A Generalization of the Blackburn-Etzion Construction for Private Information Retrieval Array Codes

Yeow Meng Chee*, Han Mao Kiah†, Eitan Yaakobi‡, and Hui Zhang†

*Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore
†School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
‡Computer Science Department, Technion—Israel Institute of Technology, Haifa, Israel
Emails: pvocym@nus.edu.sg, hmkiah@ntu.edu.sg, yaakobi@cs.technion.ac.il, huizhang@ntu.edu.sg

***Abstract*—Private Information Retrieval (PIR) array codes were introduced by Fazeli et al. (2015) to reduce the storage overhead in designing PIR protocols. Blackburn and Etzion (2017) introduced the (virtual server) rate to quantify the storage overhead of the codes, and when $s > 2$ (here, $\frac{1}{s}$ is the proportion of the database storing in one server), they gave a general construction of PIR array codes with the highest rate known so far. In this paper, we generalize their construction and reduce the number of servers, while maintaining the rate. In order to give PIR array codes with significantly fewer servers, we also construct classes of codes with a smaller rate $\frac{s}{2s-1}$.**

I. Introduction

Chor et al. introduced *private information retrieval* (PIR) protocols to preserve the privacy of users when users retrieve information from a database stored on distributed servers [5]. Recently, Fazelli et al. [6], [7] borrowed coding techniques from distributed storage and showed that their proposed codes can be combined with known PIR protocols to reduce the storage overhead, while preserving privacy and maintaining low communication complexity [6], [7]. Specifically, suppose that the database is partitioned into p parts. A *PIR code* is a linear code where these p items are encoded into a set of m servers such that each item has k disjoint recovery sets.

PIR codes were studied in [1], [3], [4], [6], [7], [9]–[11] with the objective of minimizing the number of servers, and thus the storage overhead, given fixed values of p and k. Most of the work about PIR codes in [1], [6], [7], [9] was restricted to the case that each server stores only one symbol, which is a linear combination of the p items in the database. In [7], Fazeli et al. also introduced *PIR array codes* where each server may store t symbols, with $t > 1$. They then showed that it is possible to reduce the number of servers, and thus storage overhead, using PIR array codes.

In [3], [4], Blackburn and Etzion studied the *virtual server rate* of PIR array codes which is the ratio k/m. For any positive integers $t \geq 2$ and $d = p - t$, they provided an upper bound for the virtual server rate, that is, $\frac{k}{m} \leq \frac{(2d+1)t+d^2}{(t+d)(2d+1)}$. Their constructions of PIR array codes were considered in two cases, separated by a rational number $s = \frac{p}{t}$, where $\frac{1}{s}$ denotes the proportion of the database stored in one server. For $1 < s \leq 2$, they obtained optimal codes reaching the above bound. For $s > 2$, they also gave a general construction of PIR array codes [4, Construction 4] and proved that the codes constructed were asymptotically optimal. In other words, their virtual server rate tends to the upper bound $\frac{s+1}{2s}$ when $t \to \infty$.

Following the same model, PIR array codes were also studied by Zhang et al. with the objective of increasing the virtual server rate [11]. When $t > d^2 - d$ with $d = p - t$, Zhang et al. constructed PIR array codes with the smallest number of servers with optimal rate. When $s > 2$, they improved the upper bound of virtual server rate to $\frac{d^2+2t^2+3td+2t}{2(t+d)(d+t+1)}$. For the lower bound, they gave a construction of codes with fewer servers by slightly sacrificing the rate. They also proved that the codes constructed have rate less than that in [4, Construction 4], but larger than $\frac{ts+t-1}{2ts}$.

In this paper, we generalize the Blackburn and Etzion's construction on PIR array codes to reduce the number of servers, while keeping the rate. We also show that it is possible to reduce the number of servers of codes from [4, Construction 4] with the generalization. Especially, when $s = 3$, we obtain a class of PIR array codes with $\frac{8t^2+5t+1}{t}\binom{3t}{t-1}$ servers, which is significantly fewer than $\frac{8t^2+5t+1}{2t^2}\binom{2t}{t-1}\binom{3t}{t-1}$ and $\binom{3t}{t}\binom{2t-1}{t-1} + \binom{3t}{t-1}2^{2t}$ servers from [4], [11] respectively. Our codes achieve the same rate as [4, Construction 4], however the number of servers may still be too large for real applications, especially when t is extremely large. Thus, we also construct classes of codes with significantly fewer servers and a smaller rate $\frac{s}{2s-1}$ when s is small.

The paper is organized as follows. In Section II, we introduce some basic definitions and results as well as the motivation of our work. In Section III, we show the details of the construction of PIR array codes with $p = st$ for $s > 2$ and the comparisons. In Section IV, we show the constructions of PIR array codes with rate $\frac{s}{2s-1}$ when s is small, and conclude the paper in Section V.

II. Preliminaries

In this section, we define formally PIR array codes and some other useful definitions. The motivation for our work and overview on previous results will be given as well. For integers $a \leq b$, let $[a, b] = \{a, a+1, \ldots, b\}$ and for $b \geq 1$, let $[b] = [1, b]$.

*This work was done while Y. M. Chee was with the School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore.

A. Definitions

In a PIR array code, the database is partitioned into p parts, x_i, $i \in [p]$, where each part is encoded as a single item. All the items of the database are encoded into a two dimensional array, where each column represents the encoded symbols stored in one of the servers. The formal definition of these codes is given as follows.

Definition 1 *Given positive integers t, m, p and k, a $[t \times m, p]$ k-PIR array code is a $t \times m$ array, where each entry is a linear combination of the p items $\{x_1, \ldots, x_p\}$ over a certain field $\mathbb{F}$. Furthermore, for every $i \in [p]$, there exist k pairwise disjoint subsets of columns (called* recovering sets*) such that the i-th item x_i can be retrieved, by reading encoded symbols from each column, one copy from each subset.*

Assume each server stores at most t symbols. For each $j \in [m]$, let $y_{1,j}, y_{2,j}, \ldots, y_{t,j}$ be the encoded symbols stored in the j-th server, where each $y_{i,j}$, $i \in [t]$, $j \in [m]$ is a linear combination of $x_1, x_2, \ldots, x_p$. We usually write the code as the matrix $\mathbb{C} = (\boldsymbol{y}_1 | \boldsymbol{y}_2 | \cdots | \boldsymbol{y}_m)$, where $\boldsymbol{y}_j = (y_{1,j}, y_{2,j}, \ldots, y_{t,j})^T$, $j \in [m]$. Let $\mathbb{C} = (\boldsymbol{y}_1 | \boldsymbol{y}_2 | \cdots | \boldsymbol{y}_{m_1})$ and $\mathbb{C}' = (\boldsymbol{y}'_1 | \boldsymbol{y}'_2 | \cdots | \boldsymbol{y}'_{m_2})$ be two codes, we define the *concatenation* of $\mathbb{C}$ and $\mathbb{C}'$ to be the code $(\mathbb{C}|\mathbb{C}') = (\boldsymbol{y}_1 | \boldsymbol{y}_2 | \cdots | \boldsymbol{y}_{m_1} | \boldsymbol{y}'_1 | \boldsymbol{y}'_2 | \cdots | \boldsymbol{y}'_{m_2})$.

The entries of the array which are equal to one item of the database (that is, not a linear combination of more than one item) are called *singletons*. Otherwise, we call them *sums*. In particular, if a symbol is just a summation of i items, we call it a sum of *size* i.

B. Known Results and the Motivation

PIR array codes were first introduced in [6], [7] to reduce storage overhead of private information retrieval protocol. Plenty of bounds and constructions for one-dimensional PIR code, that is, $t = 1$, were well studied in [1], [6], [7], [9]. In this paper, we only focus on the case when $t > 1$, in which each server may store more than one symbol.

In [3], [4], PIR array codes were studied with the intention of maximizing the *(virtual server) rate*, which is defined to be the ratio $\frac{k}{m}$ to quantify the storage overhead. For any fixed positive integer t and rational number s, let $g(s,t)$ be the largest virtual server rate of any $[t \times m, p = st]$ k-PIR array code. Another important parameter of PIR array code is the smallest possible number of servers m, denoted as $M(s,t,k)$, given t, k and $s = \frac{p}{t}$. As pointed in [4], these parameters are closely connected. That is,

$$g(s = \frac{p}{t}, t) = \max_{k \geq 1} \left\{ \frac{k}{M(s,t,k)} \right\}$$
$$= \max \left\{ \frac{k}{m} : \text{there is a } [t \times m, p] \text{ } k\text{-PIR array code} \right\}.$$

For any integer $t \geq 2$, explicit bound of $g(s,t)$ and constructions of optimal PIR array codes reaching the bound when $1 < s \leq 2$ were given in [3], [4], [11]. Since we mainly focus on the case when $s > 2$ in this paper, we will not give all the details here. The interested readers may refer to the references therein. When $s > 2$, the authors also gave a general construction as below.

Construction 1 *[4, Construction 4] Let $s > 1$ and $t > 1$ be integers and $p = st$. Let $\xi_1, \xi_2, \ldots, \xi_s$ be positive integers such that*

$$\begin{aligned} \binom{p-t}{(r-1)t+1} \xi_r &= \binom{p-t}{rt} \xi_{r+1} \text{ for } r \in [2, s-1], \\ (s-1)\xi_1 &= \binom{p-t}{t} \xi_2. \end{aligned} \tag{1}$$

There are s types of servers: types $\mathrm{T}_1, \mathrm{T}_2, \ldots, \mathrm{T}_s$.

(i) Servers of type T_1 have only singletons. Each t-subset of $\{x_i : i \in [p]\}$ occurs ξ_1 times as the entries of a type T_1 server, so there are $\xi_1 \binom{p}{t}$ servers of type T_1.

(ii) Servers of type T_r with $r \in [2, s]$ store $t-1$ singletons together with a sum of size of $(r-1)t+1$. Each possible disjoint pair of a $(t-1)$-subset (which forms the singletons), and an $((r-1)t+1)$-subset (which forms a sum) of $\{x_i : i \in [p]\}$, occurs exactly ξ_r times. So there are $\xi_r \binom{p}{t-1} \binom{p-t+1}{(r-1)t+1}$ servers of type T_r.

The parameters of codes obtained from Construction 1 are formulated in the following theorem.

Theorem 2 *[4, Theorem 8] Let p, t, s and $\xi_1, \xi_2, \ldots, \xi_s$ be integers defined as in Construction 1. There exists a $[t \times m, p]$ k-PIR array code with $k = b + c$ and $k/m = (b+c)/(b+2c)$ where*

$$b = \xi_1 \binom{p-1}{t-1} + \sum_{r \in [2,s]} \xi_r \binom{p-1}{t-2} \binom{p-t+1}{(r-1)t+1} \text{ and}$$
$$c = \sum_{r \in [s-1]} \xi_{r+1} \binom{p-1}{t-1} \binom{p-t}{rt}.$$

The rate of the code is $(\beta + \gamma)/(\beta + 2\gamma)$ where

$$\beta = \xi_1 (p-t+1) + \sum_{r \in [2,s]} (t-1) \xi_r \binom{p-t+1}{(r-1)t+1} \text{ and}$$
$$\gamma = (p-t+1) \sum_{r \in [s-1]} \xi_{r+1} \binom{p-t}{rt}.$$

Construction 1 was also extended to the case when s is a rational number in [4, Construction 5]. In [11], a construction of PIR array codes with fewer servers was also provided by sacrificing slightly the rate. For the completeness of comparison, we also present their construction here.

Construction 2 *[11] Given any $t, s > 2$ and $p = st$.*

(i) Firstly, take all t-subsets of $\{x_i : i \in [p]\}$, each appearing $\binom{p-t-1}{t-1}$ times as the entries of a server, which contains only singletons.

(ii) A server with $t-1$ singletons and a sum of size j out of the remaining $p-t+1$ items is called a server of type j. Secondly, take all those servers of types $t+1$, $t+2$, $\ldots$, $p-t+1$, each appearing exactly once.

Theorem 3 *[11, Theorem 16] Given any $t, s > 2$ and $p = st$, there exists a $[t \times m, p]$ k-PIR array code with $k = \frac{p+t}{2p}\binom{p}{t}\binom{p-t-1}{t-1} + \frac{p+t-1}{2p}\binom{p}{t-1}\sum_{t+1\le j\le p-t+1}\binom{p-t+1}{j}$ and $m = \binom{p}{t}\binom{p-t-1}{t-1} + \binom{p}{t-1}\sum_{t+1\le j\le p-t+1}\binom{p-t+1}{j}$.*

In [11], the authors also showed that the rate of the code in Theorem 3 is less than that from Theorem 2, but larger than $\frac{ts+t-1}{2ts}$. Therefore, the codes from these two constructions are all asymptotically optimal when $t \to \infty$ because of the following upper bound.

Theorem 4 *[4, Theorem 3] For each rational number $s > 1$, $g(s) = \overline{\lim}_{t\to\infty} g(s,t) \le \frac{s+1}{2s}$. There is no t such that $g(s,t) = \frac{s+1}{2s}$.*

C. Main Results and Our Contributions

In this paper, we will give a generalization of Blackburn and Etzion's construction for PIR array codes when $s > 2$ to reduce the number of servers, while keeping the rate. We will only focus on the case when s is an integer. By giving examples for comparison, we show that it is possible to reduce the number of servers of codes from [4, Construction 4] by the generalization. Especially, when $s = 3$, we obtain a class of code with $\frac{8t^2+5t+1}{t}\binom{3t}{t-1}$ servers, which are much fewer than $\frac{8t^2+5t+1}{2t^2}\binom{2t}{t-1}\binom{3t}{t-1}$ and $\binom{3t}{t}\binom{2t-1}{t-1} + \binom{3t}{t-1}2^{2t}$ from [4], [11] respectively. In the end, to construct PIR array codes with significantly fewer servers, we will also give classes of codes with a smaller rate $\frac{s}{2s-1}$ when s is small.

III. A Generalization of Blackburn-Etzion Construction for PIR Array Codes when $s > 2$

In this section, we will give our main construction and show the improvement compared to Construction 1. We only focus on the case when $s = p/t > 2$ is an integer.

A. The Main Construction

We first let α_r for $r \in [s-1]$ be some $s-1$ positive integers that satisfy the following $s-2$ equations:

$$\alpha_{r-1}\binom{p}{(r-1)t+1} = \alpha_r\binom{p}{rt} \text{ for } r \in [2, s-1]. \quad (2)$$

For any given t and s, we will give a general construction of $[t \times m, p = st]$ k-PIR array codes in which m and k are functions of α_r, $r \in [s-1]$ satisfying (2). And we will provide explicit choices of the values of α_r's later.

Now, we define some multisets $\mathcal{T}_r$, $r \in [2, s]$ and $\mathcal{S}_r$, $r \in [s-1]$ whose elements are subsets of $[p]$. More precisely, we define that

- $\mathcal{T}_r$ for any $r \in [2, s]$, consists of all the $((r-1)t+1)$-subsets of $[p]$, each appearing α_{r-1} times, and
- $\mathcal{S}_r$ for any $r \in [s-1]$ consists of all the (rt)-subsets of $[p]$, each appearing α_r times.

Therefore, we see that $|\mathcal{T}_r| = \alpha_{r-1}\binom{p}{(r-1)t+1}$ for $r \in [2, s]$ and $|\mathcal{S}_r| = \alpha_r\binom{p}{rt}$ for $r \in [s-1]$, and according to (2), $|\mathcal{T}_r| = |\mathcal{S}_r|$ for all $r \in [2, s-1]$.

For each $r \in [2, s-1]$, we construct a bipartite graph $\mathcal{G}_r = (\mathcal{T}_r, \mathcal{S}_r, E)$ such that the disjoint sets of vertices are $\mathcal{T}_r$ and $\mathcal{S}_r$, and there is an edge $e \in E$ between $T \in \mathcal{T}_r$ and $S \in \mathcal{S}_r$ if and only if $T \subseteq S$. It is well known that if a bipartite graph is regular, it has a *perfect matching*, which is a set of edges containing all the vertices of the graph exactly once. For example, it can be derived from Hall's theorem in [8]. Since each bipartite graph $\mathcal{G}_r$ is regular, it has a perfect matching. We are now ready to present our main construction.

Construction 3 *For any integers $t \ge 2$, $s \ge 3$, assume $\mathcal{T}_r$, for $r \in [2, s]$ and α_r, $\mathcal{S}_r$, for $r \in [s-1]$ are defined as above. We construct the code $\mathbb{C} = (\mathbb{C}_1|\mathbb{C}_2|\cdots|\mathbb{C}_s)$, in which $\mathbb{C}_r$ for $r \in [s]$ are defined as below:*

(i) *Each column of $\mathbb{C}_1$ corresponds to an element $S \in \mathcal{S}_1$, and consists of t singletons $\{x_i : i \in S\}$.*

(ii) *For each $r \in [2, s-1]$, we construct the code $\mathbb{C}_r$ according to the perfect matching of the bipartite graph $\mathcal{G}_r$. For any edge (T, S) in the perfect matching of $\mathcal{G}_r$ with $T \in \mathcal{T}_r$ and $S \in \mathcal{S}_r$, we construct a column of $\mathbb{C}_r$ by storing the sum $\sum_{i\in T} x_i$ and $t-1$ singletons x_j, $j \in S \setminus T$.*

(iii) *Each column of $\mathbb{C}_s$ corresponds to an element $T \in \mathcal{T}_s$, and consists of the sum $\sum_{i\in T} x_i$ and $t-1$ singletons x_j, $j \in [p] \setminus T$.*

We can see that in the code $\mathbb{C}$, each server in $\mathbb{C}_r$, $r \in [s]$ stores t symbols, and $|\mathbb{C}_1| = \alpha_1\binom{p}{t}$, $|\mathbb{C}_r| = \alpha_{r-1}\binom{p}{(r-1)t+1}$ for $r \in [2, s]$. We present the parameters of codes obtained from Construction 3 in the following theorem.

Theorem 5 *The code $\mathbb{C}$ is a $[t \times m, p = st]$ k-PIR array code, where*

$$m = \alpha_1\binom{p}{t} + \sum_{r\in[2,s]} \alpha_{r-1}\binom{p}{(r-1)t+1},$$

$$k = m - \sum_{r\in[s-1]} \alpha_r\binom{p-1}{rt}.$$

Proof: The value of m is straightforward from the construction. Because of the symmetry of x_i, $i \in [p]$, we only need to check that x_1 has k recovering sets.

Each server that contains x_1 as a singleton forms a recovering set. Furthermore, for each $r \in [2, s]$, the servers in $\mathbb{C}_r$ that contain x_1 in a sum and the servers in $\mathbb{C}_{r-1}$ that do not contain x_1 also form recovering sets. More precisely, for any $((r-1)t)$-subset $\{x_{i_1}, x_{i_2}, \cdots, x_{i_{(r-1)t}}\}$ that does not contain x_1, there exist α_{r-1} servers in $\mathbb{C}_r$ that contain the sum $x_1 + x_{i_1} + \cdots + x_{i_{(r-1)t}}$, and there also exist α_{r-1} servers in $\mathbb{C}_{r-1}$ that contain all the $(r-1)t$ items $\{x_{i_1}, x_{i_2}, \cdots, x_{i_{(r-1)t}}\}$ either as a singleton or in a sum, but not x_1, and thus they form α_{r-1} recovering sets. This is to say, any server that either contains x_1 as a singleton or in a sum provides exactly one recovering set of x_1.

Note that, the number of servers in $\mathbb{C}_r$, $r \in [s-1]$ that neither contain x_1 in a sum nor as a singleton is $\alpha_r\binom{p-1}{rt}$. Therefore, we have $k = m - \sum_{r\in[s-1]} \alpha_r\binom{p-1}{rt}$. ∎

B. The Connection with Construction 1

In this part, we show that Construction 3 is indeed a generalization of Construction 1. We denote the PIR array code obtained from Construction 1 by $\mathbb{T}$, then we have:

Proposition 6 *For any positive integers $t \geq 2$, $s \geq 3$ and $\xi_1, \xi_2, \ldots, \xi_s$ satisfying* (1)*, the code $\mathbb{T}$ can be obtained from Construction 3 by taking*

$$\alpha_r = \xi_{r+1}\binom{p-rt-1}{t-1} \text{ for } r \in [s-1]. \tag{3}$$

Proof: At first, we can check that given any ξ_r's, $r \in [s]$ satisfying (1), the α_r's, $r \in [s-1]$ obtained from (3) also satisfy (2). Now, we show how to obtain $\mathbb{T}$ from Construction 3.

In Construction 3, for $r \in [2, s-1]$, we define the multiset

$$\begin{aligned}\mathcal{S}_r = \{&S_1 \cup S_2 : \ S_1 \text{ and } S_2 \text{ are all possible disjoint pairs of} \\ &\text{a } (t-1)\text{-subset and an } ((r-1)t+1)\text{-subset of } [p], \\ &\text{and each pair appears } \xi_r \text{ times}\}.\end{aligned}$$

Then for each $S \in \mathcal{S}_r$, $|S| = rt$, and each (rt)-subset of $[p]$ appears exactly $\binom{rt}{t-1}\xi_r$ times in $\mathcal{S}_r$. Take $\mathcal{T}_r$ to be the multiset containing all the $((r-1)t+1)$-subsets $S_2 \in S \in \mathcal{S}_r$. Finally, $\mathbb{C}_r$ in $\mathbb{C}$ corresponds to servers of type T_r in $\mathbb{T}$ for $r \in [s]$ respectively, and $\alpha_r = \binom{rt}{t-1}\xi_r = \xi_{r+1}\binom{p-rt-1}{t-1}$. ∎

It was pointed in [4] that for any fixed s, t, the rate of $\mathbb{T}$ does not depend on the choices of ξ_r's, since all possible solutions of $\{\xi_r : r \in [s]\}$ are all equal up to a scalar multiple. The codes from Construction 3 also have this property.

Corollary 7 *For any fixed s, t, the PIR array codes obtained from Construction 3 have the same rate as the codes from Construction 1, and therefore they are asymptotically optimal as $t \to \infty$.*

Proof: By (2), any two distinct choices of $\{\alpha_r : r \in [s-1]\}$ only differ by a rational factor. Thus, the rate of $\mathbb{C}$ in Theorem 5 does not depend on the choice of α_r's. Since there is a choice of α_r's that gives the code $\mathbb{T}$, the rate of $\mathbb{C}$ is always the same as $\mathbb{T}$ for given s and t. ∎

C. The Choices of α_r's

In this subsection, we present several choices of the values of α_r's, and show that there exist some parameters of codes that cannot be achieved by Construction 1.

Firstly, by (2) we have $\alpha_{r-1}\binom{p}{(r-1)t+1} = \alpha_r\binom{p}{rt}$ for $r \in [2, s-1]$, that is, $\frac{\alpha_{r-1}}{\alpha_r} = \frac{\binom{p}{rt}}{\binom{p}{(r-1)t+1}} = \frac{\binom{p-rt+t-1}{t-1}}{\binom{rt}{t-1}}$, and therefore, the first choice is to take

$$\begin{aligned}\alpha_1 &= \prod_{i\in[2,s-1]}\binom{p-it+t-1}{t-1}, \\ \alpha_r &= \prod_{i\in[2,r]}\binom{it}{t-1}\prod_{j\in[r+1,s-1]}\binom{p-jt+t-1}{t-1} \\ &\qquad \text{for any } r \in [2, s-1].\end{aligned} \tag{4}$$

It is straightforward to check that this choice of α_r's satisfy (2). Actually, the codes obtained from this choice of α_r's can also be obtained from Construction 1 because $\binom{p-rt-1}{t-1} \mid \alpha_r$ for $r \in [s-1]$ and we can get corresponding ξ_r by Proposition 6.

Note that if α_r's $r \in [s-1]$ have some common divisor, we can always divide them by the common divisor to get smaller α_r's, and thus smaller k and m. Since we are interested in smaller α_r's, we divide the α_r's in (4) by the common divisor $\frac{\prod_{i\in\left[\left\lfloor\frac{s}{2}\right\rfloor+1,s-1\right]}\binom{p-it+t-1}{t-1}}{\prod_{j\in\left[2,\left\lceil\frac{s}{2}\right\rceil\right]}(jt-t+1)}$ and get the formulas in (5). In the following examples, we will show that it is possible to reduce the number of servers of codes from Construction 1 by Construction 3.

Example 8 *When $s = 3$, in [4], the authors took $(\xi_1, \xi_2, \xi_3) = (\binom{2t-1}{t-1}, 1, \binom{2t}{t-1})$ to get a $[t \times m, 3t]$ k-PIR array code with $k = \frac{16t^2+7t+1}{24t^2+15t+3}m$ and*

$$\begin{aligned}m &= \xi_1\binom{3t}{t} + \xi_2\binom{3t}{t-1}\binom{2t+1}{t+1} + \xi_3\binom{3t}{t-1} \\ &= \frac{8t^2+5t+1}{2t^2}\binom{2t}{t-1}\binom{3t}{t-1}.\end{aligned}$$

Since $\frac{\alpha_1}{\alpha_2} = \binom{3t}{2t}/\binom{3t}{t+1} = \frac{t+1}{2t}$, we can take $(\alpha_1, \alpha_2) = (t+1, 2t)$, and get a $[t\times m, 3t]$ k-PIR array code by Construction 3 with the same rate and

$$m = \alpha_1\binom{3t}{t} + \alpha_1\binom{3t}{t+1} + \alpha_2\binom{3t}{2t+1} = \frac{8t^2+5t+1}{t}\binom{3t}{t-1}.$$

We can see when $t = 2$, a $[2 \times 129, 6]$ 79-PIR array code can be obtained from both constructions, but when $t > 2$, the codes obtained from the second construction cannot be achieved by the first one.

Example 9 *When $(s, t) = (4, 2)$, it was shown in [4] that, by taking $(\xi_1, \xi_2, \xi_3, \xi_4) = (15, 3, 4, 24)$, a $[2\times 2124, 8]$ 1221-PIR array code was obtained. By (5), we can take $(\alpha_1, \alpha_2, \alpha_3) = (15, 12, 24)$ and get the same parameters of code as in [4]. However, since α_i's have a common divisor three, we can further take $(\alpha_1, \alpha_2, \alpha_3) = (5, 4, 8)$ and get a $[2 \times 708, 8]$ 407-PIR array code by Theorem 5. Note that, 407 and 708 are coprime, and thus are the smallest k and m of $[2 \times m, 8]$ k-PIR array codes that reach the same rate.*

From Examples 8 and 9, we can see that when s is odd, it is possible to reduce the value of k and m in Construction 1 significantly. However, when s is even, Construction 3 maybe cannot reduce k and m significantly. This is because when s is odd by Proposition 6, if Constructions 1 and 3 coincide, then $\alpha_{\frac{s-1}{2}} = \xi_{\frac{s+1}{2}}\binom{\frac{s+1}{2}t-1}{t-1}$. But by (5), we have $\alpha_{\frac{s-1}{2}} = \prod_{i\in\left[2,\frac{s-1}{2}\right]}\binom{it}{t-1}\prod_{u\in\left[2,\frac{s+1}{2}\right]}(ut-t+1)$, which may not contain all the factors in $\binom{\frac{s+1}{2}t-1}{t-1}$.

$$\alpha_r = \prod_{i\in[2,r]} \binom{it}{t-1} \prod_{j\in\left[r+1,\lfloor \frac{s}{2} \rfloor\right]} \binom{p-jt+t-1}{t-1} \prod_{u\in\left[2,\lceil \frac{s}{2} \rceil\right]} (ut-t+1) \text{ for any } r \in \left[\left\lfloor \frac{s}{2} \right\rfloor\right],$$
$$\alpha_r = \prod_{i\in[2,s-r]} \binom{it}{t-1} \prod_{j\in\left[\lceil \frac{s}{2} \rceil+1,r\right]} \binom{jt}{t-1} \prod_{u\in[2,s-r]} (ut-t+1) \prod_{v\in\left[s-r+1,\lceil \frac{s}{2} \rceil\right]} (vt) \text{ for any } r \in \left[\left\lfloor \frac{s}{2} \right\rfloor + 1, s-1\right]. \tag{5}$$

IV. Classes of PIR Array Codes with Rate $\frac{s}{2s-1}$

In Section III, we generalized Blackburn and Etzion's construction to reduce the number of servers while keeping the rate. However, the number of servers may still be large for real applications. For example, when $(s,t) = (3,2)$, we get a $[2\times m, 6]$ k-PIR array code with $(k,m) = (79,129)$ (see Example 8), which are the smallest k and m of codes that reach the same rate. Thus, it will be interesting to construct PIR array codes with significantly fewer servers by decreasing the rate by a small value. This is our main target in this section.

For any integers $t \geq 2$, a class of $[t\times m, p = t(t+1)]$ k-PIR array codes with $k = \binom{p}{t}$, $m = \binom{p}{t} + \binom{p}{t+1}/t$ was given in [7, Theorem 20]. The codes have only two types of servers: servers with t singletons, and servers with t sums of size $t+1$. We give a generalization of this construction first.

Construction 4 *For any integers s,t with $s \leq t+1$ and $p = st$, we construct the code $\mathbb{C}$ as follows.*

(i) Take all t-subsets of $\{x_i : i \in [p]\}$, each appearing $\binom{t}{t-s+1}$ times as the entries of a server, which contains only singletons.

(ii) All the s-subsets of $[p]$ can be separated into $\binom{p-1}{s-1}$ partitions of all the elements in $[p]$[1]. For each partition $\mathcal{B}$, there exist exactly $\binom{p-s}{t-s+1}$ servers consisting of the t sums $\{\sum_{i\in B} x_i : B \in \mathcal{B}\}$.

Theorem 10 *For any integers s,t with $s \leq t+1$, the code $\mathbb{C}$ is a $[t\times m, p = st]$ k-PIR array code, where $k = \binom{t}{t-s+1}\binom{p}{t}$, $m = \binom{t}{t-s+1}\binom{p}{t} + \binom{p-s}{t-s+1}\binom{p-1}{s-1}$ and rate $k/m = \frac{s}{2s-1}$.*

Theorem 10 can also be proved with the existence of perfect matching of regular bipartite graphs as in Theorem 5. However the proof is omitted here due to the lack of space. Now, we show a simple technique to obtain codes with large item sets from small ones.

Proposition 11 *For any integers s,t with $s \leq t+1$, let t_0 be an integer such that $s \leq t_0+1$ and $t_0 \mid t$. Then there exists a $[t\times m, p = st]$ k-PIR array code with $k = \binom{t_0}{t_0-s+1}\binom{st_0}{t_0}$, $m = \binom{t_0}{t_0-s+1}\binom{st_0}{t_0} + \binom{st_0-s}{t_0-s+1}\binom{st_0-1}{s-1}$ and rate $k/m = \frac{s}{2s-1}$.*

Proof: Let $\mathbb{C}_1, \mathbb{C}_2, \cdots, \mathbb{C}_{t/t_0}$ be the t/t_0 copies of the $[t_0\times m, st_0]$ k-PIR array code from Theorem 10 on disjoint item sets. Then the code $\begin{pmatrix} \mathbb{C}_1 \\ \vdots \\ \mathbb{C}_{t/t_0} \end{pmatrix}$ is the desired code. ∎

[1]The partitions of all the s-subsets of $[p]$ are also called a 1-*factorization*, and exist if and only if $s \mid p$, which was proved in [2].

Example 12 *Take $s = 3$ and $t_0 \in \{2,3\}$ in Proposition 11. Then for any $t \equiv 0 \pmod 2$, we obtain a $[t\times 25, 3t]$ 15-PIR array code; for any $t \equiv 0 \pmod 3$, we also obtain a $[t\times 420, 3t]$ 252-PIR array code. However, the rate $\frac{3}{5}$ may be a bit small compared to the upper bound $\frac{2}{3}$ from Theorem 4. Thus, finding constructions of PIR array codes with both reasonable number of servers and good rate will still be an interesting goal in our future work.*

In particular, taking $t_0 = s-1$ in Proposition 11, we obtain a $[t\times m, p = st]$ k-PIR array code with $k = \binom{s(s-1)}{s-1}$, $m = \binom{s(s-1)}{s-1} + \binom{s(s-1)-1}{s-1}$ and rate $\frac{s}{2s-1}$ when $t \equiv 0 \pmod{s-1}$.

V. Conclusion

In this paper, we gave a generalization of PIR array codes constructed by Blackburn and Etzion in [3], [4] for $p = st$ when $s > 2$ is an integer, and obtained PIR array codes with the same rate but fewer servers. To further decrease the number of servers, we also gave classes of codes with rate $\frac{s}{2s-1}$.

Acknowledgment

Research of Y. M. Chee, H. M. Kiah and H. Zhang is partially funded by the Singapore Ministry of Education under grant MOE2015-T2-2-086. E. Yaakobi was supported in part by the ISF grant 1817/18.

References

[1] H. Asi and E. Yaakobi, "Nearly optimal constructions of PIR and batch codes," *IEEE Trans. Inform. Theory*, vol. 65, pp. 947–964, 2019.

[2] Z. Baranyai, "On the factorization of the complete uniform hypergraph," in *Infinite and Finite Sets*, (A. Hajnal, R. Rado, V. T. Sós, eds), North-Holland, pp. 91–107, 1975.

[3] S. Blackburn and T. Etzion, "PIR array codes with optimal PIR rates," in *Proc. IEEE Int. Symp. on Inf. Theory*, pp. 2658–2662, Aachen, Germany, Jun. 2017.

[4] S. Blackburn and T. Etzion, "PIR array codes with optimal virtual server rate," *arXiv:1607.00235v6*, Feb. 2018.

[5] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, "Private information retrieval," *J. ACM*, vol. 45, pp. 965–981, 1998.

[6] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with low storage overhead," in *Proc. IEEE Int. Symp. on Inf. Theory*, pp. 2852–2856, Hong Kong, China, Jun. 2015.

[7] A. Fazeli, A. Vardy, and E. Yaakobi, "PIR with low storage overhead: coding instead of replication," *arXiv:1505.06241*, May 2015.

[8] P. Hall, "On representatives of subsets," *J. London Math. Soc.*, vol. 10, pp. 26–30, 1935.

[9] S. Rao and A. Vardy, "Lower bound on the redundancy of PIR codes," *arXiv:1605.01869v2*, Feb. 2017.

[10] V. Skachek, "Batch and PIR codes and their connections to locally repairable codes," in *Network Coding and Subspace Designs*, Cham, Switzerland: Springer-Verlag, pp. 427–442, 2018.

[11] Y. Zhang, X. Wang, H, Wei, and G. Ge, "On private information retrieval array codes," *arXiv:1609.09167*, Sep. 2016.