Private Information Retrieval from MDS Coded Data With Colluding Servers: Settling a Conjecture by Freij-Hollanti *et al.*

Hua Sun, *Member, IEEE*, and Syed Ali Jafar, *Fellow, IEEE*

***Abstract*— A (K, N, T, K_c) instance of private information retrieval from MDS coded data with colluding servers (in short, MDS-TPIR), is comprised of K messages and N distributed servers. Each message is separately encoded through a (K_c, N) MDS storage code. A user wishes to retrieve one message, as efficiently as possible, while revealing no information about the desired message index to any colluding set of up to T servers. The fundamental limit on the efficiency of retrieval, i.e., the capacity of MDS-TPIR is known only at the extremes where either T or K_c belongs to $\{1, N\}$. The focus of this work is a recent conjecture by Freij-Hollanti, Gnilke, Hollanti, and Karpuk which offers a general capacity expression for MDS-TPIR. We prove that the conjecture is false by presenting as a counterexample a PIR scheme for the setting $(K, N, T, K_c) = (2, 4, 2, 2)$, which achieves the rate 3/5, exceeding the conjectured capacity, 4/7. Insights from the counterexample lead us to capacity characterizations for various instances of MDS-TPIR, including all cases with $(K, N, T, K_c) = (2, N, T, N - 1)$, where N and T can be arbitrary.**

***Index Terms*— Capacity, private information retrieval, colluding servers, MDS coded data.**

I. Introduction

PRIVATE Information Retrieval (PIR) is the problem of retrieving one out of K messages from N distributed servers (each stores all K messages) in such a way that any individual server learns no information about which message is being retrieved. The rate[1] of a PIR scheme is the ratio of the number of bits of the desired message to the total number of bits downloaded from all servers. The supremum of achievable rates is the capacity of PIR. The capacity of PIR was shown in [2] to be

$$C_{\text{PIR}} = \left(1 + \frac{1}{N} + \frac{1}{N^2} + \cdots + \frac{1}{N^{K-1}}\right)^{-1} \tag{1}$$

The capacity of several variants of PIR has also since been characterized in [2]–[6].

Manuscript received February 2, 2017; revised September 20, 2017; accepted November 27, 2017. Date of publication December 4, 2017; date of current version January 18, 2018. This work was supported in part by NSF under Grant CCF-1617504 and Grant CNS-1731384, in part by ONR under Grant N00014-16-1-2629 and Grant N00014-18-1-2057, and in part by ARO under Grant W911NF-16-1-0215. This paper was presented in part at the 2017 IEEE International Symposium on Information Theory.

H. Sun is with the Department of Electrical Engineering, University of North Texas, Denton, TX 76203 USA (e-mail: hua.sun@unt.edu).

S. A. Jafar is with the Center for Pervasive Communications and Computing, Department of Electrical Engineering and Computer Science, University of California at Irvine, Irvine, CA 92697 USA (e-mail: syed@uci.edu).

Communicated by P. Sadeghi, Associate Editor for Coding Techniques.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2017.2779454

The focus of this work is on a recent conjecture by Freij-Hollanti, Gnilke, Hollanti and Karpuk (FGHK conjecture, in short) [7] which offers a capacity expression for a generalized form of PIR, called MDS-TPIR. MDS-TPIR involves two additional parameters: K_c and T, which generalize the storage and privacy constraints, respectively. Instead of replication, each message is encoded through a (K_c, N) MDS storage code, so that the information stored at any K_c servers is exactly enough to recover all K messages. Privacy must be preserved not just from each individual server, but from any colluding set of up to T servers. MDS-TPIR is a generalization of PIR, because setting both $T = 1$ and $K_c = 1$ reduces MDS-TPIR to the original PIR problem for which the capacity is already known (see (1)).

The capacity of MDS-TPIR is known only at the degenerate extremes – when either T or K_c takes the value 1 or N. If either T or K_c is equal to N then by analogy to the single server setting it follows immediately that the user must download all messages, i.e., the capacity is $1/K$. If $K_c = 1$ or $T = 1$, then the problem specializes to TPIR, and MDS-PIR, respectively. The capacity of TPIR ($K_c = 1$) was shown in [3] to be

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

The capacity of MDS-PIR ($T = 1$) was characterized by Banawan and Ulukus in [6], as

$$C_{\text{MDS-PIR}} = \left(1 + \frac{K_c}{N} + \frac{K_c^2}{N^2} + \cdots + \frac{K_c^{K-1}}{N^{K-1}}\right)^{-1} \tag{3}$$

It is notable that K_c and T play similar roles in the two capacity expressions.

The capacity achieving scheme of Banawan and Ulukus [6] improved upon a scheme proposed earlier by Tajeddine and Rouayheb [8]. Tajeddine and Rouayheb also proposed an achievable scheme for MDS-TPIR for the $T = 2$ setting. The scheme was generalized by Freij-Hollanti *et al.* [7] to the (K, N, T, K_c) setting, $T + K_c \le N$, where it achieves the rate $1 - \frac{T+K_c-1}{N}$. Remarkably, the rate achieved by this scheme

[1] PIR originated in theoretical computer science [1], where the focus is typically on studying the growth rate of communication complexity (total upload and download cost) for a large number ($K \to \infty$) of short (typically one bit) messages that are stored on replicated servers. The information theoretic capacity formulations of PIR focus on an arbitrary number (K) of large messages (length of each message, $L \to \infty$), such that the download cost dominates the communication complexity, often with coded storage. For a detailed discussion, see the introduction of [2].

does not depend on the number of messages, K. In support of the plausible asymptotic ($K \to \infty$) optimality of their scheme, and based on the intuition from existing capacity expressions for PIR, MDS-PIR and TPIR, Freij-Hollanti et al. conjecture that if $T + K_c \le N$, then the capacity of MDS-TPIR is given by the following expression.

FGHK CONJECTURE [7]:

$$C^{\text{conj}}_{\text{MDS-TPIR}} = \left(1+\frac{T+K_c-1}{N}+\cdots+\frac{(T+K_c-1)^{K-1}}{N^{K-1}}\right)^{-1} \tag{4}$$

The conjecture is appealing for its generality and elegance as it captures all four parameters, K, N, T, K_c in a compact form. T and K_c appear as interchangeable terms, and the capacity expression appears to be a natural extension of the capacity expressions for TPIR and MDS-PIR. Indeed, the conjectured capacity recovers the known capacity of TPIR if we set $K_c = 1$ and that of MDS-PIR if we set $T = 1$. However, in all non-degenerate cases where $T, K_c \notin \{1, N\}$, the capacity of MDS-TPIR, and therefore the validity of the conjecture is unknown. In fact, in all these cases the problem is open on *both* sides, i.e., the conjectured capacity expression is neither known to be achievable, nor known to be an outer bound. The lack of any non-trivial outer bounds for MDS-TPIR is also recently highlighted in [9]. This intriguing combination of plausibility, uncertainty and generality of the FGHK conjecture motivates our work. Our contribution is summarized next.

A. Summary of Contribution

As the main outcome of this work, we disprove the FGHK conjecture. For our counterexample, we consider the setting $(K, N, T, K_c) = (2, 4, 2, 2)$ where the data is stored using the $(2, 4)$ MDS code $(x, y) \to (x, y, x + y, x + 2y)$. The conjectured capacity for this setting is $4/7$. We show that the rate $3/5 > 4/7$ is achievable, thus disproving the conjecture. As a converse argument, we show that no (scalar or vector) linear PIR scheme can achieve a rate higher than $3/5$ for this MDS storage code subject to $T = 2$ privacy.

The insights from the counterexample lead us to characterize the exact capacity of various instances of MDS-TPIR. This includes all cases with $(K, N, T, K_c) = (2, N, T, N-1)$, where N and T can be arbitrary. The capacity for these cases turns out to be

$$C = \frac{N^2 - N}{2N^2 - 3N + T} \tag{5}$$

Note that this is the information theoretic capacity, i.e., for $K = 2$ messages, no $(N - 1, N)$ MDS storage code and no PIR scheme (linear or non-linear) can beat this rate, which is achievable with the simple MDS storage code $(x_1, x_2, \cdots, x_{N-1}) \to (x_1, x_2, \cdots, x_{N-1}, \sum_{i=1}^{N-1} x_i)$[2] and a linear PIR scheme.

The general capacity expression for MDS-TPIR remains unknown. However, we are able to show that it cannot be symmetric in K_c and T, i.e., the two parameters are not interchangeable in general. Also, between K_c and T the capacity expression does not consistently favor one over the other. These findings are illustrated by the following four cases for which the capacity is settled.

	(K, N, T, K_c)			
	$(2, 4, 2, 3)$	$(2, 4, 3, 2)$	$(2, 4, 1, 3)$	$(2, 4, 3, 1)$
Capacity	$6/11$	$4/7$	$4/7$	$4/7$
Ref.	Theorem 3	Appendix E2	[6]	[3]

The first two columns show that the capacity is not symmetric in K_c and T, since switching their values changes the capacity. The first two columns also suggest that increasing K_c hurts capacity more than increasing T. However, considering columns 3 and 4 as the baseline where the capacities are equal, and comparing the drop in capacity from column 3 to column 1 when T is increased, versus no change in capacity from column 4 to column 2 when K_c is increased shows the opposite trend. Therefore, neither T nor K_c is consistently dominant in terms of the sensitivity of capacity to these two parameters.

Finally, taking an asymptotic view of capacity of MDS-TPIR, we show that if $T + K_c > N$, then the capacity collapses to 0 as the number of messages $K \to \infty$. This is consistent with the restriction of $T + K_c \le N$ that is required by the achievable scheme of Freij-Hollanti et al. whose rate does not depend on K.

Notation: For $n_1, n_2 \in \mathbb{Z}$, define the notation $[n_1 : n_2]$ as the set $\{n_1, n_1 + 1, \cdots, n_2\}$, $A_{n_1:n_2}$ as the vector $(A_{n_1}, A_{n_1+1}, \cdots, A_{n_2})$, and $S(n_1 : n_2, :)$ as the submatrix of a matrix S formed by retaining only the n_1^{th} to the n_2^{th} rows. The notation $X \sim Y$ is used to indicate that X and Y are identically distributed. The cardinality of a set $\mathcal{I}$ is denoted as $|\mathcal{I}|$. The determinant of a matrix S is denoted as $|S|$. For an index set $\mathcal{I} = \{i_1, \cdots, i_n\}$ such that $i_1 < \cdots < i_n$, the notation $A_{\mathcal{I}}$ represents the vector $(A_{i_1}, \cdots, A_{i_n})$. $(V_1; V_2; \cdots; V_n)$ refers to a matrix whose i^{th} row vector is $V_i, i \in [1 : n]$.

II. Problem Statement

Consider[3] K independent messages $W_1, \cdots, W_K \in \mathbb{F}_p^{L\times 1}$, each represented as an $L \times 1$ vector comprised of L i.i.d. uniform symbols from a finite field $\mathbb{F}_p$ for a prime p. In p-ary units,

$$H(W_1) = \cdots = H(W_K) = L \tag{6}$$

$$H(W_1, \cdots, W_K) = H(W_1) + \cdots + H(W_K) \tag{7}$$

There are N servers. The n^{th} server stores $(W_{1n}, W_{2n}, \cdots, W_{Kn})$, where $W_{kn} \in \mathbb{F}^{\frac{L}{K_c}\times 1}$ represents L/K_c symbols from $W_k, k \in [1 : K]$.

$$H(W_{kn}|W_k) = 0, \quad H(W_{kn}) = L/K_c \tag{8}$$

We require the storage system to satisfy the MDS property, i.e., from the information stored in any K_c servers, we can recover each message, i.e.,

$$[\text{MDS}]\ H(W_k|W_{k\mathcal{K}_c}) = 0, \quad \forall \mathcal{K}_c \subset [1 : N], |\mathcal{K}_c| = K_c \tag{9}$$

[2]The last symbol is the parity symbol and the code may be called the parity code.

[3]While the problem statement is presented in its general form, we will primarily consider cases with $K = 2$ messages in this paper (outer bounds for larger K are presented in Section VII-A).

Let us use $\mathcal{F}$ to denote a random variable privately generated by the user, whose realization is not available to the servers. $\mathcal{F}$ represents the randomness in the strategies followed by the user. Similarly, $\mathcal{G}$ is a random variable that determines the random strategies followed by the servers, and whose realizations are assumed to be known to all the servers and to the user. The user privately generates θ uniformly from $[1:K]$ and wishes to retrieve W_θ while keeping θ a secret from each server. $\mathcal{F}$ and $\mathcal{G}$ are generated independently and before the realizations of the messages or the desired message index are known, so that

$$H(\theta,\mathcal{F},\mathcal{G},W_1,\cdots,W_K)=H(\theta)+H(\mathcal{F})+H(\mathcal{G})+H(W_1)+\cdots+H(W_K) \quad (10)$$

Suppose $\theta=k$. In order to retrieve $W_k, k\in[1:K]$ privately, the user privately generates N random queries, $Q_1^{[k]},\cdots,Q_N^{[k]}$.

$$H(Q_1^{[k]},\cdots,Q_N^{[k]}|\mathcal{F})=0,\quad \forall k\in[1:K] \quad (11)$$

The user sends query $Q_n^{[k]}$ to the n^{th} server, $n\in[1:N]$. Upon receiving $Q_n^{[k]}$, the n^{th} server generates an answering string $A_n^{[k]}$, which is a function of the received query $Q_n^{[k]}$, the stored information $W_{1n},\cdots,W_{Kn}$ and $\mathcal{G}$,

$$H(A_n^{[k]}|Q_n^{[k]},W_{1n},\cdots,W_{Kn},\mathcal{G})=0 \quad (12)$$

Each server returns to the user its answer $A_n^{[k]}$.[4]

From all the information that is now available at the user $(A_{1:N}^{[k]},Q_{1:N}^{[k]},\mathcal{F},\mathcal{G})$, the user decodes the desired message W_k according to a decoding rule that is specified by the PIR scheme. Let P_e denote the probability of error achieved with the specified decoding rule.

To protect the user's privacy, the K strategies must be indistinguishable (identically distributed) from the perspective of any subset $\mathcal{T}\subset[1:N]$ of at most T colluding servers, i.e., the following privacy constraint must be satisfied.[5]

$$\begin{aligned}&[T\text{-Privacy}]\ (Q_{\mathcal{T}}^{[k]},A_{\mathcal{T}}^{[k]},\mathcal{G},W_{1\mathcal{T}},\cdots,W_{K\mathcal{T}})\\&\sim(Q_{\mathcal{T}}^{[k']},A_{\mathcal{T}}^{[k']},\mathcal{G},W_{1\mathcal{T}},\cdots,W_{K\mathcal{T}}),\\&\forall k,k'\in[1:K],\forall\mathcal{T}\subset[1:N],\ |\mathcal{T}|=T\end{aligned} \quad (13)$$

The PIR rate characterizes how many bits of desired information are retrieved per downloaded bit and is defined as follows.

$$R=L/D \quad (14)$$

where D is the expected value (over random queries) of the total number of bits downloaded by the user from all the servers.

A rate R is said to be ϵ-error achievable if there exists a sequence of PIR schemes, indexed by L, each of rate greater than or equal to R, for which $P_e\to 0$ as $L\to\infty$. Note that for such a sequence of PIR schemes, from Fano's inequality, we must have

$$\begin{aligned}[\text{Correctness}]\ o(L)&=H(W_k|A_{1:N}^{[k]},Q_{1:N}^{[k]},\mathcal{F},\mathcal{G}) &(15)\\&\overset{(11)}{=}H(W_k|A_{1:N}^{[k]},\mathcal{F},\mathcal{G}) &(16)\end{aligned}$$

where any function of L, say $f(L)$, is said to be $o(L)$ if $\lim_{L\to\infty}f(L)/L=0$. The supremum of ϵ-error achievable rates is called the capacity C.[6]

III. Settling the Conjecture

Our main result, which settles the FGHK conjecture, is stated in the following theorem.

Theorem 1: For the MDS-TPIR problem with $K=2$ messages, $N=4$ servers, $T=2$ privacy and the $(K_c,N)=(2,4)$ MDS storage code $(x,y)\to(x,y,x+y,x+2y)$, a rate of $3/5$ is achievable. Since the achievable rate exceeds the conjectured capacity of $4/7$ for this setting, the FGHK conjecture is false.

Proof: We divide the proof into 6 sections. In Section III-A, we first specify the storage code. Then we construct the queries in Section III-B. After receiving the queries from the user, the servers produce the answers (described in Section III-C) and return them to the user. Finally, we prove that the scheme is correct in Section III-D, that the scheme is private in Section III-E, and that the rate of the scheme is $3/5$, as desired, in Section III-F. Next we proceed to the details of the proof.

We present a scheme that achieves rate $3/5$. We assume that each message is comprised of $L=12$ symbols from $\mathbb{F}_p$ for a sufficiently[7] large prime p. Define $\mathbf{a}\in\mathbb{F}_p^{6\times 1}$ as the 6×1 vector $(a_1;a_2;\cdots;a_6)$ comprised of i.i.d. uniform symbols $a_i\in\mathbb{F}_p$. Vectors $\mathbf{b},\mathbf{c},\mathbf{d}$ are defined similarly. Messages W_1,W_2 are defined in terms of these vectors as follows.

$$W_1=(\mathbf{a};\mathbf{b})\quad W_2=(\mathbf{c};\mathbf{d}) \quad (17)$$

A. Storage Code

The storage is specified as

$$(W_{11},W_{12},W_{13},W_{14})=(\mathbf{a},\mathbf{b},\mathbf{a}+\mathbf{b},\mathbf{a}+2\mathbf{b}) \quad (18)$$
$$(W_{21},W_{22},W_{23},W_{24})=(\mathbf{c},\mathbf{d},\mathbf{c}+\mathbf{d},\mathbf{c}+2\mathbf{d}) \quad (19)$$

Recall that W_{kn} is the information about message W_k that is stored at Server n. Thus, Server 1 stores $(\mathbf{a},\mathbf{c})$, Server 2 stores $(\mathbf{b},\mathbf{d})$, Server 3 stores $(\mathbf{a}+\mathbf{b},\mathbf{c}+\mathbf{d})$, and Server 4 stores $(\mathbf{a}+2\mathbf{b},\mathbf{c}+2\mathbf{d})$. In particular, each server stores 6 symbols for each message, for a total of 12 symbols per server. Any two servers store just enough information to recover both messages, thus the MDS storage criterion is satisfied.

[4] If the $A_n^{[k]}$ are obtained as inner products of query vectors and stored message vectors, then such a PIR scheme is called a linear PIR scheme.

[5] The privacy constraint is equivalently expressed as $I(\theta;Q_{\mathcal{T}}^{[\theta]},A_{\mathcal{T}}^{[\theta]},\mathcal{G},W_{1\mathcal{T}},\cdots,W_{K\mathcal{T}})=0$.

[6] Alternatively, the capacity may be defined with respect to zero error criterion, i.e., the supreme of zero error achievable rates where a rate R is said to be zero error achievable if there exists (for some L) a PIR scheme of rate greater than or equal to R for which $P_e=0$.

[7] It suffices to choose $p=349$ for Theorem 1. In general, the appeal to large field size, analogous to the random coding argument in information theory, is made to prove the existence of a scheme, but may not be essential to the construction of the PIR scheme. To underscore this point, Section VII-E includes some examples of MDS-TPIR capacity achieving schemes over small fields.

B. Construction of Queries

The query to each server $Q_n^{[k]}$[8] is comprised of two parts, denoted as $Q_n^{[k]}(W_1), Q_n^{[k]}(W_2)$. Each part contains 3 row vectors, also called query vectors, along which the server should project its corresponding stored message symbols.

$$Q_n^{[k]} = (Q_n^{[k]}(W_1), Q_n^{[k]}(W_2)) \tag{20}$$

In preparation for the construction of the queries, let us denote the set of all full rank 6×6 matrices over $\mathbb{F}_p$ as $\mathcal{S}$. The user privately chooses two matrices, S and S', independently and uniformly from $\mathcal{S}$. Label the rows of S as $V_1, V_2, V_3, V_4, V_5, V_6$, and the rows of S' as $U_0, U_1, U_2, U_3, U_4, U_5$. Define[9]

$$\mathcal{V}_1 = \{V_1, V_2, V_3\}, \quad \mathcal{U}_1 = \{U_0, U_6, U_8\} \tag{21}$$
$$\mathcal{V}_2 = \{V_1, V_4, V_5\}, \quad \mathcal{U}_2 = \{U_0, U_7, U_9\} \tag{22}$$
$$\mathcal{V}_3 = \{V_2, V_4, V_6\}, \quad \mathcal{U}_3 = \{U_0, U_1, U_3\} \tag{23}$$
$$\mathcal{V}_4 = \{V_3, V_5, V_6\}, \quad \mathcal{U}_4 = \{U_0, U_2, U_4\} \tag{24}$$

U_6, U_7, U_8, U_9 are obtained as follows.

$$U_6 = U_1 + U_2, \quad U_7 = U_1 + 2U_2 \tag{25}$$
$$U_8 = U_3 + U_4, \quad U_9 = U_3 + 2U_4 \tag{26}$$

As a preview of what we are trying to accomplish, we note that for Server $n \in [1:4]$, $\mathcal{V}_n$ will be used as the query vectors for desired message symbols, while $\mathcal{U}_n$ will be used as query vectors for undesired message symbols. Since $K_c = 2$, the same query vector V_i sent to two different servers will recover 2 independent desired symbols. Each $V_i, i \in [1:6]$, is used exactly twice, so all queries for desired symbols will return independent information for a total of 12 independent desired symbols. On the other hand, for undesired symbols note that U_0 is used as the query vector to all 4 servers, but because $K_c = 2$, it can only produce 2 independent symbols, i.e., 2 of the 4 symbols are redundant. The dependencies introduced via (25),(26) are carefully chosen to ensure that the queries along U_1, U_2, U_6, U_7 will produce only 3 independent symbols. Similarly, the queries along U_3, U_4, U_8, U_9 will produce only 3 independent symbols. Thus, all the queries for the undesired message will produce a total of only 8 independent symbols.[10] The 12 independent desired symbols and 8 independent undesired symbols will be resolved from a total of $12 + 8 = 20$ downloaded symbols, to achieve the rate $12/20 = 3/5$. To ensure $T = 2$ privacy, the $\mathcal{U}_i$ and $\mathcal{V}_i$ queries will be made indistinguishable from the perspective of any 2 colluding servers. The key to the $T = 2$ privacy is that any $\mathcal{V}_n, \mathcal{V}_{n'}$, $n \neq n'$ have one element in common. Similarly, any $\mathcal{U}_n, \mathcal{U}_{n'}$, $n \neq n'$ also have one element in common. This is a critical aspect of the construction.

Next we provide a detailed description of the queries and downloads for message $W_k, k \in [1:2]$, both when W_k is desired and when it is not desired. To simplify the notation, we will denote $W_k = (\mathbf{x};\mathbf{y})$. Note that when $k = 1$, $(\mathbf{x};\mathbf{y}) = (\mathbf{a};\mathbf{b})$ and when $k = 2$, $(\mathbf{x};\mathbf{y}) = (\mathbf{c};\mathbf{d})$.

1) Cases 1 (W_k Is Desired): The query sent to Server n is a 3×6 matrix whose rows are the 3 vectors in $\mathcal{V}_n$. The ordering of the rows is uniformly random,[11] i.e.,

$$\text{Server } n: \ Q_n^{[k]}(W_k) = \pi_n(\mathcal{V}_n), \quad n \in [1:4] \tag{27}$$

For a set $\mathcal{V} = \{V_{i_1}, V_{i_2}, V_{i_3}\}$, $\pi_n(\mathcal{V})$ is equally likely to return any one of the 6 possibilities: $(V_{i_1};V_{i_2};V_{i_3})$, $(V_{i_1};V_{i_3};V_{i_2})$, $(V_{i_2};V_{i_1};V_{i_3})$, $(V_{i_2};V_{i_3};V_{i_1})$, $(V_{i_3};V_{i_1};V_{i_2})$ and $(V_{i_3};V_{i_2};V_{i_1})$. The π_n are independently chosen for each $n \in [1:4]$.

After receiving the 3 query vectors $Q_n^{[k]}(W_k)$, Server n projects its stored W_{kn} symbols along these vectors. This creates three linear combinations of W_{kn} symbols (denoted as $A_n^{[k]}(W_k)$).

$$A_n^{[k]}(W_k) = Q_n^{[k]}(W_k)W_{kn} \tag{28}$$

Define $k^c = 3-k$ as the complement of k, i.e., $k^c = 1$ if $k = 2$ and vice versa. The answers $A_n^{[k]}$ to be sent to the user will be constructed eventually by combining $A_n^{[k]}(W_k)$ and $A_n^{[k]}(W_{k^c})$, since separately sending these answers will be too inefficient. The details of this combining process will be specified later. Next we note an important property of the construction.

Desired symbols are independent: We show that if the user can recover $A_{1:4}^{[k]}(W_k)$ from the downloads, then he can recover all 12 symbols of W_k. From $A_{1:4}^{[k]}(W_k)$ the user recovers the 12 symbols $V_1\mathbf{x}$, $V_2\mathbf{x}$, $V_3\mathbf{x}$, $V_1\mathbf{y}$, $V_4\mathbf{y}$, $V_5\mathbf{y}$, $V_2(\mathbf{x}+\mathbf{y})$, $V_4(\mathbf{x}+\mathbf{y})$, $V_6(\mathbf{x}+\mathbf{y})$, $V_3(\mathbf{x}+2\mathbf{y})$, $V_5(\mathbf{x}+2\mathbf{y})$, $V_6(\mathbf{x}+2\mathbf{y})$. From these 12 symbols, he recovers $V_i\mathbf{x}$ and $V_i\mathbf{y}$ for all $i \in [1:6]$. Since $S = (V_1;V_2;V_3;V_4;V_5;V_6)$ has full rank (invertible) and the user knows $V_{1:6}$, he recovers all symbols in $\mathbf{x}$ and $\mathbf{y}$ (thus W_k).

2) Case 2 (W_k Is Undesired): Similarly, the query sent to Server n is a 3×6 matrix whose rows are the 3 vectors in $\mathcal{U}_n$. The ordering of the rows is uniformly random for each n, and independent across all $n \in [1:4]$.

$$\text{Server } n: \ Q_n^{[k^c]}(W_k) = \pi'_n(\mathcal{U}_n), \quad n \in [1:4] \tag{29}$$

Each server projects its stored W_{kn} symbols along the 3 query vectors to obtain,

$$A_n^{[k^c]}(W_k) = Q_n^{[k^c]}(W_k)W_{kn} \tag{30}$$

Interfering Symbols Have Dimension 8: $A_{1:4}^{[k^c]}(W_k)$ is comprised of $U_0\mathbf{x}$, $U_6\mathbf{x}$, $U_8\mathbf{x}$, $U_0\mathbf{y}$, $U_7\mathbf{y}$, $U_9\mathbf{y}$, $U_0(\mathbf{x}+\mathbf{y})$, $U_1(\mathbf{x}+\mathbf{y})$, $U_3(\mathbf{x}+\mathbf{y})$, $U_0(\mathbf{x}+2\mathbf{y})$, $U_2(\mathbf{x}+2\mathbf{y})$, $U_4(\mathbf{x}+2\mathbf{y})$. We now show that these 12 symbols are dependent and have dimension only 8.[12] Because of (25) and (26), we have

$$\begin{aligned} U_0\mathbf{x} + U_0\mathbf{y} &= U_0(\mathbf{x}+\mathbf{y}) \\ U_0\mathbf{x} + 2U_0\mathbf{y} &= U_0(\mathbf{x}+2\mathbf{y}) \\ U_6\mathbf{x} + U_7\mathbf{y} - U_1(\mathbf{x}+\mathbf{y}) &= U_2(\mathbf{x}+2\mathbf{y}) \\ U_8\mathbf{x} + U_9\mathbf{y} - U_3(\mathbf{x}+\mathbf{y}) &= U_4(\mathbf{x}+2\mathbf{y}) \end{aligned} \tag{31}$$

[8]Throughout the proof, we consistently use k to refer to the desired message index and k^c to refer to the other undesired message index.

[9]Similar assignments as these V_i vectors have appeared in the distributed storage repair literature [10].

[10]The intuition is to produce as few independent undesired symbols as possible, subject to the constraint that the vectors overlap pairwise in one dimension.

[11]The permutation of the rows ensures that the relative order of the rows does not carry any information.

[12]Equivalently, the joint entropy of these 12 variables, conditioned on $U_{0:9}$ is only 8 p-ary units.

Thus, of the 12 symbols recovered from $A_{1:4}^{[k^c]}(W_k)$, at least 4 are linear combinations of the remaining 8. It follows that $A_{1:4}^{[k^c]}(W_k)$ contains no more than 8 dimensions (i.e., the joint entropy of the symbols is only 8 p-ary units). The number of dimensions is also not less than 8 because, the following 8 undesired symbols (two symbols from each server) are independent,

$$\begin{aligned}
&\text{Server 1}: \quad U_0\mathbf{x}, U_6\mathbf{x} = (U_1+U_2)\mathbf{x}\\
&\text{Server 2}: \quad U_0\mathbf{y}, U_9\mathbf{y} = (U_3+2U_4)\mathbf{y}\\
&\text{Server 3}: \quad U_1(\mathbf{x}+\mathbf{y}), \quad U_3(\mathbf{x}+\mathbf{y})\\
&\text{Server 4}: \quad U_2(\mathbf{x}+2\mathbf{y}), \quad U_4(\mathbf{x}+2\mathbf{y})
\end{aligned} \tag{32}$$

To see that the 8 symbols are independent, we add 4 new symbols ($U_1\mathbf{x}$, $U_3\mathbf{y}$, $U_5\mathbf{x}$, $U_5\mathbf{y}$) such that from the 12 symbols, we can recover all 12 undesired symbols ($S'\mathbf{x}$, $S'\mathbf{y}$). Since the 4 new symbols cannot contribute more than 4 dimensions, the original 8 symbols must occupy at least 8 dimensions.

C. Combining Answers for Efficient Download

Based on the queries, each server has 3 linear combinations of symbols of W_1 in $A_n^{[k]}(W_1)$ and 3 linear combinations of symbols of W_2 in $A_n^{[k]}(W_2)$ for a total of 12 linear combinations of desired symbols and 12 linear combinations of undesired symbols across all servers. However, recall that there are only 8 independent linear combinations of undesired symbols. This is a fact that can be exploited to improve the efficiency of download. Specifically, we will combine the 6 queried symbols (i.e., the 6 linear combinations) from each server into 5 symbols to be downloaded by the user. Intuitively, 5 symbols from each server will give the user a total of 20 symbols, from which he can resolve the 12 desired and 8 undesired symbols. Next we specify the details of the combining function. Note that we have random permutations of the query vectors and the combining function must work regardless of the realization of the permutations.

The following function maps 6 queried symbols to 5 downloaded symbols.

$$\mathcal{L}(X_1, X_2, X_3, Y_1, Y_2, Y_3) = (X_1, X_2, Y_1, Y_2, X_3+Y_3) \tag{33}$$

Note that the first four symbols are directly downloaded and only the last symbol is mixed. The desired and undesired symbols are combined to produce the answers as follows.

$$A_n^{[k]} = \mathcal{L}(C_n A_n^{[k]}(W_1), C_n A_n^{[k]}(W_2)) \tag{34}$$

where C_n are deterministic 3×3 matrices, that are required to satisfy the following two properties. Denote the first 2 rows of C_n as $\overline{C}_n$.

P1. All C_n must have full rank.

P2. For all $(3!)^4$ distinct realizations of $\pi'_n, n \in [1:4]$,[13] the 8 linear combinations of the undesired message symbols that are directly downloaded (2 from each server), $\overline{C}_1 A_1^{[k]}(W_{k^c})$, $\overline{C}_2 A_2^{[k]}(W_{k^c})$, $\overline{C}_3 A_3^{[k]}(W_{k^c})$, $\overline{C}_4 A_4^{[k]}(W_{k^c})$ are independent.

[13]Note that the permutations π'_n are not revealed to the servers so that C_n must be chosen independently of π'_n.

As we will prove in the sequel, it is not difficult to find matrices that satisfy these properties. In fact, these properties are 'generic', i.e., uniformly random choices of C_n matrices will satisfy these properties with probability approaching 1 as the field size approaches infinity. The appeal to generic property will be particularly useful as we consider larger classes of MDS-TPIR settings. Those (weaker) proofs apply here as well. However, for the particular setting of Theorem 1, based on a brute force search we are able to strengthen the proof by presenting the following explicit choice of $C_n, n \in [1:4]$ which satisfies both properties over $\mathbb{F}_{349}$.

$$C_1 = \begin{pmatrix} 1 & 2 & 3\\ 6 & 5 & 4\\ 0 & 0 & 1\end{pmatrix}, \quad C_2 = \begin{pmatrix} 1 & 7 & 3\\ 11 & 9 & 8\\ 0 & 0 & 1\end{pmatrix},$$
$$C_3 = \begin{pmatrix} 1 & 10 & 8\\ 7 & 5 & 4\\ 0 & 0 & 1\end{pmatrix}, \quad C_4 = \begin{pmatrix} 1 & 3 & 5\\ 12 & 9 & 3\\ 0 & 0 & 1\end{pmatrix}. \tag{35}$$

Property $P1$ is trivially verified. Property $P2$ is verified by considering one by one, all of the 6^4 distinct realizations of $\pi'_n, n \in [1:4]$. To show how this is done, let us consider one case here. Suppose the realization of the permutations is such that

$$\pi'_1(\mathcal{U}_1) = (U_0, U_6, U_8) \tag{36}$$
$$\pi'_2(\mathcal{U}_2) = (U_0, U_9, U_7) \tag{37}$$
$$\pi'_3(\mathcal{U}_3) = (U_1, U_3, U_0) \tag{38}$$
$$\pi'_4(\mathcal{U}_4) = (U_2, U_4, U_0) \tag{39}$$

then we have

$$\begin{aligned}
&(\overline{C}_1 A_1^{[k]}(W_{k^c}); \cdots; \overline{C}_4 A_4^{[k]}(W_{k^c}))\\
&= \underbrace{\begin{pmatrix} 1 & 2 & 0 & -3 & 0 & 3 & 0 & 3\\ 6 & 5 & 0 & -4 & 0 & 4 & 0 & 4\\ 0 & -3 & 1 & 7 & 3 & 0 & 3 & 0\\ 0 & -8 & 11 & 9 & 8 & 0 & 8 & 0\\ 8 & 0 & 8 & 0 & 1 & 10 & 0 & 0\\ 4 & 0 & 4 & 0 & 7 & 5 & 0 & 0\\ 5 & 0 & 10 & 0 & 0 & 0 & 1 & 3\\ 3 & 0 & 6 & 0 & 0 & 0 & 12 & 9\end{pmatrix}}_{\triangleq \mathcal{C}} \begin{pmatrix} U_0\mathbf{x}\\ U_6\mathbf{x}\\ U_0\mathbf{y}\\ U_9\mathbf{y}\\ U_1(\mathbf{x}+\mathbf{y})\\ U_3(\mathbf{x}+\mathbf{y})\\ U_2(\mathbf{x}+2\mathbf{y})\\ U_4(\mathbf{x}+2\mathbf{y})\end{pmatrix}
\end{aligned} \tag{40}$$

The determinant of $\mathcal{C}$ over $\mathbb{F}_{349}$ is 321. Since the determinant is non-zero, all of its 8 rows are linearly independent. Note that the test for property $P2$ does not depend on the realizations of U_i vectors. To see why this is true, note that the 8 linear combinations of $(\mathbf{x}, \mathbf{y})$ in the rightmost column vector of (40) are linearly independent. Therefore, if $\mathcal{C}$ is an invertible matrix then the 8 directly downloaded linear combinations on the LHS of (40) are also independent (have joint entropy 8 p-ary units, conditioned on $U_{0:9}$).

At this point the construction of the scheme is complete. All that remains now is to prove that the scheme is correct, i.e., it retrieves the desired message, and that it is $T=2$ private.

D. The Scheme Is Correct (Retrieves Desired Message)

As noted previously, the first 4 variables in the output of the $\mathcal{L}$ function are obtained directly, i.e., $\overline{C}_1 A_1^{[k]}(W_1)$, $\overline{C}_2 A_2^{[k]}(W_1)$, $\overline{C}_3 A_3^{[k]}(W_1)$, $\overline{C}_4 A_4^{[k]}(W_1)$ and $\overline{C}_1 A_1^{[k]}(W_2)$, $\overline{C}_2 A_2^{[k]}(W_2)$, $\overline{C}_3 A_3^{[k]}(W_2)$, $\overline{C}_4 A_4^{[k]}(W_2)$ are all directly recovered. By property *P2* of C_n, $\overline{C}_1 A_1^{[k]}(W_{k^c})$, $\overline{C}_2 A_2^{[k]}(W_{k^c})$, $\overline{C}_3 A_3^{[k]}(W_{k^c})$, $\overline{C}_4 A_4^{[k]}(W_{k^c})$ are linearly independent. Since the user has recovered 8 independent dimensions of interference, and interference only spans 8 dimensions, all interference is recovered and eliminated. Once the interference is eliminated, since C_n matrices have full rank, the user is left with 12 independent linear combinations of desired symbols, from which he is able to recover the 12 desired message symbols. Therefore the scheme is correct.

E. The Scheme Is Private (to any $T = 2$ Colluding Servers)

To prove that the scheme is $T = 2$ private (refer to (13)), it suffices to show that the queries for any 2 servers are identically distributed, regardless of which message is desired. Since each query is made up of two independently generated parts, one for each message, it suffices to prove that the query vectors for a message (say W_k) are identically distributed, regardless of whether the message is desired or undesired,

$$\left(Q_{n_1}^{[k]}(W_k), Q_{n_2}^{[k]}(W_k)\right) \sim \left(Q_{n_1}^{[k^c]}(W_k), Q_{n_2}^{[k^c]}(W_k)\right), \quad \forall n_1, n_2 \in [1:4], n_1 < n_2 \tag{41}$$

Note that

$$\left(Q_{n_1}^{[k]}(W_k), Q_{n_2}^{[k]}(W_k)\right) = \left(\pi_{n_1}(\mathcal{V}_{n_1}), \pi_{n_2}(\mathcal{V}_{n_2})\right) \tag{42}$$

$$\left(Q_{n_1}^{[k^c]}(W_k), Q_{n_2}^{[k^c]}(W_k)\right) = \left(\pi'_{n_1}(\mathcal{U}_{n_1}), \pi'_{n_2}(\mathcal{U}_{n_2})\right) \tag{43}$$

Therefore, to prove (41) it suffices to show the following.

$$\left(V_{i_1}, V_{i_2}, V_{i_3}, V_{i_4}, V_{i_5}\right) \sim \left(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4}\right) \tag{44}$$

where $\mathcal{V}_{n_1} = \{V_{i_1}, V_{i_2}, V_{i_3}\}$, $\mathcal{V}_{n_2} = \{V_{i_1}, V_{i_4}, V_{i_5}\}$, $\mathcal{U}_{n_1} = \{U_0, U_{j_1}, U_{j_2}\}$, $\mathcal{U}_{n_2} = \{U_0, U_{j_3}, U_{j_4}\}$. Because S is uniformly chosen from the set of all full rank matrices, we have

$$(V_{i_1}, V_{i_2}, V_{i_3}, V_{i_4}, V_{i_5}) \sim (V_1, V_2, V_3, V_4, V_5) \tag{45}$$

Next we note that there is a bijection between

$$(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4}) \leftrightarrow (U_0, U_1, U_2, U_3, U_4) \tag{46}$$

This is because $(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4})$ always includes U_0, two terms out of U_1, U_2, U_6, U_7 and two terms out of U_3, U_4, U_8, U_9. But from any two terms of U_1, U_2, U_6, U_7 there is a bijection to U_1, U_2, and from any two terms of U_3, U_4, U_8, U_9 there is a bijection to U_3, U_4. Now since $S' = (U_0; U_1; U_2; U_3; U_4; U_5)$ is picked uniformly from $\mathcal{S}$, conditioned on any feasible value of U_5, $(U_0, U_1, U_2, U_3, U_4)$ is uniformly distributed over all possible values that preserve full rank for S'. Since $(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4})$ spans the same space as $(U_0, U_1, U_2, U_3, U_4)$, they have the same set of feasible values. The bijection between them then means that $(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4})$ is also uniformly distributed over all possibilities that preserve full rank for S', conditioned on any feasible U_5. That means[14]

$$(U_0, U_{j_1}, U_{j_2}, U_{j_3}, U_{j_4}) \sim (U_0, U_1, U_2, U_3, U_4) \tag{47}$$

Finally, we note that S and S' are identically distributed, so we have

$$(V_1, V_2, V_3, V_4, V_5) \sim (U_0, U_1, U_2, U_3, U_4) \tag{48}$$

Combining (45), (47) and (48), we arrive at (44) and (41).

F. Rate Achieved Is 3/5

The rate achieved is $12/20 = 3/5$, because we download 20 symbols in total (5 from each server) and the desired message size is 12 symbols.

IV. Optimality of Rate $3/5$

We presented a scheme that achieves the rate $3/5$ for the setting $(K, N, T, K_c) = (2, 4, 2, 2)$ with the MDS storage code $(x, y) \rightarrow (x, y, x + y, x + 2y)$. But is the scheme optimal? i.e., is the rate $3/5$ the highest rate possible for this setting? To settle this question we need an upper bound. So far the best information theoretic upper bound that we are able to prove is $8/13$[15] (see Appendix A1), which leaves the information theoretic capacity open for this setting. However, let us define the notion of "linear capacity" as the highest rate that can be achieved by any (scalar or vector) linear PIR scheme. It turns out that we are able to settle the linear capacity.

Theorem 2: For the MDS-TPIR problem with $(K, N, T, K_c) = (2, 4, 2, 2)$ and the MDS storage code $(x, y) \rightarrow (x, y, x + y, x + 2y)$, the linear capacity is $3/5$.

Proof: Since the achievability of $3/5$ has already been shown, we are left to prove the converse, i.e., the upper bound.

Let $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d} \in \mathbb{F}_p^{L/2 \times 1}$ be i.i.d. uniform $L/2 \times 1$ vectors over $\mathbb{F}_p$. Without loss of generality, the MDS storage code for message W_k is represented as follows.

$$W_1 = (\mathbf{a}; \mathbf{b}) \quad W_2 = (\mathbf{c}; \mathbf{d}) \tag{49}$$

and the storage is specified as

$$\begin{aligned}(W_{11}, W_{12}, W_{13}, W_{14}) &= (\mathbf{a}, \mathbf{b}, \mathbf{a} + \mathbf{b}, \mathbf{a} + 2\mathbf{b}) \\ (W_{21}, W_{22}, W_{23}, W_{24}) &= (\mathbf{c}, \mathbf{d}, \mathbf{c} + \mathbf{d}, \mathbf{c} + 2\mathbf{d})\end{aligned} \tag{50}$$

The scheme is linear so that the download from each server consists of linear combinations of the stored symbols of both messages. Furthermore, without loss of generality, we assume that the scheme is symmetric[16] and the download from each

[14]Note that the bijection by itself does not suffice to show the two distributions are the same. We further need uniformity.

[15]Remarkably, $8/13$ can be shown to be the capacity if the colluding sets of servers are restricted to servers $\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 1\}$ (see Section VI-D1).

[16]Any scheme can be made symmetric, e.g., by repeating the original scheme for each of the $N!$ permutations of the servers to retrieve a correspondingly expanded message of length $L' = N!L$.

server is comprised of $d \leq L/2$ independent symbols from each message. Therefore, the downloads can be expressed as

$$A_n^{[k]} = V_{1n}^{[k]} W_{1n} + V_{2n}^{[k]} W_{2n}, \quad \forall n \in [1:4], k \in [1:2] \tag{51}$$

$$\text{rank}(V_{1n}^{[k]}) = \text{rank}(V_{2n}^{[k]}) = d \tag{52}$$

where $V_{in}^{[k]}$ are $D/4 \times L/2$ matrices that may be chosen randomly by the user (functions of $\mathcal{F}$). Clearly we must have $4d \geq L$ otherwise the L symbols of the desired message cannot be recovered. Define $\epsilon \geq 0$ such that

$$4d = L(1+\epsilon) \tag{53}$$

Without loss of generality, let us assume henceforth that W_2 is the desired message. For the next set of arguments, we focus only on the downloads corresponding to W_2, i.e., set all W_1 symbols to 0. Further, let us use the notation V to represent the row span of the matrix V. The symbols downloaded from Server n along $\mathsf{V} \subset \mathsf{V}_{2n}^{[2]}$, are called redundant if they can be expressed as linear combinations of symbols downloaded from other servers, i.e., they contribute no new information.

$$H(VW_{2n}|V_{2n_1}^{[2]}W_{2n_1}, V_{2n_2}^{[2]}W_{2n_2}, V_{2n_3}^{[2]}W_{2n_3}, \mathcal{F}, V) = 0 \tag{54}$$

where n, n_1, n_2, n_3 are distinct indices in $[1:4]$. Note that we download no more than a total of $L(1+\epsilon)$ (possibly dependent) symbols of W_2 from all 4 servers, from which we must be able to decode all L independent symbols of W_2. Therefore, we cannot have more than ϵL redundant symbols. Therefore, for any V that satisfies (54) we must have

$$\dim(\mathsf{V}) \leq \epsilon L \tag{55}$$

Next, let us consider the pairwise overlap between $\mathsf{V}_{2i}^{[2]}$ and $\mathsf{V}_{2j}^{[2]}$, $i < j, i, j \in [1:4]$. By the symmetry of the scheme, there exist $V_{ij}, \forall i, j \in [1:4], i \neq j$, and $\alpha \geq 0$ such that

$$\mathsf{V}_{ij} = \mathsf{V}_{2i}^{[2]} \cap \mathsf{V}_{2j}^{[2]}, \quad \dim(\mathsf{V}_{ij}) = \alpha d \tag{56}$$

The following lemma formalizes the intuition that the overlaps α must be small enough to ensure that we have enough independent symbols to recover W_2.

Lemma 1:

$$3\alpha d \leq d + 2\epsilon L \tag{57}$$

$$\text{Equivalently, } \alpha \leq \frac{1}{3} + \frac{8}{3}\left(\frac{\epsilon}{1+\epsilon}\right) \tag{58}$$

Proof: First, we show that

$$\dim(\mathsf{V}_{12} \cap \mathsf{V}_{13}) \leq \epsilon L \tag{59}$$

For any vector $v \in \mathsf{V}_{12} \cap \mathsf{V}_{13}$ (note that v belongs simultaneously to $\mathsf{V}_{21}^{[2]}, \mathsf{V}_{22}^{[2]}, \mathsf{V}_{23}^{[2]}$), the symbol vW_{23} (downloaded from Server 3) is redundant because it is a linear combination of downloads from servers 1 and 2,

$$v(\mathbf{c}+\mathbf{d}) = v\mathbf{c} + v\mathbf{d} \tag{60}$$

$$\therefore vW_{23} = vW_{21} + vW_{22} \tag{61}$$

$$\Rightarrow H(vW_{23}|V_{21}^{[2]}W_{21}, V_{22}^{[2]}W_{22}, \mathcal{F}, v) = 0 \tag{62}$$

From (62) and (55), we have (59).

Second, we show that

$$\dim\left((\mathsf{V}_{12} \cup \mathsf{V}_{13}) \cap \mathsf{V}_{14}\right) \leq \epsilon L \tag{63}$$

Consider any vector $v \in \mathsf{V}_{12}$. Because v belongs to both $\mathsf{V}_{21}^{[2]}$ and $\mathsf{V}_{22}^{[2]}$, we have downloaded $vW_{21} = v\mathbf{c}$ and $vW_{22} = v\mathbf{d}$ from servers 1 and 2. Similarly, for any vector $v' \in \mathsf{V}_{13}$, we have downloaded $v'W_{21} = v'\mathbf{c}$ and $v'W_{23} = v'(\mathbf{c}+\mathbf{d}) = v'W_{21} + v'W_{22}$ (from servers 1 and 3), from which we can recover $v'W_{21} = v'\mathbf{c}$ and $v'W_{22} = v'\mathbf{d}$. Consider now any vector $v^* \in (\mathsf{V}_{12} \cup \mathsf{V}_{13}) \cap \mathsf{V}_{14}$. Suppose $v^* = h_1\, v + h_2\, v', v \in \mathsf{V}_{12}, v' \in \mathsf{V}_{13}$ for constants h_1, h_2. The symbol $v^*W_{24} = v^*(\mathbf{c}+2\mathbf{d})$ (downloaded from Server 4) is redundant because it is a linear combination of downloads from servers 1, 2 and 3,

$$\begin{aligned} v^*W_{24} &= (h_1 v + h_2 v')(\mathbf{c}+2\mathbf{d}) && (64)\\ &= h_1 v\mathbf{c} + 2h_1 v\mathbf{d} + h_2 v'\mathbf{c} + 2h_2 v'\mathbf{d} && (65)\\ &= h_1 vW_{21} + 2h_1 vW_{22} + h_2 v'W_{21} + 2h_2 v'W_{22} && (66)\\ &\Rightarrow H(v^*W_{24}|V_{21}^{[2]}W_{21}, V_{22}^{[2]}W_{22}, V_{23}^{[2]}W_{23}, \mathcal{F}, v^*) = 0 && (67)\end{aligned}$$

From (67) and (55), we have (63). Next, consider $\dim(\mathsf{V}_{12} \cup \mathsf{V}_{13})$.

$$\begin{aligned} &\dim(\mathsf{V}_{12} \cup \mathsf{V}_{13}) && (68)\\ &\quad = \dim(\mathsf{V}_{12}) + \dim(\mathsf{V}_{13}) - \dim(\mathsf{V}_{12} \cap \mathsf{V}_{13}) && (69)\\ &\quad \geq 2\alpha d - \epsilon L \quad \text{(from (56)(59))} && (70)\end{aligned}$$

Finally, consider $\dim(\mathsf{V}_{12} \cup \mathsf{V}_{13} \cup \mathsf{V}_{14})$.

$$\begin{aligned} d &= \dim(\mathsf{V}_{21}^{[2]}) \geq \dim(\mathsf{V}_{12} \cup \mathsf{V}_{13} \cup \mathsf{V}_{14}) && (71)\\ &= \dim(\mathsf{V}_{12} \cup \mathsf{V}_{13}) + \dim(\mathsf{V}_{14}) \\ &\quad - \dim\left((\mathsf{V}_{12} \cup \mathsf{V}_{13}) \cap \mathsf{V}_{14}\right) && (72)\\ &\geq 2\alpha d - \epsilon L + \alpha d - \epsilon L \quad \text{(from (70)(56)(63))} && (73)\\ &\Rightarrow 3\alpha d \leq d + 2\epsilon L && (74)\end{aligned}$$

∎

We now proceed to complete the converse.

$$\begin{aligned} &D + o(L) \\ &\quad \geq H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) + o(L) && (75)\\ &\quad \overset{(16)}{=} H(A_{1:4}^{[1]}, W_1|\mathcal{F}, \mathcal{G}) && (76)\\ &\quad \overset{(10)}{=} H(W_1) + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{2:4}^{[1]}|W_1, A_1^{[1]}\mathcal{F}, \mathcal{G}) && (77)\\ &\quad \geq H(W_1) + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{3:4}^{[1]}|W_1, W_{21}, A_1^{[1]}, \mathcal{F}, \mathcal{G}) && (78)\\ &\quad \overset{(8)(11)(12)}{=} H(W_1) + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{3:4}^{[1]}|W_1, W_{21}, \mathcal{F}, \mathcal{G}) && (79)\\ &\quad \overset{(8)(10)(13)}{=} H(W_1) + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{3:4}^{[2]}|W_1, W_{21}, \mathcal{F}, \mathcal{G}) && (80)\\ &\quad \overset{(49)(50)}{=} H(\mathbf{a}, \mathbf{b}) + H(V_{21}^{[2]}\mathbf{c}|\mathcal{F}) \\ &\qquad + H(V_{23}^{[2]}(\mathbf{c}+\mathbf{d}), V_{24}^{[2]}(\mathbf{c}+2\mathbf{d})|\mathbf{c}, \mathcal{F}) && (81)\end{aligned}$$

$$
\begin{aligned}
&= H(\mathbf{a},\mathbf{b}) + H(V_{21}^{[2]}\mathbf{c}|\mathcal{F}) \\
&\quad + H(V_{23}^{[2]}\mathbf{d}, 2V_{24}^{[2]}\mathbf{d}|\mathcal{F}) && (82)\\
&\overset{(6)}{=} L + \dim(\mathsf{V}_{21}^{[2]}) + \dim(\mathsf{V}_{23}^{[2]} \cup \mathsf{V}_{24}^{[2]}) && (83)\\
&\overset{(52)(56)}{=} L + d + 2d - \alpha d && (84)\\
&\overset{(58)}{\geq} L + \left(3 - \frac{1}{3} - \frac{8}{3}\left(\frac{\epsilon}{1+\epsilon}\right)\right)\frac{(1+\epsilon)L}{4} && (85)\\
&= 5L/3 && (86)
\end{aligned}
$$

Letting $L \to \infty$, we have $R = L/D \leq 3/5$.

A more detailed derivation of (80) appears in (127) of Lemma 2.

Remark: The assumption of linear schemes is useful for bounding the term $H(A_{3:4}^{[2]}|W_1, W_{21}, \mathcal{F}, \mathcal{G})$ in (80), for which a strictly tighter lower bound (compared to the information theoretic converse in Appendix A1) is obtained. In particular, we are able to separate the contribution of $\mathbf{c}$ and $\mathbf{d}$ in $A_{3:4}^{[2]}$ as in (81), (82), which may not be possible in general if the answers depend on stored information in a non-linear fashion. ■

V. Capacity of a Class of MDS-TPIR Instances

Building upon the insights from the achievable scheme and linear converse presented in the previous sections, we are able to settle the information theoretic capacity of a non-trivial class of MDS-TPIR instances.

Theorem 3: For the class of MDS-TPIR instances with $(K, N, T, K_c) = (2, N, T, N-1)$, with arbitrary T, N, the capacity is $C = \frac{N^2-N}{2N^2-3N+T}$.

Remark: When $N \to \infty$, there is no redundancy in storage (code rate $\frac{N}{N-1} \to 1$). For code rate 1, it is shown by Banawan and Ulukus [6] that the capacity of PIR is $1/K$, i.e., the optimal scheme is the trivial scheme which downloads all K messages. This can also be seen from the capacity expression in Theorem 3 where $C \to 1/2$ as $N \to \infty$.

The case $T = N$ is trivial because if all servers collude then the situation is equivalent to the single database scenario, i.e., it is optimal to download everything, and the capacity is $1/K = 1/2$. For the remaining cases, $T < N$, and the proof of converse is presented in Appendix A2. The proof of achievability for $T = 2$ setting appears in Appendix B where we present a scheme with zero error.[17] The proof of achievability for $T > 2$ settings appears in Appendix C where we present a scheme with vanishing probability of error. The remainder of this section presents two examples (one with $T = 2$ and one with $T = 3$) to illustrate the key ideas.

A. Example: Capacity Achieving Scheme for $(K, N, T, K_c) = (2, 4, 2, 3)$

Let us present a scheme that achieves the rate 6/11, which is the capacity for this setting according to Theorem 3. As evident from the description below, the scheme builds upon the ideas that were introduced for Theorem 1.

[17]Note that zero error schemes automatically satisfy ϵ-error criterion.

1) Message and Storage Code: Let each message be comprised of $L = N(N-1) = 12$ independent symbols from a sufficiently large finite field $\mathbb{F}_p$. Define $\mathbf{a} \in \mathbb{F}_p^{4\times 1}$ as the vector $(a_1; a_2; a_3; a_4)$ comprised of i.i.d. uniform symbols $a_i \in \mathbb{F}_p$. Vectors $\mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}$ are defined similarly. Messages W_1, W_2 are defined in terms of these vectors as follows.

$$W_1 = (\mathbf{a}; \mathbf{b}; \mathbf{c}) \quad W_2 = (\mathbf{d}; \mathbf{e}; \mathbf{f}) \tag{87}$$

The $(N-1, N) = (3, 4)$ MDS storage code is specified as follows.

$$(W_{11}, W_{12}, W_{13}, W_{14}) = (\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{a}+\mathbf{b}+\mathbf{c}) \tag{88}$$

$$(W_{21}, W_{22}, W_{23}, W_{24}) = (\mathbf{d}, \mathbf{e}, \mathbf{f}, \mathbf{d}+\mathbf{e}+\mathbf{f}) \tag{89}$$

Note that each server stores 4 symbols for each message and any three serves store just enough information to recover both messages (MDS property is satisfied).

2) Construction of Queries: The query to each server consists of 6 vectors, the first three for W_1 (denoted as $Q_n^{[k]}(W_1)$) and the last three for W_2 (denoted as $Q_n^{[k]}(W_2)$). The queries and downloads for $W_k, k \in [1:2]$ are described next. We denote $W_k = (\mathbf{x}; \mathbf{y}; \mathbf{z})$. When $k = 1$, $(\mathbf{x}; \mathbf{y}; \mathbf{z}) = (\mathbf{a}; \mathbf{b}; \mathbf{c})$ and when $k = 2$, $(\mathbf{x}; \mathbf{y}; \mathbf{z}) = (\mathbf{d}; \mathbf{e}; \mathbf{f})$.

Denote the set of all full rank 4×4 matrices over $\mathbb{F}_p$ as $\mathcal{S}_4$. The user privately chooses two matrices S, S', independently and uniformly from $\mathcal{S}_4$. Label the rows of S as V_1, V_2, V_3, V_4, and the rows of S' as $\overline{U}_1, \overline{U}_2, U_1, U_2$. Define the following sets

$$
\begin{array}{lll}
\mathcal{V}_1 = \{\ V_2,\ V_3,\ V_4\}, & \mathcal{U}_1 = \{\overline{U}_1, \overline{U}_2, U_1\} & \\
\mathcal{V}_2 = \{V_1,\ \ V_3,\ V_4\}, & \mathcal{U}_2 = \{\overline{U}_1, \overline{U}_2, U_2\} & \\
\mathcal{V}_3 = \{V_1,\ V_2,\ \ V_4\}, & \mathcal{U}_3 = \{\overline{U}_1, \overline{U}_2, U_3\} & (90)\\
\mathcal{V}_4 = \{V_1,\ V_2,\ V_3\ \}, & \mathcal{U}_4 = \{\overline{U}_1, \overline{U}_2, U_4\} &
\end{array}
$$

where U_3, U_4 are obtained as follows.

$$U_3 = U_1 + U_2, \tag{91}$$

$$U_4 = U_1 + 2U_2 \tag{92}$$

A preview of the scheme is as follows. For Server $n \in [1:4]$, the vectors in $\mathcal{V}_n$ are for the desired message and the vectors in $\mathcal{U}_n$ are for the undesired message. Since $K_c = N - 1 = 3$, and each query vector V_i is used no more than three times, all queries for the desired message will return independent symbols for a total of 12 desired symbols. For the undesired message, the same query vector $\overline{U}_1$ is used 4 times such that only 3 independent symbols are produced. Similarly the 4 uses of $\overline{U}_2$ produce only 3 independent symbols. Thus all queries for the undesired message will produce at most $6 + 4 = 10$ independent undesired symbols. The 12 independent desired symbols and 10 undesired symbols will be resolved from a total of $12 + 10 = 22$ downloaded symbols, to achieve the rate $12/22 = 6/11$. Privacy is ensured by the observation that any $\mathcal{V}_n, \mathcal{V}_n', n \neq n'$ have two elements in common and similarly any $\mathcal{U}_n, \mathcal{U}_n', n \neq n'$ have two elements in common. We now proceed to the details.

When W_k is desired, we have $\forall n \in [1:4]$,

$$
\begin{aligned}
\text{Server } n: \ Q_n^{[k]}(W_k) &= \pi_n(\mathcal{V}_n), \\
A_n^{[k]}(W_k) &= Q_n^{[k]}(W_k)W_{kn}. && (93)
\end{aligned}
$$

a) Desired symbols are independent: From $A_{1:4}^{[k]}(W_k)$, the user can recover the 12 symbols $V_2\mathbf{x}, V_3\mathbf{x}, V_4\mathbf{x}, V_1\mathbf{y}, V_3\mathbf{y}, V_4\mathbf{y}, V_1\mathbf{z}, V_2\mathbf{z}, V_4\mathbf{z}, V_1(\mathbf{x}+\mathbf{y}+\mathbf{z}), V_2(\mathbf{x}+\mathbf{y}+\mathbf{z}), V_3(\mathbf{x}+\mathbf{y}+\mathbf{z})$ and therefore all 12 symbols $(\mathbf{x};\mathbf{y};\mathbf{z})$ of W_k, since $S=(V_1;V_2;V_3;V_4)$ has full rank.

When W_k is undesired, we have $\forall n \in [1:4]$,

$$\begin{aligned} \text{Server } n: \quad & Q_n^{[k^c]}(W_k) = \pi_n'(\mathcal{U}_n), \\ & A_n^{[k^c]}(W_k) = Q_n^{[k^c]}(W_k)W_{kn}. \end{aligned} \tag{94}$$

b) Interfering symbols are dependent and have dimension at most 10: Consider the interfering symbols along the common vectors $\overline{U}_1, \overline{U}_2$. Note that

$$\overline{U}_1\mathbf{x}+\overline{U}_1\mathbf{y}+\overline{U}_1\mathbf{z} = \overline{U}_1(\mathbf{x}+\mathbf{y}+\mathbf{z}) \tag{95}$$
$$\overline{U}_2\mathbf{x}+\overline{U}_2\mathbf{y}+\overline{U}_2\mathbf{z} = \overline{U}_2(\mathbf{x}+\mathbf{y}+\mathbf{z}) \tag{96}$$

Since at least 2 interfering symbols are linear combinations of the rest, the 12 interfering symbols cannot have more than 10 dimensions, i.e., their joint entropy is no more than 10 in p-ary units.

3) Combining Answers, Correctness and Rate: The combining process and correctness proof are similar to that in Theorem 1. The difference is that in Theorem 1, we find the explicit choice of combining matrices, here we will only prove the existence of combining matrices over a sufficiently large field. The details are deferred to the general proof in Appendix B. We repeat the above query construction two times independently such that each server has $6 \times 2 = 12$ symbols (6 in W_1 and 6 in W_2). These 12 symbols at each server are combined to 11 downloaded symbols, $A_n^{[k]}$ and it is ensured that we can decode all interfering symbols and then extract the desired symbols.

Thus, the rate achieved is 6/11.

4) Privacy Proof: The privacy proof is virtually identical to that in Theorem 1, so the details are deferred to the general proof in Appendix B.

B. Example: Capacity Achieving Scheme for $(K,N,T,K_c)=(2,4,3,3)$

Let us present a scheme that achieves the rate 12/23, which is the capacity for this setting according to Theorem 3. The key distinction of this $T=3$ case with the $T=2$ case presented in the previous section is that permutations of the query vectors are no longer enough to ensure the privacy. So we will resort to sending the space spanned by the query vectors instead of the query vectors themselves. Furthermore, instead of guaranteeing zero-error, we will only show that the probability of error can be made arbitrarily small by choosing a sufficiently large message size.

1) Message and Storage Code: The message construction and storage code are the same as when $T=2$. Let each message be comprised of $L=N(N-1)=12$ independent symbols from a sufficiently large finite field $\mathbb{F}_p$. Define $\mathbf{a} \in \mathbb{F}_p^{4\times 1}$ as the vector $(a_1;a_2;a_3;a_4)$ comprised of i.i.d. uniform symbols $a_i \in \mathbb{F}_p$. Vectors $\mathbf{b},\mathbf{c},\mathbf{d},\mathbf{e},\mathbf{f}$ are defined similarly. Messages W_1, W_2 are defined in terms of these vectors as follows.

$$W_1 = (\mathbf{a};\mathbf{b};\mathbf{c}) \quad W_2 = (\mathbf{d};\mathbf{e};\mathbf{f}) \tag{97}$$

The $(N-1,N)=(3,4)$ MDS storage code is specified as follows.

$$(W_{11},W_{12},W_{13},W_{14}) = (\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{a}+\mathbf{b}+\mathbf{c}) \tag{98}$$
$$(W_{21},W_{22},W_{23},W_{24}) = (\mathbf{d},\mathbf{e},\mathbf{f},\mathbf{d}+\mathbf{e}+\mathbf{f}) \tag{99}$$

2) Construction of Queries: The query to each server consists of two vector spaces, one for W_1 (span of the rows of $Q_n^{[k]}(W_1)$) and one for W_2 (span of the rows of $Q_n^{[k]}(W_2)$). The queries and downloads for $W_k, k\in[1:2]$ are described next. We denote $W_k=(\mathbf{x};\mathbf{y};\mathbf{z})$. When $k=1$, $(\mathbf{x};\mathbf{y};\mathbf{z})=(\mathbf{a};\mathbf{b};\mathbf{c})$ and when $k=2$, $(\mathbf{x};\mathbf{y};\mathbf{z})=(\mathbf{d};\mathbf{e};\mathbf{f})$.

Denote the set of all full rank 4×4 matrices over $\mathbb{F}_p$ as $\mathcal{S}_4$. The user privately chooses two matrices S, S', independently and uniformly from $\mathcal{S}_4$. Label the rows of S as V_1,V_2,V_3,V_4, and the rows of S' as $\overline{U}_1, U_1, U_2, U_3$. Define the following sets

$$\begin{aligned} \mathcal{V}_1 &= \{\quad V_2,\quad V_3,\quad V_4\} \\ \mathcal{V}_2 &= \{V_1,\quad\quad V_3,\quad V_4\} \\ \mathcal{V}_3 &= \{V_1,\quad V_2,\quad\quad V_4\} \\ \mathcal{V}_4 &= \{V_1,\quad V_2,\quad V_3\quad\} \end{aligned} \tag{100}$$

$$\begin{aligned} \mathcal{U}_1 &= \{\overline{U}_1,\widetilde{U}_1,\widetilde{U}_2\} = \{\overline{U}_1,U_1,U_2\} \\ \mathcal{U}_2 &= \{\overline{U}_1,\widetilde{U}_3,\widetilde{U}_4\} = \{\overline{U}_1,U_3,U_1+U_2\} \\ \mathcal{U}_3 &= \{\overline{U}_1,\widetilde{U}_5,\widetilde{U}_6\} = \{\overline{U}_1,U_1+U_3,U_2+U_3\} \\ \mathcal{U}_4 &= \{\overline{U}_1,\widetilde{U}_7,\widetilde{U}_8\} \\ &= \{\overline{U}_1,U_1+U_2+U_3,U_1+2U_2+2U_3\} \end{aligned} \tag{101}$$

where $\widetilde{U}_1,\cdots,\widetilde{U}_8$ are the rows of $\widetilde{U}$, obtained as follows.

$$\widetilde{U} = P(U_1;U_2;U_3) \tag{102}$$

$$\text{i.e.,} \begin{pmatrix} \widetilde{U}_1 \\ \widetilde{U}_2 \\ \widetilde{U}_3 \\ \widetilde{U}_4 \\ \widetilde{U}_5 \\ \widetilde{U}_6 \\ \widetilde{U}_7 \\ \widetilde{U}_8 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 2 & 2 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} \tag{103}$$

A preview of the scheme is as follows. For Server $n\in[1:4]$, the span of $\mathcal{V}_n$ is the query space for the desired message and the span of $\mathcal{U}_n$ is the query space for the undesired message. Since $K_c=N-1=3$, and each query vector V_i is used no more than three times, all queries for the desired message will return independent symbols for a total of 12 desired symbols. For the undesired message, the same query vector $\overline{U}_1$ is used 4 times such that only 3 independent symbols will be produced. Thus all queries for the undesired message will produce at most $3+8=11$ independent undesired symbols. The 12 independent desired symbols and 11 undesired symbols will be resolved from a total of $12+11=23$ downloaded symbols, to achieve the rate 12/23. Privacy is ensured by choosing P in such a way that it allows a bijective mapping between the $\mathcal{U}_n$ or $\mathcal{V}_n$ spaces that may be observed by any set of up to $T=3$ colluding servers. The bijection shows that the queries for both desired and undesired messages are uniformly distributed, and therefore indistinguishable. While a specific P

is chosen for this example, there are many choices of P that will work. In fact, P only needs to be sufficiently generic, so as the field size grows, almost all choices of P will work. We now proceed to the details.

When W_k is desired, we have $\forall n \in [1:4]$,

$$\text{Server } n: \; Q_n^{[k]}(W_k) = \mathbb{B}(\mathcal{V}_n), \quad A_n^{[k]}(W_k) = Q_n^{[k]}(W_k)W_{kn}. \tag{104}$$

where $\mathbb{B}(\mathcal{V})$ represents the reduced row echelon form of a matrix whose rows are the elements of $\mathcal{V}$. The reduced row echelon form ensures that the queries reveal only the space spanned by the corresponding V_i vectors to each server, and not directly the V_i vectors themselves. Note that from the space spanned by a set of vectors, we could not determine the set of vectors because for a given space, there are multiple choices of the vectors that constitute the same space.

a) Desired symbols are independent: From $A_{1:4}^{[k]}(W_k)$, we can recover the 12 symbols of W_k. Note that because the user knows $V_{1:4}$, from $A_{1:4}^{[k]}(W_k)$ he can recover the projections along V_i. For example, the row reduced echelon form for $\mathcal{V}_1$ is a change of basis operation that can be represented as $\mathbb{B}(\mathcal{V}_1) = B_1(V_2; V_3; V_4)$ for some invertible matrix B_1. Since the user knows B_1, he can multiply $A_1^{[k]}(W_k)$ with $(B_1)^{-1}$ as follows

$$\begin{aligned} B_1^{-1}A_1^{[k]}(W_k) &= B_1^{-1}Q_n^{[k]}(W_k)W_{k1} && (105)\\ &= B_1^{-1}B_1(V_2; V_3; V_4)\mathbf{x} && (106)\\ &= (V_2\mathbf{x}; V_3\mathbf{x}; V_4\mathbf{x}) && (107)\end{aligned}$$

Thus, from $A_{1:4}^{[k]}(W_k)$ the user recovers the 12 symbols $V_2\mathbf{x}, V_3\mathbf{x}, V_4\mathbf{x}, V_1\mathbf{y}, V_3\mathbf{y}, V_4\mathbf{y}, V_1\mathbf{z}, V_2\mathbf{z}, V_4\mathbf{z}, V_1(\mathbf{x}+\mathbf{y}+\mathbf{z})$, $V_2(\mathbf{x}+\mathbf{y}+\mathbf{z}), V_3(\mathbf{x}+\mathbf{y}+\mathbf{z})$ and therefore all 12 symbols $(\mathbf{x}; \mathbf{y}; \mathbf{z})$ of W_k, since $S = (V_1; V_2; V_3; V_4)$ has full rank.

When W_k is undesired, we have $\forall n \in [1:4]$,

$$\text{Server } n: \; Q_n^{[k^c]}(W_k) = \mathbb{B}(\mathcal{U}_n), \quad A_n^{[k^c]}(W_k) = Q_n^{[k^c]}(W_k)W_{kn}. \tag{108}$$

b) Interfering symbols are dependent and have dimension at most 11*:* Consider the interfering symbols along the common vector $\overline{U}_1$. Note that

$$\overline{U}_1\mathbf{x} + \overline{U}_1\mathbf{y} + \overline{U}_1\mathbf{z} = \overline{U}_1(\mathbf{x}+\mathbf{y}+\mathbf{z}) \tag{109}$$

Since at least 1 interfering symbol is a linear combination of the rest, the 12 interfering symbols cannot have more than 11 dimensions, i.e., their joint entropy is no more than 11 in p-ary units.

3) Combining Answers, Correctness and Rate: The combining process and correctness proof are similar to that in Theorem 1 except that the combining matrices C_n are chosen in a uniformly random manner now (so the matrices are no longer deterministic). We will show in Appendix C that independent and uniformly random choices of C_n are enough to guarantee that as the field size approaches infinity, i.e., $p \to \infty$, the probability of error, $P_e \to 0$. The reasoning for the rate calculation is as follows. We repeat the above query construction four times independently such that each server has $6 \times 4 = 24$ symbols (12 in W_1 and 12 in W_2). These 24 symbols at each server are combined to 23 downloaded symbols, $A_n^{[k]}$ and it is ensured that we can almost surely decode all interfering symbols and then extract the desired symbols. Thus, the rate achieved is 12/23.

4) Privacy Proof: Since the privacy proof is a bit more involved now, let us use this example to introduce the key ideas. To show that the scheme is private to any $T = 3$ colluding servers, it suffices to show that the queries for W_k for any $T = 3$ servers are identically distributed, regardless of which message is desired. Consider 3 distinct indices $i, j, l, i < j < l$ in $[1:4]$, we require

$$\begin{aligned} &\left(Q_i^{[k]}(W_k), Q_j^{[k]}(W_k), Q_l^{[k]}(W_k)\right)\\ &\quad\sim \left(Q_i^{[k^c]}(W_k), Q_j^{[k^c]}(W_k), Q_l^{[k^c]}(W_k)\right) && (110)\\ &\quad\Longleftrightarrow \left(\mathbb{B}(\mathcal{V}_i), \mathbb{B}(\mathcal{V}_j), \mathbb{B}(\mathcal{V}_l)\right)\\ &\quad\sim \left(\mathbb{B}(\mathcal{U}_i), \mathbb{B}(\mathcal{U}_j), \mathbb{B}(\mathcal{U}_l)\right) && (111)\end{aligned}$$

Note that

$$\begin{aligned} &\left(\mathbb{B}(\mathcal{V}_i), \mathbb{B}(\mathcal{V}_j), \mathbb{B}(\mathcal{V}_l)\right)\\ &\quad= \left(\mathbb{B}(\{V_m, V_j, V_l\}), \mathbb{B}(\{V_m, V_i, V_l\}), \mathbb{B}(\{V_m, V_i, V_j\})\right) && (112)\end{aligned}$$

where $m \notin \{i, j, l\}, m \in [1:4]$. To prove (111), we wish to transform the spaces on the RHS to the form that is the same as (112). To this end, we first compute the vectors that lie in the span of both $\mathbb{B}(\mathcal{U}_i)$ and $\mathbb{B}(\mathcal{U}_j)$, $i < j$. Note that the matrix P is designed such that except $\overline{U}_1$, we have only one such vector (up to scaling), denoted as $U_{\{i,j\}}$. $U_{\{i,j\}}$ are computed explicitly as follows. Further, we fix the scaling factor such that the $U_{\{i,j\}}$ vector is unique.

$$\begin{aligned} U_{\{1,2\}} &= U_1 + U_2 && (113)\\ U_{\{1,3\}} &= U_1 - U_2 && (114)\\ U_{\{1,4\}} &= U_1 && (115)\\ U_{\{2,3\}} &= U_1 + U_2 + 2U_3 && (116)\\ U_{\{2,4\}} &= U_1 + U_2 + U_3 && (117)\\ U_{\{3,4\}} &= U_2 + U_3 && (118)\end{aligned}$$

It is easy to verify that $U_{\{i,j\}}, U_{\{i,l\}}, U_{\{j,l\}}, i, j, l \in [1:4], i < j < l$, are linearly independent, i.e.,

$$\begin{aligned} (i, j, l) &= (1,2,3): \operatorname{rank}(U_{\{1,2\}}; U_{\{1,3\}}; U_{\{2,3\}})\\ &= \operatorname{rank}(U_1 + U_2; U_1 - U_2; U_1 + U_2 + 2U_3) = 3\\ (i, j, l) &= (1,2,4): \operatorname{rank}(U_{\{1,2\}}; U_{\{1,4\}}; U_{\{2,4\}})\\ &= \operatorname{rank}(U_1 + U_2; U_1; U_1 + U_2 + U_3) = 3\\ (i, j, l) &= (1,3,4): \operatorname{rank}(U_{\{1,3\}}; U_{\{1,4\}}; U_{\{3,4\}})\\ &= \operatorname{rank}(U_1 - U_2; U_1; U_2 + U_3) = 3\\ (i, j, l) &= (2,3,4): \operatorname{rank}(U_{\{2,3\}}; U_{\{2,4\}}; U_{\{3,4\}})\\ &= \operatorname{rank}(U_1+U_2+2U_3; U_1+U_2+U_3; U_2+\upsilon U_3) = 3 && (119)\end{aligned}$$

As a result, we may equivalently represent $Q_i^{[k^c]}(W_k)$ as

$$\begin{aligned} Q_i^{[k^c]}(W_k) &= \mathbb{B}(\mathcal{U}_i) = \mathbb{B}(\{\overline{U}_1, U_{\{i,j\}}, U_{\{i,l\}}\}),\\ &\forall i, j, l \in [1:4], i \neq j, i \neq l, j \neq l && (120)\end{aligned}$$

Note that equipped with this representation, $\big(\mathbb{B}(\mathcal{U}_i), \mathbb{B}(\mathcal{U}_j), \mathbb{B}(\mathcal{U}_l)\big)$ is now of the same form as $\big(\mathbb{B}(\mathcal{V}_i), \mathbb{B}(\mathcal{V}_j), \mathbb{B}(\mathcal{V}_l)\big)$ and we are now ready to prove the privacy condition (111).

$$\begin{aligned}(111) \Longleftrightarrow &\big(\mathbb{B}(\{V_m, V_j, V_l\}), \mathbb{B}(\{V_m, V_i, V_l\}), \mathbb{B}(\{V_m, V_i, V_j\})\big)\\ \sim\ &\big(\mathbb{B}(\{\overline{U}_1, U_{\{i,j\}}, U_{\{i,l\}}\}), \mathbb{B}(\{\overline{U}_1, U_{\{i,j\}}, U_{\{j,l\}}\}),\\ &\cdots, \mathbb{B}(\{\overline{U}_1, U_{\{i,l\}}, U_{\{j,l\}}\})\big) \end{aligned} \tag{121}$$

Therefore, it suffices to show the following.

$$(V_m, V_i, V_j, V_l) \sim (\overline{U}_1, U_{\{j,l\}}, U_{\{i,l\}}, U_{\{i,j\}}) \tag{122}$$

Because S is uniformly chosen from the set of all full rank matrices, we have

$$(V_m, V_i, V_j, V_l) \sim (V_1, V_2, V_3, V_4) \tag{123}$$

Based on (119), there is a bijection between

$$(\overline{U}_1, U_{\{j,l\}}, U_{\{i,l\}}, U_{\{i,j\}}) \leftrightarrow (\overline{U}_1, U_1, U_2, U_3) \tag{124}$$

Now since $S' = (\overline{U}_1; U_1; U_2; U_3)$ is uniform in all full rank matrices, the above bijection then means that $(\overline{U}_1; U_{\{j,l\}}; U_{\{i,l\}}; U_{\{i,j\}})$ is also uniform in all full rank matrices, i.e.,

$$(\overline{U}_1, U_{\{j,l\}}, U_{\{i,l\}}, U_{\{i,j\}}) \sim (\overline{U}_1, U_1, U_2, U_3) \tag{125}$$

Finally, note that S and S' have the same distribution, so we have

$$(V_1, V_2, V_3, V_4) \sim (\overline{U}_1, U_1, U_2, U_3) \tag{126}$$

Therefore, from (123), (125) and (126), we have proved (122) and (111). ■

VI. Conclusion

We settle a conjecture on the capacity of MDS-TPIR by Freij-Hollanti *et al.* [7] by constructing a scheme that beats the conjectured capacity for one particular instance of MDS-TPIR. The rate achieved by the new scheme is shown to be the best possible rate that can be achieved by any linear scheme for the same MDS storage code. The insights from the achievability and converse arguments allow us to characterize the capacity of a class of MDS-TPIR instances. Through another counterexample, we are also able to prove that the capacity expression cannot be symmetric in T and K_c parameters, i.e., these parameters cannot be interchangeable in general. Nevertheless, the general capacity expression for MDS-TPIR remains unknown.

Appendix

A. Converse for Arbitrary K

In this section, we consider the information theoretic converse of MDS-TPIR, for two scenarios, one with $(K, N, T, K_c) = (K, 4, 2, 2)$ and the other with (K, N, T, K_c) such that $N < T + K_c$. For both scenarios, we provide outer bounds that hold for arbitrary K.

Let us start with two useful lemmas that hold for arbitrary K, N, T, K_c.

Lemma 2: For all $\mathcal{T} \subset [1:N], |\mathcal{T}| = T$ and $k, k' \in [1:K]$,

$$\begin{aligned}&H(A_{\mathcal{T}}^{[k]} | f(W_1, \cdots, W_K), \mathcal{F}, \mathcal{G})\\ &\quad = H(A_{\mathcal{T}}^{[k']} | f(W_1, \cdots, W_K), \mathcal{F}, \mathcal{G})\end{aligned} \tag{127}$$

where $f(W_1, \cdots, W_K)$ represents an arbitrary function of the messages $W_1, \cdots, W_K$.

Proof: First, note that

$$\begin{aligned}&I(Q_{\mathcal{T}}^{[\theta]}, \theta, \mathcal{F}; W_1, \cdots, W_K, \mathcal{G})\\ &\quad\overset{(11)}{=} I(\theta, \mathcal{F}; W_1, \cdots, W_K, \mathcal{G}) &(128)\\ &\quad\overset{(10)}{=} 0 &(129)\\ &\quad\Longrightarrow I(Q_{\mathcal{T}}^{[\theta]}; W_1, \cdots, W_K, \mathcal{G})\\ &\quad\ = I(Q_{\mathcal{T}}^{[\theta]}; W_1, \cdots, W_K, \mathcal{G} | \theta)\\ &\quad\ = 0 &(130)\end{aligned}$$

Next, we have

$$\begin{aligned}&I(\theta; W_1, \cdots, W_K, \mathcal{G}, Q_{\mathcal{T}}^{[\theta]})\\ &\quad= I(\theta; W_1, \cdots, W_K, \mathcal{G})\\ &\qquad + I(\theta; Q_{\mathcal{T}}^{[\theta]} | W_1, \cdots, W_K, \mathcal{G}) &(131)\\ &\quad\overset{(10)}{=} I(\theta; Q_{\mathcal{T}}^{[\theta]} | W_1, \cdots, W_K, \mathcal{G}) &(132)\\ &\quad= H(Q_{\mathcal{T}}^{[\theta]} | W_1, \cdots, W_K, \mathcal{G})\\ &\qquad - H(Q_{\mathcal{T}}^{[\theta]} | W_1, \cdots, W_K, \mathcal{G}, \theta) &(133)\\ &\quad\overset{(130)}{=} H(Q_{\mathcal{T}}^{[\theta]}) - H(Q_{\mathcal{T}}^{[\theta]} | \theta) &(134)\\ &\quad\overset{(13)}{=} 0 &(135)\\ &\quad\overset{(8)(12)}{\Longrightarrow} I(\theta; W_1, \cdots, W_K, \mathcal{G}, Q_{\mathcal{T}}^{[\theta]}, A_{\mathcal{T}}^{[\theta]}) = 0 &(136)\\ &\quad\Longrightarrow (W_1, \cdots, W_K, \mathcal{G}, Q_{\mathcal{T}}^{[k]}, A_{\mathcal{T}}^{[k]})\\ &\qquad \sim (W_1, \cdots, W_K, \mathcal{G}, Q_{\mathcal{T}}^{[k']}, A_{\mathcal{T}}^{[k']}) &(137)\\ &\quad\Longrightarrow H(A_{\mathcal{T}}^{[k]} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G})\\ &\quad= H(A_{\mathcal{T}}^{[k']} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k']}, \mathcal{G}) &(138)\end{aligned}$$

Further note that

$$\begin{aligned}&H(A_{\mathcal{T}}^{[k]} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G})\\ &\quad= H(A_{\mathcal{T}}^{[k]} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{F}, \mathcal{G}) &(139)\\ &\quad= H(A_{\mathcal{T}}^{[k]} | f(W_1, \cdots, W_K), \mathcal{F}, \mathcal{G}) &(140)\end{aligned}$$

where (139) follows from the equality that $I(A_{\mathcal{T}}^{[k]}; \mathcal{F} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G}) = 0$, proved as follows.

$$\begin{aligned}&I(A_{\mathcal{T}}^{[k]}; \mathcal{F} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G})\\ &\quad\leq I(A_{\mathcal{T}}^{[k]}, W_1, \cdots, W_K; \mathcal{F} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G}) &(141)\\ &\quad= I(W_1, \cdots, W_K; \mathcal{F} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G})\\ &\qquad + I(A_{\mathcal{T}}^{[k]}; \mathcal{F} | W_1, \cdots, W_K, f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G}) &(142)\\ &\quad\overset{(8)(12)}{=} I(W_1, \cdots, W_K; \mathcal{F} | f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k]}, \mathcal{G}) &(143)\\ &\quad\leq I(W_1, \cdots, W_K, f(W_1, \cdots, W_K); \mathcal{F} | Q_{\mathcal{T}}^{[k]}, \mathcal{G}) &(144)\\ &\quad= I(W_1, \cdots, W_K; \mathcal{F} | Q_{\mathcal{T}}^{[k]}, \mathcal{G}) &(145)\end{aligned}$$

$$\leq \quad I(W_1, \cdots, W_K; \mathcal{F}, Q_{\mathcal{T}}^{[k]}, \mathcal{G}) \tag{146}$$

$$\stackrel{(10)(11)}{=} 0 \tag{147}$$

Symmetrically, we have

$$H(A_{\mathcal{T}}^{[k']}|f(W_1, \cdots, W_K), Q_{\mathcal{T}}^{[k']}, \mathcal{G}) = H(A_{\mathcal{T}}^{[k']}|f(W_1, \cdots, W_K), \mathcal{F}, \mathcal{G}) \tag{148}$$

Combining (138), (140) and (148), we have proved (127) and the proof of Lemma 2 is complete. ■

Lemma 3: For all $\mathcal{K}_c = \{n_1, n_2, \cdots, n_{K_c}\} \subset [1:N]$,

$$H(A_{\mathcal{K}_c}^{[1]}|W_1, \mathcal{F}, \mathcal{G}) = \sum_{n \in \mathcal{K}_c} H(A_n^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \tag{149}$$

Proof: From (8) and (9), we know that for any K_c servers, the stored information is independent.

$$H(W_{k\mathcal{K}_c}) = \sum_{n \in \mathcal{K}_c} H(W_{kn}), \forall k \in [1:K] \tag{150}$$

$$\stackrel{(10)}{\Longrightarrow} H(W_{2\mathcal{K}_c}, \cdots, W_{K\mathcal{K}_c}|W_1, \mathcal{F}, \mathcal{G}) = \sum_{n \in \mathcal{K}_c} \sum_{k=2}^{K} H(W_{kn}|W_1, \mathcal{F}, \mathcal{G}) \tag{151}$$

As answers are functions of the storage, the answers from any K_c servers are independent as well. Consider two arbitrary subsets of $\mathcal{K}_c$ that have no overlap, $\mathcal{K}_1, \mathcal{K}_2 \subset \mathcal{K}_c, \mathcal{K}_1 \cap \mathcal{K}_2 = \emptyset$.

$$I(A_{\mathcal{K}_1}^{[1]}; A_{\mathcal{K}_2}^{[1]}|W_1, \mathcal{F}, \mathcal{G})$$

$$\leq \quad I(A_{\mathcal{K}_1}^{[1]}; A_{\mathcal{K}_2}^{[1]}, W_{2\mathcal{K}_2}, \cdots, W_{K\mathcal{K}_2}|W_1, \mathcal{F}, \mathcal{G}) \tag{152}$$

$$\stackrel{(11)(12)}{=} I(A_{\mathcal{K}_1}^{[1]}; W_{2\mathcal{K}_2}, \cdots, W_{K\mathcal{K}_2}|W_1, \mathcal{F}, \mathcal{G}) \tag{153}$$

$$\stackrel{(11)(12)}{\leq} I(W_{2\mathcal{K}_1}, \cdots, W_{K\mathcal{K}_1}; W_{2\mathcal{K}_2}, \cdots, W_{K\mathcal{K}_2}|W_1, \mathcal{F}, \mathcal{G}) \tag{154}$$

$$\stackrel{(151)}{=} 0 \tag{155}$$

Using (155) repeatedly, we obtain (149). ■

Next we proceed to the two scenarios. To highlight the parameter K, in this section, the capacity C and the download cost D are denoted as $C(K)$ and $D(K)$, respectively.

1) $(K, N, T, K_c) = (K, 4, 2, 2)$: For the setting with $(K, N, T, K_c) = (K, 4, 2, 2)$, we obtain a recursive upper bound that holds for arbitrary K. This result is stated in the following theorem.

Theorem 4: For the class of MDS-TPIR instances with $(K, N, T, K_c) = (K, 4, 2, 2)$, with arbitrary K, the following recursive relation on the capacity outer bound $\overline{C}(K) \geq C(K)$ holds.

$$\overline{C}(K) \leq \left(1 + \frac{3}{8}\left(\frac{1}{\overline{C}(K-1)}\right) + \left(1 - \left(\frac{2}{3}\right)^{K-1}\right)\frac{3}{4}\right)^{-1}, \quad \forall K \geq 2$$

$$\overline{C}(1) = 1 \tag{156}$$

Proof: Consider an MDS-TPIR instance with $(K, N, T, K_c) = (K, 4, 2, 2)$. When $K = 1$, $\overline{C}(1) = 1$ is a trivial bound on $C(1)$. Next we consider $K \geq 2$. Define

$$\overline{C}(K) = L/H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) \tag{157}$$

$$\overline{C}(K-1) = L/H(A_{1:4}^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \tag{158}$$

$\overline{C}(K)$ is a valid outer bound on $C(K)$, since

$$\overline{C}(K) = L/H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) \geq L/D(K) = C(K) \tag{159}$$

Similarly, $\overline{C}(K-1)$ is a valid outer bound on $C(K-1)$. Now, substituting (157) and (158) to (156), we have

$$H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G})/L \geq 1 + \frac{3}{8}H(A_{1:4}^{[2]}|W_1, \mathcal{F}, \mathcal{G})/L + \left(1 - \left(\frac{2}{3}\right)^{K-1}\right)\frac{3}{4} \tag{160}$$

We proceed to prove (160). To simplify the notation, we define $(W_{1i}, W_{2i}, \cdots, W_{Ki}) = W_{*i}, i \in [1:N]$.

$$H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G})$$

$$\stackrel{(16)}{=} H(A_{1:4}^{[1]}, W_1|\mathcal{F}, \mathcal{G}) + o(L) \tag{161}$$

$$\stackrel{(10)}{=} H(W_1) + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_{2:4}^{[1]}|W_1, A_1^{[1]}, \mathcal{F}, \mathcal{G}) + o(L) \tag{162}$$

$$\geq H(W_1) + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_{3:4}^{[1]}|W_1, W_{*1}, A_1^{[1]}, \mathcal{F}, \mathcal{G}) + o(L) \tag{163}$$

$$\stackrel{(6)(8)(12)}{=} L + H(A_1^{[1]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_{3:4}^{[1]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) + o(L) \tag{164}$$

$$\stackrel{(8)(127)}{=} L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_{3:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) + o(L) \tag{165}$$

Advancing the databases indices, from (165), we have

$$H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) \geq L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_{2:3}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) + o(L) \tag{166}$$

Adding (165) and (166), we have

$$H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) + o(L)$$

$$\geq L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{2}\big(H(A_{3:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) + H(A_{2:3}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G})\big) \tag{167}$$

$$\geq L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{2}\big(H(A_{2:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) + H(A_3^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G})\big) \tag{168}$$

$$\stackrel{(8)(9)(153)}{=} L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{2}H(A_3^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{2}H(A_{2:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) \tag{169}$$

where we use the sub-modular property of entropy functions to obtain (168). Now consider the term $H(A_{2:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G})$. This corresponds to the total download for the setting where we have 3 servers (servers 2, 3 and 4), $K-1$ messages $(W_2, W_3, \cdots, W_K)$, each message is of length $L/2$ and the MDS code is fully replicated (conditioning on W_{*1}, each other server contains the other half information of entropy $L/2$ about each message), i.e., the TPIR setting. W_2 is the desired message. As the capacity of this TPIR setting is $\frac{1}{3}\left(1-\left(\frac{2}{3}\right)^{K-1}\right)^{-1}$ [3], we have[18]

$$H(A_{2:4}^{[2]}|W_1, W_{*1}, \mathcal{F}, \mathcal{G}) \geq 3\left(1-\left(\frac{2}{3}\right)^{K-1}\right)\frac{L}{2} \quad (170)$$

Substituting back to (169) and advancing database indices, we have $\forall i, j \in [1:4], i \neq j$,

$$\begin{aligned} &H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) + o(L) \\ &\quad \geq L + H(A_i^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{2}H(A_j^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + \left(1-\left(\frac{2}{3}\right)^{K-1}\right)\frac{3L}{4} \end{aligned} \quad (171)$$

Adding (171) for all $i, j \in [1:4]$, we have

$$\begin{aligned} &H(A_{1:4}^{[1]}|\mathcal{F}, \mathcal{G}) + o(L) \\ &\geq L + \frac{1}{4}\sum_{i=1}^{4} H(A_i^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \frac{1}{8}\sum_{j=1}^{4} H(A_j^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\quad + \left(1-\left(\frac{2}{3}\right)^{K-1}\right)\frac{3L}{4} \quad (172) \\ &\geq L + \frac{3}{8}H(A_{1:4}^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + \left(1-\left(\frac{2}{3}\right)^{K-1}\right)\frac{3L}{4} \quad (173) \end{aligned}$$

Normalizing both sides by L, we arrive at (160). ■

Two observations from the converse argument are listed below.

1) When we set $K = 2$, we obtain the information theoretic bound $8/13$.

$$\begin{aligned} C(2) &\leq \overline{C}(2) \quad (174) \\ &\overset{(156)}{\leq} (1 + 3/8 \times 1/\overline{C}(1) + (1-2/3) \times 3/4)^{-1} \quad (175) \\ &\overset{(156)}{=} (1 + 3/8 \times 1 + (1-2/3) \times 3/4)^{-1} = 8/13 \quad (176) \end{aligned}$$

2) As $K \to \infty$, the capacity upper bound converges to $5/14$. Since the MDS-TPIR scheme of Freij-Hollanti *et al.* [7] achieves the rate $1/4$ for this setting as $K \to \infty$, we note that the asymptotic optimality of the scheme remains open.

2) (K, N, T, K_c) With $N < T + K_c$: For the setting with (K, N, T, K_c) and $N < T + K_c$, we obtain a recursive upper bound that holds for arbitrary K. This result is stated in the following theorem.

Theorem 5: For the class of MDS-TPIR instances (K, N, T, K_c) such that $N < T + K_c$, with arbitrary K, N, T, K_c, the following recursive relation on the capacity outer bound $\overline{C}(K) \geq C(K)$ holds.

$$\begin{aligned} \overline{C}(K) &\leq \left(1 + \frac{N-T}{N}\left(\frac{1}{\overline{C}(K-1)}\right)\right. \\ &\qquad \left. + (K-1)\left(1 - \frac{N-T}{K_c}\right)\right)^{-1}, \quad \forall K \geq 2 \\ \overline{C}(1) &= 1 \end{aligned} \quad (177)$$

Therefore, for constant N, T, K_c, when $K \to \infty$, $C(K) \leq \overline{C}(K) \leq \frac{1}{(K-1)\left(1-\frac{N-T}{K_c}\right)}$, which decays $\propto 1/K$ such that downloading everything (rate $1/K$) is order optimal.

Proof: Consider an MDS-TPIR instance (K, N, T, K_c) such that $N < T + K_c$. When $K = 1$, $\overline{C}(1) = 1$ is a trivial bound on $C(1)$. Next we consider $K \geq 2$. Define

$$\overline{C}(K) = L/H(A_{1:N}^{[1]}|\mathcal{F}, \mathcal{G}) \quad (178)$$

$$\overline{C}(K-1) = L/H(A_{1:N}^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \quad (179)$$

$\overline{C}(K)$ is a valid outer bound on $C(K)$, since

$$\overline{C}(K) = L/H(A_{1:N}^{[1]}|\mathcal{F}, \mathcal{G}) \geq L/D(K) = C(K) \quad (180)$$

Similarly, $\overline{C}(K-1)$ is a valid outer bound on $C(K-1)$. Now, substituting (178) and (179) to (177), we have

$$\begin{aligned} \frac{H(A_{1:N}^{[1]}|\mathcal{F}, \mathcal{G})}{L} &\geq 1 + \frac{N-T}{N}\frac{H(A_{1:N}^{[2]}|W_1, \mathcal{F}, \mathcal{G})}{L} \\ &\qquad + (K-1)\left(1 - \frac{N-T}{K_c}\right) \end{aligned} \quad (181)$$

We proceed to prove (181). Consider an index set $\mathcal{N} \subset [1:N]$ with cardinality $|\mathcal{N}| = N - T < K_c$. Denote the complement of $\mathcal{N}$ as $\mathcal{N}^c$.

$$\begin{aligned} &H(A_{1:N}^{[1]}|\mathcal{F}, \mathcal{G}) \\ &\overset{(16)}{=} H(A_{1:N}^{[1]}, W_1|\mathcal{F}, \mathcal{G}) + o(L) \quad (182) \\ &\overset{(10)}{=} H(W_1) + H(A_{\mathcal{N}}^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{\mathcal{N}^c}^{[1]}|W_1, A_{\mathcal{N}}^{[1]}, \mathcal{F}, \mathcal{G}) + o(L) \quad (183) \\ &\overset{(6)(149)}{\geq} L + \sum_{n\in\mathcal{N}} H(A_n^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{\mathcal{N}^c}^{[1]}|W_1, W_{*\mathcal{N}}, A_{\mathcal{N}}^{[1]}, \mathcal{F}, \mathcal{G}) + o(L) \quad (184) \\ &\overset{(8)(11)(12)}{=} L + \sum_{n\in\mathcal{N}} H(A_n^{[1]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{\mathcal{N}^c}^{[1]}|W_1, W_{*\mathcal{N}}, \mathcal{F}, \mathcal{G}) + o(L) \quad (185) \\ &\overset{(8)(127)}{=} L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{\mathcal{N}^c}^{[2]}|W_1, W_{*\mathcal{N}}, \mathcal{F}, \mathcal{G}) + o(L) \quad (186) \\ &\overset{(8)(11)(12)}{=} L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{1:N}^{[2]}|W_1, W_{*\mathcal{N}}, \mathcal{F}, \mathcal{G}) + o(L) \quad (187) \\ &\overset{(16)}{\geq} L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1, \mathcal{F}, \mathcal{G}) \\ &\qquad + H(A_{1:N}^{[2]}, W_2|W_1, W_{*\mathcal{N}}, \mathcal{F}, \mathcal{G}) + o(L) \quad (188) \end{aligned}$$

[18] The proof follows from that in [3] by noting that all definitions for the TPIR problem are satisfied.

$$\geq L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1,\mathcal{F},\mathcal{G}) + H(W_2|W_1,W_{*\mathcal{N}},\mathcal{F},\mathcal{G}) + H(A_{1:N}^{[2]}|W_1,W_2,W_{*\mathcal{N}},\mathcal{F},\mathcal{G}) + o(L) \quad (189)$$

$$\overset{(10)(8)(9)}{=} L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1,\mathcal{F},\mathcal{G}) + L(K_c - |\mathcal{N}|)/K_c + H(A_{1:N}^{[2]}|W_1,W_2,W_{*\mathcal{N}},\mathcal{F},\mathcal{G}) + o(L) \quad (190)$$

$$\overset{(8)(11)(12)}{=} L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1,\mathcal{F},\mathcal{G}) + L(K_c - N + T)/K_c + H(A_{\mathcal{N}^c}^{[2]}|W_1,W_2,W_{*\mathcal{N}},\mathcal{F},\mathcal{G}) + o(L) \quad (191)$$

To bound the term $H(A_{\mathcal{N}^c}^{[2]}|W_1,W_2,W_{*\mathcal{N}},\mathcal{F},\mathcal{G})$, we repeat (185) to (191) for messages $W_3,\cdots,W_K$. This gives us

$$H(A_{1:N}^{[1]}|\mathcal{F},\mathcal{G}) \geq L + \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1,\mathcal{F},\mathcal{G}) + L(K-1)\left(1-\frac{N-T}{K_c}\right) + o(L) \quad (192)$$

Consider (192) for all subsets of $[1:N]$ that have exactly $N-T$ elements and average over all such subsets. We have

$$H(A_{1:N}^{[1]}|\mathcal{F},\mathcal{G}) \geq L + \frac{1}{\binom{N}{N-T}} \sum_{\mathcal{N}:|\mathcal{N}|=N-T} \sum_{n\in\mathcal{N}} H(A_n^{[2]}|W_1,\mathcal{F},\mathcal{G}) + L(K-1)\left(1-\frac{N-T}{K_c}\right) + o(L) \quad (193)$$

$$\geq L + \frac{N-T}{N} H(A_{1:N}^{[2]}|W_1,\mathcal{F},\mathcal{G}) + L(K-1)\left(1-\frac{N-T}{K_c}\right) + o(L) \quad (194)$$

Letting $L\to\infty$ and normalizing by L, we have proved (181) and (177). ■

Based on Theorem 5 the following observations are relevant.

1) When we set $K = 2, K_c = N-1$, we obtain the information theoretic bound for Theorem 3, i.e., $(N^2 - N)/(2N^2-3N+T)$.

$$C(2) \leq \overline{C}(2) \quad (195)$$

$$\overset{(177)}{\leq} \left(1+\frac{N-T}{N}\times\frac{1}{\overline{C}(1)} + (2-1)\left(1-\frac{N-T}{N-1}\right)\right)^{-1} \quad (196)$$

$$\overset{(177)}{=} \left(1+\frac{N-T}{N}\times 1+\frac{T-1}{N-1}\right)^{-1} \quad (197)$$

$$= \frac{N^2-N}{2N^2-3N+T} \quad (198)$$

2) As $K\to\infty$, Theorem 5 shows that the capacity decays as $1/K$, so that it converges to 0. As a sanity check, we note that indeed, the MDS-TPIR scheme of Freij-Hollanti *et al.* [7], which does not depend on the number of messages K, does not apply when $N < T + K_c$. Thus, in this case the asymptotic optimality as $K\to\infty$ is trivially settled.

B. Achievability Proof for Theorem 3 When $T = 2$

The proof for the general setting (arbitrary N) follows the same route as the $N=4$ example presented earlier. We assume that each message is comprised of $L = N(N-1)$ independent symbols from a sufficiently large finite field $\mathbb{F}_p$.

1) Storage Code: The $(N-1,N)$ MDS storage code is as follows.

$$W_{kn} \in \mathbb{F}_p^{N\times 1}, k\in[1:2], n\in[1:N] \quad (199)$$

$$W_k = (W_{k1};W_{k2};\cdots;W_{k(N-1)}) \in \mathbb{F}_p^{L\times 1} \quad (200)$$

$$W_{kN} = W_{k1}+W_{k2}+\cdots+W_{k(N-1)} \quad (201)$$

2) Construction of Queries: The query to each server consists of $2(N-1)$ vectors, the first $N-1$ vectors for W_1 ($Q_n^{[k]}(W_1)$) and the last $N-1$ vectors for W_2 ($Q_n^{[k]}(W_2)$). The queries and downloads for $W_k, k\in[1:2]$ are described next.

Denote the set of all full rank $N\times N$ matrices over $\mathbb{F}_p$ as $\mathcal{S}_N$. The user privately chooses two matrices S, S', independently and uniformly from $\mathcal{S}_N$. Label the rows of S as $V_1,\cdots,V_N$, and the rows of S' as $\overline{U}_1,\cdots,\overline{U}_{N-2},U_1,U_2$. Define $\forall n\in[1:N]$

$$\mathcal{V}_n = \{V_1,\cdots,V_{n-1},V_{n+1},\cdots,V_N\} \quad (202)$$

$$\mathcal{U}_n = \{\overline{U}_1,\cdots,\overline{U}_{N-2},\widetilde{U}_n\} \quad (203)$$

where $\widetilde{U}_n, n\in[1:N]$ are the rows of $\widetilde{U}$, obtained as follows.

$$\widetilde{U} = \mathrm{MDS}_{N\times 2}(U_1;U_2) \quad (204)$$

where $\mathrm{MDS}_{N\times 2}$ is an $N\times 2$ matrix such that any two of its rows are linearly independent.

When W_k is desired, we have $\forall n$,

$$\text{Server } n:\ Q_n^{[k]}(W_k) = \pi_n(\mathcal{V}_n),\quad A_n^{[k]}(W_k) = Q_n^{[k]}(W_k)W_{kn}. \quad (205)$$

a) Desired Symbols are independent: From $A_{1:N}^{[k]}(W_k)$, we can recover all $N(N-1)$ symbols of W_k. This is easily seen because the storage is an $(N-1,N)$ MDS code, no query dimension is repeated more than $N-1$ times and the matrix S has full rank.

When W_k is undesired, we have $\forall n$,

$$\text{Server } n:\ Q_n^{[k^c]}(W_k) = \pi'_n(\mathcal{U}_n),\quad A_n^{[k^c]}(W_k) = Q_n^{[k^c]}(W_k)W_{kn}. \quad (206)$$

b) Interfering symbols are dependent and have dimension at most $N(N-1)-(N-2)$: Consider the interfering symbols along the common vectors $\overline{U}_i, i\in[1:N-2]$. Note that

$$\overline{U}_i W_{k1}+\cdots+\overline{U}_i W_{k(N-1)} = \overline{U}_i W_{kN} \quad (207)$$

Therefore $(N-2)$ interfering symbols are linear combinations of the other N^2-2N+2 symbols.

3) Combining Answers for Efficient Download: Based on the queries, each server has $2(N-1)$ symbols, $N-1$ in W_1, $A_n^{[k]}(W_1)$ and $N-1$ in W_2, $A_n^{[k]}(W_2)$ for a total of $L = N(N-1)$ desired symbols and $L = N(N-1)$ undesired symbols. Note that there are at most $N^2 - 2N + 2 \triangleq I$ independent undesired symbols. Exploiting this fact, we will combine the $2(N-1)$ queried symbols from each server into $(I+L)/N$ symbols to be downloaded by the user. Intuitively, $(L+I)/N$ symbols from each server will give the user a total of $L+I$ symbols, from which he can resolve the L desired and I undesired symbols.

Define the following function that maps $2L/N \in \mathbb{Z}_+$ input symbols to $(L+I)/N \in \mathbb{Z}_+$ output symbols.

$$\begin{aligned}&\mathcal{L}^*(X_1, X_2, \cdots, X_{L/N}, Y_1, Y_2, \cdots, Y_{L/N})\\&\quad = (X_1, \cdots, X_{I/N}, Y_1, \cdots, Y_{I/N}, X_{I/N+1} + Y_{I/N+1},\\&\qquad \cdots, X_{L/N} + Y_{L/N})\end{aligned} \tag{208}$$

We formalize the combining process in the following lemma.

Lemma 4: Suppose each server has L/N desired symbols and L/N undesired symbols. Across all servers, the L desired symbols are independent, while the L undesired symbols have dimension at most I, i.e., all L undesired symbols can be expressed as linear combinations of symbols in $\mathbf{s}$, where $\mathbf{s}$ is a set of I symbols. Further, each server contains I/N distinct symbols in $\mathbf{s}$.

The desired and undesired symbols are combined to produce the answers as follows.

$$A_n^{[k]} = \mathcal{L}^*(C_n A_n^{[k]}(W_1), C_n A_n^{[k]}(W_2)) \tag{209}$$

where C_n are deterministic $L/N \times L/N$ matrices, that are required to satisfy the following two properties. Denote the first I/N rows of C_n as $\overline{C}_n$.

P1. All C_n have full rank.

P2. For all $(N-1)!^N$ distinct realizations of $\pi'_n, n \in [1:N]$, the I symbols of the undesired message that are directly downloaded (I/N from each server), $\overline{C}_1 A_1^{[k]}(W_{k^c}), \overline{C}_2 A_2^{[k]}(W_{k^c}), \cdots, \overline{C}_N A_N^{[k]}(W_{k^c})$ are independent in variables in $\mathbf{s}$.

Then we have the following claim.

Claim: The C_n satisfying the two required properties exist over $\mathbb{F}_p$ for a sufficiently large prime p.[19]

The proof of Lemma 4 is deferred to Appendix B7.

Next we prove that the scheme retrieves the desired message, and that it is T private.

4) The Scheme Is Correct (Retrieves Desired Message): Note that from (207), independent undesired message symbols distribute evenly across the databases, such that Lemma 4 applies. Note that the first $2I/N$ variables in the output of the $\mathcal{L}^*$ function are obtained directly, i.e., $\overline{C}_1 A_1^{[k]}(W_1)$, $\overline{C}_2 A_2^{[k]}(W_1), \ldots, \overline{C}_N A_N^{[k]}(W_1)$ and $\overline{C}_1 A_1^{[k]}(W_2)$, $\overline{C}_2 A_2^{[k]}(W_2), \ldots, \overline{C}_N A_N^{[k]}(W_2)$ are all directly recovered. By property *P2* of C_n, $\overline{C}_1 A_1^{[k]}(W_{k^c})$, $\overline{C}_2 A_2^{[k]}(W_{k^c}), \ldots, \overline{C}_N A_N^{[k]}(W_{k^c})$ are linearly independent. Since we have recovered I independent dimensions of interference, and interference only spans at most I dimensions, all interference is recovered and eliminated. Further, since the L desired symbols are independent and since the C_n matrices have full rank, the user is able to recover the L desired message symbols after the interference symbols are recovered and subtracted from the downloaded equations. Therefore the scheme is correct with zero error.

[19]In fact, the properties are generic, i.e., they are satisfied by almost all matrices over large fields.

5) The Scheme Is Private (to any $T = 2$ Colluding Servers): To prove that the scheme is $T = 2$ private (refer to (13)), it suffices to show that the queries for any 2 servers are identically distributed, regardless of which message is desired. Since each query is made up of $2(N-1)$ vectors, $N-1$ for each message and the vectors for W_1 and the vectors for W_2 are generated independently, it suffices to prove that the vectors for one message (say W_k) are identically distributed, i.e.,

$$\left(Q_{n_1}^{[k]}(W_k), Q_{n_2}^{[k]}(W_k)\right) \sim \left(Q_{n_1}^{[k^c]}(W_k), Q_{n_2}^{[k^c]}(W_k)\right), \quad \forall n_1, n_2 \in [1:4], n_1 < n_2 \tag{210}$$

Note that

$$\left(Q_{n_1}^{[k]}(W_k), Q_{n_2}^{[k]}(W_k)\right) = \left(\pi_{n_1}(\mathcal{V}_{n_1}), \pi_{n_2}(\mathcal{V}_{n_2})\right) \tag{211}$$

$$\left(Q_{n_1}^{[k^c]}(W_k), Q_{n_2}^{[k^c]}(W_k)\right) = \left(\pi'_{n_1}(\mathcal{U}_{n_1}), \pi'_{n_2}(\mathcal{U}_{n_2})\right) \tag{212}$$

Therefore, to prove (210) it suffices to show the following.

$$\begin{aligned}&\big(V_1, \cdots, V_{i_{n_1-1}}, V_{i_{n_1+1}}, \cdots, V_{i_{n_2-1}}, V_{i_{n_2+1}},\\&\quad \cdots, V_N, V_{i_{n_2}}, V_{i_{n_1}}\big) \sim \left(\overline{U}_1, \cdots, \overline{U}_{N-2}, \widetilde{U}_{n_1}, \widetilde{U}_{n_2}\right)\end{aligned} \tag{213}$$

Because S is uniformly chosen from the set of all full rank matrices, we have

$$\begin{aligned}&\big(V_1, \cdots, V_{i_{n_1-1}}, V_{i_{n_1+1}}, \cdots, V_{i_{n_2-1}}, V_{i_{n_2+1}},\\&\qquad \cdots, V_N, V_{i_{n_2}}, V_{i_{n_1}}\big) \sim S\end{aligned} \tag{214}$$

Recall that $S = (V_1, \cdots, V_N)$. Next we note that there is a bijection between

$$\left(\overline{U}_1, \cdots, \overline{U}_{N-2}, \widetilde{U}_{n_1}, \widetilde{U}_{n_2}\right) \leftrightarrow S' \tag{215}$$

because of (204) so that there is a bijection between $\widetilde{U}_{n_1}, \widetilde{U}_{n_2}$ and U_1, U_2. Recall that $S' = \left(\overline{U}_1, \cdots, \overline{U}_{N-2}, U_1, U_2\right)$. Now as S' is uniform over all full rank matrices, $\left(\overline{U}_1, \cdots, \overline{U}_{N-2}, \widetilde{U}_{n_1}, \widetilde{U}_{n_2}\right)$ is also uniform over all full rank matrices,

$$\left(\overline{U}_1, \cdots, \overline{U}_{N-2}, \widetilde{U}_{n_1}, \widetilde{U}_{n_2}\right) \sim S' \tag{216}$$

Finally, we note that S and S' are identically distributed, so we have

$$S \sim S' \tag{217}$$

Combining (214), (216) and (217), we arrive at (213) and (210).

6) Rate Achieved Is $(N^2-N)/(2N^2-3N+2)$: The rate achieved is $(N^2-N)/(2N^2-3N+2)$, because we download $2N^2-3N+2$ symbols in total and the desired message size is $N(N-1)$ symbols.

7) Proof of Lemma 4 (Existence of C_n): This proof of existence of C_n will use Schwartz-Zippel lemma [11], [12] about the roots of a polynomial. The variables for the polynomial are the coefficients of the C_n matrices. Let us start with an arbitrary choice of $\pi_n', n \in [1:N]$. Since all $A_n^{[k]}(W_{k^c})$ can be expressed in terms of the I symbols in the vector $\mathbf{s}$ with constant coefficients, we can express

$$(\overline{C}_1 A_1^{[k]}(W_{k^c}); \cdots ; \overline{C}_N A_N^{[k]}(W_{k^c})) = \mathcal{C}_{I\times I}\mathbf{s} \tag{218}$$

Now consider the polynomial given by the determinant of $\mathcal{C}$. This is not the zero polynomial[20] because we can easily assign values to $\overline{C}_n$ to make $\mathcal{C} = I$, the identity matrix. This is because the queried symbols from each server include I/N distinct symbols in $\mathbf{s}$.

Next do the same for *every* realization of $\pi_n', n \in [1:N]$. As there are N permutations involved, and each can take $(N-1)!$ different values, so we have a total of $(N-1)!^N$ different possibilities. We will consider each of them separately. Each time we find a different $\mathcal{C}$, which gives us a different non-zero polynomial.

Next consider the determinant of each C_n. This gives us another N non-zero polynomials.

For each of these $(N-1)!^N + N$ polynomials, Schwartz-Zippel lemma guarantees that a uniformly random choice of C_n produces a non-zero evaluation with high probability over a large field (probability approaching 1 as $p \to \infty$). Since the intersection of finite number of high probability events is also a high probability event, there must exist a realization of C_n over a large field for which all $(N-1)!^N + N$ polynomials simultaneously evaluate to non-zero values, i.e., a realization that satisfies both properties. Hence, the claim is true.

C. Achievability Proof of Theorem 3 When $T > 2$

The proof for the general setting follows the same route as the $N = 4, T = 3$ example presented earlier. We assume that each message is comprised of $L = N(N-1)$ independent symbols from a sufficiently large finite field $\mathbb{F}_p$.

1) Storage Code: The $(N-1, N)$ MDS storage code is as follows.

$$W_{kn} \in \mathbb{F}_p^{N\times 1}, \quad k \in [1:2],\ n \in [1:N] \tag{219}$$

$$W_k = (W_{k1}; W_{k2}; \cdots ; W_{k(N-1)}) \in \mathbb{F}_p^{L\times 1} \tag{220}$$

$$W_{kN} = W_{k1} + W_{k2} + \cdots + W_{k(N-1)} \tag{221}$$

2) Construction of Queries: The query to each server consists of two vector spaces, one for W_1 (span of the rows of $Q_n^{[k]}(W_1)$) and one for W_2 (span of the rows of $Q_n^{[k]}(W_2)$). The queries and downloads for $W_k, k \in [1:2]$ are described next.

Denote the set of all full rank $N\times N$ matrices over $\mathbb{F}_p$ as $\mathcal{S}_N$. The user privately chooses two matrices S, S', independently and uniformly from $\mathcal{S}_N$. Label the rows of S as $V_1, \cdots, V_N$, and the rows of S' as $\overline{U}_1, \cdots, \overline{U}_{N-T}, U_1, \cdots, U_T$. Define $\forall n \in [1:N]$

$$\mathcal{V}_n = \{V_1, \cdots, V_{n-1}, V_{n+1}, \cdots, V_N\} \tag{222}$$

[20] A polynomial is a zero polynomial if all its coefficients are zero.

$$\mathcal{U}_n = \{\overline{U}_1, \cdots, \overline{U}_{N-T}, \widetilde{U}_{(n-1)(T-1)+1}, \cdots, \widetilde{U}_{n(T-1)}\} \tag{223}$$

where $\widetilde{U}_1, \cdots, \widetilde{U}_{N(T-1)}$ are the rows of $\widetilde{U}$, obtained as follows.

$$\widetilde{U} = P(U_1; \cdots ; U_T) \tag{224}$$

P is a deterministic $N(T-1)\times T$ matrix that is chosen in such a way that it allows a bijective mapping between the $\mathcal{U}_n$ or $\mathcal{V}_n$ spaces that may be observed by any set of up to T colluding servers. Intuitively, the only requirement on this matrix is that it is sufficiently 'generic', so that almost all $N(T-1)\times T$ matrices over large finite fields are acceptable. Here unlike the previous example where we explicitly construct the matrix P, we will specify (later) the properties of this matrix and prove that such a matrix exists.

When W_k is desired, we have $\forall n$,

$$\begin{aligned} \text{Server } n: \ \ & Q_n^{[k]}(W_k) = \mathbb{B}(\mathcal{V}_n), \\ & A_n^{[k]}(W_k) = Q_n^{[k]}(W_k) W_{kn}. \end{aligned} \tag{225}$$

a) Desired symbols are independent: From $A_{1:N}^{[k]}(W_k)$, we can recover all $N(N-1)$ symbols of W_k. This is easily seen because the storage is an $(N-1, N)$ MDS code and the matrix S has full rank.

When W_k is undesired, we have $\forall n$,

$$\begin{aligned} \text{Server } n: \ \ & Q_n^{[k^c]}(W_k) = \mathbb{B}(\mathcal{U}_n), \\ & A_n^{[k^c]}(W_k) = Q_n^{[k^c]}(W_k) W_{kn}. \end{aligned} \tag{226}$$

b) Interfering symbols are dependent and have dimension at most $N(N-1)-(N-T)$: Consider the interfering symbols along the common vectors $\overline{U}_i, i \in [1:N-T]$. Note that

$$\overline{U}_i W_{k1} + \cdots + \overline{U}_i W_{k(N-1)} = \overline{U}_i W_{kN} \tag{227}$$

Therefore $(N-T)$ interfering symbols are linear combinations of the other $N^2 - 2N + T$ symbols.

3) Combining Answers for Efficient Download: The idea of combining is the same as the $T = 2$ setting. That is, we will combine the $2(N-1)$ queried symbols from each server into $(2N^2-3N+T)/N = (L+I)/N$ symbols to be downloaded by the user. We will use the same combining function $\mathcal{L}^*$ defined in (208). The difference lies in the combining matrices C_n. For $T = 2$, C_n are deterministic and the scheme has zero-error, while here C_n are random[21] and the scheme has ϵ-error, with ϵ approaching zero as the message size approaches infinity. The combining process is described in the following lemma, which corresponds to Lemma 4 (with differences brought by random C_n accounted).

Lemma 5: Suppose each server has L/N desired symbols and L/N undesired symbols from $\mathbb{F}_p$. Across all servers, the L desired symbols are independent, while the L undesired symbols have dimension at most I, i.e., all L undesired symbols can be expressed as linear combinations of symbols

[21] The reason for choosing random C_n is that the queries are vector spaces (permutations of vector sets no longer suffice) and the number of vector spaces depends on the field size (the number of permutations does not depend on the field size) so that we cannot directly guarantee the existence of one single C_n that works for all vector spaces.

in $\mathbf{s}$, where $\mathbf{s}$ is a set of I symbols. Further, each server contains I/N distinct symbols in $\mathbf{s}$.

The desired and undesired symbols are combined to produce the answers as follows.

$$A_n^{[k]} = \mathcal{L}^*(C_n A_n^{[k]}(W_1), C_n A_n^{[k]}(W_2)) \tag{228}$$

where C_n are random $L/N \times L/N$ matrices, that are required to satisfy the following two properties. Denote the first I/N rows of C_n as $\overline{C}_n$.

P1. All C_n are full rank.

P2. The I symbols of the undesired message that are directly downloaded (I/N from each server), $\overline{C}_1 A_1^{[k]}(W_{k^c})$, $\overline{C}_2 A_2^{[k]}(W_{k^c}), \cdots, \overline{C}_N A_N^{[k]}(W_{k^c})$ are independent in variables in $\mathbf{s}$.

Then the following claim must be true.

Claim: The probability that $C_n, n \in [1:N]$ with each element chosen independently and uniformly over $\mathbb{F}_p$, satisfy the two required properties, approaches 1 as $p \to \infty$.

The proof of Lemma 5 is deferred to Appendix C7.

Next we prove that the scheme retrieves the desired message, and that it is T private.

4) The Scheme Is Correct (Retrieves Desired Message): Note that from (227), independent undesired message symbols distribute evenly across the databases, such that Lemma 5 applies. Note that the first $2I/N$ variables in the output of the $\mathcal{L}^*$ function are obtained directly, i.e., $\overline{C}_1 A_1^{[k]}(W_1)$, $\overline{C}_2 A_2^{[k]}(W_1), \cdots, \overline{C}_N A_N^{[k]}(W_1)$ and $\overline{C}_1 A_1^{[k]}(W_2)$, $\overline{C}_2 A_2^{[k]}(W_2)$, $\cdots, \overline{C}_N A_N^{[k]}(W_2)$ are all directly recovered. By property *P2* of C_n, $\overline{C}_1 A_1^{[k]}(W_{k^c}), \overline{C}_2 A_2^{[k]}(W_{k^c}), \cdots, \overline{C}_N A_N^{[k]}(W_{k^c})$ are linearly independent with probability approaching 1 as $p \to \infty$. Since we have recovered I independent dimensions of interference, and interference only spans at most I dimensions, all interference is recovered and eliminated. Further, since the L desired symbols are independent and since the C_n matrices have full rank, the user is able to recover the L desired message symbols after the interference symbols are recovered and subtracted from the downloaded equations. Therefore the scheme is correct with a probability of error ϵ that approaches 0 as the field size p approaches infinity. Note that since each message is comprised of L independent and uniformly random symbols in $\mathbb{F}_p$, as p approaches infinity, the size of each message also approaches infinity. So, given any $\epsilon > 0$, we can find a sufficiently large p, and a correspondingly large message size value such that the probability of error of the scheme described above, is less than ϵ.

5) The Scheme Is Private (to any T Colluding Servers): To prove that the scheme is T private (refer to (13)), it suffices to show that the queries for any T servers are identically distributed, regardless of which message is desired. Since each query is made up of two vector spaces, one for each message and the two vector spaces are generated independently, it suffices to prove that the query spaces for one message (say W_k) are identically distributed whether it is desired or undesired. Consider an index set $\mathcal{T} = \{i_1, i_2, \cdots, i_T\} \subset [1:N]$ such that $i_1 < i_2 < \cdots < i_T$. For all $\mathcal{T}$, we require

$$\left(Q_{i_1}^{[k]}(W_k), \cdots, Q_{i_T}^{[k]}(W_k)\right) \sim \left(Q_{i_1}^{[k^c]}(W_k), \cdots, Q_{i_T}^{[k^c]}(W_k)\right) \tag{229}$$

$$\iff \left(\mathbb{B}(\mathcal{V}_{i_1}), \cdots, \mathbb{B}(\mathcal{V}_{i_T})\right) \sim \left(\mathbb{B}(\mathcal{U}_{i_1}), \cdots, \mathbb{B}(\mathcal{U}_{i_T})\right) \tag{230}$$

Note that

$$\begin{aligned}&\left(\mathbb{B}(\mathcal{V}_{i_1}), \mathbb{B}(\mathcal{V}_{i_2}), \cdots, \mathbb{B}(\mathcal{V}_{i_T})\right)\\ &= \left(\mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_2}, \cdots, V_{i_T}\}), \mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_1}, V_{i_3}, \cdots, V_{i_T}\}),\right.\\ &\quad \left.\times \mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_1}, \cdots, V_{i_{T-1}}\})\right) \end{aligned}\tag{231}$$

Next we transform the spaces on the RHS of (230) to the form that is the same as (231). In particular, any $V_{i_t}, t \in [1:T]$ vector appears in $T-1$ terms in (231). We wish to find such vectors in the $\mathcal{U}_{i_t}$ spaces. To do this, we require the matrix P to satisfy the following properties.

P1. For all $\mathcal{T}^* = \{j_1, j_2, \cdots, j_{T-1}\} \subset [1:N], |\mathcal{T}^*| = T-1$, $j_1 < j_2 < \cdots < j_{T-1}$, there exists a function $m_{\mathcal{T}^*}(P)$ that returns a non-zero vector which lies simultaneously in the spans of each of $P_{j_t} \triangleq P((j_t - 1)(T-1)+1 : j_t(T-1), :)$, $t \in [1:T-1]$. Note that $m_{\mathcal{T}^*}(P)$ is a $1 \times T$ row vector that only depends on P (it does not depend on U).

P2. For each $\mathcal{T} = \{i_1, i_2, \cdots, i_T\} \subset [1:N]$, the vectors $m_{\mathcal{T}^*}(P), \forall \mathcal{T}^* \subset \mathcal{T}, |\mathcal{T}^*| = T-1$ (found in *P1*) are linearly independent. Equivalently, we require the following $T \times T$ matrix to have full rank.

$$\begin{aligned}P_{\mathcal{T}} \triangleq (&m_{\{i_{[1:T]/\{T\}}\}}(P); m_{\{i_{[1:T]/\{T-1\}}\}}(P);\\ &\cdots; m_{\{i_{[1:T]/\{1\}}\}}(P))\end{aligned} \tag{232}$$

Claim: The P satisfying the two required properties exists over $\mathbb{F}_p$ for a sufficiently large p.

The proof of this claim on the existence of P is deferred to Appendix C8.

Because of the two properties, we may equivalently represent $Q_{i_t}^{[k^c]}(W_k), t \in [1:T]$ as

$$\begin{aligned}Q_{i_t}^{[k^c]}(W_k) = \mathbb{B}(\mathcal{U}_i) = \mathbb{B}(\{&\overline{U}, U_{\{i_{[1:T]/\{1\}}\}},\\ &\cdots, U_{\{i_{[1:T]/\{t-1\}}\}}, U_{\{i_{[1:T]/\{t+1\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}}\}).\end{aligned} \tag{233}$$

We are now ready to prove the privacy condition (230).

$$\begin{aligned}(230) \iff &\left(\mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_2}, \cdots, V_{i_T}\}),\right.\\ &\left.\times \mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_1}, V_{i_3}, \cdots, V_{i_T}\}), \mathbb{B}(\{V_{\mathcal{T}^c}, V_{i_1}, \cdots, V_{i_{T-1}}\})\right)\\ &\sim \Big(\mathbb{B}(\{\overline{U}, U_{\{i_{[1:T]/\{2\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}}\}),\\ &\times \mathbb{B}(\{\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, U_{\{i_{[1:T]/\{3\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}}\}),\\ &\times \cdots, \mathbb{B}(\{\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, \cdots, U_{\{i_{[1:T]/\{T-1\}}\}}\})\Big)\end{aligned} \tag{234}$$

Therefore, it suffices to show the following.

$$\begin{aligned}&(V_{\mathcal{T}^c}, V_{i_1}, V_{i_2}, \cdots, V_{i_T})\\ &\quad \sim (\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, U_{\{i_{[1:T]/\{2\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}})\end{aligned} \tag{235}$$

Because S is uniformly chosen from the set of all full rank matrices, we have

$$(V_{\mathcal{T}^c}, V_{i_1}, V_{i_2}, \cdots, V_{i_T}) \sim (V_1, V_2, \cdots, V_N) \tag{236}$$

Because of Property *P2*, there is a bijection between

$$(\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, U_{\{i_{[1:T]/\{2\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}}) \leftrightarrow (\overline{U}, U) \tag{237}$$

Now since $S' = (\overline{U}; U)$ is uniform in all full rank matrices, the bijection implies that $(\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, U_{\{i_{[1:T]/\{2\}}\}}, \ldots, U_{\{i_{[1:T]/\{T\}}\}})$ is also uniform in all full rank matrices, i.e.,

$$(\overline{U}, U_{\{i_{[1:T]/\{1\}}\}}, U_{\{i_{[1:T]/\{2\}}\}}, \cdots, U_{\{i_{[1:T]/\{T\}}\}}) \sim (\overline{U}, U) \tag{238}$$

Finally, note that S and S' have the same distribution, so we have

$$(V_1, V_2, \cdots, V_N) \sim (\overline{U}, U) \tag{239}$$

Therefore, from (236), (238) and (239), we have proved (235) and (230).

6) Rate Achieved Is $(N^2 - N)/(2N^2 - 3N + T)$: The rate achieved is $(N^2 - N)/(2N^2 - 3N + T)$, because we download $2N^2 - 3N + T$ symbols in total and the desired message size is $N(N - 1)$ symbols.

7) Proof of Lemma 5 (Existence of C_n): Without loss of generality, we assume that I/N is an integer. There is no loss of generality because if I/N is not an integer, we may repeat the scheme a number of times (say M) such that IM/N becomes an integer.

The proof relies on Schwartz-Zippel lemma [11], [12] about the roots of a polynomial. The variables for the polynomial are the coefficients of the C_n matrices. Consider an arbitrary realization of the query spaces $\mathcal{U}_n$. Generate uniformly random C_n, independent of $\mathcal{U}_n$. Given $\mathcal{U}_n, n \in [1 : N]$, since all $A_n^{[k]}(W_{k^c})$ can be expressed in terms of the I symbols of the vector $\mathbf{s}$ with constant coefficients, we can express

$$(\overline{C}_1 A_1^{[k]}(W_{k^c}); \cdots ; \overline{C}_N A_N^{[k]}(W_{k^c})) = \mathcal{C}_{I\times I}\mathbf{s} \tag{240}$$

Now consider the polynomial given by the determinant of $\mathcal{C}$. This is not the zero polynomial because we can easily assign values to $\overline{C}_n$ to make $\mathcal{C} = I$, the identity matrix. This is because each server contains I/N distinct symbols in $\mathbf{s}$. By the Schwartz-Zippel lemma, a non-zero polynomial evaluates to a non-zero value with probability approaching 1 as the field size p increases and C_n are chosen uniformly over $\mathbb{F}_p$. Therefore Property $P2$ is satisfied with high probability.

Next consider the determinant of each C_n. This gives us another N non-zero polynomials. When we choose C_n uniformly, the determinant of C_n is not zero almost surely for large p, so that C_n have full rank and Property $P1$ is satisfied with high probability.

Now, because Property $P1$ and $P2$ are each satisfied with probability approaching 1, the probability that the two are simultaneously satisfied also approaches 1 (union bound). Since this is true conditioned on every possible realization of $\mathcal{U}_n, n \in [1 : N]$, it is also true unconditionally.

8) Proof of Existence of P: Similar to the proof of existence of C_n matrices presented earlier, this proof of existence will use Schwartz-Zippel lemma [11], [12] about the roots of a polynomial. The variables for the polynomial are the coefficients of the P matrix. Since P is a $N(T-1)\times T$ matrix, we have a total of $NT(T - 1)$ variables. Define a set $\mathcal{P}$ that is comprised of all non-zero polynomials with $NT(T - 1)$ variables of P as its variables, and coefficients from $\mathbb{F}_p$.

We first consider Property *P1*. Recall that there are $\binom{N}{T-1}$ choices for $\mathcal{T}^*$. Let us start with an arbitrary choice of $\mathcal{T}^* = \{j_1, j_2, \cdots, j_{T-1}\}$ such that $j_1 < j_2 < \cdots < j_{T-1}$. The required non-zero vector $m_{\mathcal{T}^*}(P)$ is found as follows.

$$\begin{aligned} &m_{\mathcal{T}^*}(P) \\ &= H_1 P_{j_1} = H_2\ P_{j_1} = \cdots = H_{T-1} P_{j_{T-1}} \qquad (241)\\ &\Rightarrow \begin{bmatrix} H_1 & H_2 & \cdots & H_{T-1}\end{bmatrix} \underbrace{\begin{bmatrix} P_{j_1} & P_{j_1} & \cdots & P_{j_1} \\ -P_{j_2} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & -P_{j_3} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -P_{j_{T-1}} \end{bmatrix}}_{\triangleq P_{\mathcal{J}}} \\ &= \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} \qquad (242) \end{aligned}$$

where $P_{j_t}, t \in [1 : T - 1]$ are $(T - 1) \times T$ matrices, $\mathbf{0}$ is the $(T - 1) \times T$ matrix with all elements equal to 0 and $P_{\mathcal{J}}$ is a $(T - 1)^2 \times T(T - 2)$ matrix. Note that the left null space of $P_{\mathcal{J}}$ is exactly of one dimension if $P_{\mathcal{J}}$ has full rank. Consider the matrix $P_{\mathcal{J}}^*$, which is a square matrix formed by the last $T(T - 2)$ rows of $P_{\mathcal{J}}$. We claim that the determinant of $P_{\mathcal{J}}^*$ is a non-zero polynomial, i.e., $|P_{\mathcal{J}}^*| \in \mathcal{P}$. This is because we can identify a specific choice of P_{j_t} such that $|P_{\mathcal{J}}^*|$ is not zero, as follows. We set P_{j_t} to be the matrix obtained by inserting an all zero column as the $(T + 1 - t)^{th}$ column of the $(T - 1) \times (T - 1)$ identity matrix $\mathbf{I}_{T-1}$. Equivalently, this means that

$$P_{j_t} U = (U_1; \cdots ; U_{T-t}; U_{T+2-t}; \cdots ; U_T), \quad t \in [1 : T - 1] \tag{243}$$

Since $U_1, \cdots, U_T$ are independent, $m_{\mathcal{T}^*}(P)U$ can only be some scaled version of the U_1 vector. This means that $P_{\mathcal{J}}^*$ has full rank (which is also easily verified by plugging the vaules of P_{j_t} in $P_{\mathcal{J}}^*$). Therefore, $|P_{\mathcal{J}}^*| \in \mathcal{P}$. To make $m_{\mathcal{T}^*}(P)$ a function, i.e., to remove ambiguity due to scaling factors, let us normalize the vector $[H_1, \cdots, H_{T-1}]$ by its first element, h, such that this vector is unique (scaling is fixed). Note that $h \in \mathcal{P}$ because if we use the same special choice of P_{j_t} as above, we find that $h = 1$ (non-zero). With normalized $[H_1, \cdots, H_{T-1}]$, we obtain $m_{\mathcal{T}^*}(P)$. Note that each element of $m_{\mathcal{T}^*}(P)$ also belongs to $\mathcal{P}$.

Now do the same for *every* possible choice of $\mathcal{T}^*$. There are $\binom{N}{T-1}$ possibilities. We will consider each of them separately. Each time we obtain different $|P_{\mathcal{J}}^*|, h \in \mathcal{P}$ and find a different $m_{\mathcal{T}^*}(P)$. Putting all of these together, we have a set of $2\binom{N}{T-1}$ non-zero polynomials.

Next consider Property *P2*. Similarly, we consider all choices of $\mathcal{T}$ separately. For each choice of $\mathcal{T} = \{i_1, i_2, \cdots, i_T\}$ such that $i_1 < i_2 \cdots < i_T$, we consider the determinant of $P_{\mathcal{T}}$. This determinant polynomial is non-zero because we may set $P_{i_t}, t \in [1:T]$ to be the matrix obtained by inserting an all zero column as the $(T+1-t)^{th}$ column of $\mathbf{I}_{T-1}$, such that the common vector $m_{\mathcal{T}^*}(P), \forall \mathcal{T}^* \in \mathcal{T}, |\mathcal{T}^*| = T-1$ can be computed explicitly

$$P_{i_t}U = (U_1; \cdots; U_{T-t}; U_{T+2-t}; \cdots; U_T) \quad (244)$$

$$m_{\{i_{[1:T]/\{t\}}\}}(P) = e_{T+1-t}, \quad \forall t \in [1:T] \quad (245)$$

where e_i represents the $1 \times T$ unit row vector with a 1 in the i^{th} location and 0 at all other locations. Therefore, $P_{\mathcal{T}}$ is an identity matrix and the determinant is 1 (non-zero). With all choices of $\mathcal{T}$, we have another $\binom{N}{T}$ non-zero polynomials.

By Schwartz-Zippel lemma, as the field size grows, for each of the polynomials mentioned above, a uniform choice of P produces a non-zero evaluation with probability approaching 1. By the union bound, the probability that all polynomials simultaneously produce a non-zero value also approaches 1. In particular, for a sufficiently large field this probability is not zero, so there must exist a P matrix that satisfies both properties.

D. Restricted Colluding Sets

Recall that for the setting of our counterexample, i.e., $(K, N, T, K_c) = (2, 4, 2, 2)$, while the linear capacity is settled, the information theoretic capacity remains open. In particular, the best information theoretic capacity upper bound that we were able to obtain is $8/13$. To gain insights into the potential tightness of this bound, here we look into the capacity of this setting with restricted colluding sets, a line of inquiry recently initiated by Tajeddine *et al.* in [13]. Our motivation for studying restricted colluding sets comes from the following observation.

Consider TPIR, for which the capacity is known [3]. The TPIR formulation allows the possibility that *any* set of up to T servers may collude. However, suppose we relax the privacy constraint, by allowing only collusions between cyclically contiguous servers, i.e., the colluding servers must belong to the set of servers indexed $\{n, n+1, \cdots, n+T-1\}$ for some $n \in [1:N]$, with the indices interpreted modulo N. Because of the symmetry that is still maintained across servers, it is readily verified that the converse proof for TPIR in [3] still goes through unchanged. Thus, even though the restriction on colluding sets to cyclically contiguous servers relaxes the privacy constraint, it does not affect the capacity of TPIR.

This leads us to question if a similar property might hold for MDS-TPIR. If so, then we could gain insights into the capacity of MDS-TPIR by imposing similar restrictions on the colluding sets. This line of thought leads us to two somewhat contrasting observations, that are presented in the following two subsections.

1) $(K, N, T, K_c) = (2,4,2,2)$ With Cyclically Adjacent Colluding Sets: Our first observation is in favor of the tightness of the upper bound $8/13$. Indeed, if colluding sets were restricted to cyclically contiguous sets then $8/13$ is the capacity for the MDS-TPIR setting $(K, N, T, K_c) = (2, 4, 2, 2)$. This observation is summarized in a bit more detail next.

For our counterexample we considered the MDS-TPIR setting $(K, N, T, K_c) = (2, 4, 2, 2)$ where any 2 servers may collude. Suppose, now we restrict the colluding sets of servers to cyclically adjacent pairs, i.e., any one of $\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 1\}$. Essentially we have relaxed the privacy constraint by eliminating the possibilities that Server 1 might collude with Server 3, or that Server 2 might collude with Server 4. For this setting, we show that the capacity is $8/13$.

The converse is similar to that with $T = 2$, presented in Section VI-A1. (170) holds with restricted colluding sets when $K = 2$, because we are left with only $K - 1 = 1$ message. All other steps follow similarly because the assumption of symmetry across servers holds under cyclically adjacent colluding sets. As a result, the capacity upper bound of $8/13$ (refer to (176)) holds here.

Next, we summarize the achievable scheme. The message construction and the storage code are specified as follows.

$$W_{kn} \in \mathbb{F}_p^{4\times 1}, \quad k \in [1:2], n \in [1:4] \quad (246)$$

$$W_k = (W_{k1}; W_{k2}) \in \mathbb{F}_p^{8\times 1} \quad (247)$$

$$W_{k3} = W_{k1} + W_{k2}, \quad W_{k4} = W_{k1} + 2W_{k2} \quad (248)$$

The construction of queries is similar to that with $T = 2$ in Section III. The query to each server $Q_n^{[k]}$ is comprised of two parts, $Q_n^{[k]}(W_1), Q_n^{[k]}(W_2)$. Each part contains 2 row vectors, along which the server should project its corresponding stored message symbols. To generate the query vectors, the user privately chooses two matrices, $S = (V_1; V_2; V_3; V_4)$ and $S' = (U_0; U_1; U_2; U_3)$, independently and uniformly from $\mathcal{S}_4$, the set of all full rank 4×4 matrices over $\mathbb{F}_p$. Define

$$\mathcal{V}_1 = \{V_1, V_2\}, \quad \mathcal{U}_1 = \{U_0, U_1 + U_2\} \quad (249)$$

$$\mathcal{V}_2 = \{V_2, V_3\}, \quad \mathcal{U}_2 = \{U_0, U_1 + 2U_2\} \quad (250)$$

$$\mathcal{V}_3 = \{V_3, V_4\}, \quad \mathcal{U}_3 = \{U_0, U_1\} \quad (251)$$

$$\mathcal{V}_4 = \{V_4, V_1\}, \quad \mathcal{U}_4 = \{U_0, U_2\} \quad (252)$$

Independent random orderings of the rows in $\mathcal{V}_n$ are the queries to Server n for the desired message and independent random orderings of the rows in $\mathcal{U}_n$ are the queries to Server n for the undesired message. The rate achieved is $8/13$ because the 8 desired symbols along the V_i vectors are all independent and the 8 undesired symbols occupy only 5 dimensions (the 4 symbols along U_0 contribute only 2 independent dimensions and the remaining 4 symbols contribute only 3 independent dimensions). Privacy follows from the observation that for each cyclically adjacent colluding set of servers, say Server 1 and Server 2, the sets $\mathcal{V}_1, \mathcal{V}_2$ intersect in one of their elements, as do the sets $\mathcal{U}_1, \mathcal{U}_2$, and both are otherwise uniformly random, thus making the distinction of $\mathcal{U}, \mathcal{V}$ invisible to the colluding servers. Note that this scheme is not private to the non-adjacent colluding servers, say Server 1 and Server 3, because, $\mathcal{V}_1, \mathcal{V}_3$ contain no common vectors, while $\mathcal{U}_1, \mathcal{U}_3$ do share a common vector. The remaining details are virtually identical to the settings already covered in Section III and Appendix B and are omitted.

2) Disjoint Colluding Sets of T Servers Each: Our second observation provides a counterpoint to the first observation.

The first observation favored the tightness of $8/13$ bound based on the insight originating from TPIR, that certain restrictions on colluding sets may not affect capacity. The second observation challenges this viewpoint by showing that insights from TPIR do not carry over to MDS-TPIR.

Consider again the TPIR problem. Suppose T divides N, i.e., $mT = N$ for some $m \in \mathbb{Z}_+$, and we partition the N servers into the m disjoint sets of T elements each: $\mathcal{T}_1 = \{1, 2, \cdots, T\}$, $\mathcal{T}_2 = \{T+1, T+2, \cdots, 2T\}$, $\cdots$, $\mathcal{T}_m = \{(m-1)T+1, (m-1)T+2, \cdots, N\}$. Further, suppose we relax the privacy constraint and allow collusions between only those servers that belong to the same $\mathcal{T}_i$, $i \in [1:m]$. Then, note that the TPIR problem with restricted colluding sets becomes equivalent to the PIR problem with $N/T = m$ servers.[22] However, the capacity of PIR with N/T servers is the *same* as the capacity of TPIR with N servers. Therefore, relaxing the privacy constraint by restricting the colluding sets to disjoint sets of cardinality T each, in the manner described above, does not affect the capacity of TPIR. However, as we will show next, the same is not true for MDS-TPIR.

Consider MDS-TPIR with $(K, N, T, K_c) = (2, 4, 3, 2)$, where any $T = 2$ of the $N = 4$ servers may collude. From Theorem 3 we know that the capacity of this setting is $6/11$. However, now suppose we partition the servers into disjoint sets $\mathcal{T}_1 = \{1, 2\}$, $\mathcal{T}_2 = \{3, 4\}$, each of cardinality $T = 2$. Now we allow collusions only between servers in the same $\mathcal{T}_i$ set, i.e., Server 1 can only collude with Server 2, while Server 3 can only collude with Server 4. Then, in contrast to TPIR where such a restriction on colluding sets does not affect the capacity, we now show that with these restricted colluding sets, the capacity of MDS-TPIR changes — it increases from $6/11$ to $4/7$.

The converse for rate $4/7$ is trivial, because the rate can not be higher than that of MDS-PIR with $(K, N, K_c) = (2, 4, 3)$, where privacy needs to be ensured only to each individual server. From [6], we know that the capacity of MDS-PIR with $(K, N, K_c) = (2, 4, 3)$ is $4/7$. Therefore, the upper bound follows.

Next, we consider the achievable scheme. Each message consists of 12 symbols. The storage code is specified as follows.

$$W_{kn} \in \mathbb{F}_p^{4\times 1}, \quad k \in [1:2],\ n \in [1:4] \tag{253}$$
$$W_k = (W_{k1}; W_{k2}; W_{k3}) \in \mathbb{F}_p^{12\times 1} \tag{254}$$
$$W_{k4} = W_{k1} + W_{k2} + W_{k3} \tag{255}$$

The query to each server $Q_n^{[k]}$ is comprised of vectors in $\mathcal{V}_n$ and $\mathcal{U}_n$, given as follows.

$$\mathcal{V}_1 = \{V_1, V_3, V_5\}, \quad \mathcal{U}_1 = \{U_0, U_1, U_2\} \tag{256}$$
$$\mathcal{V}_2 = \{V_1, V_3, V_5\}, \quad \mathcal{U}_2 = \{U_0, U_1, U_2\} \tag{257}$$
$$\mathcal{V}_3 = \{V_2, V_4, V_6\}, \quad \mathcal{U}_3 = \{U_0, U_1, U_2\} \tag{258}$$
$$\mathcal{V}_4 = \{V_2, V_4, V_6\}, \quad \mathcal{U}_4 = \{U_0, U_1, U_2\} \tag{259}$$

where $S = (V_1; V_2; V_3; V_4; V_5; V_6)$ and $S' = (U_0; U_1; U_2; U_3; U_4; U_5)$ are independent and uniform from the set of all full rank 6×6 matrices. The rate achieved is $12/(12+9) = 4/7$ because the 12 desired symbols along the V_i vectors are all independent and the 12 undesired symbols occupy only 9 dimensions (the symbols along each U_i, $i \in \{0, 1, 2\}$, occupy only $K_c = 3$ dimensions). Privacy follows from the observation that for either colluding set $\{1, 2\}$ or $\{3, 4\}$, the vectors in $\mathcal{V}$ and $\mathcal{U}$ are both the same. The remaining details can be filled in based on Section III and Appendix B and are omitted.

In light of the two contrasting observations, the tightness of the $8/13$ upper bound, as well as the general impact of restricted colluding sets on the capacity of MDS-TPIR remain intriguing open problems for future work. For readers interested in the latter problem, we conclude this section with two simple examples of such capacity characterizations.

3) Examples of Capacity of MDS-TPIR Under Restricted Colluding Sets: As usual in this section, we will omit details of achievability arguments that follow directly from Section III and Appendix B.

a) Example 1: Consider the setting $(K, N, K_c) = (2, 4, 2)$ and let the restricted colluding sets be $\{1, 2\}, \{3, 4\}$. Alternatively, let the restricted colluding sets be $\{1, 2\}, \{3\}, \{4\}$. In either case, the capacity is $2/3$, same as that of MDS-PIR with $(K, N, K_c) = (2, 4, 2)$ [6] so that the converse is implied. The scheme that achieves rate $4/6 = 2/3$ is as follows.

$$W_{kn} \in \mathbb{F}_p^{2\times 1}, \quad k \in [1:2],\ n \in [1:4] \tag{260}$$
$$W_k = (W_{k1}; W_{k2}) \in \mathbb{F}_p^{4\times 1} \tag{261}$$
$$W_{k3} = W_{k1} + W_{k2}, W_{k4} = W_{k1} + 2W_{k2} \tag{262}$$
$$\mathcal{V}_1 = \{V_1\}, \quad \mathcal{U}_1 = \{U_0\} \tag{263}$$
$$\mathcal{V}_2 = \{V_1\}, \quad \mathcal{U}_2 = \{U_0\} \tag{264}$$
$$\mathcal{V}_3 = \{V_2\}, \quad \mathcal{U}_3 = \{U_0\} \tag{265}$$
$$\mathcal{V}_4 = \{V_2\}, \quad \mathcal{U}_4 = \{U_0\} \tag{266}$$

where $S = (V_1; V_2)$ and $S' = (U_0; U_1)$ are independently and uniformly chosen from the set of all full rank 2×2 matrices.

b) Example 2: Suppose $(K, N, K_c) = (2, 3, 2)$ and the colluding sets are either $\{1, 2\}, \{2, 3\}$. Alternatively, suppose the colluding sets are $\{1, 2\}, \{3\}$. In both cases, the capacity is $4/7$. The scheme that achieves rate $4/7$ is as follows.

$$W_{kn} \in \mathbb{F}_p^{2\times 1}, \quad k \in [1:2],\ n \in [1:3] \tag{267}$$
$$W_k = (W_{k1}; W_{k2}) \in \mathbb{F}_p^{4\times 1} \tag{268}$$
$$W_{k3} = W_{k1} + W_{k2} \tag{269}$$
$$\mathcal{V}_1 = \{V_1\}, \quad \mathcal{U}_1 = \{U_0\} \tag{270}$$
$$\mathcal{V}_2 = \{V_1, V_2\}, \quad \mathcal{U}_2 = \{U_0, U_1\} \tag{271}$$
$$\mathcal{V}_3 = \{V_2\}, \quad \mathcal{U}_3 = \{U_0\} \tag{272}$$

where $S = (V_1; V_2)$ and $S' = (U_0; U_1)$ are independent and uniformly chosen from the set of all full rank 2×2 matrices over $\mathbb{F}_p$.

For the converse, consider (190). Plugging in $K = 2$, $K_c = 2$, $\mathcal{N} = \{3\}$, $N = 3$, we have

$$D \geq H(A_{1:3}^{[1]} | \mathcal{F}, \mathcal{G}) \tag{273}$$
$$\geq L + H(A_3^{[2]} | W_1, \mathcal{F}, \mathcal{G}) + L/2 + o(L) \tag{274}$$

[22]This is because storage is fully replicated, so that each disjoint set of T colluding servers may be equivalently replaced with 1 server.

Note that (190) still holds when $|\mathcal{N}| = K_c$. Plugging in $K = 2, K_c = 2, \mathcal{N} = \{1, 2\}, N = 3$, we have

$$D \geq H(A_{1:3}^{[1]}|\mathcal{F}, \mathcal{G}) \tag{275}$$

$$\geq L + H(A_1^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + H(A_2^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + o(L) \tag{276}$$

Adding the two inequalities above, we have

$$2D \geq 5L/2 + H(A_1^{[2]}, A_2^{[2]}, A_3^{[2]}|W_1, \mathcal{F}, \mathcal{G}) + o(L) \tag{277}$$

$$\overset{(16)}{\geq} 5L/2 + H(W_2|W_1, \mathcal{F}, \mathcal{G}) + o(L) \tag{278}$$

$$\overset{(10)(6)}{=} 7L/2 + o(L) \tag{279}$$

Normalizing by L and taking limits as L approaches infinity, gives us the upper bound on the rate L/D as $4/7$, which completes the converse.

E. Examples of Optimal Schemes over Small Fields

To highlight that the assumption of large field size (which was made convenience) may not be essential, in this section, we provide two examples of explicit MDS-TPIR capacity achieving schemes over small fields.

1) Example 1: Consider the MDS-TPIR instance with $(K, N, T, K_c) = (2, 3, 2, 2)$. Note that the capacity of this setting is $6/11$, as established in Theorem 3. We provide an alternative achievable scheme for rate $6/11$. In particular, the scheme operates over the binary field and the upload is 4 bits per server (the query to each server takes values in a set with cardinality $2^4 = 16$).

We assume that each message is $L = 6$ bits. Denote $a_1, \cdots, a_6, b_1, \cdots, b_6$ as 12 i.i.d. uniform bits, $a_i, b_i \in \mathbb{F}_2$. Messages W_1, W_2 are defined in terms of these bits as follows.

$$W_1 = (a_1; a_2; a_3; a_4; a_5; a_6), W_2 = (b_1; b_2; b_3; b_4; b_5; b_6) \tag{280}$$

The storage is specified as

$$\text{Server 1}: W_{11} = (a_1; a_2; a_3), \quad W_{21} = (b_1; b_2; b_3) \tag{281}$$

$$\text{Server 2}: W_{12} = (a_4; a_5; a_6), \quad W_{22} = (b_4; b_5; b_6) \tag{282}$$

$$\text{Server 3}: W_{13} = (\alpha_1; \alpha_2; \alpha_3), \quad W_{23} = (\beta_1; \beta_2; \beta_3) \tag{283}$$

where $\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3$ are obtained as follows.

$$\alpha_1 = a_1 + a_2 + a_5, \quad \beta_1 = b_1 + b_2 + b_5 \tag{284}$$

$$\alpha_2 = a_1 + a_3 + a_6, \quad \beta_2 = b_1 + b_3 + b_6 \tag{285}$$

$$\alpha_3 = a_2 + a_4 + a_6, \quad \beta_3 = b_2 + b_4 + b_6 \tag{286}$$

Further define

$$\alpha_4 = \alpha_1 + \alpha_2 + \alpha_3 = a_3 + a_4 + a_5 \tag{287}$$

$$\beta_4 = \beta_1 + \beta_2 + \beta_3 = b_3 + b_4 + b_5 \tag{288}$$

Note that each server stores 3 bits of each message and the storage at any 2 servers is just enough to recover both messages (MDS storage property is satisfied).

Define a function that maps 4 input bits to 3 output bits as follows.

$$\mathcal{L}_3(X_1, X_2, Y_1, Y_2) = (X_1 + Y_2, X_2 + Y_2, Y_1 + Y_2) \tag{289}$$

We now describe the PIR scheme. $\mathcal{F}$ is a uniform random variable in $[1 : 16]$. Depending on the value of $\mathcal{F}$ and the desired message index $\theta \in [1 : 2]$, the user's query is specified by Table I. The double-quotes notation around a random variable represents the *query* about its realization. Note that the queries to Server 1 and Server 2 are the same, regardless of the value of θ and the query to Server 3 is a deterministic function of that to Server 1 and Server 2.

We show that the scheme is both correct and private. The schemes is correct because our scheme satisfies the important property (*P1*) that from the answers $A_1^{[k]}, A_2^{[k]}$, we always know one undesired bit in $A_3^{[k]}$ and then we can extract the 2 desired bits in $A_3^{[k]}$ (because if any 1 of the 4 input bits of the $\mathcal{L}_3$ function is known, the remaining 3 input bits can be solved from the 3 output bits). Combining these 2 desired bits with the other 4 desired bits (2 from Server 1 and 2 from Server 2), we obtain the desired message (easy to verify that these 6 bits are independent). The property (*P1*) is easy to verify. For example, consider $k = 1$ and $\mathcal{F} = 8$. From $A_1^{[1]}, A_2^{[1]}$, we obtain b_1, b_3, b_4, b_5, from which we further obtain $\beta_4 = b_3 + b_4 + b_5$ and β_4 appears in $A_3^{[1]}$. The scheme is private because it is easy to verify that for any 2 servers, the queries are identically distributed no matter which message is desired and then the privacy condition (13) is satisfied.

The scheme downloads 4 bits from Server 1, 4 bits from Server 2 and 3 bits from Server 3. It retrieves 6 desired message bits. Therefore the rate is $6/11$.

2) Example 2: Consider the MDS-TPIR instance with $(K, N, T, K_c) = (2, 4, 3, 2)$. The capacity of this setting turns out to be $4/7$. The rate can not be more than $4/7$ because the capacity of TPIR with $(K, N, T) = (2, 4, 3)$ is $4/7$ as shown in [3] and reducing K_c from 2 to 1 can not hurt.[23] We provide an achievable scheme for rate $4/7$. In particular, the scheme operates over the finite field $\mathbb{F}_{13}$ and the upload is 6 bits per server (the query to each server takes values in a set with cardinality $2^6 = 64$).

We assume that each message is $L = 4$ symbols. Denote $a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4$ as 8 i.i.d. uniform symbols, $a_i, b_i \in \mathbb{F}_{13}$. Messages W_1, W_2 are defined in terms of these symbols as follows.

$$W_1 = (a_1; a_2; a_3; a_4), \quad W_2 = (b_1; b_2; b_3; b_4) \tag{290}$$

The storage is specified as

$$\text{Server 1}: W_{11} = (a_1; a_2), \quad W_{21} = (b_1; b_2) \tag{291}$$

$$\text{Server 2}: W_{12} = (a_3; a_4), \quad W_{22} = (b_3; b_4) \tag{292}$$

$$\text{Server 3}: W_{13} = (\alpha_1; \alpha_2), \quad W_{23} = (\beta_1; \beta_2) \tag{293}$$

$$\text{Server 4}: W_{14} = (\alpha_3; \alpha_4), \quad W_{24} = (\beta_3; \beta_4) \tag{294}$$

where $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \beta_1, \beta_2, \beta_3, \beta_4$ are obtained as follows.

$$\alpha_1 = 3a_1 + 2a_2 + 4a_3 + a_4, \quad \beta_1 = 3b_1 + 2b_2 + 4b_3 + b_4$$

[23]Reducing K_c from 2 to 1 amounts to increasing the storage at each server so that every server stores every message. Since the queries are independent of message realizations, any achievable scheme for MDS-TPIR with $(K, N, T, K_c) = (2, 4, 3, 2)$ continues to be a valid achievable scheme for TPIR with $(K, N, T) = (2, 4, 3)$. Thus, the capacity in the TPIR setting cannot be smaller than the capacity in the corresponding MDS-TPIR setting.

TABLE I
THE SCHEME FOR MDS-TPIR WITH $(K, N, T, K_c) = (2, 3, 2, 2)$

$\mathcal{F}$	Prob.	$Q_1^{[\theta]}$ (Server 1)	$Q_2^{[\theta]}$ (Server 2)	$Q_3^{[1]}$ (Server 3)	$Q_3^{[2]}$ (Server 3)
1	1/16	"a_1, a_2, b_1, b_2"	"a_4, a_5, b_4, b_5"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"
2	1/16	"a_1, a_3, b_1, b_2"	"a_4, a_5, b_4, b_5"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"
3	1/16	"a_1, a_2, b_1, b_3"	"a_4, a_5, b_4, b_5"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"
4	1/16	"a_1, a_3, b_1, b_3"	"a_4, a_5, b_4, b_5"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"
5	1/16	"a_1, a_2, b_1, b_2"	"a_4, a_6, b_4, b_5"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"
6	1/16	"a_1, a_3, b_1, b_2"	"a_4, a_6, b_4, b_5"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"
7	1/16	"a_1, a_2, b_1, b_3"	"a_4, a_6, b_4, b_5"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"
8	1/16	"a_1, a_3, b_1, b_3"	"a_4, a_6, b_4, b_5"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"
9	1/16	"a_1, a_2, b_1, b_2"	"a_4, a_5, b_4, b_6"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"
10	1/16	"a_1, a_3, b_1, b_2"	"a_4, a_5, b_4, b_6"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"
11	1/16	"a_1, a_2, b_1, b_3"	"a_4, a_5, b_4, b_6"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"
12	1/16	"a_1, a_3, b_1, b_3"	"a_4, a_5, b_4, b_6"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"
13	1/16	"a_1, a_2, b_1, b_2"	"a_4, a_6, b_4, b_6"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"
14	1/16	"a_1, a_3, b_1, b_2"	"a_4, a_6, b_4, b_6"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"
15	1/16	"a_1, a_2, b_1, b_3"	"a_4, a_6, b_4, b_6"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_3, \beta_4)$"
16	1/16	"a_1, a_3, b_1, b_3"	"a_4, a_6, b_4, b_6"	"$\mathcal{L}_3(\alpha_3, \alpha_4, \beta_1, \beta_2)$"	"$\mathcal{L}_3(\alpha_1, \alpha_2, \beta_3, \beta_4)$"

TABLE II
THE SCHEME FOR MDS-TPIR WITH $(K, N, T, K_c) = (2, 4, 3, 2)$

Prob.	$Q_1^{[\theta]}$ (Server 1)	$Q_2^{[\theta]}$ (Server 2)	$Q_3^{[\theta]}$ (Server 3)	$Q_4^{[1]}$ (Server 4)	$Q_4^{[2]}$ (Server 4)
1/64	"a_{i_1}, b_{j_1}"	"a_{i_2}, b_{j_2}"	"$\alpha_{i_3}, \beta_{j_3}$"	"$\alpha_{i_4} + \beta_{j_4}$"	"$\alpha_{i'_4} + \beta_{j'_4}$"

i_1, j_1, i_3, j_3 are i.i.d. and uniform in $\{1, 2\}$. i_2, j_2 are i.i.d. and uniform in $\{3, 4\}$.
$i_1, i_2, i_3, j_1, j_2, j_3$ are independent. i_4, j_4, i'_4, j'_4 are determined as follows.

$$
\begin{aligned}
&(i_1, i_2, i_3) = (1, 3, 1) \Rightarrow i_4 = 4, i'_4 = 3, && (j_1, j_2, j_3) = (1, 3, 1) \Rightarrow j_4 = 3, j'_4 = 4\\
&(i_1, i_2, i_3) = (1, 3, 2) \Rightarrow i_4 = 3, i'_4 = 4, && (j_1, j_2, j_3) = (1, 3, 2) \Rightarrow j_4 = 4, j'_4 = 3\\
&(i_1, i_2, i_3) = (1, 4, 1) \Rightarrow i_4 = 3, i'_4 = 4, && (j_1, j_2, j_3) = (1, 4, 1) \Rightarrow j_4 = 4, j'_4 = 3\\
&(i_1, i_2, i_3) = (1, 4, 2) \Rightarrow i_4 = 4, i'_4 = 3, && (j_1, j_2, j_3) = (1, 4, 2) \Rightarrow j_4 = 3, j'_4 = 4\\
&(i_1, i_2, i_3) = (2, 3, 1) \Rightarrow i_4 = 3, i'_4 = 4, && (j_1, j_2, j_3) = (2, 3, 1) \Rightarrow j_4 = 4, j'_4 = 3\\
&(i_1, i_2, i_3) = (2, 3, 2) \Rightarrow i_4 = 4, i'_4 = 3, && (j_1, j_2, j_3) = (2, 3, 2) \Rightarrow j_4 = 3, j'_4 = 4\\
&(i_1, i_2, i_3) = (2, 4, 1) \Rightarrow i_4 = 4, i'_4 = 3, && (j_1, j_2, j_3) = (2, 4, 1) \Rightarrow j_4 = 3, j'_4 = 4\\
&(i_1, i_2, i_3) = (2, 4, 2) \Rightarrow i_4 = 3, i'_4 = 4, && (j_1, j_2, j_3) = (2, 4, 2) \Rightarrow j_4 = 4, j'_4 = 3
\end{aligned}
$$

$$
\begin{aligned}
\alpha_2 &= 2a_1 + 3a_2 + a_3 + 4a_4, \quad \beta_2 = 2b_1 + 3b_2 + b_3 + 4b_4\\
\alpha_3 &= 3a_1 + 12a_2 + 4a_3 + 6a_4, \quad \beta_3 = 3b_1 + 12b_2 + 4b_3 + 6b_4\\
\alpha_4 &= 12a_1 + 3a_2 + 6a_3 + 4a_4, \quad \beta_4 = 12b_1 + 3b_2 + 6b_3 + 4b_4
\end{aligned} \tag{295}
$$

Note that each server stores 2 symbols of each message and the storage at any 2 servers is just enough to recover both messages (MDS storage property is satisfied).

We now describe the PIR scheme. $\mathcal{F}$ is a uniform random variable in [1 : 64]. The user's query is uniform over 64 choices and is specified by Table II. Note that the queries to servers 1, 2 and 3 are the same, regardless of the value of θ and the query to Server 4 is a deterministic function of that to servers 1, 2 and 3.

The key to the scheme is that the undesired symbol downloaded from Server 4 is known based on what is downloaded from servers 1, 2 and 3, while desired symbols are all independent. To satisfy this property, the storage code (i.e., the linear combining coefficients in α_i, β_j) is carefully designed. In particular, it is guaranteed that if we arbitrarily choose one of a_1, a_2 and one of a_3, a_4 and set the chosen variables to zero, α_3 (α_4) is linearly independent of one of α_1, α_2 and is a scaled version of the other one of α_1, α_2. Now, in the PIR scheme, we first download one arbitrary stored symbol from the first three servers. Then the α symbol downloaded from the last server is the independent one (when a_i is desired) and is the dependent one (when a_i is undesired). The same property applies to b_j and β_j as the same code structure is used. This observation is formalized in the following lemma.

Lemma 6: For all values of $i_1, i_2, i_3, i_4, i_4', j_1, j_2, j_3, j_4, j_4'$ in Table II, we have

$$\dim(a_{i_1}, a_{i_2}, \alpha_{i_3}, \alpha_{i_4}) = 4, \quad \dim(a_{i_1}, a_{i_2}, \alpha_{i_3}, \alpha_{i_4'}) = 3 \tag{296}$$

$$\dim(b_{j_1}, b_{j_2}, \beta_{j_3}, \beta_{j_4}) = 3, \quad \dim(b_{j_1}, b_{j_2}, \beta_{j_3}, \beta_{j_4'}) = 4 \tag{297}$$

Lemma 6 is proved by brute force, i.e., verifying (296) and (297) hold for each case.

We show that the scheme is both correct and private. The schemes is correct because as Lemma 6 has proved, the 4 undesired symbols only have dimension 3 and it is easy to see that the 3 undesired symbols in answers from the first 3 servers have dimension 3. Therefore, from the answers $A_1^{[k]}, A_2^{[k]}, A_3^{[k]}$, we always know the undesired symbol in $A_4^{[k]}$. Subtracting the undesired symbol out from $A_4^{[k]}$, we obtain the desired symbol interference freely. Lemma 6 has proved that the 4 desired symbols are independent such that we can recover the desired message. The scheme is private because it is easy to verify that for any 3 servers, the queries are identically distributed no matter which message is desired and then the privacy condition (13) is satisfied.

The scheme downloads 2 symbols from Server 1, Server 2 and Server 3 each, and 1 symbol from Server 4. It retrieves 4 desired message symbols. Therefore the rate is $4/7$.

Let us conclude this example with the observation that this MDS-TPIR instance with $(K, N, T, K_c) = (2, 4, 3, 2)$ is not covered by Theorem 3, but we were still able to find its capacity. Let us also note that we are able to cast this example into a similar framework as Theorem 3 and prove the existence of PIR schemes that achieve the same capacity for the $(x, y) \rightarrow (x, y, x + y, x + 2y)$ MDS storage code, subject to the assumption of a sufficiently large finite field. The details are repetitive, and therefore omitted. However, we believe this example may provide useful insights for further generalizations.

References

[1] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, "Private information retrieval," *J. ACM*, vol. 45, no. 6, pp. 965–981, 1998.

[2] H. Sun and S. A. Jafar, "The capacity of private information retrieval," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4075–4088, Jul. 2017.

[3] H. Sun and S. A. Jafar. (2016). "The capacity of robust private information retrieval with colluding databases." [Online]. Available: https://arxiv.org/abs/1605.00635

[4] H. Sun and S. A. Jafar. (2016). "The capacity of symmetric private information retrieval." [Online]. Available: https://arxiv.org/abs/1606.08828

[5] H. Sun and S. A. Jafar. (2016). "Multiround private information retrieval: Capacity and storage overhead." [Online]. Available: https://arxiv.org/abs/1611.02257

[6] K. Banawan and S. Ulukus. (2016). "The capacity of private information retrieval from coded databases." [Online]. Available: https://arxiv.org/abs/1609.08138

[7] R. Freij-Hollanti, O. Gnilke, C. Hollanti, and D. Karpuk. (2016). "Private information retrieval from coded databases with colluding servers." [Online]. Available: https://arxiv.org/abs/1611.02062

[8] R. Tajeddine, O. W. Gnilke, and S. E. Rouayheb. (2016). "Private information retrieval from MDS coded data in distributed storage systems." [Online]. Available: https://arxiv.org/abs/1602.01458

[9] S. Kumar, E. Rosnes, and A. G. I. Amat. (2016). "Private information retrieval in distributed storage systems using an arbitrary linear code." [Online]. Available: https://arxiv.org/abs/1612.07084

[10] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, "Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1837–1852, Mar. 2012.

[11] J. T. Schwartz, "Fast probabilistic algorithms for verification of polynomial identities," *J. ACM*, vol. 27, no. 4, pp. 701–717, 1980.

[12] R. Zippel, *Probabilistic Algorithms for Sparse Polynomials*. Berlin, Germany: Springer, 1979.

[13] R. Tajeddine, O. W. Gnilke, D. Karpuk, R. Freij-Hollanti, C. Hollanti, and S. E. Rouayheb. (2017). "Private information retrieval schemes for coded data with arbitrary collusion patterns." [Online]. Available: https://arxiv.org/abs/1701.07636

Hua Sun (S'12–M'17) received his B.E. in Communications Engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 2011, M.S. in Electrical and Computer Engineering from University of California Irvine, USA, in 2013, and Ph.D. in Electrical Engineering from University of California Irvine, USA, in 2017. He is an Assistant Professor in the Department of Electrical Engineering at the University of North Texas, USA. His research interests include information theory and its applications to communications, privacy, networking, and storage.

Dr. Sun received the IEEE Jack Keil Wolf ISIT Student Paper Award in 2016, an IEEE GLOBECOM Best Paper Award in 2016, and the University of California Irvine CPCC Fellowship for the year 2011-2012.

Syed Ali Jafar (S'99–M'04–SM'09–F'14) received his B. Tech. from IIT Delhi, India, in 1997, M.S. from Caltech, USA, in 1999, and Ph.D. from Stanford, USA, in 2003, all in Electrical Engineering. His industry experience includes positions at Lucent Bell Labs, Qualcomm Inc. and Hughes Software Systems. He is a Professor in the Department of Electrical Engineering and Computer Science at the University of California Irvine, Irvine, CA USA. His research interests include multiuser information theory, wireless communications and network coding.

Dr. Jafar is a recipient of the New York Academy of Sciences Blavatnik National Laureate in Physical Sciences and Engineering, the NSF CAREER Award, the ONR Young Investigator Award, the UCI Academic Senate Distinguished Mid-Career Faculty Award for Research, the School of Engineering Mid-Career Excellence in Research Award, the School of Engineering Maseeh Outstanding Research Award, the IEEE Information Theory Society Best Paper Award, IEEE Communications Society Best Tutorial Paper Award, IEEE Communications Society Heinrich Hertz Award, and three IEEE GLOBECOM Best Paper Awards. His student co-authors received the IEEE Signal Processing Society Young Author Best Paper Award, and the Jack Wolf ISIT Best Student Paper Award. Dr. Jafar received the UC Irvine EECS Professor of the Year award six times, in 2006, 2009, 2011, 2012, 2014 and 2017 from the Engineering Students Council and the Teaching Excellence Award in 2012 from the School of Engineering. He was a University of Canterbury Erskine Fellow in 2010 and an IEEE Communications Society Distinguished Lecturer for 2013-2014. Dr. Jafar was recognized as a Thomson Reuters Highly Cited Researcher and included by Sciencewatch among The World's Most Influential Scientific Minds in 2014, 2015 and 2016. He served as Associate Editor for IEEE TRANSACTIONS ON COMMUNICATIONS 2004-2009, for IEEE COMMUNICATIONS LETTERS 2008-2009 and for IEEE TRANSACTIONS ON INFORMATION THEORY 2009–2012.