Scilight

Pragmatic Cybersecurity
https://www.sciltp.com/journals/pc

Review

A Unified Framework for Constructing Information-Theoretic Private Information Retrieval

Liangfeng Zhang

School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China; zhanglf@shanghaitech.edu.cn

How To Cite: Zhang, L.F. A Unified Framework for Constructing Information-Theoretic Private Information Retrieval. *Pragmatic Cybersecurity* **2025**, *1*(1), 3.

Received: 12 August 2025
Revised: 7 November 2025
Accepted: 7 November 2025
Published: 18 November 2025

Abstract: Retrieving up-to-date information from a publicly accessible database poses significant threats to the user's privacy. *Private information retrieval* (PIR) protocols allow a user to retrieve any entry from a database, without revealing the identity of the entry being retrieved to the server(s). Such protocols have found numerous applications in both theoretical studies and real-life scenarios. The existing PIR constructions mainly give multi-server *information-theoretic* PIR (IT-PIR) protocols or single-server computational PIR (CPIR) protocols. Compared with CPIR, IT-PIR protocols are computationally more efficient and secure in the presence of unbounded servers. The most classical and challenging problem in the realm of IT-PIR is constructing protocols with lower *communication complexity*. In this review, we introduce a new discrete structure called *families of orthogonal arrays with span capability* (FOASC) and propose a unified framework for constructing IT-PIR protocols. We show how the most influential IT-PIR protocols in the literature can be captured by the framework. We also put forward several interesting open problems concerning FOASC, whose solutions may result in innovative IT-PIR protocols.

Keywords: private information retrieval; families of orthogonal arrays with span capability

1. Introduction

Publicly accessible databases are indispensable resources for retrieving up-to-date information. Access to such databases poses significant risks to the privacy of the user, since the database server(s) may monitor the user's queries and infer what the user is after. Usually the user's retrieval intent is highly valuable and needs careful protection. For example, for a stock-market database an investor's retrieval intent may influence the stock's price; for a patent database a company's retrieval intent may attract unexpected pursuer of the patent; for a Merkle proof database on which a blockchain system such as Ethereum is based, a user's retrieval intent may link the user to the account being read and eventually lead to deanonymization.

Private information retrieval (PIR) protocols [1] are cryptographic protocols that are specifically designed to ensure the users' privacy. Such protocols allow a user to retrieve an entry x_i from a database $\boldsymbol{x} = x_1 \cdots x_n \in \{0,1\}^n$, without revealing the retrieval index $i \in [n]$ to the server. At first glance, the requirements posed by PIR seem quite absurd. However, there does exist a trivial solution that *perfectly* hides i from the server, where the user simply downloads the entire database $\boldsymbol{x}$ from the server and then locally extracts x_i. In particular, the perfect privacy is *information-theoretic* and means that the server learns absolutely no information about i, even if it has unlimited computing power. This trivial solution incurs a *communication cost* of $O(n)$, which could be prohibitive if the database consists of millions or billions of entries. Unfortunately, in their pioneer work [1], Chor, Goldreich, Kushilevitz and Sudan showed that the $O(n)$ communication cost of the trivial solution is asymptotically optimal, if there is only *one server* and *perfect privacy* is needed. Therefore, to have a PIR solution of communication cost $o(n)$, the user must consider two possible relaxations: (1) resort to multiple servers; (2) give up the perfect privacy.

Under the first relaxation, the user may communicate with k ($k > 1$) servers, send a query to every server, receive an answer from the server, and finally reconstruct x_i from the k answers. Specifically, each of the servers should store a copy of the same database $\boldsymbol{x}$ and answer the user's query with $\boldsymbol{x}$. To differentiate from single-server solutions, the k servers must not collude with each other. If the user's retrieval index i is perfectly (i.e., information-theoretically) hidden from the collusion of any t ($t < k$) out of the k servers, then the protocol is said to be a t-*private* k-server information-theoretic PIR (IT-PIR) [1], or (t, k)-*PIR* for short. Under the second relaxation, the user may properly encode its retrieval index i as a query, which essentially leaks no information about i to any *computationally bounded* server that runs polynomial-time algorithms, such that the server remains able to compute an encoded form of x_i to the user. In particular, the privacy of i must be built on various number-theoretic problems (e.g., the quadratic residuosity problem, the composite residuosity problem), which are hard to solve in feasible time by the computationally bounded server. Protocols in this category have been called single-server computational PIR [2], or *CPIR* for short.

Both CPIR and IT-PIR are important cryptographic primitives that have practical influences. Today PIR protocols have found numerous applications in real-life scenarios, e.g., private database search [3], metadata hiding messaging [4,5], private media consumption [6], private contact discovery [7], private blocklist lookups [8], privacy-friendly advertising [9,10], certificate transparency [11], private web search [12], private electronic commerce [13], and private location based services [14], among others. Recently, commercial systems such as Microsoft's Password Monitor [15], Google's Device Enrollment [16], Blyss's Spiral [17], and Brave's FrodoPIR [18] have integrated the functionality of PIR and signed the real world deployment of PIR.

On the theoretical side, both IT-PIR and CPIR are fundamental building blocks of many other cryptographic primitives and have their featured applications. IT-PIR protocols may give locally decodable codes (LDCs) [19–21], error-correcting codes that can recover any bit of the message by reading a few bits of the codeword and guarantee correct recovery even if a constant fraction of the codeword have been *adversarially* corrupted. IT-PIR protocols can also be used to construct multi-party information-theoretically private protocols [22,23]. CPIR protocols imply many important cryptographic primitives such as unconditionally hiding commitment [24], oblivious transfer [25,26], collision-resistant hash functions [27], and efficient zero-knowledge arguments [28].

The efficiency of PIR protocols is mainly measured by *communication complexity* [1], the total number of bits that have to be exchanged between the user and the server(s) in order to retrieve one bit from the database. The most classical and challenging problem in the realm of PIR is constructing protocols with lower communication complexity for a given number of servers. While there are $O(\log n)$-server PIR protocols with polylogrithmic (in n) communication complexity, the main focus has been protocols that use a *constant* number of servers. For IT-PIR, after a long line of arduous explorations [1,29–42], today the most efficient protocols that use a constant number of servers have reached a communication complexity that is subpolynomial in n. For CPIR, protocols [2,43–53] based on various cryptographic assumptions have been proposed and the up-to-date ones may achieve an optimal rate that is close to 1.

Beimel, Ishai and Malkin [54] showed that in any PIR protocol every entry of the database $\boldsymbol{x}$ must be accessed at least once by the servers, in order for the user's retrieval index to be private. The observation is reasonable because any non-accessed entry x_j cannot be of the user's interest and thus reveals partial information about the user's retrieval index i (i.e., $i \neq j$) to the server(s). Consequently, in any PIR protocol the servers computation cost must be $\Omega(n)$, which could be rather undesirable for a large n. In particular, for IT-PIR the servers may need to perform $\Omega(n)$ field operations; for CPIR, the $\Omega(n)$ operations could be expensive public-key operations such as exponentiations. Sion and Carbunar [55] even concluded that deployment of non-trivial CPIR protocols on real hardware would be orders of magnitude less time-efficient than trivially transferring the entire database. Starting from [54], there have been a long line of research that tried to obtain computationally efficient IT-PIR [33,54,56,57] and CPIR [5,11,17,18,58–71,71–79] protocols.

While most of the existing PIR protocols assume *honest-but-curious* servers that always faithfully execute the protocol, *malicious* servers may arbitrarily deviate from the predefined specifications and thus prevent the correct execution of the protocol. In particular, the malicious servers may not respond to the user's queries or even tamper with the responses. Such behaviors may lead to failure in retrieval. Beimel and Stahl [80] initiated the study of *robust* k out of ℓ PIR protocols that allow the user to contact ℓ servers and successfully retrieve x_i as long as at least k out of the ℓ servers respond, and b-*Byzantine* robust k out of ℓ PIR protocols that still guarantee successful retrieval even if b out of the k responses are tampered with. For $k = \ell$, today such protocols are also termed as b-*error correcting* k-server PIR protocols [80–86]. Zhang and Safavi-Naini [87] initiated the study of b-*error detecting* k-server PIR protocols that can detect the existence of wrong responses. Such protocols [88–96] are particularly useful when the PIR servers are implemented by untrusted cloud servers.

Compared with IT-PIR protocols, CPIR protocols do not require the user to communicate with multiple non-colluding servers, an arguably strong assumption. Furthermore, they may achieve much lower communication complexity, compared with constant-server IT-PIR. On the negative side, CPIR protocols are computationally extensive and cannot have short queries or responses, which are crucial for constructing LDCs. Also, the cryptographic assumptions underlying CPIR may become fragile in the presence of modern computing technologies, which however cannot affect the security of IT-PIR. In this review, we are restricted to IT-PIR and focus on the long line of works on constructing communication efficient protocols, which has been the most challenging research problem.

Our review focuses on a unified framework for constructing IT-PIR protocols in the honest-but-curious server model and is different from several excellent existing reviews, which either cover IT-PIR constructions before 2007 and provide no unified framework [97–99] or focus on CPIR [100].

Organization. In Section 2 we give the definitions of IT-PIR and orthogonal arrays; In Section 3 we propose the notion of family of OAs with span capability (FOASC) and give an FOASC based framework for constructing IT-PIR; in Section 4 we show how several most influential IT-PIR constructions can be captured by the proposed framework; in Section 5 we discuss several open problems concerning FOASC. Finally, Section 6 concludes the review.

2. Preliminaries

For any integer $n > 0$, we denote $[n] = \{1, \ldots, n\}$. For any prime power p, we denote by $\mathbb{F}_p$ the finite field of p elements. For any two vectors $\boldsymbol{u} = (u_1, \ldots, u_m)$ and $\boldsymbol{v} = (v_1, \ldots, v_n)$, we denote by $\boldsymbol{u}\|\boldsymbol{v} = (u_1, \ldots, u_m, v_1, \ldots, v_n)$ the concatenation of $\boldsymbol{u}$ and $\boldsymbol{v}$. For any $m \times n$ matrix $\boldsymbol{Q}$, we denote by $\boldsymbol{Q}^\top$ the transpose of $\boldsymbol{Q}$ and denote by $Q_{i,j}$ the (i, j)-entry of $\boldsymbol{Q}$ for all $i \in [m]$ and $j \in [n]$. For any integers $h > 0$ and $i \in [h]$, we denote by $\boldsymbol{e}_h^{(i)}$ the length-h *unit vector* whose ith entry is 1 and all other entries are 0. For any predicate P, we denote by 1_P the indicator value for P, i.e., $1_P = 1$ if P is true and 0 otherwise. For example, $1_{3\in[2]} = 0$. For any vectors $\boldsymbol{z} = (z_1, \ldots, z_n)$ and $\boldsymbol{u} = (u_1, \ldots, u_n)$, we denote $\boldsymbol{z}^{\boldsymbol{u}} = (z_1)^{u_1} \cdots (z_n)^{u_n}$.

2.1. Private Information Retrieval

A t-private k-server PIR ((t, k)-PIR) protocol involves k *servers* $\mathcal{S}_1, \ldots, \mathcal{S}_k$, each storing a copy of the database $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0, 1\}^n$, and a *user* $\mathcal{U}$ who wants to retrieve a database entry x_i, without revealing the retrieval index $i \in [n]$ to any t out of the k servers.

Definition 1 (Private Information Retrieval). *A* t-private k-server private information retrieval ((t, k)-PIR) *protocol* $\mathcal{P} = (\mathcal{Q}, \mathcal{A}, \mathcal{C})$ *consists of three algorithms as follows:*

- $(q_1, \ldots, q_k, \mathsf{aux}) \leftarrow \mathcal{Q}(k, n, i)$*: a randomized* querying *algorithm that takes the public parameters* k, n *and the user's private* retrieval index $i \in [n]$ *as input, and outputs* k queries $q_1, \ldots, q_k$ *together with an* auxiliary information string aux *for reconstruction.*
- $a_j \leftarrow \mathcal{A}(k, j, \boldsymbol{x}, q_j)$*: a deterministic* answering *algorithm that takes the database* $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0, 1\}^n$ *and the query* q_j *as input, and outputs an* answer a_j.
- $x_i \leftarrow \mathcal{C}(k, n, a_1, \ldots, a_k, \mathsf{aux})$*: a deterministic* reconstructing *algorithm that takes* aux *and the* k *answers* $a_1, \ldots, a_k$ *as input, and outputs the* target entry x_i.

For a protocol as above to be a (t, k)*-PIR, the following requirements should be satisfied:*

- **Correctness** . *Informally, if all algorithms of the protocol* $\mathcal{P}$ *are faithfully executed, then the reconstructing algorithm always outputs the correct value of the target entry. Formally, for any* $\boldsymbol{x} \in \{0, 1\}^n$, $i \in [n]$, $(q_1, \ldots, q_k, \mathsf{aux}) \leftarrow \mathcal{Q}(k, n, i)$*, and* $\{a_j \leftarrow \mathcal{A}(k, j, \boldsymbol{x}, q_j)\}_{j=1}^k$,
$$\mathcal{C}(k, n, a_1, \ldots, a_k, \mathsf{aux}) = x_i.$$
- t**-Privacy**. *Informally, any collusion of* $\leq t$ *servers learns no information about the user's retrieval index* i*. Formally, for any* $i_1, i_2 \in [n]$*, any subset* $T \subseteq [k]$ *of size* $\leq t$,
$$\mathcal{Q}_T(k, n, i_1) \overset{\text{id}}{=\!=} \mathcal{Q}_T(k, n, i_2),$$
where $\mathcal{Q}_T$ *denotes concatenation of* j*-th outputs of* $\mathcal{Q}$ *for all* $j \in T$ *and "*$\overset{\text{id}}{=\!=}$*" means that two distributions are identical.*

PIR System. In a (t, k)-PIR system (Figure 1), the user $\mathcal{U}$ starts the execution of the protocol by invoking $\mathcal{Q}(k, n, i)$ to pick a random k-tuple of queries $(q_1, \ldots, q_k)$ along with an auxiliary information string aux, and then sending each query q_j to the server $\mathcal{S}_j$. Subsequently, each server $\mathcal{S}_j$ invokes the answering algorithm $\mathcal{A}(k, j, \boldsymbol{x}, q_j)$ to compute an answer a_j to the user. Finally, $\mathcal{U}$ reconstructs x_i by executing the reconstructing algorithm $\mathcal{C}(k, n, a_1, \ldots, a_k, \mathsf{aux})$.

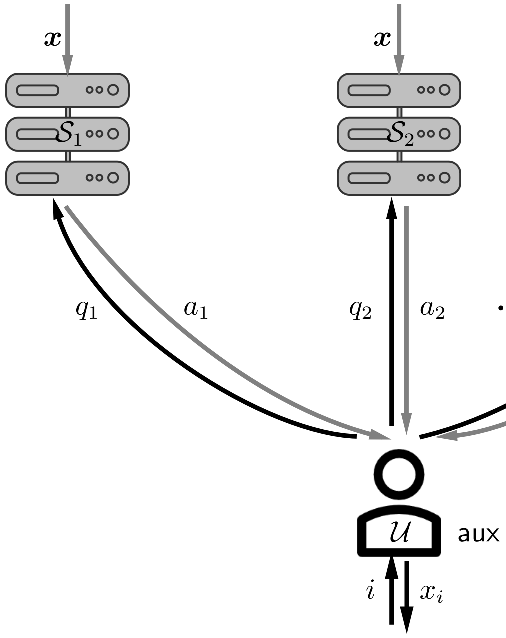

Figure 1. k-Server information-theoretic PIR system.

Communication Complexity. The *communication complexity* of a PIR protocol $\mathcal{P}$, denoted by $\mathbf{C}_{\mathcal{P}}(n, k)$, is a function of k and n that measures the total number of bits communicated between the user and k servers, maximized over all choices of the database $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0, 1\}^n$, the retrieval index $i \in [n]$, and the random coins of the querying algorithm $\mathcal{Q}$.

2.2. Orthogonal Arrays

Orthogonal arrays (OAs) [101] have played prominent roles in the design of experiments and found many applications in computer science. In this review, we shall use OAs to give a unified framework for IT-PIR. For any integers $N, k, t > 0$, we use the term *"$N \times k$ array"* to refer to a matrix $\boldsymbol{Q}$ with N rows and k columns, and use the term *"$N \times t$ subarray"* to refer to a submatrix of $\boldsymbol{Q}$ that consists of t columns of $\boldsymbol{Q}$, where $t \leq k$.

Definition 2 (Orthogonal Array). *Let $N, k, s, t > 0$ be integers. Let $\mathbb{S}$ be a set of s symbols or levels. An $N \times k$ array $\boldsymbol{Q}$ is said to be an* orthogonal array (OA) *of* level *s,* strength *t, and* index *λ, or $\mathsf{OA}(N, k, s, t)$ for short, if every $N \times t$ subarray of $\boldsymbol{Q}$ contains every element of $\mathbb{S}^t$ exactly λ times as a row.*

Example 1. *The following 8×4 array is an $\mathsf{OA}(8, 4, 2, 3)$ with index 1 (where $\mathbb{S} = \{0, 1\}$):*

$$
\begin{matrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1
\end{matrix}
$$

3. A Framework Based on Families of Orthogonal Arrays

In this section, we propose a unified framework that captures several of the most influential constructions of IT-PIR protocols [1,31–34,37,39,40,42] during the past 30 years.

3.1. Families of Orthogonal Arrays with Span Capability

Our framework is based on a new discrete structure called family of orthogonal arrays with span capability (FOASC), which is a set of OAs that satisfy special algebraic properties.

Definition 3 (Families of Orthogonal Arrays with Span Capability). *Let $N, k, s, t, n > 0$ be integers. Let $\mathbb{S}$ be a set of s levels and let $\mathbb{R}$ be a commutative ring with identity. Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ be n functions with domain $\mathbb{S}$ and range $\mathbb{R}$. We say that a set $\{\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}\}$ of $\mathsf{OA}(N, k, s, t)$'s is a* family of orthogonal arrays with $\boldsymbol{\alpha}$-span capability*, or $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$ for short, if for all $i \in [n]$ and $\ell \in [N]$, the columns of the following matrix*

$$\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) = \begin{pmatrix} \alpha_1(Q_{\ell,1}^{(i)}) & \alpha_1(Q_{\ell,2}^{(i)}) & \cdots & \alpha_1(Q_{\ell,k}^{(i)}) \\ \alpha_2(Q_{\ell,1}^{(i)}) & \alpha_2(Q_{\ell,2}^{(i)}) & \cdots & \alpha_2(Q_{\ell,k}^{(i)}) \\ \vdots & \vdots & \cdots & \vdots \\ \alpha_n(Q_{\ell,1}^{(i)}) & \alpha_n(Q_{\ell,2}^{(i)}) & \cdots & \alpha_n(Q_{\ell,k}^{(i)}) \end{pmatrix}$$

span a nonzero multiple of $\boldsymbol{e}_n^{(i)}$, where $\boldsymbol{Q}_\ell^{(i)}$ stands for the ℓth row of $\boldsymbol{Q}^{(i)}$.

Example 2. *The following OAs $\boldsymbol{Q}^{(1)}, \boldsymbol{Q}^{(2)}$ form an $\mathsf{FOASC}(9, 2, 9, 1; \boldsymbol{\alpha})$, where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$ and α_1, α_2 are functions with domain $\mathbb{S} = \mathbb{F}_3^2$ and range $\mathbb{R} = \mathbb{F}_3$ such that $\alpha_1(a, b) = a, \alpha_2(a, b) = b$.*

$\boldsymbol{Q}^{(1)}$ col 1	$\boldsymbol{Q}^{(1)}$ col 2	$\boldsymbol{Q}^{(2)}$ col 1	$\boldsymbol{Q}^{(2)}$ col 2
$(1, 0)$	$(1, 0)$	$(0, 1)$	$(0, 1)$
$(1, 1)$	$(1, 2)$	$(0, 2)$	$(0, 0)$
$(1, 2)$	$(1, 1)$	$(0, 0)$	$(0, 2)$
$(2, 0)$	$(0, 0)$	$(1, 1)$	$(2, 1)$
$(2, 1)$	$(0, 2)$	$(1, 2)$	$(2, 0)$
$(2, 2)$	$(0, 1)$	$(1, 0)$	$(2, 2)$
$(0, 0)$	$(2, 0)$	$(2, 1)$	$(1, 1)$
$(0, 1)$	$(2, 2)$	$(2, 2)$	$(1, 0)$
$(0, 2)$	$(2, 1)$	$(2, 0)$	$(1, 2)$

In fact, there is a vector $\boldsymbol{\lambda} = (2, 2)^\top$ such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda} = \boldsymbol{e}_2^{(i)}$ for all $i \in [2]$ and $\ell \in [9]$.

3.2. The Framework

In this section, we show a unified framework (see Figure 2) for constructing (t, k)-PIR protocols from FOASC. Given an $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N, k, s, t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, the main idea underlying our framework is as follows: *interpret the database $\boldsymbol{x} \in \{0, 1\}^n$ as a vector in $\mathbb{R}^n$, encode the database $\boldsymbol{x}$ as a function $F_{\boldsymbol{x}} : \mathbb{S} \to \mathbb{R}$, which is essentially a linear combination of the n functions $\alpha_1, \ldots, \alpha_n$, i.e.,*

$$F_{\boldsymbol{x}}(\boldsymbol{z}) = \sum_{\tau=1}^{n} x_\tau \cdot \alpha_\tau(\boldsymbol{z}), \tag{1}$$

and finally reduce the problem of retrieving x_i to that of evaluating $F_{\boldsymbol{x}}$ on a random row of $\boldsymbol{Q}^{(i)}$.

Theorem 1. *If there is an $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ are n functions from $\mathbb{S}$ to $\mathbb{R}$, then there is a (t, k)-PIR protocol $\mathcal{P}$ with communication complexity $\mathbf{C}_{\mathcal{P}}(n, k) = k(\log |\mathbb{S}| + \log |\mathbb{R}|)$.*

Proof. It suffices to show that the protocol $\mathcal{P}$ defined by Figure 2 is a (t, k)-PIR with the claimed communication complexity. For the correctness of $\mathcal{P}$, we have that

$$
\begin{aligned}
y &= \sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot a_j \\
&= \sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot F_{\boldsymbol{x}}(q_j) \\
&= \sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot F_{\boldsymbol{x}}(Q_{\ell,j}^{(i)}) \\
&= \sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot \left(\sum_{\tau=1}^{n} x_\tau \cdot \alpha_\tau \left(Q_{\ell,j}^{(i)} \right) \right) \\
&= \sum_{\tau=1}^{n} x_\tau \cdot \left(\sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot \alpha_\tau \left(Q_{\ell,j}^{(i)} \right) \right) \\
&= \boldsymbol{x} \cdot \boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda}_\ell^{(i)} \\
&= \boldsymbol{x} \cdot \omega_\ell^{(i)} \boldsymbol{e}_n^{(i)} \\
&= \omega_\ell^{(i)} x_i.
\end{aligned}
$$

Clearly, we have that $1_{y=\omega_\ell^{(i)}} = x_i$ and thus the protocol is correct.

For t-privacy, we consider the collusion of any t servers, say $\mathcal{S}_{j_1}, \ldots, \mathcal{S}_{j_t}$, and let $T = \{j_1, \ldots, j_t\}$. As per the querying algorithm $\mathcal{Q}$ in Figure 2, for any $i_1, i_2 \in [n]$, $\mathcal{Q}_T(k,n,i_1)$ (resp. $\mathcal{Q}_T(k,n,i_2)$) is a random row of the $N \times t$ subarray of $\boldsymbol{Q}^{(i_1)}$ (resp. $\boldsymbol{Q}^{(i_2)}$) that consists of the columns indexed by T. Since $\boldsymbol{Q}^{(i_1)}$ and $\boldsymbol{Q}^{(i_2)}$ are $\mathsf{OA}(N,k,s,t)$'s, $\mathcal{Q}_T(k,n,i_1)$ and $\mathcal{Q}_T(k,n,i_2)$ are both uniformly distributed over $\mathbb{S}^t$. Hence, $\mathcal{Q}_T(k,n,i_1) \stackrel{\text{id}}{=\!=} \mathcal{Q}_T(k,n,i_2)$, i.e., the protocol $\mathcal{P}$ is t-private.

In our framework, the client sends a query $q_j \in \mathbb{S}$ to every server $\mathcal{S}_j$ and the server returns an answer $a_j \in \mathbb{R}$ to the client. Thus, the communication complexity is $\mathbf{C}_{\mathcal{P}}(n,k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|)$, where $\log|\mathbb{S}|$ (resp. $\log|\mathbb{R}|$) is the bit length of every element of $\mathbb{S}$ (resp. $\mathbb{R}$). □

The underlying public parameters and structures:

- $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$: an $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ consists of n functions $\alpha_1, \ldots, \alpha_n : \mathbb{S} \to \mathbb{R}$ from a set $\mathbb{S}$ of s levels to a commutative ring $\mathbb{R}$ with identity.
- $F_{\boldsymbol{x}}$: a function representing the database $\boldsymbol{x}$, based on the $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ (Equation (1)).
- $\{\boldsymbol{\lambda}_\ell^{(i)}, \omega_\ell^{(i)}\}$: a vector $\boldsymbol{\lambda}_\ell^{(i)} = (\lambda_{\ell,1}^{(i)}, \ldots, \lambda_{\ell,k}^{(i)})^\top \in \mathbb{R}^k$ and a nonzero ring element $\omega_\ell^{(i)} \in \mathbb{R}$ such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda}_\ell^{(i)} = \omega_\ell^{(i)} \boldsymbol{e}_n^{(i)}$ for all $i \in [n]$ and $\ell \in [N]$.

The private information retrieval protocol $\mathcal{P} = (\mathcal{Q}, \mathcal{A}, \mathcal{C})$:

- $\mathcal{Q}(k,n,i)$: Choose $\ell \in [N]$ uniformly. Output $(q_1, \ldots, q_k) = \left(Q_{\ell,1}^{(i)}, \ldots, Q_{\ell,k}^{(i)}\right)$ and $\mathsf{aux} = \ell$.
- $\mathcal{A}(k,j,\boldsymbol{x},q_j)$: Output $a_j = F_{\boldsymbol{x}}(q_j)$.
- $\mathcal{C}(k,n,a_1,\ldots,a_k,\mathsf{aux})$: Compute $y = \sum_{j=1}^{k} \lambda_{\ell,j}^{(i)} \cdot a_j$ and output $1_{y=\omega_\ell^{(i)}}$.

Figure 2. A unified framework for constructing (t,k)-PIR from $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$

Remark 1. *Except Dvir and Gopi [40], our framework can capture all PIR protocols considered by this review with* $\omega_\ell^{(i)} = 1$ *and thus the reconstructing algorithm can simply output* y.

4. PIR Constructions within the Proposed Framework

In this section, we show how several of the most influential constructions [1,33–35,37,40,42] of (t,k)-PIR protocols are captured by the proposed framework, which may inspire new constructions with lower communication complexity.

4.1. Protocols Based on Covering Codes

Chor, Goldreich, Kushilevitz, and Sudan [1] proposed a $(1,2)$-PIR with communication complexity $O(n^{1/3})$ in 1995, which had been the most influential (1,2)-PIR for almost 20 years.

To better understand their protocol, we identify every integer $i \in [n]$ with a tuple $(i_1, i_2, i_3) \in [n^{1/3}]^3$, which can be done by sorting the tuples in $[n^{1/3}]^3$ alphabetically and identifying every $i \in [n]$ with the ith tuple. Underlying [1] is a $(1,8)$-PIR with communication complexity $O(n^{1/3})$, where the n bits of the database $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0,1\}^n$ are organized as a hypercube of side length $h = n^{1/3}$ (for ease of exposition, assume that n is a cubic number), every bit x_i is located at a position $(i_1, i_2, i_3) \in [h]^3$ of the hypercube, and the 8 servers are named as $\mathcal{S}_{000}, \ldots, \mathcal{S}_{111}$. The $(1,2)$-PIR is obtained from the $(1,8)$-PIR by asking $\mathcal{S}_{000}$ to simulate half of the servers, i.e., $\mathcal{S}_{000}, \mathcal{S}_{100}, \mathcal{S}_{010}, \mathcal{S}_{001}$, and asking $\mathcal{S}_{111}$ to simulate the remaining servers, i.e., $\mathcal{S}_{111}, \mathcal{S}_{011}, \mathcal{S}_{101}, \mathcal{S}_{110}$. Specifically, the simulation strategy is based on a *covering code* with radius 1 for $\{0,1\}^3$.

The FOASC and Database Representation. Let $\mathcal{H} = \{H_1, \ldots, H_\zeta\}$ be the power set of $[h]$, where $\zeta = 2^h$. Denote by $A \oplus B = (A \setminus B) \cup (B \setminus A)$ the symmetric difference of any two sets A, B. Within our framework, underlying the $(1,2)$-PIR of [1] is an $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N,k,s,t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = \zeta^3, k = 2, s = \zeta^3, t = 1$, and

$$Q_{\ell,1}^{(i)} = (H_{\ell_1}, H_{\ell_2}, H_{\ell_3}); \quad Q_{\ell,2}^{(i)} = (H_{\ell_1} \oplus \{i_1\}, H_{\ell_2} \oplus \{i_2\}, H_{\ell_3} \oplus \{i_3\}) \tag{2}$$

for all $i = (i_1, i_2, i_3) \in [h]^3$ and $\ell = (\ell_1, \ell_2, \ell_3) \in [\zeta]^3$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathcal{H}^3$ and range $\mathbb{R} = (\mathbb{F}_2)^{3h+1}$, and for all $\tau \in [n]$ and $\boldsymbol{z} = (U, V, W) \in \mathbb{S}$,

$$\begin{aligned}\alpha_\tau(\boldsymbol{z}) = 1_{\tau \in U \times V \times W} \big\| \big(1_{\tau \in (U \oplus \{c\}) \times V \times W}\big)_{c \in [h]} \big\| \big(1_{\tau \in U \times (V \oplus \{c\}) \times W}\big)_{c \in [h]} \\ \big\| \big(1_{\tau \in U \times V \times (W \oplus \{c\})}\big)_{c \in [h]}.\end{aligned} \tag{3}$$

The Reconstruction Coefficients. To see that the FOASC Equation (2) gives a $(1,2)$-PIR, it suffices to note that for all $i = (i_1, i_2, i_3) \in [h]^3$ and $\ell = (\ell_1, \ell_2, \ell_3) \in [\zeta]^3$, there is a vector

$$\boldsymbol{\lambda}_\ell^{(i)} = 1 \| \boldsymbol{e}_h^{(i_1)} \| \boldsymbol{e}_h^{(i_2)} \| \boldsymbol{e}_h^{(i_3)} \| 1 \| \boldsymbol{e}_h^{(i_1)} \| \boldsymbol{e}_h^{(i_2)} \| \boldsymbol{e}_h^{(i_3)} \tag{4}$$

such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda}_\ell^{(i)} = \boldsymbol{e}_n^{(i)}$. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = 2(\log|\mathbb{S}| + \log|\mathbb{R}|) = 2(3h + 3h + 1) = 12h + 2 = O(n^{1/3}).$$

4.2. Protocols Based on Polynomial Interpolations

4.2.1. Lagrange Interpolations

Chor, Goldreich, Kushilevitz, and Sudan [1] proposed a Lagrange interpolation based (t,k)-PIR with communication complexity $O(n^{1/\lfloor (k-1)/t \rfloor})$ in 1995, which introduced the polynomial interpolation techniques to the realm of PIR and inspired many subsequent constructions.

The FOASC and Database Representation. Let $d = \lfloor (k-1)/t \rfloor$ and let h be the least integer such that $\binom{h}{d} \geq n$. Then there exist n vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \in \{0,1\}^h \subseteq \mathbb{F}_p^h$ of Hamming weight d, where $p > k$ is a prime. Let $\boldsymbol{R}_1, \ldots, \boldsymbol{R}_N$ be the $h \times t$ matrices over $\mathbb{F}_p$, where $N = p^{ht}$. For every $i \in [n]$ and $\ell \in [N]$, the user's retrieval index i is hidden with a degree t curve

$$\boldsymbol{q}_\ell^{(i)}(\theta) = \boldsymbol{u}_i + \boldsymbol{R}_\ell \cdot (\theta, \theta^2, \ldots, \theta^t)^\top.$$

Within our framework, underlying their (t,k)-PIR is an $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N,k,s,t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = p^{ht}, s = p^h$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{q}_\ell^{(i)}(j) \tag{5}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathbb{F}_p^h$ and range $\mathbb{R} = \mathbb{F}_p$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$,

$$\alpha_\tau(\boldsymbol{z}) = \boldsymbol{z}^{\boldsymbol{u}_\tau}. \tag{6}$$

The Reconstruction Coefficients. To see the FOASC (5) gives a (t,k)-PIR, it suffices to note that for all $i \in [n]$

and $\ell \in [N]$, there is a vector

$$\boldsymbol{\lambda}_\ell^{(i)} = \left(\prod_{j\in[k]\setminus\{1\}} \frac{j}{j-1}, \prod_{j\in[k]\setminus\{2\}} \frac{j}{j-2}, \ldots, \prod_{j\in[k]\setminus\{k\}} \frac{j}{j-k} \right)^\top \tag{7}$$

such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda}_\ell^{(i)} = \boldsymbol{e}_n^{(i)}$. Specifically, the entries of $\boldsymbol{\lambda}_\ell^{(i)}$ are k coefficients for Lagrange interpolation. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|) = k(h\log p + \log p) = O(n^{1/\lfloor (k-1)/t \rfloor}).$$

4.2.2. Hermite Interpolations

Woodruff and Yekhanin [33] proposed a Hermite interpolation based (t,k)-PIR with communication complexity $O(n^{1/\lfloor (2k-1)/t \rfloor})$ in 2005, which refined the Lagrange interpolation techniques of [1] and has been the most influential (t,k)-PIR for $t > 1$ during the past 20 years.

The FOASC and Database Representation. Let $d = \lfloor (2k-1)/t \rfloor$ and let h be the least integer such that $\binom{h}{d} \geq n$. Then there exist n vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \in \{0,1\}^h \subseteq \mathbb{F}_p^h$ of Hamming weight d, where $p > k$ is a prime. Let $\boldsymbol{R}_1, \ldots, \boldsymbol{R}_N$ be the $h \times t$ matrices over $\mathbb{F}_p$, where $N = p^{ht}$. For every $i \in [n]$ and $\ell \in [N]$, the user's retrieval index i is hidden with a degree t curve

$$\boldsymbol{q}_\ell^{(i)}(\theta) = \boldsymbol{u}_i + \boldsymbol{R}_\ell \cdot (\theta, \theta^2, \ldots, \theta^t)^\top.$$

Within our framework, underlying their (t,k)-PIR is an $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N,k,s,t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = p^{ht}$, $s = p^h$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{q}_\ell^{(i)}(j) \tag{8}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathbb{F}_p^h$ and range $\mathbb{R} = (\mathbb{F}_p)^{h+1}$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$,

$$\alpha_\tau(\boldsymbol{z}) = \left(\boldsymbol{z}^{\boldsymbol{u}_\tau}, \frac{\partial(\boldsymbol{z}^{\boldsymbol{u}_\tau})}{\partial z_1}, \ldots, \frac{\partial(\boldsymbol{z}^{\boldsymbol{u}_\tau})}{\partial z_h} \right). \tag{9}$$

Hermite Interpolation Basics. Based on an observation from [33], for any k distinct nonzero field elements $\theta_1, \theta_2, \ldots, \theta_k \in \mathbb{F}_p^*$, the $(2k) \times (2k)$ matrix

$$\boldsymbol{M}_{\theta_1,\theta_2,\ldots,\theta_k} = \begin{pmatrix} 1 & \theta_1 & \theta_1^2 & \cdots & \theta_1^{2k-1} \\ 0 & 1 & 2\theta_1 & \cdots & (2k-1)\theta_1^{2k-2} \\ 1 & \theta_2^1 & \theta_2^2 & \cdots & \theta_2^{2k-1} \\ 0 & 1 & 2\theta_2 & \cdots & (2k-1)\theta_2^{2k-2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \theta_k^1 & \theta_k^2 & \cdots & \theta_k^{2k-1} \\ 0 & 1 & 2\theta_k & \cdots & (2k-1)\theta_k^{2k-2} \end{pmatrix}^\top$$

is nonsingular. Specifically, $\boldsymbol{M}_{1,2,\ldots,k}$ is nonsingular and thus there is a vector $\boldsymbol{\mu} \in \mathbb{F}_p^{2k}$ such that

$$\boldsymbol{M}_{1,2,\ldots,k} \cdot \boldsymbol{\mu} = \boldsymbol{e}_{2k}^{(1)}.$$

For any degree $< 2k$ univariate polynomial $\varphi(\theta) = \varphi_0 + \varphi_1\theta + \cdots + \varphi_{2k-1}\theta^{2k-1}$, we note that

$$\big(\varphi(1), \varphi'(1), \ldots, \varphi(k), \varphi'(k)\big) = \big(\varphi_0, \varphi_1, \ldots, \varphi_{2k-1}\big) \cdot \boldsymbol{M}_{1,2,\ldots,k}.$$

Therefore, one can easily recover $\varphi_0 = \varphi(0)$ from $\{\varphi(j), \varphi'(j)\}_{j=1}^k$ as follows

$$\varphi_0 = \big(\varphi(1), \varphi'(1), \ldots, \varphi(k), \varphi'(k)\big) \cdot \boldsymbol{\mu}.$$

The Reconstruction Coefficients. For every $\tau \in [n]$, $\phi_\tau(\theta) = (\boldsymbol{q}_\ell^{(i)}(\theta))^{\boldsymbol{u}_\tau}$ is a polynomial of degree $< 2k$.

Clearly, there is a $k(h+1) \times (2k)$ matrix $\boldsymbol{T}_\ell$ that only depends on $\boldsymbol{R}_\ell$ such that

$$\big(\phi_\tau(1), \phi'_\tau(1), \ldots, \phi_\tau(k), \phi'_\tau(k)\big) = \alpha_\tau\big(\boldsymbol{Q}_\ell^{(i)}\big) \cdot \boldsymbol{T}_\ell.$$

Note that $\phi_\tau(0) = (\boldsymbol{q}_\ell^{(i)}(0))^{\boldsymbol{u}_\tau} = (\boldsymbol{u}_i)^{\boldsymbol{u}_\tau} = 1_{\tau=i}$. Therefore, for all $i \in [n]$ and $\ell \in [N]$, we have

$$\boldsymbol{e}_n^{(i)} = \big(\phi_1(0), \phi_2(0), \ldots, \phi_n(0)\big)^\top = \boldsymbol{\alpha}\big(\boldsymbol{Q}_\ell^{(i)}\big) \cdot \underbrace{\boldsymbol{T}_\ell \cdot \boldsymbol{\mu}}_{\boldsymbol{\lambda}_\ell^{(i)}}. \tag{10}$$

Hence, the FOASC (8) gives a (t,k)-PIR. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|) = k(h\log p + (h+1)\log p) = O(n^{1/\lfloor(2k-1)/t\rfloor}).$$

4.3. Protocols Based on Matching Vectors over Finite Fields

Yekhanin [34] proposed a $(1,3)$-PIR with communication complexity $O(n^{1/r})$ for any integer r such that $p = 2^r - 1$ is a Mersenne prime in 2007. Assuming that there are infinitely many Mersenne primes, his construction gives a $(1,3)$-PIR with communication complexity $O(n^{1/\log\log n})$, which is the *first* PIR protocol that uses a *constant* number of servers and achieves a *subpolynomial* communication complexity. While the protocols in Sections 4.1 and 4.2 are among the *first* generation of PIR, Yekhanin's construction [34] is best known for initiating the constructions of the *third* generation of PIR (The *second* generation of PIR consists of [102] and attracts limited attention in the realm of PIR.).

4.3.1. Yekhanin's Construction

The core building block underlying Yekhanin's PIR is a subset of $\mathbb{F}_p^*$ that is both combinatorially nice and algebraically nice. A set $S \subseteq \mathbb{F}_p^*$ is (h,n)-*combinatorially nice* if there exist two sets $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n\}, \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\} \subseteq \mathbb{F}_p^h$ of vectors such that

- $\langle \boldsymbol{u}_i, \boldsymbol{v}_i \rangle = 0$ for all $i \in [n]$; and
- $\langle \boldsymbol{u}_i, \boldsymbol{v}_j \rangle \in S$ for all $i, j \in [n]$ such that $i \neq j$.

The two sets of vectors are said to form an *S-matching family* in $\mathbb{F}_p^h$. Yekhanin [34] showed that for any Mersenne prime $p = 2^r - 1$ and any integer $d \geq p-1$, the subgroup

$$S = \langle 2 \rangle = \{1, 2, \ldots, 2^{r-1}\}$$

of $\mathbb{F}_p^*$ is (h,n)-combinatorially nice for $h = \binom{d-1+(p-1)/r}{(p-1)/r}$ and $n = \binom{d}{p-1}$. Specifically, if we denote by $\mathbf{1}_h$ the all-one vector of length h, then the $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n$ constructed by [34] satisfy $\langle \boldsymbol{u}_i, \mathbf{1}_h \rangle \neq 0$ for all $i \in [n]$. A set $S \subseteq \mathbb{F}_p^*$ is k-*algebraically nice* if there exist two sets $S_0, S_1 \subseteq \mathbb{F}_p$ such that

- $|S_0| > 0, |S_1| = k$, and
- $|S_0 \cap (\sigma + \delta S_1)| \equiv 0 \pmod 2$ for all $\sigma \in \mathbb{F}_p$ and $\delta \in S$.

Yekhanin [34] showed that for any Mersenne prime $p = 2^r - 1$, the set $S = \langle 2 \rangle$ is k-algebraically nice for $k = 3$. In particular, if g is a generator of $\mathbb{F}_{2^r}^*$ and $\gamma \in \mathbb{F}_p$ is an integer such that $1 + g + g^\gamma = 0$, then one can choose

$$S_1 = \{0, 1, \gamma\}.$$

Furthermore, if L is the linear subspace of $\mathbb{F}_2^p$ that consists of the incidence vectors of the sets $\{\sigma + \delta S_1\}_{\sigma\in\mathbb{F}_p, \delta\in S}$, then S_0 can be any nonempty subset of $\mathbb{F}_p$ whose indicator vector belongs to $L^\perp$, the dual space of L. From now on, we denote $d_1 = 0, d_2 = 1$, and $d_3 = \gamma$.

The FOASC and Database Representation. Let $\mathbb{F}_p^h = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N\}$, where $N = p^h$. Within our framework, underlying Yekhanin's $(1,3)$-PIR is an $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N,k,s,t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = p^h, k = 3, s = p^h, t = 1$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{w}_\ell + d_j \cdot \boldsymbol{v}_i \tag{11}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathbb{F}_p^h$ and range $\mathbb{R} = \mathbb{F}_2^p$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$

$$\alpha_\tau(\boldsymbol{z}) = \left(1_{\langle \boldsymbol{u}_\tau, \boldsymbol{z}+\rho\cdot \mathbf{1}_h\rangle \in S_0}\right)_{\rho\in\mathbb{F}_p}. \tag{12}$$

The Reconstruction Coefficients. Note that $\langle \mathbf{u}_\tau, \mathbf{1}_h\rangle \neq 0$ for all $\tau \in [n]$. There is a field element $\rho_\ell \in \mathbb{F}_p$ such that $\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell + \rho_\ell \cdot \mathbf{1}_h\rangle \in S_0$. To see that the FOASC (11) gives a $(1,3)$-PIR, it suffices to note that for any $i \in [n]$ and $\ell \in [N]$, there is a binary vector

$$\boldsymbol{\lambda}_\ell^{(i)} = \Big(0,\ldots,0,\underbrace{1}_{(\rho_\ell+1)\text{st entry}},0,\ldots,0,\underbrace{1}_{(p+\rho_\ell+1)\text{st entry}},0,\ldots,0,\underbrace{1}_{(2p+\rho_\ell+1)\text{st entry}},0,\ldots,0\Big)^\top \tag{13}$$

of length $3p$ and weight 3 (which is the cardinality of S_1) such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)})\cdot\boldsymbol{\lambda}_\ell^{(i)} = \boldsymbol{e}_n^{(i)}$. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = 3(\log|\mathbb{S}| + \log|\mathbb{R}|) = 3(h\log p + p) = O(n^{1/r}).$$

4.3.2. Raghavendra's Interpretation

Raghavendra [35] presented a more friendly interpretation of Yekhanin's $(1,3)$-PIR, which had inspired Efremenko [37], a milestone in the third generation PIR.

The FOASC and Database Representation. With the same notation as in Section 4.3.1, Raghavendra [35] considered a polynomial

$$P(\theta) = 1 + \theta + \theta^\gamma \in \mathbb{F}_{2^r}[\theta]$$

such that $P(g^\delta) = 0$ for $g \in \mathbb{F}_{2^r}^*$ and all $\delta \in S$, and $P(1) = 1$. He represented the database $\boldsymbol{x}$ as a function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) with domain $\mathbb{S} = \mathbb{F}_p^h$ and range $\mathbb{R} = \mathbb{F}_{2^r}$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$,

$$\alpha_\tau(\boldsymbol{z}) = g^{\langle \boldsymbol{u}_\tau, \boldsymbol{z}\rangle}. \tag{14}$$

The Reconstruction Coefficients. Note that $P(g^{\langle \boldsymbol{u}_\tau, \boldsymbol{v}_i\rangle}) = 1_{\tau=i}$. To see that the new function $F_{\boldsymbol{x}}$ defined by (1), (14) and the FOASC defined by (11) give a $(1,3)$-PIR, it suffices to note that for any $i \in [n]$ and $\ell \in [N]$, there is a vector

$$\boldsymbol{\lambda}_\ell^{(i)} = g^{-\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell\rangle}\cdot(1,1,1)^\top \tag{15}$$

such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)})\cdot\boldsymbol{\lambda}_\ell^{(i)} = \boldsymbol{e}_n^{(i)}$. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = 3(\log|\mathbb{S}| + \log|\mathbb{R}|) = 3(h\log p + r) = O(n^{1/r}).$$

4.4. Protocols Based on Matching Vectors over Finite Rings

For any integer $r \geq 2$, Efremenko [37] proposed a $(1,2^r)$-PIR with communication complexity $\mathcal{L}_r(n) = \exp(O((\log n)^{1/r}(\log\log n)^{1-1/r}))$ in 2009. Specifically, for $r = 2$, their construction can be optimized to give the first $(1,3)$-PIR with subpolynomial communication complexity, without making any assumptions such as the infinity of Mersenne primes [34]. In several subsequent works [38–40,42], the number of servers required by [37] was further reduced.

4.4.1. Efremenko's Construction

Let $m = p_1p_2\cdots p_r$ be the product of r distinct primes $p_1, p_2, \ldots, p_r$ and let p be a prime/prime power such that $m|(p-1)$. The *canonical set* of m is the set $S_m \subseteq \mathbb{Z}_m$ of $2^r - 1$ nonzero integers $\delta \in \mathbb{Z}_m$ that satisfy $\delta \bmod p_j \in \{0,1\}$ for all $j \in [r]$. Underlying [37] is an S_m*-matching family* $\{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_n\},\{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\} \subseteq \mathbb{Z}_m^h$ of size $n = \exp(O((\log h)^r/(\log\log h)^{r-1}))$ such that $\langle \boldsymbol{u}_i, \boldsymbol{v}_i\rangle = 0$ for all $i \in [n]$; and $\langle \boldsymbol{u}_i, \boldsymbol{v}_j\rangle \in S_m$ for all $i \neq j$. Such families can be obtained from Gromulsz's set systems [103]. Another ingredient of [37] is an S_m*-decoding polynomial*

$$P(\theta) = \rho_1\theta^{d_1} + \cdots + \rho_k\theta^{d_k} \in \mathbb{F}_p[\theta],$$

such that $P(g^\delta) = 0$ for all $\delta \in S_m$ and $P(1) = 1$, where $g \in \mathbb{F}_p^*$ is of order m. A trivial construction of $P(\theta)$, i.e., $P(\theta) = \prod_{\delta\in S_m}(\theta - g^\delta)/\prod_{\delta\in S_m}(1-g^\delta)$, requires $k = 2^r$.

The FOASC and Database Representation. Let $\mathbb{Z}_m^h = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N\}$, where $N = m^h$. Within our framework, underlying [37] is an $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N, k, s, t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = m^h, k = 2^r, s = m^h, t = 1$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{w}_\ell + d_j \cdot \boldsymbol{v}_i \tag{16}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathbb{Z}_m^h$ and range $\mathbb{R} = \mathbb{F}_p$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$

$$\alpha_\tau(\boldsymbol{z}) = g^{\langle \mathbf{u}_\tau, \boldsymbol{z} \rangle}. \tag{17}$$

The Reconstruction Coefficients. Note that $P(g^{\langle \boldsymbol{u}_\tau, \boldsymbol{v}_i \rangle}) = 1_{\tau = i}$. To see that the FOASC (16) gives a $(1, k)$-PIR, it suffices to note that for all $i \in [n]$ and $\ell \in [N]$, there is a vector

$$\boldsymbol{\lambda}_\ell^{(i)} = g^{-\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell \rangle} \cdot (\rho_1, \rho_2, \ldots, \rho_k)^\top \tag{18}$$

such that $\boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{\lambda}_\ell^{(i)} = \boldsymbol{e}_n^{(i)}$. By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n, k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|) = k(h \log m + \log p) = \mathcal{L}_r(n).$$

4.4.2. Sparse Decoding Polynomials

Efremenko [37] observed that for $r = 2$, a specific modulus such as $m = 511 = 7 \times 73$ may have an S_m-decoding polynomial with 3 monomials and thus give a $(1, 3)$-PIR rather than a $(1, 4)$-PIR. He left it as an open problem to find S_m-decoding polynomials that consist of $< 2^r$ monomials for a general modulus $m = p_1 p_2 \cdots p_r$.

Shortly after [37], Itoh and Suzuki [38] showed a composition theorem which states that if $m = p_1 \cdots p_r$ and $m' = p_1' \cdots p_{r'}'$ are two coprime moduli and there exist an S_m-decoding polynomial with $\leq k$ monomials and an $S_{m'}$-decoding polynomial with $\leq k'$ monomials, then there is an $S_{m''}$-decoding polynomial with $k'' \leq kk'$ monomials for $m'' = mm'$. In general, we say that a modulus $m = p_1 p_2 \cdots p_r$ is *good* if it has an S_m-decoding polynomial with $< 2^r$ monomials. The composition theorem implies that a good modulus can help reduce the number of required servers in the matching vector based PIR protocols.

Chee, Feng, Ling, Wang and Zhang [39] conducted an in-depth study of Efremenko's open problem and showed that a surprising result: *Any Mersenne number (numbers of the form $2^\sigma - 1$) that is the product of two distinct primes must be a good modulus in Efremenko's construction.* By computer search, they identified 50 new good modulus of such form, the least of which is $M_{11} = 2^{11} - 1$ and the largest of which is $M_{7331} = 2^{7331} - 1$. With these good moduli, they obtained $(1, k_r)$-PIR protocols with communication complexity $\mathcal{L}_r(n)$, where $r \geq 2$ and

$$k_r = \begin{cases} 3^{r/2}, & 1 < r \leq 103, r \text{ is even;} \\ 8 \cdot 3^{(r-3)/2}, & 1 < r \leq 103, r \text{ is odd;} \\ (\frac{3}{4})^{51} \cdot 2^r, & r \geq 104. \end{cases} \tag{19}$$

However, it remains an open problem to show that there are infinitely many such Mersenne numbers. Further study of the good moduli can be found in [104].

4.4.3. Hermite-like Interpolations over Exotic Rings

The transition from Lagrange interpolation [1] to Hermite interpolation [34] allows each server to return more information and thus halves the number of required servers, in order to achieve the same asymptotic communication complexity. Inspired by this transition, Dvir and Gopi [40] halved the number of servers required by Efremenko [37] and obtained a $(1, 2^{r-1})$-PIR with communication complexity $\mathcal{L}_r(n)$ for any integer $r \geq 2$ in 2015. Specifically, for $r = 2$, they got a $(1, 2)$-PIR with communication complexity $\mathcal{L}_2(n)$, which eventually broke the communication complexity record of $O(n^{1/3})$ set by [1]. Their construction was obtained with Hermite-like interpolations with generalized derivatives over an exotic ring $\mathcal{R}$.

The FOASC and Database Representation. Let $m = p_1 p_2 \cdots p_r$ be the product of r distinct primes $p_1, p_2, \ldots, p_r$. Let S_m be the canonical set of m. Let $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n\}, \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\} \subseteq \mathbb{Z}_m^h$ be an S_m-matching family of size $n = \exp(O((\log h)^r / (\log\log h)^{r-1}))$. Let $\mathcal{R} = \mathbb{Z}_m[g]/(g^m - 1)$. Let $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N$ be all elements of $\mathbb{Z}_m^h$, where $N = m^h$. For $k = 2^{r-1}$ and every $j \in [k]$, set $d_j = j - 1$. Within our framework, underlying [40] is an $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N, k, s, t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = m^h, k = 2^{r-1}, s = m^h,$

$t = 1$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{w}_\ell + d_j \cdot \boldsymbol{v}_i \tag{20}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ (Equation (1)) has domain $\mathbb{S} = \mathbb{Z}_m^h$ and range $\mathbb{R} = \mathcal{R}^{h+1}$, where for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$

$$\alpha_\tau(\boldsymbol{z}) = \big(1, \boldsymbol{u}_\tau\big) \cdot g^{\langle \boldsymbol{u}_\tau, \boldsymbol{z} \rangle}. \tag{21}$$

Hermite Interpolation on Multiplicative Lines. Consider any polynomial of the form

$$\varphi(\theta) = \varphi_0 + \sum_{\delta \in S_m} \varphi_\delta \theta^\delta \in \mathbb{R}[\theta].$$

If we denote $\bar{\varphi}(\theta) = \sum_{\delta \in S_m} \delta \cdot \varphi_\delta \cdot \theta^\delta$, then there is a $(2k) \times (2k)$ matrix $\boldsymbol{M}$ such that

$$\big(\varphi(g^{d_1}), \bar{\varphi}(g^{d_1}), \ldots, \varphi(g^{d_k}), \bar{\varphi}(g^{d_k})\big) = (\varphi_0, \ldots, \varphi_\delta, \ldots) \cdot \underbrace{\begin{pmatrix} 1 & \cdots & g^{d_1\delta} & \cdots \\ 0 & \cdots & \delta g^{d_1\delta} & \cdots \\ \vdots & \cdots & \vdots & \cdots \\ 1 & \cdots & g^{d_k\delta} & \cdots \\ 0 & \cdots & \delta g^{d_k\delta} & \cdots \end{pmatrix}^\top}_{\boldsymbol{M}}$$

Dvir and Gopi [40] showed that there is a vector $\boldsymbol{\mu} \in \mathcal{R}^{2k}$ and a ring element $\nu \in \mathcal{R}$ such that

$$(\nu, 0, \ldots, 0)^\top = \boldsymbol{M} \cdot \boldsymbol{\mu}$$

and $\nu \bmod p_j \neq 0$ for all $j \in [r]$. Therefore, we have that

$$\big(\varphi(g^{d_1}), \bar{\varphi}(g^{d_1}), \ldots, \varphi(g^{d_k}), \bar{\varphi}(g^{d_k})\big) \cdot \boldsymbol{\mu} = \varphi_0 \nu.$$

The Reconstruction Coefficients. For every $\tau \in [n]$, consider the univariate polynomial

$$\phi_\tau(\theta) = g^{\langle \boldsymbol{u}_\tau, \boldsymbol{w}_\ell \rangle} \cdot \theta^{\langle \boldsymbol{u}_\tau, \boldsymbol{v}_i \rangle}.$$

Note that the constant term of this function is $g^{\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell \rangle} \cdot 1_{\tau = i}$. Therefore,

$$\big(\phi_\tau(g^{d_1}), \bar{\phi}_\tau(g^{d_1}), \ldots, \phi_\tau(g^{d_k}), \bar{\phi}_\tau(g^{d_k})\big) \cdot \boldsymbol{\mu} = \begin{cases} g^{\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell \rangle} \cdot \nu, & \tau = i; \\ 0, & \text{otherwise.} \end{cases}$$

On the other hand, it is not hard to see that there is a $(k(h+1)) \times (2k)$ matrix $\boldsymbol{U}$ such that

$$\big(\phi_\tau(g^{d_1}), \bar{\phi}_\tau(g^{d_1}), \ldots, \phi_\tau(g^{d_k}), \bar{\phi}_\tau(g^{d_k})\big) = \alpha_\tau(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{U}.$$

To see the FOASC (20) gives a $(1, k)$-PIR, it suffices to note that for all $i \in [n]$ and $\ell \in [N]$,

$$\underbrace{g^{\langle \boldsymbol{u}_i, \boldsymbol{w}_\ell \rangle} \nu}_{\omega_\ell^{(i)}} \cdot \boldsymbol{e}_n^{(i)} = \big(\phi_1(0), \phi_2(0), \ldots, \phi_n(0)\big)^\top \cdot \nu = \boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \underbrace{\boldsymbol{U}\boldsymbol{\mu}}_{\boldsymbol{\lambda}_\ell^{(i)}}. \tag{22}$$

By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n, k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|) = k(h \log m + (h+1) m \log m) = \mathcal{L}_r(n).$$

4.4.4. Hermite-like Interpolations over Finite Fields

Recently, Ghasemi, Kopparty and Sudan [42] proposed a new method of combining the Hasse derivatives with the matching vector based PIR protocols [34,37] and obtained a $(1, \kappa_r)$-PIR protocol with communication complexity $\mathcal{L}_{r+1}(n)$, where $\kappa_1 = 2$ and $\kappa_r = k_r$ for all $r \geq 2$. Specifically, for $r = 2$, they got a $(1, 3)$-PIR with communication complexity $\mathcal{L}_3(n)$, which is more efficient than Efremenko [37], the best $(1, 3)$-PIR previously.

Their construction was obtained with Hermite-like interpolations with Hasse derivatives over a finite field.

The Decoding Problem in Efremenko [37]. Let $m = p_1 p_2 \cdots p_r$ be the product of r distinct primes $p_1, p_2, \ldots, p_r$. Let p be a prime/prime power such that $\gcd(p, m) = 1$ and $m|(p-1)$. Let $H_m \subseteq \mathbb{F}_p^*$ be the group of mth roots of unity and let g be a generator of H_m. Ghasemi, Kopparty and Sudan [42] observed that the decoding problem in Efremenko [37] is nothing else but the problem of interpolating a polynomial of the form

$$\varphi_S(\theta) = \sum_{\delta \in S} \varphi_\delta \theta^\delta$$

with evaluations of $\varphi_S(\theta)$ on a set $B = \{b_1, \ldots, b_k\} \subseteq H_m$, where S is a subset of $\mathbb{Z}_m$.

0-Interpolation Set. Let $m' = mp$ and let $\phi : \mathbb{Z}_m \times \mathbb{Z}_p \to \mathbb{Z}_{m'}$ be the Chinese remainder isomorphism, i.e., $\phi^{-1}(a) = (a \bmod m, a \bmod p)$. Suppose that $S \subseteq \mathbb{Z}_m, S' \subseteq \mathbb{Z}_{m'}$ and $e \in \{1, 2, \ldots, p\}$ is an integer such that $S' \subseteq \phi(S \times \{0, 1, \ldots, e-1\})$. For any multivariate polynomial $F(\boldsymbol{z}) \in \mathbb{F}_p[\boldsymbol{z}] = \mathbb{F}_p[z_1, \ldots, z_h]$ and any nonnegative integer vector $\boldsymbol{i} = (i_1, \ldots, i_h)$, the $\boldsymbol{i}$th *Hasse derivative* $F^{(\boldsymbol{i})}(\boldsymbol{z})$ is the coefficient of $\boldsymbol{y}^{\boldsymbol{i}}$ in the expansion of $F(\boldsymbol{z} + \boldsymbol{y})$, i.e.,

$$F(\boldsymbol{z} + \boldsymbol{y}) = \sum_{\boldsymbol{i}} F^{(\boldsymbol{i})}(\boldsymbol{z}) \boldsymbol{y}^{\boldsymbol{i}}.$$

For any integer $e \geq 1$, let $F^{(<e)}(\boldsymbol{z})$ be the vector of $\boldsymbol{i}$th Hasse derivatives $F^{(\boldsymbol{i})}(\boldsymbol{z})$ for all $\boldsymbol{i} = (i_1, \ldots, i_h)$ that satisfies $i_1 + \cdots + i_h < e$. Ghasemi, Kopparty and Sudan [42] showed that if B is a *0-interpolation set for* S, then B is a *0-interpolation set of multiplicity* e *for* S'. In other words, if $\varphi_S(0)$ is a linear combination of $\{\varphi_S(b)\}_{b \in B}$, then $\varphi_{S'}(0)$ is a linear combination of $\{\varphi_{S'}^{(<e)}(b)\}_{b \in B}$. This critical observation allows them to use a matching family in $\mathbb{Z}_{m'}^h$ to construct PIR but use an $S_{m'}$-decoding polynomial that is as sparse as an S_m-decoding polynomial to reconstruct, and thus reduce the number of servers required by the resulting PIR.

The FOASC and Database Representation. Let $S_m \subseteq \mathbb{Z}_m, S_{m'} \subseteq \mathbb{Z}_{m'}$ be the canonical sets of m and m'. Let $\bar{S}_m = S_m \cup \{0\}$ and $\bar{S}_{m'} = S_{m'} \cup \{0\}$. Then $\bar{S}_{m'} = \bar{S}_m \times \{0, 1\}$. By [42], if $B = \{b_1, \ldots, b_k\} \subseteq H_m$ is a 0-interpolation set for $\bar{S}_m$, then B is a 0-interpolation set of multiplicity $e = 2$ for $\bar{S}_{m'}$. Chee, Feng, Ling, Wang and Zhang [39] and Dvir and Gopi [40] showed that $\bar{S}_m$ has a 0-interpolation set $B = \{b_1, \ldots, b_k\} \subseteq H_m$ of size $k = \kappa_r$. Therefore, $\bar{S}_{m'}$ has a 0-interpolation set (i.e., B) of multiplicity $e = 2$ and size κ_r. Let $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n\}, \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\} \subseteq \mathbb{Z}_{m'}^h$ be an $S_{m'}$-matching family of size $n = \exp(O((\log h)^{r+1}/(\log\log h)^r))$. Let $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_N$ be the elements of H_m^h, where $N = m^h$. Within our framework, underlying [42] is an $\mathsf{FOASC}(N, k, s, t; \boldsymbol{\alpha})$ that consists of n $\mathsf{OA}(N, k, s, t)$'s $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(n)}$, where $N = m^h, k = \kappa_r, s = m^h, t = 1$, and

$$Q_{\ell,j}^{(i)} = \boldsymbol{g}_\ell \cdot (b_j)^{\boldsymbol{v}_i} \tag{23}$$

for all $i \in [n]$, $\ell \in [N]$ and $j \in [k]$. The function $F_{\boldsymbol{x}}(\boldsymbol{z})$ has domain $\mathbb{S} = H_m^h$ and range $\mathbb{R} = \mathbb{F}_p$ such that for all $\tau \in [n]$ and $\boldsymbol{z} \in \mathbb{S}$,

$$\alpha_\tau(\boldsymbol{z}) = (\boldsymbol{z}^{\boldsymbol{u}_\tau})^{(<e)}. \tag{24}$$

Hermite Interpolation on with Hasse Derivatives. Consider any polynomial of the form

$$\varphi(\theta) = \varphi_0 + \sum_{\delta \in S_{m'}} \varphi_\delta \theta^\delta \in \mathbb{R}[\theta].$$

If we denote $\bar{\varphi}(\theta) = \varphi^{(1)}(\theta)$, then there is a column vector $\boldsymbol{\mu} \in \mathbb{R}^{2k}$ such that

$$\varphi_0 = \big(\varphi(b_1), \bar{\varphi}(b_1), \ldots, \varphi(b_k), \bar{\varphi}(b_k)\big) \cdot \boldsymbol{\mu}.$$

The Reconstruction Coefficients. For every $\tau \in [n]$, consider the univariate polynomial

$$\phi_\tau(\theta) = (\boldsymbol{g}_\ell)^{\boldsymbol{u}_\tau} \cdot \theta^{\langle \boldsymbol{u}_\tau, \boldsymbol{v}_i \rangle}.$$

Note that the constant term of this function is $(\boldsymbol{g}_\ell)^{\boldsymbol{u}_i} \cdot 1_{\tau = i}$. Therefore,

$$\big(\phi_\tau(b_1), \bar{\phi}_\tau(b_1), \ldots, \phi_\tau(b_k), \bar{\phi}_\tau(b_k)\big) \cdot \boldsymbol{\mu} = \begin{cases} (\boldsymbol{g}_\ell)^{\boldsymbol{u}_i}, & \tau = i; \\ 0, & \text{otherwise.} \end{cases}$$

On the other hand, it is not hard to see that there is a $(k(h+1)) \times (2k)$ matrix $\boldsymbol{U}$ such that

$$\big(\phi_\tau(b_1), \bar{\phi}_\tau(b_1), \ldots, \phi_\tau(b_k), \bar{\phi}_\tau(b_k)\big) = \alpha_\tau(\boldsymbol{Q}_\ell^{(i)}) \cdot \boldsymbol{U}.$$

To see the FOASC (23) gives a $(1,k)$-PIR, it suffices to note that for all $i \in [n]$ and $\ell \in [N]$,

$$\boldsymbol{e}_n^{(i)} = \big(\phi_1(0), \phi_2(0), \ldots, \phi_n(0)\big)^\top \cdot (\boldsymbol{g}_\ell)^{-\boldsymbol{u}_i} = \boldsymbol{\alpha}(\boldsymbol{Q}_\ell^{(i)}) \cdot \underbrace{(\boldsymbol{g}_\ell)^{-u_i} \cdot \boldsymbol{U}\boldsymbol{\mu}}_{\boldsymbol{\lambda}_\ell^{(i)}}. \tag{25}$$

By Theorem 1, the communication complexity of the protocol is

$$\mathbf{C}(n,k) = k(\log|\mathbb{S}| + \log|\mathbb{R}|) = k(h\log m + \log p) = \mathcal{L}_{r+1}(n).$$

5. Open Problems

The framework of Section 3 gives a unified method of constructing information-theoretic (t,k)-PIR protocols that can capture the most influential constructions to date. Given the state of the art of IT-PIR, there are several interesting directions for future research.

5.1. FOASCs for Constructing (t,k)-PIR with $t = 1$

The best known constructions of (t,k)-PIR for $t = 1$ are due to Dvir and Gopi [40] for $k \leq 26$ and due to Ghasemi, Kopparty and Sudan [42] for all $k > 26$. These constructions require a composite modulus m and depend on two critical ingredients: the superpolynomial sized S_m-matching families from Grolmusz [103] and the sparse S_m-decoding polynomials from Chee, Feng, Ling, Wang and Zhang [39]. Given a composite modulus $m = p_1p_2\cdots p_r$ with r prime factors, the communication complexity of the resulting IT-PIR can be as low as $\mathcal{L}_r(n)$ or $\mathcal{L}_{r+1}(n)$. However, there is still a big gap between the communication complexity of these protocols and the well-known lower bounds [105], which show that $\mathbf{C}_{\mathcal{P}}(n,k) \geq \Omega(k^2/(k-1)\cdot\log n)$ for any k-server IT-PIR. New improved constructions of $\mathsf{FOASC}(N,k,s,1;\boldsymbol{\alpha})$ may help close the gaps by giving protocols with lower communication complexity. A natural idea of developing better FOASCs includes constructing larger S_m-matching families [106–108] or much sparser S_m-decoding polynomials. It is an interesting open problem to construct new $\mathsf{FOASC}(N,k,s,1;\boldsymbol{\alpha})$ that may result in $(1,k)$-PIR with communication complexity $o(\mathcal{L}_r(n))$ for $k \leq 26$ and $o(\mathcal{L}_{r+1}(n))$ for $k > 26$.

5.2. FOASCs for Constructing (t,k)-PIR with $t > 1$

The best known constructions of (t,k)-PIR for $t > 1$ are due to Woodruff and Yekhanin [33] and achieve a communication complexity of $O(n^{1/\lfloor(2k-1)/t\rfloor})$, which however is much worse than the matching vector based $(1,k)$-PIR protocols with subpolynomial communication. Barkol, Ishai and Weinreb [109] proposed a general transformation from $(1,k)$-PIR to (t,k^t)-PIR that preserves the asymptotic communication complexity. By applying this transformation to the matching vector based $(1,k)$-PIR one can obtain (t,k^t)-PIR with subpolynomial communication for any $t > 1$. However, such a transformation results in an exponential blowup in the number of required servers. In particular, for a general number k' that is not a tth power of some integer, it could be very inefficient or even impossible to use such a transformation to construct a (t,k')-PIR. It is an interesting open problem to construct new $\mathsf{FOASC}(N,k,s,t;\boldsymbol{\alpha})$ that may result in (t,k)-PIR with communication complexity $o(n^{1/\lfloor(2k-1)/t\rfloor})$ for constant t and k.

6. Conclusions

In this review, we formally define families of orthogonal arrays with span capability (FOASC) and provide a unified framework for constructing multi-server IT-PIR protocols. We show how to capture the most influential IT-PIR protocols with the proposed framework. We also put forward several interesting open problems concerning the construction of FOASCs. With the proposed framework, we expect to inspire new FOASCs and thus more efficient IT-PIR protocols with communication complexity approaching the best known lower bounds.

Funding

This research is partially supported by the National Natural Science Foundation of China (grant No. 62372299) and the Open Project Funding of the Key Laboratory of Cyberspace Security Defense (grant No. 2024-MS-13).

Data Availability Statement

Not applicable.

Conflicts of Interest

The author declares no conflict of interest. Given the role as an editorial board member of the journal, Liangfeng Zhang had no involvement in the peer review of this paper and had no access to information regarding its peer-review process. Full responsibility for the editorial process of this paper was delegated to another editor of the journal.

Use of AI and AI-Assisted Technologies

No AI tools were utilized for this paper.

References

1. Chor, B.; Kushilevitz, E.; Goldreich, O.; et al. Private information retrieval. In Proceedings of the FOCS 1995, Milwaukee, WI, USA, 23–25 October 1995; pp. 41–50.
2. Kushilevitz, E.; Ostrovsky, R. Replication is not needed: Single database, computationally-private information retrieval. In Proceedings of the 38th Annual Symposium on Foundations of Computer Science, Miami Beach, FL, USA, 20–22 October 1997; pp. 364–373.
3. Wang, F.; Yun, C.; Goldwasser, S.; et al. Splinter: Practical private queries on public data. In Proceedings of the 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17), Boston, MA, USA, 27–29 March 2017; pp. 299–313.
4. Angel, S.; Setty, S. Unobservable communication over fully untrusted infrastructure.In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 551–569.
5. Angel, S.; Chen, H.; Laine, K.; et al. PIR with compressed queries and amortized query processing. In Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 21–23 May 2018; pp. 962–979.
6. Gupta, T.; Crooks, N.; Mulhern, W.; et al. Scalable and private media consumption with Popcorn. In Proceedings of the 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16), Santa Clara, CA, USA, 16–18 March 2016; pp. 91–107.
7. Kales, D.; Rechberger, C.; Schneider, T.; et al. Mobile private contact discovery at scale. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1447–1464.
8. Kogan, D.; Corrigan-Gibbs, H. Private blocklist lookups with Checklist. In Proceedings of the 30th USENIX Security Symposium (USENIX Security 21), Vancouver, BC, Canada, 11–13 August 2021; pp. 875–892.
9. Backes, M.; Kate, A.; Maffei, M.; et al. ObliviAd: Provably secure and practical online behavioral advertising. In Proceedings of the 2012 IEEE Symposium on Security and Privacy, San Francisco, CA USA, 20–23 May 2012; pp. 257–271.
10. Servan-Schreiber, S.; Hogan, K.; Devadas, S. AdVeil: A private targeted–advertising ecosystem. *IACR Cryptol. Eprint Arch.* **2021**, *2021*, 1032.
11. Henzinger, A.; Hong, M.M.; Corrigan-Gibbs, H.; et al. One server for the price of two: Simple and fast single-server private information retrieval. In Proceedings of the 32nd USENIX Security Symposium (USENIX Security 23), Anaheim, CA, USA, 9–11 August 2023; pp. 3889–3905.
12. Henzinger, A.; Dauterman, E.; Corrigan-Gibbs, H.; et al. Private web search with Tiptoe. In Proceedings of the 29th Symposium on Operating Systems Principles, Koblenz, Germany, 23–26 October 2023; pp. 396–416.
13. Henry, R.; Olumofin, F.; Goldberg, I. Practical PIR for electronic commerce. In Proceedings of the 18th ACM Conference on Computer and Communications Security, Chicago, IL, USA, October 17–21 2011; pp. 677–690.
14. Ghinita, G.; Kalnis, P.; Khoshgozaran, A.; Shahabi, C.; et al. Private queries in location based services: Anonymizers are not necessary. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008; pp. 121–132.
15. Chen, H.; Huang, Z.; Laine, K.; et al. Labeled PSI from fully homomorphic encryption with malicious security. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1223–1237.
16. Google. Available online: https://services.google.com/fh/files/misc/zero_touch_enrollment_for_chrome_os_devices_one_pager.pdf (accessed on 11 August 2025).
17. Menon, S.J.; Wu, D.J. SPIRAL: Fast, high-rate single-server PIR via FHE composition. In Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 23–26 May 2022; pp. 930–947.
18. Davidson, A.; Pestana, G.; Celi, S. FrodoPIR: Simple, scalable, singleserver private information retrieval. *Proc. Priv. Enhancing Technol.* **2023**, *2023*, 365–383.

19. Katz, J.; Trevisan, L. On the efficiency of local decoding procedures for error-correcting codes. In Proceedings of the Thirty-Second Annual ACM Aymposium on Theory of Computing Portland, OR, USA, 21–23 May 2000; pp. 80–86.
20. Yekhanin, S. Locally decodable codes. *Found. Trends Theor. Comput. Sci.* **2012**, *6*, 139–255.
21. Zhang, L.F. A coding-theoretic application of Baranyai's theorem. *IEEE Trans. Inf. Theory* **2014**, *60*, 6988–6992.
22. shai, Y.; Kushilevitz, E.. On the hardness of information-theoretic multiparty computation. In Proceedings of the EUROCRYPT, Interlaken, Switzerland, 2–6 May 2004; pp. 439–455.
23. Beimel, A.; Chee, Y.M.; Wang, H.; et al. Communication-efficient distributed oblivious transfer. *J. Comput. Syst. Sci.* **2012**, *78*, 1142–1157.
24. Beimel, A.; Ishai, Y.; Kushilevitz, E.; et al. One-way functions are essential for single-server private information retrieval. In Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, Atlanta, GA, USA, 1–4 May 1999; pp. 89–98.
25. Di Crescenzo, G.; Malkin, T.; Ostrovsky, R. Single database private information retrieval implies oblivious transfer. In Proceedings of the EUROCRYPT 2000, Bruges, Belgium, 14–18 May 2000; pp.122–138.
26. Naor, M.; Pinkas, B. Oblivious transfer and polynomial evaluation. In Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, Dallas, TX, USA, 23–26 May 1999; pp. 245–254.
27. Ishai, Y.; Kushilevitz, E.; Ostrovsky, R. Sufficient conditions for collision-resistant hashing. In Proceedings of the TCC, Cambridge, MA, USA, 10–12 February 2005; pp. 445–456.
28. Kalai, Y.T.; Raz, R. Succinct non-interactive zero-knowledge proofs with preprocessing for LOGSNP. In Proceedings of the 2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06), Berkeley, CA, USA, 21–24 October 2006; pp. 355–366.
29. Ambainis, A. Upper bound on communication complexity of private information retrieval. In Proceedings of the International Colloquium on Automata, Languages, and Programming (ICALP), Bologna, Italy, 7–11 July 1997; pp. 401–407.
30. Itoh, T. Efficient private information retrieval. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci. E82-A(1)* **1999**, *82*, 11–20.
31. Ishai, Y.; Kushilevitz, E. Improved Upper Bounds on Information-Theoretic Private Information Retrieval. In Proceedings of the thirty-first annual ACM symposium on Theory of computing (STOC), Atlanta, GA, USA, 1–4 May 1999; pp. 79–88.
32. Beimel, A.; Ishai, Y. Information-theoretic private information retrieval: A unified construction. In Proceedings of the ICALP, Crete, Greece, 8–12 July 2001; pp. 912–926.
33. Woodruff, D.; Yekhanin, S. A geometric approach to information-theoretic private information retrieval. In Proceedings of the 20th Annual IEEE Conference on Computational Complexity (CCC), San Jose, CA, USA, 11–15 June 2005; pp. 275–284.
34. Yekhanin, S. Towards 3-query locally decodable codes of subexponential length. In Proceedings of the STOC, San Diego, CA, USA, 11–13 June 2007; pp. 266–274.
35. Raghavendra, P. A note on Yekhanin's locally decodable codes. In Proceedings of the Electronic Colloquium on Computational Complexity TR07, Weizmann Institute of Science, Rehovot, Israel, 15 January 2007.
36. Kedlaya, K.S.; Yekhanin, S. Locally decodable codes from nice subsets of finite fields and prime factors of Mersenne numbers. In Proceedings of the CCC, College Park, MD, USA, 23–26 June 2008; pp. 175–186.
37. Efremenko, K. 3-query locally decodable codes of subexponential length. In Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing (STOC), Bethesda, MD, USA, 31 May–2 June 2009; pp. 39–44.
38. Itoh, T.; Suzuki, Y. Improved constructions for query-efficient locally decodable codes of subexponential length. *IEICE Trans. Inf. Syst.* **2010**, *93*, 263–270.
39. Chee, Y.M.; Feng, T.; Ling, S.; et al. Query-efficient locally decodable codes of subexponential length. *Comput. Complex.* **2013**, *22*, 159–189.
40. Dvir, Z.; Gopi, S. 2-Server PIR with sub-polynomial communication. In Proceedings of the STOC, Portland, OR, USA, 14–17 June 2015; pp.577–584.
41. Alon, B.; Beimel, A.; Lasri, O. Simplified PIR and CDS Protocols and improved linear secret-sharing protocols. *IACR Cryptol. Eprint Arch.* **2024**, *2024*, 1599.
42. Ghasemi, F.; Kopparty, S.; Sudan, M. Improved PIR protocols using matching vectors and derivatives. In Proceedings of the STOC, Prague, Czech Republic, 23–27 June 2025; pp. 1648–1656.
43. Cachin, C.; Micali, S.; Stadler, M. Computationally private information retrieval with polylogarithmic communication. In Proceedings of the EUROCRYPT, Prague, Czech Republic, 2–6 May 1999; pp. 402–414.
44. Chang, Y.-C. Single database private information retrieval with logarithmic communication. In Proceedings of the ACISP, Sydney, Australia, 13–15 July 2004; pp. 50–61.
45. Lipmaa, H. An oblivious transfer protocol with log-squared communication. In Proceedings of the ISC, Singapore, 20–23 September 2005; pp.314–328.
46. Gentry,C.; Ramzan, Z. Single-database private information retrieval with constant communication rate. In Proceedings of the ICALP 2005, Lisbon, Portugal, 11–15 July 2005; pp. 803–815.
47. Melchor, C.A.; Gaborit, P. A lattice-based computationally-efficient private information retrieval protocol. *IACR Cryptol.*

Eprint Arch. **2007**, *2007*, 446.
48. Lipmaa, H. First CPIR protocol with data-dependent computation. In Proceedings of the ICISC 2009, Seoul, Korea, 2–4 December 2009; pp. 193–210.
49. Groth, J.; Kiayias, A.; Lipmaa, H. Multi-query computationally-private information retrieval with constant communication rate. In Proceedings of the Public Key Cryptography 2010, Paris, France, 26–28 May 2010; pp. 107–123.
50. Yi X.; Kaosar, M.G.; Paulet, R.; et al. Single-database private information retrieval from fully homomorphic encryption. *IEEE Trans. Knowl. Data Eng.* **2013**, *25*, 1125–1134.
51. Dong,C.; Chen, L. A fast single server private information retrieval protocol with low communication cost. In Proceedings of the ESORICS (1) 2014, Wroclaw, Poland, 7–11 September 2014; pp. 380–399
52. Kiayias, A.; Leonardos, N.; Lipmaa, H.; et al. Optimal rate private information retrieval from homomorphic encryption. In Proceedings of the Privacy Enhancing Technologies, Philadelphia, PA, USA, 30 June–2 July 2015; pp. 222–243.
53. Lipmaa,H.; Pavlyk, K. A simpler rate-optimal CPIR protocol. In Proceedings of the Financial Cryptography 2017, Sliema, Malta, 3–7 April 2017; pp. 621–638
54. Beimel, A.; Ishai,Y.; Malkin, T. Reducing the servers computation in private information retrieval: PIR with preprocessing. In Proceedings of the CRYPTO 2000, Santa Barbara, CA, USA, 20–24 August 2000; pp. 55–73.
55. Sion, R.; Carbunar, B. On the practicality of private information retrieval. In Proceedings of the NDSS 2007, San Diego, CA, USA, 26 February–4 March 2007.
56. Singh, J.; Wei,Y.; Zikas, V. Information-theoretic multi-server private information retrieval with client preprocessing. In Proceedings of the TCC (4) 2024, Milan, Italy, 2–6 December 2024; pp. 423–450.
57. Ishai, Y.; Shi, E.; Wichs, D. PIR with client-side preprocessing: Information-theoretic constructions and lower bounds. In Proceedings of the CRYPTO (9) 2024, Santa Barbara, CA, USA, 18–22 August 2024; pp. 148–182.
58. Boyle, E.; Ishai, Y.; Pass, R.; et al. Can we access a database both locally and privately? In Proceedings of the TCC (2) 2017, Baltimore, MD, USA, 12–15 November 2017; pp. 662–693.
59. Canetti, R.; Holmgren, J.; Richelson, S. Towards doubly efficient private information retrieval. In Proceedings of the TCC (2) 2017, Baltimore, MD, USA, 12–15 November 2017; pp. 694–726.
60. Corrigan-Gibbs,H.; Kogan, D. Private information retrieval with sublinear online time. In Proceedings of the EUROCRYPT (1) 2020, Zagreb, Croatia, 10–14 May 2020; pp. 44–75.
61. Ghoshal, A.; Zhou, M.; Shi, E. Efficient pre-processing PIR without public-key cryptography. In Proceedings of the EUROCRYPT (6) 2024, Zurich, Switzerland, 26–30 May 2024; pp. 210–240.
62. Wei–Kai Lin, W.-K.; Mook, E.; Wichs, D. Doubly efficient private information retrieval and fully homomorphic ram computation from ring LWE. In Proceedings of the STOC 2023, Orlando, FL, USA, 20–23 June 2023; pp. 595–608.
63. Ren, L.; Mughees,M.H.; Sun, I. Simple and practical amortized sublinear private information retrieval using dummy subsets. In Proceedings of the CCS 2024, V Vancouver, BC, USA, 24–27 October 2024; pp. 1420–1433.
64. Patel,S.; Persiano, G.; Yeo, K. Private stateful information retrieval. In Proceedings of the CCS 2018, Toronto, ON, Canada, 15–19 October 2018; pp. 1002–1019.
65. Shi, E.; Aqeel, W.; rasekaran, B.;et al. Puncturable pseudorandom sets and private information retrieval with near-optimal online bandwidth and time. In Proceedings of the 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, 16–20 August 2021; pp. 641–669.
66. Zhou, M.; Park, A.; Zheng, W.; et al. Piano: Extremely simple, single-server PIR with sublinear server computation. In Proceedings of the IEEE Symposium on Security and Privacy 2024, San Francisco, CA, USA, 19–23 May 2024; pp. 4296–4314.
67. Ali, A.; Lepoint, T.; Patel, S.;et al. Communication-computation trade-offs in PIR. In Proceedings of the USENIX Security Symposium 2021, Vancouver, BC, Canada, 11–13 August 2021; pp. 1811–1828.
68. Mughees, M.H.; Chen, H.; Ren, L. OnionPIR: Response efficient single-server PIR. In Proceedings of the CCS 2021, New York, NY, USA, 15–19 November 2021; pp. 2292–2306.
69. Mahdavi, R.A.; Kerschbaum, F. Constant-weight PIR: Single-round keyword PIR via constant-weight equality operators. In Proceedings of the USENIX Security Symposium 2022, Boston, MA, USA, 10–12 August 2022; pp. 1723–1740.
70. Corrigan-Gibbs, H.; Henzinger, A.; Kogan, D. Single-server private information retrieval with sublinear amortized time. In Proceedings of the EUROCRYPT (2) 2022, Trondheim, Norway, 30 May–3 June 2022; pp. 3–33.
71. Zhou, M.; Lin, W.-K.; Tselekounis, Y.; et al. Optimal single-server private information retrieval. In Proceedings of the EUROCRYPT (1) 2023, Lyon, France, 23–27 April 2023; pp. 395–425.
72. Lazzaretti,A.; Papamanthou, C. TreePIR: Sublinear-time and polylog-bandwidth private information retrieval from DDH. In Proceedings of the CRYPTO (2) 2023, Santa Barbara, CA, USA, 19–24 August 2023; pp. 284–314.
73. Luo, M.; Liu, F.-H.; Wang, H. Faster FHE-based single-server private information retrieval. In Proceedings of the CCS 2024, Vancouver, BC, Canada, 24–27 October 2024; pp. 1405–1419.
74. Li, B.; Micciancio, D.; Raykova, M.; et al. Hintless single-server private information retrieval. In Proceedings of 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, 18–22 August 2024 ; pp. 183–217.
75. Lazzaretti, A.; Papamanthou, C. Single pass client-preprocessing private information retrieval. In Proceedings of the

USENIX Security Symposium 2024, Philadelphia, PA, USA, 14–16 August 2024; pp. 5967–5984.

76. Fisch, B.; Lazzaretti, A.; Liu,Z.; et al. ThorPIR: Single server PIR via homomorphic thorp shuffles. In Proceedings of the CCS 2024, Vancouver, BC, Canada, 24–27 October 2024; pp. 1448–1462.
77. Menon,S.J.; Wu, D.J. YPIR: High-throughput single-server PIR with silent preprocessing. In Proceedings of the USENIX Security Symposium 2024, Philadelphia, PA, USA, 14–16 August 2024; pp. 5985–6002.
78. Hoover, A.; Patel,S.; Persiano, G.; et al. Plinko: Single-server PIR with efficient updates via invertible PRFs. In Proceedings of the EUROCRYPT (6) 2025, Madrid, Spain, 3–4 May 2025; pp. 3–33.
79. Wang, Z.; Ren, L. Single-server client preprocessing PIR with tight space-time trade-off. In Proceedings of the EUROCRYPT (6) 2025, Madrid, Spain, 3–4 May 2025; pp. 94–122.
80. Amos Beimel, A.; Yoav Stahl, Y. Robust information-theoretic private information retrieval. In Proceedings of the SCN 2002, Amalfi, Italy, 11–13 September 2002; pp. 326–341.
81. Goldberg, I. Improving the robustness of private information retrieval. In Proceedings of the IEEE Symposium on Security and Privacy 2007, Oakland, CA, USA, 20–23 May 2007; pp. 131–148.
82. Devet,C.; Goldberg,I.; Heninger, N. Optimally robust private information retrieval. In Proceedings of the USENIX Security Symposium 2012, Bellevue, WA, USA, 8–10 August 2012; pp. 269–283.
83. Kurosawa, K. How to correct errors in multi-server PIR. In Proceedings of the ASIACRYPT (2) 2019, Kobe, Japan, 8–12 December 2019; pp. 564–574.
84. Zhang, L.F.; Wang,H.; Wang, L.-P. Byzantine-robust private information retrieval with low communication and efficient decoding. In Proceedings of the AsiaCCS 2022, Nagasaki, Japan, 30 May–3 June 2022; pp. 1079–1085.
85. Eriguchi, R.; Kurosawa, K.; Nuida, K. On the optimal communication complexity of error-correcting multi-server PIR. In Proceedings of the TCC (3) 2022, Chicago, IL, USA, 7–10 November 2022, 60–88.
86. Eriguchi, R.; Kurosawa, K.; Nuida, K. Efficient and generic methods to achieve active security in private information retrieval and more advanced database search. In Proceedings of the EUROCRYPT (5) 2024, Zurich, Switzerland, 26–30 May 2024; pp. 92–121.
87. Zhang, L.F; Safavi-Naini, R. Verifiable multi-server private information retrieval. In Proceedings of the ACNS 2014, Lausanne, Switzerland, 10–13 June 2014; pp. 62–79.
88. Zhu, L.; Lin, C.; Lin, F.; et al. Postquantum cheating detectable private information retrieval. In Proceedings of the SEC 2022, Boston, MA, USA, 10–12 August 2022; pp. 431–448.
89. Cao, Q.; Tran, H.Y.; Dau, S.H.; et al. Committed private information retrieval. In Proceedings of the ESORICS (1) 2023, The Hague, The Netherlands, 25–29 September 2023; pp. 393–413.
90. Zhang, L.F.; Wang, H. Multi-server verifiable computation of low-degree polynomials. In Proceedings of the IEEE Symposium on Security and Privacy 2022, San Francisco, CA, USA, 23–25 May 2022; pp. 596–613.
91. Ke, P.; Zhang, L.F. Two-server private information retrieval with result verification. In Proceedings of the ISIT 2022, Espoo, Finland, 26 June–1 July 2022; pp. 408–413.
92. Ke, P.; Zhang, L.F. Private information retrieval with result verification for more servers. In Proceedings of the ACNS 2023, Kyoto, Japan, 19–22 June 2023; pp. 197–216.
93. Kruglik, S.; Dau, S.H.; Kiah, H.M.; et al. Querying twice to achieve information-theoretic verifiability in private information retrieval. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 8172–8187.
94. Eriguchi, R.; Kurosawa, K.; Nuida, K. Multi-server PIR with full error detection and limited error correction. In Proceedings of the ITC 2022, Cambridge, MA, USA, 23–30 September 2022.
95. Colombo, S.; Nikitin, K.; Corrigan-Gibbs, H.; et al. Authenticated private information retrieval. In Proceedings of the USENIX Security Symposium 2023, Anaheim, CA, USA, 9–11 August 2023; pp. 3835–3851.
96. Dietz, M.; Tessaro, S. Fully malicious authenticated PIR. In Proceedings of the 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, 18–22 August 2024; pp. 113–147
97. Gasarch, W.I. A survey on private information retrieval. *Bull. EATCS* **2004**, *82*, 72–107.
98. Beimel, A. Private Information Retrieval: A primer. Available online: www.cs.bgu.ac.il /beimel/ Papers/PIRsurvey.ps (accessed on 11 August 2025).
99. Blundo, C. *Private Information Retrieval, Encyclopedia of Cryptography and Security*; Springer Nature: Cham, Switzerland, 2011; pp. 974–976.
100. Ostrovsky, R.; Skeith, W.E., III A survey of single-database private information retrieval: Techniques and Applications. *Public Key Cryptogr.* **2007**, 393–411.
101. Hedayat, A.S.; Sloane, N.J.A.; Stufken, J. *Orthogonal Arrays: Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 1999.
102. Beimel, A.; Ishai,Y.; Kushilevitz, E.; et al. Breaking the $O(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval. In Proceedings of the FOCS 2002, Vancouver, BC, Canada, 16–19 November 2002; pp. 61–270.
103. Grolmusz, V. Superpolynomial size set-systems with restricted intersections mod 6 and explicit Ramsey graphs. *Combinatorica* **2000**, *20*, 71–86
104. Zhu, L.; Li,W.M.; Zhang, L.F. On the modulus in matching vector codes. *Comput. J.* **2022**, *65*, 2991–2997.

105. Mann, E. Private Access to Distributed Information. Master's Thesis, Technion-Israel Institute of Technology, Haifa, Israel, 1998
106. Dvir, Z.; Gopalan, P.; Yekhanin, S.; Matching vector codes. In Proceedings of the FOCS 2010, Las Vegas, NV, USA, 23–26 October 2010; pp. 705–714
107. Bhowmick, A.; Dvir, Z.; Lovett, S. New bounds for matching vector families. In Proceedings of the STOC 2013, Palo Alto, CA, USA, 2–4 June 2013; pp. 823–832
108. Chee, Y.M.; Ling, S.; Wang,H.; et al. Upper bounds on matching families in $\mathbb{Z}_{pq}^n$. *IEEE Trans. Inf. Theory* **2013**, *59*, 5131–5139.
109. Barkol, O.; Ishai,Y.; Weinreb, E. On locally decodable codes, self-correctable codes, and t-private PIR. In Proceedings of the APPROX-RANDOM 2007, Princeton, NJ, USA, 20–22 August 2007; pp. 311–325.