Distance-Preserving Mappings From Binary Vectors to Permutations

Jen-Chun Chang, Rong-Jaye Chen, Torleiv Kløve, *Fellow, IEEE*, and Shi-Chun Tsai

***Abstract*—Mappings of the set of binary vectors of a fixed length to the set of permutations of the same length are useful for the construction of permutation codes. In this correspondence, several explicit constructions of such mappings preserving or increasing the Hamming distance are given. Some applications are given to illustrate the usefulness of the construction. In particular, a new lower bound on the maximal size of permutation arrays (PAs) is given.**

***Index Terms*—Code constructions, distance, mapping, permutation arrays (PAs).**

I. Introduction

The main objects studied in this correspondence are mappings from the set of binary vectors of length n to permutations of the same length that preserve (or increase) the Hamming distance. We call them n-DPMs (distance-preserving mappings).

The inspiration comes partly from the paper [3] where Ferreira and Vinck used n-DPMs to construct permutation trellis codes. They found a 4-DPM by computer search. From this mapping they constructed n-DPMs for $5 \leq n \leq 8$, using an *ad hoc* "prefix method." It was not clear from their paper if and how the method could be generalized to $n > 8$. One result of this correspondence is a general method to systematically construct explicit DPMs for all $n \geq 4$.

A *permutation array* (PA) is a set of permutations of the first n natural numbers. PAs were somewhat studied in the 1970s, some important papers from that period are [1] and [4]. A recent application by Vinck [10] of PAs to a coding/modulation scheme for communication over power lines has created renewed interest in PAs, see [2], [3], [5], [6], [11]–[13]. Other recent papers on PAs are [7] and [9]. In this correspondence, the n-DPMs we construct are applied to give new constructions of PAs and resulting (improved) lower bounds on the size of PAs.

II. Basic Notations

Let S_n denote the set of all $n!$ permutations of $Z_n = \{1, 2, \ldots, n\}$. We represent a permutation $\pi\colon Z_n \to Z_n$ by listing its values in an n-tuple: $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$ (where $\pi_i = \pi(i)$).

The Hamming distance $d_H(\boldsymbol{a}, \boldsymbol{b})$ between two n-tuples $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)$ and $\boldsymbol{b} = (b_1, b_2, \ldots, b_n)$ of elements of any kind is the number of positions where they differ

$$d_H(\boldsymbol{a}, \boldsymbol{b}) = |\{j \in Z_n \mid a_j \neq b_j\}|.$$

The set Z_2^n denotes the set of binary vectors of length n.

Manuscript received March 6, 2002; revised October 28, 2002. This work was supported in part by The Norwegian Research Council.

J.-C. Chang is with the Department of Information Management, Ming Hsin University of Science and Technology, Hsin Chu, Taiwan (e-mail: jcchang@csie.nctu.edu.tw).

R.-J. Chen and S.-C. Tsai are with the Department of Computer Science and Information Engineering, Chiao Tung University, Hsin Chu, Taiwan (e-mail: rjchen@csie.nctu.edu.tw; sctsai@csie.nctu.edu.tw).

T. Kløve is with the Department of Informatics, University of Bergen, N-5020, Norway (e-mail: Torleiv.Klove@ii.uib.no).

Communicated by C. Carlet, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2003.809507

A DPM for length n (for short: an n-DPM) is a mapping $f\colon Z_2^n \to S_n$ such that

$$d_H(\boldsymbol{x}, \boldsymbol{y}) \leq d_H(f(\boldsymbol{x}), f(\boldsymbol{y})), \qquad \text{for all } \boldsymbol{x}, \boldsymbol{y} \in Z_2^n.$$

Let $\mathcal{F}_n$ denote the set of n-DPMs. Since $|S_n| < |Z_2^n|$ for $n < 4$, an obvious necessary condition for the existence of an n-DPM is $n \geq 4$. A heuristic argument indicates that $\mathcal{F}_n$ is quite large, even for small n, but $\mathcal{F}_n$ is known to be nonempty only for $4 \leq n \leq 8$ (Ferreira and Vinck [3]).

The main results in this correspondence are several explicit methods to construct mappings in $\mathcal{F}_n$ for all $n \geq 4$.

From any mapping in $\mathcal{F}_n$, we can obtain new mappings by permuting Z_2^n, S_n, and the positions of the elements in the values. More precisely, let f be an n-DPM and let

$$(\pi_1, \pi_2, \ldots, \pi_n) = f(x_1, x_2, \ldots, x_n).$$

Let $\sigma, \rho, \tau \in S_n$. Then the mapping defined by

$$g(x_{\sigma(1)}, x_{\sigma(2)}, \ldots, x_{\sigma(n)}) = (\rho(\pi_{\tau(1)}), \rho(\pi_{\tau(1)}), \ldots, \rho(\pi_{\tau(1)}))$$

is again an n-DPM. We call f and g equivalent.

III. On $\mathcal{F}_4$

We start by a systematic study of $\mathcal{F}_4$; in particular doing a complete listing. Renaming the elements if necessary (that is, permuting $\{1, 2, 3, 4\}$), we may assume that

$$f(0, 0, 0, 0) = (1, 2, 3, 4). \tag{1}$$

For $(\pi_1, \pi_2, \pi_3, \pi_4) = f(1, 1, 1, 1)$, there are nine possibilities since we must have

$$d_H((1, 2, 3, 4), (\pi_1, \pi_2, \pi_3, \pi_4)) = 4.$$

These nine possible π fall into two classes, depending on whether

$$\pi_a = 1, \qquad \text{where } a = \pi_1 \tag{2}$$

(that is, 1 is a fixpoint for the permutation $\pi \cdot \pi$) or not. There are three possibilities for (2) to be satisfied, namely.

$$(\pi_1, \pi_2, \pi_3, \pi_4) = (2, 1, 4, 3), (3, 4, 1, 2), (4, 3, 2, 1),$$

where π_1 is 2, 3, and 4, respectively. By permuting the positions in a suitable way and renaming the elements, we can always obtain

$$(\pi_1, \pi_2, \pi_3, \pi_4) = (2, 1, 4, 3)$$

in this case. We say that the corresponding f is of type 1. Similarly, if (2) is not satisfied, there are six possibilities for $(\pi_1, \pi_2, \pi_3, \pi_4)$, namely,

$$(2, 3, 4, 1), \quad (2, 4, 1, 3), \quad (3, 1, 4, 2)$$
$$(3, 4, 2, 1), \quad (4, 1, 2, 3), \quad (4, 3, 1, 2).$$

By permuting and renaming, we can get

$$(\pi_1, \pi_2, \pi_3, \pi_4) = (2, 3, 4, 1)$$

in this case. We say that the corresponding f is of type 2. Hence, we only have to consider

$$f(1, 1, 1, 1) \in \{(2, 1, 4, 3), (2, 3, 4, 1)\}. \tag{3}$$

Rearranging the positions of the argument values (that is, permuting $\{x_1, x_2, x_3, x_4\}$) does not affect $(0, 0, 0, 0)$ or $(1, 1, 1, 1)$ and will give 24 mappings. One of the 24 mappings obtained will satisfy

$$f(0, 0, 0, 1) < f(0, 0, 1, 0) < f(0, 1, 0, 0) < f(1, 0, 0, 0). \tag{4}$$

We call f normalized if it satisfies (1), (3), and (4). For each normalized f of type 1, there are $3 \cdot 24^2 = 1728$ mappings which normalizes to f. For f of type 2, there are $6 \cdot 24^2 = 3456$.

How many normalized f of each type are there? A complete search, using standard backtracking, has shown that the exact number of normalized f is

$$23\,227\,834 \text{ of type } 1 \quad \text{and} \quad 30\,910\,400 \text{ of type } 2.$$

Hence, the total number of mappings in $\mathcal{F}_4$ is

$$24^2(3 \cdot 23\,227\,834 + 6 \cdot 30\,910\,400) = 146\,964\,039\,552.$$

For $f \in \mathcal{F}_4$, let D_{ij} denote the number of (unordered) pairs $\{\boldsymbol{x}, \boldsymbol{y}\} \subset Z_2^4$ such that

$$d_H(\boldsymbol{x}, \boldsymbol{y}) = i \quad \text{and} \quad d_H(f(\boldsymbol{x}), f(\boldsymbol{y})) = j.$$

These numbers tell us to what extent f increases distances. By definition of DPM, $D_{ij} = 0$ if $i > j$. A DPM that increases many distances may be more interesting for applications.

We have $D_{i1} = 0$ for all i, $D_{44} = 8$, and

$$D_{12} + D_{13} + D_{14} = 32 \tag{5}$$
$$D_{22} + D_{23} + D_{24} = 48 \tag{6}$$
$$D_{33} + D_{34} = 32. \tag{7}$$

We have computed the numbers D_{ij} for all $f \in \mathcal{F}_4$. We noted that the D_{ij} are all invariant under the normalizing operations described above. Hence, we only had to consider normalized mappings.

The number of *a priori* possibilities of the D_{ij}, that is, nonnegative integral solutions of (5)–(7), is 22 678 425, but, as it turns out, only 27 458 of these actually occur.

We note that $D_{22} + D_{33}$ is the number of distances (2 or 3) which do not increase under f. For applications, it is interesting to have $D_{22} + D_{33}$ as small as possible. It turns out that there exist (exactly) two normalized mappings for which $D_{22} = D_{33} = 0$. One is obtained from the other by simple transformations. Further such mappings are, of course, the equivalent mappings.

One example (denoted by h) is given in the following table. It is not normalized but has a nice structure that can be generalized, something we will do in a later section.

$\boldsymbol{x}$	$h(\boldsymbol{x})$	$\boldsymbol{x}$	$h(\boldsymbol{x})$
(0, 0, 0, 0)	(1, 2, 3, 4)	(0, 0, 0, 1)	(1, 4, 3, 2)
(0, 0, 1, 0)	(3, 2, 1, 4)	(0, 0, 1, 1)	(3, 4, 1, 2)
(0, 1, 0, 0)	(1, 2, 4, 3)	(0, 1, 0, 1)	(1, 3, 4, 2)
(0, 1, 1, 0)	(4, 2, 1, 3)	(0, 1, 1, 1)	(4, 3, 1, 2)
(1, 0, 0, 0)	(2, 1, 3, 4)	(1, 0, 0, 1)	(2, 4, 3, 1)
(1, 0, 1, 0)	(3, 1, 2, 4)	(1, 0, 1, 1)	(3, 4, 2, 1)
(1, 1, 0, 0)	(2, 1, 4, 3)	(1, 1, 0, 1)	(2, 3, 4, 1)
(1, 1, 1, 0)	(4, 1, 2, 3)	(1, 1, 1, 1)	(4, 3, 2, 1)

The values of D_{ij} for h are $D_{12} = D_{23} = D_{34} = 32$, $D_{24} = 16$, and $D_{ij} = 0$ otherwise. The simple structure of h gives a simple algorithm for computing its values.

```
Mapping algorithm for h
Input: (x1, x2, x3, x4) ∈ Z2^4.
Output: (π1, π2, π3, π4) = h(x1, x2, x3, x4).
begin
    (π1, π2, π3, π4) ← (1, 2, 3, 4);
    if x2 = 1 then swap (π1, π2);
    if x3 = 1 then swap (π3, π4);
    if x4 = 1 then swap (π1, π3);
    if x4 = 1 then swap (π2, π4);
end
```

Note that the first two swaps are independent, as are the two last. The other mappings with $D_{22} = D_{33} = 0$ have similar simple algorithms, but the tests must be done in different orders and with different swaps.

Example 1: We illustrate the algorithm for computing $h(\boldsymbol{x})$ on one example, namely, $\boldsymbol{x} = (x_1, x_2, x_3, x_4) = (1, 1, 0, 1)$. In the following table we show how $(\pi_1, \pi_2, \pi_3, \pi_4)$ develops in each step.

		π_1	π_2	π_3	π_4
		1	2	3	4
$x_1 = 1$,	swap(π_1, π_2)	$\underline{2}$	$\underline{1}$	3	4
$x_2 = 1$,	swap(π_3, π_4)	2	1	$\underline{4}$	$\underline{3}$
$x_3 = 0$,	no swap	2	1	4	3
$x_4 = 1$,	swap(π_2, π_4)	2	$\underline{3}$	4	$\underline{1}$

IV. Recursive Constructions of DPM

In this section, we give a couple of recursive constructions of DPMs.

For a permutation $\rho = (\rho_1, \rho_2, \ldots, \rho_m) \in S_m$ and an integer n, let $\rho + n = (\rho_1 + n, \rho_2 + n, \ldots, \rho_m + n)$. This is a permutation of $\{n + 1, n + 2, \ldots, n + m\}$.

Construction 1: Let $f \in \mathcal{F}_n$ and $g \in \mathcal{F}_m$. Define $f \diamond g$ by

$$(f \diamond g)(\boldsymbol{x}, \boldsymbol{x'}) = (f(\boldsymbol{x}), g(\boldsymbol{x'}) + n).$$

Example 2: Using $f = g = h$, we get a mapping $h \diamond h \in \mathcal{F}_8$. Since $h(1, 1, 0, 0) = (2, 1, 4, 3)$ and $h(1, 1, 0, 1) = (2, 3, 4, 1)$, for example, we get

$$(h \diamond h)(1, 1, 0, 0, 1, 1, 0, 1) = (2, 1, 4, 3, 6, 7, 8, 5).$$

Theorem 1: If $f \in \mathcal{F}_n$ and $g \in \mathcal{F}_m$, then $f \diamond g \in \mathcal{F}_{n+m}$.

Proof: We have

$$\begin{aligned} &d_H((f \diamond g)(\boldsymbol{x}, \boldsymbol{x'}), (f \diamond g)(\boldsymbol{y}, \boldsymbol{y'})) \\ &\quad = d_H(f(\boldsymbol{x}), f(\boldsymbol{y})) + d_H(g(\boldsymbol{x'}) + n, g(\boldsymbol{y'}) + n) \\ &\quad = d_H(f(\boldsymbol{x}), f(\boldsymbol{y})) + d_H(g(\boldsymbol{x'}), g(\boldsymbol{y'})) \\ &\quad \geq d_H(\boldsymbol{x}, \boldsymbol{y}) + d_H(\boldsymbol{x'}, \boldsymbol{y'}) \\ &\quad = d_H((\boldsymbol{x}, \boldsymbol{x'}), (\boldsymbol{y}, \boldsymbol{y'})). \end{aligned}$$

QED

Corollary 1: For all n and m we have

$$|\mathcal{F}_{n+m}| \geq |\mathcal{F}_n| \cdot |\mathcal{F}_m|.$$

Proof: This follows directly from Theorem 1. QED

Corollary 2: For all $n \geq 4$ we have $|\mathcal{F}_n| > 0$.

Proof: The statement is true for $4 \leq n \leq 8$ (see [3]). The general statement follows by induction, using Corollary 1. QED

In our next class of recursive constructions, we construct n-DPMs from $(n - 1)$-DPMs. A position function (of length m) is a function p: $Z_2^m \to Z_m$.

Construction 2: Let $g \in \mathcal{F}_{n-1}$ and let p be a position function of length $n - 1$. For $\boldsymbol{x} \in Z_2^{n-1}$, let

$$(\pi_1, \pi_2, \ldots, \pi_{n-1}) = g(\boldsymbol{x}), \qquad q = p(\boldsymbol{x}).$$

Define $f\colon Z_2^n \to S_n$ by

$$f(\boldsymbol{x}, 0) = (\pi_1, \pi_2, \ldots, \pi_{n-1}, n),$$
$$f(\boldsymbol{x}, 1) = (\pi_1, \ldots, \pi_{q-1}, n, \pi_{q+1}, \ldots, \pi_{n-1}, \pi_q).$$

Informally, we can describe the construction as follows: first we append the element n at the end of π; if $x_n = 1$ we further swap the elements in positions q and n.

The function f depends on g and p; sometimes, we explicitly include them in the notation and write $f = f_{g,p}$.

Example 3: In this simple example, we let $n = 6$ and p is defined by $p(\boldsymbol{x}) = 2$ for all $\boldsymbol{x}$. Since p is a constant function, $f(\boldsymbol{x}, x_6)$ depends only on $g(\boldsymbol{x})$ and x_6. We give a few examples.

$g(\boldsymbol{x})$	x_6	$f(\boldsymbol{x}, x_6)$
(1, 2, 3, 4, 5)	0	(1, 2, 3, 4, 5, 6)
(1, 2, 3, 4, 5)	1	(1, 6, 3, 4, 5, 2)
(1, 3, 5, 2, 4)	0	(1, 3, 5, 2, 4, 6)
(1, 3, 5, 2, 4)	1	(1, 6, 5, 2, 4, 3)
(5, 4, 3, 2, 1)	0	(5, 4, 3, 2, 1, 6)
(5, 4, 3, 2, 1)	1	(5, 6, 3, 2, 1, 4)

We see that $f(\boldsymbol{x}, 0)$ always has 6 in the last position and $f(\boldsymbol{x}, 1)$ always has 6 in the second position (since $p(\boldsymbol{x}) = 2$).

Theorem 2: If $g \in \mathcal{F}_{n-1}$ and p is a position function of length $n-1$, then $f = f_{g,p} \in \mathcal{F}_n$.

Proof: Let $\boldsymbol{x}, \boldsymbol{y} \in Z_2^{n-1}$. We only need to show that

$$d_H(f(\boldsymbol{x}, x_n), f(\boldsymbol{y}, y_n)) \geq d_H(g(\boldsymbol{x}), g(\boldsymbol{y})) + d_H(x_n, y_n) \quad (8)$$

since this inequality and the fact that $g \in \mathcal{F}_{n-1}$ implies that

$$\begin{aligned} d_H(f(\boldsymbol{x}, x_n), f(\boldsymbol{y}, y_n)) &\geq d_H(\boldsymbol{x}, \boldsymbol{y}) + d_H(x_n, y_n) \\ &= d_H((\boldsymbol{x}, x_n), (\boldsymbol{y}, y_n)). \end{aligned}$$

We let

$$(\pi_1, \pi_2, \ldots, \pi_{n-1}) = g(\boldsymbol{x}), \qquad q = p(\boldsymbol{x})$$
$$(\rho_1, \rho_2, \ldots, \rho_{n-1}) = g(\boldsymbol{y}), \qquad r = p(\boldsymbol{y}).$$

To prove (8), we consider the possible values of x_n and y_n. By symmetry, we may assume without loss of generality that $x_n \leq y_n$. There are four cases to consider.

Case I: $x_n = y_n = 0$. Then

$$f(\boldsymbol{x}, n) = (\pi_1, \pi_2, \ldots, \pi_{n-1}, 0)$$
$$f(\boldsymbol{y}, 0) = (\rho_1, \rho_2, \ldots, \rho_{n-1}, n).$$

Clearly, (8) is satisfied (with equality) in this case.

Case II: $x_n = 0$ and $y_n = 1$. Then

$$f(\boldsymbol{x}, 0) = (\pi_1, \ldots, \pi_{r-1}, \pi_r, \pi_{r+1} \ldots, \pi_{n-1}, n)$$
$$f(\boldsymbol{y}, 1) = (\rho_1, \ldots, \rho_{r-1}, n, \rho_{r+1} \ldots, \rho_{n-1}, \rho_r).$$

We see that $f(\boldsymbol{x}, 0)$ and $f(\boldsymbol{y}, 1)$ differ in positions r and n. On the other hand, we have "lost" one differing position if $\pi_r \neq \rho_r$. More precisely

$$\begin{aligned} d_H(f(\boldsymbol{x}, 0), f(\boldsymbol{y}, 1)) &= d_H(g(\boldsymbol{x}), g(\boldsymbol{y})) - d_H(\pi_r, \rho_r) + 2 \\ &\geq d_H(g(\boldsymbol{x}), g(\boldsymbol{y})) + 1 \end{aligned}$$

and so (8) is satisfied also in this case.

Case III: $x_n = y_n = 1$ and $r = q$. Then

$$f(\boldsymbol{x}, 1) = (\pi_1, \ldots, \pi_{q-1}, n, \pi_{q+1} \ldots, \pi_{n-1}, \pi_q)$$
$$f(\boldsymbol{y}, 1) = (\rho_1, \ldots, \rho_{q-1}, n, \rho_{q+1} \ldots, \rho_{n-1}, \rho_q).$$

Clearly, (8) is satisfied (with equality).

Case IV: $x_n = y_n = 1$ and $r \neq q$. Then

$$\begin{aligned} f(\boldsymbol{x}, 1) = (&\pi_1, \ldots, \pi_{q-1}, n, \pi_{q+1}, \ldots, \\ &\pi_{r-1}, \pi_r, \pi_{r+1}, \ldots, \pi_{n-1}, \pi_q) \\ f(\boldsymbol{y}, 1) = (&\rho_1, \ldots, \rho_{q-1}, \rho_q, \rho_{q+1}, \ldots, \\ &\rho_{r-1}, n, \rho_{r+1} \ldots, \rho_{n-1}, \rho_r) \end{aligned}$$

(if $q < r$ and, similarly, if $q > r$). We see that $f(\boldsymbol{x}, 1)$ and $f(\boldsymbol{y}, 1)$ differ in positions q and r and possibly in position n. On the other hand, it may be that $g(\boldsymbol{x})$ and $g(\boldsymbol{y})$ differ in positions q and r, differences that are "lost" in the step from g to h. More precisely, we get

$$\begin{aligned} d_H(f(\boldsymbol{x}, 1), f(\boldsymbol{y}, 1)) &= d_H(g(\boldsymbol{x}), g(\boldsymbol{y})) + 2 + d_H(\pi_q, \rho_r) \\ &\quad - d_H(\pi_q, \rho_q) - d_H(\pi_r, \rho_r) \\ &\geq d_H(g(\boldsymbol{x}), g(\boldsymbol{y})). \end{aligned}$$

Hence, (8) is satisfied also in this case. QED

Corollary 3: For all $n \geq 4$ we have $|\mathcal{F}_{n+1}| \geq n^{2^n} |\mathcal{F}_n|$.

Proof: Since a position function can take any of n values on each of the 2^n arguments, the number of positions functions is n^{2^n}. Hence, we have $|\mathcal{F}_n|$ choices for g and n^{2^n} choices for p. We have to show that each choice gives a distinct f. Let $g_1, g_2 \in \mathcal{F}_n$ be distinct. Then there exists an $\boldsymbol{x} \in Z_2^n$ such that $g_1(\boldsymbol{x}) \neq g_2(\boldsymbol{x})$. Therefore,

$$f_{g_1,p_1}(\boldsymbol{x}, 0) = (g_1(\boldsymbol{x}), n+1) \neq (g_2(\boldsymbol{x}), n+1) = f_{g_2,p_2}(\boldsymbol{x}, 0)$$

for all position functions p_1 and p_2.

Let p_1 and p_2 be distinct position functions. Then there exists an $\boldsymbol{x} \in Z_2^n$ such that $p_1(\boldsymbol{x}) \neq p_2(\boldsymbol{x})$. Therefore, $f_{g,p_1}(\boldsymbol{x}, 1)$ and $f_{g,p_2}(\boldsymbol{x}, 1)$ differ in positions $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ for all g. QED

Construction 2 gives mappings that are implicitly defined (however, with an algorithm to compute them). By carefully keeping track of the recursive steps, it is possible to obtain more explicit expressions for some mappings in $\mathcal{F}_n$ (for all n). For example, starting from some $g_4 \in \mathcal{F}_4$ and the positions function that is constant 1 in each step, we obtain a mapping $g_n = f_{g_{n-1},1} \in \mathcal{F}_n$ which can be described as follows (a proof follows by induction on n).

Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in Z_2^n$

$$\text{and } g_n(\boldsymbol{x}) = (\pi_1, \pi_2, \ldots, \pi_n).$$

Let $(\rho_1, \rho_2, \rho_3, \rho_4) = g_4(x_1, x_2, x_3, x_4)$.
If $x_i = 0$ for all i, $5 \leq i \leq n$, then

$$\begin{aligned} \pi_i &= \rho_i, && \text{for } 1 \leq i \leq 4 \\ \pi_i &= i, && \text{for } 5 \leq i \leq n. \end{aligned}$$

If $x_i = 1$ exactly for $i = i_m$, $m = 1, 2, \ldots, u$ where $u \geq 1$ and $5 \leq i_1 < i_2 < \cdots < i_u \leq n$, then

$$\begin{aligned} \pi_1 &= i_u \\ \pi_i &= \rho_i, && \text{for } 2 \leq i \leq 4 \\ \pi_{i_1} &= \rho_1 \\ \pi_{i_m} &= i_{m-1}, && \text{for } 2 \leq m \leq u \\ \pi_i &= i, && \text{otherwise.} \end{aligned}$$

Example 4: If

$$\boldsymbol{x} = (1,1,0,0,0,1,1,0,1,0) \text{ and } g_4(1,1,0,0) = (4,3,2,1)$$

then $u = 3$, $i_1 = 6$, $i_2 = 7$, $i_3 = 9$, and so

$$g_{10}(1,\ 1,\ 0,\ 0,\ 0,\ 1,\ 1,\ 0,\ 1,\ 0) = (9,\ 3,\ 2,\ 1,\ 5,\ 4,\ 6,\ 8,\ 7,\ 10).$$

The description above gives rise to a simple algorithm to compute $g_n(\boldsymbol{x})$ in this case.

Mapping algorithm
Input: $(x_1, x_2, \ldots, x_n) \in Z_2^n$ and $g_4 \in \mathcal{F}_4$.
Output: $(\pi_1, \pi_2, \ldots, \pi_n) = g_n(x_1, x_2, \ldots, x_n)$
where $g_{j+1} = f_{g_j,1}$ for $j \geq 4$.
begin
$(\pi_1, \pi_2, \pi_3, \pi_4) \leftarrow g_4(x_1, x_2, x_3, x_4)$;
$t \leftarrow \pi_1$;
for $i = 5$ **to** n **do**
if $x_i = 1$ **then begin** $\pi_i \leftarrow t$; $t \leftarrow i$; **end**
else $\pi_i \leftarrow i$;
$\pi_1 \leftarrow t$;
end

If we choose $g_4 = h$, for example, we can substitute the explicit algorithm of h for the part $(\pi_1, \pi_2, \pi_3, \pi_4) \leftarrow g_4(x_1, x_2, x_3, x_4)$ of the algorithm above.

V. Explicit n-DPMs for Even n

Mappings similar to $h \in \mathcal{F}_4$ may be constructed for all even $n = 2m \geq 4$. These mappings, which we denote by h_{2m}, we define by an algorithm.

Construction 3:
Mapping algorithm for h_{2m}
Input: $(x_1, x_2, \ldots, x_{2m}) \in Z_2^{2m}$.
Output: $(\pi_1, \pi_2, \ldots, \pi_{2m}) = h_{2m}(x_1, x_2, \ldots, x_{2m})$.
begin
$(\pi_1, \pi_2, \ldots, \pi_{2m}) \leftarrow (1, 2, \ldots, 2m)$;
for i **from** 1 to m **do**
if $x_i = 1$ **then** swap (π_{2i-1}, π_{2i});
for i **from** $m+i$ to $2m$ **do**
if $x_i = 1$ **then** swap (π_{i-m}, π_i);
end

The first m swaps are independent, and so are the last m. Therefore, it is easy to give an explicit description of the mapping. Let ψ denote the permutation after the first m steps. Then, for $1 \leq j \leq m$

$$(\psi_{2j-1},\ \psi_{2j}) = \begin{cases} (2j-1,\ 2j), & \text{if } x_j = 0 \\ (2j,\ 2j-1), & \text{if } x_j = 1. \end{cases} \tag{9}$$

The step from ψ to π is defined by

$$(\pi_{i-m},\ \pi_i) = \begin{cases} (\psi_{i-m},\ \psi_i), & \text{if } x_i = 0 \\ (\psi_i,\ \psi_{i-m}), & \text{if } x_i = 1 \end{cases} \tag{10}$$

for $m+1 \leq i \leq 2m$.

Combining the preceding two steps, we get explicit expressions for π_i. Each π_i depends on i and the values of exactly two x_j (which also depends on i). For $1 \leq i \leq m$, we get

x_{i+m}	$x_{\lceil i/2 \rceil}$	$i \bmod 2$	π_i
0	0		i
0	1	0	$i-1$
0	1	1	$i+1$
x_{i+m}	$x_{\lceil (i+m)/2 \rceil}$	$(i+m) \bmod 2$	π_i
1	0		$i+m$
1	1	0	$i+m-1$
1	1	1	$i+m+1$

For $m+1 \leq i \leq 2m$ we get

x_i	$x_{\lceil i/2 \rceil}$	$i \bmod 2$	π_i
0	0		i
0	1	0	$i-1$
0	1	1	$i+1$
x_i	$x_{\lceil (i-m)/2 \rceil}$	$(i-m) \bmod 2$	π_i
1	0		$i-m$
1	1	0	$i-m-1$
1	1	1	$i-m+1$

Theorem 3: i) For all $m \geq 2$ we have $h_{2m} \in \mathcal{F}_{2m}$. ii) For $m = 2$ or $m \geq 3$ and odd, if $0 < d_H(\boldsymbol{x}, \boldsymbol{y}) < 2m$, then

$$d_H(h_{2m}(\boldsymbol{x}),\ h_{2m}(\boldsymbol{y})) > d_H(\boldsymbol{x}, \boldsymbol{y}).$$

Proof: For m even, we see that for any even i, $m+1 < i \leq 2m$, the four elements $(\pi_{i-m-1}, \pi_{i-m}, \pi_{i-1}, \pi_i)$ only depend on $(x_{(i-m)/2}, x_{i/2}, x_{i-1}, x_i)$; and are obtained by the following four steps.

if $x_{(i-m)/2} = 1$ **then** swap(π_{i-m-1}, π_{i-m});
if $x_{i/2} = 1$ **then** swap(π_{i-1}, π_i);
if $x_{i-1} = 1$ **then** swap(π_{i-m-1}, π_{i-1});
if $x_i = 1$ **then** swap(π_{i-m}, π_i).

This is exactly the algorithm for h (except for a renaming of the elements). Hence, h_{2m} is essentially $m/2$ copies of h; more precisely, h_{2m} is equivalent to

$$h \diamond h \diamond \cdots \diamond h \qquad (m/2 \text{ factors}). \tag{11}$$

By Theorem 1, h_{2m} is a DPM. This proves i) for m even.

Consider m odd. Let

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_{2m}),\ \ \boldsymbol{y} = (y_1, y_2, \ldots, y_{2m}) \in Z_2^{2m}$$

and let

$$\pi = (\pi_1, \pi_2, \ldots, \pi_{2m}) = h_{2m}(\boldsymbol{x})$$
$$\rho = (\rho_1, \rho_2, \ldots, \rho_{2m}) = h_{2m}(\boldsymbol{y}).$$

Further, let

$$\boldsymbol{x}_L = (x_1, x_2, \ldots, x_m) \quad \text{and} \quad \boldsymbol{x}_R = (x_{m+1}, x_{m+2}, \ldots, x_{2m})$$

and define $\boldsymbol{y}_L$ and $\boldsymbol{y}_R$ similarly. We will first show that

$$d_H(\pi, \rho) \geq 2d_H(\boldsymbol{x}_L, \boldsymbol{y}_L) \tag{12}$$

$$d_H(\pi, \rho) \geq 2d_H(\boldsymbol{x}_R, \boldsymbol{y}_R). \tag{13}$$

First, suppose that $x_j \neq y_j$ for some j, $1 \leq j \leq m$. Without loss of generality, we assume that $x_j = 0$ and $y_j = 1$. Let $i = 2j$. Since $i-m$ is odd, the explicit expressions above show that

$$\pi_{i-1}, \rho_i \in \{i-m-2, i-m-1, i-1, i+m-2, i+m-1\},$$
$$\pi_i, \rho_{i-1} \in \{i-m, i-m+1, i, i+m, i+m+1\}.$$

This implies that $\pi_{i-1} \neq \rho_{i-1}$ and $\pi_i \neq \rho_i$. This shows that for each j contributing to $d_H(\boldsymbol{x}_L, \boldsymbol{y}_L)$, both $2j-1$ and $2j$ contribute to $d_H(\pi, \rho)$. This proves (12). The proof of (13) is similar. Let $m+1 \leq i \leq 2m$ and suppose that $x_i = 0$ and $y_i = 1$. Then

$$\pi_i, \rho_{i-m} \in \{i-1, i, i+1\}$$
$$\pi_{i-m}, \rho_i \in \{i-m-1, i-m, i-m+1\}.$$

Since $m > 2$, we have $i - m + 1 < i - 1$, and so $\pi_i \neq \rho_i$ and $\pi_{i-m} \neq \rho_{i-m}$. This proves (13). Combining (12) and (13), we get

$$2d_H(\pi, \rho) \geq 2d_H(\boldsymbol{x}_L, \boldsymbol{y}_L) + 2d_H(\boldsymbol{x}_R, \boldsymbol{y}_R) = 2d_H(\boldsymbol{x}, \boldsymbol{y}). \tag{14}$$

This proves i) for m odd. To prove ii), we first note that for $m = 2$ this is a property of $h_4 = h$ described before. Consider m odd, and suppose that

$$0 < d_H(\pi, \rho) = d_H(\boldsymbol{x}, \boldsymbol{y}) < 2m. \tag{15}$$

Then (14) and (15) imply that

$$d_H(\pi, \rho) = 2d_H(\boldsymbol{x}_L, \boldsymbol{y}_L) = 2d_H(\boldsymbol{x}_R, \boldsymbol{y}_R) \tag{16}$$

and so

$$0 < d_H(\boldsymbol{x}_L, \boldsymbol{y}_L) = d_H(\boldsymbol{x}_R, \boldsymbol{y}_R) < m. \tag{17}$$

Let

$$A = \{i \mid x_i = y_i\}, \qquad B = \{i \mid \pi_i = \rho_i\}.$$

Then (16) and the proof above imply that

$$\text{for } 1 \leq j \leq m, \qquad j \in A \text{ if and only if } \{2j-1, 2j\} \subset B \tag{18}$$

and

$$\text{for } m+1 \leq i \leq 2m, \qquad i \in A \text{ if and only if } \{i-m, i\} \subset B. \tag{19}$$

In particular, this implies that

$$\text{for } 1 \leq j \leq m, \qquad 2j-1 \in B \text{ if and only if } 2j \in B \tag{20}$$

and

$$\text{for } m+1 \leq i \leq 2m, \qquad i \in B \text{ if and only if } i-m \in B. \tag{21}$$

We will first show that

$$\text{if } m+1 \leq i \leq 2m, \qquad \text{then } i \in B. \tag{22}$$

We note that (17) implies that there exists at least one $i \in A$ such that $m+1 \leq i \leq 2m$. By (19), $i \in B$. We show next that

$$i-1 \in B. \tag{23}$$

If i is even, then (20) implies that $i-1 \in B$. If i is odd, then $i-m$ is even, and $i-m \in B$ by (21). Hence, $i-m-1 \in B$ by (20) and so $i-1 \in B$ by (21). This proves (23). Using (23) repeatedly, we get

$$j \in B, \qquad \text{for all } j,\ m \leq j \leq i. \tag{24}$$

In particular, $m \in B$, and so $2m \in B$ by (21). Since i was an arbitrary element of B, (24) is in particular valid for $i = 2m$. By (19), $j \in A$ for all j, $m+1 \leq j \leq 2m$, that is, $d_H(\boldsymbol{x}_R, \boldsymbol{y}_R) = 0$, contradicting (17). The assumption that led to the contradiction was (15). Hence, (15) is not possible. QED

Let h_n^{2m} be defined recursively for $n \geq 2m$ by Construction 2, starting with $h_{2m}^{2m} = h_{2m}$ and using the position function that is constant 1 in all steps.

We get a simple algorithm for computing h_n^{2m} using only swaps:

```
Mapping algorithm for h_n^{2m}
Input: (x_1, x_2, ..., x_n) ∈ Z_2^n.
Output: (π_1, π_2, ..., π_n) = h_n^{2m}(x_1, x_2, ..., x_n).
begin
  (π_1, π_2, ..., π_n) ← (1, 2, ..., n);
  for i from 1 to m do
    if x_i = 1 then swap (π_{2i-1}, π_{2i});
  for i from m+1 to 2m do
    if x_i = 1 then swap (π_{i-m}, π_i);
  for i from 2m+1 to n do
    if x_i = 1 then swap (π_1, π_i);
end
```

Theorem 4: Let $m = 2$ or $m \geq 3$ and odd. If $n \geq 2m$, $\boldsymbol{x}, \boldsymbol{y} \in Z_2^n$, and $0 < d_H(\boldsymbol{x}, \boldsymbol{y}) < 2m$, then

$$d_H(h_n^{2m}(\boldsymbol{x}), h_n^{2m}(\boldsymbol{y})) > d_H(\boldsymbol{x}, \boldsymbol{y}).$$

Proof: For $n = 2m$, this was shown in Theorem 3 ii). For $n \geq 2m+1$ it follows by induction, using (8). QED

We note that the property shown in Theorem 4 is the main feature of the mapping h_n^{2m}. In the next section we show how this property comes into play.

VI. Applications to Permutation Arrays

An (n, d) PA is a subset of S_n with the property that the Hamming distance between any two distinct permutations in the array is at least d. The maximal size of such a PA is denoted by $P(n, d)$.

One application of the DPMs is to construct PAs from binary codes. Clearly, if C is an (n, d) code and $f \in \mathcal{F}_n$, then $f(C)$ is an (n, d) PA (this is a main reason for studying DPMs in the first place).

Let $A(n, d)$ denote the maximal size of an (n, d) code, that is, a binary code of length n and minimum distance d. This quantity is well studied. A number of lower bounds on $A(n, d)$ exist, see, e.g., [8, Ch. 5]. From the construction $f(C)$ we immediately get the following bound. For $n \geq 4$ we have

$$P(n, d) \geq A(n, d). \tag{25}$$

Before we go on, we give a short survey of known bounds on $P(n, d)$. Since we clearly have $P(n, d) \geq P(n, d+1)$ for all $d = 1, 2, \ldots, n-1$, and $P(n, n) = n$, we have the trivial lower bound

$$P(n, d) \geq n. \tag{26}$$

In most cases, this has been the best lower bound known. If n is a power of a prime, then $P(n, n-1) = n(n-1)$ and so

$$P(n, d) \geq n(n-1), \qquad \text{for } d \leq n-1. \tag{27}$$

Using the existence of DPMs we can improve these bounds in many cases.

Even if the lower bound (25) often is better than the bound (27), it is probably quite weak in most cases. The only general upper bound on $P(n, d)$ known was given by Deza and Vanstone [1]

$$P(n, d) \leq \frac{n!}{(d-1)!}. \tag{28}$$

The gap between the two bounds is quite large in most cases.

Example 5: It is known that $A(23, 7) = 2^{12} = 4096$ (shown by the perfect $[23, 12, 7]$ Golay code). By (25), $P(23, 7) \geq 4096$. In comparison, (27) which applies since 23 is a prime, only gives $P(23, 7) \geq 506$.

Example 6: It is known (see, e.g., [8, p. 300]) that we have

$$A(n, d) \leq \frac{2d}{2d - n} \tag{29}$$

for $d > \frac{n}{2}$ (the Plotkin bound). Hence, if $d \geq \frac{n}{2} + 1$, then $A(n, d) < n$ and so (25) is weaker than the trivial bound (26). Therefore, (25) is of interest only if $d \leq \frac{n+1}{2}$.

In some cases, the minimum distance may increase, that is, $f(C)$ may be an (n, d') where $d' > d$. A general such result is obtained from Theorem 4, giving a bound that is stronger than (25).

Theorem 5: For $n \geq 4$ and $2 \leq d \leq n$

$$P(n, d) \geq A(n, d - 1). \tag{30}$$

Proof: If there exists an odd $m \geq 3$ (or $m = 2$) such that

$$n \geq 2m > d - 1 \tag{31}$$

let C be an $(n, d-1)$ code and consider $h_n^{2m}(C)$. This is a PA of length n, and by Theorem 4, all distances between distinct permutations are at least d. This proves the theorem in this case. In particular, such an m exists if $n - (d - 1) \geq 4$, that is, $d \leq n - 3$. On the other hand, the Plotkin bound (29) shows that

$$A(n, d - 1) < n \leq P(n, d)$$

if $d \geq \frac{n}{2} + 2$. It remains to check the cases when $n - 2 \leq d \leq \frac{n+3}{2}$. For $n > 7$, there are no such cases. The only remaining cases for (n, d) are, therefore, $(7, 5)$, $(6, 4)$, $(5, 3)$, $(5, 4)$, $(4, 2)$, $(4, 3)$. The first two satisfy (31) for $m = 3$, the last four satisfy (31) for $m = 2$. QED

Example 7: For $n = 15$, (27) can not be used since 15 is not a prime. Hence, we only had the trivial bound (26) that gives $P(15, 4) \geq 15$. The bound (30) gives $P(15, 4) \geq A(15, 3) = 2^{11}$.

VII. Possible Generalizations

The problem of finding DPMs can be generalized in two directions, considering mappings $Z_q^m \rightarrow S_n$ where we may have $m \neq n$ and $q \neq 2$.

Since Z_q^m contains pairs of vectors of mutual distance m, a necessary condition for the existence of DPMs is $m \leq n$. Any mapping $f\colon Z_q^m \rightarrow S_m$ can be trivially extended to a mapping $g\colon Z_q^m \rightarrow S_n$ by defining $g(\boldsymbol{x}) = (f(\boldsymbol{x}), m + 1, m + 2, \ldots, n)$. Therefore, it seems that considering $m < n$ is not particularly interesting, at least from an application point of view.

For existence of DPMs $Z_q^n \rightarrow S_n$, a necessary condition is clearly $q^n \leq n!$. Our constructions do not immediately generalize to $q > 2$. We have done some preliminary work on general q, and may return to this in a future paper.

References

[1] M. Deza and S. A. Vanstone, "Bounds on permutation arrays," *J. Statist. Planning and Inference*, vol. 2, pp. 197–209, 1978.

[2] C. Ding, F.-W. Fu, T. Kløve, and V. K. Wei, "Constructions of permutation arrays," *IEEE Trans. Inform. Theory*, vol. 48, pp. 977–980, Apr. 2002.

[3] H. C. Ferreira and A. J. H. Vinck, "Inference cancellation with permutation trellis arrays," *Proc. IEEE Vehicular Technology Conf.*, pp. 2401–2407, 2000.

[4] P. Frankel 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.

[5] F.-W. Fu and T. Kløve, "Two constructions of permutation arrays," *IEEE Trans. Inform. Theory*, submitted for publication.

[6] T. Kløve, "Classification of permutation codes of length 6 and minimum distance 5," in *Proc. Int. Symp. Information Theory and Its Applications*, 2000, pp. 465–468.

[7] R. Mathon and A. P. Street, "Overlarge sets of 2 (11, 5, 2) designs and related configurations," *Discr. Math*, vol. 255, pp. 275–286, 2002.

[8] V. S. Pless and W. C. Huffman, Eds., *Handbook of Coding Theory*. Amsterdam, The Netherlands: Elsevier, 1998.

[9] H. Tarnanen, "Upper bounds on permutation codes via linear programming," *Europ. J. Combin.*, vol. 20, pp. 101–114, 1999.

[10] A. J. H. Vinck, "Coded modulation for powerline communications," *AEÜ Int. J. Electron. Commun.*, vol. 54, no. 1, pp. 45–49, 2000.

[11] A. J. H. Vinck and J. Häring, "Coding and modulation for power-line communications," in *Proc. Int. Symp. Power Line Communication*, Limerick, Ireland, Apr. 5–7, 2000.

[12] A. J. H. Vinck, J. Häring, and T. Wadayama, "Coded M-FSK for power-line communications," in *Proc. IEEE Int. Symp. Information Theory*, Sorrento, Italy, June 2000, p. 137.

[13] T. Wadayama and A. J. H. Vinck, "A multilevel construction of permutation codes," *IEICE Trans. Fundamentals Electron., Commun. Comp. Sci.*, vol. 84, pp. 2518–2522, 2001.

On Two High-Rate Algebraic Space–Time Codes

Mohamed Oussama Damen, *Member, IEEE*, and Norman C. Beaulieu, *Fellow, IEEE*

***Abstract*—We examine some algebraic properties of two high-rate linear space–time block codes over $M = 2, 3$ transmit antennas. Although these high-rate codes have positive coding gain, the gain decreases when increasing the constellation size. We give tight upper and lower bounds on the achieved coding gains as functions of the size of the constellations used. We show that when using the irrational numbers $\sqrt{3}$ and $\sqrt{2}$, the coding gains express the approximation of these numbers by continued fractions depending on the constellations used. The poor approximation of these numbers by rational numbers is then shown to make the coding gains decrease slowly when increasing the constellation size.**

***Index Terms*—Block codes, continued fractions, diversity methods, multiple-input–multiple-output (MIMO) systems, number theory.**

I. Introduction and System Model

The use of algebraic constellations over multitransmit and multireceive antennas [1], [2] allows for the exploitation of the large capacity of multiantenna systems [3] while achieving the full transmit diversity [4]. However, the coding gains of these high data-rate codes are not the same for all constellations since they correspond to the degree of approximation of some irrational numbers by rational numbers [2], [5].

In this correspondence, we examine the coding gain properties of two high-rate linear space-time block codes (STBCs) for $M = 2, 3$

Manuscript received June 5, 2001; revised August 10, 2002. The material in this correspondence was presented in part at the IEEE GLOBECOM, San Antonio, TX, November 2001.

The authors are with the Department of Electrical and Computer Engineering, ECERF W2-073, University of Alberta, Edmonton, AB T6G 2V4, Canada (e-mail: damen@ee.ualberta.ca; beaulieu@ee.ualberta.ca).

Communicated by G. Caire, Associate Editor for Communications.

Digital Object Identifier 10.1109/TIT.2003.809509