UNIVERSITY OF CALIFORNIA,
IRVINE

On the Capacity of K-Star-Graph Private Information Retrieval

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Electrical and Computer Engineering

by

Yuhang Yao

Thesis Committee:
Professor Syed A. Jafar, Chair
Assistant Professor Zhiying Wang
Assistant Professor Yanning Shen

2024

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

Firstly, I would like to express my greatest gratitude to my advisor, Professor Jafar, for his invaluable guidance and constructive insight that helped me conduct this research.

Secondly, I would like to thank my committee members, Professor Zhiying Wang and Professor Yanning Shen, for their insightful comments and valuable questions that shaped my final thesis.

I also thank my group members, especially Yuxiang Lu, for the useful discussions and help on this research.

My sincere appreciation goes to my family and friends for their unwavering support and encouragement throughout my academic journey.

This work was supported in part by research grants NSF CCF-1907053, CCF-2221379, and ONR N00014-21-1-2386.

In the end, thanks for IEEE for the permission to incorporate my published conference paper into this thesis.

VITA

Yuhang Yao

EDUCATION

Bachelor of Engineering **2017–2021**
Sun Yat-sen University *Guangzhou, China*

RESEARCH EXPERIENCE

Graduate Research Assistant **2021–Present**
University of California, Irvine *Irvine, California*

REFEREED CONFERENCE PUBLICATIONS

Capacity of 4-Star-Graph PIR **2023**
IEEE International Symposium on Information Theory (ISIT) 2023

ABSTRACT OF THE THESIS

On the Capacity of K-Star-Graph Private Information Retrieval

By

Yuhang Yao

Master of Science in Electrical and Computer Engineering

University of California, Irvine, 2024

Professor Syed A. Jafar, Chair

We study the capacity of the K-star-graph private information retrieval (PIR) problem introduced by Sadeh et al. The problem is so labeled because the storage graph corresponds to a star-graph with K edges (corresponding to the edges) and $K+1$ vertices (corresponding to the servers): K messages are separately (one each) stored in K dedicated servers and meanwhile a universal server stores all K messages. While it is one of the simplest PIR settings to describe, the capacity C_K of K-star-graph PIR is open for $K \geq 4$. We study the critical $K = 4$ setting, for which prior work establishes the bounds $2/5 \leq C_4 \leq 3/7$. As our main contribution, we characterize the exact capacity of 4-star-graph PIR as $C_4 = 5/12$, thus improving upon both the prior lower-bound as well as the prior upper-bound. The main technical challenge resides in the new converse bound, whose non-trivial structure is deduced indirectly from the achievable schemes that emerge from the study of a finer tradeoff between the download costs from the dedicated servers versus the universal server. A sharp characterization of this tradeoff is also obtained for $K = 4$. The connection of the PIR problem to caching and interference alignment indicates that our result may provide insight for these problems as well.

Chapter 1

Introduction

1.1 Overview of Private Information Retrieval (PIR)

1.1.1 Private Information Retrieval

Suppose there are K messages replicated into N distributed databases. The goal of Private Information Retrieval (PIR) (See Fig. 1.1) is to allow a user to retrieve one of the messages from the databases while keeping the demand private from the database(s). In terms of the privacy, there are two parallel lines of research. One considers information-theoretic (or perfect) privacy, whereas the other line focuses on computational privacy, where privacy needs to be satisfied only for computationally bounded databases. In this work, we consider the information-theoretic privacy model. If $N = 1$, it is shown in the initial work [1] that there is no way to obtain privacy more than to download all messages from the database. However, when $N = 2$, much more can be done. In fact, a simple way to obtain information-theoretic privacy is as follows. Let W_k denotes the k^{th} message, and let W_Θ be the message that the user demands. The user first generates a random vector $v = [v_1, v_2, \cdots, v_K], v_k \in$

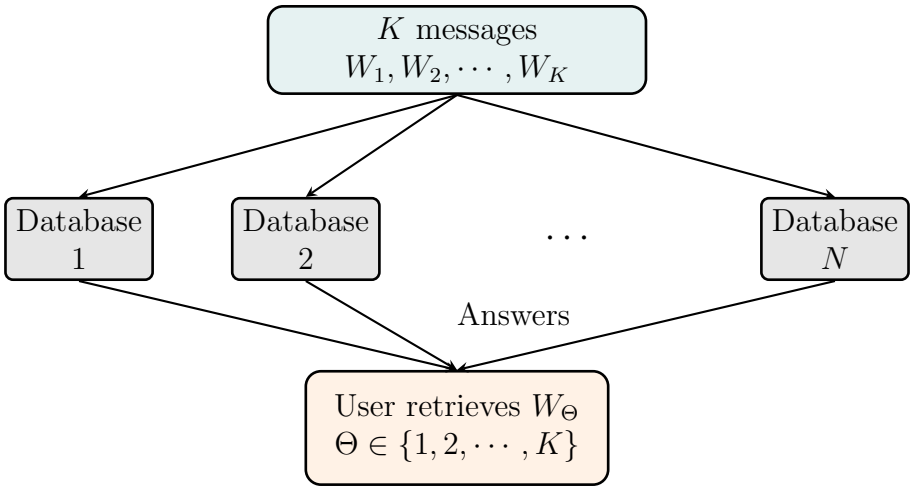

Figure 1.1: Private information retrieval

$\{0,1\}, \forall k \in \{1,2,\cdots,K\}$. The user then requires the sum of messages $\sum_{k\in\{1,2,\cdots,K\}} v_k W_k$ from the first database, and requires the another sum of messages $\sum_{k\in\{1,2,\cdots,K\}\setminus\{\Theta\}} v_k W_k + (1-v_\Theta)W_\Theta$ from the second database. All additions are defined as bitwise XOR. By adding these two answers, the user gets W_Θ since the term $\sum_{k\in\{1,2,\cdots,K\}\setminus\{\Theta\}} v_k W_k$ is cancelled, only leaving W_Θ after decoding. The privacy is guaranteed because to each database, the user requires a sum of messages from a random subset of $\{1,2,\cdots,K\}$, therefore the database can tell nothing about Θ. The communication cost from the databases to the user is referred to as the download cost. In this simple protocol, the download cost is 2 times the size of one message. We further defined the rate of the protocol as $1/2$, since each information bit the user retrieve corresponds to 2 bits of download.

1.1.2 Capacity of PIR

The rate $1/2$ may not be optimal since there exist schemes that achieve a higher rate, especially for small K. The (asymptotically) highest rate is called the capacity. It was first shown by [2] that the capacity of PIR with 2 databases is actually $\left(1+\frac{1}{2}+\frac{1}{4}+\cdots+\frac{1}{2^{K-1}}\right)^{-1}$. Therefore, if there are only $K=3$ messages, the capacity for $N=2$ databases is equal to

4/7, which is higher that the one we previously achieved. [2] also characterizes the general capacity for arbitrary N, K, as

$$C = \left(1 + \frac{1}{N} + \frac{1}{N^2} + \cdots + \frac{1}{N^{K-1}}\right)^{-1}. \tag{1.1}$$

In general, the scheme that achieves the capacity requires the message to be divided into multiple sections, and the download symbols are linear combinations with finer structures.

1.1.3 Variants of PIR

Many PIR schemes are proposed after [2], based on different assumptions on the privacy, storage, database behavior, user-side-information, measure of cost, and so on. In the following, let us provide a brief explanation for some of these models.

1. T-private information retrieval (TPIR): This model is the generalization of PIR setting to guarantee privacy even if any T of the databases collude, meaning that these databases may share the requests they received from the user. These schemes are considered as T-private information retrieval. The capacity is also characterized for T-private information retrieval in [3].

2. Symmetric PIR: This model further assumes that not only the databases cannot learn the information about which message the user wants, but also the user cannot learn anything more than the desired message from the download. The capacity is characterized by [4].

3. PIR with coded storage: In the above settings, one common assumption is that all messages are replicated into all databases. However, this may cause two issues. One is that this may require too much storage at the databases. The other one is that if any database is controlled by a malicious party, then the data will be at risk. To solve

these issues, there are works on PIR with coded storage, where instead of storing all of the messages at all databases, they allow the databases to store coded data. Coded storage can be designed to allow each database to store less data, or be designed to prevent any database(s) from learning useful information about the original messages. The capacity of such PIR schemes are also fully or partially characterized in works such as [5, 6].

4. PIR with graph-based (uncoded) storage: Unlike coded storages, this model deals with PIR in the setting where each database knows a subset of uncoded messages. The storage pattern may be modeled as a (hyper)-graph such that each vertex represents a database, and each edge (or hyper-edge) represents a message that is replicated at the databases that correspond to the vertices of that edge. This problem has been studied in [7–10].

5. PIR with stragglers and Byzantine databases: Considering that the databases may not follow the schemes or protocols, there are works that consider stragglers or Byzantine databases. Stragglers are those databases that do not provide an answer, whereas the Byzantine databases are those databases that provide incorrect answers maliciously. The capacity for these settings are also studied in works such as [11–13].

6. Quantum PIR: This is a branch of PIR works that consider quantum communications between the databases and the user. By enabling quantum transmission from the databases to the user, a rate of 1 PIR can be realized in some regimes [14]. This means the user can retrieve one bit of the desired message per qubit of the download. The user need not have quantum resource before the transmission, but the databases are assumed to establish quantum entanglement in the first place. The capacity of Quantum PIR in various settings are studied in works such as [15–17].

7. Other extensions: There are also many other extensions of the prior PIR works, focusing on other sides of the problem, such as the trade-off between upload cost vs

download cost [18], different capacity definitions, user-side-information [19], and so on [20]. These problems are also very meaningful. In general, the study of PIR problems are motivated not only by growing privacy concerns, but also, and perhaps even more so, by the fundamental connections of PIR to other ideas, such as locally decodable codes, interference alignment, caching, to name a few. These problems are attractive, because they are simple to describe (allowing broader connections), but challenging to solve (allowing deeper insights) — a trait also evident in the celebrated index coding problem [21], similarly simple-to-describe and broadly insightful.

1.2 PIR with Graph-based Storage

The model most closely related to our focus in this work is one of the variants of the PIR problems we previously mentioned, that being PIR with graph-based storage, where the storage at the databases is such that each message is replicated into a subset of databases. Henceforth, we refer to the databases as servers. Recall that the name 'graph' comes from the fact that the storage pattern may be described by a (hyper)-graph. If each each message is only replicated into at most 2 servers, then the corresponding graph is a planer graph. For the capacity related to these graphs, [8] characterizes the capacity associated with two classes of graphs, the complete graphs and the cyclic graphs, as shown in Fig. 1.2 and Fig. 1.3. The capacity for the complete graph is equal to $\min\{2/K, 1/2\}$ and the capacity for the cyclic graph is equal to $2/(K+1)$ where K denotes the number of messages. Note that there is symmetry in the storages for these two types of graphs. On the other hand, the capacity for the star-graphs (Fig. 1.4), is only asymptotically characterized by [10] as $\Theta(1/\sqrt{K})$.

Note that for the star graph with $K+1$ nodes, there are $K+1$ servers and K messages. There are two types of servers. There is one server that stores all messages while the other servers store only message each. This creates asymmetry in the storages. For $K \leq 3$, the exact

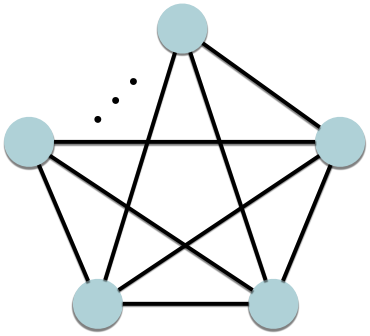

Figure 1.2: A Complete Graph

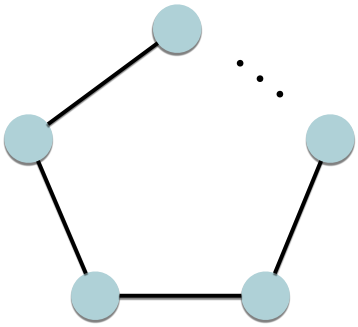

Figure 1.3: A Cyclic Graph

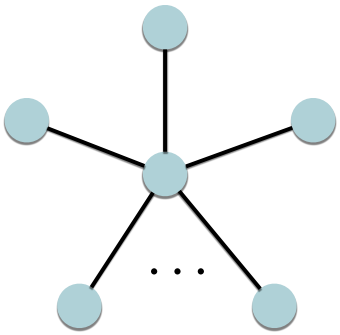

Figure 1.4: A Star Graph

capacity C_K is obtained, whereas for $K = 4$, the known results showed that the capacity is bounded by $2/5 \leq C_4 \leq 3/7$. Besides the non-triviality caused by storage asymmetry, the motivation for studying the capacity of K-star-graph PIR also comes from the connection of the problem to other problems such as caching and retrospective interference alignment. These connections will be mentioned in Section 5.1.

1.3 Contributions of This Work

In this work, we study the capacity of K-star-graph PIR, i.e., PIR with star-graph based storage with K messages. As a main achievement of this work, we successfully characterized the capacity for the case $K = 4$ to be $C_4 = 5/12$. The proof of the result relies on both

the design of a new PIR coding scheme for K-star-graph PIR, and a novel converse proof, thus improving the previously best known lower and upper bounds on the capacity. The result can be also applied to finding the capacity of a caching problem in the many-to-one communication setting, and the design of respective interference alignment schemes in wireless communications.

1.4 Thesis Organization

We present the formal definition of the problems of PIR and K-star-graph PIR in Chapter 2. In Chapter 3, coding schemes for the K-star-graph PIR are proposed. In Chapter 4, we prove several converse bounds for the K-star-graph PIR, in particular, our improved bound for the $K \geq 4$ case. Finally, in Chapter 5, we conclude the work and present examples showing connections of the K-star-graph PIR to other interesting problems.

Chapter 2

Problem Formulation

2.1 Notations

Let $\mathbb{N}$ denote the set of positive integers. For $n \in \mathbb{N}$, $[n]$ denotes the set $\{1, 2, \cdots, n\}$. For $n_1, n_2 \in \mathbb{N}$, $[n_1 : n_2]$ denotes the set $\{n_1, n_1 + 1, \cdots, n_2\}$ if $n_1 \leq n_2$ and $\emptyset$ otherwise. $\binom{n}{r} \triangleq \frac{n!}{r!(n-r)!}$ with the convention that $\binom{n}{r} = 0$ if $n < r$ or $r < 0$. $\mathbb{C}$ denotes the set of complex numbers. $\mathbb{R}_+$ denotes the set of non-negative real numbers. $\mathbb{F}_q$ denotes the finite field with $q = p^r$ a power of a prime. Define compact notations $A^{[n]} \triangleq (A^{(1)}, A^{(2)}, \cdots, A^{(n)})$ and $A_{[n]} \triangleq (A_1, A_2, \cdots, A_n)$. $\mathcal{S}^{a \times b}$ denotes the set of $a \times b$ matrices with elements in $\mathcal{S}$. For a set $\mathcal{N}$, the set of its cardinality-m sub-sets is denoted as $\binom{\mathcal{N}}{m} \triangleq \{\mathcal{A} \subset \mathcal{N} \mid |\mathcal{A}| = m\}$ if $|\mathcal{N}| \geq m$. The notation $2^{\mathcal{N}}$ denotes the power set of $\mathcal{N}$. The notation $f : \mathcal{A} \mapsto \mathcal{B}$ denotes a map f from $\mathcal{A}$ to $\mathcal{B}$. For classical random variables X, Y, Z, $H(X), H(X|Y), I(X;Y)$ and $I(X;Y|Z)$ denote the entropy of X, the conditional entropy of X given Y, the mutual information between X and Y, and the conditional mutual information between X and Y given Z, respectively. By default the base of the logarithm is 2 in the calculations of entropy.

2.2 A General Formulation

The problem of private information retrieval (PIR) contains

- K messages $W_1, W_2, \cdots, W_K$;
- N servers with S_n denoting the storage of Server $n \in [N]$. For $n \in [N]$, S_n is determined by the collection of the messages $(W_1, W_2, \cdots, W_K)$;
- The random index Θ that determines which message is demanded by the user, i.e., the user demands the message W_Θ.
- The local randomness Z at the user;
- The queries $Q_n, n \in [N]$. Q_n is the query sent to Server n for $n \in [N]$. $(Q_1, Q_2, \cdots, Q_N)$ is determined by (Θ, Z). Conditioned on $\Theta = \theta \in [K]$, the query sent to Server n is denoted by Q_n^θ;
- The answers $A_n, n \in [N]$. Given $\Theta = \theta \in [K]$, A_n is the answer from Server n, and A_n is determined by (Q_n, S_n) for $n \in [N]$. Conditioned on $\Theta = \theta \in [K]$, the answer from Server n is denoted by A_n^θ.
- It is also implicitly assumed that Θ is independent of $(W_1, W_2, \cdots, W_K, Z)$.

An information-theoretic PIR scheme must guarantee the following two properties,

- Correctness: W_Θ is determined by $(\Theta, Z, \{A_i\}_{i\in[N]})$;
- Privacy: For $n \in [N]$, Q_n^θ and $Q_n^{\theta'}$ has the same probability distribution for $\theta \neq \theta' \in [K]$. In other words, Q_n is independent of Θ.

The above privacy constraint guarantees information-theoretic privacy when there is no communications between the servers. A stronger privacy constraint may guarantee perfect

privacy even when some subset of servers talk to each other. In general, if the scheme is still private even if a subset $\mathcal{N} \subset [N]$ shares their queries, we say that the scheme can resist to the collusion of the servers in $\mathcal{N} = \{n_1, n_2, \cdots, n_{|\mathcal{N}|}\}$. Mathematically, this means that in addition to the the above constraints posed to privacy, the scheme must also satisfy that $(Q^{\theta}_{n_1}, Q^{\theta}_{n_2}, \cdots, Q^{\theta}_{n_{|\mathcal{N}|}})$ and $(Q^{\theta'}_{n_1}, Q^{\theta'}_{n_2}, \cdots, Q^{\theta'}_{n_{|\mathcal{N}|}})$ has the same probability distribution for $\theta \neq \theta' \in [K]$. For an example that has been widely studied, a scheme call T-private [3] if it can resist to the collusion of *any* subset of T servers.

2.3 PIR with Graph-based Storage

The storage $(S_1, S_2, \cdots, S_N)$ may be specified in several ways. Most PIR works assume that all messages are completely available at all servers, i.e., $S_n = (W_1, W_2, \cdots, W_K)$ for $n \in [N]$. Some PIR works assume coded storage, such as MDS PIR. Coded storage can be used to provide secrecy of the messages, preventing the servers from knowing the information of the messages. If S_n is a subset of uncoded messages for $n \in [N]$, it is called a PIR with graph-based storage, since a (hyper)-graph G can be used to describe the storage. Specifically, each server is mapped to a unique vertex, and each message is mapped to a unique edge, identifying servers that store that message. For a planar graph $G(V, E)$ with V and E denoting the vertex set and the edge set, each message is known by only a pair of servers. In contrast to the extreme setting where all messages are completely available at all servers, PIRs with planar graph based storage represent the other extreme, where each message is known at fewer servers.

2.4 K-star-graph PIR

The problem of K-star-graph PIR is a PIR with graph-based storage when the graph is a K-star graph defined as follows. A K-star graph contains $K+1$ vertices and K edges. The first vertex, indexed with 0 is connected to Vertex i for $i \in [K]$. In other words, the K-star-graph PIR contains $N = K+1$ servers and K messages. There are K dedicated servers that store one message each, with Server k storing only $W_k, k \in [K]$. Besides, there is one universal server, called Server 0 that stores all K messages. Fig. 2.1 illustrates the setting of K-star-graph PIR.

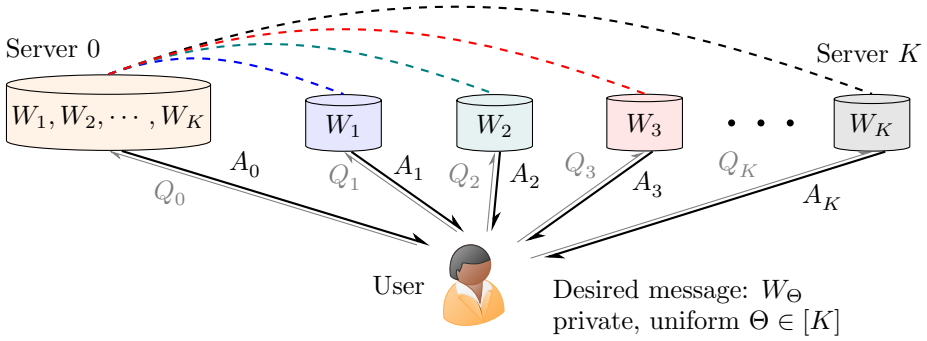

Figure 2.1: K-star-graph PIR with messages $W_1, W_2, \cdots, W_K$ stored among $K+1$ servers according to the K-star storage-graph illustrated with dashed-edges.

2.4.1 Scheme, Rate and Capacity

The main goal of this work is to study the capacity of K-star-graph PIR. To define the capacity, we formulate a coding scheme as follows. A coding scheme is specified by a tuple $\mathcal{C} = (L, Z, \mu_0, \mu_1, \cdots, \mu_K, \phi_0, \phi_1, \cdots, \phi_K, \psi)$. $L \in \mathbb{N}$ denotes the batch size, which is the number of bits contained in one message. $Z \in \mathcal{Z}$ is the local randomness generated privately by the user. $\mu_0, \mu_1, \cdots, \mu_K$ are functions that generate queries sent by the user to the servers. $\phi_0, \phi_1, \cdots, \phi_K$ are functions that generate answers returned by the servers the user. ψ is the

final decoding function. A feasible coding scheme must satisfy the correctness and privacy constraints defined in Section 2.2. The functions and the constraints are described next.

- The $K+1$ functions $\mu_k \colon [K] \times \mathcal{Z} \to \mathcal{Q}_k, \forall k \in [0:K]$, map (Θ, Z) to the $K+1$ queries $Q_0, Q_1, \cdots, Q_K$, respectively, i.e.,

$$Q_k = \mu_k(\Theta, Z), \qquad \forall k \in [0:K]. \tag{2.1}$$

- The $K+1$ functions, $\phi_0 : [(\mathbb{F}_2^L)^K] \times \mathcal{Q}_0 \to \mathcal{A}_0$, and $\phi_k : \mathbb{F}_2^L \times \mathcal{Q}_k \to \mathcal{A}_k, \forall k \in [K]$, map the stored message(s) and the query at each server to the answer that the server returns to the user, i.e.,

$$A_0 = \phi_0(W_1, W_2, \cdots, W_K, Q_0) \ \text{ and } \ A_k = \phi_k(W_k, Q_k), \quad \forall k \in [K]. \tag{2.2}$$

- The decoding function $\phi \colon \mathcal{A}_0 \times \mathcal{A}_1 \times \cdots \times \mathcal{A}_K \times [K] \times \mathcal{Z} \to \mathbb{F}_2^L$ allows the user to retrieve the desired message W_Θ, i.e.,

$$W_\Theta = \psi(A_0, A_1, \cdots, A_K, \Theta, Z). \tag{2.3}$$

- The privacy of the scheme requires

$$I(\Theta; Q_k, A_k, W_1, W_2, \cdots, W_K) = 0, \qquad \forall k \in [0:K]. \tag{2.4}$$

The download cost (measured in bits) from Server k is defined as

$$D_k = \log_2 |\mathcal{A}_k|, \qquad \forall k \in [0:K]. \tag{2.5}$$

Since each alphabet set $\mathcal{A}_k$ is deterministic, our focus in this work is limited to the maximum (instead of average) download costs across queries. The rate achieved by the scheme $\mathcal{C}$ is defined as the ratio

$$R(\mathcal{C}) = \frac{L}{D_0 + D_1 + \cdots + D_K}, \tag{2.6}$$

and the capacity of K-star-graph PIR is then defined as

$$C_K \triangleq \sup_{\mathcal{C} \in \mathfrak{C}_K} R(\mathcal{C}), \tag{2.7}$$

where $\mathfrak{C}_K$ denotes the set of all feasible coding schemes for K-star-graph PIR.

2.4.2 Balanced Download Costs

In general, a scheme may have different costs for different dedicated servers, e.g., $D_0 \neq D_1 \neq D_2$. If $D_1 = D_2 = \cdots = D_K$, i.e., the download costs from all dedicated servers are the same, we say that the scheme has balanced download costs. Otherwise, we say that the scheme has imbalanced download costs. Let $\overline{\mathfrak{C}}_K$ be the set of all coding schemes of K-star-graph PIR with balanced download costs. It is important to note that the inherent symmetry of the K-star-graph guarantees that any rate that can be achieved by a scheme with imbalanced download costs can be also achieved by a scheme with balanced download costs. Therefore, for capacity, there is no loss of generality in restricting the coding schemes to $\overline{\mathfrak{C}}_K$. We formalize it into the following theorem.

Theorem 2.1 (Balanced Cost Achieves Capacity)**.**

$$C_K = \sup_{\mathcal{C} \in \mathfrak{C}_K} R(\mathcal{C}) = \sup_{\mathcal{C} \in \overline{\mathfrak{C}}_K} R(\mathcal{C}). \tag{2.8}$$

Proof. The proof is similar to the proof of [22, Thm. 4] and the idea is called 'time-sharing'.

For a scheme $\mathcal{C} = (L, Z, \mu_0, \mu_1, \cdots, \mu_K, \phi_0, \phi_1, \cdots, \phi_K, \psi) \in \mathfrak{C}_K$ that has download costs $(D_0, D_1, \cdots, D_K)$, let us map it to a scheme $\mathcal{C}' \in \overline{\mathfrak{C}}_K$ with balanced costs, while preserving the rate of the original scheme, i.e., $R(\mathcal{C}') = R(\mathcal{C})$. To do so, let us first define the cyclic permutation functions $\{\pi_k\}_{k\in[0:K-1]}$ as

$$\pi_k(i) \triangleq i + k \mod K, \qquad \forall k \in [K], i \in [K]. \tag{2.9}$$

Then, consider the K schemes $\mathcal{C}_{(0)}, \mathcal{C}_{(1)}, \cdots, \mathcal{C}_{(K-1)}$ with

$$\mathcal{C}_{(k)} \triangleq (L, Z_{(k)}, \mu_0, \mu_{\pi_k(1)}, \cdots, \mu_{\pi_k(K)}, \phi_0, \phi_{\pi_k(1)}, \cdots, \phi_{\pi_k(K)}, \psi), \qquad \forall k \in [0 : K-1]. \tag{2.10}$$

Here, $Z_{(0)}, Z_{(1)}, \cdots, Z_{(K-1)}$ are independent and each has the same distribution as Z. Next, combine the K schemes to a scheme $\mathcal{C}'$ with batch size $L' = KL$. The scheme $\mathcal{C}'$ by definition has download cost $D'_k = D_1 + D_2 + \cdots + D_K$ for Server $k \in [K]$, and $D'_0 = KD_0$ for Server 0. Thus, we obtain that

$$R(\mathcal{C}') = \frac{L'}{D'_0 + D'_1 + \cdots + D'_K} \tag{2.11}$$

$$= \frac{KL}{KD_0 + K(D_1 + \cdots + D_K)} \tag{2.12}$$

$$= \frac{L}{D_0 + D_1 + \cdots + D_K} \tag{2.13}$$

$$= R(\mathcal{C}). \tag{2.14}$$

This shows that the new scheme $\mathcal{C}'$ has balanced download costs from the dedicated servers. The correctness of $\mathcal{C}'$ follows directly from the correctness of $\mathcal{C}$. The privacy of $\mathcal{C}'$ follows from the reasoning that the queries to a server in each $\mathcal{C}_{(k)}$ are individually independent of Θ (because the original scheme $\mathcal{C}$ is private), and conditioned on $\Theta = \theta \in [K]$ the queries to the same server in the different $\mathcal{C}_{(k)}$ schemes are independent because $Z_{(0)}, Z_{(1)}, \cdots, Z_{(K-1)}$

are independent. For example, consider queries to Server 1 for schemes $\mathcal{C}_{(0)}, \mathcal{C}_{(1)}$, namely Q_1, Q_1'. We have by the privacy of $\mathcal{C}$ that $I(\Theta; Q_1) = I(\Theta; Q_1') = 0$, and by the independence of $Z_{(0)}, Z_{(1)}$ that $I(Q_1; Q_1' \mid \Theta) = 0$. This implies that $H(Q_1, Q_1' \mid \Theta) = H(Q_1 \mid \Theta) + H(Q_1' \mid \Theta) - I(Q_1; Q_1' \mid \Theta) = H(Q_1) + H(Q_1') \geq H(Q_1, Q_1')$. Since conditioning cannot increase entropy, we have $H(Q_1, Q_1' \mid \Theta) = H(Q_1, Q_1')$, i.e., Θ is independent of the combined query (Q_1, Q_1') to Server 1. The reasoning extends to any server, and to the combination of all $\mathcal{C}_{(k)}, k \in [0 : K-1]$, i.e., $\mathcal{C}'$. From this it follows that the combined-query in $\mathcal{C}'$ is independent of Θ. In the following, for a scheme $\mathcal{C} \in \overline{\mathfrak{C}}_K$, the download cost for Server 0 is denoted as D_0, and the download cost for Server 1 to Server K is denoted simply as D. □

We finally remark that a scheme that has balanced download cost means that the download costs from all dedicated servers are the same, but the cost from the universal server may still be different. The asymmetry in the storage of dedicated servers and the universal server makes this difference.

2.4.3 Feasible Normalized Cost Region

In order to study the capacity of K-star-graph PIR, it is a first step to study the region $\mathcal{D}_K^*$ of the feasible normalized cost tuple, defined as follows. A tuple $(\Delta_0, \Delta) \in \mathbb{R}_+^2$ is said to be a feasible normalized cost tuple if there exists a coding scheme with batch size L and download costs D_0, D, such that

$$\Delta_0 \geq D_0/L - \epsilon, \text{ and } \Delta \geq D/L - \epsilon \tag{2.15}$$

for any $\epsilon > 0$. The set of such tuples, denoted as $\mathcal{D}_K^*$, is called the feasible normalized cost region for the K-star-graph PIR. Given $\mathcal{D}_K^*$, the capacity of K-star-graph PIR reduces to

the following optimization problem

$$C_K = \left(\min_{(\Delta_0, \Delta) \in \mathcal{D}_K^*} \Delta_0 + K\Delta \right)^{-1}. \tag{2.16}$$

Chapter 3

Coding Schemes for K-star-graph PIR

3.1 Introductory Examples: $K = 3$

In order to show the idea of constructing private transmission schemes for the K-star-graph PIR, let us begin with a small case with $K = 3$. Recall that there are 3 messages W_1, W_2 and W_3, and 4 servers, the universal server, Server 0, and the three dedicated servers, Server 1, Server 2 and Server 3, with Server 0 storing (W_1, W_2, W_3), and Server k storing W_k for $k \in [3]$. For a scheme with batch size L, let us use $W_k(\ell)$ to denote the ℓ-th bit of the message W_k, where $k \in \{1, 2, 3\}$ and $\ell \in [L]$.

3.1.1 Scheme I: $L = 1, D_0 = 0, D = 1$

Consider that each message has one bit. The desired message is W_Θ. The user can require nothing from Server 0, but require $W_1(1)$ from Server 1, $W_2(1)$ from Server 2 and $W_3(1)$ from Server 3. This makes sure that there is nothing revealed about Θ to the servers since

the queries are independent of the queries. The scheme can be written as

$$A_0 = \emptyset, \quad A_k = W_k(1), \quad \forall k \in [3]. \tag{3.1}$$

The download cost from the universal server is $D_0 = 0$, and the download cost from the dedicated servers is $D = 1$. This scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (D_0/L, D/L) = (0, 1)$ is feasible.

3.1.2 Scheme II: $L = 2, D_0 = 1, D = 1$

Consider that each message has two bits. Let the user locally generate z that is uniformly drawn in $\{1, 2\}$, and let $\bar{z} = 3 - z$. The user require $W_1(z) + W_2(z) + W_3(z)$ from the universal server. If the index of the desired message is $\Theta = \theta$, then require $W_\theta(\bar{z})$ from Server θ, and $W_n(z)$ from Server n for $n \neq \theta$. Since z is uniformly distributed in $\{1, 2\}$ and so is $\bar{z}$, it can be easily calculated that the query sent to Server θ or Server $n \neq \theta$ is independent of Θ. This makes sure that the dedicated servers do not know which message is desired by the user. Note that from the downloads, the user can now decode $W_\theta(z)$ from the answers of Servers $n, n \neq \theta$. Besides, the user directly get $W_\theta(\bar{z})$ from the answer of Server θ. Therefore, the user can recover $(W_\theta(1), W_\theta(2))$, which is the whole desired message. The scheme can be concisely written as

$$A_0 = W_1(z) + W_2(z) + W_3(z), \quad A_\theta = W_\theta(\bar{z}), \quad A_n = W_n(z), \quad n \neq \theta, \quad \text{given } \Theta = \theta. \tag{3.2}$$

Note that writing in this way, we can easily tell the queries sent to the servers from the indices in the round parentheses '()'. For example, the query sent to Server 0 is $Q_0^\theta = z$, and the query sent to Server k is $Q_k^\theta = \bar{z}$ for $k \in [3]$, given that $\Theta = \theta$. The download cost from the universal server is $D_0 = 1$, and the download cost from the dedicated servers is

$D = 1$. Since each message contains 2 bits, this scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (D_0/L, D/L) = (1/2, 1/2)$ is feasible. Do note that, if the three dedicated servers collude, i.e., the join together and share the requests they received from the user, they can figure out Θ. To resolve this issue, and to enhance the robustness of the scheme, let $(z_n, \bar{z}_n)$ be uniformly drawn in $\{(1,2),(2,1)\}$ for $n \in [3]$, and independent across different n. Then download

$$A_0 = W_1(z_1) + W_2(z_2) + W_3(z_3), \quad A_\theta = W_\theta(\bar{z}_\theta), \quad A_n = W_n(z_n), \quad n \neq \theta, \quad \text{given } \Theta = \theta. \tag{3.3}$$

This makes sure that even if the three dedicated servers collude, the joint query does not reveal anything about Θ.

3.1.3 Scheme III: $L = 3, D_0 = 3, D = 1$

Consider that each message has three bits. Let the user locally generate

$$(z_1^1, z_1^2, z_1^3), (z_2^1, z_2^2, z_2^3), (z_3^1, z_3^2, z_3^3)$$

being three i.i.d. uniform permutations of $(1,2,3)$. The downloads are then specified as

$$A_0 = \begin{bmatrix} W_1(z_1^2) + W_2(z_2^1) \\ W_1(z_1^3) + W_3(z_3^1) \\ W_2(z_2^3) + W_3(z_3^2) \end{bmatrix}, \quad A_\theta = W_\theta(z_\theta^\theta), \quad A_n = W_n(z_n^\theta), \quad \forall n \neq \theta, \quad \text{given } \Theta = \theta. \tag{3.4}$$

We can similarly tell the queries from the indices in the parentheses. Given $\Theta = \theta$, the query sent to Server 0 is $Q_0^\theta = (z_1^2, z_2^1, z_1^3, z_3^1, z_2^3, z_3^2)$. The query sent to Server θ is $Q_\theta^\theta = z_\theta^\theta$. The queries sent to Server $n \neq \theta$ is $Q_n^\theta = z_n^\theta$. As an example, if $\Theta = 1$, then the query to Server 1

is z_1^1. If $\Theta = 2$, the query to Server 1 is z_1^2. If $\Theta = 3$, the query to Server 1 is z_1^3. Note that z_1, z_2, z_3 have the same distribution. Thus, the query sent to Server 1 is independent of Θ. It can be similarly verified that the scheme is also private even if all dedicated servers collude. This scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (D_0/L, D/L) = (1, 1/3)$ is feasible.

3.1.4 Scheme IV: $L = 1, D_0 = 3, D = 0$

This scheme only needs the download from the universal server. For privacy, the user download all messages from the universal server to hide the index Θ from it. The scheme is written as,

$$A_0 = \begin{bmatrix} W_1(1) \\ W_2(1) \\ W_3(1) \end{bmatrix}, \quad A_n = \emptyset, \quad \forall n \in [3]. \tag{3.5}$$

This scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (D_0/L, D/L) = (3, 0)$ is feasible.

3.1.5 Rate

Let us compute the rates for the 4 schemes, respectively. By definition, the rate can be computed as $R = 1/(\Delta_0 + 3\Delta)$. Thus, Scheme I – Scheme IV have rates $1/3, 1/2, 1/2, 1/3$, respectively. The highest rate we achieved so far is $1/2$. This means that in order to get one bit of the desired message, the user download 2 bits from the 4 servers in total. In fact, the rate of $1/2$ can be proved to be optimal for $K = 3$. The optimality analysis will be given in the next section.

3.1.6 Time-sharing of Schemes

Given two schemes, Scheme_1 with batch size L, download tuple (D_0, D), and Scheme_2 with batch size L', download tuple (D_0', D'), by combining the schemes, one can construct a new scheme with batch size $L + L'$ and download tuple $(D_0 + D_0', D + D')$. This immediately implies the following result: If (Δ_0, Δ) and (Δ_0', Δ') are two feasible cost tuple, then $(\lambda\Delta_0 + (1-\lambda)\Delta_0', \lambda\Delta + (1-\lambda)\Delta')$ is a feasible normalized cost tuple.

3.1.7 Feasible Normalized Cost Region

From Scheme I – Scheme IV, together with the time-sharing argument, we are now able to show that the feasible normalized cost region at least contains the region shown in Fig. 3.1, i.e., any point on and above the curve is a feasible normalized cost tuple. In fact, it will be shown in the next section that this region is equal to $\mathcal{D}_3^*$, i.e., any point that is below the curve is not a feasible normalized download cost tuple.

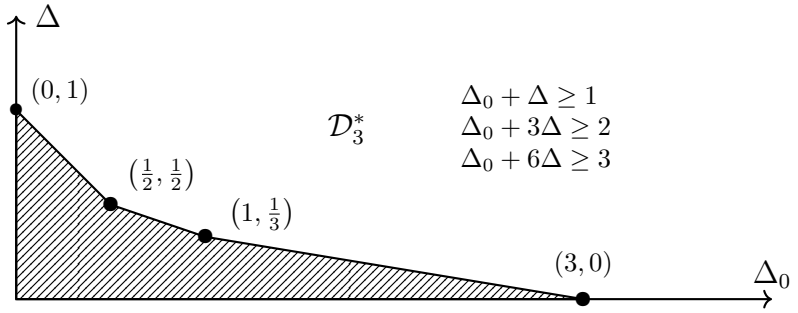

Figure 3.1: Feasible normalized cost region $\mathcal{D}_3^*$. The feasibility is the direct result of the aforementioned 4 schemes and time-sharing. The converse result will be proved by (4.12), (4.20) and (4.30) in the next section.

3.2 Schemes for $K = 4$

Similar to $K = 3$, it is easy to see that $(0, 1)$ and $(4, 0)$ (in general $(K, 0)$) are feasible normalized download cost tuples. Also, it is not difficult to generalize the Scheme II in Section 3.1.2, with downloads specified as

$$A_0 = W_1(z) + W_2(z) + W_3(z) + W_4(z), \quad A_\theta = W_\theta(\bar{z}), \quad A_n = W_n(z), \quad n \neq \theta, \\ \text{given } \Theta = \theta, \tag{3.6}$$

thus showing that $(1/2, 1/2)$ is a feasible normalized download cost tuple also for $K = 4$. Next, we construct another two schemes that work for $K = 4$ and proving another two feasible normalized cost tuple associated with them.

3.2.1 Scheme V: $L = 4, D_0 = 6, D = 1$

Consider that each message is composed of 5 bits. For $k \in [K]$, let $(z_k^1, z_k^2, z_k^3, z_k^4)$ be a uniform permutation of $\{1, 2, 3, 4\}$. Then the downloads are specified as

$$A_0 = \begin{bmatrix} W_1(z_1^2) + W_2(z_2^1) \\ W_1(z_1^3) + W_3(z_3^1) \\ W_1(z_1^4) + W_4(z_4^1) \\ W_2(z_2^3) + W_3(z_3^2) \\ W_2(z_2^4) + W_4(z_4^2) \\ W_3(z_3^4) + W_4(z_4^3) \end{bmatrix}, \; A_\theta = W_\theta(z_\theta^\theta), \; A_n = W_n(z_n^\theta), \; \forall n \neq \theta, \; \text{given } \Theta = \theta. \tag{3.7}$$

This scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (3/2, 1/4)$ is feasible.

3.2.2 Scheme VI: $L = 5, D_0 = 4, D = 2$

Let $(z_1^{\{2,3\}}, z_1^{\{2,4\}}, z_1^{\{3,4\}}, \bar{z}_1^1, \bar{z}_1^2)$ be a uniform permutation of $\{1,2,3,4,5\}$. Similarly, let

$$(z_2^{\{1,3\}}, z_2^{\{1,4\}}, z_2^{\{3,4\}}, \bar{z}_2^1, \bar{z}_2^2), \tag{3.8}$$

$$(z_3^{\{1,2\}}, z_3^{\{1,4\}}, z_3^{\{2,4\}}, \bar{z}_3^1, \bar{z}_3^2), \tag{3.9}$$

$$(z_4^{\{1,2\}}, z_4^{\{1,3\}}, z_4^{\{2,3\}}, \bar{z}_4^1, \bar{z}_4^2) \tag{3.10}$$

be another three uniform permutations of $\{1,2,3,4,5\}$. The downloads are then specified as

$$A_0 = \begin{bmatrix} W_1(z_1^{\{2,3\}}) + W_2(z_2^{\{1,3\}}) + W_3(z_3^{\{1,2\}}) \\ W_1(z_1^{\{2,4\}}) + W_2(z_2^{\{1,4\}}) + W_4(z_4^{\{1,2\}}) \\ W_1(z_1^{\{3,4\}}) + W_3(z_3^{\{1,4\}}) + W_4(z_4^{\{1,3\}}) \\ W_2(z_2^{\{2,4\}}) + W_3(z_3^{\{2,4\}}) + W_4(z_4^{\{2,3\}}) \end{bmatrix}, \tag{3.11}$$

and

$$A_\theta = \begin{bmatrix} W_\theta(\bar{z}_\theta^1) \\ W_\theta(\bar{z}_\theta^2) \end{bmatrix}, \ A_n = \begin{bmatrix} W_n(z_n^{\mathcal{S}}), \ \theta \in \mathcal{S} \end{bmatrix}, \ \forall n \neq \theta, \text{ given } \Theta = \theta. \tag{3.12}$$

Given $\Theta = \theta$, note that the user can retrieve the desired message W_θ by cancelling $W_n(z_n^\theta)$ in A_0 from $A_n, n \neq \theta$, and collecting the remaining pieces from A_θ. This scheme shows that the normalized cost tuple $(\Delta_0, \Delta) = (4/5, 2/5)$ is feasible. It will be shown later that this scheme achieves the capacity of the 4-star-graph PIR.

3.3 Generalization

In this section, we generalize the idea used in constructing the previous schemes to K-star-graph PIR.

For K-star-graph PIR, we construct $K+1$ schemes. We denote the schemes as Schemes $(K,0)$, $(K,1)$, $\cdots$, (K,K).

Scheme $(K,0)$: It is the trivial scheme in which the user downloads nothing from Server 0 and downloads the whole message W_k from Server k for all $k \in [K]$. This scheme shows the feasibility of $(\Delta_0, \Delta) = (0,1)$.

Scheme (K,t): For $t \in [K]$, the scheme is designed as follows. Let $L = \binom{K-1}{t-1} + \binom{K-2}{t-2}$. First, the user locally generates K independent uniform random permutations of $\{1,2,\cdots,L\}$. The k^{th} permutation is denoted as

$$\mathbf{z}_k = \left(z_k^{\mathcal{S}}, \mathcal{S} \in \binom{[K]\setminus\{k\}}{t-1};\ \bar{z}_k^i, i \in \left[\binom{K-2}{t-2} \right] \right). \tag{3.13}$$

Here, $\binom{\mathcal{A}}{a}$ denotes the set of all a-subset of a set $\mathcal{A}$, in lexicographic order. Take $K=5, t=4$ and $k=1$ as an example, (3.13) is

$$\mathbf{z}_1 = \left(z_1^{\{2,3,4\}}, z_1^{\{2,3,5\}}, z_1^{\{2,4,5\}}, z_1^{\{3,4,5\}}, \bar{z}_1^1, \bar{z}_1^2, \bar{z}_1^3 \right). \tag{3.14}$$

Given $\Theta = \theta$, the download from Server 0 is

$$A_0 = \left[\textstyle\sum_{k\in\mathcal{S}} W_k(z_k^{\mathcal{S}\setminus\{k\}}), \mathcal{S} \subset \binom{[K]}{t} \right] \tag{3.15}$$

which is a vector with length $\binom{K}{t}$. The download from Server θ is

$$A_\theta = \left[W_\theta(\bar{z}_\theta^i), i \in \left[\binom{K-2}{t-2}\right] \right] \tag{3.16}$$

which is a vector with length $\binom{K-2}{t-2}$. The download from Server $n \neq \theta$ is

$$A_n = \left[W_n(z_n^{\mathcal{S}}), \mathcal{S} : \theta \in \mathcal{S} \right] \tag{3.17}$$

which is a vector with length $\binom{K-2}{t-2}$. To retrieve W_θ, the user first cancels the terms $W_n(z_n^{\mathcal{S}})$ in A_0 using A_n, for $n \neq \theta$. This gives $\binom{K-1}{t-1}$ bits of W_θ. Then the user collects the remaining bits of W_θ from A_θ. Note that each message contains L bits, the download from Server 0 has $\binom{K}{t}$ bits since there are $\binom{K}{t}$ components in A_0, and the download from Server $k \in [K]$ has $\binom{K-2}{t-2}$ bits since there are $\binom{K-2}{t-2}$ components in both A_θ and A_n. This scheme shows the feasibility of $(\Delta_0, \Delta) = \left(\frac{\binom{K}{t}}{\binom{K-1}{t-1}+\binom{K-2}{t-2}}, \frac{\binom{K-2}{t-2}}{\binom{K-1}{t-1}+\binom{K-2}{t-2}} \right)$.

By a time-sharing argument, the above $K+1$ schemes $(K,0), (K,1), \cdots, (K,K)$ together show that the feasible normalized download cost region at least contains

$$\mathcal{D}_K = \left\{ (\Delta_0, \Delta): \begin{array}{c} \binom{t+1}{2}\Delta_0 + \left(\binom{K}{2} + Kt\right)\Delta \geq Kt, \\ \text{for } t \in \{1, 2, \cdots, K-1\}; \\ \Delta_0 + \Delta \geq 1. \end{array} \right\}. \tag{3.18}$$

Next we prove the privacy of the scheme. Suppose $\Theta = \theta$. First let us note that A_0 does not have θ by definition. Therefore, Server 0 learns nothing about Θ. The query sent to Server θ is $\bar{z}_\theta^i$ for $i \in [\binom{K-2}{t-2}]$ which is a uniformly distributed random subset of $\binom{K-2}{t-2}$ elements from $[L]$ by definition. Therefore, Server θ learns nothing about Θ. Finally, the query sent to Server $n \neq \theta$ is $z_n^{\mathcal{S}}$ for $\mathcal{S}$ such that $\theta \in \mathcal{S}$. This is also a uniformly distributed random

subset of $\binom{K-2}{t-2}$ elements from $[L]$. Therefore, for $n \neq \theta$, Server n learns nothing about Θ.

Chapter 4

Converse Bounds

In the previous section, we have constructed schemes for the K-star-graph PIR, which provide inner regions for the feasible normalized cost region. In this section, the goal is to show the converse, i.e., the outer regions for the feasible normalized cost region. As a main result, we will show that the inner regions and the outer regions match for $K \leq 4$. In other words, we will characterize the feasible normalized cost region for K up to 4. The following lemma serves as a preliminary. Given discrete random variables A, B, C,

$$\text{[Submodularity]} \qquad H(A,B) + H(A,C) \geq H(A) + H(A,B,C). \tag{4.1}$$

4.1 Known Bounds

In [10], a series of converse bounds are proved for K-star-graph PIR as

$$\Delta_0 + \frac{t(t+1)}{2}\Delta \geq t, \ \forall t \in [K]. \tag{4.2}$$

Let us provide the proof here for completeness. The idea is essentially the same as the proof in [10], but the proof will be useful to establish some notations that also help to prove our new bounds later. First, recall that the query sent to Server 0 is $Q_0 = \mu_0(\Theta, Z)$, which is determined by Θ, Z. By the privacy constraint $I(Q_0; \Theta) = 0$, there must be $z_1, z_2, \cdots, z_K \in \mathcal{Z}$ such that

$$\mu_0(1, z_1) = \mu_0(2, z_2) = \cdots = \mu_0(K, z_K) = q_0, \tag{4.3}$$

i.e., the same query q_0 works for all $\Theta = \theta \in [K]$. Let us denote $X_0 = \phi_0(W_1, W_2, \cdots, W_k, q_0)$, i.e., X_0 is the answer from Server 0 corresponding to this query q_0. Then, denote

$$X_k^\theta = \phi_k(W_k, \mu_k(\theta, z_\theta)), \ \forall k \in [K], \theta \in [K]. \tag{4.4}$$

We obtain the following.

1. W_θ is determined by $X_0, X_1^\theta, X_2^\theta, \cdots, X_K^\theta$, for $\theta \in [K]$.
2. $X_k^1, X_k^2, \cdots, X_k^K$ are determined by W_k.

They imply the following entropic conditions.

$$H(W_\theta \mid X_0, X_1^\theta, X_2^\theta, \cdots, X_K^\theta) = 0, \forall \theta \in [K], \tag{4.5}$$

$$H(X_k^1, X_k^2, \cdots, X_k^K \mid W_k) = 0, \forall k \in [K]. \tag{4.6}$$

To show the bounds in (4.2), we continue as follows. For any K, and $t = 1$,

$$
\begin{aligned}
& D_0 + D \\
& \geq H(X_0 \mid W_{[2:K]}) + H(X_1^1 \mid W_{[2:K]}) && (4.7) \\
& \geq H(X_0, X_1^1 \mid W_{[2:K]}) && (4.8) \\
& \overset{(4.5)}{\geq} H(W_1 \mid W_{[2:K]}) && (4.9) \\
& = L && (4.10) \\
& \implies D_0/L + D/L \geq 1 && (4.11) \\
& \implies \Delta_0 + \Delta \geq 1 && (4.12)
\end{aligned}
$$

(4.12) follows from the definition of the feasible download cost tuple in Section 2.4.3.

For $t = 2$,

$$
\begin{aligned}
& D_0 + 3D \\
& \geq H(X_0 \mid W_{[3:K]}) + H(X_1^1 \mid W_{[3:K]}) + H(X_2^1 \mid W_{[3:K]}) + H(X_2^2 \mid W_{[3:K]}) && (4.13) \\
& \geq H(X_0, X_1^1, X_2^1, X_2^2 \mid W_{[3:K]}) && (4.14) \\
& \overset{(4.5)}{\geq} H(W_1, X_0, X_2^2 \mid W_{[3:K]}) && (4.15) \\
& \overset{(4.6)}{=} H(W_1, X_0, X_1^2, X_2^2 \mid W_{[3:K]}) && (4.16) \\
& \overset{(4.5)}{\geq} H(W_1, W_2 \mid W_{[3:K]}) && (4.17) \\
& = 2L && (4.18) \\
& \implies D_0/L + 3D/L \geq 2 && (4.19) \\
& \implies \Delta_0 + 3\Delta \geq 2 && (4.20)
\end{aligned}
$$

For $t = 3$,

$$
\begin{aligned}
& D_0 + 6D \\
& \geq H(X_0 \mid W_{[3:K]}) + H(X_1^1 \mid W_{[4:K]}) + H(X_2^1 \mid W_{[4:K]}) + H(X_3^1 \mid W_{[4:K]}) \\
& \qquad + H(X_2^2 \mid W_{[4:K]}) + H(X_3^2 \mid W_{[4:K]}) + H(X_3^3 \mid W_{[4:K]}) && (4.21) \\
& \geq H(X_0, X_1^1, X_2^1, X_3^1, X_2^2, X_3^2, X_3^3 \mid W_{[4:K]}) && (4.22) \\
& \overset{(4.5)}{\geq} H(W_1, X_0, X_2^2, X_3^2, X_3^3 \mid W_{[4:K]}) && (4.23) \\
& \overset{(4.6)}{=} H(W_1, X_0, X_1^2, X_2^2, X_3^2, X_3^3 \mid W_{[4:K]}) && (4.24) \\
& \overset{(4.5)}{\geq} H(W_1, W_2, X_0, X_3^3 \mid W_{[4:K]}) && (4.25) \\
& \overset{(4.6)}{=} H(W_1, W_2, X_0, X_1^3, X_2^3, X_3^3 \mid W_{[4:K]}) && (4.26) \\
& \overset{(4.5)}{\geq} H(W_1, W_2, W_3 \mid W_{[4:K]}) && (4.27) \\
& = 3L && (4.28) \\
& \implies D_0/L + 6D/L \geq 3 && (4.29) \\
& \implies \Delta_0 + 6\Delta \geq 3 && (4.30)
\end{aligned}
$$

These three bounds, (4.12), (4.20) and (4.30) prove that the region shown in Fig. 3.1 is equal to $\mathcal{D}_3^*$.

Similarly, for $t \in [K]$,

$$
\begin{aligned}
& D_0 + \frac{t(t+1)}{2} D \\
& = D_0 + \big(t + (t-1) + \cdots + 1\big) D && (4.31) \\
& \geq H(X_0 \mid W_{[t+1,K]}) + \underbrace{H(X_1^1 \mid W_{[t+1,K]}) + H(X_2^1 \mid W_{[t+1,K]}) + \cdots + H(X_t^1 \mid W_{[t+1:K]})}_{t \text{ terms}} \\
& \qquad\qquad + \underbrace{H(X_2^2 \mid W_{[t+1:K]}) + H(X_3^2 \mid W_{[t+1:K]}) + \cdots + H(X_t^2 \mid W_{[t+1:K]})}_{t-1 \text{ terms}} \\
& \qquad\qquad + \cdots \\
& \qquad\qquad + \underbrace{H(X_t^t \mid W_{[t+1:K]})}_{1 \text{ term}} && (4.32) \\
& \geq H(X_0, X_1^1, \cdots, X_t^1,\ X_2^2, \cdots, X_t^2, \cdots, X_t^t \mid W_{[t+1:K]}) && (4.33) \\
& \overset{(4.5)}{\geq} H(W_1, X_0, X_2^2, \cdots, X_t^2, \cdots, X_t^t \mid W_{[t+1:K]}) && (4.34) \\
& \overset{(4.6)}{=} H(W_1, X_0, X_1^2, X_2^2, \cdots, X_t^2, \cdots, X_t^t \mid W_{[t+1:K]}) && (4.35) \\
& \overset{(4.5)}{\geq} H(W_1, W_2, X_0, X_3^3, \cdots, X_t^3, \cdots, X_t^t \mid W_{[t+1:K]}) && (4.36) \\
& \geq \cdots && (4.37) \\
& \overset{(4.5)}{\geq} H(W_1, W_2, \cdots, W_t \mid W_{[t+1:K]}) && (4.38) \\
& = tL && (4.39) \\
& \implies D_0/L + \frac{t(t+1)}{2} D/L \geq t && (4.40) \\
& \implies \Delta_0 + \frac{t(t+1)}{2} \Delta \geq t && (4.41)
\end{aligned}
$$

4.2 New bound: $3\Delta_0 + 14\Delta \geq 8$ for $K \geq 4$

For $K \geq 4$, we now prove the new bound, which is stated in the following theorem

Theorem 4.1. *For any feasible normalized cost tuple* (Δ_0, Δ)*, we have*

$$3\Delta_0 + 14\Delta \geq 8. \tag{4.42}$$

Note that this bound is not implied by the bounds in (4.2). Indeed, this bound requires the use of submodularity in a non-trivial way. In order to prove it, we proceed as follows. First,

$$\begin{aligned} &H(W_1) + H(X_2^2) + H(X_3^2) + H(X_4^2) + H(X_0) \\ &\geq H(W_1, X_2^2, X_3^2, X_4^2, X_0) \\ &\overset{(4.6)(4.5)}{\geq} \underbrace{H(W_1, W_2, X_3^2, X_4^2, X_0)}_{T_1}. \end{aligned} \tag{4.43}$$

Due to symmetry, we have

$$\begin{aligned} &H(W_3) + H(X_1^1) + H(X_2^1) + H(X_4^1) + H(X_0) \\ &\geq \underbrace{H(W_3, W_1, X_2^1, X_4^1, X_0)}_{T_2}, \end{aligned} \tag{4.44}$$

$$\begin{aligned} &H(W_4) + H(X_1^1) + H(X_2^1) + H(X_3^1) + H(X_0) \\ &\geq \underbrace{H(W_4, W_1, X_2^1, X_3^1, X_0)}_{T_3}. \end{aligned} \tag{4.45}$$

Then, by submodularity and (4.6),

$$\begin{aligned} &T_1 + T_2 \\ &\geq \underbrace{H(W_1, X_3^2, X_2^1, X_0)}_{T_4} + \underbrace{H(W_1, W_2, W_3, X_4^1, X_4^2, X_0)}_{T_5}. \end{aligned} \tag{4.46}$$

Again by submodularity,

$$\begin{aligned} & T_3 + T_4 \\ & \geq \underbrace{H(W_1, X_2^1, X_0)}_{T_6} + \underbrace{H(W_1, W_4, X_2^1, X_3^1, X_3^2, X_0)}_{T_7}. \end{aligned} \tag{4.47}$$

Then

$$T_5 + H(X_4^4) \overset{(4.6)(4.5)}{\geq} H(W_1, W_2, W_3, W_4), \tag{4.48}$$

and by submodularity

$$\begin{aligned} & T_6 + H(W_2) + H(X_3^3) + H(X_4^3) + H(X_4^4) \\ & \geq T_6 + H(W_2, X_3^3, X_4^3, X_4^4) \\ & \overset{(4.5)}{\geq} H(X_2^1) + H(W_1, W_2, X_3^3, X_4^3, X_4^4, X_0) \\ & \overset{(4.6)(4.5)}{\geq} H(X_2^1) + H(W_1, W_2, W_3, W_4), \end{aligned} \tag{4.49}$$

and finally

$$\begin{aligned} & T_7 + H(X_2^2) + H(X_3^3) \\ & \geq H(W_1, W_4, X_2^1, X_3^1, X_3^2, X_2^2, X_3^3, X_0) \\ & \overset{(4.6)(4.5)}{\geq} H(W_1, W_2, W_3, W_4). \end{aligned} \tag{4.50}$$

Adding (4.43), (4.44), (4.45), (4.46), (4.47), (4.48), (4.49), (4.50) (two terms of $H(X_2^1)$ are

canceled from both sides), we have

$$\begin{aligned}
&2H(X_1^1) + 2H(X_2^2) + 2H(X_3^3) + 2H(X_4^4) \\
&+ H(X_2^1) + H(X_3^1) + H(X_4^1) + H(X_3^2) + H(X_4^2) + H(X_4^3) \\
&+ 3H(X_0) + H(W_1) + H(W_2) + H(W_3) + H(W_4) \\
&\geq 3H(W_1, W_2, W_3, W_4). \qquad (4.51)
\end{aligned}$$

Since $D_0 \geq H(X_0)$ and $D \geq H(X_k^\theta)$ for all $k, \theta \in [K]$, we conclude that

$$14D + 3D_0 + 4L \geq 12L \implies 3\Delta_0 + 14\Delta \geq 8. \qquad (4.52)$$

Finally, let us note that although the argument holds for $K = 4$, the bound is also true for $K > 4$. This can be seen by rewriting all the entropic terms in the proof conditioned on $H(W_{[5:K]})$.

4.3 Feasible Normalized Cost Region $\mathcal{D}_4^*$

Together with the coding schemes presented in Section 3.2, we obtain that the feasible normalized cost region $\mathcal{D}_4^*$ as illustrated in Fig. 4.1. The capacity C_4 for the $K = 4$-star-

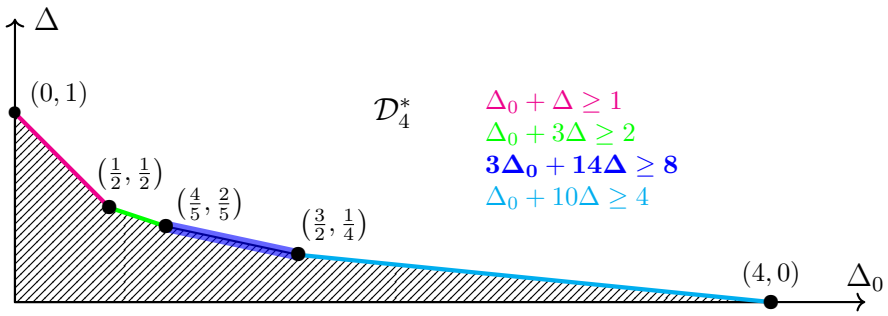

Figure 4.1: Feasible normalized cost region $\mathcal{D}_4^*$. The feasibility is a direct result of the schemes constructed in Section 3.2 and time-sharing. The converse result is proved in Section 4.1 and Section 4.2. The section in thick blue highlights our new bound.

graph PIR is then the optimization

$$C_4 = \Big(\min_{(\Delta_0,\Delta)\in\mathcal{D}_4^*} \Delta_0 + K\Delta \Big)^{-1} = \frac{5}{12}, \tag{4.53}$$

achieved by $(\Delta_0, \Delta) = \left(\frac{4}{5}, \frac{2}{5}\right)$. It is noteworthy that with only the previously best known bounds provided in Section 4.1, the best upper bound for the capacity is $C_4 \leq 3/7$. With our new bound $3\Delta_0 + 14\Delta \geq 8$, the upper bound is improved to $C_4 \leq 5/12$, thus successfully characterizing the capacity, as well as the complete feasible region of the 4-star-graph PIR.

4.4 On the Capacity C_5

We have characterized the capacities of K-star-graph PIR for K up to 4. Now, let us see how close we can bound the capacity C_5 for 5-star-graph PIR with the previous result only. For the direct part, the general scheme in Section 3.3 implies that $C_5 \geq 9/25$, achieved by $(\Delta_0, \Delta) = (10/9, 1/3)$. For the converse part, the new bound together with the previously-known bounds imply that $C_5 \leq 4/11$. Specifically,

$$\Delta_0 + 10\Delta \geq 4,\ 3\Delta_0 + 14\Delta \geq 8 \implies \Delta_0 + 5\Delta \geq \frac{11}{4}. \tag{4.54}$$

Therefore, we have $9/25 \leq C_5 \leq 4/11$. The gap between the lower and upper bounds is $4/11 - 9/25 = 1/275 \approx 0.0037$.

Our next result further reduces this gap. In the following, let us prove another non-trivial converse bound, which is stated in the following theorem.

Theorem 4.2. *For $K \geq 5$, any feasible normalized cost tuple (Δ_0, Δ) must satisfy*

$$3\Delta_0 + 21\Delta \geq 10. \tag{4.55}$$

Let us start from

$$\begin{aligned}
&H(W_1) + H(W_2) + H(X_3^3) + H(X_4^3) + H(X_5^3) + H(X_0) \\
&\geq H(W_1, W_2, X_3^3, X_4^3, X_5^3, X_0) \\
&\overset{(4.6)(4.5)}{\geq} \underbrace{H(W_1, W_2, W_3, X_4^3, X_5^3, X_0)}_{T_1}. \qquad (4.56)
\end{aligned}$$

Also,

$$\begin{aligned}
&H(W_3) + H(X_1^4) + H(X_2^4) + H(X_4^4) + H(X_5^4) + H(X_0) \\
&\geq H(W_3, X_1^4, X_2^4, X_4^4, X_5^4, X_0) \\
&\overset{(4.6)(4.5)}{\geq} \underbrace{H(W_3, W_4, X_1^4, X_2^4, X_5^4, X_0)}_{T_2}, \qquad (4.57)
\end{aligned}$$

and similarly,

$$\begin{aligned}
&H(W_5) + H(X_1^3) + H(X_2^3) + H(X_3^3) + H(X_4^3) + H(X_0) \\
&\geq H(W_5, X_1^3, X_2^3, X_3^3, X_4^3, X_0) \\
&\overset{(4.6)(4.5)}{\geq} \underbrace{H(W_3, W_5, X_1^3, X_2^3, X_4^3, X_0)}_{T_3}. \qquad (4.58)
\end{aligned}$$

Adding (4.56) and (4.57), by submodularity and (4.6), we have

$$\begin{aligned}
&T_1 + T_2 \\
&\geq \underbrace{H(W_3, X_4^3, X_1^4, X_2^4, X_0)}_{T_4} + \underbrace{H(W_1, W_2, W_3, W_4, X_5^3, X_5^4, X_0)}_{T_5} \qquad (4.59)
\end{aligned}$$

Again by submodularity,

$$\begin{aligned}
&T_3 + T_4 \\
&\geq \underbrace{H(W_3, X_4^3, X_0)}_{T_6} + \underbrace{H(W_3, W_5, X_1^4, X_2^4, X_0)}_{T_7}. \qquad (4.60)
\end{aligned}$$

Then

$$\begin{aligned}
&T_5 + H(X_5^5) \\
&\geq H(W_1, W_2, W_3, W_4, X_5^5, X_0) \\
&\overset{(4.6)(4.5)}{\geq} H(W_1, W_2, W_3, W_4, W_5), \qquad (4.61)
\end{aligned}$$

and by submodularity

$$\begin{aligned}
&T_6 + H(W_4) + H(X_1^1) + H(X_2^1) + H(X_5^1) + H(X_2^2) + H(X_5^2) + H(X_5^5) \\
&\geq T_6 + H(W_4, X_1^1, X_2^1, X_5^1, X_2^2, X_5^2, X_5^5) \\
&\overset{(4.6)}{\geq} H(X_4^3) + H(W_3, W_4, X_1^1, X_2^1, X_5^1, X_2^2, X_5^2, X_5^5) \\
&\overset{(4.6)(4.5)}{\geq} H(X_4^3) + H(W_1, W_2, W_3, W_4, W_5), \qquad (4.62)
\end{aligned}$$

and finally

$$\begin{aligned}
&T_7 + H(X_4^4) + H(X_1^1) + H(X_2^1) + H(X_2^2) \\
&\geq H(W_3, W_5, X_1^4, X_2^4, X_4^4, X_1^1, X_2^1, X_2^2, X_0) \\
&\overset{(4.6)(4.5)}{\geq} H(W_1, W_2, W_3, W_4, W_5). \qquad (4.63)
\end{aligned}$$

Adding (4.56), (4.57), (4.58), (4.59), (4.60), (4.61), (4.62) and (4.63) (two terms of $H(X_4^3)$

are canceled from both sides), we have

$$
\begin{aligned}
&2H(X_1^1) + 2H(X_2^2) + 2H(X_3^3) + 2H(X_4^4) + 2H(X_5^5) \\
&+ H(X_1^3) + H(X_1^4) + 2H(X_2^1) + H(X_2^3) + H(X_2^4) + H(X_4^3) \\
&+ H(X_5^1) + H(X_5^2) + H(X_5^3) + H(X_5^4) \\
&+ H(W_1) + H(W_2) + H(W_3) + H(W_4) + H(W_5) \\
&+ 3H(X_0) \qquad (4.64) \\
&\geq 3H(W_1, W_2, W_3, W_4, W_5). \qquad (4.65)
\end{aligned}
$$

Since $D_0 \geq H(X_0)$ and $D \geq H(X_\theta^k)$ for all $k, \theta \in [K]$, we conclude that

$$
21D + 3D_0 + 5L \geq 15L \implies 3\Delta_0 + 21\Delta \geq 10. \qquad (4.66)
$$

Similarly the bound (4.55) also extends to any $K > 5$ by conditioning on $W_{[6:K]}$. Note that the two bounds

$$
3\Delta_0 + 14\Delta \geq 8,\ 3\Delta_0 + 21\Delta \geq 10 \implies \Delta_0 + 5\Delta \geq \frac{58}{21}. \qquad (4.67)
$$

Therefore, $C_5 \leq 21/58$. With this improved upper bound, the gap between our best known lower bound and upper bound becomes $21/58 - 9/25 = 3/1450 \approx 0.0021$. Closing this gap requires finding possibly stronger converse bounds and/or constructing coding schemes with higher rates.

Chapter 5

Conclusion

We studied the capacity of K-star-graph PIR, and characterized the capacity C_K for K up to 4. The case $K = 4$ is non-trivial particularly because it requires a new converse bound based on a technically involved way of applying submodularity. The capacity C_5 for 5-star-graph PIR is still open and may require further improvement of the design of the coding scheme. Generalizing the converse bounds to $K > 4$ may require deeper understanding of the structure of the coding/decoding constraints posed by the problem or it may require the use of non-Shannon inequalities [23]. Our result also includes the construction of a general coding schemes for the K-star-graph PIR. As we have mentioned, the problems of PIR are broadly insightful as they have connection to other important problems. Therefore, before we conclude, let us briefly point out two such problems that have connections to the K-star-graph PIR.

5.1 Connection with Other Problems

5.1.1 Caching Over MAC

Caching is a technique to reduce peak traffic rates by prefetching content. With coded caching and transmission, one can design schemes that deliver the desired message to the end users more efficiently [24] than the uncoded counterpart.

Let us consider the following caching problem in a multiple access scenario. Consider a setting with K distributed servers, so that the k^{th} server has a message W_k for $k \in [K]$. For simplicity, say each message contains L bits. A user is connected to the servers through a network. Suppose with each use of the network, the user is able to get one bit from each of the servers separately. In the off-peak hours, the user can prefetch up to M bits of data from the servers. In the peak hours, the user demands one of the K messages, and uses the network D times, together with its cached data to recover its desired message. We are interested in finding the optimal trade-off between the normalized cache size $\mu = M/L$ and the average download cost for the peak hours, $\Delta = D/L$.

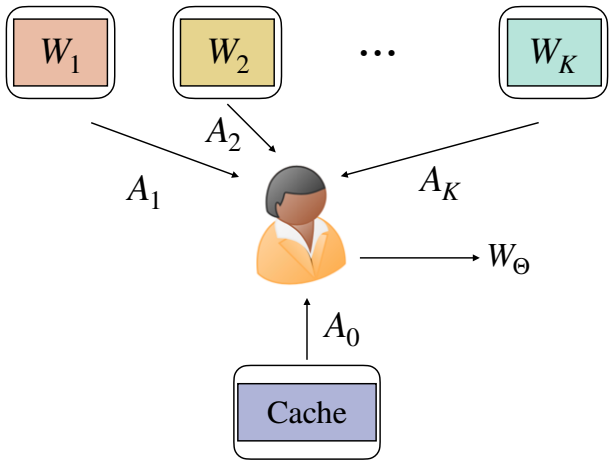

Figure 5.1: Caching Over MAC

Since the user is unaware of the index of the desired message in the off-peak hours, the cache content must be independent of this index, which is similar to the setting of the K-star-graph PIR, where the server A_0 does not learn the index of the message. This may be more transparent when we look at the the PIR schemes proposed in Chapter 3, in which the transmission from the universal server, i.e., A_0, is always determined without knowing the message index Θ. Therefore, the scheme can be applied to the caching setting as well. Specifically, let A_0 be the prefetched data in the off-peak hours. Then, in the peak hours, if the user desires W_θ, then it downloads A_k^θ from Server k. Note that our schemes has balanced download, i.e., $A_1, A_2, \cdots, A_K$ are of the same length. Therefore, the number of uses of the network is equal to the size of the answers A_k for any $k \in [K]$. The trade-off between μ and Δ in the caching problem is exactly the trade-off between Δ_0 and Δ in the K-star-graph PIR problem. Therefore, the study of the feasible region $\mathcal{D}_K^*$ for the K-star-graph PIR is also the study of the memory-download trade-off in the aforementioned caching setting.

5.1.2 Retrospective Interference Alignment

The K-star-graph PIR scheme also resembles a type of retrospective interference alignment schemes in the study of the DoF of the wireless interference network. Specifically, consider the following wireless network, with K transmitters, Tx-1–Tx-K and K receivers, Rx-1–Rx-K. Each transmitter/receiver is equipped with one antenna. For $k \in [K]$, Rx-k desires a message W_k which is known by Tx-k.

Let $n \in \mathbb{N}$. For the n^{th} use of the wireless channel, denote the input at Tx-k as $X_k(n)$ and

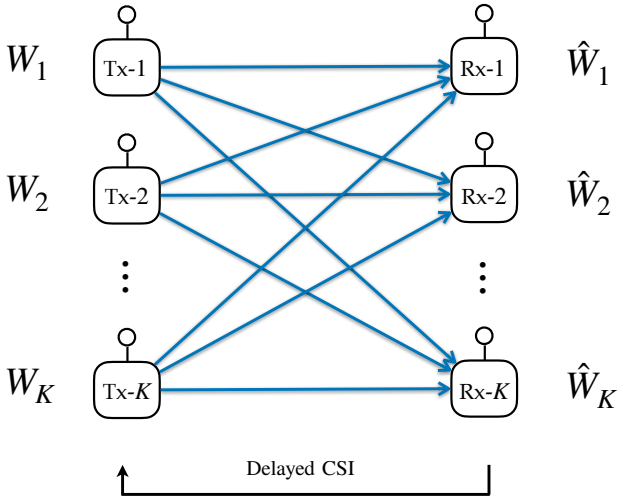

Figure 5.2: Retrospective Interference Alignment

the output at Rx-k as $Y_k(n)$. Then

$$\underbrace{\begin{bmatrix} Y_1(n) \\ Y_2(n) \\ \vdots \\ Y_K(n) \end{bmatrix}}_{\mathbf{Y}(n)} = \underbrace{\begin{bmatrix} h_{11}(n) & h_{12}(n) & \cdots & h_{1K}(n) \\ h_{21}(n) & h_{22}(n) & \cdots & h_{2K}(n) \\ \vdots & \vdots & \ddots & \vdots \\ h_{K1}(n) & h_{K2}(n) & \cdots & h_{KK}(n) \end{bmatrix}}_{\text{Channel Matrix } \mathbf{H}(n)} \underbrace{\begin{bmatrix} X_1(n) \\ X_2(n) \\ \vdots \\ X_K(n) \end{bmatrix}}_{\mathbf{X}(n)} + \underbrace{\begin{bmatrix} Z_1(n) \\ Z_2(n) \\ \vdots \\ Z_K(n) \end{bmatrix}}_{\text{i.i.d. noise } \mathbf{Z}(n)} . \tag{5.1}$$

We assume that the channel matrix $\mathbf{H}(n)$ is available at the receivers at and after the n^{th} channel uses, but is only available at the transmitters not earlier than the $(n+1)^{th}$ channel use, because it may take the time equivalent to one channel use for the feedback of the channel state information. This restriction poses constraint on the design of the possible coding schemes. For example, the K-user interference alignment scheme which achieves $K/2$ DoF as shown in [25] requires perfect channel state information at the transmitter, thus not applicable to this setting. Schemes that works in this setting are considered in e.g., [26].

A possible construction of coding schemes in this wireless channel contains two phases. Each phase contains multiple uses of the wireless channel. In the first phase, the transmitters send

their message symbols simultaneously, therefore each receiver will receive some noisy version of linear combinations of the message symbols. Let us refer to these linear combinations as the side information at the receivers. In the second phase, each channel is only occupied by one transmitter. Since in the second phase the channel state information of the first phase is fed back to the transmitters, the transmitters may use this information to design beamforming vectors in a way that it exploits the side information at the receivers. From each receiver's respective, the symbols it receives in the first phase (side information) has a similar structure to A_0, i.e., the download from the universal server of the PIR problem, and the symbols it receives in the second phase has a similar structure to $A_k, k \in [K]$, i.e., the download from the dedicated servers. Based on a similar idea, the retrospective interference alignment scheme of [26] achieves a total DoF of $\Theta(\sqrt{K})$, which is $\Theta(1/\sqrt{K})$ DoF per Tx-Rx pair. Meanwhile, recall that [10] shows the capacity of the K-star-graph PIR being also $\Theta(1/\sqrt{K})$. The two results together indicate a deeper connection between these problems.

Bibliography

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

[2] H. Sun and S. A. Jafar, "The Capacity of Private Information Retrieval," *IEEE Transactions on Information Theory*, vol. 63, no. 7, pp. 4075–4088, 2017.

[3] ——, "The capacity of robust private information retrieval with colluding databases," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 2361–2370, 2017.

[4] ——, "The capacity of symmetric private information retrieval," *IEEE Transactions on Information Theory*, vol. 65, no. 1, pp. 322–329, 2018.

[5] R. Tajeddine, O. W. Gnilke, and S. El Rouayheb, "Private information retrieval from mds coded data in distributed storage systems," *IEEE Transactions on Information Theory*, vol. 64, no. 11, pp. 7081–7093, 2018.

[6] Z. Jia, H. Sun, and S. A. Jafar, "Cross Subspace Alignment and the Asymptotic Capacity of X -Secure T -Private Information Retrieval," *IEEE Transactions on Information Theory*, vol. 65, no. 9, pp. 5783–5798, 2019.

[7] N. Raviv, I. Tamo, and E. Yaakobi, "Private Information Retrieval in Graph-Based Replication Systems," *IEEE Transactions on Information Theory*, vol. 66, no. 6, pp. 3590–3602, 2020.

[8] K. Banawan and S. Ulukus, "Private information retrieval from non-replicated databases," *International Symposium on Information Theory (ISIT)*, July 2019.

[9] Z. Jia and S. A. Jafar, "On the Asymptotic Capacity of X-Secure T-Private Information Retrieval With Graph-Based Replicated Storage," *IEEE Transactions on Information Theory*, vol. 66, no. 10, pp. 6280–6296, 2020.

[10] B. Sadeh, Y. Gu, and I. Tamo, "Bounds on the Capacity of PIR over Graphs," 2021. [Online]. Available: https://arxiv.org/abs/2105.07704

[11] K. Banawan and S. Ulukus, "The capacity of private information retrieval from byzantine and colluding databases," *IEEE Transactions on Information Theory*, vol. 65, no. 2, pp. 1206–1219, 2018.

[12] R. Tajeddine, O. W. Gnilke, D. Karpuk, R. Freij-Hollanti, and C. Hollanti, "Private information retrieval from coded storage systems with colluding, byzantine, and unresponsive servers," *IEEE Transactions on information theory*, vol. 65, no. 6, pp. 3898–3906, 2019.

[13] Z. Jia and S. A. Jafar, "X-secure t-private information retrieval from mds coded storage with byzantine and unresponsive servers," *IEEE Transactions on Information Theory*, vol. 66, no. 12, pp. 7427–7438, 2020.

[14] S. Song and M. Hayashi, "Capacity of quantum private information retrieval with multiple servers," *IEEE Transactions on Information Theory*, vol. 67, no. 1, pp. 452–463, 2020.

[15] ——, "Capacity of quantum private information retrieval with colluding servers," *IEEE Transactions on Information Theory*, vol. 67, no. 8, pp. 5491–5508, 2021.

[16] ——, "Capacity of quantum symmetric private information retrieval with collusion of all but one of servers," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 1, pp. 380–390, 2021.

[17] M. Allaix, L. Holzbaur, T. Pllaha, and C. Hollanti, "High-rate quantum private information retrieval with weakly self-dual star product codes," in *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 1046–1051.

[18] C. Tian, H. Sun, and J. Chen, "Capacity-achieving private information retrieval codes with optimal message size and upload cost," *IEEE Transactions on Information Theory*, vol. 65, no. 11, pp. 7613–7627, 2019.

[19] Y. Lu and S. A. Jafar, "On single server private information retrieval with private coded side information," *IEEE Transactions on Information Theory*, 2023.

[20] S. Ulukus, S. Avestimehr, M. Gastpar, S. A. Jafar, R. Tandon, and C. Tian, "Private Retrieval, Computing, and Learning: Recent Progress and Future Challenges," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 3, pp. 729–748, 2022.

[21] M. Effros, S. El Rouayheb, and M. Langberg, "An Equivalence Between Network Coding and Index Coding," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2478–2487, 2015.

[22] H. Sun and S. A. Jafar, "Optimal Download Cost of Private Information Retrieval for Arbitrary Message Length," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 2920–2932, 2017.

[23] Z. Zhang and R. W. Yeung, "On characterization of entropy function via information inequalities," *IEEE Transactions on Information Theory*, vol. 44, no. 4, pp. 1440–1452, 1998.

[24] M. Maddah-Ali and U. Niesen, "Fundamental Limits of Caching," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[25] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the k-user interference channel," *IEEE transactions on information theory*, vol. 54, no. 8, pp. 3425–3441, 2008.

[26] D. Castanheira, A. Silva, and A. Gameiro, "Retrospective Interference Alignment for the K -User $M \times N$ MIMO Interference Channel," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8368–8379, 2016.