DOWNLOAD COST OF CACHE-AIDED PRIVATE UPDATING WITH UNKNOWN PREFETCHING

by

Bryttany Herren

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Electrical Engineering

Charlotte

2022

Approved by:

____

Dr. Ahmed Arafa

____

Dr. Asis Nasipuri

____

Dr. Andrew Willis

ABSTRACT

BRYTTANY HERREN. Download cost of cache-aided private updating with unknown prefetching. (Under the direction of DR. AHMED ARAFA)

We consider the problem of privately updating a message out of K messages from N replicated and non-colluding databases. In this problem, a user has an outdated version of the message $\hat{W}_\theta$ of length L bits that differ from the current version W_θ in at most f bits. In addition, the user also has access to a cache Z containing linear combinations of each of the K messages, the realizations of which are unknown to the N databases (unknown prefetching). The cache Z contains ℓ linear combinations from each of the K messages in the databases, and we say that $r = \frac{\ell}{L}$ is the caching ratio. The user needs to retrieve W_θ correctly using a private information retrieval (PIR) scheme with the least number of downloads without leaking any information about the message index θ to any individual database. To that end, we propose a novel achievable scheme based on *syndrome decoding*. Specifically, the user downloads the syndrome corresponding to W_θ, according to a linear block code with carefully designed parameters, using an optimal PIR scheme for messages with a length constraint. For this scheme, the cached linear combinations in Z are chosen to be bits pertaining to the syndrome of each message in the database. We derive a lower bound on the optimal download cost for general $0 \leq r \leq 1$, and upper bounds on the optimal download cost for when r is exceptionally low or high. In particular, when deriving our upper bounds, we develop novel *cache-aided arbitrary message length* PIR schemes. Our bounds match if the term $\log_2\left(\sum_{i=0}^{f}\binom{L}{i}\right)$ is an integer. Our results imply that there is a significant reduction in the download cost if $f < \frac{L}{2}$ compared with downloading W_θ directly using cached-aided PIR approaches without taking the correlation between W_θ and $\hat{W}_\theta$ into consideration.

ACKNOWLEDGEMENTS

Throughout the writing of this thesis I have received a great deal of support and assistance.

I would first like to thank my advisor, Dr. Ahmed Arafa, who originally developed the main idea for this thesis. His expertise and insight has been invaluable in our investigation of this research problem, and during our time together he pushed me to be the best academic that I could be.

I would also like to thank Dr. Karim Banawan from Alexandria University in Egypt. His published work within as well as his general knowledge of the field of private information retrieval has served as an instrumental foundation for the techniques used in this thesis.

I would like to thank the other members of my advisory committee, Dr. Asis Nasipuri and Dr. Andrew Willis. Both have been extremely accommodating and helpful while working on this thesis.

This work was supported, in part, by funds provided by The University of North Carolina at Charlotte.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

PIR	private information retrieval

CHAPTER 1: INTRODUCTION

The problem of private information retrieval (PIR), introduced by Chor et al. in [1], seeks to find the most efficient way for a user to privately retrieve a single message from a set of K messages from N fully replicated and non-communicating databases. PIR schemes are designed to download a *mixture* of all K messages, with the least number of overhead downloaded bits, such that no single database can infer the identity of the desired message. The user accomplishes this task by sending a query to each database. The databases respond truthfully to the submitted query with an answer string. The user can then reconstruct the desired message from jointly *decoding* the returned answer strings. Recently, the problem of PIR has received a growing interest from the information and coding theory communities. The classical PIR problem is re-formulated using information-theoretic measures in the seminal work of Sun-Jafar [2]. In there, the performance metric of the PIR scheme is the retrieval rate, which is the ratio of the number of the desired message symbols to the total number of downloaded bits. The supremum of this ratio is denoted by the PIR capacity, C. Sun and Jafar characterize the PIR capacity of the classical PIR model to be

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

Following [2], the capacity (or its reciprocal, the normalized download cost) of many variations of the problem have been investigated, see, e.g., [3–17].

In all these works, the user is assumed to have no information about the desired message prior to retrieval. Thus, the queries are designed independently from the message contents. This is not always the case in practice. To see that, consider the following classical motivational example of PIR: in the stock market, investors need to privately retrieve some of the stock records, since showing interest in a specific record may undesirably affect its value. PIR is a natural solution to this problem. Now, consider the case when an investor has already retrieved a specific stock record some time ago but this record has been changed. The investor needs to update the record at his/her side. A trivial solution to this problem is to re-apply the original PIR scheme again. Nevertheless, this solution overlooks the fact that stock records are *correlated* in time. Another example arises in the context of private federated submodel learning [18], in which a user needs to retrieve the up-to-date desired submodel without leaking any information about its identity. The weights of each submodel are usually correlated in time as in the stock market example. In both examples, it is interesting to investigate whether or not the investor (user) can exploit the correlation between the outdated record (submodel) and its up-to-date counterpart to drive down the download cost. In this work, we focus our attention on a specific type of correlation, in which the up-to-date message is a distorted version of the outdated message according to a *Hamming distortion* measure. The most closely related works to this problem are the PIR problems with side information, e.g., [19–25]. We also assume that the user has access to a private local cache containing equal portions of each message. Caching systems of this variety have been explored before in the PIR setting, e.g., [26,27], but not in conjunction with other forms of side information (outdated or updated). In the works regarding PIR with side information, the user has side information in the form of a subset of *undesired* messages, which are utilized to assist in privately retrieving the desired message. This is different from our setting, in which the user possesses side information in the form of an outdated *desired* message. Furthermore, these works differ from each other in whether the privacy of the side information should be maintained or not. This is different from our problem in which the identity of the desired and side information is the same, and therefore the privacy constraint in our problem is modified to reflect this fact.

In this thesis, we introduce the problem of *cache-aided private updating with unknown prefetching* for a message out of a K-message library from N replicated and non-colluding databases. In this problem, the user has an *outdated* version $\hat{W}_\theta$ of the desired message θ, and wishes to update it to its up-to-date version W_θ. Furthermore, the user has information about the *maximum* Hamming distance f between the up-to-date message and its outdated counterpart, i.e., the user possesses $\hat{W}_\theta$, which differs in *at most* f bits from the desired up-to-date message W_θ. Based on $\hat{W}_\theta$ and f, the user needs to design a query set to reliably and privately decode the up-to-date version of the desired message W_θ with the least number of downloaded bits. Equivalently, the user needs to privately retrieve an *auxiliary* message that corresponds to the flipped bit positions in the desired message. Similar to the works of [28, 29], we assume that the databases can construct a *mapping* from the original library of messages into a more appropriate form that can assist the user in the retrieval process. The user also has access to a private cache Z containing ℓ linear combinations of each message, and we say that $r = \frac{\ell}{L}$ is the caching ratio. We aim at characterizing the optimal download cost needed to update $\hat{W}_\theta$ to W_θ given Z without disclosing the desired message index θ to any of the databases.

To that end, we propose a novel achievable scheme that is based on the *syndrome decoding* idea introduced in [30], and adapt it to our setting to exploit the correlation between W_θ and $\hat{W}_\theta$. Hence, syndrome decoding is used to *compress* the desired message based on the user's side information (i.e., the outdated message $\hat{W}_\theta$). More specifically, the databases apply a linear transformation to the stored library of messages using the parity check matrix of a linear block code with carefully chosen parameters. The existence of such a code can be readily inferred from the Gilbert-Varshamov and the Hamming bounds [31]. This transformation, in effect, maps the messages into their corresponding syndromes. Thus, the problem is reduced to retrieving the auxiliary messages (i.e., the syndrome representation) that comprises of $\lceil \bar{L} \rceil = \left\lceil \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right) \right\rceil \leq L$ bits, where L is the original message length.

In the case of $r = 0$, this enables us to directly apply the PIR scheme in [32] to the auxiliary messages of length $\lceil \bar{L} \rceil$, which is optimal under message length constraints. In the case where r satisfies $0 < r \leq \frac{1}{1+N+N^2+\cdots+N^{K-1}}$ (denoted very low r) or $\frac{1}{1+N} \leq r \leq 1$ (denoted very high r), we extend the PIR scheme in [32] to the cache-aided setting in [27], and develop a novel *cache-aided arbitrary message length* PIR scheme to solve our problem. Like with the $r = 0$ case, we can then use this new cache-aided arbitrary message length scheme to download the auxiliary messages of length $\lceil \bar{L} \rceil$ with an effective caching ratio of $\tilde{r} = \frac{\ell}{\lceil \bar{L} \rceil}$. For each of these cases, we confirm the validity of our proposed scheme by deriving a matching converse proof. Our converse proof is inspired by the converse proof of the cache-aided PIR problem with unknown and uncoded prefetching in [27], with the main difference being the fact that in addition to a private cache, the user has access to the outdated message $\hat{W}_\theta$, the index of which they wish to keep private. Consequently, we show that the optimal download cost, $\bar{D}_L$, is bounded by $\left\lceil (\bar{L} - Lr) \sum_{j=0}^{K-1} \frac{1}{N^j} - Lr \sum_{j=0}^{K-2} \frac{K-1-j}{N^j} \right\rceil \leq \bar{D}_L \leq \left\lceil \left(\lceil \bar{L} \rceil - Lr \right) \sum_{j=0}^{K-1} \frac{1}{N^j} - Lr \sum_{j=0}^{K-2} \frac{K-1-j}{N^j} \right\rceil$ when $\tilde{r}$ is very low, and that $\bar{D}_L = \lceil \bar{L} \rceil - Lr$ when $\tilde{r}$ is very high. Our achievable scheme for very low $\tilde{r}$ is optimal if $\bar{L}$ is an integer, otherwise the gap between the upper and lower bounds is upper bounded by 2 bits. This justifies the efficacy of using syndromes as a message mixing technique in our setting. Furthermore, our results show that performing direct PIR on the original library of messages is strictly sub-optimal as long as the maximum Hamming distance $f < \frac{L}{2}$.

CHAPTER 2: SYSTEM MODEL

We consider a classical PIR problem with K independent, uncoded, messages $W_1, \cdots, W_K$, with each message consisting of L independent and uniformly distributed bits. We have

$$H(W_i) = L, \quad 1 \leq i \leq K, \tag{2.1}$$

$$H(W_1, \ldots, W_K) = H(W_1) + \cdots + H(W_K). \tag{2.2}$$

The K messages are stored in N replicated and non-communicating databases. The user (retriever) has a local copy of one of the messages whose index $\theta \in [K]$ is known to the user,[1] but not the database.[2] However, this message stored locally is *outdated*, and the user wishes to update it so that it is consistent with the copies in the databases without revealing to any of the databases what the message index is.

The user also has a local cache memory whose contents is denoted by a random variable Z. The cache is populated through a *prefetching phase* in which the user randomly and independently caches ℓ bits of linear combinations from each of the up-to-date messages W_i, $i \in [K]$, with $\ell < L$. Such linear combinations are represented by a matrix multiplication $W_i R_i$, where R_i is of dimension $L \times \ell$. Thus, we have

$$Z = [W_1 R_1, \ W_2 R_2, \ \cdots, \ W_K R_K]. \tag{2.3}$$

We assume that the contents of the cache are *unknown* to the databases, as in,

[1] $[K]$ denotes the set $\{1, 2, \ldots, K\}$.

[2] This is true if message θ has been previously obtained in a private manner.

e.g., [19, 25, 27]. We define the *caching ratio* as

$$r = \frac{\ell}{L}. \tag{2.4}$$

Observe that the number of cached bits pertaining to each message is equal to Lr. It now follows that

$$H(Z) = \sum_{i=1}^{K} H(W_i R_i) \leq KLr, \tag{2.5}$$

$$I(W_i; Z) = H(W_i R_i) \leq Lr, \quad 1 \leq i \leq K. \tag{2.6}$$

The setting described above defines the *cache-aided private updating problem with unknown prefetching.*

Since each message is a string of L bits, the problem can be formulated as privately determining which subset of the message bits need to be flipped in order to fully update it. To model this, we use $\hat{W}_\theta$ to represent the locally stored outdated message, $\bar{W}_\theta$ to represent the subset of bit indices that need to be flipped, and f to represent the *maximum* Hamming distance between W_θ and $\hat{W}_\theta$.[3] Therefore, in order to update message θ the user needs to flip *at most* f bits, i.e., $\bar{W}_\theta$ takes a value out of $\sum_{i=0}^{f} \binom{L}{i}$ choices. We assume that such choices are uniformly distributed and independently realized from $\hat{W}_\theta$. Based on this model, the following holds:

$$H(W_\theta) = H(\hat{W}_\theta) = L, \tag{2.7}$$

$$H(\bar{W}_\theta) = \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right) \triangleq \bar{L}, \tag{2.8}$$

$$H(W_\theta | \hat{W}_\theta) = H(\bar{W}_\theta | \hat{W}_\theta) = \bar{L}, \tag{2.9}$$

$$H(\bar{W}_\theta | \hat{W}_\theta, W_\theta) = 0, \tag{2.10}$$

[3] Clearly, $f \geq 1$ must hold; otherwise there is not need to update $\hat{W}_\theta$.

$$|\bar{W}_\theta| \leq f \leq L, \tag{2.11}$$

where $|\cdot|$ denotes cardinality.[4] We assume that the maximum Hamming distance f between the outdated and updated message is known to the user. By (2.8), one can see that $\lceil \bar{L} \rceil$ bits should be sufficient to update $\hat{W}_\theta$. Hence, one can set a maximum value on the number of cached bits from each message as

$$\ell \leq \lceil \bar{L} \rceil . \tag{2.12}$$

In order to retrieve W_θ, the user sends a set of queries $Q_1^{[\theta]}, \ldots, Q_N^{[\theta]}$ to the N databases to efficiently obtain $\bar{W}_\theta$. The queries are generated according to $\hat{W}_\theta$, f, and Z; and are jointly independent of the realizations of the $[K]\backslash\{\theta\}$ messages and $\bar{W}_\theta$ given $\hat{W}_\theta$. Therefore we have[5]

$$I\left(W_{[K]\backslash\{\theta\}}, \bar{W}_\theta; Q_{1:N}^{[\theta]} \middle| \hat{W}_\theta, Z\right) = 0. \tag{2.13}$$

Upon receiving the query $Q_n^{[\theta]}$, the nth database replies with an answering string $A_n^{[\theta]}$, which is a function of $Q_n^{[\theta]}$ and all the K messages stored. Therefore, $\forall \theta \in [K]$, $\forall n \in [N]$, we have

$$H\left(A_n^{[\theta]} \middle| Q_n^{[\theta]}, W_{1:K}\right) = 0. \tag{2.14}$$

To ensure that individual databases do not know which message is being updated, we need to satisfy the following *privacy constraint*, $\forall n \in [N]$, $\forall k \in [K]$:

$$\left(Q_n^{[1]}, A_n^{[1]}, \hat{W}_1, W_{1:K}\right) \sim \left(Q_n^{[k]}, A_n^{[k]}, \hat{W}_k, W_{1:K}\right), \tag{2.15}$$

[4] We have a brief discussion on deriving (2.8) in Appendix A.

[5] We use the notation x_S to denote the collection of $\{x_i,\ i \in S\}$.

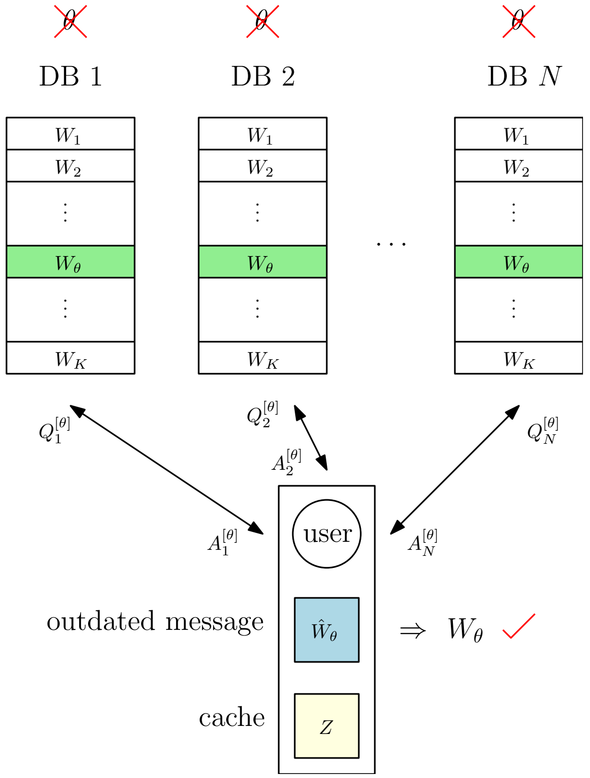

Figure 2.1: Cache-aided private updating with unknown prefetching system model.

where $\sim$ denotes statistical equivalence. After receiving the answering strings $A_{1:N}^{[\theta]}$ from all the N databases, the user needs to decode the desired information W_θ with no uncertainty, satisfying the following *correctness constraint*:

$$H\left(W_\theta \middle| A_{1:N}^{[\theta]}, Q_{1:N}^{[\theta]}, \hat{W}_\theta, Z\right) = 0. \tag{2.16}$$

The overall system model is depicted in Fig. 2.1.

For fixed N, K, f, and r, a pair $(\bar{D}, L)$ is *achievable* if there exists a cache-aided private updating with unknown prefetching scheme for messages of length L bits long satisfying the privacy constraint (2.15) and the correctness constraint (2.16). In this pair, $\bar{D}$ represents the expected number of downloaded bits received from the N

databases independently via the answering strings $A_{1:N}^{[k]}$, i.e.,

$$\bar{D} = \sum_{n=1}^{N} H\left(A_n^{[\theta]}\right). \tag{2.17}$$

Our goal is to characterize the optimal download cost $\bar{D}_L$ for the cache-aided private updating problem with unknown prefetching for fixed arbitrary L, N, K, f, and r. That is, to solve for

$$\bar{D}_L = \min\left\{\bar{D} : (\bar{D}, L) \text{ is achievable}\right\}. \tag{2.18}$$

Clearly, the user can ignore its outdated message $\hat{W}_\theta$ and re-download the whole new message W_θ using standard cache-aided PIR schemes [2, 27]. Our main result, however, shows that we can use $\hat{W}_\theta$ to do strictly better.

CHAPTER 3: MAIN RESULT

Our first result characterizes a converse bound for the optimal download cost $\bar{D}_L$ for general N, K, f, and r.

Theorem 1 (Converse) *In the cache-aided private updating problem with unknown prefetching, the optimal download cost is lower bounded by*

$$\bar{D}_L \geq \left\lceil \max_{i\in\{2,\dots,K+1\}} (\bar{L} - Lr) \sum_{j=0}^{K+1-i} \frac{1}{N^j} - Lr \sum_{j=0}^{K-i} \frac{K+1-i-j}{N^j} \right\rceil, \tag{3.1}$$

with $\bar{L}$ defined in (2.8).

The proof of Theorem 1 is provided in Chapter 4.

For our next result, we characterize an achievability bound for specific values of the caching ratios, and otherwise general L, N, K, and f. Before we present our result, we need to introduce some notation. Specifically, as in [27], for $s \in \{1, 2, \dots, K-1\}$, we define a caching ratio r_s as

$$r_s = \frac{\binom{K-2}{s-1}}{\binom{K-2}{s-1} + \sum_{i=0}^{K-1-s} \binom{K-1}{s+i} (N-1)^i N}. \tag{3.2}$$

Now, we say that a caching ratio r is *very low* if $0 \leq r \leq r_1 = \frac{1}{1+N+N^2+\cdots+N^{K-1}}$, and *very high* if $r_{K-1} = \frac{1}{1+N} \leq r \leq 1$. Our results will depend on a normalized version of r, and so we denote

$$\tilde{r} = \frac{Lr}{\lceil \bar{L} \rceil} \tag{3.3}$$

as the *effective* caching ratio. Clearly, by (2.12), $0 \leq \tilde{r} \leq 1$. We are now ready to present our achievability result.

Theorem 2 (Achievability) *In the cache-aided private updating problem with unknown prefetching, for very low* effective *caching ratios, the optimal download cost is upper bounded by*

$$\bar{D}_L \leq \left\lceil \left(\lceil \bar{L} \rceil - Lr\right) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - Lr \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil, \tag{3.4}$$

and for very high effective *caching ratios, the optimal download cost is upper bounded by*

$$\bar{D}_L \leq \lceil \bar{L} \rceil - Lr, \tag{3.5}$$

with $\bar{L}$ *defined in* (2.8)*, and the effective caching ratio defined in* (3.3)

The proof of Theorem 2 is provided in Chapter 5.

In Chapter 6, we include a discussion on extending these achievability results to effective caching ratios $\tilde{r}$ with $r_1 < \tilde{r} < r_{K-1}$.

Combining the achievability bounds in Theorem 2 with the converse bound in Theorem 1, we obtain a fairly tight characterization of the optimal download cost $\bar{D}_L$ for very low and very high effective caching ratios. This is stated in the following corollary:

Corollary 1 *In the cache-aided private updating problem with unknown prefetching, for very low caching effective ratios, we have*

$$\begin{aligned} \left\lceil (\bar{L} - Lr) \sum_{j=0}^{K-1} \frac{1}{N^j} - Lr \sum_{j=0}^{K-2} \frac{K-1-j}{N^j} \right\rceil &\leq \bar{D}_L \\ &\leq \left\lceil \left(\lceil \bar{L} \rceil - Lr\right) \sum_{j=0}^{K-1} \frac{1}{N^j} - Lr \sum_{j=0}^{K-2} \frac{K-1-j}{N^j} \right\rceil, \end{aligned} \tag{3.6}$$

and for very high caching effective ratios, we have

$$\bar{D}_L = \lceil \bar{L} \rceil - Lr \tag{3.7}$$

Proof: The right hand side inequality of (3.6) is given directly by Theorem 2. By choosing $i = 2$ in (3.1), we obtain the left hand side inequality in (3.6). Similarly, by choosing $i = K - 1$ in (3.1), we obtain the result in (3.7) (note that Lr is an integer, and so in this case the converse and achievability bounds match). This concludes the proof. ■

We conclude this chapter with some remarks.

Remark 1 *The result in Corollary 1 generalizes our preliminary work on the private updating problem with no caching involved [33]. Specifically, plugging in $r = \tilde{r} = 0$ in Corollary 1 directly gives [33, Theorem 1].*

Remark 2 *Consider the result in* (3.6)*. From* (2.8) *and* (2.11)*, it follows that $\lceil \bar{L} \rceil = L$ for all values of $f \geq \frac{L}{2}$; and that $\lceil \bar{L} \rceil < L$ for all values of $f < \frac{L}{2}$.*[1] *Combining this with the results in [27, Corollary 2] (which is the analog of our result in case the user does not have an outdated message), this means that there is a* Hamming distance threshold *of $\frac{L}{2}$ beyond which there is no advantage to using a private updating strategy, and below which there will always be some savings in download cost. This can be seen in Figure 3.1, where we also note that the non-linearity of the upper and lower bounds are a result of the ceiling functions that appear in these bounds.*

Remark 3 *If L and f are such that $\bar{L} = \lceil \bar{L} \rceil$ then the upper and lower bounds in* (3.6) *match. We will see that this holds if a* perfect code[2] *by which the queries are sent exists (cf. Chapter 5). Otherwise, if $\bar{L} < \lceil \bar{L} \rceil$, one can show using similar arguments as in [32, Section 7.2] that the two bounds are within 2 bits for $N \geq 2$ databases.*

[1] This can be readily shown using the binomial theorem. Details are in Appendix B.

[2] Perfect codes are those that attain the Hamming bound with equality [31].

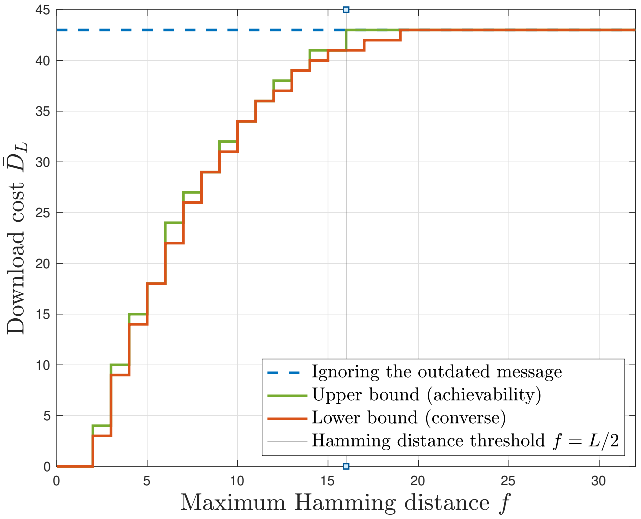

Figure 3.1: Download cost of cache-aided private updating with unknown prefetching with $L = 32$ bits, $N = 2$ databases, $K = 3$ messages, and $r = \frac{1}{10}$ caching ratio (Corrolary 1's results for the very low effective caching ratio).

CHAPTER 4: PROOF OF MAIN RESULT: CONVERSE

In this chapter, we derive the general (converse) lower bound for the download cost in Theorem 1. To do so, we prove two useful lemmas, which were previously used in the cache-aided PIR setting of [27], for the case of our cache-aided private updating problem. The two lemmas are then combined to prove the general lower bound. The key difference between our lemmas and those in [27] is that in addition to some uniform portion of each message being cached, the user is given an outdated message $\hat{W}_\theta$, requiring careful handling of the correlation between W_θ and $\hat{W}_\theta$.

Lemma 1 (Interference lower bound) *In the cache-aided private updating problem with unknown prefetching, the interference from undesired messages within the answering strings, $\bar{D} - (\bar{L} - Lr)$, satisfies*

$$\bar{D} - (\bar{L} - Lr) \geq I\left(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} \middle| W_{1:k-1}, \hat{W}_{k-1}, Z\right) \tag{4.1}$$

for all $k \in \{2, \dots, K\}$.

Proof: We start with the right hand side of (4.1),

$$\begin{aligned}
&I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}, \hat{W}_{k-1}, Z) \\
&= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}, W_{k-1} | W_{1:k-2}, \hat{W}_{k-1}, Z) - I(W_{k:K}; W_{k-1} | W_{1:k-2}, \hat{W}_{k-1}, Z) && (4.2) \\
&= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-2}, \hat{W}_{k-1}, Z) \\
&\qquad + I(W_{k:K}; W_{k-1} | Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) && (4.3) \\
&\overset{(2.16)}{=} I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-2}, \hat{W}_{k-1}, Z) && (4.4)
\end{aligned}$$

$$\overset{(2.13)}{=} I(W_{k:K}; A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) \tag{4.5}$$

$$= H(A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{4.6}$$

$$\overset{(2.16)}{=} H(A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(A_{1:N}^{[k-1]}, W_{k-1} | Q_{1:N}^{[k-1]}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{4.7}$$

$$\leq H(A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(W_{k-1} | Q_{1:N}^{[k-1]}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{4.8}$$

$$\overset{(2.13)}{=} H(A_{1:N}^{[k-1]} | Q_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(W_{k-1} | \hat{W}_{k-1}, Z) \tag{4.9}$$

$$\overset{(2.17),(2.2)}{\leq} \bar{D} - H(W_{k-1} | \hat{W}_{k-1}, W_{k-1}R_{k-1}) \tag{4.10}$$

$$= \bar{D} - \Big(H(W_{k-1}, W_{k-1}R_{k-1} | \hat{W}_{k-1}) - H(W_{k-1}R_{k-1} | \hat{W}_{k-1}) \Big) \tag{4.11}$$

$$= \bar{D} - \Big(H(W_{k-1} | \hat{W}_{k-1}) + H(W_{k-1}R_{k-1} | \hat{W}_{k-1}, W_{k-1}) - H(W_{k-1}R_{k-1} | \hat{W}_{k-1}) \Big) \tag{4.12}$$

$$\overset{(2.9),(2.6)}{\leq} \bar{D} - (\bar{L} - Lr). \tag{4.13}$$

This concludes the proof. ■

Note that if privacy was not a constraint, then $\bar{D} = \bar{L} - Lr$ and the interference from undesired messages would be non-existent. However, when the privacy constraint is present, $\bar{D} - (\bar{L} - Lr)$ characterizes the number of bits that will be downloaded and used as side information to preserve privacy from the databases in a given scheme.

Lemma 2 (Induction lemma) *For all $k \in \{2, \dots, K\}$, the mutual information term in Lemma 1 can be inductively lower bounded as*

$$I\left(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} \middle| W_{1:k-1}, \hat{W}_{k-1}, Z \right) \geq \frac{1}{N} I\left(W_{k+1:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} \middle| W_{1:k}, \hat{W}_k, Z \right) + \frac{\bar{L} - Lr}{N} - (K - k + 1)Lr. \tag{4.14}$$

Proof: We start with the left hand side of (4.14),

$$I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}, \hat{W}_{k-1}, Z)$$

$$= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}, Z, \hat{W}_{k-1} | W_{1:k-1}) - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}) \tag{4.15}$$

$$= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}) + I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}, Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}) \tag{4.16}$$

$$- I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1})$$

$$\geq I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}) - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}) \tag{4.17}$$

Now, for the first term in (4.17), we have

$$I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}) \tag{4.18}$$

$$\geq \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; Q_n^{[k-1]}, A_n^{[k-1]} | W_{1:k-1}) \tag{4.19}$$

$$\overset{(2.15)}{=} \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; Q_n^{[k]}, A_n^{[k]} | W_{1:k-1}) \tag{4.20}$$

$$\geq \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; A_n^{[k]} | W_{1:k-1}, Q_n^{[k]}) \tag{4.21}$$

$$\overset{(2.14)}{=} \frac{1}{N} \sum_{n=1}^{N} H(A_n^{[k]} | W_{1:k-1}, Q_n^{[k]}) \tag{4.22}$$

$$\geq \frac{1}{N} \sum_{n=1}^{N} H(A_n^{[k]} | W_{1:k-1}, \hat{W}_k, Z, Q_{1:N}^{[k]}, A_{1:n-1}^{[k]}) \tag{4.23}$$

$$\overset{(2.14)}{=} \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; A_n^{[k]} | W_{1:k-1}, \hat{W}_k, Z, Q_{1:N}^{[k]}, A_{1:n-1}^{[k]}) \tag{4.24}$$

$$= \frac{1}{N} I(W_{k:K}; A_{1:N}^{[k]} | W_{1:k-1}, \hat{W}_k, Z, Q_{1:N}^{[k]}) \tag{4.25}$$

$$\overset{(2.13)}{=} \frac{1}{N} I(W_{k:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k-1}, \hat{W}_k, Z) \tag{4.26}$$

$$\overset{(2.16)}{=} \frac{1}{N} I(W_{k:K}; W_k, Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k-1}, \hat{W}_k, Z) \tag{4.27}$$

$$= \frac{1}{N} I(W_{k:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k}, \hat{W}_k, Z) + \frac{1}{N} I(W_{k:K}; W_k | W_{k-1}, \hat{W}_k, Z) \tag{4.28}$$

$$= \frac{1}{N} I(W_{k:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k}, \hat{W}_k, Z) + \frac{1}{N} H(W_k | \hat{W}_k, Z) \tag{4.29}$$

$$\overset{(2.9),(2.6)}{\geq} \frac{1}{N} I(W_{k+1:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k}, \hat{W}_k, Z) + \frac{\bar{L} - Lr}{N}. \tag{4.30}$$

Note that (4.30) follows from a similar argument in Lemma 1 starting at (4.9). Next, for the second term in (4.17), we have

$$\begin{aligned} I(W_{k:K}; Z, \hat{W}_{k-1} | W_{k-1}) &= H(W_{k:K} | W_{k-1}) - H(W_{k:K} | W_{k-1}, Z, \hat{W}_{k-1}) && (4.31) \\ &= (K-k+1)L - (K-k+1)L(1-r) && (4.32) \\ &= (K-k+1)Lr && (4.33) \end{aligned}$$

Finally, we have

$$\begin{aligned} & I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{k-1}) - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{k-1}) \\ & \quad \geq \frac{1}{N} I\left(W_{k+1:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} \middle| W_{1:k}, \hat{W}_k, Z\right) + \frac{\bar{L} - Lr}{N} - (K-k+1)Lr. \end{aligned} \tag{4.34}$$

This concludes the proof. ■

We now apply the result of Lemma 2 recursively on that of Lemma 1 to get the general lower bound.

Lemma 3 *The optimal download cost of cache-aided private updating with unknown prefetching satisfies the following lower bound:*

$$\bar{D}_L \geq \left\lceil \max_{i \in \{2,\dots,K+1\}} (\bar{L} - Lr) \sum_{j=0}^{K+1-i} \frac{1}{N^j} - Lr \sum_{j=0}^{K-i} \frac{K+1-i-j}{N^j} \right\rceil \tag{4.35}$$

Proof: The download cost of any cache-aided private updating with unknown prefetching scheme satisfies the following series of inequalities:

$$\bar{D} \overset{(4.1)}{\geq} (\bar{L} - Lr) + I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}, \hat{W}_1, Z) \tag{4.36}$$

$$\overset{(4.14)}{\geq} (\bar{L} - Lr) + \frac{\bar{L} - Lr}{N} + \frac{1}{N} I(W_{k+1:K}; Q_{1:N}^{[k]}, A_{1:N}^{[k]} | W_{1:k}, \hat{W}_k, Z) - (K-1)Lr \tag{4.37}$$

$$\overset{(4.14)}{\geq} (\bar{L} - Lr) + \frac{\bar{L} - Lr}{N} + \frac{\bar{L} - Lr}{N^2} + \frac{1}{N^2} I(W_{k+2:K}; Q_{1:N}^{[k+1]}, A_{1:N}^{[k+1]} | W_{1:k+1}, \hat{W}_{k+1}, Z) - (K-1)Lr + \frac{(K-2)Lr}{N} \tag{4.38}$$

$$\overset{(4.14)}{\geq} \dots \tag{4.39}$$

$$= (\bar{L} - Lr) \sum_{j=0}^{K+1-k} \frac{1}{N^j} - Lr \sum_{j=0}^{K-k} \frac{K+1-k-j}{N^j} \tag{4.40}$$

Next, (4.40) gives K intersecting line segments, therefore, the download cost $\bar{D}$ is lower bounded by their maximum value

$$\bar{D} \geq \max_{i \in \{2,\dots,K+1\}} (\bar{L} - Lr) \sum_{j=0}^{K+1-i} \frac{1}{N^j} - Lr \sum_{j=0}^{K-i} \frac{K+1-i-j}{N^j}. \tag{4.41}$$

Since (4.41) lower bounds the download cost $\bar{D}$ for *any* cache-aided private updating with unknown prefetching scheme, it also lower bounds the download cost of the *optimal* private updating scheme $\bar{D}_L$. Finally, since $\bar{D}_L$ is an integer, we take the ceiling of (4.41) to get (4.35). ■

This concludes the converse proof.

CHAPTER 5: PROOF OF MAIN RESULT: ACHIEVABILITY

Our achievability scheme makes use of the correlation between W_θ and $\hat{W}_\theta$ through the knowledge of their maximum Hamming distance f in order to reduce the download cost. This approach is related to the problem tackled in [30] (without privacy constraints), in which a source is compressed given that it is correlated with some side information that is available only at the decoder. The retrieving user represents the decoder in our case, with side information $\hat{W}_\theta$. By the Slepian-Wolf coding theorem [34], one can noiselessly compress the source W_θ at the rate of $H(W_\theta|\hat{W}_\theta) = \bar{L}$. The *compressed* source is treated as a *new message* to be downloaded using a PIR scheme, as opposed to downloading the whole message W_θ. Such scheme, however, has a message length constraint (unlike most of the PIR works in the literature). For that reason, we leverage tools from the PIR scheme with arbitrary message length in [32], and extend them to work in the caching setting at hand, to accomplish our task.

While our achievability schemes make use of the local cache Z, we will first give some motivating examples without the user having knowledge of Z, which represents the case $r = 0$ tackled in our preliminary work [33].

5.1 Motivating Examples without Caching

5.1.1 $L = 3$, $N = 2$, $K = 2$, $f = 1$, and $r = 0$

In this example, we have $\bar{L} = \log_2(1+3) = 2$, and $C = 2/3$.[1] Setting $r = 0$ in (3.4), we need to show that $\bar{D} = \lceil \lceil \bar{L} \rceil / C \rceil = 3$ bits is achievable. We first start by constructing a $[3, 1, 3]$ linear block code, which is in this case a repetition code with

[1] $C = (1 + 1/N + \cdots + 1/N^{K-1})^{-1}$ is the classical PIR capacity [2].

generator matrix G and parity check matrix H given by

$$\mathsf{G} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \quad \mathsf{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \tag{5.1}$$

Note that such code is capable of correcting at most $f = 1$ error. The syndromes associated with this code are $\mathsf{s} \in \{00, 01, 10, 11\}$. Observe that the length of s is exactly $\lceil \bar{L} \rceil$.

Instead of requesting W_θ, the user retrieves the index of the coset in which W_θ resides in the code's standard array. That is, its corresponding syndrome

$$\mathsf{s}_\theta = W_\theta \mathsf{H}^T. \tag{5.2}$$

The user then compares $\hat{W}_\theta$ to all the words in that coset, and decodes W_θ as the one closest in Hamming distance. This is guaranteed to yield the unique correct message [30]. Therefore, the syndrome s_θ efficiently represents the flipped bits' indices $\bar{W}_\theta$, and one is able to reduce the effective message length from $L = 3$ to $\lceil \bar{L} \rceil = 2$ by dealing with the syndrome s_θ instead of W_θ.

Let $W_1 = [a_1, a_2, a_3]$, and $W_2 = [b_1, b_2, b_3]$. The syndromes (the new messages) are given by

$$\begin{aligned} \mathsf{s}_1 = W_1 \mathsf{H}^T &= \begin{bmatrix} a_1 + a_2 & a_1 + a_3 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{a}_1 & \bar{a}_2 \end{bmatrix}, \end{aligned} \tag{5.3}$$

$$\begin{aligned} \mathsf{s}_2 = W_2 \mathsf{H}^T &= \begin{bmatrix} b_1 + b_2 & b_1 + b_3 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{b}_1 & \bar{b}_2 \end{bmatrix}. \end{aligned} \tag{5.4}$$

Assume $\theta = 1$. Since $\lceil \bar{L} \rceil = N^{K-1}$, we can apply a *non-symmetric* PIR scheme as follows to decode s_1 [32]:

Database 1	Database 2
$\bar{a}_1$, $\bar{b}_1$	$\bar{a}_2 + \bar{b}_1$

This has a download cost of $\bar{D} = 3$ bits, which is optimal in this case since it meets the converse bound.

The repetition code used in this example is a *perfect code.* While this makes $\bar{L}$ an integer, and meets the converse bound, perfect codes are scarce. In the next example, we show how the proposed scheme performs with non-perfect codes.

5.1.2 $L = 5$, $N = 2$, $K = 2$, $f = 1$, and $r = 0$

In this example, we have $\bar{L} = \log_2(1 + 5) = 2.58$, and $C = 2/3$. We show that $\bar{D} = \lceil \lceil \bar{L} \rceil / C \rceil = 5$ bits is achievable. As in the previous example, we start by constructing a $[5, 2, 3]$ linear block code. Differently though, this is not a repetition code, and is characterized by

$$\mathsf{G} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}, \quad \mathsf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}. \tag{5.5}$$

The syndromes s have length $\lceil \bar{L} \rceil$. Specifically,

$$\begin{aligned} \mathsf{s}_1 = W_1 \mathsf{H}^T &= \begin{bmatrix} a_1 + a_2 + a_3 & a_1 + a_2 + a_4 & a_2 + a_5 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{a}_1 & \bar{a}_2 & \bar{a}_3 \end{bmatrix}, \end{aligned} \tag{5.6}$$

$$\begin{aligned} \mathsf{s}_2 = W_2 \mathsf{H}^T &= \begin{bmatrix} b_1 + b_2 + b_3 & b_1 + b_2 + b_4 & b_2 + b_5 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{b}_1 & \bar{b}_2 & \bar{b}_3 \end{bmatrix}. \end{aligned} \tag{5.7}$$

Since $\lceil \bar{L} \rceil = N^{K-1} + 1$, we follow the methodology in [32]; we privately download $N^{K-1} = 2$ bits ($\bar{a}_1$ and $\bar{a}_2$) using the non-symmetric PIR scheme in the previous example, and then privately download the remaining 1 bit ($\bar{a}_3$) using the scheme in [35]. The technique in [35] in this case is such that the user requests random linear combinations of $[\bar{a}_3\ \bar{b}_3]$ from database 1 using a random binary vector $\boldsymbol{h}$, and the same from database 2 yet with $\boldsymbol{h}' = \boldsymbol{h} + \boldsymbol{e}_\theta$, where $\boldsymbol{e}_i$ is the ith standard basis vector. The full PIR scheme is as follows:

Database 1	Database 2
$\bar{a}_1$, $\bar{b}_1$	$\bar{a}_2 + \bar{b}_1$
$h_1\bar{a}_3 + h_2\bar{b}_3$	$(h_1+1)\bar{a}_3 + h_2\bar{b}_3$

This has a download cost of $\bar{D} = 5$ bits, which is 1 bit away from the converse bound since the code used is non-perfect.

5.2 The General Scheme with Caching

For general L, N, K, and f, we construct an $[L, L - \lceil \bar{L} \rceil, 2f+1]$ linear block code. From the Gilbert-Varshamov bound [31], we know that such a code exists if

$$2^{\lceil \bar{L} \rceil} \leq \sum_{j=0}^{2f} \binom{L}{j}. \tag{5.8}$$

In addition, such a code must satisfy the Hamming bound [31]:

$$\sum_{j=0}^{f} \binom{L}{j} \leq 2^{\lceil \bar{L} \rceil}. \tag{5.9}$$

By the definition of $\bar{L}$ in (2.8), both (5.8) and (5.9) are satisfied, and so the code exists and is able to correct f bit flips.

Next, we map each message to its corresponding syndrome of the constructed code, which is of length $L - (L - \lceil \bar{L} \rceil) = \lceil \bar{L} \rceil$. The user then retrieves the syndrome s_θ according to a PIR scheme with N databases, K messages, and $\lceil \bar{L} \rceil$ message length. For the case $r = 0$, by [32, Theorem 1], a download cost of $\lceil \lceil \bar{L} \rceil / C \rceil$ is achievable in this case. Finally, correctness is guaranteed since querying for the syndrome s_θ

allows the user to decode W_θ as the unique word in the syndrome's coset with the least Hamming distance from $\hat{W}_\theta$ [30]. This shows that (3.4) holds specifically when $r = 0$.

For the case when $r \neq 0$, the user will have access to cached linear combinations of W_i for all $i \in [K]$. These cached linear combinations are given by $W_i R_i$, where R_i is a matrix of dimension $(L \times \lceil \bar{L} \rceil)$. For the purposes of our cache-aided achievability, we let

$$R_i = \mathsf{H}^T, \quad \forall i \in [K], \tag{5.10}$$

where H is the parity check matrix of the code. *This means that during the prefetching phase, bits from our desired syndrome are being cached,* and what is left to download is the remaining $\lceil \bar{L} \rceil - Lr$ bits.

To this end, we develop some novel schemes for cache-aided PIR with arbitrary message length that utilize the results from [27]. In particular, for all $s \in \{1, 2, \ldots, K-1\}$ we define the message length of a cache-aided PIR scheme from [27] with caching ratio r_s as

$$L_r(s) = \binom{K-2}{s-1} + \sum_{i=0}^{K-1-s} \binom{K-1}{s+i} (N-1)^i N, \tag{5.11}$$

and the normalized download cost of such a scheme as

$$D_r(s) = \frac{\sum_{i=0}^{K-1-s} \binom{K}{s+1+i} (N-1)^i N}{\binom{K-2}{s-1} + \sum_{i=0}^{K-1-s} \binom{K-1}{s+i} (N-1)^i N}. \tag{5.12}$$

For very low caching ratio r, we recall from [27] that the optimal normalized download cost of a cache-aided PIR scheme is

$$D^*(r) = (1-r) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i}, \tag{5.13}$$

and that for very high caching ratio r (in the context of this work), the optimal normalized download cost of a cache-aided PIR scheme is

$$D^*(r) = (1-r). \tag{5.14}$$

With these tools in hand, in the remainder of this chapter, we describe our achievable schemes for very low and very high caching ratios for cache-aided PIR with arbitrary message length, and show that they achieve the download costs in Theorem 2.

5.3 Very Low Caching Ratio: Proof of (3.4)

What follows is a cache-aided achievable scheme for retrieving an arbitrary L bits, for very low caching ratios ($0 < r \leq r_1 = \frac{1}{1+N+N^2+\cdots+N^{K-1}}$). We first use an optimal cache-aid PIR scheme with message size $L_r(1)$. Within the desired L bits (including the cached bits), we view each $L_r(1)$ bits as a group, and proceed until the number of desired bits remaining is strictly less than $L_r(1)$. To this end, we have

$$L = G_0 L_r(1) + L_0, \tag{5.15}$$

where $G_0 = \left\lfloor \frac{L}{L_r(1)} \right\rfloor$ and $0 \leq L_0 \leq L_r(1)-1$. If $L_0 = 0$, then the retrieval is completed. If not, then for the L_0 bits that remain, we use an optimal asymmetric PIR scheme with message size N^{K-1}. Within the remaining L_0 desired bits, we view each N^{K-1} bits as a group, and proceed until the number of desired bits remaining is strictly less than N^{K-1}. To this end, we have

$$L_0 = G_1 N^{K-1} + L_1, \tag{5.16}$$

where $G_1 = \lfloor \frac{L_0}{N^{k-1}} \rfloor$ and $0 \leq L_1 \leq N^{K-1} - 1$. If $L_1 = 0$, then the retrieval is completed. If not, then for the L_1 bits that remain, we use the scheme in [35] with N

databases and message size $N-1$. Within the remaining L_1 bits, We view each $N-1$ bits as a group, and proceed until the number of desired bits remaining is strictly less than $N-1$. To this end, we have

$$L_1 = G_2(N-1) + L_2, \tag{5.17}$$

where $G_2 = \lfloor \frac{L_1}{N-1} \rfloor$ and $0 \leq L_2 \leq N-2$. If $L_2 = 0$, then the retrieval is completed. If L_2 bits still remain, we use the scheme in [35] with $L_2 + 1$ databases and message size L_2. Therefore, the message size and the achievable download cost are

$$L = G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) + L_2, \tag{5.18}$$

$$D = \begin{cases} G_0 L_r(1) D^*(r_1) + G_1 \dfrac{N^{K-1}}{C} + G_2 N, & \text{if } L_2 = 0, \\ G_0 L_r(1) D^*(r_1) + G_1 \dfrac{N^{K-1}}{C} + G_2 N + L_2 + 1, & \text{otherwise.} \end{cases} \tag{5.19}$$

We next show that the achievable download cost in (5.19) satisfies $D \leq \lceil D^*(r) \cdot L \rceil$. To this end, we have the following lemma:

Lemma 4 *For two very low caching ratios r_a and r_b with $0 \leq r_a \leq r_b \leq r_1$, we have*

$$D^*(r_a) - D^*(r_b) = (r_b - r_a) \cdot D_c, \tag{5.20}$$

where $D_c = \sum_{i=0}^{K-1} \frac{K-i}{N^i}$.

Proof: We begin from the left hand side of (5.20) and use (5.13) to write

$$\begin{aligned} D^*(r_a) - D^*(r_b) = &\left((1-r_a) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r_a \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right) \\ &- \left((1-r_b) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r_b \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right) \quad (5.21) \\ = &\left((1-r_a) - (1-r_b)\right) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - (r_a - r_b) \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \quad (5.22) \end{aligned}$$

$$= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} + (r_b - r_a) \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \tag{5.23}$$

$$= (r_b - r_a) \cdot \left(\sum_{i=0}^{K-1} \frac{1}{N^i} + \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right) \tag{5.24}$$

$$= (r_b - r_a) \cdot \left(\sum_{i=0}^{K-1} \frac{1}{N^i} + \sum_{i=0}^{K-1} \frac{K-1-i}{N^i} \right) \tag{5.25}$$

$$= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{1 + (K-1-i)}{N^i}. \tag{5.26}$$

$$= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{K-i}{N^i}. \tag{5.27}$$

Defining $D_c = \sum_{i=0}^{K-1} \frac{K-i}{N^i}$ concludes the proof. ■

Now towards proving $D \leq \lceil D^*(r) \cdot L \rceil$, it suffices to show that $D < D^*(r) \cdot L + 1$ for two cases. For the first case, let $L_2 = 0$. We wish to show that

$$G_0 L_r(1) D^*(r_1) + G_1 \frac{N^{K-1}}{C} + G_2 N < D^*(r) \cdot \big(G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) \big) + 1. \tag{5.28}$$

First, we group the terms in (5.28); we need to show that

$$G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r) \right) - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) - G_2(N-1) D^*(r) < 1 - G_2 N. \tag{5.29}$$

Focusing on the left hand side of (5.29), we use Lemma 4 to simplify the expression. In doing this, note that $D^*(0) = \frac{1}{C}$.

$$G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r) \right) - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) - G_2(N-1) D^*(r)$$

$$= G_1 N^{K-1} D_c r - G_0 L_r(1) D_c (r_1 - r) - G_2(N-1) \left(\frac{1}{C} - D_c r \right), \tag{5.30}$$

$$= D_c \cdot \big(G_1 N^{K-1} r - G_0 L_r(1)(r_1 - r) + G_2(N-1) r \big) - G_2 \frac{N-1}{C}, \tag{5.31}$$

$$= D_c \cdot \Big(r \left(G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1)\right) - G_0 L_r(1) r_1\Big) - G_2 \frac{N-1}{C}, \tag{5.32}$$

$$= D_c \cdot (Lr - G_0 L_r(1) r_1) - G_2 \frac{N-1}{C}, \tag{5.33}$$

$$= D_c \cdot (Lr - G_0) - G_2 \frac{N-1}{C}. \tag{5.34}$$

Note that Lr is the number of cached bits, and that G_0 is the number of times a cache-aided PIR scheme is used. For very low caching ratios, these quantities are equal, and so we have

$$D_c \cdot (Lr - G_0) - G_2 \frac{N-1}{C} = -G_2 \frac{N-1}{C}. \tag{5.35}$$

Now, substituting (5.35) back into (5.29), we now need to show

$$0 < 1 - G_2 N + G_2 \frac{N-1}{C}. \tag{5.36}$$

If $N = 1$, then $G_2 = 0$, and so (5.36) clearly follows. For the case when $N \geq 2$, plugging in $C = \frac{N^{K-1}(N-1)}{N^K - 1}$ to the right hand side of (5.36) gives

$$1 - G_2 N + G_2 \frac{(N-1)(N^K-1)}{(N-1)N^{K-1}} = 1 - G_2 N + G_2 \frac{(N^K-1)}{N^{K-1}}, \tag{5.37}$$

$$= 1 - G_2 N + G_2 N - G_2 \frac{1}{N^{K-1}}, \tag{5.38}$$

$$= 1 - G_2 \frac{1}{N^{K-1}}. \tag{5.39}$$

Now, note that the maximum value of G_2 is $\left\lfloor \frac{N^{K-1}-1}{N-1} \right\rfloor$. It follows that

$$G_2 \leq \left\lfloor \frac{N^{K-1}-1}{N-1} \right\rfloor = \left\lceil \frac{N^{K-1}}{N-1} \right\rceil - 1 < \frac{N^{K-1}}{N-1} \tag{5.40}$$

Substituting (5.40) into (5.39), we have

$$1 - G_2 \frac{1}{N^{K-1}} > 1 - \frac{N^{K-1}}{N-1} \cdot \frac{1}{N^{K-1}}, \tag{5.41}$$

$$= 1 - \frac{1}{N-1}. \tag{5.42}$$

Finally, using (5.42) as a lower bound for the right hand side of (5.36), we have

$$0 \leq 1 - \frac{1}{N-1} < 1 - G_2 N + G_2 \frac{N-1}{C}, \tag{5.43}$$

and so (5.36) clearly holds for $N \geq 2$.

For the second case, let $L_2 \geq 1$. We wish to show that

$$\begin{aligned} &G_0 L_r(1) D^*(r_1) + G_1 \frac{N^{K-1}}{C} + G_2 N + L_2 + 1 \\ &\quad < D^*(r) \cdot \big(G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) + L_2\big) + 1. \end{aligned} \tag{5.44}$$

First, we group the terms in (5.44); we need to show that

$$\begin{aligned} &G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r)\right) - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) - G_2(N-1)D^*(r) - L_2 D^*(r) \\ &\quad < 1 - G_2 N - L_2 - 1. \end{aligned} \tag{5.45}$$

Focusing on the left hand side of (5.45), we use Lemma 4 to simplify the expression as follows:

$$\begin{aligned} &G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r)\right) - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) - G_2(N-1)D^*(r) - L_2 D^*(r) \\ &= G_1 N^{K-1} D_c r - G_0 L_r(1) D_c(r_1 - r) - G_2(N-1)\left(\frac{1}{C} - D_c r\right) - L_2\left(\frac{1}{C} - D_c r\right) \end{aligned} \tag{5.46}$$

$$= D_c \cdot \big(G_1 N^{K-1} r - G_0 L_r(1)(r_1 - r) + G_2(N-1)r + L_2 r\big) - G_2 \frac{N-1}{C} - \frac{L_2}{C} \tag{5.47}$$

$$= D_c \cdot \big(r\left(G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) + L_2\right) - G_0 L_r(1) r_1\big) - G_2 \frac{N-1}{C} - \frac{L_2}{C} \tag{5.48}$$

$$= D_c \cdot (Lr - G_0 L_r(1) r_1) - G_2 \frac{N-1}{C} - \frac{L_2}{C} \tag{5.49}$$

$$= D_c \cdot (Lr - G_0) - G_2 \frac{N-1}{C} - \frac{L_2}{C}. \tag{5.50}$$

Note that Lr is the number of cached bits, and that G_0 is the number of times a cache-aided PIR scheme is used. For very low caching ratios, these quantities are equal, and so we have

$$D_c \cdot (Lr - G_0) - G_2 \frac{N-1}{C} - \frac{L_2}{C} = -G_2 \frac{N-1}{C} - \frac{L_2}{C}. \tag{5.51}$$

Now, substituting (5.51) back into (5.45), we have

$$0 < 1 - G_2 N + G_2 \frac{N-1}{C} + L_2 \left(\frac{1}{C} - 1\right) - 1. \tag{5.52}$$

Since $L_2 \geq 1$, we have $N \geq 2$. Plugging in $C = \frac{N^{K-1}(N-1)}{N^K - 1}$ into the right hand side of (5.52) gives

$$\begin{aligned} &1 - G_2 N + G_2 \frac{N-1}{C} + L_2 \left(\frac{1}{C} - 1\right) - 1 \\ &\quad = -G_2 N + G_2 \frac{N^K - 1}{N^{K-1}} + L_2 \left(\frac{N^K - 1}{N^{K-1}(N-1)} - 1\right) && (5.53) \\ &\quad = -G_2 \frac{1}{N^{K-1}} + L_2 \left(\frac{N^K - 1 - N^{K-1}(N-1)}{N^{K-1}(N-1)}\right) && (5.54) \\ &\quad = -G_2 \frac{1}{N^{K-1}} + L_2 \left(\frac{N^{K-1} - 1}{N^{K-1}(N-1)}\right). && (5.55) \end{aligned}$$

We wish to find a lower bound for the right hand side of (5.52). To this end, we want to maximize G_2 and minimize L_2. We know that $L_2 \geq 1$, but this also means that $G_2(N-1) < L_1 \leq N^K - 1$ from (5.17). Plugging these values into (5.55) gives

$$
\begin{aligned}
-G_2\frac{1}{N^{K-1}} + L_2\left(\frac{N^{K-1}-1}{N^{K-1}(N-1)}\right) &\geq -\frac{G_2(N-1)}{N^{K-1}(N-1)} + \frac{N^{K-1}-1}{N^{K-1}(N-1)} && (5.56)\\
&> -\frac{N^{K-1}-1}{N^{K-1}(N-1)} + \frac{N^{K-1}-1}{N^{K-1}(N-1)} && (5.57)\\
&= 0. && (5.58)
\end{aligned}
$$

Thus, (5.52) clearly holds. This completes the proof that $D \leq \lceil D^*(r) \cdot L \rceil$ for very low caching ratios.

Since the above PIR scheme is constructed as a concatenation of several PIR schemes that are both correct and private, by [32, Theorem 4], the above scheme is both correct and private. Furthermore, since the above PIR scheme retrieves L bits (including cached bits) at a download cost of $D \leq \lceil D^*(r) \cdot L \rceil$, this scheme can used to retrieve $\lceil \bar{L} \rceil$ bits (including some Lr cached bits) at a download cost of $\bar{D} \leq \lceil D^*(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$. Expanding this statement gives

$$
\begin{aligned}
\bar{D} &\leq \lceil D^*(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil && (5.59)\\
&= \left\lceil \lceil \bar{L} \rceil (1-\tilde{r}) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - \lceil \bar{L} \rceil \tilde{r} \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil && (5.60)\\
&= \left\lceil \lceil \bar{L} \rceil (1-\frac{Lr}{\lceil \bar{L} \rceil}) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - \lceil \bar{L} \rceil \frac{Lr}{\lceil \bar{L} \rceil} \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil && (5.61)\\
&= \left\lceil (\lceil \bar{L} \rceil - Lr) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - Lr \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil, && (5.62)
\end{aligned}
$$

which is precisely (3.4).

5.4 Very High Caching Ratio: Proof of (3.5)

What follows is a cache-aided achievable scheme for retrieving an arbitrary L bits, for very high caching ratios ($r_{K-1} = \frac{1}{1+N} \leq r \leq 1$). In this scheme, we only use an optimal cache-aided PIR scheme with message size $L_r(K-1) = 1+N$. We note that in this scheme, for each bit we have cached, we can download 1 bit from each of the N databases to get a total of N unknown bits at a download cost of N bits.

Within the desired L bits (including cached bits), we view each $L_r(K-1)$ bits as a group, and proceed until the number of desired and *unknown* $L - Lr$ bits remaining is strictly less than N. To this end, we have

$$L = G_0 L_r(K-1) + L_0, \tag{5.63}$$

where $G_0 = \lfloor \frac{L-Lr}{N} \rfloor$, and $L_0 = L - G_0 L_r(K-1)$. We define $C_0 = Lr - G_0$ as the number of *unused* cached bits thus far in our scheme. If we have $L_0 = C_0$, then we have all of our desired information, and we are done. Otherwise, we still have $L_0 - C_0 < N$ bits left to download. Since the caching ratio r is very high, we have $C_0 \geq 1$, and so we can use this bit, as noted above, to download 1 bit from $L_0 - C_0 < N$ databases each to get the remaining $L_0 - C_0$ unknown bits at a download cost of $L_0 - C_0$ bits. Therefore, the message size and the achievable download cost are

$$L = G_0 L_r(K-1) + L_0, \tag{5.64}$$

$$D = G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0. \tag{5.65}$$

We next show that the achievable download cost in (5.65) satisfies $D \leq \lceil D^*(r) \cdot L \rceil$. To this end, it it suffices to show that $D < D^*(r) \cdot L + 1$, or more specifically, that

$$G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0 < D^*(r) \cdot (G_0 L_r(K-1) + L_0) + 1. \tag{5.66}$$

First, we rearrange the terms in (5.66) as

$$G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0 - D^*(r) \cdot (G_0 L_r(K-1) + L_0) < 1, \tag{5.67}$$

and then we reduce the left hand side of (5.67) as follows

$$
\begin{aligned}
&G_0L_r(K-1)D^*(r_{K-1}) + L_0 - C_0 - D^*(r)\cdot(G_0L_r(K-1)+L_0) \\
&= G_0(1+N)(1-\frac{1}{1+N}) + L_0 - C_0 - (1-r)\cdot(G_0(1+N)+L_0) && (5.68)\\
&= G_0N + L_0 - C_0 - (1-r)\cdot(G_0 + G_0N + L_0) && (5.69)\\
&= G_0N + L_0 - C_0 - (G_0 + G_0N + L_0) + r\,(G_0 + G_0N + L_0) && (5.70)\\
&= -C_0 - G_0 + r\,(G_0 + G_0N + L_0) && (5.71)\\
&= G_0 - Lr - G_0 + r\,(G_0 + G_0N + L_0) && (5.72)\\
&= -Lr + r\,(G_0(1+N) + L_0) && (5.73)\\
&= 0. && (5.74)
\end{aligned}
$$

Thus, (5.66) holds, and so this completes the proof that $D \leq \lceil D^*(r)\cdot L\rceil$ for very high caching ratios.

Again, since the above PIR scheme is constructed as a concatenation of several PIR schemes that are both correct and private, by [32, Theorem 4], the above scheme is both correct and private. Furthermore, since the above PIR scheme retrieves L bits (including cached bits) at a download cost of $D \leq \lceil D^*(r)\cdot L\rceil$, this scheme can used to retrieve $\lceil \bar{L}\rceil$ bits (including some Lr cached bits) at a download cost of $\bar{D} \leq \lceil D^*(\tilde{r})\cdot\lceil\bar{L}\rceil\rceil$. Expanding this statement gives

$$
\begin{aligned}
\bar{D} &\leq \lceil D^*(\tilde{r})\cdot\lceil\bar{L}\rceil\rceil && (5.75)\\
&= \lceil (1-\tilde{r})\cdot\lceil\bar{L}\rceil\rceil && (5.76)\\
&= \left\lceil (1-\frac{Lr}{\lceil\bar{L}\rceil})\cdot\lceil\bar{L}\rceil\right\rceil && (5.77)\\
&= \lceil\lceil\bar{L}\rceil - Lr\rceil = \lceil\bar{L}\rceil - Lr, && (5.78)
\end{aligned}
$$

which is precisely (3.5).

CHAPTER 6: DISCUSSION

As seen in Corollary 1, for very low and very high effective caching ratios, we obtain full characterizations of the optimal download cost $\bar{D}_L$ for fixed L, N, K, and f. What remains is to do the same for an effective caching ratio $\tilde{r}$ with $\frac{1}{1+N+N^2+\cdots+N^{K-1}} = r_1 \leq \tilde{r} \leq r_{K-1} = \frac{1}{1+N}$. We call such a caching ratio *mid-range*.

Our approach for our achievability results when $\tilde{r} \neq 0$ has been to describe an arbitrary message length PIR scheme for a setting with unknown prefetching, and then show that the download cost D of such a scheme satisfies $D \leq \lceil D^*(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$. This approach mirrors what was done in [32] for the classical PIR setting.

From [27], for $r_s < r < r_{s+1}$ and $\alpha \in [0, 1]$ with $r = \alpha r_s + (1-\alpha) r_{r+1}$ we define

$$\bar{D}(r) = \alpha D_r(s) + (1-\alpha) D_r(s+1). \tag{6.1}$$

We know that $\bar{D}(r) = D^*(r)$ for very low and very high caching ratio r, and this is used in our approach for Theorem 2. For when $\bar{D}(r) \neq D^*(r)$, as is the case for most mid-range caching ratios, we can still attempt to describe a scheme, and show that the download cost $D \leq \lceil \bar{D}(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$ to obtain some useful result.

Our goal in this chapter is to present some motivating examples that show what these result may look like. Future investigations of this problem setting include formulating such results in concrete theorems.

6.1 Example Set 1: $N = 2$, $K = 3$, and $r_1 \leq r \leq r_2 = r_{K-1}$

Recall that for the $N = 2$, $K = 3$ setting, we have $r_1 = \frac{1}{7}$ and $r_{K-1} = \frac{1}{3}$. With this in mind, we start with a standard cache-aided PIR scheme from [27] with $N = 2$, $K = 3$, and $r = r_1 = \frac{1}{7}$:

Database 1	Database 2
$a_2 + b_1$	$a_4 + b_1$
$a_3 + c_1$	$a_5 + c_1$
$b_2 + c_2$	$b_3 + c_3$
$a_6 + b_3 + c_3$	$a_7 + b_2 + c_2$

$$Z = \{a_1, b_1, c_1\}$$

This scheme is for a very low caching ratio, and we know that it is optimal in obtaining 7 bits of useful information (including cached bits) at a download cost of 8. By truncating the above scheme, the same query structure can be used to obtain $L_6^1 = 6$ bits of useful information at a download cost $D_6^1 = 7$:

Database 1	Database 2
$a_2 + b_1$	$a_4 + b_1$
$a_3 + c_1$	$a_5 + c_1$
$b_2 + c_2$	$b_3 + c_3$
$a_6 + b_3 + c_3$	

$$Z = \{a_1, b_1, c_1\}$$

Note that the setting of this new scheme has a caching ratio of $r = \frac{1}{6}$. Also, we have $\lceil \bar{D}(\frac{1}{6}) \cdot L_6^1 \rceil = 7$, and so $D_6^1 \leq \lceil \bar{D}(\frac{1}{6}) \cdot L_6^1 \rceil$ holds.

For another example, consider the same setting, but with caching ratio $r = \frac{1}{5}$:

Database 1	Database 2
$a_2 + b_1$	$a_4 + b_1 + c_1$
$a_3 + c_1$	$a_5 + b_2 + c_2$
$b_2 + c_2$	

$$Z = \{a_1, b_1, c_1\}$$

Here, we have $L_5^1 = 5$, and $D_5^1 = 5 = \lceil \bar{D}(\frac{1}{5}) \cdot L_5^1 \rceil$, and so $D_5^1 \leq \lceil \bar{D}(\frac{1}{5}) \cdot L_5^1 \rceil$ clearly holds. Like before, this scheme can be truncated by removing the $a_5 + b_2 + c_2$ query to obtain $L_4^1 = 4$ bits of useful information at a download cost of $D_4^1 = 4$. It can be shown that $\lceil \bar{D}(\frac{1}{4}) \cdot L_4^1 \rceil = 4$, and so $D_4^1 \leq \lceil \bar{D}(\frac{1}{4}) \cdot L_4^1 \rceil$ holds as well.

As a final example for this section, consider this same setting, but with a caching ratio $r = \frac{2}{8}$. While this may seem redundant given the previous example, we state r

in this way to highlight how in this case, there are 2 bits from each message cached. This is shown in the following scheme:

Database 1	Database 2
$a_3 + b_1 + c_1$	$a_4 + b_1 + c_1$
$a_5 + b_2$	$a_7 + b_2 + c_2$
$a_6 + c_2$	$a_8 + b_3 + c_3$
$b_3 + c_3$	

$$Z = \{a_1, a_2, b_1, b_2, c_1, c_2\}$$

Here, we obtain $L^2_8 = 8$ bits of useful information at a download cost of $D^2_8 = 7$. We have $\lceil \bar{D}(\frac{2}{8}) \cdot L^2_8 \rceil = 7$, and so $D^2_8 \leq \lceil \bar{D}(\frac{2}{8}) \cdot L^2_8 \rceil$ holds. Again, this scheme can be truncated by removing the $a_8 + b_3 + c_3$ query to obtain $L^2_7 = 7$ bits of useful information at a download cost $D^2_7 = 6$. It can be shown that $\lceil \bar{D}(\frac{2}{7}) \cdot L^2_7 \rceil = 6$, and so $D^2_7 \leq \lceil \bar{D}(\frac{2}{7}) \cdot L^2_7 \rceil$ holds as well.

What we have shown here with these examples is that for the $N = 2$, $K = 3$ setting, there appears to be a pattern where for any $i, j \in \mathbb{N}$ with $r_1 \leq \frac{i}{j} \leq r_{K-1}$, it can shown that $D^i_j \leq \left\lceil \bar{D}(\frac{i}{j}) \cdot L^i_j \right\rceil$. Hence, as a future work, these examples may lead us to show that $D^i_j \leq \left\lceil \bar{D}(\frac{i}{j}) \cdot L^i_j \right\rceil$ holds for mid-range caching ratio $r = \frac{i}{j}$ when $K = 3$ and $N \geq 1$.

In the next section, we show how this result may not hold if $K \neq 3$.

6.2 Example Set 2: $N = 3$, $K = 4$, and $r_{K-2} \leq r \leq r_{K-1}$

For the $N = 3$, $K = 4$ setting, we have $r_1 = \frac{1}{40}$ and $r_{K-1} = \frac{1}{4}$, and so a caching ratio is mid-range in this setting if $\frac{1}{40} \leq r \leq \frac{1}{4}$. However, for our purposes, we will focus on the subset of mid-range caching ratios r satisfying $r_{K-2} = \frac{2}{17} \leq r \leq \frac{1}{4}$. With this in mind, for our first example, we consider a setting with a caching ratio $r = \frac{1}{6}$. We wish to obtain $L^1_6 = 6$ bits of useful information (1 of which is cached), and to that end, we download 3 of those bits using an optimal cache-aid PIR scheme from [27]:

Now, in order to download the 2 remaining bits of useful information, we use the

Database 1	Database 2	Database 3
$a_2 + b_1 + c_1 + d_1$	$a_3 + b_1 + c_1 + d_1$	$a_4 + b_1 + c_1 + d_1$

$$Z = \{a_1, b_1, c_1, d_1\}$$

scheme in [35] to privately download these $N - 1 = 2$ bits at a download cost of $N = 3$. The result is a total download cost of $D_6^1 = 6$ for this scheme. Also, we have $\lceil \bar{D}(\frac{1}{6}) \cdot L_6^1 \rceil = 6$, and so $D_6^1 \leq \lceil \bar{D}(\frac{1}{6}) \cdot L_6^1 \rceil$ holds. Note that if we instead use the scheme in [35] to privately download 1 bit at a download cost of 2, then we have a scheme for a caching ratio of $r = \frac{1}{5}$ obtaining $L_5^1 = 5$ bits at a download cost of $D_5^1 = 5$. It can be shown that $\lceil \bar{D}(\frac{1}{5}) \cdot L_5^1 \rceil = 5$, and so $D_5^1 \leq \lceil \bar{D}(\frac{1}{5}) \cdot L_5^1 \rceil$ holds.

Next, consider the same setting, but with a caching ratio $r = \frac{1}{8}$. We wish to obtain $L_8^1 = 8$ useful bits of information (again, 1 of which is cached), and to this end, we can use the scheme described above to get 5 bits at a download cost of 6. Just as before, there are 2 remaining bits of useful information to download, and we do so using the scheme in [35] to obtain $N - 1 = 2$ bits at a download cost of $N = 3$. The result is obtaining $L_8^1 = 8$ bits of useful information at a download cost $D_8^1 = 9$. Just as with the previous examples, it can be shown that $\lceil \bar{D}(\frac{1}{8}) \cdot L_8^1 \rceil = 9$, and so $D_8^1 \leq \lceil \bar{D}(\frac{1}{8}) \cdot L_8^1 \rceil$ holds.

If this pattern continued, we would be able to truncate the above query structure to get a new scheme for a setting with caching ratio $r = \frac{1}{7}$. In fact, doing so would give a scheme obtaining $L_7^1 = 7$ bits of useful information at a download cost of $D_7^1 = 8$. However, $\lceil \bar{D}(\frac{1}{7}) \cdot L_7^1 \rceil = 7$, and so $D_7^1 \leq \lceil \bar{D}(\frac{1}{7}) \cdot L_7^1 \rceil$ *does not hold,* breaking the pattern we have witnessed up to this point. Hence, as a future work, these examples may lead us to show that $D_j^i \leq \left\lceil \bar{D}(\frac{i}{j}) \cdot L_j^i \right\rceil + 1$ for all $i, j \in \mathbb{N}$ with $r_1 \leq \frac{i}{j} \leq r_{K-1}$ for $K \geq 2$ and $N \geq 1$.

The question remains on why this is the case. That is, *why this pattern breaks, and why it is difficult to find an alternative query structure.* To help answer these questions, let us look at 2 final examples for the $N = 3$, $K = 4$ case, one with a caching ratio of $r = \frac{2}{14}$ and the other with a caching ratio of $r = \frac{3}{21}$. Again, we state r this way to highlight how 2 and 3 bits are cached from each message in each scheme, respectively:

$r = \frac{2}{14}$ scheme:

Database 1	Database 2	Database 3
$a_3 + b_1 + c_1$	$a_6 + b_1 + c_1$	$a_9 + b_1 + c_1 + d_1$
$a_4 + b_2 + d_1$	$a_7 + b_2 + d_1$	$a_{10} + b_2 + c_2 + d_2$
$a_5 + c_2 + d_2$	$a_8 + c_2 + d_2$	$a_{13} + b_3 + c_3 + d_3$
$b_3 + c_3 + d_3$	$b_4 + c_4 + d_4$	$a_{14} + b_4 + c_4 + d_4$
$a_{11} + b_4 + c_4 + d_4$	$a_{12} + b_3 + c_3 + d_3$	

$$Z = \{a_1, a_2, b_1, b_2, c_1, c_2, d_1, d_2\}$$

$r = \frac{3}{21}$ scheme:

Database 1	Database 2	Database 3
$a_4 + b_1 + c_1$	$a_7 + b_1 + c_1$	$a_{10} + b_1 + c_1$
$a_5 + b_2 + d_1$	$a_8 + b_2 + d_1$	$a_{11} + b_2 + d_1$
$a_6 + c_2 + d_2$	$a_9 + c_2 + d_2$	$a_{12} + b_2 + d_2$
$b_4 + c_4 + d_4$	$b_5 + c_5 + d_5$	$b_6 + c_6 + d_6$
$a_{13} + b_5 + c_5 + d_5$	$a_{15} + b_4 + c_4 + d_4$	$a_{17} + b_4 + c_4 + d_4$
$a_{14} + b_6 + c_6 + d_6$	$a_{16} + b_6 + c_6 + d_6$	$a_{18} + b_5 + c_5 + d_5$
$a_{19} + b_3 + c_3 + d_3$	$a_{20} + b_3 + c_3 + d_3$	$a_{21} + b_3 + c_3 + d_3$

$$Z = \{a_1, a_2, a_3, b_1, b_2, b_3, c_1, c_2, c_3, d_1, d_2, d_3\}$$

Both of these query structures satisfy $D^2_{14} \leq \lceil \bar{D}(\frac{2}{14}) \cdot L^2_{14} \rceil$ and $D^3_{21} \leq \lceil \bar{D}(\frac{3}{21}) \cdot L^3_{21} \rceil$, respectively, but have an equivalent caching ratio to the scheme with $r = \frac{1}{7}$ where $D^1_7 \leq \lceil \bar{D}(\frac{1}{7}) \cdot L^1_7 \rceil$ did not hold. This suggests that the cause of our issues regarding this $r = \frac{1}{7}$ scheme has not to do with with the value of the r, but with *the number number of cached bits Lr*. More specifically, there may be some additional limitation on how low of a download cost can be achieved with a cache-aided arbitrary message length PIR scheme when Lr is relatively low (or in this case, when $Lr = 1$). Investigating such limitations is left to future works.

CHAPTER 7: CONCLUSIONS

In this thesis, we introduced the cache-aided private updating problem with unknown prefetching, in which a user's outdated message is to be privately updated by utilizing a private cache and querying a set of replicated and non-colluding databases that have the up-to-date version. Under a Hamming distortion measure between the outdated and the up-to-date messages, a syndrome decoding technique is leveraged to compress the number of bits that needs to be downloaded in order to correctly update the message. In our preliminary work without caching, this was combined with PIR schemes with message length constraints to guarantee privacy [33]. However, in a cache-aided setting, there did not exist PIR schemes with message length constraints. In this thesis, we remedy this fact by developing novel *arbitrary message length cache-aided* PIR schemes for very low caching ratios and very high caching ratios. These schemes are then combined with syndrome decoding techniques to guarantee privacy. For very low and very high effective caching ratios, the proposed cache-aided private updating with unknown prefetching scheme has been shown to be optimal when the system parameters enable the construction of a perfect code according to which the syndrome decoding technique is worked out. In other cases, the achievable download cost has been shown to be within at most 2 bits from a derived converse bound.

In the cache-aided private updating problem with unknown prefetching, the most pertinent item that remains to be resolved is the characterization of the optimal download cost $\bar{D}_L$ for mid-range caching ratios. This is dependent entirely on finding and formalizing an arbitrary message length cache-aided PIR scheme for such caching ratios, the progress of which has been discussed thoroughly in Chapter 6. Such schemes would ideally resemble the results from [27], but with an arbitrary message length constraint. Another item that could be resolved in this problem is the inflexible nature of the cache in our achievability. Specifically, the fact that for each $i \in [K]$, we fix $R_i = \mathsf{H}^T$ during the prefetching phase. Ideally, we would be caching individual bits from each message in the databases, and if we must cache linear combinations, we would impose less control over the realization of each R_i. Resolving each of these items of interest would help to strengthen our current solutions to the cache-aided private updating problem with unknown prefetching.

REFERENCES

[1] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, “Private information retrieval,” *J. ACM*, vol. 45, p. 965â981, Nov. 1998.

[2] H. Sun and S. A. Jafar, “The capacity of private information retrieval,” *IEEE Trans. Inf. Theory*, vol. 63, pp. 4075–4088, July 2017.

[3] K. Banawan and S. Ulukus, “The capacity of private information retrieval from coded databases,” *IEEE Trans. Inf. Theory*, March 2018.

[4] H. Sun and S. A. Jafar, “The capacity of symmetric private information retrieval,” *IEEE Transactions on Information Theory*, vol. 65, pp. 322–329, January 2019.

[5] K. Banawan and S. Ulukus, “Multi-message private information retrieval: Capacity results and near-optimal schemes,” *IEEE Trans. on Info. Theory*, vol. 64, pp. 6842–6862, October 2018.

[6] R. Tajeddine, O. W. Gnilke, D. Karpuk, R. Freij-Hollanti, C. Hollanti, and S. E. Rouayheb, “Private information retrieval schemes for coded data with arbitrary collusion patterns,” in *Proc. IEEE ISIT*, June 2017.

[7] Q. Wang and M. Skoglund, “On PIR and symmetric PIR from colluding databases with adversaries and eavesdroppers,” *IEEE Trans. Inf. Theory*, vol. 65, pp. 3183–3197, May 2019.

[8] C. Tian, H. Sun, and J. Chen, “Capacity-achieving private information retrieval codes with optimal message size and upload cost,” *IEEE Trans. Inf. Theory*, vol. 65, pp. 7613–7627, November 2019.

[9] T. Guo, R. Zhou, and C. Tian, “On the information leakage in private information retrieval systems,” *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 2999–3012, March 2020.

[10] K. Banawan and S. Ulukus, “The capacity of private information retrieval from byzantine and colluding databases,” *IEEE Trans. Inf. Theory*, vol. 65, pp. 1206–1219, February 2019.

[11] M. A. Attia, D. Kumar, and R. Tandon, “The capacity of private information retrieval from uncoded storage constrained databases,” *IEEE Trans. Inf. Theory*, vol. 66, pp. 6617–6634, November 2020.

[12] H. Sun and S. A. Jafar, “The capacity of private computation,” *IEEE Trans. Inf. Theory*, vol. 65, pp. 3880–3897, June 2019.

[13] S. Kumar, A. G. i Amat, E. Rosnes, and L. Senigagliesi, “Private information retrieval from a cellular network with caching at the edge,” *IEEE Trans. Commun.*, vol. 67, pp. 4900–4912, July 2019.

[14] N. Raviv, I. Tamo, and E. Yaakobi, "Private information retrieval in graph-based replication systems," *IEEE Trans. Inf. Theory*, vol. 66, pp. 3590–3602, June 2020.

[15] X. Yao, N. Liu, and W. Kang, "The capacity of multi-round private information retrieval from Byzantine databases," in *Proc. IEEE ISIT*, July 2019.

[16] I. Samy, R. Tandon, and L. Lazos, "On the capacity of leaky private information retrieval," in *Proc. IEEE ISIT*, July 2019.

[17] R. G. L. DâOliveira and S. El Rouayheb, "One-shot PIR: Refinement and lifting," *IEEE Trans. Inf. Theory*, vol. 66, pp. 2443–2455, April 2020.

[18] Z. Jia and S. Jafar, "X-secure T-private federated submodel learning," [Online]. Available: arXiv:2010.01059.

[19] Z. Chen, Z. Wang, and S. A. Jafar, "The capacity of T-private information retrieval with private side information," *IEEE Trans. Inf. Theory*, vol. 66, pp. 4761–4773, August 2020.

[20] Y.-P. Wei, K. Banawan, and S. Ulukus, "The capacity of private information retrieval with partially known private side information," *IEEE Trans. Inf. Theory*, vol. 65, pp. 8222–8231, December 2019.

[21] Y.-P. Wei and S. Ulukus, "The capacity of private information retrieval with private side information under storage constraints," *IEEE Trans. Inf. Theory*, 2019. Early Access.

[22] S. P. Shariatpanahi, M. J. Siavoshani, and M. A. Maddah-Ali, "Multi-message private information retrieval with private side information," in *Proc. IEEE ITW*, November 2018.

[23] A. Heidarzadeh, B. Garcia, S. Kadhe, S. E. Rouayheb, and A. Sprintson, "On the capacity of single-server multi-message private information retrieval with side information," in *Proc. Allerton*, October 2018.

[24] S. Li and M. Gastpar, "Single-server multi-message private information retrieval with side information," in *Proc. Allerton*, October 2018.

[25] S. Kadhe, B. Garcia, A. Heidarzadeh, S. El Rouayheb, and A. Sprintson, "Private information retrieval with side information," *IEEE Trans. Inf. Theory*, vol. 66, pp. 2032–2043, April 2020.

[26] R. Tandon, "The capacity of cache aided private information retrieval," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1078–1082, 2017.

[27] Y.-P. Wei, K. Banawan, and S. Ulukus, "Fundamental limits of cache-aided private information retrieval with unknown and uncoded prefetching," *IEEE Transactions on Information Theory*, vol. 65, no. 5, pp. 3215–3232, 2019.

[28] Z. Chen, Z. Wang, and S. A. Jafar, "The asymptotic capacity of private search," *IEEE Trans. Inf. Theory*, vol. 66, pp. 4709–4721, August 2020.

[29] Z. Wang, K. Banawan, and S. Ulukus, "Private set intersection: A multi-message symmetric private information retrieval perspective," [Online]. Available: arXiv:1912.13501.

[30] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Trans. Inf. Theory*, vol. 49, pp. 626–643, March 2003.

[31] R. E. Blahut, *Algebraic codes for data transmission.* Cambridge university press, 2003.

[32] H. Sun and S. A. Jafar, "Optimal download cost of private information retrieval for arbitrary message length," *IEEE Trans. Inf. Forensics Security*, vol. 12, pp. 2920–2932, December 2017.

[33] B. Herren, A. Arafa, and K. Banawan, "Download cost of private updating," in *ICC 2021 - IEEE International Conference on Communications*, pp. 1–6, 2021.

[34] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, pp. 471–480, July 1973.

[35] N. B. Shah, K. V. Rashmi, and K. Ramchandran, "One extra bit of download ensures perfectly private information retrieval," in *Proc. IEEE ISIT*, June 2014.

APPENDIX A: Evaluation of $H(\bar{W}_\theta)$

In (2.8), we state that

$$H(\bar{W}_\theta) = \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right). \tag{A.1}$$

For completeness, we briefly show that this is indeed the case here.

Fix some $f \leq L$, then let $M = \sum_{i=0}^{f} \binom{L}{i}$, which is to say that M is the number of possible realizations of $\bar{W}_\theta$. Since, given our problem formulation, every realization of $\bar{W}_\theta$ is equally likely, we have

$$\begin{aligned} H(\bar{W}_\theta) &= -\sum_{i=1}^{M} \frac{1}{M} \cdot \log_2 \left(\frac{1}{M} \right) && \text{(A.2)} \\ &= -M \cdot \frac{1}{M} \cdot \log_2 \left(\frac{1}{M} \right) && \text{(A.3)} \\ &= \log_2 (M) \,. && \text{(A.4)} \end{aligned}$$

Since $M = \sum_{i=0}^{f} \binom{L}{i}$, this shows that (A.1) holds.

APPENDIX B: Bound on Effective Value of f

In this appendix, for completeness, we show that

$$f < \frac{L}{2} \iff \lceil \bar{L} \rceil < L, \tag{B.1}$$

and hence if the maximum number of bit flips is more than half the message length it is optimal to ignore the outdated message (as per Corollary 1's result).

First, suppose that $f = \left\lfloor \frac{L-1}{2} \right\rfloor < \frac{L}{2}$. If L is odd, then $f = \frac{L-1}{2}$, and so it follows that

$$\sum_{i=0}^{L} \binom{L}{i} = 2 \cdot \sum_{i=0}^{\frac{L-1}{2}} \binom{L}{i} = 2^L \iff \sum_{i=0}^{f} \binom{L}{i} = 2^{L-1}. \tag{B.2}$$

This means that for odd L, we have

$$\bar{L} = \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right) = L - 1, \tag{B.3}$$

and so $\frac{L-1}{2}$ is the maximum value of f satisfying $\lceil \bar{L} \rceil < L$ when L is odd.

Next, suppose that L is even. It follows that

$$\sum_{i=0}^{L} \binom{L}{i} = 2 \cdot \sum_{i=0}^{\left\lfloor \frac{L-1}{2} \right\rfloor} \binom{L}{i} + \binom{L}{\frac{L}{2}} = 2^L \iff \sum_{i=0}^{f} \binom{L}{i} < 2^{L-1}. \tag{B.4}$$

This means that for even L, we have

$$\bar{L} = \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right) < L - 1. \tag{B.5}$$

Also, note that for even L

$$\sum_{i=0}^{\frac{L}{2}} \binom{L}{i} = \sum_{i=0}^{\left\lfloor \frac{L-1}{2} \right\rfloor} \binom{L}{i} + \binom{L}{\frac{L}{2}} > 2^{L-1}. \tag{B.6}$$

This means that if $f \geq \frac{L}{2}$, then $\lceil \bar{L} \rceil = L$, and so $\left\lfloor \frac{L-1}{2} \right\rfloor$ is the maximizing value of f satisfying $\lceil \bar{L} \rceil < L$ when L is even.

Therefore, for any message length L, we have

$$f < \frac{L}{2} \iff \lceil \bar{L} \rceil < L. \tag{B.7}$$

This completes the proof.