Simple Distance-Preserving Mappings from Ternary Vectors to Permutations

Te-Tsung Lin, Shi-Chun Tsai, *Member, IEEE* Hsin-Lung Wu

***Abstract*—We give a simple construction of distance-preserving mappings from ternary vectors to permutations (3-DPM). Our result gives a lower bound for permutation arrays, i.e., $P(n,d) \geq A_3(n,d)$, which significantly improves previous lower bounds for $d \leq \frac{3n}{5}$.**

***Index Terms*—Code constructions, distance-preserving mappings, permutation arrays, ternary codes.**

I. Introduction and notations

In this paper, we construct distance-preserving mappings from ternary vectors of dimension n to permutations of $\{1, 2, \cdots, n\}$ for $n \geq 16$. A permutation array is a subset of permutations that satisfies some distance constraints. A systematic study of DPMs (distance-preserving mappings) was initiated in [4]. Later, Chang [3] gave a construction of DIMs (distance-increasing mappings) and proved a lower bound on the size of permutation arrays, i.e. $P(n,d) \geq A(n,d-k)$ for any k if n is sufficiently large, where $P(n,d)$ denotes the maximal size among all permutation arrays of length n with minimum distance d, and $A(n,d)$ denotes the maximal size among all binary codes of length n and minimum distance d.

Our Results: In [4], the authors asked a question of finding a distance-preserving mapping from q-ary vectors to permutations (in short, q-DPM). Here we give a simple construction of 3-DPM[1]. This answers the question for $q = 3$. Then we prove that $P(n,d) \geq A_3(n,d)$ where $A_3(n,d)$ denotes the maximal size among all ternary codes of length n and minimum distance d. Suppose that $d < 3n/5$. Then $A_3(n,d)$ is much larger than $A(n,d-k)$ for any k if n is large enough. Thus, our result significantly improves the previous bounds obtained from DIMs over binary vectors which increases distance by at least k. Clearly, the best lower bounds by previous approaches can only achieve at most 2^n. Here, for some fixed constant c, our lower bound is 3^{cn} which is significantly larger than 2^n when n is sufficiently large.

Construction Idea: Our 3-DPM construction is inspired by [9]. It is based on a crucial "local" property which we discuss as follows. Intuitively, an algorithm has the local property if each element of the permutation is not far away from its initial position after running the algorithm. From a 2-DPM with local property, we can obtain a 3-DPM as follows. First, given a ternary input vector, we view the ternary digit 2 as 0 and run a 2-DPM algorithm such that every element in the permutation is not far from its initial position, i.e. with a small position difference. Now, how can we make up the distance loss caused by seeing the ternary digit 2 as 0? Our approach is swapping those positions whose corresponding input digits are 2 and far enough, i.e. with the position difference larger than the difference resulting from the initial 2-DPM. This will give us a 3-DPM if we have a 2-DPM with local property. We constructed a two-pass 3-DPM by using a 2-DPM, which is very similar to the one constructed in [8], [9]. However, in these papers, the local property is not fully exploited.

The swaps in our two-pass algorithm are similar to the multi-level construction of DPMs from binary vectors in [6] where swaps for each level are independent from swaps for another level. Similarly, swaps for PASS 1 and PASS 2 in our algorithm are also independent.

Notations: Let $[n] = \{1, \cdots, n\}$, S_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 any $\pi \in S_n$ and $i \in [n]$, $\pi^{-1}(i)$ denotes the position of i in π, i.e. if $\pi(j) = i$ then $\pi^{-1}(i) = j$. Given an $x \in Z_q^n$, we use $x_{[i..j]}$ to denote the subvector $(x_i, \cdots, x_j)$ for any $i < j$. The Hamming distance $d_H(a,b)$ between two n-tuples $a = (a_1, a_2, \cdots, a_n)$ and $b = (b_1, b_2, \cdots, b_n)$ is the number of positions where they differ, i.e. $d_H(a,b) = |\{j : a_j \neq b_j\}|$. A mapping $f : Z_q^n \rightarrow S_n$ is a q-ary distance-preserving mapping (q-DPM) if, for any $x, y \in Z_q^n$, $d_H(f(x), f(y)) \geq d_H(x,y)$. Let $\delta : Z_q \times Z_q \rightarrow \{0,1\}$ be the function defined by $\delta(s,t) = 1$ if $s \neq t$ and 0 otherwise, i.e. the Hamming distance for single elements. In Subsection II-A, values of permutations and subscripts are represented by elements in $Z_{8m} = [8m]$. For example, if $a, b \in Z_{8m}$ then the output of $a + b$ is $a + b$ mod $8m$ if $a + b$ mod $8m \neq 0$, $8m$ otherwise.

This paper is organized as follows. In Section II, we show how to construct a family of distance-preserving mappings from ternary vectors to permutations. In Section III, we show a new lower bound for permutation arrays with our construction. Section IV concludes with an open problem.

II. Construction of 3-DPM

In this section, we give the construction of 3-DPM. First of all, we show the algorithm for input length $8m$ for any integer $m \geq 2$. We call the algorithm A_{8m}. Then we extend A_{8m} to an algorithm that works for all input lengths at least 16.

Te-Tsung Lin and Shi-Chun Tsai are with the Department of Computer Science, National Chiao-Tung University, Hsinchu 30050, Taiwan, Email: (atman.cs94g@nctu.edu.tw, sctsai@csie.nctu.edu.tw). The work was supported in part by the National Science Council of Taiwan under contract NSC-95-2213-E-009-034.

Hsin-Lung Wu is currently with the Department of Computer Science and Information Engineering, National Taipei University, Taipei County 237, Taiwan, Email: hsinlung@mail.ntpu.edu.tw. Part of the work was done when he was at National Chiao-Tung University.

[1] In [5], the authors independently show a recursive construction of 3-DPM. However, our construction is non-recursive and simpler than theirs.

A. 3-DPM of length $8m$ for $m \geq 2$

The 3-DPM of length $8m$ (A_{8m}) is shown in the following.

Algorithm A_{8m}:
Input: $(x_1, \cdots, x_{8m}) \in Z_3^{8m}$
Output: $(\pi_1, \cdots, \pi_{8m}) \in S_{8m}$

PASS 1 :
$(\pi_1^1, \pi_2^1, \cdots, \pi_{8m}^1) \leftarrow (1, 2, \cdots, 8m)$;
for $i = 0$ **to** $4m-1$ **do**;
  if $x_{2i+1} = 1$ **then swap** $(\pi_{2i+1}^1, \pi_{2i+2}^1)$;
for $i = 0$ **to** $4m-1$ **do**;
  if $x_{2i+2} = 1$ **then swap** $(\pi_{2i+2}^1, \pi_{2i+3}^1)$;

PASS 2 :
$(\pi_1, \pi_2, \cdots, \pi_{8m}) \leftarrow (\pi_1^1, \pi_2^1, \cdots, \pi_{8m}^1)$;
for $i = 0$ **to** $m-1$ **do**;
  if $x_{8i+1} = 2$ **then swap** (π_{8i+1}, π_{8i+5});
  if $x_{8i+2} = 2$ **then swap** (π_{8i+2}, π_{8i+6});
  if $x_{8i+3} = 2$ **then swap** (π_{8i+3}, π_{8i+7});
  if $x_{8i+4} = 2$ **then swap** (π_{8i+4}, π_{8i+8});
for $i = 0$ **to** $m-1$ **do**;
  if $x_{8i+5} = 2$ **then swap** (π_{8i+5}, π_{8i+9});
  if $x_{8i+6} = 2$ **then swap** $(\pi_{8i+6}, \pi_{8i+10})$;
  if $x_{8i+7} = 2$ **then swap** $(\pi_{8i+7}, \pi_{8i+11})$;
  if $x_{8i+8} = 2$ **then swap** $(\pi_{8i+8}, \pi_{8i+12})$;
Output $(\pi_1, \cdots, \pi_{8m})$.

Algorithm A_{8m} consists of two passes: PASS 1 and PASS 2. The transition patterns of both passes are illustrated in Fig. 1 and 2 respectively.

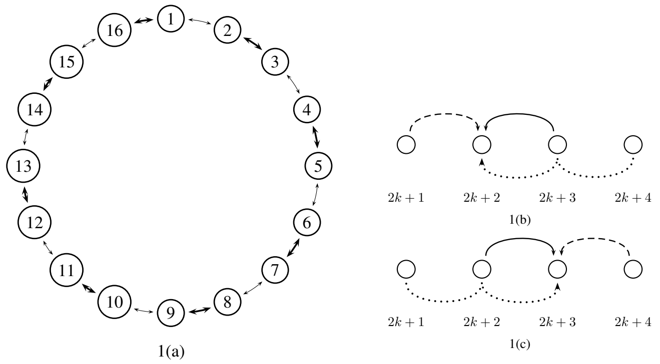

Fig. 1. Transition patterns of PASS 1.

In Fig. 1(a) and 2(a), the thin lines represent the transpositions in the first for-loop of both passes and the thick lines represent those transpositions in the second for-loop. Note that PASS 1 has the "local" property which is implicitly used in [8], [9]. Since all transpositions in a single for-loop are independent and can be done simultaneously, the local property can be observed in Fig. 1. Before proving the distance-preserving property of A_{8m}, we show some properties of the algorithm A_{8m}.

Given $x \in Z_3^{8m}$, let $\pi = A_{8m}(x)$ and π^1 be the intermediate result after PASS 1. First of all, for any fixed position s, we look into what possible values π_s and π_s^1 can be after running the corresponding passes of A_{8m}.

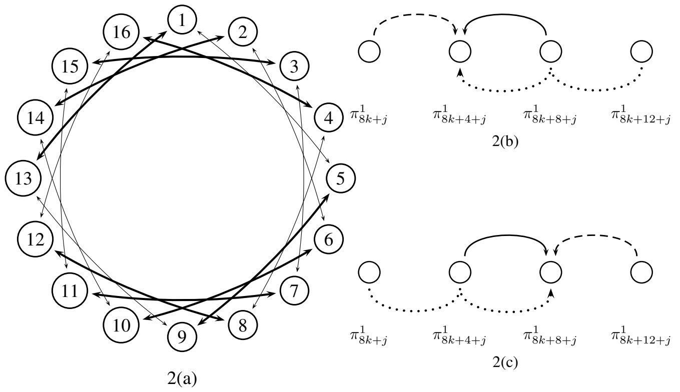

Fig. 2. Transition patterns of PASS 2. $j \in \{0, 1, 2, 3\}$

Lemma 1: If s is even, the possible values of π_s^1 are $\{s-1, s, s+1, s+2\}$. If s is odd, the possible values of π_s^1 are $\{s-2, s-1, s, s+1\}$. If $s = 8k+4+j$ for $j \in \{0,1,2,3\}$, the possible values of π_s are in $\{\pi_{s-4}^1, \pi_s^1, \pi_{s+4}^1, \pi_{s+8}^1\}$. If $s = 8k+8+j$ for $j \in \{0,1,2,3\}$, the possible values of π_s are in $\{\pi_{s-8}^1, \pi_{s-4}^1, \pi_s^1, \pi_{s+4}^1\}$.

Proof: First consider when s is even. Let $s = 2k+2$. Observing Fig. 1(b), the possible values of π_{2k+2}^1 are $\{2k+1, 2k+2, 2k+3, 2k+4\}$. For example, if $x_{2k+1} \neq 1$, $x_{2k+2} = 1$ and $x_{2k+3} = 1$ (transition indicated in dotted line), then $\pi_{2k+2}^1 = 2k+4$. If only $x_{2k+2} = 1$ (normal line), then $\pi_{2k+2}^1 = 2k+3$. If only $x_{2k+1} = 1$ (dashed line), then $\pi_{2k+2}^1 = 2k+1$. If all inputs are zero, then $\pi_{2k+2}^1 = 2k+2$. Similarly for odd s, the transition pattern is shown in Fig. 1(c). All cases are summarized in Table I.

TABLE I
POSSIBLE VALUES OF π_s^1 AFTER PASS 1 FOR $k \in \{0, 1, \cdots, 4m-1\}$.

	x_{2k+1}	x_{2k+2}	x_{2k+3}	π_{2k+2}^1	π_{2k+3}^1
1	-	-	-	$2k+2$	$2k+3$
2	-	-	1	$2k+2$	$2k+4$
3	-	1	-	$2k+3$	$2k+2$
4	-	1	1	$2k+4$	$2k+2$
5	1	-	-	$2k+1$	$2k+3$
6	1	-	1	$2k+1$	$2k+4$
7	1	1	-	$2k+3$	$2k+1$
8	1	1	1	$2k+4$	$2k+1$

In the table, each row stands for the input and the corresponding result of swap operations. For example, in row 7, if $x_{2k+1} = x_{2k+2} = 1$ and $x_{2k+3} \neq 1$, then $\pi_{2k+2}^1 = 2k+3$ and $\pi_{2k+3}^1 = 2k+1$. Thus by a similar observation from Fig. 2, we summarize the possible values of π_s in Table II, which is very similar to Table I if we replace 1 by 2. The lemma is clear by Table I and Table II. ∎

Given $x, y \in Z_3^{8m}$, let $A_{8m}(x) = \pi$, $A_{8m}(y) = \tau$, and π^1,τ^1 be the intermediate results after PASS 1 respectively.

Lemma 2: If s and t have the same parity (i.e., both even or odd) and $|s-t| \geq 4$, then $\pi_s^1 \neq \tau_t^1$.

Proof: Assume that s and t are even. By Lemma 1, the possible values of π_s^1 are $\{s-1, s, s+1, s+2\}$ and the possible values of τ_t^1 are $\{t-1, t, t+1, t+2\}$. Clearly $|s-t| \geq 4$ implies that $\pi_s^1 \neq \tau_t^1$. Similarly the lemma holds for the case when s and t are odd. ∎

TABLE II
POSSIBLE VALUES OF π_s AFTER PASS 2 FOR $k \in \{0, 1, \cdots, m-1\}$ AND $j \in \{1, 2, 3, 4\}$.

	x_{8k+j}	x_{8k+4+j}	x_{8k+8+j}	π_{8k+4+j}	π_{8k+8+j}
1	-	-	-	π^1_{8k+4+j}	π^1_{8k+8+j}
2	-	-	2	π^1_{8k+4+j}	$\pi^1_{8k+12+j}$
3	-	2	-	π^1_{8k+8+j}	π^1_{8k+4+j}
4	-	2	2	$\pi^1_{8k+12+j}$	π^1_{8k+4+j}
5	2	-	-	π^1_{8k+j}	π^1_{8k+8+j}
6	2	-	2	π^1_{8k+j}	$\pi^1_{8k+12+j}$
7	2	2	-	π^1_{8k+8+j}	π^1_{8k+j}
8	2	2	2	$\pi^1_{8k+12+j}$	π^1_{8k+j}

The following lemma shows that if the values of the s-th position of π and τ are different after running PASS 1, the difference will be kept (or the difference may be propagated to different position and preserved) after running the whole algorithm.

Lemma 3: If $\pi^1_s \neq \tau^1_s$ and $\pi_t = \pi^1_s$ for any s, t, then $\pi_t \neq \tau_t$.

Proof: Note that $\pi_t = \pi^1_s$ implies that $4|(t-s)$ since t must be one of the elements in $\{s-8, s-4, s, s+4, s+8\}$ by Lemma 1. Similarly assume that $\tau_t = \tau^1_{s'}$, then we have $4|(t-s')$. Thus, $4|(s-s')$. If $|s-s'| \geq 4$, then we obtain $\pi^1_s \neq \tau^1_{s'}$ by Lemma 2. Therefore, in this case, $\pi_t \neq \tau_t$. On the other hand, if $|s-s'| < 4$, it implies $s = s'$. By assumption, we have $\pi^1_s \neq \tau^1_s$ and this also implies $\pi_t \neq \tau_t$. ■

Next we need the following definitions to show that A_{8m} does preserve the distance.

Definition 1: For any $s \neq t$, we say that position s can be covered with position t if $\delta(x_s, y_s) > \delta(\pi_s, \tau_s)$ and $\delta(x_t, y_t) < \delta(\pi_t, \tau_t)$ (that is, $x_s \neq y_s$, $\pi_s = \tau_s$, $x_t = y_t$, and $\pi_t \neq \tau_t$). Furthermore, we say that position s is self-covered if $\delta(x_s, y_s) \leq \delta(\pi_s, \tau_s)$.

For each s with $\delta(x_s, y_s) > \delta(\pi_s, \tau_s)$, one needs some other position to make up the decrease of distance at position s in order to satisfy the distance-preserving property.

Definition 2: Let NSC be the set of positions not self-covered, that is, NSC= $\{s \in [8m] : \delta(x_s, y_s) > \delta(\pi_s, \tau_s)\}$. A covering pattern is a function $g : [8m] \to [8m]$ such that for any $s \in$ NSC, $g(s)$ covers s and for any $s \in [8m]\backslash$NSC, $g(s) = s$.

We will show that for any x and y there is a one to one covering pattern. The following is our main lemma which is crucial to show the distance-preserving property of A_{8m}.

Lemma 4: There exists a covering pattern g such that for any position $s \in$NSC, $g(s) \in \{s-1, s-4, s-5, s-8, s-9\}$. Furthermore, $|g^{-1}(t) \cap \{t+1, t+4, t+5, t+8, t+9\}| \leq 1$ for any position t.

Proof: We define such a covering pattern g by analyzing every possible position $s \in [8m]$ and setting $g(s)$ case by case. For convenience, we can let $g(s) = s$ for all s by default. If s is not self-covered, then we will set $g(s)$ to be another value. In other words, we reset $g(s)$ whenever necessary.

a) Case 1: [s with $x_s = y_s$] It implies that $\delta(x_s, y_s) = 0$, and it is always true that $\delta(\pi_s, \tau_s) \geq \delta(x_s, y_s)$. So s is self-covered and we set $g(s) = s$ by default.

b) Case 2: [s with $x_s \neq y_s$ and one of x_s and y_s is 2] W.L.O.G., we may assume that $x_s = 2$ and $y_s \neq 2$.

- Case 2-1: [$s = 8k+4+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{1, 2, 3, 4\}$] Observe that in Table II, under the case condition, the possible values of π_s are in $\{\pi^1_{8k+8+j}, \pi^1_{8k+12+j}\}$ and the possible values of τ_s are in $\{\tau^1_{8k+j}, \tau^1_{8k+4+j}\}$. Note that $\{\pi^1_{8k+8+j}, \pi^1_{8k+12+j}\} \cap \{\tau^1_{8k+j}, \tau^1_{8k+4+j}\} = \emptyset$ by Lemma 2. Thus, $\pi_s \neq \tau_s$. So s is self-covered and we set $g(s) = s$.
- Case2-2: [$s = 8k+8+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{1, 2, 3, 4\}$] In this case, the possible values of π_s are in $\{\pi^1_{8k+j}, \pi^1_{8k+4+j}, \pi^1_{8k+12+j}\}$ and the possible values of τ_s are in $\{\tau^1_{8k+j}, \tau^1_{8k+4+j}, \tau^1_{8k+8+j}\}$. Assume that $\pi_s = \pi^1_{s_1}$ and $\tau_s = \tau^1_{s_2}$. If $s_1 \neq s_2$, then $|s_1 - s_2| \geq 4$ and $\pi_s \neq \tau_s$ by Lemma 2. I.e., s is self-covered. In this case, set $g(s) = s$. On the other hand, if $s_1 = s_2$, then it must be the cases in rows 3 and 4 (i.e. $s_1 = s_2 = 8k+4+j$) or in rows 7 and 8 (i.e. $s_1 = s_2 = 8k+j$) of Table II. In both cases, observe that x_{8k+4+j} and y_{8k+4+j} must be 2 and $\pi_{8k+4+j} = \pi^1_{8k+12+j}$ and $\tau_{8k+4+j} = \tau^1_{8k+8+j}$. By Lemma 2, $\pi_{8k+4+j} \neq \tau_{8k+4+j}$. Note that it's still possible $\pi_s \neq \tau_s$; i.e. s is self-covered, and we can simply set $g(s) = s$. So if $\pi_s \neq \tau_s$, then set $g(s) = s$, else $s = 8k+8+j$ can be covered with position $s-4 = 8k+4+j$ and we set $g(s) = s-4$.

TABLE III
POSSIBLE FINAL POSITIONS OF π^1_{8k+j} AND π^1_{8k+4+j}

	x_{8k-4+j}	x_{8k+j}	x_{8k+4+j}	π_{8k-4+j}	π_{8k+j}	π_{8k+4+j}	π_{8k+8+j}
1	-	-	-		π^1_{8k+j}	π^1_{8k+4+j}	
2	-	-	2		π^1_{8k+j}		π^1_{8k+4+j}
3	-	2	-		π^1_{8k+4+j}	π^1_{8k+j}	
4	-	2	2		π^1_{8k+4+j}		π^1_{8k+j}
5	2	-	-	π^1_{8k+j}		π^1_{8k+4+j}	
6	2	-	2	π^1_{8k+j}			π^1_{8k+4+j}
7	2	2	-	π^1_{8k+4+j}		π^1_{8k+j}	
8	2	2	2	π^1_{8k+4+j}			π^1_{8k+j}

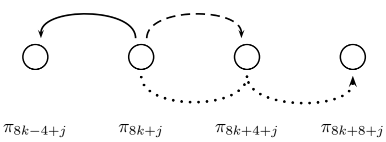

Fig. 3. The possible final positions of π^1_{8k+j}, $j \in \{0, 1, 2, 3\}$.

c) Case 3: [s with $x_s \neq y_s$ and $x_s, y_s \in \{0, 1\}$] In this case, W.L.O.G. we may assume $x_s = 1$ and $y_s = 0$. For convenience, we use Table III to show that the possible positions of π^1_{8k+j} and π^1_{8k+4+j}. For example, row 7 means that when $x_{8k-4+j} = 2$, $x_{8k+j} = 2$ and $x_{8k+4+j} \neq 2$, then after running PASS 2, π^1_{8k+j} will appear in position $8k+4+j$ (Fig. 3: dashed line) and π^1_{8k+4+j} in position $8k-4+j$.

- Case 3-1: [$\pi^1_s \neq \tau^1_s$ and $s = 8k+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{1, 2, 3, 4\}$] Note that $x_s \neq 2$. By Table III, π^1_{8k+j} can be in either position $8k+j$ or $8k-4+j$. If $\pi_{8k+j} = \pi^1_{8k+j}$, then $\pi_{8k+j} \neq \tau_{8k+j}$ by

Lemma 3. Thus, s is self-covered and we set $g(s) = s$. Similarly it applies to the case when $\tau_{8k+j} = \tau^1_{8k+j}$. The rest of this case is that both π^1_{8k+j} and τ^1_{8k+j} are in position $8k-4+j$. When this happens, it implies that $x_{s-4} = y_{s-4} = 2$ by observing Table III and we have $\pi_{8k-4+j} = \pi^1_{8k+j}$ and $\tau_{8k-4+j} = \tau^1_{8k+j}$. By assumption that $\pi^1_{8k+j} \neq \tau^1_{8k+j}$, we conclude that s can be covered with $s-4$ and set $g(s) = s-4$, if $\pi_s = \tau_s$.

- Case 3-2: [$\pi^1_s \neq \tau^1_s$ and $s = 8k+4+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{1, 2, 3, 4\}$] Again by observing Table III if $x_s \neq 2$ and $y_s \neq 2$, then π^1_{8k+4+j} and τ^1_{8k+4+j} have three possible final positions i.e., $8k+4+j$, $8k+j$, and $8k-4+j$. We divide the analysis into three subcases.
  - Subcase 3-2-I: [π^1_{8k+4+j} or τ^1_{8k+4+j} are in position $8k+4+j$] W.L.O.G. we assume that π^1_{8k+4+j} appears in position $8k+4+j$, i.e. $\pi_{8k+4+j} = \pi^1_{8k+4+j}$. By the assumption that $\pi^1_{8k+4+j} \neq \tau^1_{8k+4+j}$, we obtain $\pi_{8k+4+j} \neq \tau_{8k+4+j}$ by Lemma 3. Thus $s = 8k+4+j$ is self-covered and set $g(s) = s$ by default.
  - Subcase 3-2-II: [π^1_{8k+4+j} or τ^1_{8k+4+j} are in position $8k+j$] W.L.O.G. we assume that π^1_{8k+4+j} appears in position $8k+j$, i.e. $\pi_{8k+j} = \pi^1_{8k+4+j}$. We can assume that $\tau_{8k+4+j} \neq \tau^1_{8k+4+j}$, otherwise it has been done in Subcase 3-2-I. By Lemma 3, it's clear that $\pi_{8k+j} \neq \tau_{8k+j}$. In this subcase since $\pi_{8k+4+j} \neq \pi^1_{8k+4+j}$, $\tau_{8k+4+j} \neq \tau^1_{8k+4+j}$ and both x_{8k+4+j} and y_{8k+4+j} are not equal to 2, it must be the cases in row 3 or row 7 of Table III. In both cases we have $x_{8k+j} = y_{8k+j} = 2$. Thus s can be covered with $s-4$ and we set $g(s) = s-4$, if $\pi_s = \tau_s$.
  - Subcase 3-2-III: [Both π^1_{8k+4+j} and τ^1_{8k+4+j} are in position $8k-4+j$] I.e. $\pi_{8k-4+j} = \pi^1_{8k+4+j}$ and $\tau_{8k-4+j} = \tau^1_{8k+4+j}$. Clearly $\pi_{8k-4+j} \neq \tau_{8k-4+j}$ by the assumption of Case 3-2 that $\pi^1_{8k+4+j} \neq \tau^1_{8k+4+j}$. Again, by observing Table III, it must be the case that $x_{8k-4+j} = y_{8k-4+j} = 2$ and $x_{8k+j} = y_{8k+j} = 2$. Thus s can be covered with $s-8$ and we set $g(s) = s-8$, if $\pi_s = \tau_s$.

Next, we deal with the case that $\pi^1_s = \tau^1_s$ and $x_s, y_s \in \{0, 1\}$ with $x_s \neq y_s$. By observing Table I, in this case, s must be odd, and in rows 3 and 4 (i.e. $\pi^1_{2k+3} = \tau^1_{2k+3} = 2k+2$) or in rows 7 and 8 (i.e. $\pi^1_{2k+3} = \tau^1_{2k+3} = 2k+1$) in Table I. Observe that $x_{s-1} = y_{s-1} = 1$ and $\pi^1_{s-1} \neq \tau^1_{s-1}$ in these cases. We divide the analysis into two cases.

- Case 3-3: [$\pi^1_s = \tau^1_s$ and $s = 8k+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{3, 5\}$] Note that $x_s \neq y_s$. From the above discussion, we know that $x_{s-1} = y_{s-1} = 1$ and $\pi^1_{s-1} \neq \tau^1_{s-1}$. The possible final positions of π^1_{s-1} and τ^1_{s-1} are $s-1$ and $s-5$ by observing Table III. Thus, there are the following three cases: (1) $\pi_{s-5} = \pi^1_{s-1}$ and $\tau_{s-1} = \tau^1_{s-1}$ (or symmetrically $\pi_{s-1} = \pi^1_{s-1}$ and $\tau_{s-5} = \tau^1_{s-1}$); (2)$\pi_{s-1} = \pi^1_{s-1}$ and $\tau_{s-1} = \tau^1_{s-1}$ and (3) $\pi_{s-5} = \pi^1_{s-1}$ and $\tau_{s-5} = \tau^1_{s-1}$. For (1), by Lemma 3, $\pi_{s-1} \neq \tau_{s-1}$. Thus, s can be covered with position $s-1$ and we set $g(s) = s-1$. For (2), it is obvious that s can be covered with position $s-1$ and we set $g(s) = s-1$. For (3), note that $x_{s-5} = y_{s-5} = 2$ by observing Table III. Thus s can be covered with position $j-5$ and we set $g(s) = s-5$.
- Case 3-4: [$\pi^1_s = \tau^1_s$ and $s = 8k+4+j$ for some $k \in \{0, 1, \cdots, m-1\}$ and $j \in \{3, 5\}$] Again we have $x_{s-1} = y_{s-1} = 1$ and $\pi^1_{s-1} \neq \tau^1_{s-1}$. By observing Table III, the possible final positions of π^1_{s-1} and τ^1_{s-1} are $s-1$, $s-5$, and $s-9$. If one of the final positions of π^1_{s-1} and τ^1_{s-1} is $s-1$, then s can be covered with position $s-1$ by Lemma 3 and we can set $g(s) = s-1$. Suppose that one of the final positions is $s-5$. With the same argument as in Subcase 3-2-II, s can be covered with position $s-5$ and we can set $g(s) = s-5$. Finally, suppose that both the final positions are $s-9$. With the same argument as of Subcase 3-2-III, s can be covered with position $s-9$ and we can set $g(s) = s-9$.

By the above analysis, we can set up a covering pattern g such that $g(s) = s$ if position s is self-covered and $g(s) \in \{s-1, s-4, s-5, s-8, s-9\}$ for each $s \in$ NSC. Furthermore, we show that $|g^{-1}(t) \cap \{t+1, t+4, t+5, t+8, t+9\}| \leq 1$ for any position t. We illustrate this in Table IV.

TABLE IV
NECESSARY CONDITIONS FOR POSITION COVERING

covered case	condition	covered case	condition
$g(t+4) = t$	$x_t = y_t = 2$ $x_{t+4} \neq y_{t+4}$	$g(t+5) = t$	$x_t = y_t = 2$ $x_{t+4} = y_{t+4} = 1$ $x_{t+5} \neq y_{t+5}$
$g(t+8) = t$	$x_t = y_t = 2$ $x_{t+4} = y_{t+4} = 2$ $x_{t+8} \neq y_{t+8}$	$g(t+9) = t$	$x_t = y_t = 2$ $x_{t+4} = y_{t+4} = 2$ $x_{t+8} = y_{t+8} = 1$ $x_{t+9} \neq y_{t+9}$
$g(t+1) = t$	$x_t = y_t = 1$ $x_{t+1} \neq y_{t+1}$		

In Table IV, we list the necessary conditions for the covering pattern g. Note that those conditions are all disjoint. This implies that $g^{-1}(t)$ contains at most one position in $\{t+1, t+4, t+5, t+8, t+9\}$ Therefore we complete the proof of Lemma 4. ∎

Recall that NSC$= \{s \in [8m] : \delta(x_s, y_s) > \delta(\pi_s, \tau_s)\}$. Based on Lemma 4, we show that g on NSC is a one-to-one function.

Lemma 5: Let g be the covering pattern obtained in Lemma 4. Then g : NSC $\rightarrow [8m]$ is a one-to-one function and $g(\text{NSC}) \cap \text{NSC} = \emptyset$, and hence $|g(\text{NSC})| = |\text{NSC}|$.

Proof: Assume that $g(s_1) = g(s_2) = t$. Thus we have $t \in \{s_1-1, s_1-4, s_1-5, s_1-8, s_1-9\} \cap \{s_2-1, s_2-4, s_2-5, s_2-8, s_2-9\}$. If $s_1 \neq s_2$, then $|g^{-1}(t) \cap \{t+1, t+4, t+5, t+8, t+9\}| \geq 2$ since s_1 and s_2 are both in the intersection. However, this is impossible by Lemma 4. Thus, $s_1 = s_2$ and hence g is one-to-one. By Table IV, if t covers some other position, then $x_t = y_t$. By definition, if t can be covered with some other position, then $x_t \neq y_t$. Thus it implies $g(\text{NSC}) \cap \text{NSC} = \emptyset$. Since g is one-to-one, we have $|g(\text{NSC})| = |\text{NSC}|$. ∎

Now we show the distance-preserving property of A_{8m}.

Theorem 1: A_{8m} is a 3-DPM for all $m \geq 2$.

Proof: Given $x, y \in Z_3^{8m}$, let $A_{8m}(x) = \pi$ and $A_{8m}(y) = \tau$. By Lemma 5, there exists a covering pattern g such that the following holds. First, for any $s \in \text{NSC}$, $\delta(x_s, y_s) = 1$ and $\delta(\pi_s, \tau_s) = 0$. Also for any $s \in g(\text{NSC})$, we have $\delta(x_s, y_s) = 0$ and $\delta(\pi_s, \tau_s) = 1$. Thus $\sum_{s\in\text{NSC}} \delta(x_s, y_s) + \sum_{s\in g(\text{NSC})} \delta(x_s, y_s) = \sum_{s\in\text{NSC}} \delta(\pi_s, \tau_s) + \sum_{s\in g(\text{NSC})} \delta(\pi_s, \tau_s)$ by Lemma 5. Then $d_H(x, y) = \sum_{s=1}^{8m} \delta(x_s, y_s) = \sum_{s\in\text{NSC}\cup g(\text{NSC})} \delta(x_s, y_s) + \sum_{s\notin\text{NSC}\cup g(\text{NSC})} \delta(x_s, y_s) \leq \sum_{s\in\text{NSC}\cup g(\text{NSC})} \delta(\pi_s, \tau_s) + \sum_{s\notin\text{NSC}\cup g(\text{NSC})} \delta(\pi_s, \tau_s) = \sum_{s=1}^{8m} \delta(\pi_s, \tau_s) = d_H(\pi, \tau)$. This completes the proof of Theorem 1. ∎

B. 3-DPM for input lengths at least 16

In this section, we modify our algorithm A_{8m} such that it can be applied to any input length at least 16. To achieve this goal, we need to show another property of algorithm A_{8m}. As in the previous section, let $\pi = A_{8m}(x)$ and π^1 be the intermediate result after PASS 1.

Lemma 6: For any $s \in \{1, 2, \cdots, 8m\}$, $\pi_s \neq s - 3$.

Proof: By way of contradiction, suppose that there is an s such that $\pi_s = s - 3$. Assume that $\pi_s = \pi^1_t = s - 3$ for some t. t must satisfy $4|(s - t)$. By the structure of PASS 1, $(s - 3) - 2 \leq t \leq (s - 3) + 2$. Thus, it must be the case that $t = s - 4$, that is $\pi_s = \pi^1_{s-4} = s - 3$. If $\pi_s = \pi^1_{s-4}$, then we have $x_{s-4} = 2$. However, if $\pi^1_{s-4} = s - 3$, then we have $x_{s-4} = 1$ by observing Table I. Hence, we get a contradiction. ∎

Now the 3-DPM A_{8m+k} is shown in the following.

Algorithm A_{8m+k} $\quad (8m \geq 16$ **,** $1 \leq k \leq 7)$ **:**
Input: $(x_1, \cdots, x_{8m+k}) \in Z_3^{8m+k}$
Output: $(\pi_1, \cdots, \pi_{8m+k}) \in S_{8m+k}$
$(\pi_1, \cdots, \pi_{8m}) \leftarrow A_{8m}(x_1, x_2 \cdots, x_{8m})$;
$(\pi_{8m+1}, \cdots, \pi_{8m+k}) \leftarrow (8m + 1, \cdots, 8m + k)$;
for $i = 1$ **to** k **do;**
  if $x_{8m+i} = 1$ **then swap** $(\pi_{8m+i}, \pi_{\pi^{-1}(i-3)})$;
  if $x_{8m+i} = 2$ **then swap** (π_{8m+i}, π_i);

We prove its correctness in the following theorem.

Theorem 2: $A_{8m+k} : Z_3^{8m+k} \to S_{8m+k}$ is a 3-DPM for all $n \geq 2$ and $k \in \{1, \cdots, 7\}$.

Proof: Given two inputs $(x, w), (y, z) \in Z_3^{8m} \times Z_3^k$, suppose that $\pi = A_{8m+k}(x, w)$ and $\tau = A_{8m+k}(y, z)$. Let w^i and z^i denote the first i symbols of w and z respectively. Let π^i and τ^i be the permutations in S_{8m+i} obtained by running the i-th iteration in the for loop when the inputs are (x, w) and (y, z) respectively. We claim that $d_H((x, w^i), (y, z^i)) \leq d_H(\pi^i, \tau^i)$ for any $i \in \{0, \cdots, k\}$. We prove it by induction on i. It holds trivially for $i = 0$ since we have $d_H(x, y) \leq d_H(A_{8m}(x), A_{8m}(y)) = d_H(\pi^0, \tau^0)$. For the inductive step, suppose that $d_H(x, y) + d_H(w^{i-1}, z^{i-1}) \leq d_H(\pi^{i-1}, \tau^{i-1})$. We divide the analysis into the following cases.

- Case 1: $[w_i = z_i]$ The lemma holds trivially in this case since both swap operations in the i-th iteration are the same.
- Case 2: $[w_i \neq z_i$ and one of them is 0$]$ W.L.O.G. we assume that $w_i = 0$. In this case we have $\pi^i_{8m+i} = 8m+i$, $\pi^i_{[1..8m+i-1]} = \pi^{i-1}$ and τ^i_{8m+i} equals to either $i - 3$ or τ^{i-1}_i. Thus we have $\delta(\pi^i_{8m+i}, \tau^i_{8m+i}) = 1$. W.L.O.G., we assume that $\tau^i_{8m+i} = \tau^{i-1}_i$ and hence $\tau^i_i = 8m + i$. So $\delta(\pi^i_i, \tau^i_i) = 1$. Also note that $\tau^i_t = \tau^{i-1}_t$ for any $t \in [8m+i-1]\backslash\{i\}$. So we have $d_H((x, w^i), (y, z^i)) \leq d_H(\pi^i, \tau^i)$.
- Case 3: $[w_i \neq z_i$, and $w_i, z_i \in \{1, 2\}]$ W.L.O.G. we assume that $w_i = 1$ and $z_i = 2$. In this case, $\pi^i_{8m+i} = i - 3$ and $\tau^i_{8m+i} = \tau^{i-1}_i = \tau_i$. By Lemma 6, we know that $\pi^{-1}(i - 3) \neq i$ and $\tau_i \neq i - 3$. Now it is easy to check $d_H(\pi^i, \tau^i) = d_H(\pi^{i-1}, \tau^{i-1}) + 1$. Hence we also have $d_H((x, w^i), (y, z^i)) \leq d_H(\pi^i, \tau^i)$.

Thus Theorem 2 follows from setting $i = k$. ∎

From Theorem 1 and Theorem 2, we get an explicit construction of 3-DPM.

Corollary 1: There exists an explicit construction of 3-DPM from Z_3^n to S_n for any $n \geq 16$.

The above approach may help us find explicit constructions of q-DPM for $q > 3$. However, we need a different Lemma 6 for different q in order to obtain an explicit construction, but we don't know how to prove the lemmas systematically so far.

III. Applications to Permutation Arrays

As shown in [3], [4], we know that distance-preserving mappings are quite helpful for constructing permutation arrays. With our construction, we have new permutation array lower bounds as follows.

Theorem 3: For all $n \geq 16$ and $d \leq n$, $P(n, d) \geq A_3(n, d)$.

Proof: Let C be a ternary code of length n with minimum distance d. Let $n \geq 16$. By Corollary 1, we have a distance-preserving mapping $f : Z_3^n \to S_n$. It is easy to see that $f(C)$ is a permutation array of length n with minimum distance d. Thus $P(n, d) \geq |f(C)| = |C|$. Therefore $P(n, d) \geq A_3(n, d)$. ∎

Here we give some comparison between $A(n, d - k)$ and $A_3(n, d)$ for $k < d$. First of all, we need the well-known asymptotic Gilbert-Varshamov bound.

Fact 1: (Theorem 2.10.8 in [7]) $A_3(n, d) \geq 3^{n(1-H_3(\frac{d}{n}))}$ for $d \leq \frac{2n}{3}$ and sufficiently large n where $H_3(x) = x \log_3 2 - x \log_3 x - (1 - x) \log_3(1 - x)$ for $0 < x \leq 1$.

Thus for $d \leq \frac{3n}{5}$, we get a lower bound of $P(n, d) = 3^{\Omega(n)}$. On the other hand, $A(n, d - k) \leq 2^n$ for any k. Thus, in this case, we significantly improve previous lower bounds in [3]. Since the minimum input length of the known DIM, which increases distance at least 2, is 16 (see [3]), we give a comparison between $A(16, d - 2)$ and $A_3(16, d)$ in Table V, where our lower bound of $P(16, d)$ is much larger than the previous lower bounds via DIMs.

IV. Conclusion and Open Problem

We have shown an explicit construction of distance-preserving mappings from ternary vectors to permutations (3-DPM). Our result answers an open question posed in [4]. We also obtain new lower bounds for permutation array size, which significantly improves previous lower bounds by DIMs for $d < 3n/5$. As in the binary case [3], we are interested in constructing distance-increasing mappings from ternary vectors to permutations. For the moment, it seems to be more complicated than the binary case. We leave it as an open problem.

TABLE V
$L[A_3(16,d)]$ STANDS FOR THE LOWER BOUND OF $A_3(16,d)$ IN [2] AND $U[A(16,d-2)]$ THE UPPER BOUND OF $A(16,d-2)$ IN [1].

d	3	4	5	6	7	8
$L[A_3(16,d)]$	1062882	216513	19683	6561	729	297
$U[A(16,d-2)]$	65536	32768	3276	2048	340	256
d	9	10	11	12	13	14
$L[A_3(16,d)]$	253	54	18	9	4	3
$U[A(16,d-2)]$	37	32	6	4	2	2

REFERENCES

[1] E. Agrell, A. Vardy, and K. Zeger, "A table of upper bounds for binary codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 3004–3006, Nov. 2001.

[2] A.E. Brouwer, H.O. Hämäläinen, P.R.J. Östergård, and N.J.A. Sloane, "Bounds on mixed binary/ternary codes," *IEEE Trans. Inf. Theory*, vol. 44, pp. 140–161, Jan. 1998.

[3] J.C. Chang, "Distance-increasing mappings from binary vectors to permutations that increase Hamming distances by at least two," *IEEE Trans Inf. Theory*, vol. 52, pp. 1683–1689, Apr. 2006.

[4] J.C. Chang, R.J. Chen, T. Kløve, and S.C. Tsai, "Distance-preserving mappings from binary vectors to permutations," *IEEE Trans. Inf. Theory*, vol. 49, pp. 1054–1059, Apr. 2003.

[5] J.S. Lin, J.C. Chang, R.J. Chen and T. Kløve, "Distance-preserving and distance-increasing mappings from ternary vectors to permutations," to appear in *IEEE Transactions on Information Theory*.

[6] T. G. Swart and H. C. Ferreira, "A generalized upper bound and a multilevel construction for distance-preserving mappings," *IEEE Trans. Inf. Theory*, vol. 52, pp. 3685–3695, Aug. 2006.

[7] W.C. Huffman and V. Pless, *Fundamentals of Error-Correcting Codes.* Cambridge, U.K.: Cambridge Univ. Press, 2003.

[8] K. Lee, "Cyclic constructions of distance-preserving maps," *IEEE Trans. Inf. Theory*, vol. 51, pp. 4392–4396, Dec. 2005.

[9] K. Lee, "Distance-increasing maps of all lengths by simple mapping algorithms," *IEEE Trans. on Inf. Theory*, vol. 52, pp. 3344–3348, Jul. 2006.