UC Irvine
UC Irvine Previously Published Works

Title

The Capacity of Robust Private Information Retrieval with Colluding Databases

Permalink
https://escholarship.org/uc/item/2463f96h

Journal
IEEE Transactions on Information Theory, 64(4)

ISSN
0018-9448

Authors
Sun, H
Jafar, SA

Publication Date
2018-04-01

DOI
10.1109/TIT.2017.2777490

Copyright Information
This work is made available under the terms of a Creative Commons Attribution License, availalbe at https://creativecommons.org/licenses/by/4.0/

Peer reviewed

eScholarship.org Powered by the California Digital Library
University of California

arXiv:1605.00635v1 [cs.IT] 2 May 2016

The Capacity of Robust Private Information Retrieval with Colluding Databases

Hua Sun and Syed A. Jafar

Abstract

Private information retrieval (PIR) is the problem of retrieving as efficiently as possible, one out of K messages from N non-communicating replicated databases (each holds all K messages) while keeping the identity of the desired message index a secret from each individual database. The information theoretic capacity of PIR (equivalently, the reciprocal of minimum download cost) is the maximum number of bits of desired information that can be privately retrieved per bit of downloaded information. T-private PIR is a generalization of PIR to include the requirement that even if any T of the N databases collude, the identity of the retrieved message remains completely unknown to them. Robust PIR is another generalization that refers to the scenario where we have $M \geq N$ databases, out of which any $M - N$ may fail to respond. For K messages and $M \geq N$ databases out of which at least some N must respond, we show that the capacity of T-private and Robust PIR is $\left(1 + T/N + T^2/N^2 + \cdots + T^{K-1}/N^{K-1}\right)^{-1}$. The result includes as special cases the capacity of PIR without robustness ($M = N$) or T-privacy constraints ($T = 1$).

1 Introduction

The private information retrieval (PIR) problem is motivated by the desire to protect the privacy of a user against data providers. Besides its direct applications in data privacy, it is intimately related to many fundamental problems in cryptography, e.g., oblivious transfer [1], instance hiding [2, 3, 4], secure multiparty computation [5], and secret sharing schemes [6, 7]. The significance of PIR also extends beyond security, through its fundamental connections to other prominent topics such as locally decodable codes [8] and batch codes [9] in coding theory, relationships between communication and computation [10] in complexity theory, and most recently blind interference alignment [11] in wireless communications. In fact most constructions of locally decodable codes are translated directly from PIR schemes. Through the connections between locally decodable and locally recoverable codes [12], PIR also connects to distributed data storage repair [13] and index coding [14], which in turn encompass all of network coding [15]. Therefore PIR represents an important focal point to tackle significant challenges across these fields.

The goal of PIR is to find the most efficient way for a user to retrieve a desired message from a set of N distributed databases, each of which stores all K messages, without revealing anything

Hua Sun (email: huas2@uci.edu) and Syed A. Jafar (email: syed@uci.edu) are with the Center of Pervasive Communications and Computing (CPCC) in the Department of Electrical Engineering and Computer Science (EECS) at the University of California Irvine.

(in the information theoretic sense[1]) about which message is being retrieved, to any individual database. The PIR problem was initially studied in the setting where each message is one bit long [19, 20, 21, 22, 8, 23], where the cost of a PIR scheme is measured by the total amount of communication between the user and the databases, i.e., the sum of communications from the user to the databases (upload) and from the databases to the user (download). What is pursued in this work is the traditional Shannon theoretic formulation, where message size is allowed to be arbitrarily large, and therefore the upload cost is negligible compared to the download cost [24, 20]. The information theoretic capacity of PIR is the maximum number of bits of desired information that can be privately retrieved per bit of downloaded information. Equivalently, it is the reciprocal of the minimum possible download cost per bit of desired message. In [25], we showed that the information theoretic capacity of PIR, for arbitrary number of messages K and arbitrary number of databases N is $\left(1+1/N+1/N^2+\cdots+1/N^{K-1}\right)^{-1}$.

There are several interesting extensions of PIR that explore its limitations under additional constraints. These include extensions where up to T of the N databases may collude [26, 27] (T-private PIR); where some of the databases may not respond [28] (Robust PIR); where both the privacy of the user and the databases must be protected [1] (Symmetric PIR); where only one database holds all the messages and all other databases hold independent information [29]; where retrieval operations are unsynchronized [30]; and where beyond communications, computation is also a concern [31]. There is also much recent work in the distributed storage setting [32, 24, 33, 34] (the databases form a distributed storage system) where the main focus is on how the coding of the storage system works jointly with PIR.

In this work, we mainly consider T-private PIR in the Shannon theoretic setting, where we have an arbitrary number of messages (K), arbitrary number of databases (N), each database stores all the messages, the messages are allowed to be arbitrarily large, and the privacy of the desired message index must be guaranteed even if any T of the N databases collude. The main contribution of this work is to show that the information theoretic capacity of T-private PIR is $\left(1+T/N+T^2/N^2+\cdots+T^{K-1}/N^{K-1}\right)^{-1}$.

We further consider the extension to *robust* T-private PIR, where we have $M \geq N$ databases, out of which any $M-N$ databases may not respond, so that with answers from any N databases, we need to ensure both privacy and correctness. In this context, the contribution of this work is to show that the information theoretic capacity of robust T-private PIR remains the same as that of T-private PIR, i.e., there is no capacity cost from not knowing in advance *which* N databases will respond.

Notation: For $n_1, n_2 \in \mathbb{Z}, n_1 \leq n_2$, define the notation $[n_1 : n_2]$ as the set $\{n_1, n_1+1, \cdots, n_2\}$ and $(n_1 : n_2)$ as the vector $(n_1, n_1+1, \cdots, n_2)$. For an index set $\mathcal{I} = \{i_1, i_2, \cdots, i_n\}$, the notation $A_{\mathcal{I}}$ represents the set $\{A_i : i \in \mathcal{I}\}$. For an index vector $\mathcal{I} = (i_1, i_2, \cdots, i_n)$, the notation $A_{\mathcal{I}}$ represents the vector $(A_{i_1}, A_{i_2}, \cdots, A_{i_n})$. For a matrix S, the notation $S[\mathcal{I}, :]$ represents the submatrix of S formed by retaining only the rows corresponding to the elements of the vector $\mathcal{I}$. The notation $X \sim Y$ is used to indicate that X and Y are identically distributed.

[1] There is another line of research, where privacy needs to be satisfied only for computationally bounded databases [16, 17, 18].

2 Problem Statement

2.1 T-private PIR

Consider K independent messages $W_1, \cdots, W_K$ of size L bits each.

$$H(W_1, \cdots, W_K) = H(W_1) + \cdots + H(W_K), \tag{1}$$

$$H(W_1) = \cdots = H(W_K) = L. \tag{2}$$

There are N databases. Each database stores all the messages $W_1, \cdots, W_K$. A user wants to retrieve $W_k, k \in [1 : K]$ subject to T-privacy, i.e., without revealing anything about the message identity, k, to any colluding susbset of up to T out of the N databases.

To retrieve W_k privately, the user generates N queries $Q_1^{[k]}, \cdots, Q_N^{[k]}$, where the superscript denotes the desired message index. Since the queries are generated with no knowledge of the realizations of the messages, the queries must be independent of the messages,

$$I(W_1, \cdots, W_K; Q_1^{[k]}, \cdots, Q_N^{[k]}) = 0. \tag{3}$$

The user sends query $Q_n^{[k]}$ to the n-th database, $\forall n \in [1 : N]$. Upon receiving $Q_n^{[k]}$, the n-th database generates an answering string $A_n^{[k]}$, which is a deterministic function of $Q_n^{[k]}$ and the data stored (i.e., all messages $W_1, \cdots, W_K$),

$$H(A_n^{[k]} | Q_n^{[k]}, W_1, \cdots, W_K) = 0. \tag{4}$$

Each database returns to the user its answer $A_n^{[k]}$. From all answers $A_1^{[k]}, \cdots, A_N^{[k]}$, the user can decode the desired message W_k,

$$\text{[Correctness]}\ H(W_k | A_1^{[k]}, \cdots, A_N^{[k]}, Q_1^{[k]}, \cdots, Q_N^{[k]}) = 0. \tag{5}$$

To satisfy the privacy constraint that any T colluding databases learn nothing about the desired message index k information theoretically, any T queries must be independent of k. Let $\mathcal{T}$ be a subset of $[1 : N]$ and its cardinality be denoted by $|\mathcal{T}|$. $Q_{\mathcal{T}}^{[k]}$ represents the subset $\{Q_n^{[k]}, n \in \mathcal{T}\}$. $A_{\mathcal{T}}^{[k]}$ is defined similarly. To satisfy the T-privacy requirement we must have

$$\text{[Privacy]} \quad I(Q_{\mathcal{T}}^{[k]}; k) = 0, \forall \mathcal{T} \subset [1 : N], |\mathcal{T}| = T. \tag{6}$$

As the answering string is a deterministic function of the query and all messages, any set of T answering strings must be independent of k as well,

$$I(A_{\mathcal{T}}^{[k]}; k) = 0, \forall \mathcal{T} \subset [1 : N], |\mathcal{T}| = T. \tag{7}$$

To underscore that any set of T or fewer answering strings is independent of the desired message index, we may suppress the superscript and write $A_{\mathcal{T}}$ directly instead of $A_{\mathcal{T}}^{[k]}$, and express the elements of such a set as A_n instead of $A_n^{[k]}$.

The metric that we study in this paper is the PIR rate, which characterizes how many bits of desired information are retrieved per downloaded bit. Note that the PIR rate is the reciprocal of download cost. The rate R of a PIR scheme is defined as follows.

$$R \triangleq \frac{L}{\sum_{n=1}^{N} H(A_n)}. \tag{8}$$

The capacity, C, is the supremum of R over all PIR schemes.

2.2 Robust T-private PIR

The robust T-private PIR problem is defined similar to the T-private PIR problem. The only difference is that instead of N databases, we have $M \geq N$ databases, and the correctness condition needs to be satisfied when the user collects *any* N out of the M answering strings.

3 Main Result: Capacity of Robust T-Private PIR

The following theorem states the main result.

Theorem 1 *For T-private PIR with K messages and N databases, the capacity is*

$$C = \left(1 + T/N + T^2/N^2 + \cdots + T^{K-1}/N^{K-1}\right)^{-1}. \tag{9}$$

The capacity of PIR with T colluding databases generalizes the case without T-privacy constraints, where $T = 1$ [25]. The capacity is a strictly decreasing function of T. When $T = N$, the capacity is $1/K$, meaning that the user has to download all K messages to be private, as in this case, the colluding databases are as strong as the user. Similar to the $T = 1$ case, the capacity is strictly deceasing in the number of messages, K, and strictly increasing in the number of databases, N. When the number of messages approaches infinity, the capacity approaches $1 - T/N$, and when the number of databases approaches infinity (T remains constant), the capacity approaches 1. Finally, note that since the download cost is the reciprocal of the rate, the capacity characterization in Theorem 1 equivalently characterizes the optimal download cost per message bit for T-private PIR as $\left(1 + T/N + T^2/N^2 + \cdots + T^{K-1}/N^{K-1}\right)$ bits.

The capacity-achieving scheme that we construct for T-private PIR, generalizes easily to incorporate robustness constraints. As a consequence, we are also able to characterize the capacity of robust T-private PIR. This result is stated in the following theorem.

Theorem 2 *The capacity of robust T-private PIR is*

$$C = \left(1 + T/N + T^2/N^2 + \cdots + T^{K-1}/N^{K-1}\right)^{-1}. \tag{10}$$

Since the capacity expressions are the same, we note that there is no capacity penalty from not knowing in advance which N databases will respond. Even though this uncertainty increases as M increases, capacity is not a function of M. However, we note that the communication complexity of our capacity achieving scheme does increase with M.

4 Proof of Theorem 1: Achievability

There are two key aspects of the achievable scheme – 1) the query structure, and 2) the specialization of the query structure to ensure T-privacy and correctness. While the query structure is different from the $T = 1$ setting of [25], it draws upon the iterative application of the same three principles that were identified in [25]. These principles are listed below.

(1) *Enforcing Symmetry Across Databases*

(2) *Enforcing Message Symmetry within the Query to Each Database*

(3) *Exploiting Previously Acquired Side Information of Undesired Messages to Retrieve New Desired Information*

The specialization of the structure to ensure T-privacy and correctness is another novel element of the achievable scheme. To illustrate how these ideas work together in an iterative fashion, we will present a few simple examples corresponding to small values of K, N and T, and then generalize it to arbitrary K, N and T. Let us begin with a lemma.

Lemma 1 *Let $S_1, S_2, \cdots, S_K \in \mathcal{F}_q^{\alpha\times\alpha}$ be K random matrices, drawn independently and uniformly from all $\alpha \times \alpha$ full-rank matrices over $\mathcal{F}_q$. Let $G_1, G_2, \cdots, G_K \in \mathcal{F}_q^{\beta\times\beta}$ be K invertible square matrices of dimension $\beta\times\beta$ over $\mathcal{F}_q$. Let $\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_K \in \mathbb{N}^{\beta\times 1}$ be K index vectors, each containing β distinct indices from $[1:\alpha]$. Then*

$$(G_1S_1[\mathcal{I}_1,:], G_2S_2[\mathcal{I}_2,:], \cdots, G_KS_K[\mathcal{I}_K,:]) \sim (S_1[(1:\beta),:], S_2[(1:\beta),:], \cdots, S_K[(1:\beta),:]) \tag{11}$$

where $S_i[\mathcal{I}_i,:], i \in [1:K]$ are $\beta \times \alpha$ matrices comprised of the rows of S_i with indices in $\mathcal{I}_i$.

Proof: We wish to prove that the left hand side of (11) is identically distributed (recall that the notation $X \sim Y$ means that X and Y are identically distributed) to the right hand side of (11). Because the rank of a matrix does not depend on the ordering of the rows, we have

$$(S_1[\mathcal{I}_1,:], S_2[\mathcal{I}_2,:], \cdots, S_K[\mathcal{I}_K,:]) \sim (S_1[(1:\beta),:], S_2[(1:\beta),:], \cdots, S_K[(1:\beta),:])$$

Since S_i are picked uniformly from all full-rank matrices, conditioned on any feasible value of the remaining rows $S_i[(\beta+1:\alpha),:]$, the first β rows $S_i[(1:\beta),:]$ are uniformly distributed over all possibilities that preserve full-rank for S_i. Now note that the mapping from $S_i[(1:\beta),:]$ to $G_iS_i[(1:\beta),:]$ is bijective, and $S_i[(1:\beta),:]$ spans the same row space as $G_iS_i[(1:\beta),:]$, i.e., replacing $S_i[(1:\beta),:]$ with $G_iS_i[(1:\beta),:]$, preserves S_i as a full-rank matrix. Therefore, conditioned on any feasible $S_i[(\beta+1:\alpha),:]$, the set of feasible values of $S_i[(1:\beta),:]$ is the same as the set of feasible $G_iS_i[(1:\beta),:]$ values. Therefore, $G_iS_i[(1:\beta),:]$ is also uniformly distributed over the same set. Finally, since the S_i are chosen independently, the statement of Lemma 1 follows. ■

4.1 $K = 2$ Messages, $N = 3$ Databases, $T = 2$ Colluding Databases

The capacity for this setting, is $C = \left(1 + \frac{2}{3}\right)^{-1} = \frac{3}{5}$.

4.1.1 Query Structure

We begin by constructing a query structure, which will then be specialized to achieve correctness and privacy. Without loss of generality, let $[a_k]$ denote the symbols of the desired message, and $[b_k]$ the symbols of the undesired message.

DB1	DB2	DB3
a_1, a_2	a_3, a_4	

$\xrightarrow{(1)}$

DB1	DB2	DB3
a_1, a_2	a_3, a_4	a_5, a_6

$\xrightarrow{(2)}$

DB1	DB2	DB3
a_1, a_2	a_3, a_4	a_5, a_6
b_1, b_2	b_3, b_4	b_5, b_6

$\cdots$

$\cdots \xrightarrow{(3)}$

DB1	DB2	DB3
a_1, a_2	a_3, a_4	a_5, a_6
b_1, b_2	b_3, b_4	b_5, b_6
$a_7 + b_7$	$a_8 + b_8$	

$\xrightarrow{(1)}$

DB1	DB2	DB3
a_1, a_2	a_3, a_4	a_5, a_6
b_1, b_2	b_3, b_4	b_5, b_6
$a_7 + b_7$	$a_8 + b_8$	$a_9 + b_9$

We start by requesting the first $T^{K-1} = 2$ symbols from each of the first $T = 2$ databases: a_1, a_2 from DB1, and a_3, a_4 from DB2. Applying database symmetry, we simultaneously request a_5, a_6 from DB3. Next, we enforce message symmetry, by including queries for $b_1, \cdots, b_6$ as the counterparts for $a_1, \cdots, a_6$. Now consider the first $T = 2$ databases, i.e., DB1 and DB2, which can potentially collude with each other. Unknown to these databases the user has acquired two symbols of external side information, b_5, b_6, comprised of undesired message symbols received from DB3. Splitting the two symbols of external side information among DB1 and DB2 allows the user one symbol of side information for each of DB1 and DB2 that it can exploit to retrieve new desired information symbols. In our construction of the query structure, we will assign new labels to the external side-information exploited within each database, e.g., b_7 for DB1 and b_8 for DB2, with the understanding that eventually when the dependencies within the structure are specialized, b_7, b_8 will turn out to be functions of previously acquired side-information. Using its assigned side information, each DB acquires a new symbol of desired message, so that DB1 requests $a_7 + b_7$ and DB2 requests $a_8 + b_8$. Finally, enforcing symmetry across databases, DB3 requests $a_9 + b_9$. At this point, the construction is symmetric across databases, the query to any database is symmetric in itself across messages, and the amount of side information exploited within any T colluding databases equals the amount of side information available external to those T databases. So the skeleton of the query structure is complete.

Note that if DB1 and DB2 collude, then the external side information is b_5, b_6, so we would like the side-information that is exploited by DB1 and DB2, i.e., b_7, b_8 to be functions of the external side information that is available, i.e., b_5, b_6. However, since *any* $T = 2$ databases can collude, it is also possible that DB1 and DB3 collude instead, in which case we would like b_7, b_9 to be functions of side information that is external to DB1 and DB3, i.e., b_3, b_4. Similarly, if DB2 and DB3 collude, then we would like b_8, b_9 to be functions of b_1, b_2. How to achieve such dependencies in a manner that preserves privacy and ensures correctness is the remaining challenge. Intuitively, the key is to make b_7, b_8, b_9 depend on *all* side-information $b_1, b_2, \cdots, b_6$ in a generic sense. In other words, we will achieve the desired functional dependencies by viewing $b_1, b_2, \cdots, b_9$ as the outputs of a $(9, 6)$ MDS code, so that any 3 of these b_k are functions of the remaining 6. The details of this specialization are described next.

4.1.2 Specialization to Ensure Correctness and Privacy

Let each message consist of $N^K = 9$ symbols from a sufficiently large[2] finite field $\mathbb{F}_q$. The messages W_1, $W_2 \in \mathbb{F}_q^{9\times 1}$ are then represented as 9×1 vectors over $\mathbb{F}_q$. Let $S_1, S_2 \in \mathbb{F}_q^{9\times 9}$ represent random matrices chosen privately by the user, independently and uniformly from all 9×9 full-rank matrices over $\mathbb{F}_q$. Without loss of generality, let us assume that W_1 is the desired message. Define the 9×1 vectors $a_{[1:9]} \in \mathbb{F}_q^{9\times 1}$ and $b_{[1:9]} \in \mathbb{F}_q^{9\times 1}$, as follows

$$a_{[1:9]} = S_1 W_1 \tag{12}$$
$$b_{[1:9]} = \mathrm{MDS}_{9\times 6} S_2[(1:6), :] W_2 \tag{13}$$

where $S_2[(1:6), :]$ is a 6×9 matrix comprised of the first 6 rows of S_2. $\mathrm{MDS}_{9\times 6}$ is the generator matrix of a $(9, 6)$ MDS code (e.g., a Reed Solomon code). The generator matrix does not need to be random, i.e., it may be globally known. Note that because of the MDS property, any 6 rows of $\mathrm{MDS}_{9\times 6}$ form a 6×6 invertible matrix. Therefore, from any 6 elements of $b_{[1:9]}$, all 9 elements of

[2] The requirements on the size of the field have to do with the existence of MDS codes that are used in the construction. In this case $q \geq N^K$ is sufficient.

$b_{[1:9]}$ can be recovered. For example, from $b_1, b_2, \cdots, b_6$, one can recover b_7, b_8, b_9. The queries from each database are constructed according to the structure described earlier.

DB1	DB2	DB3
a_1, a_2 b_1, b_2 $a_7 + b_7$	a_3, a_4 b_3, b_4 $a_8 + b_8$	a_5, a_6 b_5, b_6 $a_9 + b_9$

(14)

Correctness is easy to see, because the user recovers $b_{[1:6]}$ explicitly, from which it can recover all $b_{[1:9]}$, thereby allowing it to recover all of $a_{[1:9]}$. Let us see why privacy holds. The queries for any $T = 2$ colluding databases are comprised of 6 variables from $a_{[1:9]}$ and 6 variables from $b_{[1:9]}$. Let the indices of these variables be denoted by the 6×1 vectors $\mathcal{I}_a, \mathcal{I}_b \in \mathbb{N}^{6\times 1}$, respectively, so that,

$$\begin{aligned} (a_{\mathcal{I}_a}, b_{\mathcal{I}_b}) &= (S_1[\mathcal{I}_a, :]W_1, \mathrm{MDS}_{9\times 6}[\mathcal{I}_b, :]S_2[(1:6), :]W_2) && (15)\\ &\sim (S_1[(1:6), :]W_1, S_2[(1:6), :]W_2) && (16) \end{aligned}$$

where (16) follows from Lemma 1 because $\mathrm{MDS}_{9\times 6}[\mathcal{I}_b, :]$ is an invertible 6×6 matrix. Therefore, the random map from W_1 to $a_{\mathcal{I}_a}$ variables is i.i.d. as the random map from W_2 to $b_{\mathcal{I}_b}$, and privacy is guaranteed. Note that since 9 desired symbols are recovered from a total of 15 downloaded symbols, the rate achieved by this scheme is $9/15 = 3/5$, which matches the capacity for this setting. While this specialization suffices for our purpose (it achieves capacity), we note that further simplifications of the scheme are possible, which allow it to operate over smaller fields and with lower upload cost. Such an example is provided in the conclusion section of this paper.

4.2 $K = 2$ Messages, $N = 4$ Databases, $T = 2$ Colluding Databases

The capacity for this setting, is $C = \left(1 + \frac{2}{4}\right)^{-1} = \frac{2}{3}$.

4.2.1 Query Structure

The query structure is constructed as follows.

DB1	DB2	DB3	DB4
a_1, a_2	a_3, a_4		

$\xrightarrow{(1)}$

DB1	DB2	DB3	DB4
a_1, a_2	a_3, a_4	a_5, a_6	a_7, a_8

$\xrightarrow{(2)}$

DB1	DB2	DB3	DB4
a_1, a_2 b_1, b_2	a_3, a_4 b_3, b_4	a_5, a_6 b_5, b_6	a_7, a_8 b_7, b_8

$\cdots$

$\cdots \xrightarrow{(3)}$

DB1	DB2	DB3	DB4
a_1, a_2 b_1, b_2 $a_9 + b_9$ $a_{10} + b_{10}$	a_3, a_4 b_3, b_4 $a_{11} + b_{11}$ $a_{12} + b_{12}$	a_5, a_6 b_5, b_6	a_7, a_8 b_7, b_8

$\xrightarrow{(1)}$

DB1	DB2	DB3	DB4
a_1, a_2 b_1, b_2 $a_9 + b_9$ $a_{10} + b_{10}$	a_3, a_4 b_3, b_4 $a_{11} + b_{11}$ $a_{12} + b_{12}$	a_5, a_6 b_5, b_6 $a_{13} + b_{13}$ $a_{14} + b_{14}$	a_7, a_8 b_7, b_8 $a_{15} + b_{15}$ $a_{16} + b_{16}$

As before, we start by requesting $T^{K-1} = 2$ symbols of desired message from each of the first $T = 2$ databases. After enforcing symmetries across databases and then across messages, we consider the first T databases (DB1 and DB2) and measure the total amount of external side information available outside DB1 and DB2, which turns out to be 4 symbols (b_5, b_6, b_7, b_8). This gives us our budget of 2 symbols of side information per database to be exploited to retrieve new desired information symbols. Assigning new labels to the side information symbols being exploited by each database, we add $a_9+b_9, a_{10}+b_{10}$ to the query from DB1 and $a_{11}+b_{11}, a_{12}+b_{12}$ to the query from DB2. Finally, enforcing symmetry across databases, corresponding queries are added to DB3 and DB4. At this point all symmetries are satisfied and the amounts of side-information available and exploited are balanced. Thus the structure is complete.

4.2.2 Specialization

Let each message consist of $N^K = 16$ symbols from a sufficiently large finite field $\mathbb{F}_q$. The messages W_1, $W_2 \in \mathbb{F}_q^{16\times 1}$ are therefore represented as 16×1 vectors over $\mathbb{F}_q$. Let $S_1, S_2 \in \mathbb{F}_q^{16\times 16}$ represent random matrices chosen privately by the user, independently and uniformly from all 16×16 full-rank matrices over $\mathbb{F}_q$. Without loss of generality, let us assume that W_1 is the desired message. Define 16×1 vectors $a_{[1:16]} \in \mathbb{F}_q^{16\times 1}$ and $b_{[1:16]} \in \mathbb{F}_q^{16\times 1}$, as follows

$$a_{[1:16]} = S_1 W_1 \tag{17}$$
$$b_{[1:16]} = \text{MDS}_{16\times 8} S_2[(1:8),:] W_2 \tag{18}$$

where $S_2[(1:8),:]$ is a 8×16 matrix comprised of the first 8 rows of S_2. $\text{MDS}_{16\times 8}$ is the generator matrix of a $(16,8)$ MDS code. Plugging into the query structure obtained above,

DB1	DB2	DB3	DB4
a_1, a_2	a_3, a_4	a_5, a_6	a_7, a_8
b_1, b_2	b_3, b_4	b_5, b_6	b_7, b_8
a_9+b_9	$a_{11}+b_{11}$	$a_{13}+b_{13}$	$a_{15}+b_{15}$
$a_{10}+b_{10}$	$a_{12}+b_{12}$	$a_{14}+b_{14}$	$a_{16}+b_{16}$

the proof of correctness and privacy follows from the same reasoning as in the previous example. Note that since 16 desired symbols are recovered from a total of 24 downloaded symbols, the rate achieved by this scheme is $16/24 = 2/3$, which matches the capacity for this setting.

4.3 $K = 2$ Messages, $N = 4$ Databases, $T = 3$ Colluding Databases

The capacity for this setting, is $C = \left(1+\frac{3}{4}\right)^{-1} = \frac{4}{7}$.

4.3.1 Query Structure

The query structure is constructed as follows.

DB1	DB2	DB3	DB4
a_1, a_2, a_3	a_4, a_5, a_6	a_7, a_8, a_9	

$\xrightarrow{(1)}$

DB1	DB2	DB3	DB4
a_1, a_2, a_3	a_4, a_5, a_6	a_7, a_8, a_9	a_{10}, a_{11}, a_{12}

$\cdots$

$\cdots \xrightarrow{(2)}$

DB1	DB2	DB3	DB4
a_1, a_2, a_3	a_4, a_5, a_6	a_7, a_8, a_9	a_{10}, a_{11}, a_{12}
b_1, b_2, b_3	b_4, b_5, b_6	b_7, b_8, b_9	b_{10}, b_{11}, b_{12}

$\cdots$

$\cdots \xrightarrow{(3)}$

DB1	DB2	DB3	DB4
a_1, a_2, a_3	a_4, a_5, a_6	a_7, a_8, a_9	a_{10}, a_{11}, a_{12}
b_1, b_2, b_3	b_4, b_5, b_6	b_7, b_8, b_9	b_{10}, b_{11}, b_{12}
$a_{13}+b_{13}$	$a_{14}+b_{14}$	$a_{15}+b_{15}$	

$\cdots$

$\cdots \xrightarrow{(1)}$

DB1	DB2	DB3	DB4
a_1, a_2, a_3	a_4, a_5, a_6	a_7, a_8, a_9	a_{10}, a_{11}, a_{12}
b_1, b_2, b_3	b_4, b_5, b_6	b_7, b_8, b_9	b_{10}, b_{11}, b_{12}
$a_{13} + b_{13}$	$a_{14} + b_{14}$	$a_{15} + b_{15}$	$a_{16} + b_{16}$

Starting with $T^{K-1} = 3$ symbols each requested from the first $T = 3$ databases, after the enforcing of symmetries across databases and then across messages, we jointly consider the first $T = 3$ databases, DB1, DB2 and DB3. The amount of external side information available is 3 symbols (b_{10}, b_{11}, b_{12}), which allows a budget of one symbol of side information per database to be exploited to recover new symbols of desired information. Assigning new labels to the side information symbols being exploited by each database, we include $a_{13} + b_{13}, a_{14} + b_{14}, a_{15} + b_{15}$ in the queries from DB1, DB2 and DB3, respectively. Finally, enforcing symmetry across databases, correspondingly $a_{16}+b_{16}$ is added to DB4. At this point all symmetries are satisfied and the amounts of side-information available and exploited are balanced. Thus the structure is complete.

4.3.2 Specialization

Let each message consist of $N^K = 16$ symbols from a sufficiently large finite field $\mathbb{F}_q$. The messages W_1, $W_2 \in \mathbb{F}_q^{16\times 1}$ are therefore represented as 16×1 vectors over $\mathbb{F}_q$. Let $S_1, S_2 \in \mathbb{F}_q^{16\times 16}$ represent random matrices chosen privately by the user, independently and uniformly from all 16×16 full-rank matrices over $\mathbb{F}_q$. Without loss of generality, let us assume that W_1 is the desired message. Define 16×1 vectors $a_{[1:16]} \in \mathbb{F}_q^{16\times 1}$ and $b_{[1:16]} \in \mathbb{F}_q^{16\times 1}$, as follows

$$a_{[1:16]} = S_1 W_1 \tag{19}$$

$$b_{[1:16]} = \mathrm{MDS}_{16\times 12} S_2[(1:12),:] W_2 \tag{20}$$

where $S_2[(1:12),:]$ is a 12×16 matrix comprised of the first 12 rows of S_2. $\mathrm{MDS}_{16\times 12}$ is the generator matrix of a $(16, 12)$ MDS code. After plugging these values into query structure derived earlier, the proof of correctness and privacy follows from the same reasoning as in the previous examples. Note that since 16 desired symbols are recovered from a total of 28 downloaded symbols, the rate achieved by this scheme is $16/28 = 4/7$, which matches the capacity for this setting.

4.4 $K = 3$ Messages, $N = 3$ Databases, $T = 2$ Colluding Databases

The capacity for this setting, is $C = \left(1 + \frac{2}{3} + (\frac{2}{3})^2\right)^{-1} = \frac{9}{19}$.

4.4.1 Query Structure

The query structure is constructed as follows.

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	

$\xrightarrow{(1)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$

$\cdots$

$\cdots \xrightarrow{(2)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$

$\xrightarrow{(3)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	
$a_{14} + b_{14}$	$a_{16} + b_{16}$	
$a_{17} + c_{13}$	$a_{19} + c_{15}$	
$a_{18} + c_{14}$	$a_{20} + c_{16}$	

$\cdots$

$\cdots \xrightarrow{(1)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	$a_{21} + b_{17}$
$a_{14} + b_{14}$	$a_{16} + b_{16}$	$a_{22} + b_{18}$
$a_{17} + c_{13}$	$a_{19} + c_{15}$	$a_{23} + c_{17}$
$a_{18} + c_{14}$	$a_{20} + c_{16}$	$a_{24} + c_{18}$

$\xrightarrow{(2)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	$a_{21} + b_{17}$
$a_{14} + b_{14}$	$a_{16} + b_{16}$	$a_{22} + b_{18}$
$a_{17} + c_{13}$	$a_{19} + c_{15}$	$a_{23} + c_{17}$
$a_{18} + c_{14}$	$a_{20} + c_{16}$	$a_{24} + c_{18}$
$b_{19} + c_{19}$	$b_{21} + c_{21}$	$b_{23} + c_{23}$
$b_{20} + c_{20}$	$b_{22} + c_{22}$	$b_{24} + c_{24}$

$\cdots$

$\cdots \xrightarrow{(3)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	$a_{21} + b_{17}$
$a_{14} + b_{14}$	$a_{16} + b_{16}$	$a_{22} + b_{18}$
$a_{17} + c_{13}$	$a_{19} + c_{15}$	$a_{23} + c_{17}$
$a_{18} + c_{14}$	$a_{20} + c_{16}$	$a_{24} + c_{18}$
$b_{19} + c_{19}$	$b_{21} + c_{21}$	$b_{23} + c_{23}$
$b_{20} + c_{20}$	$b_{22} + c_{22}$	$b_{24} + c_{24}$
$a_{25} + b_{25} + c_{25}$	$a_{26} + b_{26} + c_{26}$	

$\xrightarrow{(1)}$

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	$a_{21} + b_{17}$
$a_{14} + b_{14}$	$a_{16} + b_{16}$	$a_{22} + b_{18}$
$a_{17} + c_{13}$	$a_{19} + c_{15}$	$a_{23} + c_{17}$
$a_{18} + c_{14}$	$a_{20} + c_{16}$	$a_{24} + c_{18}$
$b_{19} + c_{19}$	$b_{21} + c_{21}$	$b_{23} + c_{23}$
$b_{20} + c_{20}$	$b_{22} + c_{22}$	$b_{24} + c_{24}$
$a_{25} + b_{25} + c_{25}$	$a_{26} + b_{26} + c_{26}$	$a_{27} + b_{27} + c_{27}$

Starting with $T^{K-1} = 4$ symbols each requested from the first $T = 2$ databases, we proceed through iterative steps (1) and (2) to enforce symmetries across databases and messages. In step (3) we consider the first $T = 2$ databases together (DB1 and DB2) and account for the external side information, which in this case contains 4 symbols from $[b_k]$ and 4 symbols from $[c_k]$. Distributed evenly among DB1 and DB2, this allows a budget of 2 symbols of side information from $[b_k]$ and 2 symbols from $[c_k]$ per database to be exploited to recover new symbols of desired information. Proceeding again through steps (1) and (2) to enforce symmetries across databases and messages, we end up with new downloads that contain only undesired information symbols, which can now be used to download new desired information symbols. Once again, we consider DB1 and DB2 together, and account for the new external side information, $b_{23} + c_{23}, b_{24} + c_{24}$. Thus the external side information is comprised of two symbols, each of which is a sum of the form $b_k + c_k$. Dividing the side information evenly among databases DB1 and DB2, each is assigned one side-information symbol of the form $b_k + c_k$ with new labels. Thus, $a_{25} + b_{25} + c_{25}$ is added to the query from DB1, and $a_{26} + b_{26} + c_{26}$ is added to the query from DB2. Finally, applying symmetry across databases, we include $a_{27} + b_{27} + c_{27}$ to the query from DB3. At this point, all symmetries are satisfied, all external and exploited side-information amounts are balanced, and therefore, the query structure is complete.

4.4.2 Specialization

Let each message consist of $N^K = 27$ symbols from a sufficiently large finite field $\mathbb{F}_q$. The messages $W_1, W_2, W_3 \in \mathbb{F}_q^{27\times 1}$ are then represented as 27×1 vectors over $\mathbb{F}_q$. Let $S_1, S_2, S_3 \in \mathbb{F}_q^{27\times 27}$ represent random matrices chosen privately by the user, independently and uniformly from all 27×27 full-rank matrices over $\mathbb{F}_q$. Without loss of generality, let us assume that W_1 is the desired message. Define 27×1 vectors $a_{[1:27]}, b_{[1:27]}, c_{[1:27]} \in \mathbb{F}_q^{27\times 1}$, as follows

$$a_{[1:27]} \quad = \quad S_1 W_1 \tag{21}$$

$$b_{[1:18]} = \text{MDS}_{18\times 12} S_2[(1:12),:]W_2 \tag{22}$$
$$c_{[1:18]} = \text{MDS}_{18\times 12} S_3[(1:12),:]W_3 \tag{23}$$
$$b_{[19:27]} = \text{MDS}_{9\times 6} S_2[(13:18),:]W_2 \tag{24}$$
$$c_{[19:27]} = \text{MDS}_{9\times 6} S_3[(13:18),:]W_3 \tag{25}$$

where $S_2[(1:18),:]$ is a 18×27 matrix comprised of the first 18 rows of S_2. $\text{MDS}_{18\times 12}$ is the generator matrix of a $(18,12)$ MDS code, and $\text{MDS}_{9\times 6}$ is the generator matrix of a $(9,6)$ MDS code. In particular, note that the *same* generator matrix is used in (22) and (23). Similarly, the same generator matrix is used in (24) and (25). This is important because it allows us to write

$$b_{[19:27]} + c_{[19:27]} = \text{MDS}_{9\times 6}\left(S_2[(13:18),:]W_2 + S_3[(13:18),:]W_3\right) \tag{26}$$

so that all 9 elements of the vector $b_{[19:27]} + c_{[19:27]}$ can be recovered from any 6 of its elements, e.g., from $b_{[19:24]} + c_{[19:24]}$ one can also recover $b_{25} + c_{25}, b_{26} + c_{26}, b_{27} + c_{27}$. This observation is the key to understanding the role of interference alignment in this construction. The effective number of *resolvable* undesired symbols is minimized due to interference alignment. For example, b_{19} and c_{19} are always aligned together into one symbol $b_{19} + c_{19}$ in all the downloaded equations. The two are unresolvable from each other and act as effectively one undesired symbol in the downloaded equations, thus reducing the effective number of undesired symbols, so that the same number of downloaded equations can be used to retrieve a greater number of desired symbols. Note also that desired symbols are always resolvable.

These values are plugged into the query structure derived previously.

DB1	DB2	DB3
a_1, a_2, a_3, a_4	a_5, a_6, a_7, a_8	$a_9, a_{10}, a_{11}, a_{12}$
b_1, b_2, b_3, b_4	b_5, b_6, b_7, b_8	$b_9, b_{10}, b_{11}, b_{12}$
c_1, c_2, c_3, c_4	c_5, c_6, c_7, c_8	$c_9, c_{10}, c_{11}, c_{12}$
$a_{13} + b_{13}$	$a_{15} + b_{15}$	$a_{21} + b_{17}$
$a_{14} + b_{14}$	$a_{16} + b_{16}$	$a_{22} + b_{18}$
$a_{17} + c_{13}$	$a_{19} + c_{15}$	$a_{23} + c_{17}$
$a_{18} + c_{14}$	$a_{20} + c_{16}$	$a_{24} + c_{18}$
$b_{19} + c_{19}$	$b_{21} + c_{21}$	$b_{23} + c_{23}$
$b_{20} + c_{20}$	$b_{22} + c_{22}$	$b_{24} + c_{24}$
$a_{25} + b_{25} + c_{25}$	$a_{26} + b_{26} + c_{26}$	$a_{27} + b_{27} + c_{27}$

Correctness is straightforward. Let us see why T-privacy holds. The queries for any $T = 2$ colluding databases are comprised of 18 variables from $a_{[1:27]}$, 12 variables from $b_{[1:18]}$, 6 variables from $b_{[19:27]}$, 12 variables from $c_{[1:18]}$ and 6 variables from $c_{[19:27]}$. Let the indices of these variables be denoted by the vectors $\mathcal{I}_a \in \mathbb{N}^{18\times 1}, \mathcal{I}_{b,12} \in \mathbb{N}^{12\times 1}, \mathcal{I}_{b,6} \in \mathbb{N}^{6\times 1}, \mathcal{I}_{c,12} \in \mathbb{N}^{12\times 1}$ and $\mathcal{I}_{c,6} \in \mathbb{N}^{6\times 1}$, respectively, so that,

$$a_{\mathcal{I}_a} = S_1[\mathcal{I}_a,:]W_1 \tag{27}$$
$$b_{\mathcal{I}_{b,12}} = \text{MDS}_{18\times 12}[\mathcal{I}_{b,12},:]S_2[(1:12),:]W_2 \tag{28}$$
$$b_{\mathcal{I}_{b,6}} = \text{MDS}_{9\times 6}[\mathcal{I}_{b,6},:]S_2[(13:18),:]W_2 \tag{29}$$
$$c_{\mathcal{I}_{c,12}} = \text{MDS}_{18\times 12}[\mathcal{I}_{c,12},:]S_3[(1:12),:]W_3 \tag{30}$$
$$c_{\mathcal{I}_{c,6}} = \text{MDS}_{9\times 6}[\mathcal{I}_{c,6},:]S_3[(13:18),:]W_3 \tag{31}$$

From Lemma 1, we have

$$(a_{\mathcal{I}_a}, (b_{\mathcal{I}_{b,12}};\ \ b_{\mathcal{I}_{b,6}}), (c_{\mathcal{I}_{c,12}};\ \ c_{\mathcal{I}_{c,6}})) \sim (S_1[(1:18),:]W_1, S_2[(1:18),:]W_2, S_3[(1:18),:]W_3) \tag{32}$$

Thus privacy is guaranteed. Finally, note that since 27 desired symbols are recovered from a total of 57 downloaded symbols, the rate achieved by this scheme is $27/57 = 9/19$, which matches the capacity for this setting.

4.5 Arbitrary Number of Messages K, Arbitrary Number of Databases N, Arbitrary Number of Colluding Databases T

4.5.1 Query Structure

For arbitrary K, N, T, we follow the same iterative procedure, briefly summarized below.

- Step 1: Initialization. Download T^{K-1} desired symbols each from the first T databases.
- Step 2: Invoke symmetry across databases to determine corresponding downloads from DB $T+1$ to DB N.
- Step 3: Invoke symmetry of messages to determine additional downloaded equations (comprised only of undesired symbols) from each database.
- Step 4: Consider the first T databases together. Divide the new external side information generated in the previous step evenly among the first T databases to determine the side-information budget per database. For each side information symbol allocated to a database create an additional query of the same form as the assigned side information (with new labels) combined with a new desired symbol.
- Step 5: Go back to Step 2 and run Step 2 to Step 4 a total of $(K-1)$ times.

4.5.2 Specialization

Let each message consist of N^K symbols from a sufficiently large finite field $\mathbb{F}_q$. The messages $W_1, \cdots, W_K \in \mathbb{F}_q^{N^K \times 1}$ are represented as $N^K \times 1$ vectors over $\mathbb{F}_q$. Let $S_1, \cdots, S_K \in \mathbb{F}_q^{N^K \times N^K}$ represent random matrices chosen privately by the user, independently and uniformly from all $N^K \times N^K$ full-rank matrices over $\mathbb{F}_q$. Suppose W_l, $l \in [1:K]$, is the desired message.

Consider any undesired message index $k \in [1:K]/\{l\}$, and all distinct $\Delta = 2^{K-2}$ subsets of $[1:K]$ that contain k and do not contain l. Assign distinct labels to each subset, e.g., $\mathcal{K}_1, \mathcal{K}_2, \cdots \mathcal{K}_\Delta$. For each $k \in [1:K]/\{l\}$, define the vector

$$\left[\begin{array}{c} x^{[k]}_{\mathcal{K}_1} \\ x^{[k]}_{\mathcal{K}_1 \cup \{l\}} \\ \hdashline x^{[k]}_{\mathcal{K}_2} \\ x^{[k]}_{\mathcal{K}_2 \cup \{l\}} \\ \hdashline \vdots \\ \hdashline x^{[k]}_{\mathcal{K}_\Delta} \\ x^{[k]}_{\mathcal{K}_\Delta \cup \{l\}} \end{array}\right] = \left[\begin{array}{cccc} \mathrm{MDS}_{\frac{N}{T}\alpha_1 \times \alpha_1} & 0 & 0 & 0 \\ \hdashline 0 & \mathrm{MDS}_{\frac{N}{T}\alpha_2 \times \alpha_2} & 0 & 0 \\ \hdashline 0 & \cdots & \ddots & 0 \\ \hdashline 0 & 0 & 0 & \mathrm{MDS}_{\frac{N}{T}\alpha_\Delta \times \alpha_\Delta} \end{array}\right] S_k[(1:TN^{K-1}), :]W_k$$

where $\alpha_i, i \in [1:\Delta]$ is defined as $N(N-T)^{|\mathcal{K}_i|-1}T^{K-|\mathcal{K}_i|}$, each $x^{[k]}_{\mathcal{K}_i}$ is a $\alpha_i \times 1$ vector, and each $x^{[k]}_{\mathcal{K}_i \cup \{l\}}$ is a $(\frac{N-T}{T})\alpha_i \times 1$ vector over $\mathbb{F}_q$.

Now consider the desired message index l, and all distinct $\delta = 2^{K-1}$ subsets of $[1:K]$ that contain l. Assign distinct labels to each subset, e.g., $\mathcal{L}_1, \mathcal{L}_2, \cdots \mathcal{L}_\delta$. Define the vector

$$\begin{bmatrix} x^{[l]}_{\mathcal{L}_1} \\ x^{[l]}_{\mathcal{L}_2} \\ \vdots \\ x^{[l]}_{\mathcal{L}_\delta} \end{bmatrix} = S_l W_l$$

where the length of $x^{[l]}_{\mathcal{L}_i}, i \in [1:\delta]$ is $N(N-T)^{|\mathcal{L}_i|-1}T^{K-|\mathcal{L}_i|}$.

For each non-empty subset $\mathcal{K} \subset [1:K]$ generate the query vector

$$\sum_{k\in\mathcal{K}} x^{[k]}_{\mathcal{K}} \tag{33}$$

Distribute the elements of the query vector evenly among the N databases. This completes the specialized construction of the queries.

The construction has K layers. Over the j-th layer, from each database, we download $(N-T)^{j-1}T^{K-j}\binom{K}{j}$ equations that are comprised of sums of j symbols, out of which $(N-T)^{j-1}T^{K-j}\binom{K-1}{j-1}$ involve desired data symbols. Our construction ensures that the interference $x^{[k]}_{\mathcal{K}_i\cup\{l\}}, k \neq l, i \in [1:\Delta]$ in the $(|\mathcal{K}_i|+1)$-th layer can be recovered from the corresponding symbols $x^{[k]}_{\mathcal{K}_i}$ in the $|\mathcal{K}_i|$-th layer. Therefore correctness is guaranteed.

Let us see why privacy holds. The queries for any T colluding databases are comprised of TN^{K-1} variables from each $x^{[k]}, k \in [1:K]$. In particular, $\forall k \neq l$, the variables from $x^{[k]}$ consist of α_i variables out of $\frac{N}{T}\alpha_i$ variables $x^{[k]}_{\mathcal{K}_i}, x^{[k]}_{\mathcal{K}_i\cup\{l\}}$, for each set $\mathcal{K}_i, i \in [1:\Delta]$. Note that these α_i variables are generated by the generator matrix of a $(\frac{N}{T}\alpha_i, \alpha_i)$ MDS code, so that they have full rank. Let the indices of the appeared variables be denoted by the vectors $\mathcal{I}_{x^{[k]}} \in \mathbb{N}^{TN^{K-1}\times 1}, \forall k \in [1:K]$. From Lemma 1, we have

$$x^{[k]}_{\mathcal{I}_{x^{[k]}}} \sim S_k[(1:TN^{K-1}),:]W_k \tag{34}$$

which in turn implies that $S_k[(1:TN^{K-1}),:]$ are independent and identically distributed. Thus privacy is guaranteed.

Finally, we compute the ratio of the number of desired symbols to the number of total downloaded symbols,

$$R = \frac{N}{N}\frac{T^{K-1}\binom{K-1}{0} + (N-T)T^{K-2}\binom{K-1}{1} + (N-T)^2T^{K-3}\binom{K-1}{2} + \cdots + (N-T)^{K-1}\binom{K-1}{K-1}}{T^{K-1}\binom{K}{1} + (N-T)T^{K-2}\binom{K}{2} + (N-T)^2T^{K-3}\binom{K}{3} + \cdots + (N-T)^{K-1}\binom{K}{K}} \tag{35}$$

$$= \frac{N}{N} \frac{N^{K-1}}{\frac{1}{N-T}\left[(N-T)T^{K-1}\binom{K}{1} + (N-T)^2 T^{K-2}\binom{K}{2} + \cdots + (N-T)^K\binom{K}{K}\right]} \tag{36}$$

$$= \frac{\frac{1}{N}N^K}{\frac{1}{N-T}\left(N^K - T^K\right)} = \frac{1-\frac{T}{N}}{1-\frac{T^K}{N^K}} \tag{37}$$

$$= \left(1 + \frac{T}{N} + \frac{T^2}{N^2} + \cdots + \frac{T^{K-1}}{N^{K-1}}\right)^{-1} \tag{38}$$

Thus, the PIR rate achieved by the scheme always matches the capacity.

5 Proof of Theorem 1: Converse

For compact notation, let us define

$$\mathcal{Q} \overset{\triangle}{=} \{Q_n^{[k]} : k \in [1:K], n \in [1:N]\} \tag{39}$$

$$A_{\mathcal{I}}^{[k]} \overset{\triangle}{=} \{A_n^{[k]} : n \in \mathcal{I}\} \tag{40}$$

$$\mathcal{H}_T \overset{\triangle}{=} \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} \frac{H(A_{\mathcal{T}}|\mathcal{Q})}{T}, \mathcal{T} \subset [1:N] \tag{41}$$

We first state Han's inequality (Theorem 17.6.1 in [35]), which will be used later and is described here for the sake of completeness.

Theorem 3 *(Han's inequality)*

$$\mathcal{H}_T \geq \frac{H(A_1^{[k]}, A_2^{[k]}, \cdots, A_N^{[k]}|\mathcal{Q})}{N} \tag{42}$$

We next proceed to the converse proof. The proof of outer bound for Theorem 1 is based on an induction argument. To set up the induction, we will prove the outer bound for $K = 1$ and for $K = 2$, each for arbitrary N, T, and then proceed to the case of arbitrary K.

5.1 $K = 1$ Message, N Databases

$$L = H(W_1) = H(W_1|\mathcal{Q}) \tag{43}$$

$$\leq I(A_1^{[1]}, A_2^{[1]}, \cdots, A_N^{[1]}; W_1|\mathcal{Q}) \tag{44}$$

$$= H(A_1^{[1]}, A_2^{[1]}, \cdots, A_N^{[1]}|\mathcal{Q}) \tag{45}$$

$$\leq N\mathcal{H}_T \tag{46}$$

$$\leq \sum_{n=1}^{N} H(A_n|\mathcal{Q}) \tag{47}$$

$$\Rightarrow R = \frac{L}{\sum_{n=1}^{N} H(A_n)} \leq \frac{L}{\sum_{n=1}^{N} H(A_n|\mathcal{Q})} \leq 1 \tag{48}$$

where (46) follows from Han's inequality, and (47) is due to the property that dropping conditioning does not reduce entropy.

5.2 $K = 2$ Messages, N Databases

Consider $\mathcal{T} \subset [1:N]$ with cardinality $|\mathcal{T}| = T$. Denote the complement of $\mathcal{T}$ as $\overline{\mathcal{T}}$.

Since the queries and answer strings for any T databases are independent of the message index, we will denote $A_{\mathcal{T}}^{[k]}$ simply as $A_{\mathcal{T}}, \forall k \in [1:K]$. From $A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, A_{\overline{\mathcal{T}}}^{[2]}, \mathcal{Q}$, we can decode W_1, W_2.

$$\begin{aligned}
2L = H(W_1, W_2) &= H(W_1, W_2|\mathcal{Q}) && (49)\\
&\leq I(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, A_{\overline{\mathcal{T}}}^{[2]}; W_1, W_2|\mathcal{Q}) && (50)\\
&= H(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, A_{\overline{\mathcal{T}}}^{[2]}|\mathcal{Q}) && (51)\\
&= H(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}|\mathcal{Q}) + H(A_{\overline{\mathcal{T}}}^{[2]}|A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, \mathcal{Q}) && (52)\\
&= H(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}|\mathcal{Q}) + H(A_{\overline{\mathcal{T}}}^{[2]}|A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, W_1, \mathcal{Q}) && (53)\\
&\leq N\mathcal{H}_T + H(A_{\overline{\mathcal{T}}}^{[2]}|A_{\mathcal{T}}, W_1, \mathcal{Q}) && (54)\\
&= N\mathcal{H}_T + H(A_{[1:N]}^{[2]}|W_1, \mathcal{Q}) - H(A_{\mathcal{T}}|W_1, \mathcal{Q}) && (55)\\
&= N\mathcal{H}_T + L - H(A_{\mathcal{T}}|W_1, \mathcal{Q}) && (56)
\end{aligned}$$

where (51) is due to the fact that the answering strings are deterministic functions of the messages and queries. (53) is due to the fact that W_1 is a function of $(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, \mathcal{Q})$. In (56), the second term follows from the fact that from $A_{[1:N]}^{[2]}$, we can decode W_2.

Consider (56) for all subsets of $[1:N]$ that have exactly T elements and average over all such subsets. We have

$$\begin{aligned}
2L &\leq N\mathcal{H}_T + L - \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} H(A_{\mathcal{T}}|W_1, \mathcal{Q}) && (57)\\
&\leq N\mathcal{H}_T + L - TH(A_{[1:N]}^{[2]}|W_1, \mathcal{Q})/N && (58)\\
&= N\mathcal{H}_T + L - TL/N && (59)\\
\Rightarrow L\left(1 + \frac{T}{N}\right) &\leq N\mathcal{H}_T \leq \sum_{n=1}^{N} H(A_n|\mathcal{Q}) && (60)\\
\Rightarrow R &= \frac{L}{\sum_{n=1}^{N} H(A_n)} \leq \frac{L}{\sum_{n=1}^{N} H(A_n|\mathcal{Q})} \leq \left(1 + \frac{T}{N}\right)^{-1} && (61)
\end{aligned}$$

where (58) follows from the conditional version of Han's inequality, and (59) is due to the fact that from $A_{[1:N]}^{[2]}$, we can decode W_2. The outer bound proof for $K = 2$ messages setting is complete.

5.3 $K \geq 3$ Messages, N Databases

Consider $\mathcal{T} \subset [1:N]$ with cardinality $|\mathcal{T}| = T$. From $A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, \cdots, A_{\overline{\mathcal{T}}}^{[K]}, \mathcal{Q}$, we can decode all K messages $W_1, \cdots, W_K$.

$$\begin{aligned}
&KL = H(W_1, \cdots, W_K|\mathcal{Q}) && (62)\\
\leq\ & I(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, \cdots, A_{\overline{\mathcal{T}}}^{[K]}; W_1, \cdots, W_K|\mathcal{Q}) && (63)\\
=\ & H(A_{\mathcal{T}}, A_{\overline{\mathcal{T}}}^{[1]}, \cdots, A_{\overline{\mathcal{T}}}^{[K]}|\mathcal{Q}) && (64)
\end{aligned}$$

$$= H(A_{\mathcal{T}}, A^{[1]}_{\overline{\mathcal{T}}}|\mathcal{Q}) + H(A^{[2]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}|A_{\mathcal{T}}, A^{[1]}_{\overline{\mathcal{T}}}, \mathcal{Q}) \tag{65}$$

$$\leq N\mathcal{H}_T + H(A^{[2]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}|A_{\mathcal{T}}, A^{[1]}_{\overline{\mathcal{T}}}, W_1, \mathcal{Q}) \tag{66}$$

$$\leq N\mathcal{H}_T + H(A^{[2]}_{\overline{\mathcal{T}}}|A_{\mathcal{T}}, W_1, \mathcal{Q}) + H(A^{[3]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}|A_{\mathcal{T}}, A^{[2]}_{\overline{\mathcal{T}}}, W_1, \mathcal{Q}) \tag{67}$$

$$\leq N\mathcal{H}_T + \sum_{n\in\overline{\mathcal{T}}} H(A^{[2]}_n|A_{\mathcal{T}}, W_1, \mathcal{Q}) + H(A^{[3]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}|A_{\mathcal{T}}, W_1, W_2, \mathcal{Q}) \tag{68}$$

$$= N\mathcal{H}_T + \sum_{n\in\overline{\mathcal{T}}} H(A^{[2]}_n|A_{\mathcal{T}}, W_1, \mathcal{Q}) + H(A_{\mathcal{T}}, A^{[3]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}|W_1, W_2, \mathcal{Q}) - H(A_{\mathcal{T}}|W_1, W_2, \mathcal{Q}) \tag{69}$$

$$= N\mathcal{H}_T + \sum_{n\in\overline{\mathcal{T}}} H(A^{[2]}_n|A_{\mathcal{T}}, W_1, \mathcal{Q}) + (K-2)L - H(A_{\mathcal{T}}|W_1, W_2, \mathcal{Q}) \tag{70}$$

where (66) is due to the fact that W_1 is a function of $(A_{\mathcal{T}}, A^{[1]}_{\overline{\mathcal{T}}}, \mathcal{Q})$. (68) follows from the fact that W_2 is a function of $(A_{\mathcal{T}}, A^{[2]}_{\overline{\mathcal{T}}}, \mathcal{Q})$. In (70), the third term is due to the fact that from $A_{\mathcal{T}}, A^{[3]}_{\overline{\mathcal{T}}}, \cdots, A^{[K]}_{\overline{\mathcal{T}}}$, we can decode $W_3, \cdots, W_K$.

Consider (70) for all subsets of $[1:N]$ that have exactly T elements and average over all such subsets. We have

$$KL - (K-2)L + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} H(A_{\mathcal{T}}|W_1, W_2, \mathcal{Q}) \tag{71}$$

$$\leq N\mathcal{H}_T + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} \sum_{n\in\overline{\mathcal{T}}} H(A^{[2]}_n|A_{\mathcal{T}}, W_1, \mathcal{Q}) \tag{72}$$

$$\leq N\mathcal{H}_T + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} \sum_{n\in\overline{\mathcal{T}}} \sum_{\mathcal{T}'\subset\mathcal{T}, |\mathcal{T}'|=T-1} \frac{H(A_n|A_{\mathcal{T}'}, W_1, \mathcal{Q})}{T} \tag{73}$$

$$= N\mathcal{H}_T + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}':|\mathcal{T}'|=T-1} \sum_{n\notin\mathcal{T}'} (N-T)\frac{H(A_n|A_{\mathcal{T}'}, W_1, \mathcal{Q})}{T} \tag{74}$$

$$= N\mathcal{H}_T + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} \sum_{n\in\mathcal{T}} (N-T)\frac{H(A_n|A_{\mathcal{T}/\{n\}}, W_1, \mathcal{Q})}{T} \tag{75}$$

$$\leq N\mathcal{H}_T + \frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} (N-T)\frac{H(A_{\mathcal{T}}|W_1, \mathcal{Q})}{T} \tag{76}$$

$$\leq N\mathcal{H}_T + (\frac{N}{T} - 1)\frac{1}{\binom{N}{T}} \sum_{\mathcal{T}:|\mathcal{T}|=T} H(A_{\mathcal{T}}, A^{[1]}_{\overline{\mathcal{T}}}|W_1, \mathcal{Q}) \tag{77}$$

$$= N\mathcal{H}_T + (\frac{N}{T} - 1)\left(H(A^{[1]}_{[1:N]}, W_1|\mathcal{Q}) - H(W_1|\mathcal{Q})\right) \tag{78}$$

$$= N\mathcal{H}_T + (\frac{N}{T} - 1)\left(H(A^{[1]}_{[1:N]}|\mathcal{Q}) + \underbrace{H(W_1|A^{[1]}_{[1:N]}, \mathcal{Q})}_{=0} - H(W_1)\right) \tag{79}$$

$$\leq N\mathcal{H}_T + (\frac{N}{T} - 1)\,(N\mathcal{H}_T - L) \tag{80}$$

$$= \frac{N^2}{T}\mathcal{H}_T - (\frac{N}{T} - 1)L \tag{81}$$

$$\Rightarrow N\mathcal{H}_T \quad \geq \quad L\left(1+\frac{T}{N}\right)+\frac{T^2}{N^2}N\frac{1}{\binom{N}{T}}\sum_{\mathcal{T}:|\mathcal{T}|=T}\frac{H(A_{\mathcal{T}}|W_1,W_2,\mathcal{Q})}{T} \tag{82}$$

where (73) follows because dropping conditioning increases entropy. In (73) we drop the superscript of $A_n^{[2]}$ and write it simply as A_n because $\{A_n^{[2]}, A_{\mathcal{T}'}\}$ contain responses from only T databases, which must be independent of the desired message index because of the requirement of T-privacy. Equation (74) follows from the observation that given $\mathcal{T}', n$, there are $(N-T)$ feasible choices of $\mathcal{T}$, i.e., those choices of $\mathcal{T}$ which contain $\mathcal{T}'$ and do not contain n. Thus, the same term $H(A_n|A_{\mathcal{T}'}, W_1, Q)$ repeats in the overall summation a total of $(N-T)$ times. To see why (76) holds, suppose we write $\mathcal{T} = \{n_1, n_2, \cdots, n_T\}$. By the chain rule of entropies,

$$\begin{aligned} H(A_{\mathcal{T}}|W_1,\mathcal{Q}) &= \sum_{t=1}^{T} H(A_{n_t}|A_{n_1},A_{n_2},\cdots,A_{n_{t-1}},W_1,\mathcal{Q}) && (83)\\ &\geq \sum_{t=1}^{T} H(A_{n_t}|A_{\mathcal{T}/n_t},W_1,\mathcal{Q}) && (84)\end{aligned}$$

In (79), the third term equals 0, because from $A_{[1:N]}^{[1]}$, we can decode W_1.

To proceed, we note that for the last term of (82), conditioning on W_1, W_2, the setting reduces to a PIR problem with $K-2$ messages and N databases. Thus, (82) sets up an induction argument, which claims that for the K messages setting,

$$N\mathcal{H}_T \geq L\left(1+\frac{T}{N}+\cdots+\frac{T^{K-1}}{N^{K-1}}\right) \tag{85}$$

We have proved the basis cases of $K=1$ and $K=2$ in (46) and (60). Suppose now the bound (85) holds for $K-2$. Then plugging in (82), we have that the bound (85) holds for K. Since both the basis and the inductive step have been performed, by mathematical induction, we have proved that (85) holds for all K. The desired outer bound follows as

$$R \quad = \quad \frac{L}{\sum_{n=1}^{N} H(A_n)} \leq \frac{L}{\sum_{n=1}^{N} H(A_n|\mathcal{Q})} \leq \frac{L}{N\mathcal{H}_T} \leq \left(1+\frac{T}{N}+\cdots+\frac{T^{K-1}}{N^{K-1}}\right)^{-1} \tag{86}$$

Thus, the proof of the outer bound is complete.

6 Proof of Theorem 2

Clearly the capacity of robust T-private PIR cannot be larger than the capacity of T-private PIR. Therefore, we only need to prove that the capacity of T-private PIR can be achieved in the robust PIR setting. To this end, we build upon the scheme presented in Section 4 as follows.

As before, let each message consist of N^K symbols from a sufficiently large finite field $\mathbb{F}_q$. The messages $W_1, \cdots, W_K \in \mathbb{F}_q^{N^K \times 1}$ are represented as $N^K \times 1$ vectors over $\mathbb{F}_q$. Let $S_1, \cdots, S_K \in \mathbb{F}_q^{N^K \times N^K}$ represent random matrices chosen privately by the user, independently and uniformly from all $N^K \times N^K$ full-rank matrices over $\mathbb{F}_q$. Suppose W_l, $l \in [1:K]$, is the desired message.

Consider any undesired message index $k \in [1:K]/\{l\}$, and all distinct $\Delta = 2^{K-2}$ subsets of $[1:K]$ that contain k and do not contain l. Assign distinct labels to each subset, e.g., $\mathcal{K}_1, \mathcal{K}_2, \cdots \mathcal{K}_\Delta$. For each $k \in [1:K]/\{l\}$, define the vector

$$\begin{bmatrix} x^{[k]}_{\mathcal{K}_1} \\ x^{[k]}_{\mathcal{K}_1 \cup \{l\}} \\ \hdashline x^{[k]}_{\mathcal{K}_2} \\ x^{[k]}_{\mathcal{K}_2 \cup \{l\}} \\ \hdashline \vdots \\ \hdashline x^{[k]}_{\mathcal{K}_\Delta} \\ x^{[k]}_{\mathcal{K}_\Delta \cup \{l\}} \end{bmatrix} = \begin{bmatrix} \text{MDS}_{\frac{M}{T}\alpha_1 \times \alpha_1} & 0 & 0 & 0 \\ \hdashline 0 & \text{MDS}_{\frac{M}{T}\alpha_2 \times \alpha_2} & 0 & 0 \\ \hdashline 0 & \cdots & \ddots & 0 \\ \hdashline 0 & 0 & 0 & \text{MDS}_{\frac{M}{T}\alpha_\Delta \times \alpha_\Delta} \end{bmatrix} S_k[(1:TN^{K-1}), :]W_k$$

where $\alpha_i, i \in [1:\Delta]$ is defined as $N(N-T)^{|\mathcal{K}_i|-1}T^{K-|\mathcal{K}_i|}$, each $x^{[k]}_{\mathcal{K}_i}$ is a $\frac{M}{N}\alpha_i \times 1$ vector, and each $x^{[k]}_{\mathcal{K}_i \cup \{l\}}$ is a $\frac{M}{N}(\frac{N-T}{T})\alpha_i \times 1$ vector over $\mathbb{F}_q$.

Now consider the desired message index l, and all distinct $\delta = 2^{K-1}$ subsets of $[1:K]$ that contain l. Assign distinct labels to each subset, e.g., $\mathcal{L}_1, \mathcal{L}_2, \cdots \mathcal{L}_\delta$. Define the vector

$$\begin{bmatrix} x^{[l]}_{\mathcal{L}_1} \\ x^{[l]}_{\mathcal{L}_2} \\ \vdots \\ x^{[l]}_{\mathcal{L}_\delta} \end{bmatrix} = \text{MDS}_{\frac{M}{N}N^K \times N^K} S_l W_l$$

where the length of $x^{[l]}_{\mathcal{L}_i}, i \in [1:\delta]$ is $M(N-T)^{|\mathcal{L}_i|-1}T^{K-|\mathcal{L}_i|}$.

For each non-empty subset $\mathcal{K} \subset [1:K]$ generate the query vector

$$\sum_{k \in \mathcal{K}} x^{[k]}_{\mathcal{K}} \tag{87}$$

Distribute the elements of the query vector evenly among the M databases. This completes the construction of the queries.

Suppose the user collects answering strings from any N databases. For each set $\mathcal{K}_i$, from N databases, we download α_i symbols from $x^{[k]}_{\mathcal{K}_i}, k \neq l, i \in [1:\Delta]$, from which we can recover the interference $x^{[k]}_{\mathcal{K}_i \cup \{l\}}$, as they are generated by the generator matrix of a $(\frac{M}{T}\alpha_i, \alpha_i)$ MDS code. After subtracting out all the interference, we are left with N^K desired symbols, from which we can recover the desired message, as the symbols are generated by the generator matrix of a $(\frac{M}{N}N^K, N^K)$ MDS code. Therefore correctness is guaranteed.

Let us see why privacy holds. The queries for any T colluding databases are comprised of TN^{K-1} variables from each $x^{[k]}, k \in [1:K]$. When $k = l$, the TN^{K-1} desired symbols are generated by the generator matrix of a $(\frac{M}{N}N^K, N^K)$ MDS code such that these symbols have full rank. For each $k \neq l$, the TN^{K-1}variables from $x^{[k]}$ consist of α_i variables out of $\frac{M}{T}\alpha_i$ variables $x^{[k]}_{\mathcal{K}_i}, x^{[k]}_{\mathcal{K}_i \cup \{l\}}$, for each set $\mathcal{K}_i, i \in [1:\Delta]$. Note that these α_i variables are generated by the generator matrix of a $(\frac{M}{T}\alpha_i, \alpha_i)$ MDS code, so that they have full rank. Let the indices of the appeared variables be denoted by the vectors $\mathcal{I}_{x^{[k]}} \in \mathbb{N}^{TN^{K-1} \times 1}, \forall k \in [1:K]$. From Lemma 1, we have

$$x^{[k]}_{\mathcal{I}_{x^{[k]}}} \sim S_k[(1:TN^{K-1}), :]W_k \tag{88}$$

which in turn implies that $S_k[(1:TN^{K-1}), :]$ are independent and identically distributed. Thus privacy is guaranteed. Finally, the rate achieved is the same as that achieved in the setting without the robustness constraint. This completes the proof.

7 Conclusion

We characterize the capacity of robust T-private PIR with arbitrary number of messages, arbitrary number of (responding) databases, and arbitrary privacy level. Let us conclude with a few observations. First, while in this paper we adopt the zero error framework, we note that our converse extends in a straightforward manner to the ϵ-error framework as well, where the probability of error is only required to approach zero as the message size approaches infinity. Therefore, for robust T-private PIR, the ϵ-error capacity is the same as the zero error capacity. Second, recall that the capacity achieving scheme for PIR in our prior work [25] had a remarkable feature that if some of the messages were eliminated and the scheme projected onto a subset of messages, it remained capacity optimal for that subset of messages. The same phenomenon is observed for our achievable scheme for robust T-private PIR. On the other hand, an important point of distinction of the previous achievable scheme in [25] from the achievable scheme in this paper is that the former directly uses each available side information symbol individually, whereas here we need MDS coded side information (uncoded side information symbols do not suffice). This is because of the T-privacy constraint which simultaneously creates multiple perspectives of external side-information depending upon which subset of databases decides to collude.

Finally, we note that since we focus only on download cost, upload cost is not optimized in this work. However, even with T-privacy, significant optimizations of upload cost are possible through refinements of our achievable scheme. For example, the symbols may be grouped in a manner that randomizations are needed only within smaller groups, which may reduce the number of possible queries, and the size of the field of operations significantly. For example, consider the achievable scheme for $K = 2, N = 3, T = 2$ that was presented in Section 4.1, where each message is comprised of 9 symbols. We will operate over $\mathbb{F}_2$. Suppose we divide the 9 bits into 3 groups of 3 bits each, and label the groups so that A_1 represents the first three bits of W_1, A_2 the next three and A_3 represents the last three bits from W_1. Similarly, let B_1, B_2, B_3 represent three groups of three bits each from W_2. Now, for any group of 3 bits, say $X = (x_1, x_2, x_3)$, let $X(1), X(2), X(3)$ represent three randomly chosen linearly independent elements from the set $\{x_1, x_2, x_3, x_1 + x_2, x_1 + x_3, x_2 + x_3, x_1 + x_2 + x_3\}$, i.e., selected uniformly from among the choices that do not sum to zero in $\mathbb{F}_2$. This essentially means that $X(1), X(2)$ may be freely chosen as any two distinct elements of the set and then $X(3)$ is chosen uniformly from the 4 elements that are not $X(1), X(2)$ or $X(1)+X(2)$. The queries are constructed as follows.

DB1	DB2	DB3
$A_1(1), A_2(1)$	$A_2(2), A_3(2)$	$A_3(3), A_1(3)$
$B_1(1), B_2(1)$	$B_2(2), B_3(2)$	$B_3(1+2), B_1(1+2)$
$A_3(1)+B_3(1)$	$A_1(2)+B_1(2)$	$A_2(3)+B_2(1+2)$

where we use the notation $X(1+2) = X(1)+X(2)$ for brevity. Note that for the undesired symbols B, we used the $(2,3)$ MDS code $(B(1), B(2)) \longrightarrow (B(1), B(2), B(1+2))$ within each group. Due to the grouping of symbols the upload cost is significantly reduced. Moreover, because of the grouping we are able to operate over a smaller field. Whereas the original scheme presented in Section 4.1 uses $(6,9)$ MDS codes which do not exist over $\mathbb{F}_2$, the refined example presented above uses only a $(2,3)$ MDS code which does exist over $\mathbb{F}_2$. As illustrated by this example, optimizations of upload costs as well as symbol size remain interesting avenues for future work.

References

[1] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, "Protecting data privacy in private information retrieval schemes," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM, 1998, pp. 151–160.

[2] J. Feigenbaum, "Encrypting problem instances," in *Advances in Cryptology – CRYPTO85 Proceedings*. Springer, 1985, pp. 477–488.

[3] M. Abadi, J. Feigenbaum, and J. Kilian, "On hiding information from an oracle," in *Proceedings of the nineteenth annual ACM symposium on Theory of computing*. ACM, 1987, pp. 195–203.

[4] D. Beaver and J. Feigenbaum, "Hiding instances in multioracle queries," in *STACS 90*. Springer, 1990, pp. 37–48.

[5] D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, "Locally random reductions: Improvements and applications," *Journal of Cryptology*, vol. 10, no. 1, pp. 17–36, 1997.

[6] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, pp. 612–613, 1979.

[7] A. Beimel, Y. Ishai, E. Kushilevitz, and I. Orlov, "Share Conversion and Private Information Retrieval," in *Proceedings of the 27th Annual Conference on Computational Complexity*. IEEE, 2012, pp. 258–268.

[8] S. Yekhanin, "Locally Decodable Codes and Private Information Retrieval Schemes," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.

[9] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Batch codes and their applications," in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM, 2004, pp. 262–271.

[10] Y. Ishai and E. Kushilevitz, "On the hardness of information-theoretic multiparty computation," in *Advances in Cryptology-EUROCRYPT 2004*. Springer, 2004, pp. 439–455.

[11] H. Sun and S. A. Jafar, "Blind Interference Alignment for Private Information Retrieval," *arXiv preprint arXiv:1601.07885*, 2016.

[12] P. Gopalan, C.Huang, H. Simitci, and S. Yekhanin, "On the Locality of Codeword Symbols," *IEEE Transactions on Information Theory*, vol. 58, no. 11, pp. 6925–6934, Nov. 2012.

[13] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proceedings of the IEEE*, vol. 99, pp. 476–489, 2011. [Online]. Available: http://arxiv.org/abs/1004.4438

[14] Y. Birk and T. Kol, "Coding on demand by an informed source (ISCOD) for efficient broadcast of different supplemental data to caching clients," *IEEE Trans. on Information Theory*, vol. 52, no. 6, pp. 2825–2830, June 2006.

[15] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inform. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[16] W. Gasarch, "A Survey on Private Information Retrieval," in *Bulletin of the EATCS.* Citeseer, 2004.

[17] S. Yekhanin, "Private Information Retrieval," *Communications of the ACM*, vol. 53, no. 4, pp. 68–73, 2010.

[18] R. Ostrovsky and W. E. Skeith III, "A Survey of Single-database Private Information Retrieval: Techniques and Applications," in *Public Key Cryptography–PKC 2007.* Springer, 2007, pp. 393–411.

[19] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," in *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, 1995, pp. 41–50.

[20] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, "Private Information Retrieval," *Journal of the ACM (JACM)*, vol. 45, no. 6, pp. 965–981, 1998.

[21] A. Ambainis, "Upper bound on the communication complexity of private information retrieval," in *Automata, Languages and Programming.* Springer, 1997, pp. 401–407.

[22] A. Beimel, Y. Ishai, E. Kushilevitz, and J.-F. Raymond, "Breaking the $\mathcal{O}(n^{1/(2k-1)})$ barrier for information-theoretic Private Information Retrieval," in *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science.* IEEE, 2002, pp. 261–270.

[23] Z. Dvir and S. Gopi, "2-Server PIR with Sub-polynomial Communication," *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC'15*, pp. 577–584, 2015.

[24] T. H. Chan, S.-W. Ho, and H. Yamamoto, "Private Information Retrieval for Coded Storage," *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, pp. 2842–2846, 2015.

[25] H. Sun and S. A. Jafar, "The Capacity of Private Information Retrieval," *arXiv preprint arXiv:1602.09134*, 2016.

[26] A. Beimel, Y. Ishai, and E. Kushilevitz, "General constructions for information-theoretic private information retrieval," *Journal of Computer and System Sciences*, vol. 71, no. 2, pp. 213–247, 2005.

[27] O. Barkol, Y. Ishai, and E. Weinreb, "On locally decodable codes, self-correctable codes, and t-private PIR," *Algorithmica*, vol. 58, no. 4, pp. 831–859, 2010.

[28] A. Beimel and Y. Stahl, "Robust information-theoretic private information retrieval," *Journal of Cryptology*, vol. 20, no. 3, pp. 295–321, 2007.

[29] Y. Gertner, S. Goldwasser, and T. Malkin, "A random server model for private information retrieval," in *Randomization and Approximation Techniques in Computer Science*. Springer, 1998, pp. 200–217.

[30] G. Fanti and K. Ramchandran, "Efficient private information retrieval over unsynchronized databases," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 9, no. 7, pp. 1229–1239, 2015.

[31] A. Beimel, Y. Ishai, and T. Malkin, "Reducing the servers computation in private information retrieval: PIR with preprocessing," in *Advances in CryptologyCRYPTO 2000*. Springer, 2000, pp. 55–73.

[32] N. Shah, K. Rashmi, and K. Ramchandran, "One Extra Bit of Download Ensures Perfectly Private Information Retrieval," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2014, pp. 856–860.

[33] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with low storage overhead," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 2852–2856.

[34] R. Tajeddine and S. E. Rouayheb, "Private Information Retrieval from MDS Coded Data in Distributed Storage Systems," *arXiv preprint arXiv:1602.01458*, 2016.

[35] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 2006.