Improved Storage for Efficient Private Information Retrieval

Karim Banawan[1], Batuhan Arasli[2], and Sennur Ulukus[2]

[1]Electrical Engineering Department, Faculty of Engineering, Alexandria University, Alexandria, Egypt
[2]Department of Electrical and Computer Engineering, University of Maryland, College Park, MD, USA

Abstract—We consider the problem of private information retrieval from N *storage-constrained* databases. In this problem, a user wishes to retrieve a single message out of M messages (of size L) without revealing any information about the identity of the message to individual databases. Each database stores μML symbols, i.e., a μ fraction of the entire library, where $\frac{1}{N} \leq \mu \leq 1$. Our goal is to characterize the optimal tradeoff curve for the storage cost (captured by μ) and the normalized download cost (D/L). We show that the download cost can be reduced by employing a hybrid storage scheme that combines *MDS coding* ideas with *uncoded partial replication* ideas. When there is no coding, our scheme reduces to Attia-Kumar-Tandon storage scheme, which was initially introduced by Maddah-Ali-Niesen in the context of the caching problem, and when there is no uncoded partial replication, our scheme reduces to Banawan-Ulukus storage scheme; in general, our scheme outperforms both.

I. INTRODUCTION

Private information retrieval (PIR), which was introduced by Chor et al. in [1], is a canonical problem to investigate the privacy issues that arise upon interaction with open-access databases. In classical PIR, there is a user, who needs to retrieve a message (file) out of M messages from N distributed *content-replicating* and non-colluding databases privately, i.e., in a way that the identity of the desired message is kept secret from any individual database. A direct, yet inefficient, scheme to satisfy this privacy requirement is to download the contents of all databases. The download cost, in this case, scales linearly with M. Although the PIR problem was introduced in the computer science community [2], [3], there has been a growing interest in characterizing the fundamental limits of the problem among information theorists with notable early examples [4]–[8]. Recently, Sun and Jafar have introduced the concept of PIR capacity C, which is the supremum of the ratio between the message size and the total download cost [9]. The optimal normalized download cost D^* is the reciprocal of the PIR capacity. Sun-Jafar derived the download cost for the classical PIR model to be $D^* = 1 + \frac{1}{N} + \cdots + \frac{1}{N^{M-1}}$. Many interesting variants of the classical model have been studied in [10]–[38].

In the majority of these works, the messages are replicated across the N databases, such that each database stores ML symbols. By leveraging this replication, the user can exploit the undesired symbols downloaded from one database as

side information to recover more desired symbols from other databases. Although this simplifies the PIR scheme, it results in high storage cost. To minimize the storage cost, several directions have been explored in the literature: [6] proposes a new family of codes for storage called k-server PIR codes. [12] and [26] as well store data in a non-replicated manner; [12] uses an (N, K) MDS code and [26] uses an uncoded partial replication strategy originally introduced in [39]. [37] and [38] further investigate the problem of non-replicated storage via representing storage with graphs, where each database stores full messages but not the entire message set.

The works most closely related to ours are [12] and [26]: [12] characterizes the optimal download cost of PIR from (N, K) *MDS-coded* databases as $D^* = 1 + \frac{K}{N} + \cdots + \left(\frac{K}{N}\right)^{M-1}$. In this work, each message is organized into K-length rows and each row is independently mapped into an N-length vector using an (N, K) MDS code. This effectively minimizes the storage cost as each database stores $\frac{1}{K}$ of the total size of the messages. [26] investigates a setting where each database stores a fraction μ of each message. When the storage strategy is constrained to *uncoded storage*, [26] shows that the uncoded prefetching strategy of [39] along with the PIR scheme of [9] is optimal and the optimal storage-download cost tradeoff is given by the convex hull of the $\left(\frac{t}{N}, 1 + \frac{1}{t} + \cdots + \left(\frac{1}{t}\right)^{M-1}\right)$ pairs for $t = 1, \ldots, N$. This problem is then extended to the decentralized setting in [35], where the contents are stored independently across databases according to a probability distribution, and to the heterogeneous setting in [40] where the databases have heterogeneous storage sizes. In all these works, with the exception of [12], the fundamental limits are derived for uncoded storage strategies.

In this paper, we consider the PIR problem from storage constrained databases. The storage at the databases is constrained such that each database stores a deterministic function of the messages with a total size of μML symbols picked from a finite field $\mathbb{F}_q$, for some fraction μ, $\frac{1}{N} \leq \mu \leq 1$. It is required to design such storage functions for facilitating the most efficient PIR scheme, i.e., we aim at jointly designing the storage strategy and the retrieval scheme such that the normalized download cost is minimized subject to a storage size constraint of μML. The end goal is to characterize the optimal storage-normalized download cost tradeoff $D^*(\mu)$.

To that end, we first present a motivating example, which in-

vestigates known storage strategies for PIR, namely: uncoded storage in [26] and direct MDS-coded storage in [12]. We show that no single storage scheme among these two schemes outperforms the other in all storage ratio regimes. Next, we restrict our coded-storage strategies to non-mixing MDS coding only, i.e., we allow message mapping to coded symbols via an MDS code that neither mixes different messages nor rows of any individual message. For an (N, K) MDS code [12], the storage ratio $\mu = \frac{1}{K}$ (as 1 coded-symbol is stored at each database from every row of the message). Therefore, the normalized download cost is $D = 1 + \frac{1}{N\mu} + \cdots + \left(\frac{1}{N\mu}\right)^{M-1}$, where $K = \frac{1}{\mu} \in \mathbb{Z}_+$. We aim at constructing achievable storage schemes which outperform [26] and [12] by using a mix of MDS coding and uncoded partial replication ideas.

We propose a novel storage strategy that unifies the direct MDS coded storage in [12] with the uncoded prefetching storage in [26]. Using this scheme, the messages are first coded row-by-row via an (N, K) MDS code. The indices of the message rows are partitioned into $\binom{N}{t}$ partitions, where each row partition is stored in a group of t databases. By this storage strategy, we have $\mu = \frac{t}{KN}$. We achieve a normalized download cost of $1 + \frac{K}{t} + \frac{K^2}{t^2} + \cdots + \frac{K^{M-1}}{t^{M-1}}$ for all $t, K \in [N]$ and $t \geq K$. For any other point, we employ memory-sharing. For $K = 1$, all corner points of the uncoded storage in [26] can be recovered. For $t = N$, all corner points of the direct MDS-coded storage in [12] can be attained. By changing t, K, more corner points can be attained. At these corner points, the MDS-coded storage benchmark with rational K can be achieved. We illustrate these facts by a representative example of $M = 2$, $N = 6$, and $\mu = \frac{5}{12}$.

II. System Model

Consider a storage system with N databases; see Fig. 1. A data center aims at storing a message set $\mathcal{W}$ in N distributed databases. The message set contains M i.i.d. messages of size L, picked uniformly from a sufficiently large field size $\mathbb{F}_q^L$, i.e.,

$$H(W_m) = L, \quad \text{(in } q\text{-ary units)} \tag{1}$$

$$H(\mathcal{W}) = H(W_1, \cdots, W_M) = ML \tag{2}$$

Each database has a storage capacity of μML, where $\frac{1}{N} \leq \mu \leq 1$. Each database stores a function of the message set $\mathcal{W}$. Specifically, the nth database stores $Z_n = f_n(\mathcal{W})$, such that,

$$H(Z_n) \leq \mu ML \tag{3}$$

$$H(Z_n|W_1, \cdots, W_M) = 0 \tag{4}$$

In PIR, there is a user who wants to retrieve a file $W_\theta \in \mathcal{W}$, where $\theta \in [M]$, without revealing any information about θ. To that end, the user submits a query $Q_n^{[\theta]}$ to the nth database. All the queries are independent of the messages as the user has no prior information about the message set, i.e.,

$$I(Q_1^{[\theta]}, \cdots, Q_N^{[\theta]}; W_1, \cdots, W_M) = 0 \tag{5}$$

Upon receiving the query $Q_n^{[\theta]}$, the nth database responds with an answer string $A_n^{[\theta]}$, which is a deterministic function
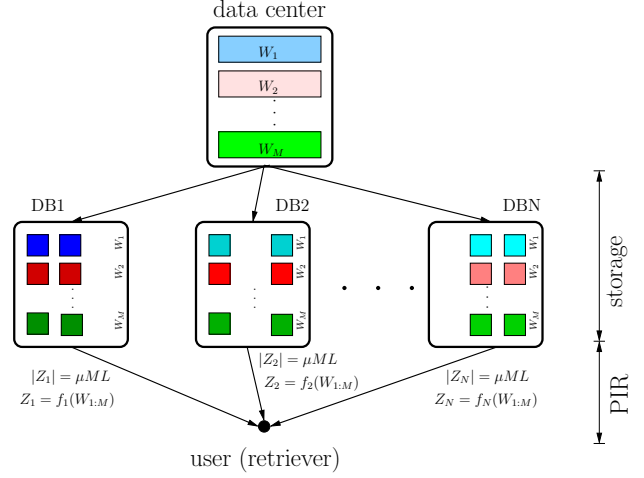

Fig. 1. System model: A data center stores the message set in N distributed databases; a user retrieves a message privately from the databases.

of the query $Q_n^{[\theta]}$ and the storage content of the database Z_n,

$$H(A_n^{[\theta]}|Q_n^{[\theta]}, Z_n) = 0 \tag{6}$$

To protect the privacy, the query submitted to the nth database to retrieve W_θ should be indistinguishable from the query submitted to retrieve $W_{\theta'}$ for all $\theta' \in [M]$, therefore,

$$(Q_n^{[\theta]}, A_n^{[\theta]}, \mathcal{W}) \sim (Q_n^{[\theta']}, A_n^{[\theta']}, \mathcal{W}), \quad \theta, \theta' \in [M] \tag{7}$$

where $\sim$ denotes statistical equivalence.

To ensure reliability, the user should be able to reconstruct the message W_θ using the answer strings $A_{1:N}^{[\theta]}$ with arbitrarily small probability of error, hence,

$$H(W_\theta|Q_{1:N}^{[\theta]}, A_{1:N}^{[\theta]}) = o(L) \tag{8}$$

where $\frac{o(L)}{L} \to 0$ as $L \to \infty$.

An achievable retrieval scheme is a scheme that satisfies (7) and (8) for some message length L and storage functions $f_n(\cdot)$ for $n \in [N]$. We measure the efficiency of a PIR scheme with its normalized download cost $D(\mu)$

$$D(\mu) = \frac{\sum_{n=1}^{N} H(A_n^{[\theta]})}{L} \tag{9}$$

In this work, we aim at jointly designing the storage system, i.e., identifying the storage functions $f_n(\cdot)$, and the retrieval scheme such that the normalized download cost is minimized. Our goal is to identify the optimal tradeoff between storage and download cost, $D^*(\mu) = \min D(\mu)$. We constrain ourselves to *non-mixing* MDS coding based [12] storage policies.

III. Motivating Example

In this section, we motivate our work by giving an example of a storage system with $N = 6$, $M = 2$. We first illustrate the uncoded storage technique in [26]. The storage technique in [26] is based on the uncoded prefetching scheme of [39]. Each message is divided into $\binom{N}{t}$ partitions such that $\mu = \frac{t}{N}$ and $t \in [N]$. For each message partition, [26] employs the PIR scheme in [9] with t databases instead of total N
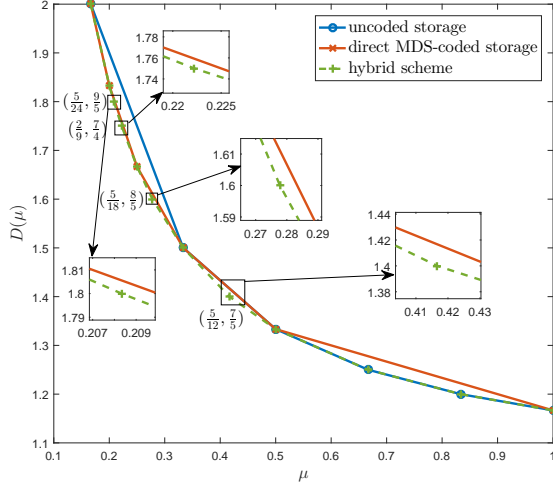
Fig. 2. Optimal tradeoff for uncoded storage [26] and direct MDS coded storage [12] for $N = 6$, $M = 2$. The extra corner points that can be achieved only by the proposed hybrid scheme are shown explicitly on the figure and are magnified for easier visualization.

databases. For any other storage ratio $\mu \neq \frac{t}{N}$ for some $t \in [N]$, [26] uses memory sharing between adjacent tradeoff points that enclose μ. This results in the lower convex hull of the following tradeoff points (see Fig. 2, the blue curve):

$$\left(\frac{t}{6}, 1 + \frac{1}{t} \right), \quad t = 1, 2, \cdots, 6 \quad (10)$$

Second, we illustrate the direct MDS coded storage technique in [12]. In [12], each message is organized into a matrix. Each K-length row is mapped using an (N, K) MDS code to an N-length vector. Each coded symbol is stored in one of the N databases. Hence, $\mu = \frac{1}{K}$, for $K \in [N]$. The PIR scheme in [12] achieves the lower convex hull of the following tradeoff points (see Fig. 2, the red curve):

$$\left(\frac{1}{K}, 1 + \frac{K}{6} \right), \quad K = 1, 2, \cdots, 6 \quad (11)$$

From Fig. 2, we note that at low storage ratios, the MDS coded storage of [12] outperforms the uncoded storage of [26], while the opposite is true for high storage ratios. In this paper, we propose a hybrid scheme that combines MDS coding ideas of [12] with uncoded prefetching storage ideas of [26]. We show that our scheme outperforms both schemes for specific storage ratios. For instance, for $N = 6$, $M = 2$, Fig. 2 shows four new non-trivial corner points achieved by our scheme (see blown up sub-figures). We note that these additional four points are on the curve $D = 1 + \frac{1}{N\mu} + \cdots + \left(\frac{1}{N\mu} \right)^{M-1}$, so are the points already achieved by [12] and [26]. It is unclear at this point if the entirety of this curve can be achieved by general schemes. This will likely require use of general storing strategies and corresponding PIR schemes.

IV. MAIN RESULTS

The main result of this work is a novel hybrid storage scheme tailored for the PIR problem, which outperforms the uncoded storage strategy in [26] and the direct MDS coded storage strategy in [12].

Theorem 1 *For PIR from storage constrained databases, the optimal tradeoff between storage and normalized download cost $D^*(\mu)$ is upper bounded by the lower convex hull of the points:*

$$\left(\mu = \frac{t}{KN}, \ D(\mu) = D(t, K) \right), \quad t, K \in [N], t \geq K \quad (12)$$

where $D(t, K)$ is given by:

$$D(t, K) = 1 + \frac{K}{t} + \frac{K^2}{t^2} + \cdots + \frac{K^{M-1}}{t^{M-1}} \quad (13)$$

The proof of Theorem 1 is given is Section V. We have the following remarks.

Remark 1 *Our achievable scheme achieves all the storage-download cost tradeoff points obtained by the uncoded storage systems in [26] if we plugged in $K = 1$. Similarly, our scheme achieves all tradeoff points obtained by the direct MDS coded storage systems in [12] if we plugged in $t = N$.*

Remark 2 *Using our achievable scheme, optimizing the storage problem for PIR becomes a two-dimensional optimization problem over (t, K). This outperforms the storage-download cost tradeoff obtained by uncoded storage in [26] and direct MDS coded storage in [12] as it results in a number of corner points that scales with N^2 [41] in contrast with the number of corner points that scale with N in [12] and [26], which already exist in our scheme.*

Remark 3 *We can write the tradeoff points obtained in (13) as a direct function of μ as:*

$$D(\mu) = 1 + \frac{1}{N\mu} + \frac{1}{(N\mu)^2} + \cdots + \frac{1}{(N\mu)^{M-1}} \quad (14)$$

where $\mu = \frac{t}{KN}$, $t, K \in [N]$, and $t \geq K$. This expression is also achieved by the uncoded storage system in [26] for $\mu = \frac{t}{N}$, $t \in [N]$ and by the MDS coded storage system in [12] for $\mu = \frac{1}{K}$, $K \in [N]$. It is unclear if this curve can be achieved for all $\mu \in [\frac{1}{N}, 1]$.

V. PROPOSED HYBRID STORAGE SCHEME

In this section, we illustrate our achievable scheme by an example without loss of generality. We continue with the example in Section III where $N = 6$ and $M = 2$. In particular, we focus on the storage point $\mu = \frac{5}{12} = 0.416$, whose download cost using the schemes in [12] and [26] lies on the common straight line between $\mu = \frac{1}{3}$ and $\mu = \frac{1}{2}$. Note that, for the scheme in [12] this corresponds to $K = 3$ and $K = 2$, and for the scheme in [26], this corresponds to $t = 2$ and $t = 3$. The download cost achieved by the schemes in [12] and [26] on this line is $\frac{17}{12} = 1.42$. Our proposed hybrid scheme achieves a strictly better download cost of $\frac{7}{5} = 1.40$.

A. Representative Example: $N = 6$, $M = 2$, $\mu = \frac{5}{12}$

1) Storage Phase: Each message is organized as a 30×2 matrix, with symbols picked from a sufficiently large finite field $\mathbb{F}_q$ (in order to have a feasible MDS code). Each message is coded first by a $(6,2)$ MDS code in the same manner as in [12]. Specifically, each row of each message is mapped by the $(6,2)$ MDS code into a vector of length 6. Denote the nth column of the generator matrix of the $(6,2)$ MDS code by $\mathbf{h}_n$, and the jth row of the mth message by $\mathbf{w}_j^{[m]}$. Therefore, the MDS coded symbol corresponding to the jth row of the mth message that is intended to be stored on the nth database is:

$$y_{n,j}^{[m]} = \mathbf{h}_n^T \mathbf{w}_j^{[m]} \tag{15}$$

Note that the MDS code is *not mixing* the messages nor the rows of each message.

Next, we use the uncoded prefetching scheme of [26] with $t = 5$ to store the coded symbols. In this case the rows of each message is divided into $\binom{6}{5} = 6$ partitions. Each partition is a non-intersecting set of row indices, whose cardinality is 5. Let the partition $\mathcal{L}_{\mathcal{S}}$ be the set of row indices that should be stored in the set $\mathcal{S}$ of databases. Therefore, a possible partition assignment for our problem can be:

$$\mathcal{L}_{\{1,2,3,4,5\}} = \{1, 7, 13, 19, 25\} \tag{16}$$
$$\mathcal{L}_{\{1,2,3,4,6\}} = \{2, 8, 14, 20, 26\} \tag{17}$$
$$\mathcal{L}_{\{1,2,3,5,6\}} = \{3, 9, 15, 21, 27\} \tag{18}$$
$$\mathcal{L}_{\{1,2,4,5,6\}} = \{4, 10, 16, 22, 28\} \tag{19}$$
$$\mathcal{L}_{\{1,3,4,5,6\}} = \{5, 11, 17, 23, 29\} \tag{20}$$
$$\mathcal{L}_{\{2,3,4,5,6\}} = \{6, 12, 18, 24, 30\} \tag{21}$$

Now, the nth database stores the coded symbols corresponding to the rows indexed by partitions $\mathcal{L}_{\mathcal{S}}$ such that $n \in \mathcal{S}$. Thus, the contents of the nth database can be written as:

$$Z_n = \bigcup_{m=1}^{2} \bigcup_{\mathcal{L}_{\mathcal{S}} : n \in \mathcal{S}} \bigcup_{j \in \mathcal{L}_{\mathcal{S}}} y_{n,j}^{[m]} \tag{22}$$

Using this storage scheme, each database stores coded symbols from 25 rows. Hence, $\mu = \frac{25}{2*30} = \frac{5}{12} = \frac{t}{KN} = 0.42$. This point lies between $K = 2$ and $K = 3$ for the direct MDS coded case [12], and between $t = 2$ and $t = 3$ for the uncoded prefetching case [26]. Using the uncoded storage or the direct MDS storage, the achievable download cost is $D(\mu) = \frac{17}{12} = 1.42$, which results from memory sharing.

2) Retrieval Phase: The user starts with permuting the row indices of each message partition independently and privately. Specifically, the user permutes $\mathcal{L}_{\mathcal{S}}$ randomly into two sets $\mathcal{L}_{\mathcal{S}}^{[1]}$ and $\mathcal{L}_{\mathcal{S}}^{[2]}$, where $\mathcal{L}_{\mathcal{S}}^{[m]}$ is the permuted set of the partition corresponding to $\mathcal{S}$ for the mth message. Note that these permutations are chosen independently of each other and known only at the user's side. We denote the jth row of the mth message after row permutations by $\mathbf{x}_j^{[m]}$.

In round 1, the user downloads $K^2 = 4$ coded symbols from each database from each message. Note that the user needs to download 2 coded symbols from the same row

from 2 different databases that contain this row, e.g., the user downloads $\mathbf{h}_1^T \mathbf{x}_1^{[1]}$, $\mathbf{h}_2^T \mathbf{x}_1^{[1]}$ from databases 1, 2, respectively. By MDS property, this is sufficient to decode the entire row. The main difference of this step from [12] is the fact that the user does not have the freedom to download these symbols from any 2 databases (but the databases that contain them).

Next, in round 2, the user exploits the side information generated in round 1. From round 1, the user successfully decoded 12 rows from the undesired message. Each side information row of these is stored in 3 databases other than the 2 databases that originally generated this side information. Hence, the user downloads the sum of a new desired coded symbol and a side information coded symbol. Note that the user needs to download the desired coded symbol from 2 different databases in round 2 as well. The user can generate 6 side information equations in round 2 at each database. This is due to the fact that at each database, the user downloads from 4 different rows in round 1, which the user cannot benefit from in round 2. This leaves 8 rows to be used in round 2 as side information. Since each database stores only 10 rows out of these 12 due to the uncoded prefetching strategy, this leaves only 6 rows to be used as side information. The complete query table is given in Table I.

Consequently, the user can decode the entire 30 rows of W_1 as the user can cancel the side information in round 1 and each desired row is decoded from 2 different databases by the MDS property. The privacy is preserved by the random permutations of the message partitions and the identical structure of the queries for both messages. The user downloads 14 coded symbols from all databases and decode the entire $L = 30*2 = 60$ symbols. Therefore, $D(\frac{5}{12}) = \frac{14*6}{30*2} = \frac{7}{5} < \frac{17}{12}$. Hence, our proposed hybrid scheme strictly outperforms uncoded storage and direct MDS coded storage schemes.

Fig. 2 shows that the hybrid scheme achieves four extra corner points than direct MDS coded storage and uncoded storage. These new points are: $(\frac{5}{24}, \frac{9}{5})$, $(\frac{5}{18}, \frac{8}{5})$, $(\frac{5}{12}, \frac{7}{5})$, and $(\frac{2}{9}, \frac{7}{4})$. At all these new points, our scheme outperforms the known storage schemes and achieves the benchmark curve (14) for four more values of the storage parameter μ.

REFERENCES

[1] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan. Private information retrieval. *Journal of the ACM*, 45(6):965–981, November 1998.
[2] W. Gasarch. A survey on private information retrieval. In *Bulletin of the EATCS*, volume 82, pages 72–107, 2004.
[3] C. Cachin, S. Micali, and M. Stadler. Computationally private information retrieval with polylogarithmic communication. In *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 1999.
[4] N. B. Shah, K. V. Rashmi, and K. Ramchandran. One extra bit of download ensures perfectly private information retrieval. In *IEEE ISIT*, June 2014.
[5] T. Chan, S. Ho, and H. Yamamoto. Private information retrieval for coded storage. In *IEEE ISIT*, June 2015.
[6] A. Fazeli, A. Vardy, and E. Yaakobi. PIR with low storage overhead: coding instead of replication. 2015. Available at arXiv:1505.06241.
[7] R. Tajeddine and S. El Rouayheb. Private information retrieval from MDS coded data in distributed storage systems. In *IEEE ISIT*, July 2016.
[8] H. Sun and S. A. Jafar. Blind interference alignment for private information retrieval. In *IEEE ISIT*, July 2016.

TABLE I
QUERY TABLE FOR $M=2$, $N=6$, $\mu=\frac{5}{12}$

DB1	DB2	DB3	DB4	DB5	DB6
$\mathbf{h}_1^T\mathbf{x}_1^{[1]}$	$\mathbf{h}_2^T\mathbf{x}_1^{[1]}$	$\mathbf{h}_3^T\mathbf{x}_2^{[1]}$	$\mathbf{h}_4^T\mathbf{x}_2^{[1]}$	$\mathbf{h}_5^T\mathbf{x}_3^{[1]}$	$\mathbf{h}_6^T\mathbf{x}_3^{[1]}$
$\mathbf{h}_1^T\mathbf{x}_4^{[1]}$	$\mathbf{h}_2^T\mathbf{x}_4^{[1]}$	$\mathbf{h}_3^T\mathbf{x}_5^{[1]}$	$\mathbf{h}_4^T\mathbf{x}_5^{[1]}$	$\mathbf{h}_5^T\mathbf{x}_6^{[1]}$	$\mathbf{h}_6^T\mathbf{x}_6^{[1]}$
$\mathbf{h}_1^T\mathbf{x}_7^{[1]}$	$\mathbf{h}_2^T\mathbf{x}_7^{[1]}$	$\mathbf{h}_3^T\mathbf{x}_8^{[1]}$	$\mathbf{h}_4^T\mathbf{x}_8^{[1]}$	$\mathbf{h}_5^T\mathbf{x}_9^{[1]}$	$\mathbf{h}_6^T\mathbf{x}_9^{[1]}$
$\mathbf{h}_1^T\mathbf{x}_{10}^{[1]}$	$\mathbf{h}_2^T\mathbf{x}_{10}^{[1]}$	$\mathbf{h}_3^T\mathbf{x}_{11}^{[1]}$	$\mathbf{h}_4^T\mathbf{x}_{11}^{[1]}$	$\mathbf{h}_5^T\mathbf{x}_{12}^{[1]}$	$\mathbf{h}_6^T\mathbf{x}_{12}^{[1]}$
$\mathbf{h}_1^T\mathbf{x}_1^{[2]}$	$\mathbf{h}_2^T\mathbf{x}_1^{[2]}$	$\mathbf{h}_3^T\mathbf{x}_2^{[2]}$	$\mathbf{h}_4^T\mathbf{x}_2^{[2]}$	$\mathbf{h}_5^T\mathbf{x}_3^{[2]}$	$\mathbf{h}_6^T\mathbf{x}_3^{[2]}$
$\mathbf{h}_1^T\mathbf{x}_4^{[2]}$	$\mathbf{h}_2^T\mathbf{x}_4^{[2]}$	$\mathbf{h}_3^T\mathbf{x}_5^{[2]}$	$\mathbf{h}_4^T\mathbf{x}_5^{[2]}$	$\mathbf{h}_5^T\mathbf{x}_6^{[2]}$	$\mathbf{h}_6^T\mathbf{x}_6^{[2]}$
$\mathbf{h}_1^T\mathbf{x}_7^{[2]}$	$\mathbf{h}_2^T\mathbf{x}_7^{[2]}$	$\mathbf{h}_3^T\mathbf{x}_8^{[2]}$	$\mathbf{h}_4^T\mathbf{x}_8^{[2]}$	$\mathbf{h}_5^T\mathbf{x}_9^{[2]}$	$\mathbf{h}_6^T\mathbf{x}_9^{[2]}$
$\mathbf{h}_1^T\mathbf{x}_{10}^{[2]}$	$\mathbf{h}_2^T\mathbf{x}_{10}^{[2]}$	$\mathbf{h}_3^T\mathbf{x}_{11}^{[2]}$	$\mathbf{h}_4^T\mathbf{x}_{11}^{[2]}$	$\mathbf{h}_5^T\mathbf{x}_{12}^{[2]}$	$\mathbf{h}_6^T\mathbf{x}_{12}^{[2]}$
$\mathbf{h}_1^T(\mathbf{x}_{13}^{[1]}+\mathbf{x}_2^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{13}^{[1]}+\mathbf{x}_2^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{14}^{[1]}+\mathbf{x}_1^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{14}^{[1]}+\mathbf{x}_1^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{15}^{[1]}+\mathbf{x}_1^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{15}^{[1]}+\mathbf{x}_2^{[2]})$
$\mathbf{h}_1^T(\mathbf{x}_{16}^{[1]}+\mathbf{x}_3^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{16}^{[1]}+\mathbf{x}_3^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{17}^{[1]}+\mathbf{x}_3^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{17}^{[1]}+\mathbf{x}_4^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{18}^{[1]}+\mathbf{x}_4^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{18}^{[1]}+\mathbf{x}_4^{[2]})$
$\mathbf{h}_1^T(\mathbf{x}_{19}^{[1]}+\mathbf{x}_5^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{19}^{[1]}+\mathbf{x}_6^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{20}^{[1]}+\mathbf{x}_6^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{20}^{[1]}+\mathbf{x}_6^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{21}^{[1]}+\mathbf{x}_5^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{21}^{[1]}+\mathbf{x}_5^{[2]})$
$\mathbf{h}_1^T(\mathbf{x}_{22}^{[1]}+\mathbf{x}_8^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{22}^{[1]}+\mathbf{x}_8^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{23}^{[1]}+\mathbf{x}_7^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{23}^{[1]}+\mathbf{x}_7^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{24}^{[1]}+\mathbf{x}_7^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{24}^{[1]}+\mathbf{x}_8^{[2]})$
$\mathbf{h}_1^T(\mathbf{x}_{25}^{[1]}+\mathbf{x}_9^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{25}^{[1]}+\mathbf{x}_9^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{26}^{[1]}+\mathbf{x}_9^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{26}^{[1]}+\mathbf{x}_{10}^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{27}^{[1]}+\mathbf{x}_{10}^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{27}^{[1]}+\mathbf{x}_{10}^{[2]})$
$\mathbf{h}_1^T(\mathbf{x}_{28}^{[1]}+\mathbf{x}_{11}^{[2]})$	$\mathbf{h}_2^T(\mathbf{x}_{28}^{[1]}+\mathbf{x}_{12}^{[2]})$	$\mathbf{h}_3^T(\mathbf{x}_{29}^{[1]}+\mathbf{x}_{12}^{[2]})$	$\mathbf{h}_4^T(\mathbf{x}_{29}^{[1]}+\mathbf{x}_{12}^{[2]})$	$\mathbf{h}_5^T(\mathbf{x}_{30}^{[1]}+\mathbf{x}_{11}^{[2]})$	$\mathbf{h}_6^T(\mathbf{x}_{30}^{[1]}+\mathbf{x}_{11}^{[2]})$

[9] H. Sun and S. A. Jafar. The capacity of private information retrieval. *IEEE Trans. on Info. Theory*, 63(7):4075–4088, July 2017.

[10] H. Sun and S. A. Jafar. The capacity of robust private information retrieval with colluding databases. *IEEE Trans. on Info. Theory*, 64(4):2361–2370, April 2018.

[11] H. Sun and S. A. Jafar. The capacity of symmetric private information retrieval. *IEEE Trans. on Info. Theory*, 65(1):322–329, January 2019.

[12] K. Banawan and S. Ulukus. The capacity of private information retrieval from coded databases. *IEEE Trans. on Info. Theory*, 64(3):1945–1956, March 2018.

[13] Q. Wang and M. Skoglund. Symmetric private information retrieval for MDS coded distributed storage. In *IEEE ICC*, May 2017.

[14] R. Freij-Hollanti, O. Gnilke, C. Hollanti, and D. Karpuk. Private information retrieval from coded databases with colluding servers. *SIAM Journal on Applied Algebra and Geometry*, 1(1):647–664, 2017.

[15] K. Banawan and S. Ulukus. Multi-message private information retrieval: Capacity results and near-optimal schemes. *IEEE Trans. on Info. Theory*, 64(10):6842–6862, October 2018.

[16] K. Banawan and S. Ulukus. The capacity of private information retrieval from byzantine and colluding databases. *IEEE Trans. on Info. Theory*, 65(2):1206–1219, February 2019.

[17] Y.-P. Wei, K. Banawan, and S. Ulukus. Fundamental limits of cache-aided private information retrieval with unknown and uncoded prefetching. *IEEE Trans. on Info. Theory*, 65(5):3215–3232, May 2019.

[18] S. Kadhe, B. Garcia, A. Heidarzadeh, S. El Rouayheb, and A. Sprintson. Private information retrieval with side information. Available at arXiv:1709.00112.

[19] Z. Chen, Z. Wang, and S. Jafar. The capacity of private information retrieval with private side information. Available at arXiv:1709.03022.

[20] Y.-P. Wei, K. Banawan, and S. Ulukus. The capacity of private information retrieval with partially known private side information. *IEEE Trans. on Info. Theory*. Submitted November 2017. Also available at arXiv:1710.00809.

[21] Y.-P. Wei, K. Banawan, and S. Ulukus. Cache-aided private information retrieval with partially known uncoded prefetching: Fundamental limits. *IEEE JSAC*, 36(6):1126–1139, June 2018.

[22] Y.-P. Wei and S. Ulukus. The capacity of private information retrieval with private side information under storage constraints. *IEEE Trans. on Info. Theory*. Submitted November 2018. Also available at arXiv:1806.01253.

[23] S. Li and M. Gastpar. Single-server multi-message private information retrieval with side information. Available at arXiv:1808.05797.

[24] M. Mirmohseni and M. A. Maddah-Ali. Private function retrieval. Available at arXiv:1711.04677.

[25] Z. Chen, Z. Wang, and S. Jafar. The asymptotic capacity of private search. In *IEEE ISIT*, June 2018.

[26] M. A. Attia, D. Kumar, and R. Tandon. The capacity of private information retrieval from uncoded storage constrained databases. Available at arXiv:1805.04104v2.

[27] K. Banawan and S. Ulukus. Asymmetry hurts: Private information retrieval under asymmetric-traffic constraints. *IEEE Trans. on Info. Theory*. Submitted January 2018. Also available at arXiv:1801.03079.

[28] K. Banawan and S. Ulukus. Private information retrieval through wiretap channel II: Privacy meets security. *IEEE Trans. on Info. Theory*. Submitted January 2018. Also available at arXiv:1801.06171.

[29] K. Banawan and S. Ulukus. Noisy private information retrieval: Separability of channel coding and information retrieval. *IEEE Trans. on Info. Theory*. Submitted July 2018. Also available at arXiv: 1807.05997.

[30] Z. Jia, H. Sun, and S. Jafar. Cross subspace alignment and the asymptotic capacity of X-secure T-private information retrieval. Available at arXiv:1808.07457.

[31] C. Tian, H. Sun, and J. Chen. Capacity-achieving private information retrieval codes with optimal message size and upload cost. Available at arXiv:1808.07536.

[32] R. Bitar and S. El Rouayheb. Staircase-PIR: Universally robust private information retrieval. Available at arXiv:1806.08825.

[33] S. Kumar, A. G. i Amat, E. Rosnes, and L. Senigagliesi. Private information retrieval from a cellular network with caching at the edge. Available at arXiv:1809.00872.

[34] S. Kumar, H.-Y. Lin, E. Rosnes, and A. G. i Amat. Achieving maximum distance separable private information retrieval capacity with linear codes. Available at arXiv:1712.03898.

[35] Y.-P. Wei, B. Arasli, K. Banawan, and S. Ulukus. The capacity of private information retrieval from decentralized uncoded caching databases. *IEEE Trans. on Info. Theory*. Submitted November 2018. Also available at arXiv:1811.11160.

[36] N. Raviv and I. Tamo. Private information retrieval in graph based replication systems. In *IEEE ISIT*, June 2018.

[37] K. Banawan and S. Ulukus. Private information retrieval from non-replicated databases. Available at arXiv:1901.00004.

[38] N. Raviv, I. Tamo, and E. Yaakobi. Private information retrieval in graph based replication systems. Available at arXiv:1812.01566.

[39] M. A. Maddah-Ali and U. Niesen. Fundamental limits of caching. *IEEE Trans. on Info. Theory*, 60(5):2856–2867, May 2014.

[40] K. Banawan, B. Arasli, Y.-P. Wei, and S. Ulukus. The capacity of private information retrieval from heterogeneous uncoded caching databases. *IEEE Trans. on Info. Theory*. Submitted February 2019. Also available at arXiv:1902.09512.

[41] E. Weisstein. Farey sequence. 2006.