Des Codes Crypt (2006) 40:139–155
DOI 10.1007/s10623-006-0003-9

# On the construction of permutation arrays via mappings from binary vectors to permutations

**Yen-Ying Huang · Shi-Chun Tsai · Hsin-Lung Wu**

Received: 28 September 2005 / Revised: 6 January 2006 /
Accepted 9 January 2006


**Abstract** An $(n, d, k)$-mapping $f$ is a mapping from binary vectors of length $n$ to permutations of length $n + k$ such that for all $x, y \in \{0, 1\}^n$, $d_H(f(x), f(y)) \geq d_H(x, y) + d$, if $d_H(x, y) \leq (n+k)-d$ and $d_H(f(x), f(y)) = n+k$, if $d_H(x, y) > (n+k)-d$. In this paper, we construct an $(n, 3, 2)$-mapping for any positive integer $n \geq 6$. An $(n, r)$-permutation array is a permutation array of length $n$ and any two permutations of which have Hamming distance at least $r$. Let $P(n, r)$ denote the maximum size of an $(n, r)$-permutation array and $A(n, r)$ denote the same setting for binary codes. Applying $(n, 3, 2)$-mappings to the design of permutation array, we can construct an efficient permutation array (easy to encode and decode) with better code rate than previous results [Chang (2005). IEEE Trans inf theory 51:359–365, Chang et al. (2003). IEEE Trans Inf Theory 49:1054–1059; Huang et al. (submitted)]. More precisely, we obtain that, for $n \geq 8$, $P(n, r) \geq A(n-2, r-3) > A(n-1, r-2) = A(n, r-1)$ when $n$ is even and $P(n, r) \geq A(n-2, r-3) = A(n-1, r-2) > A(n, r-1)$ when $n$ is odd. This improves the best bound $A(n-1, r-2)$ so far [Huang et al. (submitted)] for $n \geq 8$.

**Keywords** Permutation array · Coding theory

**AMS Classification:** 94B05

---

Communicated by R.C. Mullin.

---

The work was supported in part by the National Science Council of Taiwan under contract NSC-93-2213-E-009-117.

---

Y.-Y. Huang · S.-C. Tsai (✉)· H.-L. Wu
Department of Computer Science, National Chiao Tung University, Hsinchu 30050, Taiwan
e-mail: sctsai@csie.nctu.edu.tw

Y.-Y. Huang
e-mail: huangyy@csie.nctu.edu.tw

H.-L. Wu
e-mail: hsinlung@csie.nctu.edu.tw

## 1 Introduction

### 1.1 Background

Let $P_n$ be the set of all permutations of $\{1, 2, \ldots, n\}$. A permutation array is a subset of $P_n$. Given two permutations, one can define their Hamming distance to be the number of positions in which they differ. An $(n, r)$-permutation array is a permutation array such that any two permutations have Hamming distance at least $r$. From the combinatorial view, it is interesting to study the maximum size of an $(n, r)$-permutation array [5]. For its application, there are many results which applied permutation arrays to design coding/modulation schemes for communication over power lines [7, 10–12]. Their works stimulated a series of the efficient designs of permutation arrays with some particular minimum Hamming distance restriction [3, 4, 6, 9, 13]. However, given $(n, r)$, it is not easy to construct an $(n, r)$-permutation array. In contrast, it is relatively easier to construct the so-called $(n, r)_q$-code which is a subset of $[q]^n$ and any two binary strings of which have Hamming distance at least $r$. There are many good methods to construct $(n, r)_q$-codes such as Reed-Solomon code. Now suppose we have an efficient mapping (easy to compute the output and the inverse) from $[q]^m$ to $P_n$ such that the mapping preserves or increases the Hamming distance. Then it is clear that we can construct an efficient permutation arrays (easy to encode and decode) satisfying the desired minimum Hamming distance constraint. This motivates the study of distance-preserving (or distance-increasing) mapping from $q$-ary vectors to permutations.

There are several results for distance-preserving and distance-increasing mappings [1, 2, 8]. In particular, these papers introduced two kinds of mappings. One is distance-preserving mapping (**DPM**) [2] and the other is distance-increasing mapping (**DIM**) [1]. More precisely, an $n$-DPM is a mapping from binary vectors of length $n$ to permutations of the same length such that if the Hamming distance of any two binary strings is $d$, then the Hamming distance of the corresponding permutations must be at least $d$. Furthermore, an $n$-DIM is a DPM such that when $d$ is less than $n$, the Hamming distance of the corresponding permutations must be larger than $d$. Once we have a DPM (respectively, DIM) $f$, for any binary code $C$ with minimum distance $r$, it is easy to see that $f(C)$ is a permutation array with minimum distance $r$ (respectively, $r + 1$).

In this paper, we focus on the code rate of $(n, r)$-permutation arrays. From this viewpoint, we can easily understand why DIM is better than DPM. In order to construct a permutation array with minimum distance $d$, we only need a good binary code with minimum distance $d - 1$ once we have an efficient DIM. We know that it is easier to construct a code with shorter minimum distance. Following this point, we may wish to construct a length-preserving mapping that can increase more distance. However, this is not an easy task. Instead of a length-preserving one, we construct a non-length-preserving mapping that can increase distance more than 1. Moreover, such mappings also can be used to construct a good permutation array of good code rate.

### 1.2 Notations

Let $P_n$ denote the set of all permutations of $[n]$ and $Z_q^n$ denote the set of all $q$-ary vectors of length $n$. For a permutation $\pi = (\pi_1, \ldots, \pi_n) \in P_n$, let $\pi(i) = \pi_i$ and $\pi_{[i\ldots j]}$ denote that subarray $(\pi_i, \ldots, \pi_j)$ of $\pi$. For $i \in [n]$, $\pi^{-1}(i)$ denotes the position of $i$ in $\pi$, i.e. if $\pi(j) = i$ then $\pi^{-1}(i) = j$. The Hamming distance $d_H(a, b)$ between two $n$-tuples $a = (a_1, a_2, \ldots, a_n)$ and $b = (b_1, b_2, \ldots, b_n)$ is the number of positions where they differ, i.e.

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

We now define a class of DIMS from $q$-ary vectors to permutations.

**Definition 1** For $d \leq n+k$, an $(n,d,k,q)$-mapping $f\colon Z_q^n \to P_{n+k}$ is a mapping such that, for all $x, y \in Z_q^n$,

$$d_H(f(x), f(y)) \geq d_H(x,y) + d, \qquad \text{if } d_H(x,y) \leq (n+k) - d \text{ and}$$
$$d_H(f(x), f(y)) = n+k, \qquad \text{if } d_H(x,y) > (n+k) - d.$$

Let $\mathcal{F}(n,d,k,q)$ denote the collection of all $(n,d,k,q)$-mappings. In particular, in the case that $q = 2$, we simply denote an $(n,d,k,2)$-mapping by an $(n,d,k)$-mapping and $\mathcal{F}(n,d,k,2)$ by $\mathcal{F}(n,d,k)$.

By using this notation, the collection of DPMs is equal to $\mathcal{F}(n,0,0)$ and the collection of DIMs is equal to $\mathcal{F}(n,1,0)$.

## 1.3 Previous results

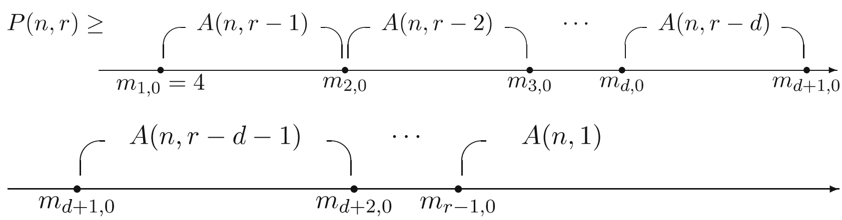
### *1.3.1 Length-preserving mappings*

The first construction of mappings in $\mathcal{F}(n,0,0)$ for $4 \leq n \leq 8$ was proposed by Ferreira and Vinck [7]. They found a mapping in $\mathcal{F}(4,0,0)$ by computer search. Then they used this mapping to construct mappings for $5 \leq n \leq 8$. Later, Chang et al. [2] extended their result to any integer $n \geq 4$ and they gave two kinds of recursively constructive mappings in $\mathcal{F}(n,0,0)$. One is that when given two mappings $g \in \mathcal{F}(m,0,0)$ and $h \in \mathcal{F}(n,0,0)$, they define $f\colon Z_2^{m+n} \to P_{m+n}$ as $f(x_1,\ldots,x_{m+n}) = (\pi_1,\ldots,\pi_m,\sigma_1+m,\ldots,\sigma_n+m)$, where $\pi = g(x_1,\ldots,x_m)$ and $\sigma = h(x_{m+1},\ldots,x_{m+n})$. Then $f$ is in $\mathcal{F}(m+n,0,0)$. Roughly speaking, it first concatenates the images of $g$ and $h$ then adjusts the values in the image of $h$. The other approach extends a mapping $g$ in $\mathcal{F}(n-1,0,0)$ one more dimension to construct an $f$ in $\mathcal{F}(n,0,0)$. i.e., suppose we have a permutation $\pi = (\pi_1,\pi_2,\ldots,\pi_{n-1}) \in P_{n-1}$ where $\pi = g(s)$ for some $s \in Z_2^{n-1}$. We extend $s$ one bit longer. If the extended bit is 0, then we replace the value at the $p$th entry with value $n$ and append the replaced value to the right of $\pi$, where $p$ can be any integer from 1 to $n-1$. If the extended bit is 1, then we just append value $n$ to the right of $\pi$. This construction gives us an $(n,0,0)$-mapping once we have mapping $g$. Recently Lee [8] gave an alternative algorithm for constructing an $(n,0,0)$-mapping for odd $n$.

Chang [1] gave the first constructive mappings in $\mathcal{F}(n,1,0)$ for $n \geq 4$. At the beginning of his construction, it does the same two steps as in the former construction of Chang et al. [2]. Then it starts to execute the swap operations: if $x_1 = 1$, swap $\pi_1$ and $\sigma_n + m$, and if $x_{m+1} = 1$, swap $\pi_n$ and $\sigma_1 + m$. These swap operations are used to remedy a bad situation that, given two strings, the first $m$ bits are exactly the same, but the last $n$ bits are totally different. In such a bad case, concatenation cannot produce a mapping in $\mathcal{F}(m+n,1,0)$ when given two mappings in $\mathcal{F}(n,1,0)$ and $\mathcal{F}(m,1,0)$, respectively. With the help of swap operations, Chang successfully construct an $(m+n,1,0)$-mapping [1]. For $d > 1$, one can easily extend Chang's method to construct inductively the mappings in $\mathcal{F}(n,d,0)$ for $n \geq n_0$ if $\mathcal{F}(n_0,d,0) \neq \emptyset$ for some integers $n_0$. Indeed, the construction is similar to the above construction for $\mathcal{F}(n,1,0)$ except for adding more swap operations. However, to find the basis case is really tough although we have the inductive construction for $\mathcal{F}(n,d,0)$ with $d > 1$. So far there is not much known on related construction.

Let $P(n, r)$ denote the maximum size of an $(n, r)$-permutation array and $A(n, r)$ denote the same setting for binary codes. In fact, $P(n, r)$ ($A(n, r)$, respectively) corresponds to the maximum code rate of an $(n, r)$-permutation array ($(n, r)$-binary code, respectively). In the works of Chang [1] and Chang et al. [2], they proved that $P(n, r) \geq A(n, r - 1)$ for $n \geq 4$ via $(n, 1, 0)$-mappings.

If one could construct a mapping in $\mathcal{F}(n, d, 0)$, then we would have $P(n, r) \geq A(n, r - d) > A(n, r - 1)$. In other words, the code rate of the $(n, r)$-permutation array constructed from $(n, d, 0)$-mappings increases as $d$ increases. Nevertheless, the least number $n_{d,0}$ with $\mathcal{F}(n_{d,0}, d, 0) \neq \emptyset$ also increases when $d$ increases. Therefore, there are many "gaps" in which the bound $P(n, r)$ cannot be improved via $(n, d, 0)$-mappings. To be more precise, let $n_{d,k}$ be the smallest integer such that for $n \geq n_{d,k}$, $\mathcal{F}(n, d, k)$ is not empty, and let $m_{d,k} = n_{d,k} + k$, i.e. the smallest image length. It is easy to see that $m_{d,k} \leq m_{d+1,k}$ and $A(n, r - d - 1) > A(n, r - d)$. Now let $k = 0$. When $n \geq m_{d+1,0}$, we can achieve $P(n, r) \geq A(n, r - d - 1)$. However when $m_{d,0} \leq n < m_{d+1,0}$, we can only achieve $P(n, r) \geq A(n, r - d)$. Hence for each "gap" $(m_{d,0}, m_{d+1,0})$, an $(n, d + 1, 0)$-mapping cannot help us improve the bound $P(n, r) \geq A(n, r - d)$ for $n$ in that gap. For convenience, we plot the bound $P(n, r)$ obtained via $(n, d, 0)$-mappings in the following diagram.




As we can see, in order to improve the bound $P(n, r)$ in those gaps, we need a different idea other than length-preserving mappings. One possible way is to relax the length-preserving constraint.

#### *1.3.2 Non-length-preserving mappings*

As mentioned above, to design an $(n, d, 0)$-mapping is harder than to design an $(n, d, 1)$-mapping. Until now we still do not know how to construct an $(n, d, 0)$-mapping for $d > 1$. Instead of $(n, 2, 0)$-mappings, Huang et al. (Submitted) gave the first construction for $(n, 2, 1)$-mappings. Their main observation is that it is easier to find a basis case for constructing $(n, d, 1)$-mappings than for $(n, d, 0)$-mappings for $d > 1$. Furthermore they observed that the code rate of the $(n, r)$-permutation array constructed by an $(n, 2, 1)$-mapping is greater than code rate by an $(n, 1, 0)$-mapping, that is $P(n, r) \geq A(n-1, r-2) \geq A(n, r-1)$. Note that $A(n - 1, r - 2) > A(n, r - 1)$ for even $n$. Therefore, their result strictly improved the previous results for even $n$. However, they cannot improve the bound $P(n, r) \geq A(n, r - 1)$ for odd $n$. In fact, $A(n, r - 1) = A(n - 1, r - 2)$ for odd $n$.

### 1.4 Our results

We give an algorithm which can construct a non-length-preserving mapping in $\mathcal{F}(n, 3, 2)$ for $n \geq 6$. This immediately gives us that $P(n, r) \geq A(n - 2, r - 3)$. Note that, for $n \geq 8$, $A(n - 2, r - 3) > A(n - 1, r - 2) = A(n, r - 1)$ when $n$ is even and $A(n - 2, r - 3) = A(n - 1, r - 2) > A(n, r - 1)$ when $n$ is odd. In both cases, our bound beats all previous bounds [1, 2]. In particular, our bound strictly improves the bound achieved by Chang [1].

In addition, our result combined with the result of Huang et al. submitted gives a better bound for $P(n, r)$ for the first "gap" $(m_{1,0}, m_{2,0})$ even when an $(n, 2, 0)$-mapping exists. This inspires us to construct $(n, d+1, 1)$-mappings for general $d$. Once we have such mappings, we can improve the lower bound for $P(n, r)$ in the gap $(m_{d,0}, m_{d+1,0})$ even when an $(n, d+1, 0)$-mapping indeed exists. However, we only can give the improvement for the first "gap" until now.

### 1.5 Organization of this paper

First of all, for completeness, we include the result of Huang et al. (submitted) in Section 2. In Section 3, we give our main construction of an $(n, 3, 2)$-mapping. Then we show the bound of code size when applying $(n, 3, 2)$-mappings to the construction of an $(n, r)$-permutation array in Section 4. In Section 5, we conclude with some open problems.

## 2 Construction of $\mathcal{F}(n, 2, 1)$

We will need four mappings as our basis constructions, i.e., $g_6 \in \mathcal{F}(6, 2, 1)$, $g_7 \in \mathcal{F}(7, 2, 1)$, $g_8 \in \mathcal{F}(8, 2, 1)$ and $g_9 \in \mathcal{F}(9, 2, 1)$. Then we inductively construct $g_{n+4} \in \mathcal{F}(n+4, 2, 1)$ from a mapping $g_n \in \mathcal{F}(n, 2, 1)$. Thus we give a construction of $(n, 2, 1)$-mappings for $n \geq 6$. Here, we only give the construction of $g_6$, since the other constructions are very similar and are included in the appendix.

We also need two auxiliary mappings, i.e., $A_6 \in \mathcal{F}(2, 2, 2)$ and $B_6 \in \mathcal{F}(4, 2, 2)$ with certain properties which will be defined later. We construct $g_6$ with these two mappings.

**Definition 2** We say that a mapping $f \in \mathcal{F}(n, d, k)$ has the position property for $\{v_1, v_2, \ldots, v_p\} \subseteq \{1, 2, \ldots, n+k\}$ if for any $i \in \{1, \ldots, p\}$, we have $|\{\pi^{-1}(v_i) | \pi \in f(Z_2^n)\}| = 2$ and for any $i, j$ with $1 \leq i < j \leq p$, we have $\{\pi^{-1}(v_i) | \pi \in f(Z_2^n)\} \cap \{\pi^{-1}(v_j) | \pi \in f(Z_2^n)\} = \emptyset$.

The following mapping $A_6$ has the position property for $\{4\}$ and $B_6$ has the position property for $\{1, 2\}$. With their position properties, we can construct a $(6, 2, 1)$-mapping $g_6$.

**Construction 1** (Basis Construction) Let $A_6 : Z_2^2 \to P_4$ and $B_6 : Z_2^4 \to P_6$ be defined as follows:

| $x$ | $A_6(x)$ | $x$ | $A_6(x)$ |
|---|---|---|---|
| 00 | (1, 4, 3, 2) | 10 | (4, 2, 3, 1) |
| 01 | (2, 4, 1, 3) | 11 | (4, 1, 2, 3) |

| $x$ | $B_6(x)$ | $x$ | $B_6(x)$ |
|---|---|---|---|
| 0000 | (1, 3, 2, 4, 5, 6) | 1000 | (3, 1, 2, 5, 4, 6) |
| 0001 | (1, 3, 2, 5, 6, 4) | 1001 | (3, 1, 2, 4, 6, 5) |
| 0010 | (1, 3, 5, 2, 4, 6) | 1010 | (3, 1, 4, 2, 5, 6) |
| 0011 | (1, 3, 4, 2, 6, 5) | 1011 | (3, 1, 5, 2, 6, 4) |
| 0100 | (1, 5, 2, 6, 4, 3) | 1100 | (4, 1, 2, 6, 5, 3) |
| 0101 | (1, 4, 2, 6, 3, 5) | 1101 | (5, 1, 2, 6, 3, 4) |
| 0110 | (1, 4, 6, 2, 5, 3) | 1110 | (5, 1, 6, 2, 4, 3) |
| 0111 | (1, 5, 6, 2, 3, 4) | 1111 | (4, 1, 6, 2, 3, 5) |

*Input:* $(x_1, x_2, \ldots, x_6) \in Z_2^6$
*Output:* $(\pi_1, \ldots, \pi_7) = g_6(x_1, \ldots, x_6) \in \mathcal{F}(6, 2, 1)$
*begin*
*0* $\rho = A_6(x_1, x_2); \tau = B_6(x_3, x_4, x_5, x_6);$
*1* $\tau_i = \tau_i + 1$ for $1 \le i \le 6$;
*2* $\rho_{\rho^{-1}(4)} = \tau_6;$
*3* $\tau_{\tau^{-1}(2)} = \rho_3$; * Step 3 and 4 are meant to be assigned simultaneously
*4* $\tau_{\tau^{-1}(3)} = \rho_4$; * Similarly, for the rest of constructions.
*5* $(\pi_1, \pi_2) = \rho_{[1..2]};$
*6* $(\pi_3, \pi_4, \pi_5, \pi_6, \pi_7) = \tau_{[1..5]};$
*end*

In addition to the position property, $B_6$ holds another property, i.e., if the Hamming distance of two binary vectors is 3, then the fifth entries of the images must be different. In other words, for $x, y \in Z_2^4$, if $d_H(x, y) = 3$, then $B_6(x)_5 \neq B_6(y)_5$.

Given $g \in \mathcal{F}(n, 2, 1)$, let $D_{n\times(n+1)}$ denote the distance expansion matrix where $D_{ij}$ represents the number of all unordered pairs $\{x, y\}$, $x, y \in Z_2^n$ such that $d_H(x, y) = i$ and $d_H(g(x), g(y)) = j$. Using this matrix, one can easily check if the mapping satisfies the required distance constraint. We show the distance expansion matrix for $g_6$ as follows:

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 128 | 64 | 0 | 0 | 0 |
| | 0 | 0 | 232 | 88 | 120 | 40 |
| | | 0 | 0 | 160 | 384 | 96 |
| | | | 0 | 0 | 192 | 288 |
| | | | | 0 | 0 | 192 |
| | | | | | 0 | 32 |

Next, we show how to construct a mapping $g_{n+4} \in \mathcal{F}(n + 4, 2, 1)$ from a mapping $g_n \in \mathcal{F}(n, 2, 1)$.

**Algorithm 1** (*Inductive Step*)

Input: $(x_1, \ldots, x_n, \ldots, x_{n+4}) \in Z_2^{n+4}$
Output: $(\pi_1, \ldots, \pi_{n+5}) = g_{n+4}(x_1, \ldots, x_{n+4})$
begin
0 $\rho = g_n(x_1, \ldots, x_n); \tau = B_6(x_{n+1}, x_{n+2}, x_{n+3}, x_{n+4});$
1 $\tau_i = \tau_i + n - 1$, for $1 \le i \le 6$;
2 $\tau_{\tau^{-1}(n)} = \rho_n;$
3 $\tau_{\tau^{-1}(n+1)} = \rho_{n+1};$
4 $(\pi_1, \ldots, \pi_{n-1}) = \rho_{[1..n-1]};$
5 $(\pi_n, \ldots, \pi_{n+5}) = \tau_{[1..6]};$
6 if $x_1 = 1$ then swap $(\pi_1, \pi_{n+4})$;
7 if $x_2 = 1$ then swap $(\pi_2, \pi_{n+5})$;
end

**Theorem 1** [8] *If* $g_n \in \mathcal{F}(n, 2, 1)$*, then* $g_{n+4} \in \mathcal{F}(n + 4, 2, 1)$ *for* $n \ge 6$.

*Proof* Given $(x, w)$ and $(y, z) \in Z_2^{n+4}$ where $x, y \in Z_2^n$ and $w, z \in Z_2^4$, let $g_n(x) = \rho = (\rho_1, \ldots, \rho_{n+1})$, $g_n(y) = \rho' = (\rho'_1, \ldots, \rho'_{n+1})$, $B_6(w) = \tau = (\tau_1, \ldots, \tau_6)$, and $B_6(z) = \tau' = (\tau'_1, \ldots, \tau'_6)$. We also have $g_{n+4}(x, w) = \pi = (\pi_1, \ldots, \pi_{n+5})$ and $g_{n+4}(y, z) = \pi' = (\pi'_1, \ldots, \pi'_{n+5})$.

**Claim 1** *After executing line 5, we have* $d_H(\pi_{[n\ldots n+5]}, \pi'_{[n\ldots n+5]}) \ge d_H(B_6(w), B_6(z))$.

*Proof* (of Claim 1) Since $B_6$ has the position property for $\{1,2\}$, we have $\{\tau^{-1}(1)|\tau \in B_6(Z_2^4)\} \cap \{\tau^{-1}(2)|\tau \in B_6(Z_2^4)\} = \emptyset$. Therefore, the effect caused by line 2 is independent of line 3. We only discuss the change of distance caused by line 2 since the argument is similar for Line 3. Note that, after executing line 1, $\rho_i, \rho_i' \in \{1,\ldots,n+1\}$ for $i = 1$–$n+1$ and $\tau_i, \tau_i' \in \{n,\ldots,n+5\}$ for $i = 1$–$6$. After executing line 2, there are only two possible cases. For the case when $\tau^{-1}(n) = \tau'^{-1}(n) = k$, we have $\tau_k = \tau_k' = n$. Moreover, $\tau_k$ and $\tau_k'$ are replaced by $\rho_n$ and $\rho_n'$, respectively. It is clear that $d_H(\rho_n, \rho_n') \geq d_H(\tau_k, \tau_k') = 0$. The other case is when $\tau^{-1}(n) = k$ and $\tau'^{-1}(n) = k'$ for $k \neq k'$, and hence we have $\tau_k = \tau_{k'}' = n$ and $d_H(\tau_k, \tau_k') + d_H(\tau_{k'} + \tau_{k'}') = 2$ before executing line 2. Moreover, $\tau_{k'}$ and $\tau_k'$ are in $\{n+2,\ldots,n+5\}$ since $B_6$ has the position property for $\{1,2\}$. Since $\rho_n$ and $\rho_n'$ are in $\{1,\ldots,n+1\}$, after executing line 2, we have $d_H(\tau_k, \tau_k') + d_H(\tau_{k'}, \tau_{k'}') = d_H(\rho_n, \tau_k') + d_H(\tau_{k'}, \rho_n') = 2$. Therefore, the distance is preserved. □

Now we discuss how these values change after executing lines 6 and 7 case by case.

- Case $[d_H(x,y) = 0]$: we know that $d_H(w,z) \neq 0$, otherwise $(x,w)$ and $(y,z)$ are identical. In this case, both of the two strings have the same operations in lines 6 and 7. Note that values 1 and 2 only appear in the first four coordinates of the images of $B_6$. Therefore, the values in the fifth and sixth coordinate are not affected by those operations before lines 6. Thus, after executing the swap operation, the distance is still preserved. More precisely, let $d_H(w,z) = t \leq 4$. Since, $B_6 \in \mathcal{F}(4,2,2)$, we have $d_H(\tau,\tau') \geq t+2$. Then, we get $d(\pi,\pi')) \geq t+2 = d_H((x,w),(y,z)) + 2$.
- Case $[0 < d_H(x,y) = s < n]$: it is clear that $d_H(\rho,\rho') \geq s+2$ since $g_n \in \mathcal{F}(n,2,1)$. So we have $d_H(\pi_{[1\ldots n-1]}, \pi'_{[1\ldots n-1]}) \geq s$. If $0 < d_H(w,z) = t$, then $d_H(\tau,\tau') \geq t+2$ since $B_6 \in \mathcal{F}(4,2,2)$. By Claim 1, we can get $d_H(\pi,\pi') = d_H(\pi_{[1\ldots n-1]}, \pi'_{[1\ldots n-1]}) + d_H(\pi_{[n\ldots n+5]}, \pi'_{[n\ldots n+5]}) \geq s + (t+2) = d_H((x,w),(y,z)) + 2$. For $d_H(w,z) = 0$, it is easy to see that $d_H(\pi,\pi') \geq s+2$.
- Case $[d_H(x,y) = n]$: in this case, it is clear that $d_H(\rho,\rho') = n+1$ since $g_n \in \mathcal{F}(n,2,1)$. Let $d_H(w,z) = t$. We know that $d_H(\pi_{[1\ldots n-1]}, \pi'_{[1\ldots n-1]}) = d_H(\rho_{[1\ldots n-1]}, \rho'_{[1\ldots n-1]}) = n-1$ and $d_H(\pi_{[n\ldots n+5]}, \pi'_{[n\ldots n+5]}) \geq d_H(\tau,\tau') \geq t+2$ (even when $t = 0$). Hence, we get $d_H(\pi,\pi') \geq n+t+1$. We argue that this lower bound is indeed at least $n+t+2$ except for $t = 4$. We divide the argument into two subcases according to $t = d_H(w,z)$.

1. Subcase $[t = 4]$: clearly we have $d_H(\tau,\tau') = 6$. It is easy to see that $d_H(\pi,\pi') = n+5$.
2. Subcase $[0 \leq t \leq 3]$: if $d_H(\tau,\tau') = 6$, then $d_H(\pi,\pi') = n+5 \geq n+t+2 = d_H((x,w),(y,z)) + 2$. if $d_H(\tau,\tau') \leq 5$, there must be one coordinate $i$ in which $\tau_i = \tau_i'$. Note that $x_1 \neq y_1$ and $x_2 \neq y_2$ since $d_H(x,y) = n$. So these two strings must have different operations in both lines 6 and 7. If $i = 5$ or 6 then, after executing the swap operations in lines 6 and 7, $d_H(\pi_{[n\ldots n+5]}, \pi'_{[n\ldots n+5]}) \geq t+3$. So $d_H(\pi,\pi') \geq n+t+2$. If $i = 1$ or 2, the value 1 in $\tau$ and in $\tau'$ must lie in the same coordinate. Thus, we have $d_H(\pi_{[n\ldots n+5]}, \pi'_{[n\ldots n+5]}) \geq t+3$. Same argument holds for $i = 3$ or 4.

This completes the proof of correctness. □

## 3 Construction of $\mathcal{F}(n,3,2)$

The approach for the construction of an $(n,3,2)$-mapping is similar to the construction of an $(n,2,1)$-mapping. We first give five basis constructions: $h_6 \in \mathcal{F}(6,3,2)$, $h_7 \in \mathcal{F}(7,3,2)$, $h_8 \in \mathcal{F}(8,3,2)$, $h_9 \in \mathcal{F}(9,3,2)$ and $h_{10} \in \mathcal{F}(10,3,2)$. Then we give the induction method for constructing an $(n+5,3,2)$-mapping $h_{n+5}$ from an $(n,3,2)$-mapping $h_n$. Therefore, we can construct an $(n, 3, 2)$-mapping for each $n \geq 6$. Here, we only show how to construct $h_6$ and leave the other four constructions in the appendix.

We will use an auxiliary mapping $C_6$ for constructing $h_6$. $C_6$ has the position property for $\{1, 2\}$.

**Construction 2** (Basis Construction) Let $C_6$: $Z_2^3 \to P_6$ be the following mapping:

| $x$ | $C_6(x)$ | $x$ | $C_6(x)$ |
|---|---|---|---|
| 000 | (1, 3, 2, 4, 5, 6) | 100 | (4, 1, 2, 6, 5, 3) |
| 001 | (1, 4, 2, 3, 6, 5) | 101 | (3, 1, 2, 5, 6, 4) |
| 010 | (1, 5, 3, 2, 4, 6) | 110 | (5, 1, 6, 2, 4, 3) |
| 011 | (1, 6, 4, 2, 3, 5) | 111 | (6, 1, 5, 2, 3, 4) |

*Input*: $(x_1, x_2, \ldots, x_6) \in Z_2^6$
*Output*: $(\pi_1, \pi_2, \ldots, \pi_8) = h_6(x_1, \ldots, x_6) \in \mathcal{F}(6, 2, 1)$
*begin*
$\rho = C_6(x_1, x_2, x_3)$; $\tau = C_6(x_4, x_5, x_6)$;
$\rho_i = \rho_i - 2$, for $1 \leq i \leq 6$;
$\tau_i = \tau_i + 2$, for $1 \leq i \leq 6$;
$\rho_{\rho^{-1}(-1)} = \tau_5$;
$\rho_{\rho^{-1}(0)} = \tau_6$;
$\tau_{\tau^{-1}(3)} = \rho_5$;
$\tau_{\tau^{-1}(4)} = \rho_6$;
$(\pi_1, \ldots, \pi_4) = \rho_{[1\ldots4]}$;
$(\pi_5, \ldots, \pi_8) = \tau_{[1\ldots4]}$;
*end*

There is no further restriction on $C_6$. In fact, we only need its position property. The distance expansion matrix for $h_6$ is as follows.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 192 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 128 | 128 | 128 | 96 |
| | | 0 | 0 | 0 | 64 | 128 | 448 |
| | | | 0 | 0 | 0 | 0 | 480 |
| | | | | 0 | 0 | 0 | 192 |
| | | | | | 0 | 0 | 32 |

Next we show how to construct a mapping $h_{n+5} \in \mathcal{F}(n + 5, 3, 2)$ inductively from a mapping $h_n \in \mathcal{F}(n, 3, 2)$. We need the mapping $E_8$: $Z_2^5 \to P_8$ defined as follows.

| $x$ | $E_8(x)$ | $x$ | $E_8(x)$ |
|---|---|---|---|
| 00000 | (1, 8, 2, 4, 3, 5, 6, 7) | 10000 | (8, 1, 2, 4, 3, 5, 6, 7) |
| 00001 | (1, 7, 2, 4, 3, 6, 5, 8) | 10001 | (7, 1, 2, 4, 3, 6, 5, 8) |
| 00010 | (1, 8, 2, 4, 5, 3, 7, 6) | 10010 | (8, 1, 2, 4, 5, 3, 7, 6) |
| 00011 | (1, 7, 2, 4, 6, 3, 8, 5) | 10011 | (7, 1, 2, 4, 6, 3, 8, 5) |
| 00100 | (1, 5, 2, 6, 3, 8, 4, 7) | 10100 | (5, 1, 2, 6, 3, 8, 4, 7) |
| 00101 | (1, 6, 2, 5, 3, 7, 4, 8) | 10101 | (6, 1, 2, 5, 3, 7, 4, 8) |
| 00110 | (1, 5, 2, 6, 8, 3, 7, 4) | 10110 | (5, 1, 2, 6, 8, 3, 7, 4) |
| 00111 | (1, 6, 2, 5, 7, 3, 8, 4) | 10111 | (6, 1, 2, 5, 7, 3, 8, 4) |
| 01000 | (1, 4, 5, 2, 3, 8, 7, 6) | 11000 | (4, 1, 5, 2, 3, 8, 7, 6) |
| 01001 | (1, 4, 6, 2, 3, 7, 8, 5) | 11001 | (4, 1, 6, 2, 3, 7, 8, 5) |
| 01010 | (1, 4, 5, 2, 8, 3, 6, 7) | 11010 | (4, 1, 5, 2, 8, 3, 6, 7) |
| 01011 | (1, 4, 6, 2, 7, 3, 5, 8) | 11011 | (4, 1, 6, 2, 7, 3, 5, 8) |

| | | | |
|---|---|---|---|
| 01100 | (1, 5, 7, 2, 3, 8, 6, 4) | 11100 | (5, 1, 7, 2, 3, 8, 6, 4) |
| 01101 | (1, 6, 8, 2, 3, 7, 5, 4) | 11101 | (6, 1, 8, 2, 3, 7, 5, 4) |
| 01110 | (1, 5, 7, 2, 8, 3, 4, 6) | 11110 | (5, 1, 7, 2, 8, 3, 4, 6) |
| 01111 | (1, 6, 8, 2, 7, 3, 4, 5) | 11111 | (6, 1, 8, 2, 7, 3, 4, 5) |

The mapping $E_8$ is produced in the following way. We first find a mapping $E_7 \in \mathcal{F}(4, 3, 3)$ with the position property for $\{1, 2\}$. Moreover value 1 only appears in the second or third coordinate and value 2 only appears in the fourth or fifth coordinate. Then, we add 1 to each entry of all the permutations. Therefore, we form a permutation $\pi_{[1...7]}$ of $\{2, \ldots, 8\}$. Finally, we define $E_8 : Z_2^5 \to P_8$ such that, for all $w \in Z_2^4$, $E_8(0w) = (1, \pi_1, \pi_2, \ldots, \pi_7))$ and $E_8(1w) = (\pi_1, 1, \pi_2, \ldots, \pi_7))$ where $\pi = E_7(w)$. It is easy to check that, for all distinct strings $x, y \in Z_2^5$, if $d_H(x, y) = d$ then $d_H(E_8(x), E_8(y)) \geq d + 3$ except when $x$ and $y$ differ only at the first bit. In such a case, we have $d_H(x, y) = 1$ but $d_H(E_8(x), E_8(y)) = 2$. We give the inductive algorithm below and then its proof of correctness.

**Algorithm 2** *(Inductive Step)*
Input: $(x_1, \ldots, x_n, \ldots, x_{n+5}) \in Z_2^{n+5}$
Output: $(\pi_1, \ldots, \pi_{n+7}) = h_{n+5}(x_1, \ldots, x_{n+5})$
begin
0 $\rho = h_n(x_1, \ldots, x_n)$; $\tau = E_8(x_{n+1}, \ldots, x_{n+5})$;
1 $\tau_i = \tau_i + n - 1$, for $1 \leq i \leq 8$;
2 $\tau_{\tau^{-1}(n)} = \rho_n$;
3 $\tau_{\tau^{-1}(n+1)} = \rho_{n+1}$;
4 $\tau_{\tau^{-1}(n+2)} = \rho_{n+2}$;
5 $(\pi_1, \ldots, \pi_{n-1}) = \rho_{[1...n-1]}$;
6 $(\pi_n, \ldots, \pi_{n+7}) = \tau_{[1...8]}$;
7 if $x_1 = 1$ then swap $(\pi_1, \pi_{n+6})$;
8 if $x_{n+1} = 1$ then swap $(\pi_2, \pi_{n+7})$.

**Theorem 2** *If $h_n \in \mathcal{F}(n, 3, 2)$, then $h_{n+5} \in \mathcal{F}(n + 5, 3, 2)$ for $n \geq 6$.*

*Proof* Given $x, y \in Z_2^n$ and $w, z \in Z_2^5$, let $h_n(x) = \rho = (\rho_1, \ldots, \rho_{n+2})$, $h_n(y) = \rho' = (\rho'_1, \ldots, \rho'_{n+2})$, $E_8(w) = (\tau_1, \tau_2, \ldots, \tau_8)$ and $E_8(z) = (\tau'_1, \tau'_2, \ldots, \tau'_8)$. □

**Claim 2** *After executing line 6, we have $d_H(\pi_{[n...n+7]}, \pi'_{[n...n+7]}) \geq d_H(E_8(w), E_8(z))$.*

*Proof* (of Claim 2) Note that $E_8$ has the position property for $\{1, 2, 3\}$. Moreover value 1 only appears in the first or second coordinate, value 2 only appears in the third or fourth coordinate and value 3 only appears in the fifth or sixth coordinate. Therefore the effect caused by line 2 is independent of lines 3 and 4. We only discuss the change of distance caused by Line 2, since the argument is similar for lines 3 and 4. Note that, after executing line 1, $\rho_i, \rho'_i \in \{1, \ldots, n+1\}$ for $i = 1 - n+1$ and $\tau_i, \tau'_i \in \{n, \ldots, n+7\}$ for $i = 1 - 8$. After executing line 2, there are only two possible cases. The first case is when $\tau^{-1}(n) = \tau'^{-1}(n) = k$, and we have $\tau_k = \tau'_k = n$. Moreover, $\tau_k$ and $\tau'_k$ are replaced by $\rho_n$ and $\rho'_n$, respectively. It is clear that $d_H(\rho_n, \rho'_n) \geq d_H(\tau_k, \tau'_k) = 0$. The second case is when $\tau^{-1}(n) = k$ and $\tau'^{-1}(n) = k'$ for $k \neq k'$, and hence, we have $\tau_k = \tau'_{k'} = n$ and $d_H(\tau_k, \tau'_k) + d_H(\tau_{k'} + \tau'_{k'}) = 2$ before executing line 2. Moreover, $\tau_{k'}$ and $\tau'_k$ are in $\{n + 3, \ldots, n + 7\}$, since $E_8$ has the position property for $\{1, 2, 3\}$. Also because $\rho_n$ and $\rho'_n$ are in $\{1, \ldots, n + 1\}$, after executing line 2, we have $d_H(\tau_k, \tau'_k) + d_H(\tau_{k'}, \tau'_{k'}) = d_H(\rho_n, \tau'_k) + d_H(\tau_{k'}, \rho'_n) = 2$. Therefore, the distance is preserved.

Now we discuss how these values change after executing lines 7 and 8 case by case.

- Case [$d_H(x, y) = 0 (x = y)$]: we know that $d_H(w, z) \neq 0$, otherwise $(x, w)$ and $(y, z)$ are identical. In this case, both of the two strings have the same operations in lines 7 and 8. Note that values 1–3 only appear in the first six coordinates of the images of $E_8$. Therefore, the values in the seventh and eighth coordinate are not affected by those operations before line 6. Therefore, after executing the swap operation, the distance is still preserved. Let $d_H(w, z) = t \leq 5$. We discuss the following subcases.
  1. Subcase [$D_H(w, z) = t \geq 2$]: by the property of $E_8$, we have $d_H(E_8(w), E_8(z)) \geq t + 3$. Therefore, we get $d(\pi, \pi')) \geq t + 3 = d_H((x, w), (y, z)) + 3$.
  2. Subcase [$D_H(w, z) = 1$ and $w_1 = z_1$]: we can use similar argument as in the above subcase.
  3. Subcase [$D_H(w, z) = 1$ and $w_1 \neq z_1$]: in this subcase, $w_{[2...5]} = z_{[2...5]} = u$. Suppose $E_8(0u) = (1, \sigma_1, \sigma_2, \ldots, \sigma_7)$ and $E_8(1u) = (\sigma_1, 1, \sigma_2, \ldots, \sigma_7)$ where $\sigma_{[1...7]}$ is a permutation of $\{2, \ldots, 8\}$. Since the the string $(x, 1u)$ triggers the swap operation at line 8 but the string $(x, 0u)$ does not, we have $\pi_{n+7} \neq \pi'_{n+7}$ after executing line 8. Therefore, we have $\pi_2 \neq \pi'_2$, $\pi_{n+7} \neq \pi'_{n+7}$, $\pi_n \neq \pi'_n$ and $\pi_{n+1} \neq \pi'_{n+1}$. Hence the distance increases by 3.
- Case [$0 < d_H(x, y) = s < n$]: it is clear that $d_H(\rho, \rho') \geq s + 3$ since $h_n \in \mathcal{F}(n, 3, 2)$. So we have $d_H(\pi_{[1...n-1]}, \pi'_{[1...n-1]}) \geq s$. We discuss the following subcases.
  1. Subcase [$D_H(w, z) = t \geq 2$]: by the property of $E_8$, we have $d_H(E_8(w), E_8(z)) \geq t + 3$. By Claim 2, we get $d_H(\pi, \pi') = d_H(\pi_{[1...n-1]}, \pi'_{[1...n-1]}) + d_H(\pi_{[n...n+5]}, \pi'_{[n...n+5]}) \geq s + (t + 3) = d_H((x, w), (y, z)) + 3$.
  2. Subcase [$D_H(w, z) = 1$ and $w_1 = z_1$]: the argument is similar to the above subcase.
  3. Subcase [$D_H(w, z) = 1$ and $w_1 \neq z_1$]: in this subcase, $w_{[2...5]} = z_{[2...5]} = u$. Suppose $E_8(0u) = (1, \sigma_1, \sigma_2, \ldots, \sigma_7)$ and $E_8(1u) = (\sigma_1, 1, \sigma_2, \ldots, \sigma_7)$ where $\sigma_{[1...7]}$ is a permutation of $\{2, \ldots, 8\}$. Without loss of generality, we only discuss the output strings of $(x, 0u)$ and $(y, 1u)$. Since the the string $(y, 1u)$ triggers the swap operation at line 8 but the string $(x, 0u)$ does not, we have $\pi_{n+7} \neq \pi'_{n+7}$ and $\pi_2 \neq \pi'_2$ after executing line 8. Furthermore, we have $\pi_n \neq \pi'_n$ and $\pi_{n+1} \neq \pi'_{n+1}$. Hence, the distance increases by 3.
  4. Subcase [$D_H(w, z) = 0$]: it is clear that $d_H(\pi, \pi') \geq s + 3$.
- Case [$d_H(x, y) = n$]: in this case, it is clear that $d_H(\rho, \rho') = n + 2$ since $h_n \in \mathcal{F}(n, 3, 2)$. Let $d_H(w, z) = t$. We discuss the following subcases.
  1. Subcase [$D_H(w, z) = t = 5$]: by the property of $E_8$, we have $d_H(E_8(w), E_8(z)) = 8$. It is easy to see that $d_H(\pi, \pi') = n + 7$.
  2. Subcase [$5 > D_H(w, z) = t \geq 2$]: in this subcase, we have $d_H(E_8(w), E_8(z)) \geq t + 3$ by the property of $E_8$. If $d_H(\tau, \tau') = 8$, then $d_H(\pi, \pi') = n + 7 \geq n + t + 3 = d_H((x, w), (y, z)) + 3$. If $d_H(\tau, \tau') \leq 7$, we must have $d_H(\tau, \tau') \leq 6$ since $\tau$ and $\tau'$ are permutations. So there must exist two coordinates $i, j$ in which $\tau_i = \tau'_i$ and $\tau_j = \tau'_j$. Note that $x_1 \neq y_1$ since $d_H(x, y) = n$. So these two strings must have different operations in line 7. If $(i, j) = (7, 8)$ or $(8, 7)$ then, after executing the swap operations in line 7, $d_H(\pi_{[n...n+5]}, \pi'_{[n...n+5]}) \geq t + 4$. So $d_H(\pi, \pi') = d_H(\pi_{[1...n-1]}, \pi'_{[1...n-1]}) = d_H(\rho_{[1..n-1]}, \rho'_{[1..n-1]}) \geq n - 1 + t + 4 = n + t + 3$. The other case is that there must exist an index $k \leq 6$ such that $\tau_k = \tau'_k$. If $k = 1$ or 2, the value 1 in $\tau$ and in $\tau'$ must lie in the same coordinate. Thus we have $d_H(\pi_{[n...n+7]}, \pi'_{[n...n+7]}) \geq t + 4$. Same argument holds for $k = 3, 4, 5$ or 6.
  3. Subcase [$D_H(w, z) = 1$ and $w_1 = z_1$]: it can be proved with similar argument as in the above subcase.

4. Subcase [$D_H(w, z) = 1$ and $w_1 \neq z_1$]: in this subcase, $w_{[2...5]} = z_{[2...5]} = u$. Suppose $E_8(0u) = (1, \sigma_1, \sigma_2, \ldots, \sigma_7)$ and $E_8(1u) = (\sigma_1, 1, \sigma_2, \ldots, \sigma_7)$ where $\sigma_{[1...7]}$ is a permutation of $\{2, \ldots, 8\}$. Without loss of generality, we only discuss the output strings of $(x, 0u)$ and $(y, 1u)$. Since the the string $(y, 1u)$ triggers the swap operation at line 8 but the string $(x, 0u)$ does not, we have $\pi_{n+7} \neq \pi'_{n+7}$ after executing line 8. Furthermore, we have $\pi_n \neq \pi'_n$ and $\pi_{n+1} \neq \pi'_{n+1}$. Finally note that value 2 (value 3, respectively) only appears in the third or fourth (fifth or sixth, respectively) coordinate. After executing lines 3 and 4, the distance $d_H(\pi_{[n+2...n+5]}, \pi'_{[n+2...n+5]})$ must increase by 2 since $\rho_{n+1} \neq \rho'_{n+1}$ and $\rho_{n+2} \neq \rho'_{n+2}$. Hence, the distance $d_H(\pi_{[n...n+7]}, \pi'_{[n...n+7]})$ must increases by 5. Finally, we get $d_H(\pi, \pi') \geq n - 1 + 5 = n + 4$.
5. Subcase [$D_H(w, z) = 0$]: note that $x_1 \neq y_1$ since $d_H(x, y) = n$. So these two strings $\tau$ and $\tau'$ must have different operations in line 7. So, we have $\pi_{n+7} \neq \pi'_{n+7}$. Furthermore, since $E_8$ has the position property for $\{1, 2, 3\}$, it is clear that $d_H(\pi, \pi') \geq n - 1 + 4 = n + 3$.

This completes the correctness proof. □

## 4 Applications to permutation arrays

Chang [1] Chang et al. [2] constructed an $(n, 1, 0)$-mapping and such a mapping can be used to construct good permutation arrays (easy to encode and decode) provided that we have a good binary code. Furthermore Huang et al. (Submitted) proposed a construction of an $(n, 2, 1)$-mapping which can be used to build a good permutation array with better code rate than Chang's construction [1]. In this paper, we construct an $(n, 3, 2)$-mapping. Thus, we can construct an $(n, r)$-permutation array of code size at least $A(n - 2, r - 3)$ when we have an $(n, r)$-binary code with the best code rate. This bound beats previous bounds for $n \geq 8$. We state our result in the following corollary.

Recall that the term $n_{d,k}$ is defined to be the smallest integer $n$ such that $\mathcal{F}(n, d, k)$ is not empty and $m_{d,k} = n_{d,k} + k$. We have the following bound.

**Theorem 3** *For $n \geq m_{d,k}$ and $d + 1 \leq r \leq n$, $P(n, r) \geq A(n - k, r - d)$.*

*Proof* Let $C$ be a binary code of length $n - k$ with minimum distance $r - d$. Since, $n \geq m_{d,k}$, then $n - k \geq n_{d,k}$. Thus, we have a mapping $f \in \mathcal{F}(n - k, d, k)$. From the definition, we know that $f(C)$ is an $(n, r)$-permutation array. Thus, $P(n, r) \geq |C|$. Therefore, $P(n, r) \geq A(n - k, r - d)$. □

Theorem 3 tells us that if we have an efficient $(n, d, k)$-mapping and an $(n - k, r - d)$-binary code with the best code rate, then we can get an efficient $(n, r)$-permutation array of size at least $A(n - k, r - d)$. Since, we give a construction of an $(n, 3, 2)$-mapping for $n \geq 6$, we immediately have the following corollary.

**Corollary 1** $P(n, r) \geq A(n - 2, r - 3) > A(n, r - 1)$ *for* $n \geq 8$.

*Proof* The first inequality holds since $\mathcal{F}(n, 3, 2) \neq \emptyset$ for $n \geq 6$. The second inequality holds because $A(n - 2, r - 3) > A(n - 1, r - 2) = A(n, r - 1)$ for odd $n$ and $A(n - 2, r - 3) = A(n - 1, r - 2) > A(n, r - 1)$ for even $n$. Therefore, $A(n - 2, r - 3) > A(n, r - 1)$ for $n \geq 8$. □

## 5 Conclusion and open problems

We show how to construct $(n, 3, 2)$-mappings for $n \geq 6$. This result gives a better bound for $P(n, r) \geq A(n - 2, r - 3) > A(n, r - 1)$ for $n \geq 8$. Furthermore this bound beats all the previous bounds [1, 2, Huang et al. (Submitted)].

As we mentioned in the introduction, our result improves the bound $P(n, r)$ in the first gap even if one knows the construction of $(n, 2, 0)$-mappings, which is not clear so far. How to design an $(n, d + 1, 1)$-mapping is an interesting problem. Since, the smallest number $n_{d+1,1}$ such that $\mathcal{F}(n_{d+1,1}, d + 1, 1) \neq \emptyset$ must be smaller than the smallest number $n_{d+1,0}$ such that $\mathcal{F}(n_{d+1,0}, d+1, 0) \neq \emptyset$, the $(n, d+1, 1)$-mappings will help us improve the bound $P(n, r)$ for $n$ in the interval $(m_{d,0}, m_{d+1,0})$ even when an $(n, d + 1, 0)$-mapping does exist.

## Appendix A Constructions of $g_7, g_8, g_9$

**Construction 3** Let $A_7$: $Z_2^3 \rightarrow P_5$ be defined as follows and $B_7$ be the same as $B_6$:

| $x$ | $A_7(x)$ | $x$ | $A_7(x)$ |
|---|---|---|---|
| 000 | (1, 5, 3, 4, 2) | 100 | (5, 2, 1, 4, 3) |
| 001 | (1, 5, 4, 2, 3) | 101 | (5, 3, 2, 4, 1) |
| 010 | (2, 5, 3, 1, 4) | 110 | (5, 4, 1, 3, 2) |
| 011 | (2, 5, 4, 3, 1) | 111 | (5, 1, 2, 3, 4) |

By the following algorithm, $g_7 \in \mathcal{F}(7, 2, 1)$ is constructed.

*Input*: $(x_1, x_2, \ldots, x_7) \in Z_2^7$

*Output*: $(\pi_1, \ldots, \pi_8) = g_7(x_1, \ldots, x_7)$

*begin*

$\rho = A_7(x_1, x_2, x_3)$; $\tau = B_7(x_4, x_5, x_6, x_7)$;

$\tau_i = \tau_i + 2$ for $1 \leq i \leq 6$;

$\rho_{\rho^{-1}(5)} = \tau_6$;

$\tau_{\tau^{-1}(3)} = \rho_4$;

$\tau_{\tau^{-1}(4)} = \rho_5$;

$(\pi_1, \pi_2, \pi_3) = \rho_{[1\ldots3]}$;

$(\pi_4, \pi_5, \pi_6, \pi_7, \pi_8) = \tau_{[1\ldots5]}$;

if $x_1 = 1$ *then swap* $(\pi_3, \pi_8)$;

*end*

The distance expansion matrix for $g_7$ is given as follows:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 256 | 128 | 32 | 32 | 0 | 0 |
| | 0 | 0 | 448 | 208 | 336 | 272 | 80 |
| | | 0 | 0 | 224 | 912 | 736 | 368 |
| | | | 0 | 0 | 224 | 1184 | 832 |
| | | | | 0 | 0 | 320 | 1024 |
| | | | | | 0 | 0 | 448 |
| | | | | | | 0 | 64 |

**Construction 4** Let $A_8$: $Z_2^4 \rightarrow P_6$ be defined as follows and $B_8$ be the same as $B_7$.

| $x$ | $A_8(x)$ | $x$ | $A_8(x)$ |
|---|---|---|---|
| 0000 | (1, 6, 3, 4, 5, 2) | 1000 | (6, 2, 1, 4, 5, 3) |
| 0001 | (1, 6, 3, 5, 2, 4) | 1001 | (6, 2, 3, 1, 5, 4) |
| 0010 | (1, 6, 4, 2, 5, 3) | 1010 | (6, 4, 5, 1, 2, 3) |
| 0011 | (1, 6, 4, 3, 2, 5) | 1011 | (6, 2, 4, 3, 1, 5) |
| 0100 | (2, 6, 5, 4, 3, 1) | 1100 | (6, 3, 2, 4, 5, 1) |
| 0101 | (2, 6, 3, 5, 4, 1) | 1101 | (6, 3, 2, 5, 1, 4) |
| 0110 | (3, 6, 1, 2, 4, 5) | 1110 | (6, 4, 1, 2, 3, 5) |
| 0111 | (3, 6, 5, 2, 1, 4) | 1111 | (6, 1, 2, 3, 4, 5) |

By the following algorithm, $g_8 \in \mathcal{F}(8, 2, 1)$ is constructed.

*Input:* $(x_1, x_2, \ldots, x_8) \in Z_2^8$
*Output:* $(\pi_1, \ldots, \pi_9) = g_8(x_1, \ldots, x_8)$
*begin*
$\rho = A_8(x_1, \ldots, x_4), \tau = B_8(x_5, \ldots, x_8)$;
$\tau_i = \tau_i + 3$, *for* $1 \leq i \leq 6$;
$\rho_{\rho^{-1}(6)} = \tau_6$;
$\tau_{\tau^{-1}(4)} = \rho_5$;
$\tau_{\tau^{-1}(5)} = \rho_6$;
$(\pi_1, \ldots, \pi_4) = \rho_{[1\ldots4]}$;
$(\pi_5, \ldots, \pi_9) = \tau_{[1\ldots5]}$;
*if* $x_1 = 1$ *then swap* $(\pi_3, \pi_9)$;
*end*

$A_8$ has another property, i.e., if the Hamming distance of any two binary vectors is 3, then their fourth entries of the images must be different. In other words, for $x, y \in Z_2^4$, if $d_H(x, y) = 3$, then $A_8(x)_4 \neq A_8(y)_4$. The distance expansion matrix for $g_8$ is given as follows:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 432 | 384 | 144 | 48 | 16 | 0 | 0 |
| | 0 | 0 | 1008 | 368 | 696 | 912 | 472 | 128 |
| | | 0 | 0 | 464 | 1416 | 2512 | 2088 | 688 |
| | | | 0 | 0 | 320 | 2368 | 3936 | 2336 |
| | | | | 0 | 0 | 352 | 3088 | 3728 |
| | | | | | 0 | 0 | 592 | 2992 |
| | | | | | | 0 | 0 | 1024 |
| | | | | | | | 0 | 128 |

**Construction 5** Let $A_9$ be the same as $A_8$ and $B_9$: $Z_2^5 \rightarrow P_7$ be defined as follows:

| $x$ | $B_9(x)$ | $x$ | $B_9(x)$ |
|---|---|---|---|
| 00000 | (1, 3, 2, 4, 5, 6, 7) | 10000 | (3, 1, 2, 4, 6, 5, 7) |
| 00001 | (1, 3, 2, 4, 6, 7, 5) | 10001 | (7, 1, 2, 5, 3, 4, 6) |
| 00010 | (1, 3, 2, 5, 7, 6, 4) | 10010 | (4, 1, 2, 3, 7, 5, 6) |
| 00011 | (1, 3, 2, 5, 4, 7, 6) | 10011 | (5, 1, 2, 4, 3, 7, 6) |
| 00100 | (1, 3, 4, 2, 6, 5, 7) | 10100 | (3, 1, 4, 2, 5, 6, 7) |
| 00101 | (1, 3, 7, 2, 5, 4, 6) | 10101 | (4, 1, 7, 2, 3, 6, 5) |
| 00110 | (1, 3, 6, 2, 7, 5, 4) | 10110 | (7, 1, 3, 2, 5, 6, 4) |
| 00111 | (1, 4, 3, 2, 5, 7, 6) | 10111 | (7, 1, 3, 2, 4, 5, 6) |
| 01000 | (1, 5, 2, 3, 6, 4, 7) | 11000 | (3, 1, 2, 6, 7, 4, 5) |
| 01001 | (1, 4, 2, 7, 6, 3, 5) | 11001 | (6, 1, 2, 7, 3, 4, 5) |

| | | | |
|---|---|---|---|
| 01010 | (1, 4, 2, 6, 7, 5, 3) | 11010 | (5, 1, 2, 6, 7, 3, 4) |
| 01011 | (1, 5, 2, 6, 3, 7, 4) | 11011 | (5, 1, 2, 7, 4, 3, 6) |
| 01100 | (1, 6, 4, 2, 7, 3, 5) | 11100 | (4, 1, 5, 2, 6, 3, 7) |
| 01101 | (1, 6, 5, 2, 3, 4, 7) | 11101 | (5, 1, 7, 2, 6, 4, 3) |
| 01110 | (1, 5, 6, 2, 4, 3, 7) | 11110 | (6, 1, 5, 2, 7, 3, 4) |
| 01111 | (1, 6, 5, 2, 4, 7, 3) | 11111 | (5, 1, 6, 2, 4, 7, 3) |

By the following algorithm, $g_9 \in \mathcal{F}(9, 2, 1)$ is constructed.

*Input:* $(x_1, x_2, \ldots, x_9) \in Z_2^9$
*Output:* $(\pi_1, \ldots, \pi_{10}) = g_9(x_1, \ldots, x_9)$
*begin*
$\rho = A_9(x_1, \ldots, x_4)$; $\tau = B_9(x_5, \ldots, x_9)$;
$\tau_i = \tau_i + 3$, *for* $1 \leq i \leq 6$;
$\rho_{\rho^{-1}(6)} = \tau_7$;
$\tau_{\tau^{-1}(4)} = \rho_5$;
$\tau_{\tau^{-1}(5)} = \rho_6$;
$(\pi_1, \ldots, \pi_4) = \rho_{[1\ldots4]}$;
$(\pi_5, \ldots, \pi_{10}) = \tau_{[1\ldots6]}$;
*if* $x_1 = 1$ *then swap* $(\pi_3, \pi_9)$;
*if* $x_5 = 1$ *then swap* $(\pi_4, \pi_{10})$;
*end*

The distance expansion matrix for $g_9$ is given as follows:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 672 | 704 | 560 | 240 | 128 | 0 | 0 | 0 |
| | 0 | 0 | 1088 | 1428 | 1480 | 2122 | 1628 | 1094 | 376 |
| | | 0 | 0 | 944 | 1402 | 4370 | 6478 | 5590 | 2720 |
| | | | 0 | 0 | 270 | 2522 | 8390 | 11998 | 9076 |
| | | | | 0 | 0 | 134 | 4284 | 12118 | 15720 |
| | | | | | 0 | 0 | 474 | 5884 | 15146 |
| | | | | | | 0 | 0 | 976 | 8240 |
| | | | | | | | 0 | 0 | 2304 |
| | | | | | | | | 0 | 256 |

## Appendix B Constructions of $h_7, h_8, h_9, h_{10}$

**Construction 6** Let $C_7$ be the same as $C_6$ and $D_7 : Z_2^4 \to P_7$ be defined as follows:

| $x$ | $D_7(x)$ | $x$ | $D_7(x)$ |
|---|---|---|---|
| 0000 | (1, 3, 2, 4, 5, 6, 7) | 1000 | (5, 1, 2, 4, 6, 7, 3) |
| 0001 | (1, 3, 2, 5, 4, 7, 6) | 1001 | (6, 1, 2, 5, 4, 3, 7) |
| 0010 | (1, 3, 4, 2, 6, 5, 7) | 1010 | (7, 1, 4, 2, 5, 6, 3) |
| 0011 | (1, 4, 3, 2, 5, 7, 6) | 1011 | (5, 1, 7, 2, 4, 3, 6) |
| 0100 | (1, 5, 2, 3, 7, 6, 4) | 1100 | (7, 1, 2, 6, 3, 5, 4) |
| 0101 | (1, 5, 2, 7, 3, 4, 6) | 1101 | (4, 1, 2, 6, 7, 3, 5) |
| 0110 | (1, 7, 5, 2, 3, 6, 4) | 1110 | (3, 1, 6, 2, 7, 5, 4) |
| 0111 | (1, 6, 3, 2, 7, 4, 5) | 1111 | (6, 1, 7, 2, 3, 4, 5) |

By the following algorithm, $h_7 \in \mathcal{F}(7, 3, 2)$ is constructed.
*Input:* $(x_1, x_2, \ldots, x_7) \in Z_2^7$
*Output:* $(\pi_1, \pi_2, \ldots, \pi_9) = h_7(x_1, \ldots, x_7)$

*begin*
$\rho = C_7(x_1, x_2, x_3);\ \tau = D_7(x_4, \ldots, x_7);$
$\rho_i = \rho_i - 2,$ *for* $1 \le i \le 6;$
$\tau_i = \tau_i + 2,$ *for* $1 \le i \le 7;$
$\rho_{\rho^{-1}(-1)} = \tau_6;$
$\rho_{\rho^{-1}(0)} = \tau_7;$
$\tau_{\tau^{-1}(3)} = \rho_5;$
$\tau_{\tau^{-1}(4)} = \rho_6;$
$(\pi_1, \ldots, \pi_4) = \rho_{[1\ldots4]};$
$(\pi_5, \ldots, \pi_9) = \tau_{[1\ldots5]};$
*end*

In addition to the position property, $D_7$ has another property such that if the Hamming distance of any two binary vectors is 3, then their fifth entries of the images must be different, i.e. for $x, y \in Z_2^4$, if $d_H(x, y) = 3$, then $D_7(x)_5 \neq D_7(y)_5$. The distance expansion matrix for $h_7$ is given as follows:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 312 | 128 | 8 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 408 | 176 | 256 | 408 | 96 |
| | | 0 | 0 | 0 | 232 | 368 | 952 | 688 |
| | | | 0 | 0 | 0 | 120 | 792 | 1328 |
| | | | | 0 | 0 | 0 | 208 | 1136 |
| | | | | | 0 | 0 | 0 | 448 |
| | | | | | | 0 | 0 | 64 |

**Construction 7** Let both $C_8$ and $D_8$ be the same as $D_7$. By the following algorithm, $h_8 \in \mathcal{F}(8, 3, 2)$ is constructed.

*Input:* $(x_1, x_2, \ldots, x_8) \in Z_2^8$
*Output:* $(\pi_1, \pi_2, \ldots, \pi_{10}) = h_8(x_1, \ldots, x_8)$
*begin*
$\rho = C_8(x_1, \ldots, x_4);\ \tau = D_8(x_5, \ldots, x_8);$
$\rho_i = \rho_i - 2,$ *for* $1 \le i \le 7;$
$\tau_i = \tau_i + 3,$ *for* $1 \le i \le 7;$
$\rho_{\rho^{-1}(-1)} = \tau_6;$
$\rho_{\rho^{-1}(0)} = \tau_7;$
$\tau_{\tau^{-1}(4)} = \rho_6;$
$\tau_{\tau^{-1}(5)} = \rho_7;$
$(\pi_1, \ldots, \pi_5) = \rho_{[1\ldots5]};$
$(\pi_6, \ldots, \pi_{10}) = \tau_{[1\ldots5]};$
*end*

The distance expansion matrix for $h_8$ is as follows:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 480 | 512 | 32 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 1120 | 392 | 300 | 818 | 760 | 194 |
| | | 0 | 0 | 0 | 688 | 672 | 2048 | 2752 | 1008 |
| | | | 0 | 0 | 0 | 416 | 1720 | 3856 | 2968 |
| | | | | 0 | 0 | 0 | 372 | 3192 | 3604 |
| | | | | | 0 | 0 | 0 | 832 | 2752 |
| | | | | | | 0 | 0 | 0 | 1024 |
| | | | | | | | 0 | 0 | 128 |

**Construction 8** Let $C_9$ be the same as $C_8$ and $D_9$: $Z_2^5 \rightarrow P_8$ be defined as follows:

| $x$ | $D_9(x)$ | $x$ | $D_9(x)$ |
|---|---|---|---|
| 00000 | (1, 3, 2, 4, 5, 6, 7, 8) | 10000 | (3, 1, 2, 5, 8, 6, 7, 4) |
| 00001 | (1, 3, 2, 4, 6, 5, 8, 7) | 10001 | (3, 1, 2, 7, 6, 5, 8, 4) |
| 00010 | (1, 3, 2, 5, 4, 7, 6, 8) | 10010 | (3, 1, 2, 4, 7, 8, 5, 6) |
| 00011 | (1, 3, 2, 5, 7, 4, 8, 6) | 10011 | (4, 1, 2, 3, 6, 7, 5, 8) |
| 00100 | (1, 3, 4, 2, 5, 8, 6, 7) | 10100 | (3, 1, 7, 2, 5, 8, 6, 4) |
| 00101 | (1, 3, 4, 2, 8, 5, 7, 6) | 10101 | (5, 1, 7, 2, 8, 4, 3, 6) |
| 00110 | (1, 3, 6, 2, 7, 8, 5, 4) | 10110 | (6, 1, 7, 2, 4, 3, 5, 8) |
| 00111 | (1, 3, 6, 2, 8, 7, 4, 5) | 10111 | (4, 1, 3, 2, 8, 7, 5, 6) |
| 01000 | (1, 4, 2, 6, 5, 8, 7, 3) | 11000 | (8, 1, 2, 6, 5, 3, 7, 4) |
| 01001 | (1, 4, 2, 6, 8, 5, 3, 7) | 11001 | (8, 1, 2, 6, 3, 5, 4, 7) |
| 01010 | (1, 4, 2, 8, 7, 6, 5, 3) | 11010 | (7, 1, 2, 8, 4, 3, 6, 5) |
| 01011 | (1, 4, 2, 8, 6, 7, 3, 5) | 11011 | (4, 1, 2, 6, 3, 7, 8, 5) |
| 01100 | (1, 5, 8, 2, 4, 6, 7, 3) | 11100 | (7, 1, 8, 2, 5, 6, 3, 4) |
| 01101 | (1, 5, 8, 2, 6, 4, 3, 7) | 11101 | (7, 1, 8, 2, 6, 5, 4, 3) |
| 01110 | (1, 4, 5, 2, 7, 3, 6, 8) | 11110 | (4, 1, 8, 2, 7, 3, 6, 5) |
| 01111 | (1, 4, 5, 2, 3, 7, 8, 6) | 11111 | (7, 1, 6, 2, 3, 4, 8, 5) |

By the following algorithm, $h_9 \in \mathcal{F}(9, 3, 2)$ is constructed.

*Input:* $(x_1, x_2, \ldots, x_9) \in Z_2^9$
*Output:* $(\pi_1, \pi_2, \ldots, \pi_{11}) = h_9(x_1, \ldots, x_9)$
*begin*
0 $\rho = C_9(x_1, \ldots, x_4)$; $\tau = D_9(x_5, \ldots, x_9)$;
1 $\rho_i = \rho_i - 2$, *for* $1 \leq i \leq 7$;
2 $\tau_i = \tau_i + 3$, *for* $1 \leq i \leq 8$;
3 $\rho_{\rho^{-1}(-1)} = \tau_7$;
4 $\rho_{\rho^{-1}(0)} = \tau_8$;
5 $\tau_{\tau^{-1}(3)} = \rho_6$;
6 $\tau_{\tau^{-1}(4)} = \rho_7$;
7 $(\pi_1, \ldots, \pi_5) = \rho_{[1\ldots5]}$;
8 $(\pi_6, \ldots, \pi_{11}) = \tau_{[1\ldots6]}$;
9 *if* $x_5 = 1$ *then swap* $(\pi_5, \pi_{10})$;
*end*

$D_9$ has another property such that if the Hamming distance of any two binary vectors is 4, then their sixth entries of the images must be different, i.e. for $x, y \in Z_2^5$, if $d_H(x, y) = 4$, then $D_9(x)_6 \neq D_9(y)_6$. The distance expansion matrix for $h_9$ is as follows:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 912 | 912 | 336 | 112 | 32 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 1952 | 1090 | 918 | 1934 | 2034 | 1024 | 264 |
| | | 0 | 0 | 0 | 1328 | 1334 | 3782 | 6916 | 5870 | 2274 |
| | | | 0 | 0 | 0 | 544 | 1910 | 8726 | 13074 | 8002 |
| | | | | 0 | 0 | 0 | 132 | 4060 | 13852 | 14212 |
| | | | | | 0 | 0 | 0 | 454 | 6756 | 14294 |
| | | | | | | 0 | 0 | 0 | 1168 | 8048 |
| | | | | | | | 0 | 0 | 0 | 2304 |
| | | | | | | | | 0 | 0 | 256 |

**Construction 9** Let both $C_{10}$ and $D_{10}$ be the same as $D_9$. By the following algorithm, $h_{10} \in \mathcal{F}(10, 3, 2)$ is constructed.

*Input:* $(x_1, x_2, \ldots, x_{10}) \in Z_2^{10}$
*Output:* $(\pi_1, \pi_2, \ldots, \pi_{12}) = h_{10}(x_1, \ldots, x_{10})$
*begin*
$\rho = C_{10}(x_1, \ldots, x_5)$; $\tau = D_{10}(x_6, \ldots, x_{10})$;
$\rho_i = \rho_i - 2$, *for* $1 \leq i \leq 8$;
$\tau_i = \tau_i + 4$, *for* $1 \leq i \leq 8$;
$\rho_{\rho^{-1}(-1)} = \tau_7$;
$\rho_{\rho^{-1}(0)} = \tau_8$;
$\tau_{\tau^{-1}(4)} = \rho_7$;
$\tau_{\tau^{-1}(6)} = \rho_8$;
$(\pi_1, \ldots, \pi_6) = \rho_{[1\ldots6]}$;
$(\pi_7, \ldots, \pi_{12}) = \tau_{[1\ldots6]}$;
*if* $x_1 = 1$ *then swap* $(\pi_5, \pi_{12})$;
*if* $x_6 = 1$ *then swap* $(\pi_6, \pi_{11})$;
*end*

The distance expansion matrix for $h_{10}$ is as follows:

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1728 | 1600 | 1216 | 448 | 128 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 3328 | 2818 | 2348 | 4540 | 4300 | 3528 | 1652 | 526 |
| | | 0 | 0 | 0 | 2624 | 2868 | 7084 | 13904 | 17352 | 12172 | 5436 |
| | | | 0 | 0 | 0 | 1024 | 2772 | 14192 | 32644 | 35416 | 21472 |
| | | | | 0 | 0 | 0 | 8 | 4352 | 26788 | 51992 | 45884 |
| | | | | | 0 | 0 | 0 | 136 | 8596 | 40320 | 58468 |
| | | | | | | 0 | 0 | 0 | 768 | 15168 | 45504 |
| | | | | | | | 0 | 0 | 0 | 2176 | 20864 |
| | | | | | | | | 0 | 0 | 0 | 5120 |
| | | | | | | | | | 0 | 0 | 512 |

## References


1. Chang JC (2005) Distance-increasing mappings from binary vectors to permutations. IEEE Trans Inf Theory 51:359–363
2. Chang JC, Chen RJ, Kløve T, Tsai SC (2003) Distance-preserving mappings from binary vectors to permutations. IEEE Trans Inf Theory 49:1054–1059
3. Colbourn CJ, Kløve T, Ling ACH (2004) Permutation arrays for powerline communication and mutually orthogonal latin squares. IEEE Trans Inf Theory 50:1289–1291
4. Ding C, Fu FW, Kløve T, Wei VK (2002) Construction of permutation arrays. IEEE Trans Inf Theory 48:977–980
5. Deza M, Vanstone SA (1978) Bounds on permutation arrays. J Stat Planning Inference 2:197–209
6. Fu FW, Kløve T (2004) Two constructions of permutation arrays. IEEE Trans Inf Theory 50:881–883
7. Ferreira HC, Vinck AJH (2000) Inference cancellation with permutation trellis arrays. Proc. IEEE Vehicular Technology Conf., 2401–2407
8. Lee K (2004) New distance-preserving maps of odd length. IEEE Trans Inf Theory 50:2539–2543
9. Tarnanen H (1999) Upper bounds on permutation codes via linear programming. Europ J Combin 20:101–114
10. Vinck AJH (2000) Coded modulation for powerline communications. AEÜ Int J Elecron Commun 54:45–49
11. Vinck AJH, Häring J (2000) Coding and modulation for power-line communications. In: Proc. Int. Symp. Power Line Communication, Limerick, Ireland, pp 5–7
12. Vinck AJH, Häring J, Wadayama T (2000) Coded M-FSK for power-line communications. In: Proc. IEEE Int. Symp. Information Theory, Sorrento, Italy, p 137
13. Wadayama T, Vinck AJH (2001) A multilevel construction of permutation codes. IEICE Trans Fundamentals Electron., Commun Comp Sci 84:2518–2522