國立交通大學

資訊工程學系

碩士論文

藉由二元向量到排列的對映之排列碼建構法

On the Construction of Permutation Arrays via Mappings from Binary Vectors to Permutations

研究生：黃彥穎

指導教授：蔡錫鈞 教授

中華民國九十四年八月

藉由二元向量到排列的對映之排列碼建構法
On the Construction of Permutation Arrays via Mapppings from Binary Vectors to Permutations

研 究 生：黃彥穎　　Student：Yen-Ying Huang

指導教授：蔡錫鈞　　Advisor：Shi-Chun Tsai

國立交通大學
資訊工程學系
碩士論文

A Thesis
Submitted to Department of Computer Science and Information Engineering
College of Electrical Engineering and Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science and Information Engineering

August 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年八月

藉由二元向量到排列的對映之排列碼建構法

學生：黃彥穎　　　　　　　　　　　　指導教授：蔡錫鈞

國立交通大學資訊工程學系（研究所）碩士班

摘　　要

本論文中, 我們持續探討如何建構從二元向量到排列的對映. 我們成功地建構出改善目前最好的排列碼邊界的對映. 由於建構這些對映的流程大致相同, 我們提出一個如何建構這特定系列對映的大綱. 另外注意到我們的對映可能不是保持長度相等, 我們將說明非保持長度對映存在的必要性. 如果只使用保持長度的對映, 會失去很多可以改進的空間.

On The Construction of Permutation Arrays via Mappings from Binary Vectors to Permutations

Student：Yen-Ying Huang Advisors：Dr. Shi-Chun Tsai

Department of Computer Science and Information Engineering
National Chiao Tung University

Abstract

In this paper, we continue the study of constructing distance-increasing mappings from binary vectors to permutations. We successfully construct some mappings which induce better lower bounds for permutation codes than the current existing one : $P(n,r) \geq A(n,r\text{-}1)$. Since the approaches for constructing our mappings are similar, we give a framework for constructing a certain class of mappings. Note that our mappings may not be length-preserving. We will show the necessity of non-length-preserving mappings. We will lose many improvements if we only use length-preserving mappings.

誌謝

首先我要感謝我的指導教授蔡錫鈞老師, 在我就讀研究所的過程中, 經由他的指導, 我逐漸了解做研究的方法和程序. 並且在老師的帶領下, 不斷地督促和提供新的想法, 本論文才得以完成. 另外感謝口試委員呂及人博士和曾文貴教授批評指教, 點出論文不足的地方和指引一些新的方向, 使得本論文更趨完整. 最後感謝吳信龍學長的協助, 過去半年來不厭其煩地和我討論, 藉由他的幫助, 才使得主要論點一一浮現.

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

Yen-Ying Huang

August 5, 2005

Contents

Chapter 1

Introduction

1.1 Background

A permutation array(PA) is a set of permutations of $1, 2, ..., n$. From the combinatorial view, it is interesting to discuss the maximum size of PAs with given minimum distance. Early studies about PAs rose in the 70's. [6] and [7] are the well-known papers from that period. In 2000, by Vinck [12] a new application of PAs to a coding/modulation scheme for communication over power lines has leaded to the design of PAs, see also [9],[13],[14]. Thus the construction of PAs has gradually become more important. Recent discussions about the constructions of PAs are [4],[5],[8],[11],[15]. Since the permutation space is quite different from the Euclidean space which we are familiar with, it is not easy to have a systematic approach. Due to the unknown structure of permutation codes, we hope to investigate the permutation arrays by something we are familiar with : the M-ary codes. There are many good M-ary codes such as Reed-Soloman Codes etc. Suppose we have an efficient transformation from M-ary space to permutation space and vice versa. Then it is clear that we can construct a good permutation codes with desired minimum distance. This motivates the design of mappings from M-ary vectors to permutations.

For the mappings preserving length and $M = 2$, there are some results, see [3],[10],[2]. These papers introduced two kinds of mappings. One is the distance-preserving mapping(DPM)[3] and the other is the distance-increasing mapping(DIM)[2]. Preciesly, an n-DPM is a mapping from binary vectors to permutations of the same length n such that if the Hamming distance of two binary strings is d_0, then the Hamming distance of the corresponding permutations must be at least d_0. A n-DIM is quite like n-DPM except that when d_0 is less than the length of the string, the Hamming distance of the images of this two binary strings must be larger than d_0. Once we have a DPM (respectively DIM) f, for any binary code C with minimum distance r, it is easy to see that the image of C, i.e. $f(C)$, is a permutation array with minimum distance r,(respectively $r + 1$). Therefore by DPMs and DIMs, permutation arrays can be constructed straightly and the size of permutation codes can be bounded by the size of binary codes.

Why is DIM better than DPM? In order to construct a permutation code with minimum distance d_0, we only need a binary code with minimum distance $d_0 - 1$ if we have a DIM. On the contrary, we need a binary code with minimum distance d_0 when we are only given a DPM. We know that it is easier to construct a code with shorter minimum distance. From this point, our goal may be to construct a length-preserving mappings that is stronger than DIMs, that is a mapping which increases more distance than DIMs. However, this is not an easy task. In order to discuss the rest of the paper clearly, we introduce some necessary notations first.

1.2 Preliminaries and Notations

Let S_n denote the set of all permutations of $Z_n = \{1, 2, \cdots, n\}$ and the set Z_q^n denote the set of all q-ary vectors of length n. For a permutation $\pi = (\pi_1, \cdots, \pi_n) \in S_n$, let $\pi(i) = \pi_i$ and $\pi_{[i..j]}$ denote that sub-array $(\pi_i, \cdots, \pi_j)$ of π. For $i \in \{1, 2, \cdots, n\}$, $\pi^{-1}(i)$ denotes the position of i in π, i.e. if

$\pi(j) = i$ then $\pi^{-1}(i) = j$. The Hamming distance $d_H(a,b)$ between two n-tuples $a = (a_1, a_2, \cdots, a_n)$ and $b = (b_1, b_2, \cdots, b_n)$ is the number of positions where they differ, i.e.

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

We now define a class of distance-increasing mappings from q-ary vectors to permutations.

Definition 1.2.1. For $d \leq n+k$, an (n,d,k,q)-mapping $f : Z_q^n \rightarrow S_{n+k}$ is a mapping such that for all $x, y \in Z_q^n$,

$$\begin{array}{ll} d_H(f(x), f(y)) \geq d_H(x,y) + d, & \text{if } d_H(x,y) \leq (n+k) - d \\ d_H(f(x), f(y)) = n+k, & \text{if } d_H(x,y) > (n+k) - d \end{array}$$

Let $\mathcal{F}(n,d,k,q)$ denote the collection of all (n,d,k,q)-mappings.

Since we are more familiar with binary vectors, we simply ignore the last parameter q if $q = 2$, i.e. let $\mathcal{F}(n,d,k)$ denotes $\mathcal{F}(n,d,k,2)$. Clearly, the collection of DPMs is equal to $\mathcal{F}(n,0,0)$ and the collection of DIMs in [2] is equal to $\mathcal{F}(n,1,0)$. Let $n_{d,k,q}$ be the smallest integer such that for $n \geq n_{d,k,q}$, $\mathcal{F}(n,d,k,q)$ is not empty, and let $m_{d,k,q} = n_{d,k,q} + k$, i.e. the smallest image length. When we say that we have a series of $\mathcal{F}(n,d,k,q)$

1.3 Previous Results

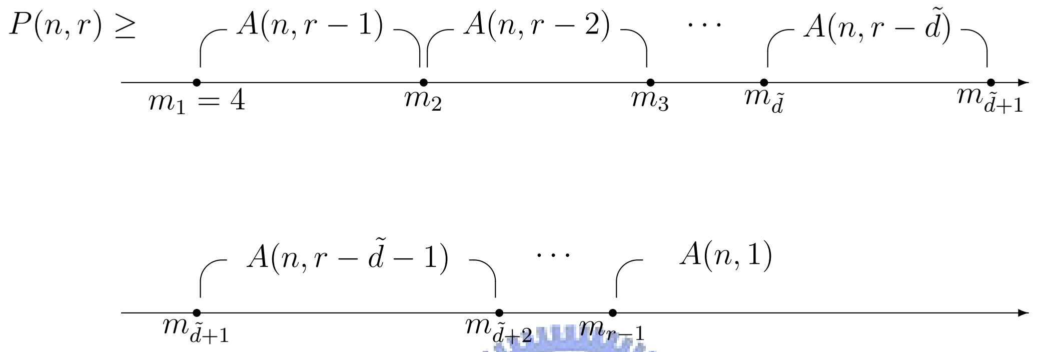
The concept of $(n,0,0)$-mappings(DPMs) was first proposed in [3]. But as the authors said, the inspiration came partly from the paper [9]. In [9], they found a $(4,0,0)$-mapping by computer search. Based on that mapping, they constructed $(n,0,0)$-mappings, for $5 \leq n \leq 8$. However the method couldn't be extended to $n > 8$. Later, the paper [3] generalized their results for $n \geq 4$ and gave two kinds of recursive constructions of $\mathcal{F}(n,0,0)$: One is that when given an $(m,0,0)$-mapping g and an $(n,0,0)$-mapping h, define $f : Z_2^{m+n} \rightarrow S_{m+n}$ as $f(x_1, \cdots, x_{m+n}) = (\pi_1, \cdots, \pi_m, \sigma_1 + m, \cdots, \sigma_n + m)$, where $\pi = g(x_1, \cdots, x_m)$ and $\sigma = h(x_{m+1}, \cdots, x_{m+n})$. Then f is an

$(m+n, 0, 0)$-mapping. Roughly speaking, it first concatenates the images of g and h then adjusts the values in the image of h. The other approach extends an $(n-1, 0, 0)$-mapping one-bit long, i.e. given one $(n-1)$-mapping f, it constructs an $(n, 0, 0)$-mapping f'. Assume we have a permutation $\pi = (\pi_1, \pi_2, \cdots, \pi_{n-1}) \in S_{n-1}$ which is the image of a binary string $s \in Z_2^{n-1}$. We extend s one bit long. If the extended bit is 0, then replace the value at the p-th entry with value n and append the replaced value to the right of π, and if the extend bit is 1, just append an entry of value n to the right of π. Later in [10], an alternative algorithm for constructing $(n, 0, 0)$-mappings of odd length was given.

In [2], a construction of $(n, 1, 0)$-mappings(DIMs) was given, which is similar to the first one in [3]. At the beginning, it does the two steps as that in [3] did. Then it starts to do some swap operations: if $x_1 = 1$, swap π_1 and $\sigma_n + m$, and if $x_{m+1} = 1$, swap π_n and $\sigma_1 + m$. These swap operations stands in order to remedy a bad situation: given two strings s_1, s_2, the first m bits are exactly the same, but the rest n bits are totally different(vice reverse). Concatenation is enough to produce an $(m+n, 0, 0)$-mapping when given an $(m, 0, 0)$-mapping g and an $(n, 0, 0)$-mapping h, but it is not enough to produce an $(m+n, 1, 0)$-mapping. In addition to swap operations, it becomes realized. Later the same author of [2] generalize the induction method for constructing $(n, 1, 0)$-mappings to $(n, d, 0)$-mappings, when $d > 1$, with only minor modification(more swap operations). The only problem is that to find the basis cases when $d > 1$ is really tough.

In both [3] and [2], the bound $P(n, r) \geq A(n, r-1)$ was given, for $n \geq 4$, where $P(n, r)$ denotes the maximal size among all permutation codes of length n and minimum distance r, and $A(n, r)$ denotes the maximal size among all binary codes. If one could construct the $(n, d, 0)$-mappings, for $d > 1$, the bound would be $P(n, r) \geq A(n, r-d)$. Let $m_{\tilde{d}}$ be the smallest integer such that for $n \geq m_{\tilde{d}}$, $\mathcal{F}(n, \tilde{d}, 0)$ is not empty. It is easy to see that

$m_{\tilde{d}} \leq m_{\tilde{d}+1}$ and $A(n, r-\tilde{d}-1) > A(n, r-\tilde{d})$. Thus for $m_{\tilde{d}} \leq n < m_{\tilde{d}+1}$, we can only apply $P(n,r) \geq A(n, r-\tilde{d})$, but for $m_{\tilde{d}+1} \leq n$, we can apply $P(n,r) \geq A(n, r-\tilde{d}-1)$. We plot the best bound for $P(n,r)$ induced by $\mathcal{F}(n,d,k)$ in the following diagram which we call the $P(n,r)$-diagram. In Chapter 3, we will illustrate our results by the $P(n,r)$-diagram many times.



1.4 Our Results

We successfully construct a series of $\mathcal{F}(n,2,1)$ and a series of $\mathcal{F}(n,3,2)$, and the bounds induced by $\mathcal{F}(n,2,1)$ and $\mathcal{F}(n,3,2)$ beat the current existing bound. Moreover we propose a framework for constructing $\mathcal{F}(n,k+1,k)$. Follow the steps we state, a series of $\mathcal{F}(n,k+1,k)$ might be built. As we involve one more parameter k, we show that by using the non-length-preserving mappings, there are many cases with the bound improved. In other words, several improvements will be lost if we abandon non-length-preserving mappings.

1.5 Organization of this paper

In Chapter 2, we discuss the constructions of $\mathcal{F}(n,d,k)$ of different settings of (d,k). First we give an alternative construction of $\mathcal{F}(n,0,0)$. Second we give two new constructions of $\mathcal{F}(n,1,0)$ by using the substitution technique.

Third we give the constructions of $\mathcal{F}(n,2,1)$ and $\mathcal{F}(n,3,2)$, and propose a framework for constructing $\mathcal{F}(n,k+1,k)$. In Chapter 3, we show how the bound induced by $\mathcal{F}(n,2,1)$ and $\mathcal{F}(n,3,2)$ beats the previous bound, and the reason why we should discuss the $\mathcal{F}(n,d,k)$ of different settings of (n,k) rather than just $\mathcal{F}(n,d,0)$. In Chapter 4, we will talk about the possible future works and make some conclusions.

Chapter 2

Mappings from Z_2^n to S_{n+k}

In this chapter, we discuss the constructions of various kinds of (n, d, k)-mappings, including constructions of basis cases and induction methods. These mappings are useful for constructing permutation arrays. We will discuss these applications in Chapter 3.

2.1 Construction of $\mathcal{F}(n, 0, 0)$

In [3], we have already known two kinds of recursive constructions of $(n, 0, 0)$-mappings: given one $(m, 0, 0)$-mapping g and one $(n, 0, 0)$-mapping h, constructing an $(m + n, 0, 0)$-mapping f, and given an $(n - 1, 0, 0)$-mapping g, constructing an $(n, 0, 0)$-mapping f. In this section we give a new construction method which is similar to the second one in [3]. We prove by Lemma 2.1.1.

Lemma 2.1.1. *[3] Given $g \in \mathcal{F}(n-1, 0, 0)$ and $f : Z_2^n \to S_n$, if f satisfies the following inequality, for all $x, y \in Z_2^{n-1}$, and $x_n, y_n =$ 0 or 1:*

$$d_H(f(x, x_n), f(y, y_n)) \geq d_H(g(x), g(y)) + d_H(x_n, y_n) \qquad (*)$$

then $f \in \mathcal{F}(n, 0, 0)$.

Our construction mainly depends on a switching operation. Assume we have a permutation $\pi = (\pi_1, \pi_2, \cdots, \pi_{n-1}) \in S_{n-1}$ which is the image of a binary string s. If we extend s one bit long, we want π to be also extended one bit long. The idea is that if the extended bit is 0, then replace value $n-1$ with n and append a entry of value $n-1$ to the right of π, and if the extend bit is 1, just append a entry of value n to the right of π. The formal definition is as follows.

Definition 2.1.2. We define an operation p, which assigns new value to each entry of a n-tuple according to the relative ordering of their original values. For $(\pi_1, \pi_2, \cdots, \pi_{n-1}) \in S_{n-1}$, assume $\pi_k = n-1$. Define $p : S_{n-1} \times \{0,1\} \to S_n$ by

$$p(\pi_1, \cdots, \pi_{k-1}, n-1, \pi_{k+1}, \cdots, \pi_{n-1}, 0) = (\pi_1, \cdots, \pi_{k-1}, n, \pi_{k+1}, \cdots, \pi_{n-1}, n-1),$$

$$p(\pi_1, \cdots, \pi_{k-1}, n-1, \pi_{k+1}, \cdots, \pi_{n-1}, 1) = (\pi_1, \cdots, \pi_{k-1}, n-1, \pi_{k+1}, \cdots, \pi_{n-1}, n).$$

Construction 2.1.3. *Let $g \in \mathcal{F}(n-1, 0, 0)$. Define $f : Z_2^n \to S_n$ by*

$$f(x, i) = p(g(x), i), \qquad x \in Z_2^{n-1}, i = 0, 1$$

where p is the operation defined above. Then $f \in \mathcal{F}(n, 0, 0)$.

Proof. (**correctness of Construction 2.1.3**)
Let $(x, x_n), (y, y_n) \in Z_2^n$, where $x, y \in Z_2^{n-1}$. Let $g(x) = \rho = (\rho_1, \rho_2, \cdots, \rho_{n-1})$ and assume $\rho_i = n-1$. Let $g(y) = \tau = (\tau_1, \tau_2, \cdots, \tau_{n-1})$ and assume $\tau_j = n-1$. We consider the following cases according to the values of x_n and y_n.

- Case $[x_n = y_n = k]$:

$$\begin{aligned} & d_H(f(x,0), f(y,0)) \\ = \; & d_H(p(g(x),0), p(g(y),0)) \\ = \; & d_H((\rho_{[1..i-1]}, n, \rho_{[i+1..n-1]}, n-1), (\tau_{[1..j-1]}, n, \tau_{[j+1..n-1]}, n-1)) \\ = \; & d_H(g(x), g(y)). \end{aligned}$$

$$\begin{aligned} & d_H(f(x,1), f(y,1)) \\ = \; & d_H(p(g(x),1), p(g(y),1)) \\ = \; & d_H((\rho_{[1..n-1]}, n), (\tau_{[1..n-1]}, n)) \\ = \; & d_H(g(x), g(y)). \end{aligned}$$

$\Rightarrow d_H(p(g(x),k), p(g(y),k)) = d_H(g(x), g(y))$.

- Case $[x_n = 1, y_n = 0]$:
If $i = j$

$$\begin{aligned} & d_H(f(x,1), f(y,0)) \\ = \; & d_H(p(g(x),1), p(g(y),0)) \\ = \; & d_H((\rho_{[1..i-1]}, n-1, \rho_{[i+1..n-1]}, n), (\tau_{[1..i-1]}, n, \tau_{[i+1..n-1]}, n-1))) \\ = \; & d_H(g(x), g(y)) + 2. \end{aligned}$$

If $i \neq j$

$$\begin{aligned} & d_H(f(x,1), f(y,0)) \\ = \; & d_H(p(g(x),1), p(g(y),0)) \\ = \; & d_H((\rho_{[1..i-1]}, n-1, \rho_{[i+1..n-1]}, n), (\tau_{[1..j-1]}, n, \tau_{[j+1..n-1]}, n-1))) \\ = \; & d_H(g(x), g(y)) + 1. \end{aligned}$$

- Case $[x_n = 0, y_n = 1]$:
  Symmetrically, we have $d_H(f(x, x_n), f(y, y_n)) \geq d_H(g(x), g(y)) + 1$.

  $\Rightarrow d_H(p(g(x), k), p(g(y), \overline{k}) \geq d_H(g(x), g(y)) + 1$.

The inequality (*) holds for all cases, thus we have $f \in \mathcal{F}(n, 0, 0)$. □

2.2 Construction of $\mathcal{F}(n, 1, 0)$

In [2], given an $(m, 1, 0)$-mapping g, and an $(n, 1, 0)$-mapping h, through concatenation with some switchings, an $(m + n, 1, 0)$-mapping f can be constructed. However the method needs many basis cases. In this section, we will use the substitution technique to give a new construction of $\mathcal{F}(n, 1, 0)$, and our methods need fewer basis cases than that in [2]. We give two almost-sequential constructions of $(n, 1, 0)$-mappings. We give the first one below. As it is known that the smallest integer n_0 such that for $n \geq n_0$, $\mathcal{F}(n, 1, 0)$ is not empty is 4. The goal is that, given an $(4, 1, 0)$-mapping h and an $(n, 1, 0)$-mapping g, we will construct an $(n + 3, 1, 0)$-mapping f. Our main idea is that when given a binary string of length $n + 3$, make the first n bits as the input of g, and the last 4 bits as the input of h. Note that the n-th bit is taken as input bit for both g and h.

Construction 2.2.1. *Let $g \in \mathcal{F}(n, 1, 0)$ and $h \in \mathcal{F}(4, 1, 0)$. By the following algorithm, a mapping $f \in \mathcal{F}(n + 3, 1, 0)$ is constructed.*

```
Input: $(x_1, \cdots, x_n, \cdots, x_{n+3}) \in Z_2^{n+3}$
Output: $(\pi_1, \cdots, \pi_{n+3}) = f(x_1, \cdots, x_{n+3})$
begin
0    Let $\rho = g(x_1, \cdots, x_n), \tau = h(x_n, \cdots, x_{n+3})$
1    $\tau_i = \tau_i + n - 1$, for $1 \leq i \leq 4$
2    $\tau_{\tau^{-1}(n)} = \rho_n$
```

```
3    $(\pi_1, \cdots, \pi_{n-1}) = \rho_{[1..n-1]}$
4    $(\pi_n, \cdots, \pi_{n+3}) = \tau_{[1..4]}$
end
```

Example 2.2.2. *Let* $g(0,1,0,0,1) = (1,5,2,4,3)$, $h(1,1,0,1) = (2,4,1,3)$. *Then* $f(0,1,0,0,1,1,0,1) = (1,5,2,4,6,8,3,7)$

Proof. (**correctness of Construction 2.2.1**)
First note that $\rho_i \in \{1,\cdots,n\}$, and after line 1, $\tau_i \in \{n,\cdots,n+3\}$. But at line 2, the value n of τ is substituted with ρ_n. Thus the rest values in τ range from $n+1$ to $n+3$.

Let $(x,w),(y,z) \in Z_2^{n+3}$, where $x,y \in Z_2^n$, and $w,z \in Z_2^3$. Let $g(x) = \rho = (\rho_1,\cdots,\rho_n)$, $g(y) = \rho' = (\rho'_1,\cdots,\rho'_n)$, $h(x_n,w) = \tau = (\tau_1,\cdots,\tau_4)$, and $h(y_n,z) = \tau' = (\tau'_1,\cdots,\tau'_4)$. And $f(x,w) = \pi = (\pi_1,\cdots,\pi_{n+3})$, $f(y,z) = \pi' = (\pi'_1,\cdots,\pi'_{n+3})$.

Now we explain the effect of the operation at line 2. Since ρ_n and ρ'_n are from $\{1,\cdots,n\}$, after the substitution, if $\tau^{-1}(n) = \tau'^{-1}(n)$ then ρ_n and ρ'_n are still in the same coordinate and the distance of this coordinate is at least preserved, else ρ_n and ρ'_n correspond to a value from $\{n+1,\cdots,n+3\}$. Thus after substituting operation at line 2, the distance $d_H(\tau,\tau')$ won't decrease.

Next we consider the following cases:

- case $d_H(x,y) = 0$: We know that $d_H(w,z) \neq 0$, otherwise $(x,w),(y,z)$ are identical. Let $d_H(w,z) = t \leq 3$. $d_H(x,y) = 0$ implies that $x_n = y_n$. So $d_H((x_n,w),(y_n,z)) = t \leq 3$. Since $h \in \mathcal{F}(4,1,0)$, we have $d_H(\tau,\tau') \geq t+1$. Therefore $d_H(\pi,\pi') \geq t+1 = d_H((x,w),(y,z))+1$.

- case $0 < d_H(x,y) = s < n$: In this case, we have $d_H(\rho,\rho') \geq s+1$, thus $d_H(\rho_{[1..n-1]},\rho'_{[1..n-1]}) \geq s$. Let $d_H(w,z) = t$. If $x_n = y_n$ and $t = 0$, it is easy to see $d_H(\pi,\pi') \geq s+1 = d_H((x,w),(y,z))+1$. If $x_n = y_n$ and $0 < t \leq 3$, then $d_H(\tau,\tau') \geq t+1$. Therefore $d_H(\pi,\pi') \geq s+(t+1) =$

$d_H((x,w),(y,z))+1$. If $x_n \neq y_n$, no matter what value t is, $d_H(\tau,\tau') \geq t+1$. Therefore $d_H(\pi,\pi') \geq s+(t+1) = d_H((x,w),(y,z))+1$.

- case $d_H(x,y) = n$: If $d_H(w,z) = t \leq 2$, then $d_H((x_n,w),(y_n,z)) = t+1 \leq 3$. By the definition $d_H(\tau,\tau') \geq t+2$. Therefore $d_H(\pi,\pi') \geq (n-1)+(t+2) = d_H((x,w),(y,z))+1$. If $d_H(w,z) = 3$, it is easy to check that $d_H(\pi,\pi') = n+3$.

□

We can simply generalize Construction 2.2.1 for the q-ary edition. Miner modifications are needed for the proof above. We will show the statement next section.

Through Construction 2.2.1, if we have three basis cases, we can construct a series of $\mathcal{F}(n,1,0)$. Next we give another almost-sequential construction method which extend 2-bit longer from the basis mapping. We will use an auxiliary mapping E as help.

Construction 2.2.3. *Let $g \in \mathcal{F}(n,1,0)$ and E defined as follows:*

x	$E(x)$	x	$E(x)$	x	$E(x)$	x	$E(x)$
00	$(1,2,3)$	10	$(2,1,3)$	01	$(1,3,2)$	11	$(3,1,2)$

Note that $E \in \mathcal{F}(2,1,1)$ and value 1 in any permutation in $E(Z_2^2)$ only appears in coordinate 1 or coordinate 2. By the following algorithm, a mapping $f \in \mathcal{F}(n+2,1,0)$ is constructed.

```
Input: $(x_1,\cdots,x_n,\cdots,x_{n+2}) \in Z_2^{n+2}$
Output: $(\pi_1,\cdots,\pi_{n+2}) = f(x_1,\cdots,x_{n+2})$
begin
    Let $\rho = g(x_1,\cdots,x_n), \tau = E(x_{n+1},x_{n+2})$
    $\tau_i = \tau_i + n - 1$, for $1 \leq i \leq 3$
    $\tau_{\tau^{-1}(n)} = \rho_n$
```

```
3    $(\pi_1, \cdots, \pi_{n-1}) = \rho_{[1..n-1]}$
4    $(\pi_n, \cdots, \pi_{n+2}) = \tau_{[1..3]}$
5    if $x_1 = 1$ then swap $(\pi_1, \pi_{n+2})$
end
```

Proof. (**correctness of Construction 2.2.3**) First note that $\rho_i \in \{1, \cdots, n\}$, and after line 1, τ_i's range from n to $n+2$. Since either τ_1 or τ_2 equals to n, τ_3 equals to $n+1$ or $n+2$.

Let $x, y \in Z_2^{n+2}$. Let $B(x_{[1..n]}) = \rho = (\rho_1, \cdots, \rho_n)$, $B(y_{[1..n]}) = \rho' = (\rho'_1, \cdots, \rho'_n)$, $E(x_{n+1}, x_{n+2}) = \tau = (\tau_1, \tau_2, \tau_3)$, and $E(y_{n+1}, y_{n+2}) = \tau' = (\tau'_1, \tau'_2, \tau'_3)$. And $f(x) = \pi = (\pi_1, \cdots, \pi_{n+2})$, $f(y) = \pi' = (\pi'_1, \cdots, \pi'_{n+2})$.

Let's first explain the change of the distance due to the swap step at line 5. If both $x_1 = 1$ and $y_1 = 1$ or both $x_1 = 0$ and $y_1 = 0$, the distance of these two coordinates remains the same. If exact one of x_1 and y_1 equals to 1, as we know that the possible values of π_1 and π'_1 range from 1 to n and π_{n+2} and π'_{n+2} equal to $n+1$ or $n+2$, thus the distance of these two coordinates won't decrease. Therefore after the swap step the distance of this two coordinates is at least the same as before. In some cases, the distance even grows up. Thus for the rest part of our proof, we won't discuss the effect due to the swap step if not necessary.

Now we explain the effect of the operation at line 2. Since the values of ρ_n and ρ'_n fall between 1 and n, after the substitution, if $\tau^{-1}(n) = \tau'^{-1}(n)$ then ρ_n and ρ'_n are still in the same coordinate and the distance of this coordinate is at least preserved, else ρ_n and ρ'_n correspond to $n+1$ or $n+2$. Thus after substituting operation at line 2, the distance $d_H(\tau, \tau')$ won't decrease.

Now we consider the following cases:

- case $d_H(x_{[1..n]}, y_{[1..n]}) = 0$: We know that $d_H((x_{n+1}, x_{n+2}), (y_{n+1}, y_{n+2})) \neq 0$, otherwise x and y are identical. Let $d_H(w, z) = t(= 1 \text{ or } 2)$. Since

$E \in \mathcal{F}(2,1,1)$, we have $d_H(\tau, \tau') \geq t+1$. Therefore $d(\pi, \pi')) \geq t+1 = d_H(x,y)+1$.

- case $0 < d_H(x_{[1..n]}, y_{[1..n]}) = s < n$: Since $g \in \mathcal{F}(n,1,0)$, we have $d_H(\rho, \rho') \geq s+1$. If $d_H((x_{n+1}, x_{n+2}), (y_{n+1}, y_{n+2})) = 0$, it is easy to see that $d_H(\pi, \pi') \geq s+1 = d_H(x,y)+1$. If $d_H((x_{n+1}, x_{n+2}), (y_{n+1}, y_{n+2})) = t(= 1$ or $2)$, then $d_H(\tau, \tau') \geq t+1$. $d_H(\pi, \pi') = d_H(\pi_{[1..n-1]}, \pi'_{[1..n-1]}) + d_H(\pi_{[n..n+2]}, \pi'_{[n..n+2]}) \geq s + (t+1) = d_H(x,y)+1$.

- case $d_H(x_{[1..n]}, y_{[1..n]}) = n$: Let $d_H((x_{n+1}, x_{n+2}), (y_{n+1}, y_{n+2})) = t$. If $t = 0$, before line 5, $d_H(\pi, \pi') = n$ and $\pi_{n+2} = \pi'_{n+2}$. Since $d_H(x_{[1..n]}, y_{[1..n]}) = n$, $x_1 \neq y_1$, one of the permutations will be swapped. Therefore after line 5, $d_H(\pi, \pi') = n+1$(now $\pi_{n+2} \neq \pi'_{n+2}$). If $t = 1$, then $d_H(\rho, \rho') = 2$. Before the substitution, suppose value n in ρ does correspond to value n in ρ', then after the substitution , $d_H(\rho, \rho') = 3$(since $\pi_n \neq \pi'_n$). Therefore $d_H(\pi, \pi') = (n-1)+3 \geq d_H(x,y)+1$. Suppose value n in ρ does not correspond to value n in ρ', then $\rho_3 = \rho'_3$. The remain proof is the same as the case when $t = 0$. If $t = 2$, then $d_H(\rho, \rho') = 3$. It is easy to check that $d_H(\pi, \pi') = n+2$.

□

By Construction 2.2.3, we will only need two basis cases to fulfill a series of $\mathcal{F}(n,1,0)$. The number of basis cases needed is fewer than that by Construction 2.2.1. But as we have already said, Construction 2.2.1 can be generalized to the q-ary edition. That is its advantage. Compare our almost-sequential induction methods with the concatenation method in [2], from the aspect of construction time, our constructions run slowlier. Nevertheless the number of basis cases needed in [2] is at least 4. If Construction 2.2.3 is adopted, we save almost half of the space.

The swap operation will be used very often for the rest constructions in this chapter. From the previous discussion, as long as the ranges of the values of these two swapped coordinate don't intersect, the distance of these two coordinates won't decrease. Therefore after many swap steps the distance of these coordinates is at least the same as before. Thus for all the remaining proofs, we won't discuss the effect due to the swap operations if not necessary.

2.3 Construction of $\mathcal{F}(n, d, 0, q)$

As we said in previous section, Construction 2.2.1 can be generalized to the q-ary edition. Next we show the statement without proof. We simply use $m_{d,q}$ instead of $m_{d,k,q}$ defined in the preliminary only in this section for convenient.

Construction 2.3.1. *Let $m = m_{1,q}$. Let $g \in \mathcal{F}(n, 1, 0, q)$ and $h \in \mathcal{F}(m, 1, 0, q)$. We construct a mapping $f \in \mathcal{F}(n+m-1, 1, 0, q)$. The construction algorithm is exact the same as that in Construction 2.2.1.*

In fact, we can even generalize Construction 2.3.1 to the d-distance-increasing edition. But there are some limitations about the choice of the mapping h.

Construction 2.3.2. *Let $g \in \mathcal{F}(n, d, 0, q)$. And let m be the smallest integer such that the function $h \in \mathcal{F}(m, d, 0, q)$ with the following limitations exists: $\forall i, j, 1 \leq i, j \leq d, i \neq j$, $\{\pi^{-1}(i) | \pi \in h(Z_2^m)\} \cap \{\pi^{-1}(j) | \pi \in h(Z_2^m)\} = \emptyset$. By the following algorithm, a mapping $f \in \mathcal{F}(n+m-d, d, 0, q)$ is constructed.*

```
Input: $(x_1, \cdots, x_n, \cdots, x_{n+m-d}) \in Z_q^{n+m-d}$
Output: $(\pi_1, \cdots, \pi_{n+m-d}) = f(x_1, \cdots, x_{n+m-d})$
begin
    Let $\rho = g(x_1, \cdots, x_n), \tau = h(x_{n-d+1}, \cdots, x_{n+m-d})$
    $\tau_i = \tau_i + n - d$, for $1 \leq i \leq m$
    $\tau_{\tau^{-1}(n-d+i)} = \rho_{n-d+i}$, for $1 \leq i \leq d$
    $(\pi_1, \cdots, \pi_{n-d}) = \rho_{[1..n-d]}$
    $(\pi_{n+1-d}, \cdots, \pi_{n+m-d}) = \tau_{[1..m]}$
end
```

Proof. (**correctness of Construction 2.3.2**) First note that ρ_i's range from 1 to n, and after line 1, and τ_i's range from $n-d+1$ to $n-d+m$. But at line 2, value $n-d+i$ of τ is substituted by ρ_{n-d+i}, for $1 \leq i \leq d$. The rest values in τ range from $n+1$ to $n-d+m$.

Let $(x,w),(y,z) \in Z_q^{n+m-d}$, where $x,y \in Z_q^n$, and $w,z \in Z_2^{m-d}$. Let $g(x) = \rho = (\rho_1, \cdots, \rho_n)$, $g(y) = \rho' = (\rho'_1, \cdots, \rho'_n)$, $h(x_{[n-d+1..n]}, w) = \tau = (\tau_1, \cdots, \tau_m)$, and $h(y_{[n-d+1..n]}, z) = \tau' = (\tau'_1, \cdots, \tau'_m)$. And $f(x,w) = \pi = (\pi_1, \cdots, \pi_{n+m-d})$, $f(y,z) = \pi' = (\pi'_1, \cdots, \pi'_{n+m-d})$.

Now we dicuss the effect of the operation at line 2. Since the values of ρ_{n-d+i} and ρ'_{n-d+i} are from $\{1, \cdots, n\}$, after the substitution, if $\tau^{-1}(n-d+i) = \tau'^{-1}(n-d+i)$, then ρ_{n-d+i} and ρ'_{n-d+i} are still in the same coordinate and the distance of this coordinate is at least preserved, else ρ_{n-d+i} and ρ'_{n-d+i} correspond to a value from $\{n+1, \cdots, n-d+m\}$ (note that value $n-d+j$, for $1 \leq j \leq d$, $i \neq j$ is impossible, since value $n-d+j$ must not be in the coordinate where value $n-d+i$ lies due to the limitations). Thus after substituting operation at line 2, the distance $d_H(\tau, \tau')$ won't decrease.

Next we consider the following cases:

- case $d_H(x,y) = 0$: We know that $d_H(w,z) \neq 0$, otherwise $(x,w),(y,z)$ are identical. Let $d_H(w,z) = t \leq m-d$. $d_H(x,y) = 0$ implies that $x_{[n-d+1..n]} = y_{[n-d+1..n]}$. So $d_H((x_{[n-d+1..n]}, w), (y_{[n-d+1..n]}, z)) = t \leq m-d$. Since $h \in \mathcal{F}(m,d,0,q)$, we have $d_H(\tau, \tau') \geq t+d$. Therefore $d(\pi, \pi') \geq t+d = d_H((x,w),(y,z)) + d$.

- case $0 < d_H(x,y) = s \leq n-d$: We have $d_H(\rho, \rho') \geq s+d$, thus $d_H(\rho_{[1..n-1]}, \rho'_{[1..n-1]}) \geq s$. Let $d_H(x_{[n-d+1..n]}, y_{[n-d+1..n]}) = c$ and $d_H(w,z) = t$. If $c = 0$ and $t = 0$, it is easy to see $d_H(\pi, \pi') \geq s+d = d_H((x,w),(y,z))+d$. If $c = 0$ and $0 < t \leq m-d$, then $d_H(\tau, \tau') \geq t+d$. Therefore $d_H(\pi, \pi') \geq s + (t+d) = d_H((x,w),(y,z)) + d$. If $c > 0$, no matter what value t is, $d_H(\tau,\tau') \geq t+d$. Therefore $d_H(\pi,\pi') \geq s+(t+d) = d_H((x,w),(y,z))+d$.

- case $n-d+1 \leq d_H(x,y) \leq n$: We know that $d_H(\rho,\rho') = n$, thus $d_H(\rho_{[1..n-d]},\rho'_{[1..n-d]}) = n-d$. Let $d_H(x_{[n-d+1..n]},y_{[n-d+1..n]}) = c$. c is at least 1, otherwise $d_H(x,y) \leq n-d$, If $d_H(w,z) = t \leq m-d-c$, then $d_H((x_{[n-d+1..n]},w),(y_{[n-d+1..n]},z)) = t+c \leq m-d$. By the definition $d_H(\tau,\tau') \geq t+c+d$. Therefore $d_H(\pi,\pi') \geq (n-d)+(t+c+d) \geq d_H((x,w),(y,z))+d$. If $d_H(w,z) > m-d-c$, then $d_H((x_{[n-d+1..n]},w),(y_{[n-d+1..n]},z)) = t+c > m-d$. By definition $d_H(\tau,\tau')) = m$. It is easy to check that $d_H(\pi,\pi') = n-d+m$.

□

Although Construction 2.3.2 is really general, there are still many obstacles. First it is hard to find the mappings h when $d > 1$. Second it is also not easy to figure out the basis cases when $d > 1$. What we are discussing is only theoretical. But once these mappings have been found, a series of $\mathcal{F}(n,d,0,q)$ are easily produced.

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

In this section, we give a systematic study on the construction of the class, $\mathcal{F}(n,2,1)$. First we will give the basic constructions: $g_6 \in \mathcal{F}(6,2,1)$, $g_7 \in \mathcal{F}(7,2,1)$, $g_8 \in \mathcal{F}(8,2,1)$, $g_9 \in \mathcal{F}(9,2,1)$. Then, we can inductively construct $g_{n+4} \in \mathcal{F}(n+4,2,1)$ from a mapping $g_n \in \mathcal{F}(n,2,1)$. Thus finally we have a series of $\mathcal{F}(n,2,1)$, for $n \geq 6$.

Consider two auxiliary mappings $A_6 \in \mathcal{F}(2,2,2)$ and $B_6 \in \mathcal{F}(4,2,2)$. We construct g_6 with these two mappings. Similarly, for each of g_7, g_8 and g_9, we will use two auxiliary mappings for the constructions. Note that in the image of A_6, 4 only appears in coordinate 1 or 2 . Similarly, in the image of B_6 the value 1 only appears in coordinate 1 or 2, and the value 2 only appears in coordinate 3 or 4. We call such property **the position property** and we give the formal definition as follows.

Definition 2.4.1. We say that a mapping $f \in \mathcal{F}(n,d,k,q)$ has the position property for $\{v_1, v_2, \cdots, v_p\} \subseteq \{1, 2, \cdots, n+k\}$ if $\forall i, 1 \le i \le p, |\{\pi^{-1}(v_i) | \pi \in f(Z_2^n)\}| = 2$ and $\forall i,j, 1 \le i,j \le p, i \ne j, \{\pi^{-1}(v_i) | \pi \in f(Z_2^n)\} \cap \{\pi^{-1}(v_j) | \pi \in f(Z_2^n)\} = \emptyset$.

A_6 has the position property for $\{4\}$ and B_6 has the position property for $\{1,2\}$. With this observation, we can construct a mapping $g_6 \in \mathcal{F}(6,2,1)$.

Construction 2.4.2. *Let $A_6 : Z_2^2 \to S_4$ and $B_6 : Z_2^4 \to S_6$ 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	$(4,1,6,2,3,5)$

By the following algorithm, a mapping $g_6 \in \mathcal{F}(6,2,1)$ is constructed.

```
Input: $(x_1, x_2, \cdots, x_6) \in Z_2^6$
Output: $(\pi_1, \cdots, \pi_7) = g_6(x_1, \cdots, x_6)$
begin
0    $\rho = A_6(x_1, x_2); \tau = B_6(x_3, x_4, x_5, x_6);$
```

```
$\tau_i = \tau_i + 1$ for $1 \le i \le 6$;
$\rho_{\rho^{-1}(4)} = \tau_6$;
$\tau_{\tau^{-1}(2)} = \rho_3$;
$\tau_{\tau^{-1}(3)} = \rho_4$;
$(\pi_1, \pi_2) = \rho_{[1..2]}$;
$(\pi_3, \pi_4, \pi_5, \pi_6, \pi_7) = \tau_{[1..5]}$;
end
```

Besides the position property, B_6 holds another property that if the Hamming distance of two binary vectors is 3, then the 5th entries of the images must be different, i.e. for $x, y \in Z_2^4$, if $d_H(x,y) = 3$, then $B_6(x)_5 \neq B_6(y)_5$. Any mapping holds these properties could be B_6.

Similar to [2], given $g \in \mathcal{F}(2,n,1)$, 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 $d_H(g(x), g(y)) = j$. The expansion matrix not only reveal the distance-increasing property, but also helps us to check the correctness. If you implement Construction 2.4.2, you will get the same distance expansion matrix. Our D is a little bit different from those in previous works. Our D is an $n \times (n+1)$ matrix, instead of $n \times n$ matrix. Since the permutations in the range of g is one dimension larger than the domain of g. We show the distance expansion matrix D for g_6 is 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

Construction 2.4.3. *Let $A_7 : Z_2^3 \rightarrow S_5$ is defined as follows and B_7 is 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, a mapping $g_6 \in \mathcal{F}(6,2,1)$ *is constructed.*

```
Input: $(x_1, x_2, \cdots, x_7) \in Z_2^7$
Output: $(\pi_1, \cdots, \pi_8) = g_7(x_1, \cdots, x_7)$
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 $1 \le i \le 6$;
    $\rho_{\rho^{-1}(5)} = \tau_6$;
    $\tau_{\tau^{-1}(3)} = \rho_4$;
    $\tau_{\tau^{-1}(4)} = \rho_5$;
    $(\pi_1, \pi_2, \pi_3) = \rho_{[1..3]}$;
    $(\pi_4, \pi_5, \pi_6, \pi_7, \pi_8) = \tau_{[1..5]}$;
    if $x_1 = 1$ then swap $(\pi_3, \pi_8)$;
end
```

In fact, B_7 could be simpler than B_6, any mapping that holds the position property would work. The distance expansion matrix D for g_7 is as follows:

$$\begin{array}{|cccccccc|}
0 & 0 & 256 & 128 & 32 & 32 & 0 & 0 \\
 & 0 & 0 & 448 & 208 & 336 & 272 & 80 \\
 & & 0 & 0 & 224 & 912 & 736 & 368 \\
 & & & 0 & 0 & 224 & 1184 & 832 \\
 & & & & 0 & 0 & 320 & 1024 \\
 & & & & & 0 & 0 & 448 \\
 & & & & & & 0 & 64
\end{array}$$

Construction 2.4.4. *Let $A_8 : Z_2^4 \rightarrow S_6$ is defined as follows and B_8 is 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, a mapping $g_8 \in \mathcal{F}(8,2,1)$ is constructed.

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

Like B_7, any mapping that satisfies the position property could be B_8. A_8 holds another property that if the Hamming distance of two binary vectors is 3, then the 4th entries of the images must be different, i.e. 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 D for

g_8 is 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 2.4.5. *Let A_9 is the same as A_8 and $B_9 : Z_2^5 \rightarrow S_7$ is 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, a mapping $g_9 \in \mathcal{F}(9, 2, 1)$ is constructed.

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

There is no other limitations about B_9 as long as the position property is satisfied. The distance expansion matrix D for g_9 is 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

Next we show how to construct a mapping $g_{n+4} \in \mathcal{F}(n+4, 2, 1)$ inductively from a mapping $g_n \in \mathcal{F}(n, 2, 1)$. To do that, we reapply the auxiliary mapping $B_6 : Z_2^4 \to S_6$. Actually we can use any mapping that holds the position property. We denote it as E. We give the algorithm below and then proof the correctness.

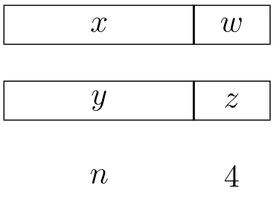
Algorithm 2.4.6. *Input:* $(x_1, \cdots, x_n, \cdots, x_{n+4}) \in Z_2^{n+4}$
Output: $(\pi_1, \cdots, \pi_{n+5}) = g_{n+4}(x_1, \cdots, x_{n+4})$

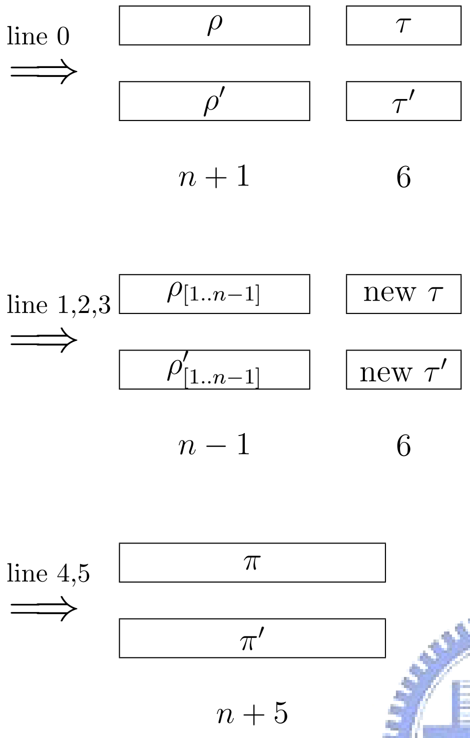
```
begin
0  $\rho = g_n(x_1, \cdots, x_n); \tau = E(x_1, x_2, x_3, x_4);$
1  $\tau_i = \tau_i + n - 1$, for $1 \leq i \leq 6$;
2  $\tau_{\tau^{-1}(n)} = \rho_n$;
3  $\tau_{\tau^{-1}(n+1)} = \rho_{n+1}$;
4  $(\pi_1, \cdots, \pi_{n-1}) = \rho_{[1..n-1]}$;
5  $(\pi_n, \cdots, \pi_{n+5}) = \tau_{[1..6]}$;
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 2.4.7. *$g_{n+4} \in \mathcal{F}(n+4, 2, 1)$, for $n \geq 6$.*

Proof. First note that after line 1, $\rho_i \in \{1, \cdots, n+1\}$, for $i = 1$ to $n+1$ and $\tau_i \in \{n, \cdots, n+5\}$, for $i = 1$ to 6. But at line 2 and 3, the value n in τ is replaced by ρ_n and the value $n+1$ in τ by ρ_{n+1}. The other values in τ range from $n+2$ to $n+5$. Thus the swap operation can be taken successfully.

Let (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 $g_n(x) = \rho = (\rho_1, \cdots, \rho_{n+1})$, $g_n(y) = \rho' = (\rho'_1, \cdots, \rho'_{n+1})$, $B_7(w) = \tau = (\tau_1, \cdots, \tau_6)$, and $B_7(z) = \tau' = (\tau'_1, \cdots, \tau'_6)$. And $g_{n+4}(x, w) = \pi = (\pi_1, \cdots, \pi_{n+5})$, $g_{n+4}(y, z) = \pi' = (\pi'_1, \cdots, \pi'_{n+5})$. We illustrate the transforms of these two strings in the following diagram.





Now we explain the effect of the operation at line 2. Since the values of ρ_n and ρ'_n are from $\{1,\cdots,n+1\}$, after the substitution, if $\tau^{-1}(n) = \tau'^{-1}(n)$ then ρ_n and ρ'_n are still in the same coordinate and the distance of this coordinate is preserved, else ρ_n and ρ'_n correspond to a value from $\{n+2,\cdots,n+5\}$ (note that $n+1$ is impossible, since value $n+1$ is in coordinate 3 or 4). Thus after substituting operation at line 2, the distance won't decrease. Same argument holds for the operation at line 3. Therefore, after line 5, we have $d_H(\pi_{[n..n+5]},\pi'_{[n..n+5]}) \geq d_H(\tau,\tau')$.

Next we consider the following cases:

- Case $[d_H(x,y)=0]$: We know that $d_H(w,z) \neq 0$, otherwise (x,w) and (y,z) are identical. Let $d_H(w,z) = t \leq 4$. Since $B_7 \in \mathcal{F}(2,4,2)$, we have $d_H(\tau,\tau') \geq t+2$. Therefore $d(\pi,\pi')) \geq t+2 = d_H((x,w),(y,z))+2$.

- Case $[0 < d_H(x,y) = s < n]$: It is clear that $d_H(\rho,\rho') \geq s+2$.

If $0 < d_H(w,z) = t$, then $d_H(\tau,\tau') \geq t+2$. Thus, by the above mentioned observation, we have $d_H(\pi,\pi') = d_H(\pi_{[1..n-1]},\pi'_{[1..n-1]}) + d_H(\pi_{[n..n+5]},\pi'_{[n..n+5]}) \geq s+(t+2) = d_H((x,w),(y,z))+2$. For $d_H(w,z) = 0$, it is easy to see $d_H(\pi,\pi') \geq s+2$.

- Case $[d_H(x,y) = n]$: In this case, it is clear that $d_H(\rho,\rho') = n+1$. Let $d_H(w,z) = t$. Again by earlier observation, we know that $d_H(\pi_{[1..n-1]},\pi'_{[1..n-1]}) = d_H(\rho_{[1..n-1]},\rho'_{[1..n-1]}) = n-1$ and $d_H(\pi_{[n..n+5]},\pi'_{[n..n+5]}) \geq d_H(\tau,\tau') \geq t+2$ (even when $t=0$). Thus $d_H(\pi,\pi') \geq n+t+1$. We argue that this lower bound is indeed at least $n+t+2$, except when $t=4$. There are two subcases on the value of $d_H(w,z)$, which is denoted as t.

  1. Subcase $[t=4]$: Then $d_H(\tau,\tau') = 6$. It is easy to see $d_H(\pi,\pi') = n+5$.

  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 such that $\tau_i = \tau'_i$. Note that $x_1 \neq y_1$ and $x_2 \neq y_2$, since $d_H(x,y) = n$. If $\tau_5 = \tau'_5$ or $\tau_6 = \tau'_6$, then after the swap steps in line 6 and line 7, $d_H(\pi_{[n..n+5]},\pi'_{[n..n+5]}) \geq t+3$. So $d_H(\pi,\pi') \geq n+t+2$. If $\tau_1 = \tau'_1$ or $\tau_2 = \tau'_2$, the value 1's in τ and in τ' must be in the same coordinate. Besides $\rho_n \neq \rho'_n$, we can get Thus we have $d_H(\pi_{[n..n+5]},\pi'_{[n..n+5]}) \geq t+3$. Same argument holds for $\tau_3 = \tau'_3$ or $\tau_4 = \tau'_4$.

This completes our proof on the correctness of construction. □

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

The approach for the construction of $\mathcal{F}(n,3,2)$ is alike that for the construction of $\mathcal{F}(n,2,1)$. We first give five basis cases: $h_6 \in \mathcal{F}(6,3,2)$, $h_7 \in \mathcal{F}(7,3,2)$, $h_8 \in \mathcal{F}(8,3,2)$, $g_9 \in \mathcal{F}(9,3,2)$, $h_{10} \in \mathcal{F}(10,3,2)$. Then, we give the induction method for constructing $h_{n+5} \in \mathcal{F}(n+5,3,2)$ from a map $h_n \in \mathcal{F}(n,3,2)$. Thus a series of $\mathcal{F}(n,3,2)$ is built, for $n \geq 6$.

Like previous section, we will use two auxiliary mappings C_i,D_i for each basis case. Both C_i and D_i have the position property for $\{1,2\}$ and $\{\pi^{-1}(1)|\pi \in C_i(\cdot), D_i(\cdot)\} = \{1,2\}$, $\{\pi^{-1}(2)|\pi \in C_i(\cdot), D_i(\cdot)\} = \{3,4\}$. There may be some further limitations about certain $C_i's$ and $D_i's$. We will state the special properties they hold as they first appear.

Construction 2.5.1. *Let $C_6 : Z_2^3 \rightarrow S_6$ define as follows and D_6 is the same as C_6:*

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	$(6,1,5,2,3,4)$

By the following algorithm, a mapping $h_6 \in \mathcal{F}(6,3,2)$ is constructed.

```
Input: $(x_1, x_2, \cdots, x_6) \in Z_2^6$
Output: $(\pi_1, \pi_2, \cdots, \pi_8) = h_6(x_1, \cdots, x_6)$
begin
    $\rho = C_6(x_1, x_2, x_3); \tau = D_6(x_4, x_5, x_6);$
    $\rho_i = \rho_i - 2$, for $1 \leq i \leq 6$;
    $\tau_i = \tau_i + 2$, for $1 \leq i \leq 6$;
    $\rho_{\rho^{-1}(-1)} = \tau_5$;
    $\rho_{\rho^{-1}(0)} = \tau_6$;
    $\tau_{\tau^{-1}(3)} = \rho_5$;
    $\tau_{\tau^{-1}(4)} = \rho_6$;
    $(\pi_1, \cdots, \pi_4) = \rho_{[1..4]}$;
    $(\pi_5, \cdots, \pi_8) = \tau_{[1..4]}$;
end
```

There is no other limitations about C_6. And C_6 and D_6 do not need to be the same, as long as the position property is satisfied. The distance expansion matrix D 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

Construction 2.5.2. *Let C_7 be the same as C_6 and $D_7 : Z_2^4 \to S_7$ 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, a mapping $h_7 \in \mathcal{F}(7,3,2)$ is constructed.

```
Input: $(x_1, x_2, \cdots, x_7) \in Z_2^7$
Output: $(\pi_1, \pi_2, \cdots, \pi_9) = h_7(x_1, \cdots, x_7)$
begin
```

```
$\rho = C_7(x_1,x_2,x_3); \tau = D_7(x_4,\cdots,x_7);$
$\rho_i = \rho_i - 2$, for $1 \le i \le 6$;
$\tau_i = \tau_i + 2$, for $1 \le i \le 7$;
$\rho_{\rho^{-1}(-1)} = \tau_6$;
$\rho_{\rho^{-1}(0)} = \tau_7$;
$\tau_{\tau^{-1}(3)} = \rho_5$;
$\tau_{\tau^{-1}(4)} = \rho_6$;
$(\pi_1,\cdots,\pi_4) = \rho_{[1..4]}$;
$(\pi_5,\cdots,\pi_9) = \tau_{[1..5]}$;
end
```

Besides the position property, D_7 holds another property that if the Hamming distance of two binary vectors is 3, then the 5th 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 D for h_7 is 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 2.5.3. *Let both C_8 and D_8 be the same as D_7. By the following algorithm, a mapping $h_8 \in \mathcal{F}(8,3,2)$ is constructed.*

```
Input: $(x_1,x_2,\cdots,x_8) \in Z_2^8$
Output: $(\pi_1,\pi_2,\cdots,\pi_{10}) = h_8(x_1,\cdots,x_8)$
begin
$\rho = C_8(x_1,\cdots,x_4); \tau = D_8(x_5,\cdots,x_8);$
$\rho_i = \rho_i - 2$, for $1 \le i \le 7$;
$\tau_i = \tau_i + 3$, for $1 \le i \le 7$;
```

```
$\rho_{\rho^{-1}(-1)} = \tau_6$;
$\rho_{\rho^{-1}(0)} = \tau_7$;
$\tau_{\tau^{-1}(4)} = \rho_6$;
$\tau_{\tau^{-1}(5)} = \rho_7$;
$(\pi_1, \cdots, \pi_5) = \rho_{[1..5]}$;
$(\pi_6, \cdots, \pi_{10}) = \tau_{[1..5]}$;
end
```

The distance expansion matrix D 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 2.5.4. *Let C_9 be the same as C_8 and $D_9 : Z_2^5 \rightarrow S_8$ 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, a mapping $h_9 \in \mathcal{F}(9,3,2)$ is constructed.

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

In fact, C_9 could be simpler than C_8, any mapping that holds the position property could be C_9. D_9 holds another property that if the Hamming distance of two binary vectors is 4, then the 6th 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 D 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 2.5.5. *Let both C_{10} and D_{10} be the same as D_9. By the following algorithm, a mapping $h_{10} \in \mathcal{F}(10, 3, 2)$ is constructed.*

```
Input: $(x_1, x_2, \cdots, x_{10}) \in Z_2^{10}$
Output: $(\pi_1, \pi_2, \cdots, \pi_{12}) = h_{10}(x_1, \cdots, x_{10})$
begin
    $\rho = C_{10}(x_1, \cdots, x_5); \tau = D_{10}(x_6, \cdots, x_{10})$;
    $\rho_i = \rho_i - 2$, for $1 \leq i \leq 8$;
    $\tau_i = \tau_i + 4$, for $1 \leq i \leq 8$;
    $\rho_{\rho^{-1}(-1)} = \tau_7$;
    $\rho_{\rho^{-1}(0)} = \tau_8$;
    $\tau_{\tau^{-1}(4)} = \rho_7$;
    $\tau_{\tau^{-1}(6)} = \rho_8$;
```

```
$(\pi_1, \cdots, \pi_6) = \rho_{[1..6]}$;
$(\pi_7, \cdots, \pi_{12}) = \tau_{[1..6]}$;
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 D 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

Next we show how to construct a mapping $h_{n+5} \in \mathcal{F}(n+5,3,2)$ inductively from a mapping $h_n \in \mathcal{F}(n,3,2)$. In fact, suppose we have a mapping $\hat{E} \in \mathcal{F}(\hat{n},3,2)$ with the position property for $\{1,2,3\}$, then the concept of the induction algorithm for $\mathcal{F}(n,2,1)$ can also be applied to $\mathcal{F}(n,3,2)$. And the algorithm only need to be modified subtly, so as the proof. Unfortunately we haven't find such $\hat{E}$, for $\hat{n} = 5$,or 6. Instead, we use a mapping $E : Z_2^5 \rightarrow S_8$ which are closed to the desired one and defined as follows.

x	$D_{11}(x)$	x	$D_{11}(x)$
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	$(6,1,8,2,7,3,4,5)$

Let's explain how the mapping E is produced. We first find a mapping $e \in \mathcal{F}(4,3,3)$ with the position property for $\{1,2\}$, and do some switchings such that value 1 appears in coordinate 2 and 3, value 2 appears in coordinate 4 and 5. Second add 1 to each entry of all the permutations in the image. Third define $E : Z_2^5 \to S_5$ as: for all $w \in Z_2^4$, $E(0w) = (1, \pi_1, \pi_2, \cdots, \pi_7))$ and $E(1w) = (\pi_1, 1, \pi_2, \cdots, \pi_7))$, where $\pi = e(w)$. It is easy to check that for all distinct strings $x, y \in Z_2^5$, if $d_H(x,y) = d$, then $d_H(E(x), E(y)) \geq d+3$ except the case when x, y only differ at the first bit. In this case, $d_H(x,y) = 1$, but $d_H(E(x), E(y)) = 2$. We give the algorithm below and then proof the correctness.

Algorithm 2.5.6. *Input:* $(x_1, \cdots, x_n, \cdots, x_{n+5}) \in Z_2^{n+5}$

Output: $(\pi_1, \cdots, \pi_{n+7}) = h_{n+5}(x_1, \cdots, x_{n+5})$

begin

```
0 $\rho = h_n(x_1, \cdots, x_n); \tau = E(x_{n+1}, \cdots, x_{n+5});$
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 $(\pi_1, \cdots, \pi_{n-1}) = \rho_{[1..n-1]}$;
6 $(\pi_n, \cdots, \pi_{n+7}) = \tau_{[1..8]}$;
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 2.5.7. *$h_{n+5} \in \mathcal{F}(n+5, 3, 2)$, for $n \geq 6$.*

Suppose ignore the exception, the proof for the rest cases is quite similar to the proof in previous section. Thus we simply omit this part. Next we proof the correctness for the exception case.

Proof. Let x, $y \in Z_2^n$ and $w \in Z_2^4$. Let $h_n(x) = \rho = (\rho_1, \cdots, \rho_{n+2})$, $h_n(y) = \rho' = (\rho'_1, \cdots, \rho'_{n+2})$, and $E(0w) = (1, \sigma_1, \sigma_2, \cdots, \sigma_7)$. By the definition of E, we know that $E(1w) = (\sigma_1, 1, \sigma_2, \cdots, \sigma_7)$. And let $h_{n+5}(x, 0w) = \pi = (\pi_1, \cdots, \pi_{n+7})$, $h_{n+5}(y, 1w) = \pi' = (\pi'_1, \cdots, \pi'_{n+7})$.

Since the the string $(y, 1w)$ will trigger the swap at line 8 but the string $(x, 0w)$ won't, then after line 8, $\pi_{n+7} \neq \pi_{n+7}$. Besides value 2 and value 3 in $E(0w)$ must be in the same position as value 2 and value 3 in $E(1w)$ respectively. Therefore after line 8, $d_H(\pi_{[n..n+7]}, \pi'_{[n..n+7]}) \geq 3+$ $d_H((\rho_{n+1}, \rho_{n+2}), (\rho'_{n+1}, \rho'_{n+2}))$. Furthermore we get $d_H(\pi, \pi') =$ $d_H(\pi_{1..n-1}, \pi'_{1..n-1}) + d_H(\pi_{[n..n+7]}, \pi'_{[n..n+7]}) \geq 3 + d_H(\rho, \rho') - 1 = 2 + d_H(\rho, \rho')$.

Next we consider the following cases:

- Case $[d_H(x, y) = 0]$: Since the string $(y, 1w)$ will trigger the swap at line 8 but the string $(x, 0w)$ won't, then after line 8, $\pi_{n+2} \neq \pi_{n+2}$. So $d_H(\pi, \pi')) \geq 4 = d_H((x, 0w), (y, 1w)) + 3$.

- Case $[0 < d_H(x,y) = s \leq n]$: It is clear that $d_H(\rho,\rho') \geq s+2$. Therefore $d_H(\pi,\pi') =\geq 2 + d_H(\rho,\rho') = 2 + (s+2) = d_H(x,y) + 1$.

□

2.6 A Framework for Constructing $\mathcal{F}(n, k+1, k)$

The approach for constructing a series of $\mathcal{F}(n,2,1)$ is similar to that for constructing a series of $\mathcal{F}(n,3,2)$. We wonder if there is a generalized approach for constructing $\mathcal{F}(n,k+1,k)$. In this section, we give a possible framework for constructing $\mathcal{F}(n,k+1,k)$.

Let's talk about how to construct the basis cases first. We use the construction of $\mathcal{F}(n,2,1)$ as example to illustrate our idea. When we dealt with $\mathcal{F}(n,2,1)$, we tried to find out the basis cases by computer search. Unfortunately, after a time consuming search, there were still no outcome. Rather than just by searching, we found a method to construct the basis cases directly. We noticed that any mappings in $\mathcal{F}(n,2,2)$ can be easily found by computer, even with some restrictions. We wondered if we could use these mappings in $\mathcal{F}(n,2,2)$ to construct a mapping $g \in \mathcal{F}(n,2,1)$. But there was a big problem. Suppose we simply concatenate a mapping $A \in \mathcal{F}(\hat{m},2,2)$ and a mapping $B \in \mathcal{F}(\hat{n},2,2)$. The resulted mapping is from $Z_2^{\hat{m}+\hat{n}}$ to $S_{\hat{m}+\hat{n}+4}$. Undoubtedly it doesn't belong to $\mathcal{F}(n,2,1)$. There should be a way to shrink the length of the image. Therefore we adopted the substitution technique again. In order to let the substitution step work well, the mappings we use should hold the position property. Let $A \in \mathcal{F}(\hat{m},2,2)$ with the position property for $\{1\}$ and $B \in \mathcal{F}(\hat{n},2,2)$ with the position property for $\{1,2\}$. Let $A(x) = \rho$, $x \in Z_2^{\hat{m}}$ and $B(y) = \tau$, $y \in Z_2^{\hat{n}}$. Replace the value 1 in ρ with an entry of τ in the coordinate where the value 1 and 2 won't appear, and replace the value 1 in τ with an entry of ρ in the coordinate where the value 1 won't appear, so does the value 2 in τ. Then the resulted image length is $\hat{m} + \hat{n} + 1$. The resulted mapping has a chance to be in $\mathcal{F}(n, 2, 1)$. However there may still be some problems. Let's look at the construction of g_6 as the example. Note that we have made a further restriction about B_6. Suppose B_6 is simply a mapping in $\mathcal{F}(4, 2, 2)$ with only the position property, and $B_6(0000) = (1, 3, 2, 4, 5, 6)$, $B_6(1110) = (4, 1, 6, 2, 5, 3)$. Through our algorithm, we will find that $g_6(000000) =$ (1,7,3,4,2,5,6) and $g_6(111110)$ = (4,1,5,2,7,3,6). $d_H(g_6(000000), g_6(111110)) = 6 \ngeq d_H(000000, 111110) + 2$. Thus such g_6 doesn't belong to $\mathcal{F}(6, 2, 1)$. That's why we made the restriction about B_6.

We state the framework for constructing the basis cases for $\mathcal{F}(n, k+1, k)$ as follows: find 2 mappings in $\mathcal{F}(n, k+1, k+1)$ with the appropriate position property, such that after substitution, the resulted image size is legal, and make some extra restrictions about these two mappings if necessary, i.e. the construction still need to be dealt case by case. In fact, the framework could be generalized for even $\mathcal{F}(n, d, k)$. We only need to make more restrictions about these 2 mappings. We have tried to find out the basis cases for $\mathcal{F}(n, 2, 0)$ based on this framework, but unsuccessful. Many restrictions make these 2 mappings hard to be found quickly by computer, even they don't exist. But we believe the framework is helpful for constructing the basis cases for $\mathcal{F}(n, k+1, k)$. We can find these 2 auxiliary mappings quickly even with additional restrictions.

Next we discuss the induction method for $\mathcal{F}(n, k+1, k)$. As a matter of fact, Algorithm 2.4.6 can be generalized to any k, as long as the the proper extension mapping E is found. We modify Algorithm 2.4.6 for $\mathcal{F}(n, k+1, k)$ as follows. Assume $E \in \mathcal{F}(l, k+1, k+1)$ with the position property for $\{1, 2, \cdots, k+1\}$ and $b_n \in \mathcal{F}(n, k+1, k)$.

Algorithm 2.6.1. *Input:* $(x_1, \cdots, x_n, \cdots, x_{n+l}) \in Z_2^{n+l}$
Output: $(\pi_1, \cdots, \pi_{n+l+k}) = b_{n+l}(x_1, \cdots, x_{n+l})$

```
begin
    $\rho = b_n(x_1, \cdots, x_n); \tau = E(x_1, \cdots, x_l)$;
    $\tau_i = \tau_i + n - 1$, for $1 \le i \le l + k + 1$;
    $\tau_{\tau^{-1}(i)} = \rho_i$, for $n \le i \le n + k$;
    $(\pi_1, \cdots, \pi_{n-1}) = \rho_{[1..n-1]}$;
    $(\pi_n, \cdots, \pi_{n+l+k}) = \tau_{[1..l+k+1]}$;
    If $x_i = 1$ then swap $(\pi_i, \pi_{(n-1)+2(k+1)+i})$, for $1 \le i \le l - (k + 1)$;
end
```

Theorem 2.6.2. *Suppose $E \in \mathcal{F}(l, k + 1, k + 1)$ with the position property for $\{1, 2, \cdots, k + 1\}$ and $b_n \in \mathcal{F}(n, k + 1, k)$. Then Algorithm 2.6.1 induces an $(n + l, k + 1, k)$-mapping.*

The proof is similar to that for Algorithm 2.4.6. We omit the proof here. Unfortunately as we have seen in the case $\mathcal{F}(n, 3, 2)$, the extension mapping E may not be found easily. We just point out a possible induction method and a framework of constructing $\mathcal{F}(n, k + 1, k)$.

Chapter 3

Application to Permutation Arrays

As shown in [3] and [2], we know that distance-increasing(distance-preserving) mappings are quite helpful for constructing permutation arrays. In this chapter we give a general view of constructions of permutation arrays via distance-increasing mappings. An (n, r)-permutation code is a permutation code of length n and minimum distance r. Let $P(n, r)$ denote the maximal size among all (n, r)-permutation codes, and $A(n, r)$ the maximal size among all (n, r)-binary codes. Recall that in the preliminaries, we define $n_{d,k,q}$ to be the smallest integer such that for $n \geq n_{d,k,q}$, $\mathcal{F}(n, d, k, q)$ is not empty, and $m_{d,k,q} = n_{d,k,q} + k$, i.e. the smallest image length. Let $n_{d,k}$ denote $n_{d,k,2}$ and $m_{d,k}$ denote $m_{d,k,2}$ for convenience. We have the following bound.

Theorem 3.0.3. *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 $(n-k, r-d)$-code. 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 3.0.3 tells us that if we have an efficient (n, d, k)-mapping and a binary $(n-k, r-d)$-code, then we get an efficient (n, r)-permutation code.

The current existing $P(n,r)$-bound, $P(n,r) \geq A(n,r-1)$, for $n \geq 4$ was shown in [3] and [2]. If $\mathcal{F}(n,2,1)$ is applied, the $P(n,r)$-bound becomes $P(n,r) \geq A(n-1,r-2)$, for $n \geq 7$. It is well-known that $A(n-1,r-2) = A(n,r-1)$ if r is odd. But when r is even, $A(n-1,r-2) > A(n,r-1)$. Thus we improve the bound when r is even. If $\mathcal{F}(n,3,2)$ is applied, the $P(n,r)$-bound becomes $P(n,r) \geq A(n-2,r-3)$, for $n \geq 8$. It is known that $A(n-2,r-3) > A(n,r-1)$ no matter r is even or odd. Thus we do improve the $P(n,r)$-bound.

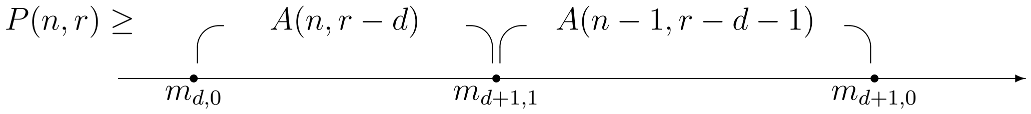
The difference between the classes of distance-increasing mappings defined by previous researches and ours is that we consider one more parameter, k. When $k = 0$, the classes of mappings we defined are the same as those before. We restate Theorem 3.0.3 when $k = 0$: For $n \geq m_{d,0}$ and $d+1 \leq r \leq n$, $P(n,r) \geq A(n,r-d)$. Here we will show that with parameter k involved, there are several cases with better bounds.

For $m_{d,0} \leq n < m_{d+1,0}$, we have $P(n,r) \geq A(n,r-d)$. We want to show that there are possible improvements for n in the gap. First it is known that $A(n,t) \leq A(n-1,t-1)$. Therefore assume $m_{d+1,1} < m_{d+1,0}$, for $m_{d+1,1} \leq n < m_{d+1,0}$, $P(n,r) \geq A(n-1,r-d-1) \geq A(n,r-d)$, where the improvement occurs. The assumption $m_{d+1,1} < m_{d+1,0}$ really makes sense. Since suppose there is a mapping $f \in \mathcal{F}(\hat{n},d+1,0)$, ignore half of the binary vectors in the domain, we get a $f' \in \mathcal{F}(\hat{n}-1,d+1,1)$. Thus $m_{d+1,1} \leq m_{d+1,0}$ and very likely $m_{d+1,1} < n_{d+1,0}$. From the above observation, we have the following lemma:

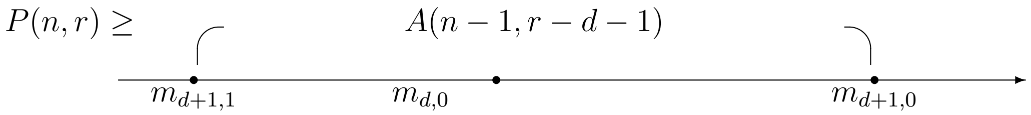
Lemma 3.0.4. $m_{d,k+1} \leq m_{d,k}$.

We plot the possible bound for $P(n,r)$ in the following diagram.

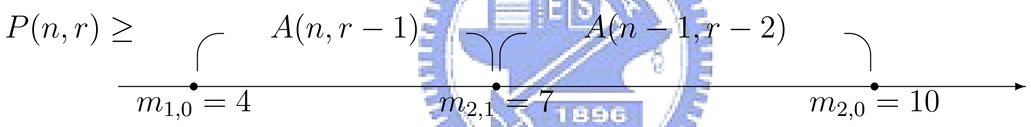
If $m_{d,0} < m_{d+1,1}$:



If $m_{d,0} \geq m_{d+1,1}$:

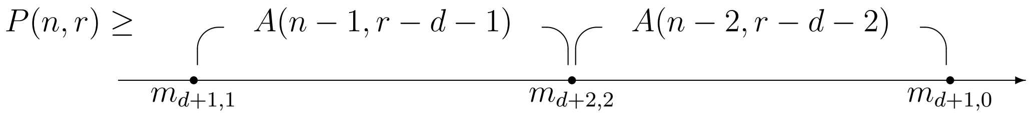


Let's apply some real examples. We know $m_{1,0} = 4$. Although we don't know what $m_{2,0}$ exact is, according to our past experience, we believe $m_{2,0} \geq 8$. Assume $m_{2,0} = 10$. The bound-diagram of this interval is plotted as follows:

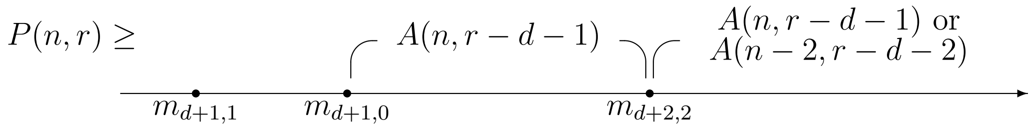


Second it is known that $A(n,t) < A(n-2, t-2)$. Assume $m_{d+2,2} < m_{d+1,0}$, for $m_{d+2,2} \leq n < m_{d+1,0}$, $P(n,r) \geq A(n-2, r-d-2) > A(n, r-d)$. Unfortunately the assumption might not hold for all time. However suppose $m_{d+2,2} \geq m_{d+1,0}$, Compare $A(n, r-d-1)$ and $A(n-2, r-d-2)$ from the optimal binary codes table[1], we can find that there are still cases such that $A(n, r-d-1) < A(n-2, r-d-2)$ depending on n and r. Usually it is believed that $m_{d,k} \leq m_{d+1,k+1}$. So assume $m_{d+1,1} < m_{d+2,2}$, the most likely bound-diagram is as follows:

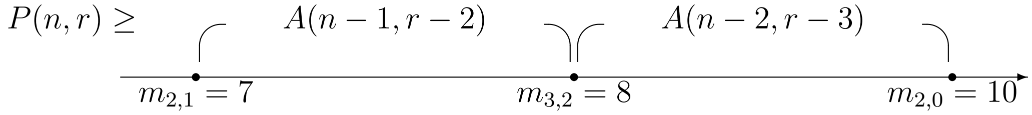
If $m_{d+2,2} < m_{d+1,0}$:



If $m_{d+2,2} \geq m_{d+1,0}$:



Again we use the real examples to illustrate the situation. Assume $m_{2,0} = 10$.



As we have said, when $n \geq m_{2,0}$, there are still cases such that $A(n, r-2) < A(n-2, r-3)$ depending on n and r. For example, $A(12, 5) = 32$, but $A(10, 4) = 40$; $A(14, 7) = 16$, but $A(12, 6) = 24$. Thus when $n \geq m_{2,0}$, $P(n,r) \geq A(n, r-2)$ or $A(n-2, r-3)$ depending on n and r. Recall that we have made a conjecture in the last paragraph, now we restate it formally in the following.

Conjecture 3.0.5. $m_{d,k} \leq m_{d+1,k+1}$.

We have discussed the comparison between the bounds induced by $\mathcal{F}(n, d, 0)$, $\mathcal{F}(n, d+1, 1)$,$\mathcal{F}(n, d+1, 0)$, and $\mathcal{F}(n, d+2, 2)$ in the interval $[m_{d,0}, m_{d+1,0}]$. In fact, we should consider all the $P(n,r)$-bounds by $\mathcal{F}(n, d_i, k_i)$'s as long as m_{d_i,k_i}'s $\leq m_{d+1,0}$. More generally when given certain $\hat{n}$ and $\hat{r}$, we should compare all the $P(\hat{n}, \hat{r})$-bound induced by $\mathcal{F}(n, d_i, k_i)$'s as long as $m_{d_i,k_i} \leq \hat{n}$ and $A(\hat{n} - k_i, \hat{r} - d_i)$ is meaningful. Now we make a formal statement about what the possible best $P(\hat{n}, \hat{r})$-bound would be when given certain $\hat{n}$ and $\hat{r}$.

Theorem 3.0.6. *Given $\hat{n}$ and $\hat{r}$, $P(\hat{n}, \hat{r}) \geq \max_i\{A(\hat{n} - k_i, \hat{r} - d_i)\}$, where (d_i, k_i) satisfy $k_i \leq \hat{n} - 1$, $d_i \leq \hat{r} - 1$, and $m_{d_i,k_i} \leq \hat{n}$.*

Chapter 4

Conclusion and Future Works

4.1 Conclusion

We have shown how to construct a series of $\mathcal{F}(n,d,k)$ for $(n,k) = (1,0)$, $(2,1)$, and $(3,2)$ in Chapter 2. We reduce the number of basis cases needed for constructing a series of $\mathcal{F}(n,1,0)$. For $\mathcal{F}(n,2,1)$, $P(n,r) \geq A(n-1,r-2) > A(n,r-1)$, $n \geq 7$ when r is even. For $\mathcal{F}(n,3,2)$, $P(n,r) \geq A(n-2,r-3) > A(n,r-1)$, $n \neq 8$. We improve the current existing bound $P(n,r) \geq A(n,r-1)$. We also propose a framework of constructing a series of $\mathcal{F}(n,k+1,k)$. We believe that this framework will be better than an exhaustive search. The idea, that with parameter k involved, we can find better bounds in the gap between $m_{d,0}$ and $m_{d+1,0}$, is presented. Thus we have shown that different settings of (d,k) may make the $P(n,r)$-bound better. Therefore it is worth discussing a wider class, $\mathcal{F}(n,d,k)$. We might lose many improvements if we only adopt the bound contributed by $\mathcal{F}(n,d,0)$.

4.2 Future Works

The basis cases for $\mathcal{F}(n,d,0)$ when $d > 1$ are still unknown so far. We have tried to find them out by computer exhausting search, but unfortunately there is still not much done. Generally speaking, it is difficult to find the basis cases for $\mathcal{F}(n, d, k)$ when $d > k + 1$. Thus we are curious whether an efficient searching algorithm or even a straight-forward construction method exists.

Second we have given several induction methods for different (n, d, k)-mappings. Although these methods look alike, they are still different. Is there a general induction method for construct a series of $\mathcal{F}(n, d, k)$?

The discussion about possible best $P(n, r)$-bound in the gap mainly depend on the value $m_{d,k}$. Is there a way to estimate $m_{d,k}$ or can we give this value a lower bound or an upper bound?

Bibliography

[1] E. Agrell, A. Vardy, and K. Zeger, A table of upper bounds for binary codes. In *IEEE Transactions on Information Theory*, vol.47: 3004 - 3006, Nov. 2001.

[2] J.C. Chang, Distance-increasing mappings from binary vectors from binary vectors to permuations. In *IEEE Transactions on Information Theory*, vol.51: 359-363, Jan. 2005.

[3] J.C. Chang, R.J. Chen, T. Kløve and S.C. Tsai, Distance-preserving mappings from binary vectors to permutations. In *IEEE Transactions on Information Theory*, vol.49: 1054-1059, Apr. 2003.

[4] C.J. Colbourn, T. Kløve, and A.C.H Ling, Permutation arrays for powerline communication and mutually orthogonal latin squares. In *IEEE Transactions on Information Theory*, vol.50: 1289-1291, June 2004.

[5] C. Ding, F.W. Fu, T. Kløve, and V.K. Wei, Construction of permutation arrays. In *IEEE Transactions on Information Theory*, vol.48: 977-980, Apr. 2002.

[6] M. Deza and S.A. Vanstone, Bounds on permutation arrays. In *J.Statist. Planning and Inference*, vol.2: 197-209, 1978.

[7] P. Frankel and M. Deza, On the maximum number of permutations with given maximal and minimal distance. In *J. Comb. Theory, Ser A*, vol.22: 352-360, 1977.

[8] F.W. Fu and T. Kløve, Two constructions of permutation arrays. In *IEEE Transactions on Information Theory*, vol.50: 881-883, May 2004.

[9] H.C. Ferreira and A.J.H. Vinck, Inference cancellation with permutation trellis arrays. In *Proc. IEEE Vehicular Technology Conf.*, 2401-2407, 2000.

[10] K. Lee, New distance-preserving maps of odd length. In *IEEE Transactions on Information Theory*, vol.50: 2539-2543, Oct. 2004.

[11] H. Tarnanen, Upper bounds on permutation codes via linear programming. In *Europ. J. Combin.*, vol.20: 101-114, 1999.

[12] A.J.H. Vinck, Coded modulation for powerline communications. In *AEÜ Int. J. Elecron. Commun.*, vol.54: 45-49, 2000.

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

[14] A.J.H. Vinck, J. Häring, and T. Wadayama, Coded M-FSK for power-line communications. In *Proc. IEEE Int. Symp. Information Theory*, Sorrento, Italy, 137, Jun. 2000.

[15] T. Wadayama and A.J.H. Vinck, Amultilevel construction of permutation codes. In *IEICE Trans. Fundamentals Electron., Commun. Comp. Sci.*, vol.84: 2518-2522, 2001.