation codes such as subsequent works by Chang et al. [2] [3].

Recently, rank modulation codes for flash memory proposed by Jiang et al. [9] [10] generated renewed interest in permutation codes. For example, for flash memory coding, Kløve et al. gave a new construction for permutation codes based on Chebyshev Distance [11], which is an appropriate distance measure for flash memory coding. Barg and Mazumdar [24] also studied some fundamental bounds on permutation codes in terms of the Kendall tau distance.

In order to employ a permutation code in a practical application, efficient encoding and soft-decoding algorithms are crucial to achieve reliable communication over noisy channels, such as an AWGN channel. Nonlinearity of permutation codes prevents the use of conventional encoding and decoding techniques based on linear algebraic properties. Although much works on permutation codes have been conducted, an aspect of efficient soft-decoding has not been intensively discussed so far. Therefore, there is still room for further researches on permutation codes with efficient encoding and soft-decoding algorithms.

In this paper, a new class of permutation codes called *LP decodable permutation codes* is introduced. An LP decodable permutation code is obtained by applying permutation matrices satisfying certain linear constraints to an n-dimensional real initial vector.

It is well known that permutation matrices are vertices of the Birkhoff polytope [35], which is the set of doubly stochastic matrices. Thus, a set of linearly constrained permutation matrices can be expressed by a set of linear equalities and linear inequalities. This property leads to the main feature of this class of permutation codes: *LP-decodable property*. For this class of codes, a decoding problem can be formulated as a linear programming (LP) problem. This means that we can exploit efficient LP solvers based on simplex methods or interior point methods to decode LP decodable permutation codes.

Furthermore, for a combination of this class of codes and its LP decoding, the maximum likelihood (ML) certificate property can be proved as in the case of the LP decoding for LDPC codes [7]. This is due to the fact that the LP problem given in this paper is a relaxed problem of an ML decoding problem.

In general, a fundamental polytope [27] [7] used for LP decoding of LDPC codes contains a number of fractional vertices, which are a major source of sub-optimality of LP decoding. The constraints corresponding to an LDPC matrix are defined based on $\mathbb{F}_2$-arithmetics. On the other hand, an LP decoder works on the real number field. This domain mismatch produces many undesirable fractional vertices on the fundamental polytope. One motivation of the present study is to establish a coding scheme without this mismatch. In other words, the LP decodable permutation codes are defined on the real number field and are decoded using an LP solver working on the real number field.

The organization of the paper is as follows. Section II introduces some definitions and notation required for discussion. Section III gives the definition of the LP decodable permutation codes and its decoding algorithm. Section IV provides analysis for decoding performance of LP decoding and ML decoding. Section V presents some classes of permutation codes which are easy to encode. Section VI offers probabilistic analysis on the cardinality and weight distribution of random LP decodable permutation codes. Section VII gives a concluding summary.

II. PRELIMINARIES

A. Notation and definition

In this paper, matrices are represented by capital letters and a vector is assumed to be a column vector. Let X be an $n \times n$ real matrix. The notation $X \geq 0$ means that every element in X is non-negative. The notation $\mathsf{vec}(X)$ represents a vectorization of X given by

$$\mathsf{vec}(X) \triangleq (X_{1,1} \cdots X_{1,n} \; X_{2,1} \; \cdots X_{2,n}, X_{3,1} \cdots X_{n,n})^T .$$

The vector $\mathbf{1}$ is the all-one vector whose length is determined by the context. The norm $|| \cdot ||$ denotes the Euclidean norm given by $||x|| \triangleq (x^T x)^{1/2}$. The trace function $\mathsf{trace}(X)$ returns the sum of the diagonal elements of X. The sets $\mathbb{R}, \mathbb{Z}$ are the sets of real numbers and integers, respectively. The set $[\alpha, \beta]$ denotes the set of consecutive integers from $\alpha \in \mathbb{Z}$ to $\beta \in \mathbb{Z}$.

The symbol $\trianglelefteq$ means

$$\begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix} \trianglelefteq \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \Leftrightarrow \forall i \in [1, m], a_i \trianglelefteq_i b_i,$$

where $\trianglelefteq_i$ is either $=$ or $\leq$. For simplicity, the notation $\trianglelefteq = (\trianglelefteq_1, \trianglelefteq_2, \ldots, \trianglelefteq_m)^T$ is used to define $\trianglelefteq$ (e.g., $\trianglelefteq = (\leq, =, \leq)^T$).

The next definition gives a class of matrices of crucial importance in this paper.

Definition 1 (Permutation matrix): An $n \times n$ binary real matrix $X \triangleq (X_{i,j})_{i,j \in [1,n]} \in \{0,1\}^{n \times n}$ is called a *permutation matrix* if and only if

$$\forall i, j \in [1,n], \sum_{j' \in [1,n]} X_{i,j'} = 1, \sum_{i' \in [1,n]} X_{i',j} = 1. \tag{1}$$

$\square$

The set of $n \times n$ permutation matrices is denoted by Π_n. The cardinality of Π_n is $n!$.

Removing the binary constraint from the definition of the permutation matrices, we have the definition of doubly stochastic matrices.

Definition 2 (Doubly stochastic matrix): An $n \times n$ non-negative real matrix $X \triangleq (X_{i,j})_{i,j \in [1,n]}$ is called a *doubly stochastic matrix* if and only if (1) holds. $\square$

The following theorem for a double stochastic matrix implies that the set of doubly stochastic matrices is a convex polytope.

Theorem 1 (Birkhoff-von Neumann theorem [35] [36]): Every doubly stochastic matrix is a convex combination of permutation matrices.

The set of $n \times n$ doubly stochastic matrices is a polytope called the *Birkhoff polytope* B_n [35], which is also known as perfect matching polytope. The Birkhoff polytope is a $(n-1)^2$-dimensional convex polytope with $n!$-vertices and n^2-facets [34]. The Birkhoff-von Neumann theorem implies that any vertex (i.e., extreme point) of the Birkhoff polytope is a permutation matrix and vice versa.

B. LP decoding for permutation vectors

Assume that $s \in \mathbb{R}^n$, called the *initial vector*, is given[1]. The set of images of s by left action of $X \in \Pi_n$ is called the *permutation vectors* of s, which is given by

$$\Lambda(s) \triangleq \{Xs \mid X \in \Pi_n\}. \tag{2}$$

[1]The elements in s are not necessarily distinct each other.

For example, if $s = (0, 1, 2)^T$, then $\Lambda(s)$ is given by

$$\Lambda(s) = \{(0, 1, 2), (0, 2, 1), (1, 0, 2), (1, 2, 0), (2, 0, 1)(2, 1, 0)\}.$$

We here consider a situation such that a vector of $\Lambda(s)$ is transmitted to a receiver over an AWGN channel. In such a case, it is desirable to use an ML decoding algorithm to estimate the transmitted vector. The ML decoding rule can be describe as

$$\hat{x} = \arg \min_{x \in \Lambda(s)} ||y - x||^2, \tag{3}$$

where y is a received word.

The next theorem states that the ML decoding for $\Lambda(s)$ can be formulated as the following LP problem.

Theorem 2 (LP decoding and ML certificate property): Assume that a vector in $\Lambda(s)$ is transmitted over an AWGN channel and that $y \in \mathbb{R}^n$ is received on the receiver side. We also suppose that $\hat{x} = \arg \min_{x \in \Lambda(s)} ||y - x||^2$ is uniquely determined from y. Let X^* be the solution of the following LP problem:

$$\text{maximize trace}(C^T X)$$

$$\text{subject to}$$

$$\begin{array}{rcl} X & \in & \mathbb{R}^{n \times n} \\ X\mathbf{1} & = & \mathbf{1} \\ \mathbf{1}^T X & = & \mathbf{1}^T \\ X & \geq & 0, \end{array} \tag{4}$$

where $C \triangleq ys^T$. If X^* is integral, $\hat{x} = X^*s$ holds.

Proof: The linear constraints in the above LP problem implies that X is constrained to be a doubly stochastic matrix.

On the other hand, the ML decoding rule can be recast as follows:

$$\begin{aligned} \hat{x} & = & \arg \min_{x \in \Lambda(s)} ||y - x||^2 \\ & = & (\arg \min_{X \in \Pi_n} ||y - Xs||^2)s \\ & = & (\arg \min_{X \in \Pi_n} (||y||^2 - 2y^T(Xs) + ||Xs||^2))s \\ & = & (\arg \max_{X \in \Pi_n} y^T Xs)s = (\arg \max_{X \in \Pi_n} \text{trace}(C^T X))s, \end{aligned}$$

where $C = ys^T$. Note that

$$\text{trace}(C^T X) = \sum_{i=1}^n \sum_{j=1}^n C_{i,j} X_{i,j}. \tag{5}$$

Since the vertices of the Birkhoff polytope is a permutation matrix, the ML decoding can be formulated as an integer LP (ILP) problem:

$$\text{maximize trace}(C^T X)$$

$$\text{subject to } X \in B_n, \quad X \text{ is an integral matrix.}$$

By removing the integral constraint (X is an integral matrix), we obtain the LP problem (4). If the solution of this LP problem is integral, it must coincide with the solution of the above ILP problem. ∎

As we have seen, the feasible set of the above LP problem is the Birkhoff polytope. Thus, an output of the above LP is highly likely integral.

The following example illustrates an LP decoding procedure.

Example 1: Let $s \triangleq (0,1)^T$. In this case, the set of permutation vectors becomes $\Lambda(s) = \{(0,1)^T, (1,0)^T\}$. Assume that $y = (0.9, 0.2)^T$ is received. In this case,

$$C = y s^T = \begin{pmatrix} 0.9 \\ 0.2 \end{pmatrix} (0\ 1) = \begin{pmatrix} 0 & 0.9 \\ 0 & 0.2 \end{pmatrix}$$

is obtained. By letting

$$X = \begin{pmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{pmatrix},$$

we have the objective function

$$\mathrm{trace}\left(\begin{pmatrix} 0 & 0 \\ 0.9 & 0.2 \end{pmatrix} \begin{pmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{pmatrix}\right) = 0.9X_{1,2} + 0.2X_{2,2}.$$

As a result, the LP decoding problem is given by

$$\text{maximize } 0.9X_{1,2} + 0.2X_{2,2} \text{ subject to}$$

$$X_{1,1} + X_{1,2} = 1, \quad X_{2,1} + X_{2,2} = 1,$$

$$X_{1,1} + X_{2,1} = 1, \quad X_{1,2} + X_{2,2} = 1$$

$$X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2} \geq 0.$$

The solution of the problem is

$$X^* = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

and then we have the estimated word $X^* s = (1,0)^T$. □

III. LINEARLY CONSTRAINED PERMUTATION MATRICES AND LP DECODABLE PERMUTATION CODES

It is natural to consider an extension of the LP decoding presented in the previous section. Additional linear constraints imposed on Π_n produce a restricted set of $\Lambda(s)$. A decoding problem of such a set can be formulated as an LP problem, as in the case of the ML decoding of $\Lambda(s)$.

A. Definitions

The next definition for linearly constrained permutations gives an LP-decodable subset of $\Lambda(s)$.

Definition 3 (linearly constrained permutation matrix): Let m, n be positive integers. Assume that $A \in \mathbb{Z}^{m \times n^2}$, $b \in \mathbb{Z}^m$ and $\trianglelefteq \in \{=, \leq\}^m$ are given. A set of *linearly constrained permutation matrices* is defined by

$$\Pi(A, b, \trianglelefteq) \triangleq \{X \in \Pi_n \mid A \operatorname{vec}(X) \trianglelefteq b\}. \tag{6}$$

□

Note that $A \operatorname{vec}(X) \trianglelefteq b$ formally represents additional m equalities and inequalities. These additional constraints provide a restriction on permutation matrices.

From the linearly constrained permutation matrices, LP decodable permutation codes are naturally defined as follows.

Definition 4 (LP decodable permutation code): Assume the same set up as in Definition 3. Suppose also that $s \in \mathbb{R}^n$ is given. The set of vectors $\Lambda(A, b, \trianglelefteq, s)$ given by

$$\Lambda(A, b, \trianglelefteq, s) \triangleq \{Xs \in \mathbb{R}^n \mid X \in \Pi(A, b, \trianglelefteq)\} \tag{7}$$

is called an LP decodable permutation code. $\qquad\square$

If $\Rightarrow X^{(1)}s \neq X^{(2)}s$ holds for any $X^{(1)}, X^{(2)}(X^{(1)} \neq X^{(2)}) \in \Pi(A, b, \trianglelefteq)$, then an LP decodable permutation code is said to be *non-singlar*. Namely, there is one-to-one correspondence between permutation matrices in $\Pi(A, b, \trianglelefteq)$ and codewords of $\Lambda(A, b, \trianglelefteq, s)$ if a code is non-singular. Note that a code may become singular if identical symbols exist in s.

The next example shows a case where an additional linear constraint imposes a restriction on permutation matrices.

Example 2: Consider the set of linearly constrained permutation matrices which consists of 4×4 permutation matrices satisfying the linear constraint $\mathsf{trace}(X) = 0$. The constraint implies that the diagonal elements of the permutation matrices are constrained to be zero. This means that such permutation matrices correspond to permutations without fixed points, which are called *derangements*. For $n = 4$, there are 9-derangement permutation matrices as follows:

$$
\begin{pmatrix} 0100 \\ 1000 \\ 0001 \\ 0010 \end{pmatrix}
\begin{pmatrix} 0100 \\ 0010 \\ 0001 \\ 1000 \end{pmatrix}
\begin{pmatrix} 0100 \\ 0001 \\ 1000 \\ 0010 \end{pmatrix}
$$

$$
\begin{pmatrix} 0010 \\ 1000 \\ 0001 \\ 0100 \end{pmatrix}
\begin{pmatrix} 0010 \\ 0001 \\ 1000 \\ 0100 \end{pmatrix}
\begin{pmatrix} 0010 \\ 0001 \\ 0100 \\ 1000 \end{pmatrix}
$$

$$
\begin{pmatrix} 0001 \\ 1000 \\ 0100 \\ 0010 \end{pmatrix}
\begin{pmatrix} 0001 \\ 0010 \\ 1000 \\ 0100 \end{pmatrix}
\begin{pmatrix} 0001 \\ 0010 \\ 0100 \\ 1000 \end{pmatrix}.
$$

In this case, the triple $(A, b, \trianglelefteq)$ is defined by

$$A = \mathsf{vec}(I), \quad b = 0, \quad \trianglelefteq = (=), \tag{8}$$

where I is the 4×4 identity matrix. Multiplying these matrices to the initial vector $s = (0, 1, 2, 3)^T$ from left, we immediately obtain the members of $\Lambda(A, b, \trianglelefteq, (0, 1, 2, 3)^T)$:

$$
\begin{array}{ccc}
(1, 0, 3, 2)^T, & (1, 2, 3, 0)^T, & (1, 3, 0, 2)^T, \\
(2, 0, 3, 1)^T, & (2, 3, 0, 1)^T, & (2, 3, 1, 0)^T, \\
(3, 0, 1, 2)^T, & (3, 2, 0, 1)^T, & (3, 2, 1, 0)^T.
\end{array} \tag{9}
$$

This code is thus non-singular. If the initial vector is

$$s = (0, 0, 0, 0)^T,$$

then the resulting code has the only codeword $(0, 0, 0, 0)$. In this case, the code becomes singular. $\qquad\square$

B. LP decoding for LP decodable permutation codes

The LP decoding of $\Lambda(A, b, \trianglelefteq, s)$ is a natural extension of the LP decoding for $\Lambda(s)$. Assume that a vector in $\Lambda(A, b, \trianglelefteq, s)$ is transmitted over an AWGN channel and $y \in \mathbb{R}^n$ is given. The procedure for the LP decoding of $\Lambda(A, b, \trianglelefteq, s)$ is given as follows.

LP decoding for an LP decodable permutation code

1) Solve the following LP problem and let X^* be the solution.

$$\text{maximize } \text{trace}(C^T X)$$

$$\text{subject to}$$

$$
\begin{aligned}
X &\in& \mathbb{R}^{n \times n}, \\
X &\geq& 0, \\
X\mathbf{1} &=& \mathbf{1}, \\
\mathbf{1}^T X &=& \mathbf{1}^T, \\
A\, \text{vec}(X) &\trianglelefteq& b,
\end{aligned}
\tag{10}
$$

where $C = ys^T$.

2) Output X^*s if X^* is integral. Otherwise, declare decoding failure.

C. Remarks

Several remarks should be made regarding the LP decoding for $\Lambda(A, b, \trianglelefteq, s)$.

The feasible set of (10) is a subset of the feasible set of (4). All the matrices in $\Pi(A, b, \trianglelefteq)$ are feasible and permutation matrices which do not belong to $\Pi(A, b, \trianglelefteq)$ are infeasible. This implies that all the integral points of the feasible set (10) coincide with $\Pi(A, b, \trianglelefteq)$.

The LP problem (10) is a relaxed problem of the ML decoding problem over AWGN channels:

$$\text{minimize } ||y - x||^2 \text{ subject to } x \in \Lambda(A, b, \trianglelefteq, s). \tag{11}$$

This can be easily shown, as in the case (4). As a consequence of the above properties on integral points and on the relaxation, it can be concluded that the LP decoding for $\Lambda(A, b, \trianglelefteq, s)$ has the ML-certificate property as well. Namely, if the output of LP decoding is not decoding failure (i.e., X^* is integral), the output is exactly the same as the solution of the minimum distance decoding problem (11). Note that the LP decoding presented above becomes the ML decoding if the code polytope is integral.

The feasible set of the LP problem (10) is the intersection of the Birkhoff polytope and a (possibly unbounded) convex set defined by the additional constraints. The intersection becomes a polytope which is called a *code polytope*. The decoding performance of LP decoding is closely related to the code polytope given by the following definition.

Definition 5 (Code polytope): The polytope $\mathcal{P}(A, b, \trianglelefteq)$ defined by

$$\mathcal{P}(A, b, \trianglelefteq) \overset{\triangle}{=} B_n \cap \{X \in \mathbb{R}^{n \times n} \mid A\, \text{vec}(X) \trianglelefteq b\} \tag{12}$$

is called the code polytope for $\Pi(A, b, \trianglelefteq)$, where B_n is the Birkhoff polytope corresponding to Π_n. □
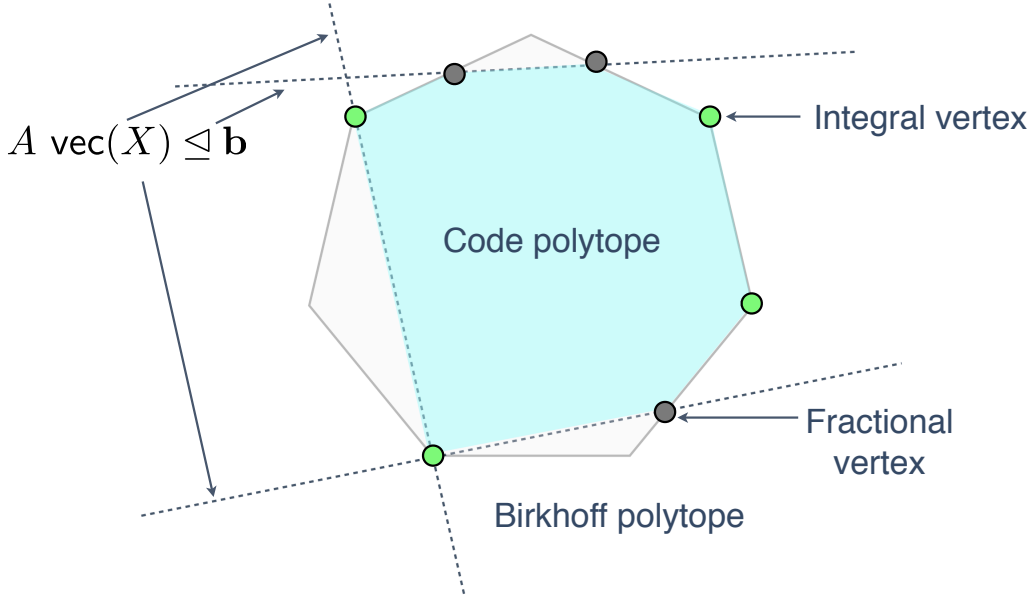
Fig. 1.   Code polytope $\mathcal{P}(A, b, \trianglelefteq)$

Figure 1 illustrates a code polytope. It should be remarked that the set of integral vertices of the code polytope coincides with $\Pi(A, b, \trianglelefteq)$. Due to additional linear constraints $A\,\mathsf{vec}(X) \trianglelefteq b$, a code polytope may have some fractional vertices, which contain components of fractional number.

In an LP decoding process, these fractional vertices become possible candidates of an LP solution. Thus, these fractional vertices can be considered as *pseudo permutation matrices* which degrade the decoding performance of the LP decoding.

IV. ANALYSIS FOR DECODING PERFORMANCE OF LP DECODING AND ML DECODING

In this section, upper bounds on decoding error probability for LP decoding and ML decoding are presented.

A. Upper bound on LP decoding error probability

An advantage of the LP formulation of a decoding algorithm is its simplicity for detailed decoding performance analysis. The geometrical properties of a code polytope is closely related to its decoding performance of the LP decoding. We can evaluate the block error probability of the proposed scheme with reasonable accuracy if we have enough information on the set of vertices of a code polytope. The bound presented in this section has close relationship to the pseudo codeword analysis on LDPC codes [5].

In this section, a set of parameters $A, b, \trianglelefteq, s$ are assumed to be given. Let V be the set of vertices of the code polytope $\mathcal{P}(A, b, \trianglelefteq, s)$. In general, V contains fractional vertices.

The next lemma gives bridge between a code polytope and corresponding decoding error probability.

Lemma 1 (Upper bound on block error rate for LPD): Assume that a codeword Xs is transmitted to a receiver via an AWGN channel, where $X \in \Pi(A, b, \trianglelefteq)$. The additive white Gaussian noise with mean 0 and variance σ^2 is assumed. The receiver uses the LP decoding algorithm presented in the previous section. In this case, the block error probability $P_{LP}(X)$ is upper bounded by

$$P_{LP}(X) \leq \sum_{\tilde{X} \in V \setminus \{X\}} Q\left(\frac{||Xs||^2 - (\tilde{X}s)^T Xs}{\sigma ||\tilde{X}s - Xs||} \right), \tag{13}$$

where the Q-function is the tail probability of the normal Gaussian distribution, which is given by

$$Q(x) \triangleq \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) dt. \tag{14}$$

Proof: Let $y = Xs + z$, where z is an additive white Gaussian noise term. We first consider the pairwise block error probability $P_e(X, \tilde{X})$ between X and $\tilde{X} \in \Pi(A, b, \trianglelefteq)$, which is given by

$$P_e(X, \tilde{X}) \triangleq Prob[y^T \tilde{X}s \geq y^T Xs]. \tag{15}$$

Namely, $P_e(X, \tilde{X})$ is the probability such that $\tilde{X}s$ is more likely than Xs for a given y under the assumption that only $\tilde{X}$ and X are allowable permutation matrices.

The difference $y^T \tilde{X}s - y^T Xs$ can be transformed into

$$
\begin{aligned}
y^T \tilde{X}s - y^T Xs &= (Xs + z)^T (\tilde{X}s - Xs) \\
&= (\tilde{X}s - Xs)^T z + (\tilde{X}s - Xs)^T Xs \\
&= (\tilde{X}s - Xs)^T z \\
&\quad - (||Xs||^2 - (\tilde{X}s)^T Xs).
\end{aligned}
\tag{16}
$$

We thus have

$$Prob[y^T \tilde{X}s \geq y^T Xs] = Prob[a^T z \geq b], \tag{17}$$

where $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are given by

$$
\begin{aligned}
a &\triangleq \tilde{X}s - Xs, &\tag{18} \\
b &\triangleq ||Xs||^2 - (\tilde{X}s)^T Xs. &\tag{19}
\end{aligned}
$$

The left-hand side of $a^T z \geq b$ is a linear combination of Gaussian noises. The mean of $a^T z$ is zero and the variance is given by

$$Var[a^T z] = \sigma^2 ||a||^2. \tag{20}$$

The probability such that the Gaussian random variable $a^T z$ takes a value larger than or equal to b can be expressed as

$$
\begin{aligned}
P_e(X, \tilde{X}) &= Prob[a^T z \geq b] \\
&= Q\left(\frac{b}{\sigma ||a||}\right).
\end{aligned}
\tag{21}
$$

Combining the union bound and this pairwise error probability, we immediately obtain the claim of this lemma. ∎

The upper bound on decoding error probability in Lemma 1 naturally leads to a pseudo distance measure on $\mathbb{R}^{n \times n}$.

Definition 6 (Pseudo distance): The function

$$D_s(X, \tilde{X}) \triangleq \frac{||Xs||^2 - (\tilde{X}s)^T Xs}{||\tilde{X}s - Xs||} \tag{22}$$

is called the *pseudo distance* where $X, \tilde{X} \in \mathbb{R}^{n \times n}$ are doubly stochastic matrices. □

Note that $D_s(\cdot, \cdot)$ is not a distance function since it does not satisfy the axioms of distance. In terms of decoding error probability, geometry of the vertices of a code polytope should be established based on this pseudo distance.

For example, in high SNR regime, the *minimum pseudo distance*

$$\Delta_s \triangleq \min_{X \in \Pi(A,b,\trianglelefteq), \tilde{X} \in V, \tilde{X} \neq X} D_s(X, \tilde{X}) \tag{23}$$

is expected to be highly influential to the decoding error probability.

Example 3: Suppose the linear constraint $\text{trace}(X) = 1$ where $n = 3$. In this case, the code polytope has the following 5-vertices:

$$M^{(1)} \triangleq \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \ M^{(2)} \triangleq \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$M^{(3)} \triangleq \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \ M^{(4)} \triangleq \begin{pmatrix} 1/3 & 0 & 2/3 \\ 2/3 & 1/3 & 0 \\ 0 & 2/3 & 1/3 \end{pmatrix},$$

$$M^{(5)} \triangleq \begin{pmatrix} 1/3 & 2/3 & 0 \\ 0 & 1/3 & 2/3 \\ 2/3 & 0 & 1/3 \end{pmatrix}. \tag{24}$$

In this case, the set of vertices consists of 3-integral vertices and 2-fractional vertices. Let $s = (0, 1, 2)^T$. The pseudo distance distribution form $M^{(1)}$ is given by

$$\begin{aligned} D_s(M^{(1)}, M^{(2)}) &= 1.388730 \\ D_s(M^{(1)}, M^{(3)}) &= 1.224745 \\ D_s(M^{(1)}, M^{(4)}) &= 1.224745 \\ D_s(M^{(1)}, M^{(5)}) &= 1.224745. \end{aligned}$$

$\square$

B. Upper bound on ML decoding error probability

Assume the same setting as in the previous subsection. In the case of ML decoding, we can neglect the effect of fractional vertices. Therefore, we obtain an upper bound on the ML block error probability

$$\begin{aligned} P_{ML}(X) &\leq \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq) \setminus \{X\}} Q\left(\frac{||Xs||^2 - (\tilde{X}s)^T Xs}{\sigma||\tilde{X}s - Xs||} \right) \\ &= \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq) \setminus \{X\}} Q\left(\frac{||\tilde{X}s - Xs||}{2\sigma} \right) \end{aligned} \tag{25}$$

based on a similar argument. The above equality holds since $||Xs|| = ||\tilde{X}s||$ holds for any $\tilde{X} \in \Pi(A, b, \trianglelefteq)$. Note that this simplification cannot apply to $\tilde{X}$ if $\tilde{X}$ is a fractional vertex. This is because the preservation of Euclidean norm does not hold in general for a doubly stochastic matrix. For example, we have

$$\left\| \begin{pmatrix} 1/3 & 2/3 & 0 \\ 0 & 1/3 & 2/3 \\ 2/3 & 0 & 1/3 \end{pmatrix} s \right\| = 1.9147 \neq ||s|| = \sqrt{5}, \tag{26}$$

where $s = (0, 1, 2)^T$.

If $\Pi(A, b, \trianglelefteq)$ have a group structure under the matrix multiplication, the above upper bound can be further simplified as

$$P_{ML} \leq \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq) \backslash \{I\}} Q\left(\frac{||\tilde{X}s - s||}{2\sigma}\right). \tag{27}$$

It should be remarked that the second upper bound (27) is independent of the transmitted codeword. In order to prove the bound (27), it is sufficient to prove $\Pi(A, b, \trianglelefteq)$ is distance invariant with respect to the Euclidean distance.

In the following, the distance invariant property of $\Pi(A, b, \trianglelefteq)$ will be shown. Let us define the Euclidean distance enumerator by

$$W_X(Z) \triangleq \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq)} Z^{||Xs - \tilde{X}s||}. \tag{28}$$

This enumerator has the information on distance distributions measured from the permutation matrix X.

The next lemma states that the Euclidean distance enumerator does not depend on the center point X if the linearly constrained permutation matrices have a group structure. This property can be regarded as a *distance invariance property* of permutation codes.

Lemma 2 (Distance invariance): If $\Pi(A, b, \trianglelefteq)$ forms a group under the matrix multiplication over $\mathbb{R}$, the equality

$$W_X(Z) = W(Z) \tag{29}$$

holds for any $X \in \Pi(A, b, \trianglelefteq)$. The weight enumerator $W(Z)$ is defined by $W(Z) = W_I(Z)$ where I is the $n \times n$ identity matrix.

Proof: Since $\Pi(A, b, \trianglelefteq)$ forms a group, the inverse X^{-1} belongs to $\Pi(A, b, \trianglelefteq)$ as well. Since the inverse X^{-1} induces a symbol-wise permutation, it is evident that

$$||Xs - \tilde{X}s|| = ||X^{-1}Xs - X^{-1}\tilde{X}s|| = ||s - X^{-1}\tilde{X}s|| \tag{30}$$

holds for any $X, \tilde{X} \in \Pi(A, b, \trianglelefteq)(X \neq \tilde{X})$. The Euclidean distance enumerator can be rewritten as

$$
\begin{aligned}
W_X(Z) &= \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq)} Z^{||Xs - \tilde{X}s||} \\
&= \sum_{\tilde{X} \in \Pi(A,b,\trianglelefteq)} Z^{||s - X^{-1}\tilde{X}s||} \\
&= \sum_{X' \in \Pi(A,b,\trianglelefteq)} Z^{||s - X's||} = W(Z).
\end{aligned} \tag{31}
$$

The second equality is a consequence of Eq. (30). The last equality is due to the assumption that $\Pi(A, b, \trianglelefteq)$ forms a group. ∎

Example 4: We have performed the following computer experiment for the following two codes:

1) LP decodable permutation code corresponding to the derangements of length 5. The additional linear constraint is $\text{trace}(X) = 0$. A transmitted word $(1, 0, 4, 2, 3)^T$ is assumed. The code polytope has 44-vertices which are all integral vertices.

2) LP decodable permutation code of length 5 corresponding to an additional linear constraint $X_{1,1} + X_{5,5} = 1$. A transmitted word $(0, 4, 3, 2, 1)^T$ is assumed. The code polytope has 330-vertices. The set of vertices contains 36-integral vertices and 294-fractional vertices.

The AWGN channel with noise variance σ^2 is assumed. The signal-to-noise ratio is defined by $SNR = 10 \log_{10}(1/\sigma^2)$. The LP decoding described in the previous section was employed for decoding.

Figure 2 presents the upper bounds and simulation results on block error probability of these permutation code. It is readily observed that the upper bounds presented in this section shows reasonable agreement with the simulation results.

The both codes have the same minimum pseudo distance 0.707107 and similar cardinalities (44 and 36) but the derangement code provides much better block error probabilities than those of the code with the constraint $X_{1,1} + X_{5,5} = 1$. This is because the existence of fractional vertices (i.e., 294-fractional vertices) severely degrades the decoding performance of the code with the constraint $X_{1,1} + X_{5,5} = 1$ compared with the derangement code. $\qquad\square$
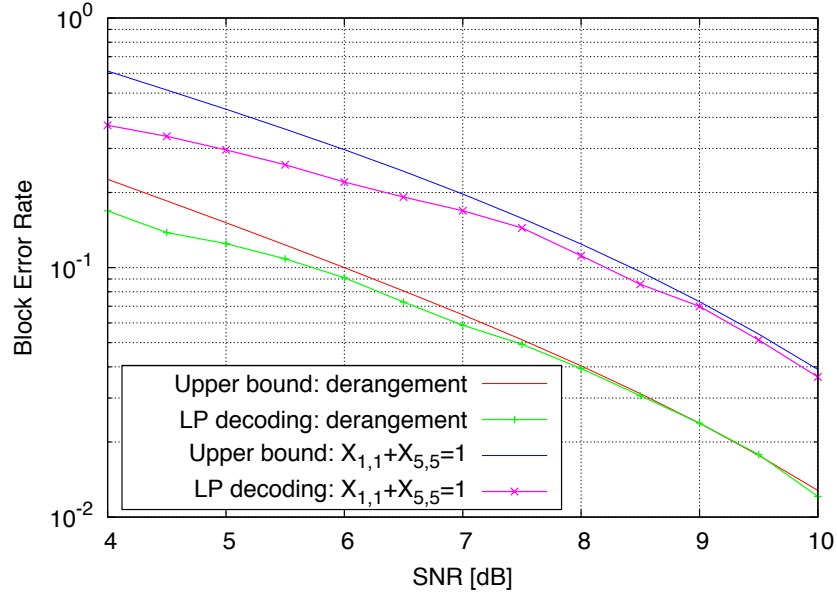

Fig. 2. Comparison of upper bounds and simulation results for LP decoding on block error probabilities ($n = 5$)

V. SOME CLASSES OF LINEARLY CONSTRAINED PERMUTATION CODES

In this section, we will discuss some classes linearly constrained permutation codes which are easy to encode.

A. Repetition permutation codes

Let η be a positive integer. Assume that a positive integer n is a multiple of η. The *repetition permutation codes* with repetition order η is defined by

$$\{((Ys_1)^T, (Ys_2)^T, \dots, (Ys_\eta)^T)^T \in \mathbb{R}^n \mid Y \in \Pi_{n/\eta}\}, \tag{32}$$

where $s_1, s_2, \dots, s_\eta \in \mathbb{R}^{n/\eta}$. We here assume that all the elements in $s_1, \dots, s_\eta$ are distinct each other. It is evident that the cardinality of the code is given by $(n/\eta)!$. The minimum Hamming distance of the code is 2η because the minimum Hamming distance of Ys_i is 2 for any $i \in [1, \eta]$.

It should be remarked that the repetition permutation code is a linearly constrained permutation code. The next example demonstrate linear constraints for the repetition permutation codes.

Example 5: Let

$$X = \begin{pmatrix} X_{1,1} & X_{1,2} & X_{1,3} & X_{1,4} \\ X_{2,1} & X_{2,2} & X_{2,3} & X_{2,4} \\ X_{3,1} & X_{3,2} & X_{3,3} & X_{3,4} \\ X_{4,1} & X_{4,2} & X_{4,3} & X_{4,4} \end{pmatrix}.$$

The permutation matrices in Π_4 satisfying the following set of linear constraints

$$X_{1,3} = X_{1,4} = X_{2,3} = X_{2,4} = 0 \tag{33}$$

$$X_{3,1} = X_{3,2} = X_{4,1} = X_{4,2} = 0 \tag{34}$$

$$X_{1,1} = X_{3,3}, \ X_{1,2} = X_{3,4} \tag{35}$$

$$X_{2,1} = X_{4,3}, \ X_{2,2} = X_{4,4} \tag{36}$$

defines the repetition permutation code of length 4 with repetition order 2. $\qquad \square$

B. Cartesian product codes

Suppose that η is a positive number and that n is positive multiple of η. A set of permutation matrices $U \subset \Pi_{n/\eta}$ is assumed to be given. The *cartesian product codes* is defined by

$$\{((Y_1 s_1)^T, (Y_2 s_2)^T, \ldots, (Y_\eta s_\eta)^T)^T \in \mathbb{R}^n \mid Y_1, \ldots, Y_\eta \in U\}, \tag{37}$$

where $s_1, s_2, \ldots, s_\eta \in \mathbb{R}^{n/\eta}$. The cardinality of cartesian product codes is thus given by $|U|^\eta$ if all the elements in $s_1, \ldots, s_\eta$ are distinct each other. Note that the class of cartesian product codes can be defined based on a set of linear constraints as well if U is defined by linear constraints.

C. Pure involution codes

In this subsection, we focus on the set of pure involutions, which produces a non-trivial class of permutation codes. It will be shown that the class of the permutation codes defined based on the pure involutions possess several good properties. This class of code can be encoded with an efficient greedy encoding algorithm. The cardinality of the code is much larger than the repetition code with the same length and the same minimum Hamming distance.

An *involution* is a permutation which coincides with its inverse permutation. Namely, the necessary and sufficient condition for a permutation matrix $X \in \Pi_n$ to be an involution is $X = X^T$ because the inverse matrix of a permutation matrix is the transposition of it. A *pure involution* is an involution without fixed point; i.e., a permutation matrix $X \in \Pi_n$ is said to be a pure involution if and only if $X = X^T$ and $\text{trace}(X) = 0$. In other words, the set of pure involutions is the intersection of the set of involutions and the set of derangements.

A pure involution exists when n is a positive even number. The reason is as follows. The lower triangle below the diagonal of X and the upper triangle above the diagonal must have the same number of ones since $X = X^T$. This implies that the number of ones in X should be even since the diagonal is constrained to be zero. A permutation matrix $X \in \Pi_n$ contains n-ones. Thus, if n is odd, it is clear that no permutation matrix meets the constraints. Throughout this subsection, we assume that n is an even positive number.

Let

$$\Omega_n \triangleq \{X \in \Pi_n \mid X = X^T, \text{trace}(X) = 0\}.$$

It is known that the cardinality of the pure involutions is given by

$$|\Omega_n| = (n-1)(n-3) \times \cdots \times 3 \times 1 = \frac{n!}{2^{n/2}(n/2)!}. \tag{38}$$

The linearly constrained permutation codes defined based on the constraints $X = X^T$, $\mathrm{trace}(X) = 0$ is called the *pure involution codes*. The triple for the pure involution codes are given by

$$
A = \begin{pmatrix} \mathrm{vec}(I_n) \\ \mathrm{vec}\left(F^{(2,1)}\right) \\ \mathrm{vec}\left(F^{(3,1)}\right) \\ \vdots \\ \mathrm{vec}\left(F^{(n,n/2-1)}\right) \end{pmatrix}, \quad b = \mathbf{0}, \quad \trianglelefteq = (=,\dots,=)^T, \tag{39}
$$

where $F^{(i,j)} \in \{0,1\}^{n \times n}$ is the binary matrix defined by

$$
F_{a,b}^{(i,j)} = \begin{cases} 1, & (a,b) = (i,j) \\ -1, & (a,b) = (j,i) \\ 0, & \text{otherwise.} \end{cases}
$$

1) Greedy encoding algorithm for pure involutions: A significant advantage of the pure involutions is that there exists an efficient encoding algorithm. The procedure EncMap shown below can be considered as a greedy algorithm for a constraint satisfaction problem without a back-tracking process.

EncMap

* Input: $m \in [1, (n-1) \times (n-3) \cdots 3 \times 1]$ (message)

* Output: $X \in \Omega_n$ (pure involution)

1) $m := m - 1$;

2) for $(p := 0;\ p < n/2;\ p := p + 1)$ {

3) $a_p := [m \bmod (2p+1)] + 1$;

4) $m := m \ \mathrm{div}\ (2p+1)$;

5) }

6) $\forall i, j \in [1, n],\ X_{i,j} := 0$;

7) $\forall i, j \in [1, n](i \neq j),\ Z_{i,j} := 1;\ \forall i \in [1, n],\ Z_{i,i} := 0$;

8) for $(p := n/2 - 1;\ p \geq 0;\ p := p - 1)$ {

9) $j := \arg\min\{j' \in [1, n] : \sum_{i'=1}^{n} Z_{i',j'} > 0\}$;

10) $i := \arg\min\left\{k \in [1, n] : \sum_{i'=1}^{k} Z_{i',j} = a_p\right\}$;

11) $X_{i,j} := 1;\ X_{j,i} := 1$;

12) $\forall q \in [1, n],\ Z_{q,j} := 0,\ Z_{j,q} := 0,\ Z_{i,q} := 0,\ Z_{q,i} := 0$;

13) }

14) Output X;

The arithmetic operation in the line 4 represents the division for integers; i.e., $5\ \mathrm{div}\ 2 = 2$. There are some remarks on EncMap. The part from the line 1 to 5 converts a message integer into an $n/2$-tuples:

$$
(a_0, a_1, \dots, a_{n/2-1}) \in [1,1] \times [1,3] \times \cdots \times [1, n-1].
$$

The remaining part generates a pure involution according to the n-tuple $(a_0, a_1, \dots, a_{n/2-1})$.

The variables $Z_{i,j}$ represents whether $X_{i,j}$ is determined ($Z_{i,j} = 0$) or not ($Z_{i,j} = 1$). On the diagonal elements of $Z_{i,j}$ are initialized to be zero which means that the diagonal elements of $X_{i,j}$ is determined to be zero.

The generation of a pure involution is performed in a greedy manner from the left columns to the right columns. The undetermined column with the smallest index is found in the line 9. In the line 10, the row index of a_p-th undetermined element is assigned to i. In the line 11, two ones are written at (i, j) and (j, i)-positions of X and the line 12 fixes the cross regions around (i, j) and (j, i).

In an encoding process, for any $p = N - t(t \in [1, N])$,

$$\sum_{i' \in [1,n]}^{n} Z_{i',j} = (n - 1) - 2(t - 1) = 2p + 1 \tag{40}$$

holds at the line 10. This is because exactly two-columns and two-rows of Z are set to zero for each iteration due to the constraints of the pure involution. In other words, the numbers of zero columns and zero rows are increased by two after an iteration. This property guarantees that

$$\sum_{i' \in [1,n]}^{n} Z_{i',j} \geq a_p \tag{41}$$

holds for all $p \in [0, N - 1]$. Therefore, for any input m, the line 10 can find an index i satisfying

$$i = \arg\min \left\{ k \in [1, n] : \sum_{i'=1}^{k} Z_{i',j} = a_p \right\}.$$

The loop from the line 2 to 5 takes $O(n)$-time under the assumption that the basic big-number arithmetics can be done within a unit time. The initialization process (lines 6 and 7) requires $O(n^2)$-time. The most time consuming part of EncMap is the loop from the line 8 to 13. In order to find i, j in lines 9 and 10, $O(n)$-times requires. The process in line 12 also needs $O(n)$-time to carry it out. Therefore, the time complexity of the loop (from the line 8 to 13.) is $O(n^2)$, which dominates the time complexity of EncMap.

From the definition shown above, it is evident that EncMap gives a injection map from $[1, (n - 1) \times (n - 3) \cdots 3 \times 1]$ to Ω_n. Since the cardinality of Ω_n is $(n - 1) \times (n - 3) \cdots 3 \times 1$, we can see that EncMap is a bijection.
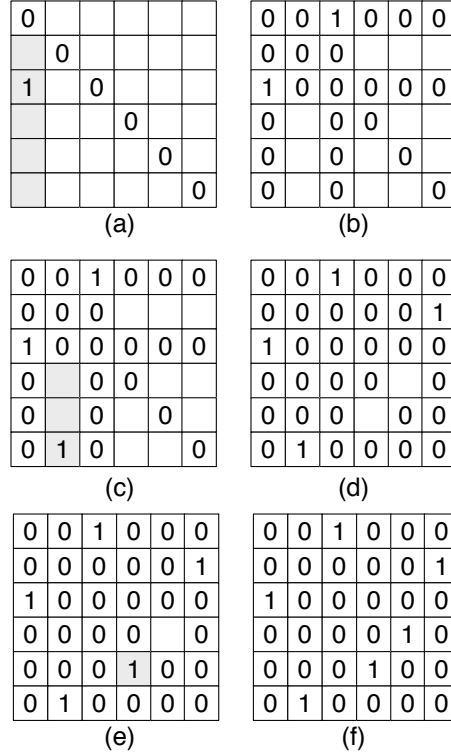
There is an inverse map of EncMap from Ω_n to $[1, (n-1) \times (n-3) \cdots 3 \times 1]$ because EncMap is a bijection. The procedure DecMap gives the inverse map of EncMap.

DecMap

* Input: $X \in \Omega_n$ (pure involution)

* Output: $m \in [1, (n - 1) \times (n - 3) \cdots 3 \times 1]$ (message)

1) $\forall i, j \in [1, n](i \neq j), Z_{i,j} := 1; \forall i \in [1, n], Z_{i,i} := 0;$

2) for $(p := n/2 - 1; p > 0; p := p - 1)$ {

3) $j := \arg\min \{ j' \in [1, n] : \sum_{i'=1}^{n} Z_{i',j'} > 0 \};$

4) $i := \sum_{i' \in [1,n]} i' \mathbb{I}[X_{i',j} = 1];$

5) $a_p := \sum_{i'=1}^{i} Z_{i',j};$

6) $\forall q \in [1, n], Z_{q,j} := 0, Z_{j,q} := 0, Z_{i,q} := 0, Z_{q,i} := 0;$

7) }

8) $m := 0;$

9) for $(p := n/2 - 1; p \geq 1; p := p - 1)$ {

10) $m := (2p + 1)m + (a_p - 1);$

11) }

12) $m := m + 1;$

13) Output m;

Example 6: An encoding process of a pure involution matrix is illustrated in Fig.3. In this example, $n = 6$ is assumed. The status of $X_{i,j}$ and $Z_{i,j}$ are depicted by 6×6 cells in Fig.3. Namely, $Z_{i,j} = 1$ (undetermined state) represents an empty cell. A cell with label 0 (resp. 1) represents $(X_{i,j}, Z_{i,j}) = (0,0)$ (resp. $(X_{i,j}, Z_{i,j}) = (1,0)$). At first, the diagonal cells are set to be zero because of the constraint $\text{trace}(X) = 0$. The message is assumed to be $m = 5$. In this case, we have $a_0 = 1, a_1 = 3, a_2 = 2$. The shaded cells in Fig.3 (a) represents possible places to write the symbol 1. According to the part of the message $a_2 = 2$, the second shaded cell is determined to be 1. In Fig.3 (b), the symbol 1 is written on the symmetric position and zeros are placed in the columns and rows corresponding to two 1's. In a similar way (Fig.3 (b)–(e)), the empty cells are filled with 0 or 1. As a result, we have a pure involution matrix (Fig.3 (f)).

The shaded cells represent are possible places to write the symbol 1. In (a) and (b), there are 5 and 3-shaded cells, respectively. This means that $5 \times 3 = 15$ pure involution matrices exist when $n = 6$.

Fig. 3.   An Encoding process of a pure involution matrix

2) Minimum Hamming distance of pure involution codes: Let $s \in \mathbb{R}^n$ be an initial vector whose components are distinct each other. It is well known that the minimum Hamming distance of $\Lambda(s)$ is given by

$$\min_{X,X' \in \Pi_n (X \neq X')} d_H(Xs, X's) = 2. \tag{42}$$

The minimum Hamming distance of the pure involution codes is larger than that of $\Lambda(s)$.

Lemma 3 (Minimum distance): The minimum Hamming distance of the pure involution codes are given by

$$\min_{X,X' \in \Omega_n (X \neq X')} d_H(Xs, X's) = 4. \tag{43}$$

Proof: Assume that $X, X' \in \Omega_n (X \neq X')$. Since $X \neq X'$, there is an index pair $(i, j) \in [1, n]^2$ satisfying $X_{i,j} \neq X'_{i,j}$. Without loss of generality, we assume that $X_{i,j} = 1$ and $X'_{i,j} = 0$. An index $l \in [1, n]$ satisfying $X_{i,l} \neq X'_{i,l}$ must exist because X and X' are permutation matrices. Due to the assumption $X_{i,j} = 1$ and $X'_{i,j} = 0$, we have $X_{i,l} = 0$ and $X'_{i,l} = 1$. In a similar manner, there must be an index k satisfying $X_{k,j} = 0, X'_{k,j} = 1$. It is possible to continue this argument until a sequence of index pairs constitutes a loop.
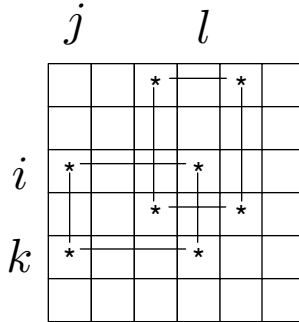
The set of the index pairs $\{(i, j) \in [1, n]^2 \mid X_{i,j} \neq X'_{i,j}\}$ is called a *difference position set*. The argument above implies that the difference position set needs to be partitioned into several loops of even length. A loop means a sequence of adjacent index pairs with the form $(i_1, i_2) \to (i_1, i_3) \to (i_4, i_3) \to \cdots \to (i_1, i_2)$. If $X_{i_1,i_2} = 1$ holds, then we have $X_{i_1,i_3} = 0, X_{i_4,i_3} = 1$ and so on. Therefore, the length of a loop should be even because a loop with odd length gives inconsistent assignment $X_{i_1,i_2} = 0$ at the end of the loop.

The shortest loop of even length have the form $(i, j) \to (i, l) \to (k, l) \to (k, j) \to (i, j)$. If the difference potion set includes this type of a loop of length 4, it must also contain another loop of length 4 with the form $(j, i) \to (l, i) \to (l, k) \to (j, k) \to (j, i)$ because $X = X^T$ holds for any $X, X' \in \Omega_n$ (See Fig.4). Let $a = Xs$ and $a' = X's$. If the difference position sets consist of only such two symmetric loops of length 4, we have

$$a_u \neq a'_u \quad \text{iff } u \in \{i, j, k, l\}.$$

This implies that the smallest number of differences between Xs and $X's$ is 4. ∎

The proof of the above lemma indicates a way to enumerate the number of codewords at the minimum Hamming distance. For a fixed Xs, the number of codewords $X's$ satisfying $d_H(Xs, X's) = 4$ can be obtained by enumerating the number of allocations of two symmetric loops.

The left loop of length 4 represents $(i, j) \to (i, l) \to (k, l) \to (k, j) \to (i, j)$ and the right loop corresponds to $(j, i) \to (l, i) \to (l, k) \to (j, k) \to (j, i)$. Note that there are 4-columns which include elements of the difference position set. These columns correspond to the positions on which the symbols of Xs and $X's$ differ.

Fig. 4. Two symmetric loops of length 4 in a difference position set.

We have seen that the repetition code of repetition order 2 yields the minimum Hamming distance 4. When the length of the code is n (even), the number of codewords is given by $(n/2)!$. On the other hand, the pure involution code provides the same minimum Hamming distance and the cardinality of the code is given by $n!/(2^{n/2}(n/2)!)$, which is much larger than $(n/2)!$ because

$$\frac{n!/(2^{n/2}(n/2)!)}{(n/2)!} = \binom{n}{n/2} 2^{-n/2} \simeq \frac{1}{\sqrt{\pi n/2}} 2^{n/2}. \tag{44}$$

For example, consider the case where $n = 64$. In this case, the number of codewords of the repetition code is $(n/2)! \simeq 2^{118}$. On the other hand, the pure involution code have

$$\frac{n!}{2^{n/2}(n/2)!} \simeq 2^{146}$$

codewords which is approximately 2^{28}-times larger than that of the repetition code.

3) Code polytope of pure involutions: The linear constraint $X = X^T$ and $\texttt{trace}(X) = 0$ for pure involutions defines a code polytope which is not an integral polytope.

Example 7: Assume that $n = 6$. The code polytope defined based on the constraints $X = X^T$ and $\texttt{trace}(X) = 0$ have 15 integral vertices and 10 fractional vertices. A fractional vertex is

$$\begin{pmatrix} 0 & 1/2 & 0 & 0 & 0 & 1/2 \\ 1/2 & 0 & 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Deriving inequality description of the convex hull of pure involution matrices is an interesting open problem.

4) Simulation results: The minimum Hamming distance of a permutation code is a universal measure for goodness of a code because it does not depend on the choice of the initial vector s. However, as we have seen in the previous section, decoding performance is mostly determined by the pseudo distance distribution of a code polytope.

In order to evaluate the decoding performance of pure involution codes, we have performed a computer experiment. Figure 5 presents the block error probability of the pure involution codes with length 64. In this experiment, the initial vector is assumed to be $s = (1, 2, \ldots, 64)$ and the LP decoding was used. The definition of the SNR is the same as in Example 4. For comparison purpose, the block error probabilities of the repetition permutation code of length 64 with the repetition order 2 and uncoded permutations vectors (i.e., $\Lambda(s)$) of length 64 are also plotted in Fig. 5. It can be observed that the pure involution code gives much small block probabilities compared with the repetition code. As we have seen in the previous section, the cardinality of a pure involution code is much larger than that of the repetition code. We may be able to conclude that the pure involution code is superior to the repetition code.

D. Block permutation codes

A block permutation codes are defined based on the block permutation matrices. The block structure is useful for encoding and evaluation of the minimum squared Euclidean distance.

1) Definitions: Suppose the situation where the set $[1, n] \times [1, n]$ is divided into mutually disjoint $\gamma \times \gamma$ square blocks of size $\nu \times \nu$ (i.e., $n = \gamma\nu$ holds). The square blocks are called *blocks* which is explicitly defined as follows.

Definition 7 (Block): For $k, b \in [1, \gamma]$, a *block* $B_{k,b}$ is defined by

$$B_{k,b} \triangleq \{(i, j) \in [1, n]^2 \mid \nu(k-1) < i \leq \nu k, \nu(b-1) < j \leq \nu b\}. \tag{45}$$

The indices k and b are called *block indices*. □

The rectangle region $T_{k,b}^{(l)}$ is defined as

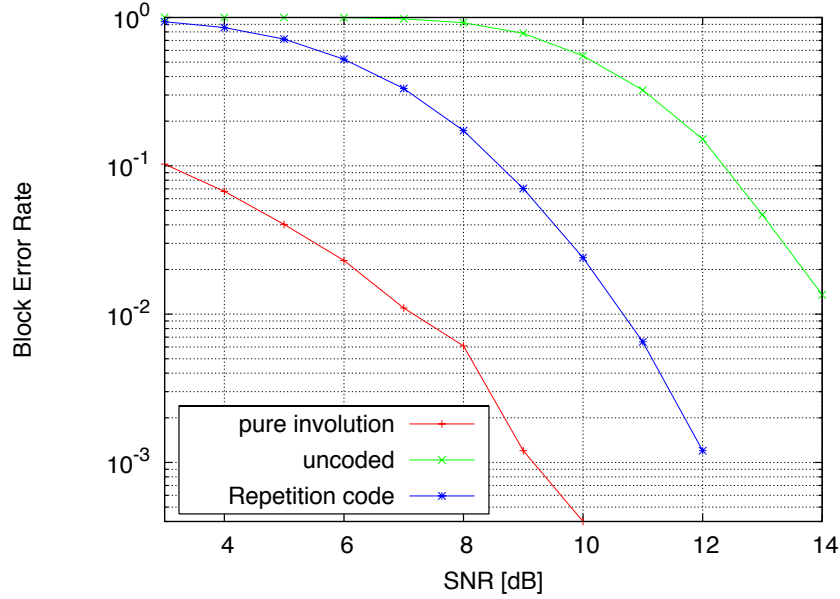$$T_{k,b}^{(l)} \triangleq \{(x, y) \in B_{k,b} \mid y = \nu(b-1) + l\} \tag{46}$$

Fig. 5. Comparison of block error probabilities: pure involution codes, repetition permutation codes, and uncoded permutation vectors of length 64

for $k, b \in [1, \gamma]$ and $l \in [1, \nu]$. The subscript k, b specifies the block where the rectangle region $T_{k,b}^{(l)}$ belongs to. The superscript $l \in [1, \nu]$, which is called a *subindex*, indicates the relative position in the block $B_{k,b}$.

We are now ready to define a block permutation matrix which is the basis for realizing a block-wise permutation group.

Definition 8 (Block permutation matrix): Assume that a permutation matrix $X \in \Pi_n$ is given. If, for any $b \in [1, \gamma]$, there exists the unique block index k satisfying

$$X(B_{k,b}) \neq 0 \tag{47}$$

then X is called a *block permutation matrix*. The notation $X(B_{k,b})$ represents the sub-matrix of X corresponding to the block $B_{k,b}$. □

From this definition, it is apparent that a nonzero $X(B_{k,b}) \in \{0, 1\}^{\nu \times \nu}$ is a permutation matrix if X is a block permutation matrix. Furthermore, there exists the unique block index b satisfying $X(B_{k,b}) \neq 0$ for any block index $k \in [1, \gamma]$. This equivalent statement can be obtained by exchanging the role of column and row in the above definition.

For block indices $k, b \in [1, \gamma]$ and subindex $l \in [1, \nu]$, the *skewed column set* is defined by

$$U_{k,b}^{(l)} \triangleq T_{k,b}^{(l)} \cup \left(\bigcup_{k' \in [1,\gamma] \setminus \{k\}} T_{k',b}^{(l \bmod \nu)+1} \right). \tag{48}$$

Figure 6 illustrates the subsets of $[1, n] \times [1, n]$ appeared so far such as the blocks, the rectangle regions, and the skewed column set.

2) Block permutation codes: The next theorem presents a set of linear constraints characterizing block permutation matrices.

Theorem 3 (Characterization of block permutation matrix): Let $X \in \Pi_n$ be a permutation matrix. The permutation matirx X is a block permutation matrix if and only if

$$\sum_{(u,v) \in U_{k,b}^{(l)}} X_{u,v} = 1 \tag{49}$$
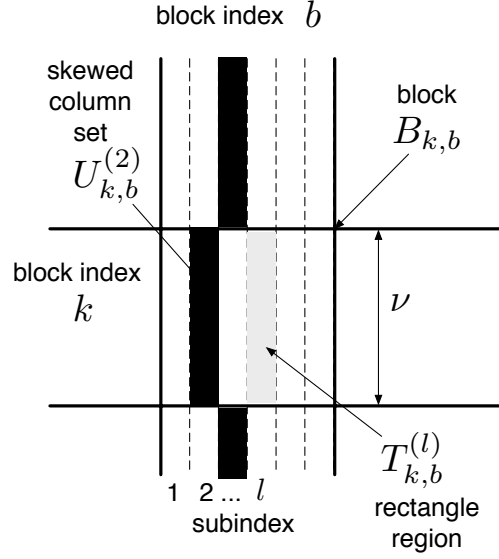
holds for any $b, k \in [1, \gamma], l \in [1, \nu]$.

Fig. 6.   Blocks, rectangle regions and skewed column set

The next example clarifies the linear constraints characterizing a 4×4 block permutation matrix.

Example 8: Let $n = 4, \nu = 2, \gamma = 2$. The necessary and sufficient condition for a permutation matrix $X \in \Pi_4$ being a block permutation matrix are as follows:

$$
\begin{aligned}
X_{1,1} + X_{2,1} + X_{3,2} + X_{4,2} &= 1 \\
X_{1,2} + X_{2,2} + X_{3,1} + X_{4,1} &= 1 \\
X_{1,3} + X_{2,3} + X_{3,4} + X_{4,4} &= 1 \\
X_{1,4} + X_{2,4} + X_{3,3} + X_{4,3} &= 1.
\end{aligned}
$$

$$(50)$$

Let us denote the set of block permutation matrices by

$$
\Pi(n, \nu) \overset{\triangle}{=} \{X \in \Pi_n \mid X \text{ satisfies } (49)\}. \tag{51}
$$

Note that we here employ a lighter notation $\Pi(n, \nu)$ instead of $\Pi(A, b, \trianglelefteq)$ since it explicitly express dependency on n and ν. It should be remarked that $\Pi(n, \nu)$ forms a group under matrix multiplication over $\mathbb{R}$.

The class of block permutation codes defined below is a class of LP decodable permutation codes.

Definition 9 (Block permutation code): Let n be a positive integer. A positive integer ν is a divisor of n. The initial vector s belongs to $\mathbb{R}^n$. The *block permutation code* $C(n, \nu, s)$ is defined by

$$
C(n, \nu, s) \overset{\triangle}{=} \{Xs \in \mathbb{R}^n : X \in \Pi(n, \nu)\}. \tag{52}
$$

$\square$

In Section IV, we saw the minimum pseudo distance is one of most influential parameters for LP decoding performance. Unfortunately, the evaluation of the minimum pseudo distance is not a trivial problem. As a possible alternative, we here evaluate the minimum squared Euclidean distance of $C(n, \nu, s)$ defined by

$$
d_{min}^2 \overset{\triangle}{=} \min_{x,y \in C(n,\nu,s)(x \neq y)} ||x - y||^2. \tag{53}
$$

At least, we can say that decoding performance degrades even with an ML decoder if $C(n, \nu, s)$ has small d^2_{min}.

The block-wise permutation structure of a block permutation code can be exploited for deriving a simple formula on the minimum squared Euclidean distance.

Let us define Δ^2_1 and Δ^2_2 by

$$
\begin{aligned}
\Delta^2_1 &= \min_{k \in [1,\gamma]} \min_{Q \in \Pi_\nu (Q \neq I)} ||s_k - Qs_k||^2 \\
\Delta^2_2 &= \min_{k,j \in [1,\gamma](k \neq j)} \min_{Q \in \Pi_\nu} ||s_k - Qs_j||^2.
\end{aligned}
\tag{54}
$$

Assume that both Δ^2_1 and Δ^2_2 are positive for given n, ν, s. In such a case, $C(n, \nu, s)$ is non-singular and it is easily proved that the minimum squared Euclidean distance of $C(n, \nu, s)$ is given by

$$
d^2_{min} = \min\{\Delta^2_1, 2\Delta^2_2\}.
\tag{55}
$$

The following example illustrates that a block permutation code can have more codewords than those of a trivial cartesian product code under the condition that both of codes have the same minimum squared Euclidean distance.

Example 9: Let $n = 8, \gamma = 2, \nu = 4$. The initial vector $s = (s_1^T, s_2^T)^T$ is assumed to be

$$
s_1 = \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \end{pmatrix}, \quad s_2 = \begin{pmatrix} 2 \\ 4 \\ 6 \\ 8 \end{pmatrix}
$$

From the definition of Δ^2_1, Δ^2_2, we easily obtain $\Delta^2_1 = 8, \quad \Delta^2_2 = 4$. From (55), we have $d^2_{min} = \min\{8, 2 \times 4\} = 8$. The number of codewords is $\gamma! \times (\nu!)^\gamma = 1152$. The cartesian product code defined by

$$
\{((Y_1\ s_1)^T, (Y_2\ s_2)^T)^T \in \mathbb{R}^{64} \mid Y_1, Y_2 \in \Pi_4\},
$$

has also squared Euclidean distance 8 but it contains 576-codewords, which is half of the number of codewords of the block permutation code. □

VI. RANDOMLY CONSTRAINED PERMUTATION MATRICES

In the previous section, we discussed a set of structured permutation matrices. Another possible choice for linear constraints is to generate them randomly. Such random linear constraints are amenable for probabilistic analysis and appears interesting from information theoretic view. In this section, we study a class of LP decodable permutation codes defined based on random constraints.

A. Sparse constraint matrix ensemble

Since the LP decodable permutation codes are non-linear codes, the cardinality of a given code cannot be determined directly from the constraints in general. In the following part of this section, we will analyze the cardinality of codes and their Hamming weight distributions.

A sparse constraint matrix ensemble is assumed in the following analysis, which has a close relationship to the analysis on average weight distribution of LDPC ensembles [12].

The linear constraint assumed here is the equality constraint for two variables such as $X_{i,j} = X_{k,l}$. As discussed in Section X, linearly constrained permutation matrices defined based on this equality constraint is important because such matrices can be used as building blocks of a generalized block permutation code.

Let S be the set of binary constraint matrices:

$$S \triangleq \{A \in \{0,1\}^{m \times n^2} : \text{every row of } A \text{ contains 2-ones}\}. \tag{56}$$

We assign the uniform probability

$$P(A) \triangleq \frac{1}{\binom{n^2}{2}^m} \tag{57}$$

to each matrix in S. The pair (S, P) can be considered as an ensemble of matrices, which becomes the basis of the following probabilistic method.

Assume that $\theta : S \to \{-1, 0, 1\}^{m \times n^2}$ is defined by $B = \theta(A)$, where

$$B_{i,j} = \begin{cases} -A_{i,j}, & \text{if } \forall j' \in [1, j-1], A_{i,j'} = 0, \\ A_{i,j}, & \text{otherwise.} \end{cases} \tag{58}$$

Note that $\theta(A)\mathsf{vec}(X) = 0$ corresponds to m equality constraints of two variables.

In this section, we focus on the LP decodable permutation code $\Lambda(\theta(A), 0, \trianglelefteq, s)$, where $A \in S$ and $\trianglelefteq \triangleq (\overbrace{=, = \ldots, =}^{m})^T$. The symbol $\mathbf{1}$ denotes the vector of length m whose entries are all ones. Extensions of the analysis for more general classes of LP decodable permutation codes are possible, but we here focus on the simplest class to explain the idea of the analysis. Throughout this section, we assume that components of the initial vector s differ each other.

B. Probabilistic analysis on average cardinality of codes

The number of codewords in $\Lambda(\theta(A), 0, \trianglelefteq, s)$ is given by

$$M(A) \triangleq \sum_{X \in \Pi_n} \mathbb{I}[\theta(A)\,\mathsf{vec}(X) \trianglelefteq 0], \tag{59}$$

where $\mathbb{I}$ is the indicator function. The indicator function takes the value one when the given condition is true and otherwise gives the value zero. The next lemma gives the average cardinality of this code.

Lemma 4 (Average cardinality of codes): The average cardinality of $\Lambda(\theta(A), 0, \trianglelefteq, s)$ is given by

$$\mathsf{E}[M(A)] = n! \left(\frac{\binom{n}{2} + \binom{n^2 - n}{2}}{\binom{n^2}{2}} \right)^m, \tag{60}$$

where the operator E denotes the expectation defined on (S, P).

Proof: From the definition of $M(A)$, the expectation of the cardinality $M(A)$ can be written as

$$\begin{aligned} \mathsf{E}[M(A)] &= \sum_{A \in S} P(A)M(A) \\ &= \sum_{A \in S} P(A) \sum_{X \in \Pi_n} \mathbb{I}[\theta(A)\,\mathsf{vec}(X) \trianglelefteq 0]. \end{aligned} \tag{61}$$

By changing the order of summation, we can further transform this into

$$\begin{aligned} \mathsf{E}[M(A)] &= \sum_{X \in \Pi_n} \sum_{A \in S} P(A)\mathbb{I}[\theta(A)\,\mathsf{vec}(X) \trianglelefteq 0] \\ &= \frac{n!}{\binom{n^2}{r}^m} \sum_{A \in S} \mathbb{I}[\theta(A)\,\mathsf{vec}(X') \trianglelefteq 0], \end{aligned} \tag{62}$$

where X' is an arbitrary permutation matrix in Π_n. The last equality is due to the symmetry of the ensemble. Namely, this means that the quantity $\sum_{A \in S} \mathbb{I}[\theta(A)\,\mathsf{vec}(X') \trianglelefteq 0]$ does not depend on the choice of X'. The evaluation of $\sum_{A \in S} \mathbb{I}[\theta(A)\,\mathsf{vec}(X') \trianglelefteq 0]$ can be performed on the basis of the following combinatorial argument.

It is evident that any $X' \in \Pi_n$ contains n-ones as its components. This implies that $x' \triangleq \text{vec}(X')$ is a binary vector of length n^2 with Hamming weight n. Let $I_1 \triangleq \{i \in [1, n^2] \mid x_i' = 1\}$, where x_i' is the ith element of x'. Consider the first row of A, which is denoted by a^T. The relation $\theta(a^T)x' = 0$ holds if and only if

$$|\{i \in I_1 \mid a_i = 1\}| = 2 \text{ or } |\{i \in [1, n^2]\backslash I_1 \mid a_i = 1\}| = 2. \tag{63}$$

The number of possible ways to choose such a vector a is given by

$$\binom{n}{2} + \binom{n^2 - n}{2}. \tag{64}$$

The term $\binom{n}{2}$ corresponds to the number of possible ways such that I_1 (of cardinality n) contains 2-ones. On the other hand, $\binom{n^2-n}{2}$ represents the number of possible ways that remaining parts contains 2-ones. Since each row of A can be chosen independently, we consequently have

$$\sum_{A \in S} I[\theta(A) \, \text{vec}(X') \trianglelefteq 0] = \left(\binom{n}{2} + \binom{n^2 - n}{2} \right)^m. \tag{65}$$

Substituting (65) into (62), we immediately obtain the claim of the lemma. ∎

Example 10: In this experiment, the number of 10×10 permutation matrices satisfying randomly generated equality constraints of two variables was counted. Figure 7 plots the cardinality of 100-samples for the cases where $m = 30, 40, 50$. The figure includes the ensemble average of the cardinality given by (60) and the sample mean of the cardinality. The figure shows that cardinalities are scattered around the ensemble average and that the sample mean agree with the ensemble average with reasonable accuracy.

This figure shows a trade-off relation between the number of additional equalities m and the cardinality. As (60) indicates, the average cardinality is an exponentially decreasing function of m. □
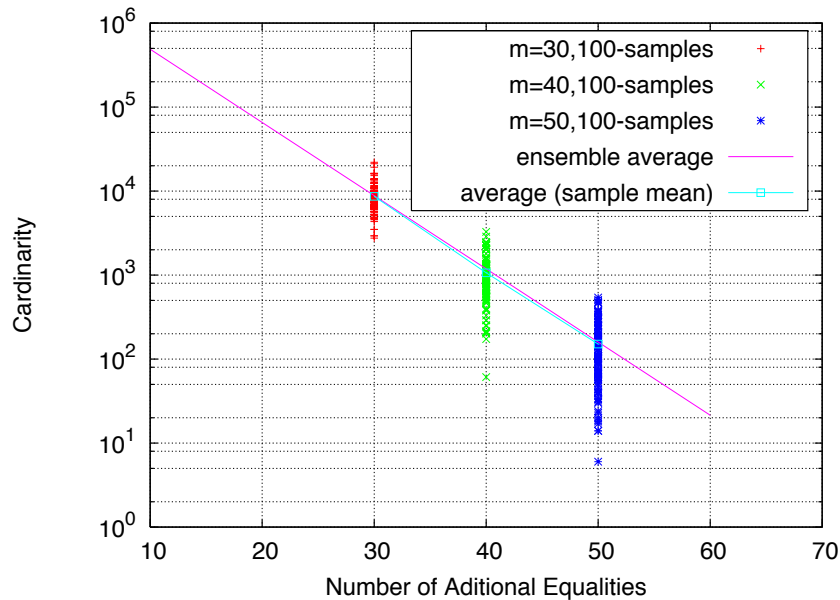

Fig. 7. Relation between additional equalities m and average cardinality

C. Probabilistic analysis on weight distribution

The origin $o \triangleq (o_1, \ldots, o_n)$ is an arbitrary permutation vector of length n; namely, $o \in \Lambda(s)$. The number of codewords of $\Lambda(\theta(A), 0, \trianglelefteq, s)$ with Hamming weight w is denoted by $L_w(A)$, where the Hamming weight $w_H(\cdot)$ is defined by

$$w_H(x) \triangleq \sum_{i=1}^n \mathbb{I}[o_i \neq x_i], \tag{66}$$

where $x = (x_1, \ldots, x_n)$. This means the Hamming weight of x is equal to the Hamming distance between the origin and x. In other words, $L_w(A)$ is defined as

$$L_w(A) \triangleq \sum_{x \in \Lambda(\theta(A), 0, \trianglelefteq, s)} \mathbb{I}[w_H(x) = w]. \tag{67}$$

The set $\{L_1(A), \ldots, L_n(A)\}$ is referred to as the weight distribution of $\Lambda(\theta(A), 0, \trianglelefteq, s)$.

The next lemma gives the ensemble average of the weight distribution.

Lemma 5: The average weight distribution of the linearly constrained permutation code $\Lambda(\theta(A), 0, \trianglelefteq, s)$ is given by

$$\mathsf{E}[L_w(A)] = \binom{n}{w} \left\lfloor \frac{w! + 1}{e} \right\rfloor \left(\frac{\binom{n}{2} + \binom{n^2 - n}{2}}{\binom{n^2}{2}} \right)^m. \tag{68}$$

Proof: The weight distribution $L_w(A)$ can also be expressed as

$$L_w(A) = \sum_{X \in Z_w(o)} \mathbb{I}[\theta(A) \, \mathsf{vec}(X) \trianglelefteq 0], \tag{69}$$

where $Z_w(o)$ is defined by

$$Z_w(o) \triangleq \{X \in \Pi_n : w_H(Xs) = w\}. \tag{70}$$

The expectation can be simplified as follows:

$$
\begin{aligned}
\mathsf{E}[L_w(A)] &= \sum_{A \in S} P(A) \sum_{X \in Z_w(o)} \mathbb{I}[\theta(A) \, \mathsf{vec}(X) \trianglelefteq 0] \\
&= \frac{1}{\binom{n^2}{r}^m} \sum_{X \in Z_w(o)} \sum_{A \in S} \mathbb{I}[\theta(A) \, \mathsf{vec}(X) \trianglelefteq 0] \\
&= \left(\frac{\binom{n}{2} + \binom{n^2 - n}{2}}{\binom{n^2}{2}} \right)^m |Z_w(o)|.
\end{aligned}
\tag{71}
$$

The last equality is due to the symmetry of the ensemble and (65).

The cardinality of $Z_w(o)$ is given by the following combinatorial argument. Let $x \in \Lambda(s)$ be an arbitrary vector satisfying $w_H(x) = w$. The index set I_{diff} is defined by $I_{diff}(x) \triangleq \{i \in [1, n] \mid o_i \neq x_i\}$. Let $T \subset [1, n]$ be an index set of cardinality w. The quantity $|\{x \in \Lambda(s) \mid T = I_{diff}(x)\}|$ is equal to the number of derangements of length w, which is known to be $\lfloor (w! + 1)/e \rfloor$ [33]. Note that the number of possible ways to choose T is $\binom{n}{w}$. Thus, we have the equality

$$|Z_w(o)| = \binom{n}{w} \left\lfloor \frac{w! + 1}{e} \right\rfloor. \tag{72}$$

This completes the proof of the lemma. ∎

Note that the origin assumed here may not be included in $\Lambda(\theta(A), 0, \trianglelefteq, s)$.

VII. Conclusion

In this paper, a novel class of permutation codes, LP decodable permutation codes, is introduced. The LP decodable property is the main feature of this class of permutation codes.

The set of doubly stochastic matrices, i.e., the Birkhoff polytope, have $n!$ integral vertices which are permutation matrices. Additional linear constraints defines a code polytope which plays a fundamental role in the coding scheme presented in this paper. An LP decodable permutation code is the set of integral vertices of a code polytope.

In an LP decoding process, a certain linear objective function is maximized under the assumption that the feasible set is a code polytope. The decoding performance can be evaluated from geometrical properties of a code polytope.

The choice of additional linear constraints are crucial to construct good codes. In this paper, two approaches are discussed; namely, structured permutation matrices and randomly constrained permutation matrices.

Section V introduces some classes of structured linearly permutation matrices. Especially, it has been shown that the pure involution codes have several nice properties; they are easy to encode and their error correction performance is much better than the trivial repetition code.

The random constraints discussed in Section VI enable us to use probabilistic methods for analyzing some properties of codes. The probabilistic methods [26] are very powerful tool for grasping the relation between the number of constraints and important code parameters such as the cardinality of a code.

Although the paper provides fundamental aspects of the LP decodable permutation codes, a number of problems remain still open. The following list is a part of open problems.

1) Construction of good block permutation codes including a choice of an initial vector
2) Efficient algorithm for solving the LP problem arising in the LP decoding.
3) Permutation modulation for linear vector channels; let H be a $n \times n$ real matrix. An ML decoding problem for a linear vector channel can be formulated as

$$\text{minimize } ||y - Hx||^2 \text{ subject to } x \in \Lambda(A, b, \unlhd, s). \tag{73}$$

As discussed in this paper, the decoding problem can be relaxed to a quadratic programming (QP) problem:

$$\text{minimize } ||y - Hx||^2 \text{ subject to } x \in \mathcal{P}(A, b, \unlhd, s). \tag{74}$$

A QP-based decoding algorithm like [31] appears interesting for this problem.

4) An application to rank modulation

Further investigation on related topics may open an interesting interdisciplinary research field among coding and combinatorial optimization.

Appendix

1) Code polytopes for some classes of linearly constrained permutation matrices: Table I presents linear constraints for some sets of permutation matrices and their integrality of corresponding code polytopes. In this table, it is assumed that $X \in \mathbb{R}^{4 \times 4}$. The integrality is numerically checked with the vertex enumeration program cdd based on double description method by K. Fukuda [32].

Some remarks on Table I are listed as follows.

TABLE I

CODE POLYTOPES AND ITS PROPERTIES ($n = 4$)

set of perm. matrices	additional constraints	integrality	$|V|$
cyclic perm. mat.	(75)	Y	4
derangement	$\text{trace}(X) = 0$	Y	9
involution	$X = X^T$	N	14
transposition (1)	$\text{trace}(X) = n - 2$	N	20
transposition (2)	$\text{trace}(X) = n - 2$	Y	6
	$X = X^T$		
2×2 block	constraints (50)	N	28
2×2 block	constraints (50) and (77)	Y	8

The column of integrality (Y/N) represents the code polytope is integral (Y) or not (N). The column $\#V$ denotes the number of vertices on the code polytope.

1) Cyclic permutation matrices The cyclic permutation matrices of order 4 is given by the following additional linear constraints:

$$X_{1,1} = X_{2,2}, \ X_{2,2} = X_{3,3}, \ X_{3,3} = X_{4,4}$$

$$X_{2,1} = X_{3,2}, \ X_{3,2} = X_{4,3}, \ X_{4,3} = X_{1,4}$$

$$X_{3,1} = X_{4,2}, \ X_{4,2} = X_{1,3}, \ X_{1,3} = X_{2,4}$$

$$X_{4,1} = X_{1,2}, \ X_{1,2} = X_{2,3}, \ X_{2,3} = X_{3,4}. \tag{75}$$

In a similar way as in the case $n = 4$, we can define the cyclic permutation matrices of order n. The general expression the constraint for $n \times n$ cyclic permutation matrices is given by

$$\forall i, j \in [1, n], \quad X_{i,j} = X_{(i \bmod n)+1, (j \bmod n)+1}. \tag{76}$$

2) Transposition: The permutation matrices satisfying the linear constraint $\text{trace}(X) = n - 2$ exactly coincides with the set of transpositions (i.e., permutations of two elements). Note that the constraint $\text{trace}(X) = n - 2$ does not give the tight polytope. Combining a redundant constraint $X = X^T$ (i.e., the involution constraint) to the trace constraint, the relaxed polytope becomes tight. This example indicates that redundant constraints are necessary for constructing a tight polytope in some cases.

3) Block constraint: The linear constraints for block permutation matrices (50) introduced in Theorem 3 does not give the tight polytope in $n = 4$. However, combining (50) and a set of redundant constraints (i.e., 90 degree rotation of (50))

$$\begin{aligned}
X_{1,1} + X_{1,2} + X_{2,3} + X_{2,4} &= 1 \\
X_{2,1} + X_{2,2} + X_{1,3} + X_{1,4} &= 1 \\
X_{3,1} + X_{3,2} + X_{4,3} + X_{4,4} &= 1 \\
X_{4,1} + X_{4,2} + X_{3,3} + X_{3,4} &= 1,
\end{aligned} \tag{77}$$

we have the convex hull of 2×2 block permutation matrices. This case also shows importance of redundant constraints from the optimization perspective. From this result, it is expected that the LP decoding performance of block permutation codes might be improved by incorporating these redundant linear equalities.

Proof of Theorem 3

Proof: In the first part of the proof, we will show that any block permutation matrix satisfies (49).

Assume that $k, b \in [1, \gamma]$ and $l \in [1, \nu]$ are arbitrary chosen. From the definition of the skewed column set $U_{k,b}^{(l)}$, the left-hand side of (49) can be rewritten as

$$\sum_{(u,v) \in U_{k,b}^{(l)}} X_{u,v} = \sum_{(u,v) \in T_{k,b}^{(l)}} X_{u,v}$$
$$+ \sum_{k' \in [1,\gamma] \setminus \{k\}} \left(\sum_{(u,v) \in T_{k',b}^{(l \bmod \nu)+1}} X_{u,v} \right). \tag{78}$$

Recall that X is assumed to be a block permutation matrix. This means that there exists a unique block index $\kappa \in [1, \gamma]$ satisfying $X(B_{\kappa,b}) \neq 0$ for given block index b, and the sub-matrix $X(B_{\kappa,b})$ is a permutation matrix. If $k = \kappa$ holds, then

$$\sum_{(u,v) \in U_{k,b}^{(l)}} X_{u,v} = \sum_{(u,v) \in T_{k,b}^{(l)}} X_{u,v} = 1 \tag{79}$$

holds. Otherwise (i.e., $k \neq \kappa$), the equality

$$\sum_{(u,v) \in U_{k,b}^{(l)}} X_{u,v} = \sum_{(u,v) \in T_{\kappa,b}^{(l \bmod \nu)+1}} X_{u,v} = 1. \tag{80}$$

holds. Thus, it has been proved that (49) holds if X is a block permutation matrix.

We then move to the opposite direction; i.e., (49) implies that X is a block permutation matrix.

Assume that a block index $b \in [1, \gamma]$ and a subindex $l \in [1, \nu]$ are arbitrary chosen. Let $j = \nu(b-1) + l$. Since X is a permutation matrix, there exists the unique row index $i \in [1, n]$ satisfying $X_{i,j} = 1$. The block $B_{k,b}$ containing the set of indices (i, j) is uniquely determined because the blocks are mutually disjoint. Under this setting, it is clear that $X(B_{k,b}) \neq 0$ holds.

In the following, we will show that

$$k' \neq k \Rightarrow X(B_{k',b}) = 0. \tag{81}$$

From the definition of the block index k, It is clear that

$$\sum_{(u,v) \in T_{k,b}^{(l)}} X_{u,v} = 1 \tag{82}$$

holds. Combining Eq. (78) and Eq. (82), we immediately obtain

$$\sum_{k' \in [1,\gamma], k' \neq k} \left(\sum_{(u,v) \in T_{k',b}^{(l \bmod \nu)+1}} X_{u,v} \right) = 0. \tag{83}$$

This equality implies that

$$(u, v) \in \bigcup_{k' \in [1,\gamma] \setminus \{k\}} T_{k',b}^{(l \bmod \nu)+1} \Rightarrow X_{u,v} = 0. \tag{84}$$

Because X is a permutation matrix,

$$\sum_{(i,j) \in T_{k,b}^{(l \bmod \nu)+1}} = 1 \tag{85}$$

should be satisfied. Applying the same argument iteratively, we consequently have

$$(u, v) \in \bigcup_{k' \in [1,\gamma] \setminus \{k\}} \bigcup_{l' \in [1,\nu]} T_{k',b}^{(l')} \Rightarrow X_{u,v} = 0. \tag{86}$$

This statement is equivalent to $k' \neq k \Rightarrow X(B_{k',b}) = 0$. Due to the definition of the block permutation matrix, it has been proved that X should be a block permutation matrix. ∎

Acknowledgement

REFERENCES

[1] I. F. Blake, G. Cohen, and M. Deza, "Coding with permutations," Inform. Contr., vol. 43, pp. 1–19, 1979.

[2] J.C. Chang, R.J. Chen, T. Kløve and S.C. Tsai, "Distance-preserving mappings from binary vectors to permutations, " IEEE Transactions on Information Theory, vol. 49, no.4, pp.1054-1059, Apr. 2003.

[3] J.C. Chang, "Distance-increasing mappings from binary vectors to permutations," IEEE Transactions on Information Theory, vol.51, pp.359-363, Jan. 2005.

[4] C. J. Colbourn, T. Kløve, and A. C. H. Ling, "Permutation arrays for powerline communication and mutually orthogonal latin squares, " IEEE Transactions on Information Theory, vol. 54, No. 6, June, 2004.

[5] G.D. Forney, Jr., R. Koetter, F.R. Kschischang, and A. Reznik, "On the effective weights of pseudocodewords for codes defined on graphs with cycles, " in Codes, Systems, and Graphical Models (B. Marcus and J. Rosenthal, eds.), vol. 123 of IMA Vol. Math. Appl., pp. 101-112, Springer Verlag, New York, Inc., 2001.

[6] C. Ding, F. W. Fu, T. Kløve and V. K. W. Wei, "Constructions of permutation arrays," IEEE Transactions on Information Theory, vol. 48, no. 4, Apr. 2002.

[7] J. Feldman, "Decoding error-correcting codes via linear programming," Massachusetts Institute of Technology, Ph. D. thesis, 2003.

[8] P. Frankl and M. Deza, "On the maximum number of permutations with given maximal and minimal distance," J. Comb. Theory, Ser. A, vol. 22, pp. 352–360, 1977.

[9] A. Jiang, R. Mateescu, M. Schwartz, and J. Bruck, "Rank modulation for flash memories," in Proc. IEEE Int. Symp. Information Theory, 2008.

[10] A. Jiang, M. Schwartz, and J. Bruck, "Error-correcting codes for rank modulation," in Proc. IEEE Int. Symp. Information Theory, 2008.

[11] T. Kløve, T. Lin, S.-C. Tsai, and W. G. Tzeng, "Permutation arrays under the Chebyshev distance," IEEE Transactions on Information Theory, vol. 56, no. 6, June 2010.

[12] S.Litsyn and V. Shevelev, "On ensembles of low-density parity-check codes: asymptotic distance distributions," *IEEE Trans. Inform. Theory*, vol.48, pp.887–908, Apr. 2002.

[13] A. J. H. Vinck, "Coded modulation for powerline communications, " AEÜ Int. J. Electron. Commun., vol. 54, pp. 45–49, Jan. 2000.

[14] A. J. H. Vinck, J. Häring, and T. Wadayama, "Coded M-FSK for power-line communications," in Proc. IEEE Int. Symp. Information Theory, 2000.

[15] A. J. H. Vinck, and H.C. Ferreira, "Permutation trellis-codes, " in Proc. IEEE Int. Symp. Information Theory, 2001.

[16] T. Wadayama and A.J.Han Vinck, "A multilevel construction of permutation codes," IEICE Transactions on Fundamentals, vol.E84-A, no.10, pp.2518–2522, 2001.

[17] D. Slepian, "Permutation modulation" ,Proc. IEEE, pp. 228-236, 1965.

[18] J. Karlof, "Permutation codes for the Gaussian channel," IEEE Trans. Inform. Theory, vol. 35, no. 4, pp. 726-732, July 1989.

[19] E. Biglieri and M. Elia, "Optimum permutation modulation codes and their asymptotic performance," IEEE Trans. Inform. Theory, vol. IT-22, no. 6, Nov. 1976.

[20] I. Ingemarsson, "Optimized permutation modulation," IEEE Trans. Inform. Theory, vol. 36, pp. 1098-1100, Sept. 1990.

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

[22] D. Slepian, "Group codes for the Gaussian channel," Bell Syst. Tech. J., vol. 47, pp. 575-602, Apr. 1968.

[23] G. Ungerboeck, "Channel coding with multilevel/phase signals," IEEE Trans. Itform. Theorv,vol.IT-28,Jan. 1982.

[24] A. Barg and A. Mazumdar, "Codes in permutations and error correction for rank modulation," IEEE Trans. Inform. Theory, vol. IT-56, pp. 3158 - 3165, July 2010.

[25] D. Knuth "The Art of Computer Programming Volume 3, " Addison-Wesley, 1998.

[26] N. Alon and J. H. Spencer, "The Probabilistic Method, 3rd. ed.," John Wiley & Sons, 2008.

[27] R. Koetter and P. O. Vontobel, "Graph covers and iterative decoding of finite-length codes", in *Proc. 3rd Int. Symp. Turbo Codes and Related Topics*, Brest, France, Sep. 2003.

[28] D. P. Bertsekas, "Nonlinear programming, " 2nd edition, Athena Scientific, 1999.

[29] S. Boyd and L. Vandenberghe, "Convex optimization," Cambridge University Press, 2004.

[30] A. Schrijver, "Combinatorial optimization: polyhedra and efficiency," Springer, 2003.

[31] T. Wadayama, "Interior point decoding for linear vector channels based on convex optimization," IEEE Trans. Inform. Theory, pp.4905-4921, vol.56, no.10, Oct. (2010)

[32] K. Fukuda, "cdd and cddplus Homepage, " `http://www.ifor.math.ethz.ch/~fukuda/cdd_home/`

[33] "The On-Line Encyclopedia of Integer Sequences, A000166", `http://oeis.org/A000166`

[34] G. M. Ziegler, "Lectures on Polytopes, " Springer-Verlag New York, 1995.

[35] G. Birkhoff, "Three observations on linear algebra," Univ. Nac. Tacuman, Rev. Ser. A 5, 147–151, 1946.

[36] J. von Neumann, "A certain zero-sum two-person game equivalent to an optimal assignment problem," Ann. Math. Studies 28, 5–12, 1953.

[37] A. Schrijver, "Combinatorial optimization, polyhedra and efficiency," Springer-Verlag Berlin, 2003.

Biography

Tadashi Wadayama was born in Kyoto, Japan,on May 9,1968. He received the B.E., the M.E., and the D.E. degrees from Kyoto Institute of Technology in 1991, 1993 and 1997, respectively. Since 1995, he has been with Okayama Prefectural University as a research associate. In 2004, he moved to Nagoya Institute of Technology as an associate professor. Since 2010, he has been a professor of Department of Computer Science, Nagoya Institute of Technology. His research interests are in coding theory, information theory, and digital communication/storage systems. He is a member of IEICE, and IEEE.

Manabu Hagiwara received the B.E. degree in mathematics from Chiba Univ. in 1997, and the M.E., and Ph.D. degrees in mathematical science from the Univ. of Tokyo in 1999 and 2002, respectively. From 2002 to 2005 he was a postdoctoral fellow at IIS, the Univ. of Tokyo. He also was a researcher at RIMS, Kyoto University, 2002. Currently, he is a research scientist of Research Center for Information Security, National Institute of Advanced Industrial Science and Technology, and is an associated professor of Center for Research and Development Initiative, Chuo Univ. He also is a research scholar at Univ. of Hawaii. His current research interests include coding theory, cryptography, information security, and algebraic combinatorics.