

Decoding Frequency Permutation Arrays under Infinite norm

Min-Zheng Shieh

Department of Computer Science
National Chiao Tung University
1001 University Road, Hsinchu, Taiwan
Email: mzhsieh@csie.nctu.edu.tw

Shi-Chun Tsai

Department of Computer Science
National Chiao Tung University
1001 University Road, Hsinchu, Taiwan
Email: sctsai@cs.nctu.edu.tw

Abstract—A frequency permutation array (FPA) of length $n = m\lambda$ and distance d is a set of permutations on a multiset over m symbols, where each symbol appears exactly λ times and the distance between any two elements in the array is at least d . FPA generalizes the notion of permutation array. In this paper, under the distance metric l_∞ -norm, we first prove lower and upper bounds on the size of FPA. Then we give a construction of FPA with efficient encoding and decoding capabilities. Moreover, we show our design is locally decodable, i.e., we can decode a message bit by reading at most $\lambda + 1$ symbols, which has an interesting application for private information retrieval.

I. INTRODUCTION

Let n, m and λ be positive integers with $n = m\lambda$, and S_n^λ be the set of all permutations on the multiset $\{\underbrace{1, \dots, 1}_\lambda, \dots, \underbrace{m, \dots, m}_\lambda\}$. A frequency permutation array (FPA) is a subset of S_n^λ for some positive integers m, λ and $n = m\lambda$. A (λ, n, d) -FPA is a subset of S_n^λ and the distance between any pair of distinct permutations is at least d under any metric, such as Hamming distance, l_∞ -norm, etc. Permutation array (PA) is simply a special case of FPA by choosing $\lambda = 1$. With a fixed length n , FPA has a smaller set of symbols than PA. Thus, codes with FPA have a better information rate than those with PA. A widely adopted approach to building PAs under Hamming distance, see for example [2], is using distance preserving mappings or distance increasing mappings from Z_2^k to S_n^1 . Most of those encoding schemes are efficient but it is not clear how to decode efficiently. Lin et al. [8] proposed a couple of novel constructions with efficient encoding and decoding algorithms for PAs under l_∞ -norm. FPA was proposed by Huczynska and Mullen [4] as a generalization of PA. They gave several constructions of FPA under Hamming distance and bounds for the maximum array size. In this paper, we extend the ideas in [8] to constructing FPA under l_∞ -norm. We prove lower and upper bounds of FPA. Then we show the efficient encoding and decoding algorithms. Besides, we show that our FPAs are locally decodable codes under l_∞ -norm.

Recently, researchers have found that PAs have applications in areas such as power line communication (e.g. [9],

[12], [13] and [14]), multi-level flash memories (see [5] and [6]). Similar to the application of PAs on power line communication, we can encode a message as a frequency permutation from S_n^λ and associate each symbol $i \in \{1, \dots, m\}$ with a frequency f_i . Then the message is transmitted as a series of corresponding frequencies. For example, to send a message encoded as $(1, 2, 2, 1, 3, 3)$, we can transmit the frequency sequence $(f_1, f_2, f_2, f_1, f_3, f_3)$ one by one.

For flash memory application, different from the approach by Jiang et al. [5], [6], we can use FPA to provide multi-level flash memory with error correcting capability. For example, suppose a multi-level flash memory, where each cell has m states, which can be changed by injecting or removing charge into or from it. Over injecting or charge leakage will alter the state as well. We can use the charge ranks of n cells to represent a permutation from S_n^λ , i.e., the cells with the lowest λ charge levels represent symbol 1, and so on. With our efficient encoding and decoding algorithms, a (λ, n, d) -FPA can be used in flash memory system to represent information and correct errors caused by charge level fluctuation.

A locally decodable code has an extremely efficient decoding for any message bit by reading at most a fixed number of symbols from the received word. Suppose that a FPA is applied to a multi-level flash memory where the length of a codeword is nearly a block of cells (about 10^5)[1]. This feature allows us to retrieve the desire message bits from a multi-level flash without accessing the whole block. With the locally decodable property, we can raise the robustness of the code without loss of efficiency. On the other hand, locally decodable codes have been under study for years, see [10] for a survey and [15], [3] for recent progress. They are related to a cryptographic protocol called *private information retrieval* (PIR for short). We show our construction of FPA can also be used in cryptographic application.

Notations: Let m and λ be positive integers and let $n = m\lambda$ throughout the paper unless stated otherwise. We use $[n]$ to represent the set $\{1, \dots, n\}$. S_n^λ denotes the permutations over the multiset $\{\underbrace{1, \dots, 1}_\lambda, \dots, \underbrace{m, \dots, m}_\lambda\}$. For two vectors \mathbf{x} and \mathbf{y} of the same dimension, let $l_\infty(\mathbf{x}, \mathbf{y}) =$

$\max_i |\mathbf{x}_i - \mathbf{y}_i|$. We say two permutations \mathbf{x} and \mathbf{y} are d -close to each other under metric $\delta(\cdot, \cdot)$ if $\delta(\mathbf{x}, \mathbf{y}) \leq d$. The identity permutation \mathbf{I}_n^λ in S_n^λ is $(1, \dots, 1, \dots, m, \dots, m)$.

II. LOWER AND UPPER BOUNDS

Let $F_\infty(\lambda, n, d)$ be the cardinality of the maximum (λ, n, d) -FPA and $V_\infty(\lambda, n, d)$ be the number of elements in S_n^λ being d -close to the identity \mathbf{I}_n^λ under ℓ_∞ -norm. In this section, we give a Gilbert type lower bound and a sphere packing upper bound of $F_\infty(\lambda, n, d)$ by bounding $V_\infty(\lambda, n, d)$.

First, we show that any d -radius ball in S_n^λ under ℓ_∞ -norm has the same cardinality.

Claim 1. For any $\mathbf{x} = (x_1, \dots, x_n) \in S_n^\lambda$, there are exactly $V_\infty(\lambda, n, d)$ \mathbf{y} 's in S_n^λ such that $l_\infty(\mathbf{x}, \mathbf{y}) \leq d$.

Proof: Since every $i \in [m]$ appears exactly λ times in \mathbf{x} , there exists a permutation $\pi \in S_n^\lambda$ such that $\mathbf{x} = \pi \circ \mathbf{I}_n^\lambda$. As a consequence, we have that $l_\infty(\mathbf{I}_n^\lambda, \mathbf{z}) = l_\infty(\mathbf{x}, \pi \circ \mathbf{z})$ for any $\mathbf{z} \in S_n^\lambda$. Let $Z = \{\mathbf{z} : \mathbf{z} \in S_n^\lambda, l_\infty(\mathbf{I}_n^\lambda, \mathbf{z}) \leq d\}$, $Y = \{\pi \circ \mathbf{z} : \mathbf{z} \in Z\}$ and $\bar{Y} = S_n^\lambda - Y$. For any $\mathbf{y} \in Y$, we have $l_\infty(\mathbf{x}, \mathbf{y}) = l_\infty(\mathbf{I}_n^\lambda, \pi^{-1} \circ \mathbf{y}) \leq d$, since $\pi^{-1} \circ \mathbf{y} \in Z$. While for $\mathbf{y}' \in \bar{Y}$, $l_\infty(\mathbf{x}, \mathbf{y}') = l_\infty(\mathbf{I}_n^\lambda, \pi^{-1} \circ \mathbf{y}') > d$. Therefore, only $|Y| = |Z| = V_\infty(\lambda, n, d)$ permutations in S_n^λ are d -close to \mathbf{x} . ■

Theorem 1.

$$\frac{|S_n^\lambda|}{V_\infty(\lambda, n, d-1)} \leq F_\infty(\lambda, n, d) \leq \frac{|S_n^\lambda|}{V_\infty(\lambda, n, \lfloor \frac{d-1}{2} \rfloor)}.$$

Proof: To prove the lower bound, we use the following algorithm to generate a (λ, n, d) -FPA with size $\geq \frac{|S_n^\lambda|}{V_\infty(\lambda, n, d-1)}$.

- 1) $C \leftarrow \emptyset$, $D \leftarrow S_n^\lambda$.
- 2) Add an arbitrary $\mathbf{x} \in D$ to C , then remove all permutations that is $(d-1)$ -close to \mathbf{x} from D .
- 3) If $D \neq \emptyset$ then repeat step 2, otherwise output C .

D has initially $|S_n^\lambda|$ elements and each iteration of step 2 removes at most $V_\infty(\lambda, n, d-1)$, so we conclude $|C| \geq \frac{|S_n^\lambda|}{V_\infty(\lambda, n, d-1)}$.

Now we turn to the upper bound. Consider a (λ, n, d) -FPA C^* with the maximum cardinality. Any two $\lfloor \frac{d-1}{2} \rfloor$ -radius balls centered at distinct permutations in C^* do not have any common permutation, since the minimum distance is d . In other words, the $\lfloor \frac{d-1}{2} \rfloor$ -radius balls centered at permutations in C^* are all disjoint. We have $|C^*| \leq \frac{|S_n^\lambda|}{V_\infty(\lambda, n, \lfloor \frac{d-1}{2} \rfloor)}$. ■

It is clear that $|S_n^\lambda| = \frac{n!}{(\lambda!)^{n/\lambda}}$. It is already known that $V_\infty(1, n, d)$ equals to the permanent of some special matrix [8]. In this paper, we generalize previous analysis to give asymptotic bounds for Theorem 1. The permanent of an $n \times n$ matrix $A = (a_{i,j})$ is defined as

$$\text{per}A = \sum_{\pi \in S_n} \prod_{i=1}^n a_{i, \pi_i}.$$

Define a symmetric $n \times n$ matrix $A^{(\lambda, n, d)} = (a_{i,j}^{(\lambda, n, d)})$, where $a_{i,j}^{(\lambda, n, d)} = 1$, if $|\lceil \frac{i}{\lambda} \rceil - \lceil \frac{j}{\lambda} \rceil| \leq d$; else $a_{i,j}^{(\lambda, n, d)} = 0$. Note that a permutation (x_1, \dots, x_n) is d -close to \mathbf{I}_n^λ and only if $a_{i, x_i}^{(\lambda, n, d)} = 1$ for every $i \in [n]$. Now we consider $A^{(\lambda, \lambda m, d)}$. Since the λ copies of a symbol are considered identical while computing the distance and the entries indexed from $(\ell\lambda - \lambda + 1)$ to $\ell\lambda$ of $\mathbf{I}_{\lambda m}^\lambda$ represent the same symbol for every $\ell \in [m]$. It implies that row $(\ell\lambda - \lambda + 1)$ through row $\ell\lambda$ of $A^{(\lambda, \lambda m, d)}$ are identical and so are columns indexed from $(\ell\lambda - \lambda + 1)$ to $\ell\lambda$ for every $\ell \in [m]$. Thus, we have $A^{(\lambda, \lambda m, d)} = A^{(1, m, d)} \otimes \mathbf{1}_\lambda$ where \otimes is the operator of tensor product and $\mathbf{1}_\lambda$ is a $\lambda \times \lambda$ matrix with all entries equal to 1. For example, take $\lambda = 2$, $m = 5$ and $d = 2$:

$$A^{(1, 5, 2)} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \mathbf{1}_2 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$$A^{(2, 10, 2)} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Let $r_i^{(1, m, d)}$ be the row sum of $A^{(1, m, d)}$'s i -th row. We have:

$$r_i^{(1, m, d)} = \begin{cases} d+i & \text{if } i \leq d, \\ 2d+1 & \text{if } d < i \leq m-d, \\ m-i+1+d & \text{if } i > m-d. \end{cases}$$

Then for $i \in [m]$ and $j \in [\lambda]$, the row sum of the $(i\lambda - \lambda + j)$ -th row of $A^{(\lambda, \lambda m, d)}$ is $\lambda r_i^{(1, m, d)}$, due to $A^{(\lambda, \lambda m, d)} = A^{(1, m, d)} \otimes \mathbf{1}_\lambda$. We first calculate $V_\infty(\lambda, n, d)$ by using $\text{per}A^{(\lambda, n, d)}$.

Lemma 1.

$$V_\infty(\lambda, n, d) = \frac{\text{per}A^{(\lambda, n, d)}}{(\lambda!)^{n/\lambda}}.$$

Proof:

$$\begin{aligned} & \text{per}A^{(\lambda, n, d)} \\ &= |\{\mathbf{x} \in S_n^\lambda : \forall i, a_{i, x_i}^{(\lambda, n, d)} = 1\}| \\ &= |\{\mathbf{x} \in S_n^\lambda : \max_i |\lceil \frac{i}{\lambda} \rceil - \lceil \frac{x_i}{\lambda} \rceil| \leq d\}| \\ &= (\lambda!)^{n/\lambda} |\{\mathbf{y} \in S_n^\lambda : \max_i |\lceil \frac{i}{\lambda} \rceil - y_i| \leq d\}| \\ &= (\lambda!)^{n/\lambda} |\{\mathbf{y} \in S_n^\lambda : l_\infty(\mathbf{I}_n^\lambda, \mathbf{y}) \leq d\}| \\ &= (\lambda!)^{n/\lambda} V_\infty(\lambda, n, d) \end{aligned}$$

The first equality holds since $A^{(\lambda, n, d)}$ is a $(0, 1)$ -matrix and by the definition of permanent. We can convert $\mathbf{x} \in S_n^\lambda$ into $\mathbf{y} \in S_n^\lambda$ by setting $y_i = \lceil \frac{x_i}{\lambda} \rceil$, and there are exactly $(\lambda!)^{n/\lambda}$

\mathbf{x} 's in S_n^1 converted to the same \mathbf{y} . Thus, we know the third equality holds. Therefore, the lemma holds by moving $(\lambda!)^{n/\lambda}$ to the left-hand side of the equation. ■

We still need to estimate $\text{per}A^{(\lambda,n,d)}$ in order to get asymptotic bounds. Kløve [7] reports some bounds and methods to approximate $\text{per}A^{(1,n,d)}$. We extend his analysis for $\text{per}A^{(\lambda,n,d)}$.

Lemma 2. $\text{per}A^{(\lambda,n,d)} \leq [(2d\lambda + \lambda)!]^{\frac{n}{2d\lambda + \lambda}}$.

Proof: It is known (Theorem 11.5 in [11]) that for $(0, 1)$ -matrix A , $\text{per}A \leq \prod_{i=1}^n (r_i!)^{\frac{1}{r_i}}$ where r_i is the sum of the i -th row. Since the sum of any row of $A^{(\lambda,n,d)}$ is at most $2d\lambda + \lambda$, we have $\text{per}A \leq \prod_{i=1}^n [(2d\lambda + \lambda)!]^{\frac{1}{2d\lambda + \lambda}} = [(2d\lambda + \lambda)!]^{\frac{n}{2d\lambda + \lambda}}$. ■

We give $\text{per}A^{(\lambda,n,d)}$ a lower bound by using the van der Waerden permanent theorem (see p.104 in [11]): *the permanent of an $n \times n$ doubly stochastic matrix A (i.e., A has nonnegative entries, and every row sum and column sum of A is 1.) is no less than $\frac{n!}{n^n}$.* Unfortunately, $A^{(\lambda,n,d)}$ is not a doubly stochastic matrix, since the row sums and columns sums range from $d\lambda + \lambda$ to $2d\lambda + \lambda$. We estimate the lower bound via a matrix derived from $A^{(\lambda,n,d)}$ as follows.

Lemma 3. $\text{per}A^{(\lambda,n,d)} \geq \frac{(2d\lambda + \lambda)^n}{2^{2d\lambda}} \cdot \frac{n!}{n^n}$.

Proof: Let $\tilde{A} = \frac{1}{2d\lambda + \lambda} A^{(\lambda,n,d)}$, which has the sum of any row or column bounded by 1, but is not a doubly stochastic matrix. Observe that every row sum of \tilde{A} is 1 except the first $d\lambda$ and last $d\lambda$ rows. For $i \in [d]$ and $j \in [\lambda]$, both row $(i\lambda - \lambda + j)$ and row $(n - i\lambda + j)$ sum to $\frac{d+i}{2d+1}$. Now we construct an $n \times n$ matrix B from \tilde{A} with each row sum equal to 1 as follows:

For $i \in [d]$ and $j \in [\lambda]$, add $\frac{1}{2d\lambda + \lambda}$ to

- 1) The first $(d - i + 1)\lambda$ entries of row $(i\lambda - \lambda + j)$.
- 2) The last $(d - i + 1)\lambda$ entries of row $(n - i\lambda + j)$.

The row sums of the first $d\lambda$ and last $d\lambda$ rows of B are now $\frac{(d-i+1)\lambda}{2d\lambda + \lambda} + \frac{d+i}{2d+1} = 1$.

We turn to check the column sums of B . Since \tilde{A} is symmetric and by the definition of B , we know B is symmetric as well. Thus we have that B is doubly stochastic and $\text{per}B \geq \frac{n!}{n^n}$.

Now we turn to bound $\text{per}A^{(\lambda,n,d)}$. Observe that the entries of the first $d\lambda$ and last $d\lambda$ rows of B are at most $\frac{2}{2d\lambda + \lambda}$ times of the corresponding entries of $A^{(\lambda,n,d)}$, and the other rows are exactly $\frac{1}{2d\lambda + \lambda}$ times of the corresponding rows of $A^{(\lambda,n,d)}$. We have $\text{per}A^{(\lambda,n,d)} \geq \frac{(2d\lambda + \lambda)^n}{2^{2d\lambda}} \text{per}B \geq \frac{(2d\lambda + \lambda)^n}{2^{2d\lambda}} \frac{n!}{n^n}$. ■

With Lemma 2 and Lemma 3, we have the asymptotic bounds as follows.

Theorem 2.

$$\frac{n!}{[(2d\lambda - \lambda)!]^{\frac{n}{2d\lambda - \lambda}}} \leq F_\infty(\lambda, n, d) \leq \frac{2^{2\lambda \cdot \lfloor \frac{d-1}{2} \rfloor} n^n}{(2\lambda \cdot \lfloor \frac{d-1}{2} \rfloor + \lambda)^n}.$$

III. ENCODING AND DECODING

Our construction idea is based on the previous work[8] by Lin, et al. We generalize their algorithm for constructing FPAs. Furthermore, we give the first locally decoding algorithm for FPAs under l_∞ -norm.

A. Encoding algorithm

We give an encoding algorithm $E_{n,k}^\lambda$ (see Figure 1) which convert k -bit message into a permutation in S_n^λ where $n \geq k + \lambda$.

Algorithm $E_{n,k}^\lambda$

Input: $(m_1, \dots, m_k) \in Z_2^k$

Output: $(x_1, \dots, x_n) \in S_n^\lambda$

$max \leftarrow n; min \leftarrow 1;$

for $i \leftarrow 1$ **to** k **do**

if $m_i = 1$

then $\{x_i \leftarrow \lceil \frac{max}{\lambda} \rceil; max \leftarrow max - 1;\}$

else $\{x_i \leftarrow \lceil \frac{min}{\lambda} \rceil; min \leftarrow min + 1;\}$

for $i \leftarrow k + 1$ **to** n **do**

$x_i \leftarrow \lceil \frac{min}{\lambda} \rceil; min \leftarrow min + 1;$

Output (x_1, \dots, x_n) .

Fig. 1: $E_{n,k}^\lambda$ encodes messages in Z_2^k with S_n^λ .

The encoding algorithm $E_{n,k}^\lambda$ maps binary vectors from Z_2^k to S_n^λ and it is a distance preserving mapping. It is clear that $E_{n,k}^\lambda$ runs in $O(n)$ time while encoding any k -bit message. Next we investigate the properties of the code obtained by $E_{n,k}^\lambda$. Let $C_{n,k}^\lambda$ be the image of $E_{n,k}^\lambda$.

Theorem 3. $C_{n,k}^\lambda$ is a $(\lambda, n, \lfloor \frac{n-k}{\lambda} \rfloor)$ -FPA with cardinality 2^k .

Proof: Consider two messages $\mathbf{p} = (p_1, \dots, p_k)$ and $\mathbf{q} = (q_1, \dots, q_k) \in Z_2^k$. Let $\mathbf{x}^{\mathbf{p}}$ and $\mathbf{x}^{\mathbf{q}}$ be the outputs of $E_{n,k}^\lambda$, respectively. Let r be the smallest index such that $p_r \neq q_r$. Without loss of generality, we assume $p_r = 1, q_r = 0$ and there are exactly z zeroes among p_1, \dots, p_{r-1} . Consequently, $x_r^{\mathbf{p}}$ is set to $\lceil \frac{max}{\lambda} \rceil = \lceil \frac{n-r+1+z}{\lambda} \rceil$ and $x_r^{\mathbf{q}}$ is set to $\lceil \frac{min}{\lambda} \rceil = \lceil \frac{1+z}{\lambda} \rceil$ by $E_{n,k}^\lambda$. The distance between $\mathbf{x}^{\mathbf{p}}$ and $\mathbf{x}^{\mathbf{q}}$ is:

$$\begin{aligned} & \left| \frac{n-r+1+z}{\lambda} \right| - \left| \frac{1+z}{\lambda} \right| \\ & > \frac{n-r+1+z}{\lambda} - \frac{1+z}{\lambda} - 1 \\ & = \frac{n-r}{\lambda} - 1 \\ & \geq \frac{n-k}{\lambda} - 1, \text{ since } r \leq k. \end{aligned}$$

The first inequality holds by the fact of ceiling function: $a \leq \lceil a \rceil < a + 1$, for any real number a . Note that the distance has integer value only here. If $\frac{n-k}{\lambda}$ is integer then the distance is at least $\lfloor \frac{n-k}{\lambda} \rfloor$; else it is at least $\lceil \frac{n-k}{\lambda} \rceil - 1$, which is $\lfloor \frac{n-k}{\lambda} \rfloor$ exactly, i.e., the distance between any two

codewords in $C_{n,k}^\lambda$ is at least $\lfloor \frac{n-k}{\lambda} \rfloor$. Since every message is encoded into a distinct codeword, we have $C_{n,k}^\lambda = 2^k$. ■

Since $C_{n,k}^\lambda$ is a $(\lambda, n, \lfloor \frac{n-k}{\lambda} \rfloor)$ -FPA, we let $d = \lfloor \frac{n-k}{\lambda} \rfloor$ for convenience.

B. Unique decoding algorithm

Unique decoding algorithms for classic error correcting codes are usually much more complicated than their encoding algorithms. While, our proposed decoding algorithm $U_{n,k}^\lambda$ (see Figure 2) remains simple.

Algorithm $U_{n,k}^\lambda$

Input: $(x_1, \dots, x_n) \in S_n^\lambda$

Output: $(m_1, \dots, m_k) \in Z_2^k$

$max \leftarrow n; min \leftarrow 1;$

for $i \leftarrow 1$ **to** k **do**

if $|x_i - \lceil \frac{max}{\lambda} \rceil| < |x_i - \lceil \frac{min}{\lambda} \rceil|$
then $\{m_i \leftarrow 1; max \leftarrow max - 1;\}$
else $\{m_i \leftarrow 0; min \leftarrow min + 1;\}$

Output (m_1, \dots, m_k) .

Fig. 2: $U_{n,k}^\lambda$ decodes words in S_n^λ to messages in Z_2^k .

The running time of $U_{n,k}^\lambda$ is clearly $O(k)$, even faster than the encoding algorithm. We show its correctness as follows.

Theorem 4. Given a permutation $\mathbf{x} = (x_1, \dots, x_n)$ which is $\frac{d-1}{2}$ -close to $E_{n,k}^\lambda(\mathbf{m})$ for some $\mathbf{m} \in Z_2^k$, algorithm $U_{n,k}^\lambda$ outputs \mathbf{m} correctly.

Proof: By contradiction, assume $U_{n,k}^\lambda$ outputs $\hat{\mathbf{m}} = (\hat{m}_1, \dots, \hat{m}_k) \neq \mathbf{m}$. Let $E_{n,k}^\lambda(\mathbf{m}) = (y_1, \dots, y_n)$, r be the smallest index such that $m_r \neq \hat{m}_r$ and z be the number of zeroes among m_1, \dots, m_{r-1} . At the beginning of the r -th iteration, $max = n - r + 1 + z$ and $min = 1 + z$ because for every $i < r$, $m_i = \hat{m}_i$. Without loss of generality, assume $1 = m_r \neq \hat{m}_r = 0$. Note that y_r is set to $\lceil \frac{max}{\lambda} \rceil = \lceil \frac{n-r+1+z}{\lambda} \rceil$ by $E_{n,k}^\lambda$. While \hat{m}_r is decoded to 0 by $U_{n,k}^\lambda$, we have $|x_r - \lceil \frac{max}{\lambda} \rceil| \geq |x_r - \lceil \frac{min}{\lambda} \rceil|$. Thus,

$$\begin{aligned} l_\infty(\mathbf{x}, E_{n,k}^\lambda(\mathbf{m})) &\geq |x_r - y_r| = |x_r - \lceil \frac{max}{\lambda} \rceil| \\ &\geq \frac{1}{2} (|x_r - \lceil \frac{max}{\lambda} \rceil| + |x_r - \lceil \frac{min}{\lambda} \rceil|) \\ &\geq \frac{1}{2} (\lceil \frac{max}{\lambda} \rceil - \lceil \frac{min}{\lambda} \rceil) \\ &= \frac{1}{2} (\lceil \frac{n-r+1+z}{\lambda} \rceil - \lceil \frac{1+z}{\lambda} \rceil) \geq \frac{d}{2}. \end{aligned}$$

The last inequality is true, since we know $\lceil \frac{n-r+1+z}{\lambda} \rceil - \lceil \frac{1+z}{\lambda} \rceil \geq \lfloor \frac{n-k}{\lambda} \rfloor = d$ from the proof of Theorem 3. This contradicts that \mathbf{x} is $\frac{d-1}{2}$ -close to $E_{n,k}^\lambda(\mathbf{m})$. ■

C. Locally decoding algorithm

Next we show a locally decoding algorithm $L_{n,k}^\lambda$, see Figure 3, which is a probabilistic algorithm. We discuss its efficiency and error probability in this subsection. We prove that it reads at most $\lambda + 1$ entries of the received word in Lemma 4, hence its running time is $O(\lambda)$. It has a chance to output wrongly, but we show that the error probability is small in Theorem 5. Furthermore, $L_{n,k}^\lambda$ always outputs

correct message bit when it was given a codeword as input, see Corollary 1.

Algorithm $L_{n,k}^\lambda$

Input: $i \in [n], (x_1, \dots, x_n) \in S_n^\lambda$

Output: m_i , the i -th message bit

$J \leftarrow \{i + 1, \dots, n\};$

do

Uniformly and randomly pick $j \in J$;

if $x_i > x_j$ **then** output 1;

if $x_i < x_j$ **then** output 0;

$J \leftarrow J - \{j\};$

loop;

Fig. 3: $L_{n,k}^\lambda$ decodes one bit by reading at most $\lambda + 1$ symbols.

Lemma 4. Given a permutation $\mathbf{x} = (x_1, \dots, x_n) \in S_{n,k}^\lambda$ and an index $i \in [k]$, $L_{n,k}^\lambda$ terminates within λ iterations.

Proof: By contradiction, assume $L_{n,k}^\lambda$ does not output before the end of the λ -th iteration. For $\ell \leq \lambda$, let j_ℓ be the index picked in the ℓ -th iteration. For every $\ell \leq \lambda$, we have $x_i = x_{j_\ell}$, otherwise $L_{n,k}^\lambda$ outputs at the ℓ -th iteration. Therefore, there are at least $\lambda + 1$ entries of \mathbf{x} equal to x_i . It implies $\mathbf{x} \notin S_{n,k}^\lambda$, a contradiction. There is some $x_{j_\ell} \neq x_i$, and $L_{n,k}^\lambda$ outputs in the ℓ -th iteration. ■

Theorem 5. Given a permutation $\mathbf{x} = (x_1, \dots, x_n)$ δ -close to a codeword $E_{n,k}^\lambda(\mathbf{m}) = (y_1, \dots, y_n) \in S_{n,k}^\lambda$ for some \mathbf{m} and an index $i \in [k]$, $L_{n,k}^\lambda$ outputs m_i with probability at least $1 - \frac{2\delta+1}{d}$ at its first iteration.

Proof: Without loss of generality, we assume $m_i = 0$, $y_i = t$ and let u be the maximum number among y_{i+1}, \dots, y_n , i.e., at the start of the i -th iteration $min = t$ and $max = u$ while encoding. Assume there are γ numbers equal to t among y_1, \dots, y_{i-1} , and there are γ' numbers equal to u among y_{i+1}, \dots, y_n . According to the encoding algorithm, we have

$$\{y_{i+1}, \dots, y_n\} = \{\overbrace{t, \dots, t}^{\lambda-\gamma-1}, \overbrace{t+1, \dots, t+1}^{\lambda}, \dots, \overbrace{u, \dots, u}^{\gamma'}\}$$

Since $l_\infty(\mathbf{x}, E_{n,k}^\lambda(\mathbf{m})) \leq \delta$, we have $|x_j - y_j| \leq \delta$ and $|x_i - y_i| \leq \delta$. The probability that $L_{n,k}^\lambda$ does not output m_i at the first iteration is:

$$\begin{aligned} \Pr[x_i \geq x_j] &\leq \Pr[y_i + \delta \geq x_j] \\ &\leq \Pr[y_i + \delta \geq y_j - \delta] \\ &= \Pr[y_i + 2\delta \geq y_j]. \end{aligned}$$

There are at most $2\delta\lambda + \lambda - \gamma - 1$ possible y_j 's less than or equal to $y_i + 2\delta$. Thus,

$$\Pr[x_i \geq x_j] \leq \frac{(2\delta + 1)\lambda - \gamma - 1}{n - i} \leq \frac{2\delta\lambda + \lambda}{d\lambda} = \frac{2\delta + 1}{d}.$$

Therefore, the probability that $L_{n,k}^\lambda$ outputs m_i correctly at the first iteration is at least $1 - \frac{2\delta+1}{d}$. ■

Corollary 1. Given a codeword $\mathbf{x} = E_{n,k}^\lambda(\mathbf{m})$ for some \mathbf{m} and an index i , $L_{n,k}^\lambda$ outputs m_i correctly.

Proof: By Lemma 4, there exists $\ell \leq \lambda$ such that $L_{n,k}^\lambda$ terminates at the ℓ -th iteration. Let j be the index picked at the ℓ -th iteration, we have $x_j \neq x_i$, where $j > i$. Note that \mathbf{x} is a codeword: $x_i < x_j$ implies $m_i = 0$ and $x_i > x_j$ implies $m_i = 1$. Hence, $L_{n,k}^\lambda$ outputs m_i correctly. ■

A private information retrieval system (PIR) consists of q servers. All servers know a codeword $\mathbf{x} = (x_1, \dots, x_n)$ representing a message $\mathbf{m} = (m_1, \dots, m_k)$, and a user wants to know one bit m_i of \mathbf{m} via query a symbol from each server. We say a PIR has *retrievability* r if the user can obtain the message bit with probability r . Let $\mathcal{D}(s, i)$ be the distribution of entry queried from server i when the user tries to retrieve m_i . A PIR has *privacy* p if $\max_{i,j \in [k], s \in [q]} \Delta(\mathcal{D}(s, i), \mathcal{D}(s, j)) \leq p$, where $\Delta(\cdot, \cdot)$ is the statistical distance. A (q, r, p) -PIR is a q -server PIR with retrievability r and privacy p . A (q, r, p) -PIR has perfect retrievability if $r = 1$ and perfect privacy if $p = 0$.

With our FPA $C_{n,k}^\lambda$, we construct a $(\lambda + 1, 1, r)$ -PIR with perfect retrievability and privacy r . The scheme is simple:

- For a message \mathbf{m} , we put $\mathbf{x} = E_{n,k}^\lambda(\mathbf{m})$ on all $\lambda + 1$ servers.
- We retrieve m_i by $L_{n,k}^\lambda$ by querying entries from servers in a random order.

The perfect retrievability is guaranteed by Corollary 1. However, in order to retrieve m_i , x_i must be queried from some servers at certain positions $\ell > i$, and we have $r > 0$. We leave the improvement on the privacy r as our future work.

REFERENCES

- [1] P. Cappelletti, C. Golla, P. Olivo, and E. Zanoni, *Flash memories*. Kluwer Academic Publishers, 1999.
- [2] J. C. Chang, R. J. Chen, T. Kløve and S. C. Tsai, "Distance-preserving mappings from binary vectors to permutations," *IEEE Trans. Inform. Th.*, vol. IT-49, pp. 1054–1059, Apr. 2003.
- [3] Klim Efremenko, "3-Query Locally Decodable Codes of Subexponential Length," *Electronic Colloquium on Computational Complexity*, TR08-069, 2008.
- [4] Sophie Huczynska and Gary L. Mullen, "Frequency permutation arrays," *Journal of Combinatorial Designs*, vol. 14, pp. 463–478, 2006.
- [5] A. Jiang, R. Mateescu, M. Schwartz and J. Bruck, "Rank Modulation for Flash Memories," in *Proc. IEEE Internat. Symp. on Inform. Th.*, 2008, pp. 1731-1735.
- [6] A. Jiang, M. Schwartz and J. Bruck, "Error-Correcting Codes for Rank Modulation," in *Proc. IEEE Internat. Symp. on Inform. Th.*, 2008, pp. 1736-1740.
- [7] Torleiv Kløve, "Spheres of Permutations under the Innity Norm - Permutations with limited displacement," *Reports in Informatics, Dept. of Informatics, Univ. Bergen*, Report no. 376, 2008.
- [8] T. T. Lin, S. C. Tsai and W. G. Tzeng, "Efficient Encoding and Decoding with Permutation Arrays," in *Proc. IEEE Internat. Symp. on Inform. Th.*, 2008, pp. 211-214.
- [9] K. W. Shum, "Permutation coding and MFSK modulation for frequency selective channel," *IEEE Personal, Indoor and Mobile Radio Communications*, vol. 13, pp. 2063–2066, Sept. 2002.
- [10] Luca Trevisan, "Some Applications of Coding Theory in Computational Complexity," *Quaderni di Matematica*, vol. 13, pp. 347-424, 2004.
- [11] J.H. van Lint, R. M. Wilson, *A Course in Combinatorics.*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2001.

- [12] A. J. H. Vinck and J. Häring, "Coding and modulation for power-line communications," in *Proc. Internat. Symp. on Power Line Commun.*, Limerick, Ireland, April 2000.
- [13] A. J. H. Vinck, J. Häring, T. Wadayama, "Coded M-FSK for power line communications," in *Proc. IEEE Internat. Symp. on Inform. Th.*, 2000, p.137.
- [14] A. J. H. Vinck, "Coded modulation for powerline communications," *Proc. Int. J. Electron. Commun.*, vol. 54, pp. 45-49, 2000.
- [15] Sergey Yekhanin, "Towards 3-query locally decodable codes of subexponential length," *J. ACM*, vol. 55(1), pp. 1-16, 2008.