entropy

MDPI

Article

Towards Characterizing the Download Cost of Cache-Aided Private Updating [†]

Bryttany Stark [1], Ahmed Arafa [1,*] and Karim Banawan [2,3]

[1] Department of Electrical and Computer Engineering, University of North Carolina at Charlotte, Charlotte, NC 28223, USA; bryttany.stark@gmail.com
[2] Department of Electronics and Communications Engineering, The American University in Cairo, New Cairo 11835, Egypt; karim.banawan@aucegypt.edu
[3] Electrical Engineering Department, Faculty of Engineering, Alexandria University, Alexandria 21544, Egypt
[*] Correspondence: aarafa@charlotte.edu

[†] This paper is an extended version of our paper published in Proceedings of the IEEE International Conference on Communications (ICC), Montreal, QC, Canada, 14–23 June 2021.

Abstract

We consider the problem of privately updating a message out of K messages from N replicated and non-colluding databases where a user has an *outdated* version of the message $\hat{W}_\theta$ of length L bits that differ from the current version W_θ in at most f bits. The user also has a cache containing coded combinations of the K messages (with a pre-specified structure), which are unknown to the N databases (unknown prefetching). The cache Z contains ℓ linear combinations from all K messages in the databases with $r = \frac{\ell}{L}$ being the caching ratio. The user needs to retrieve W_θ correctly using a private information retrieval (PIR) scheme without leaking information about the message index θ to any individual database. Our objective is to jointly design the prefetching (i.e., the structure of said linear combinations) and the PIR strategies to achieve the least download cost. We propose a novel achievable scheme based on syndrome decoding where the cached linear combinations in Z are designed to be bits pertaining to the syndrome of W_θ according to a specific linear block code. We derive a general lower bound on the optimal download cost for $0 \leq r \leq 1$, in addition to achievable upper bounds. The upper and lower bounds match for the cases when r is exceptionally low or high, or when $K = 3$ messages for arbitrary r. Such bounds are derived by developing novel *cache-aided arbitrary message length* PIR schemes. Our results show a significant reduction in the download cost if $f < \frac{L}{2}$ when compared with downloading W_θ directly using typical cached-aided PIR approaches.

Keywords: private information retrieval; coded caching; private updating; syndrome decoding

Academic Editor: Chi Wan Sung

Received: 19 May 2025
Revised: 7 July 2025
Accepted: 8 July 2025
Published: 4 August 2025

Citation: Stark, B.; Arafa, A.; Banawan, K. Towards Characterizing the Download Cost of Cache-Aided Private Updating. *Entropy* **2025**, *27*, 828. https://doi.org/10.3390/e27080828

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 growing interest from the information and coding theory communities. The classical PIR problem is reformulated 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

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

Following [2], the capacity (or its reciprocal, the normalized download cost) of many variations of the problem have been investigated; see [3–17], and the surveys in [18,19].

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 of 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 reapply 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 [20], 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., [21–27]. 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., [28,29], 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 work, we introduce the problem of *cache-aided private updating with unknown prefetching* for an L-bit length 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 [30,31], 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 (in this work, we assume that the databases are *semi-honest*, in the sense that they truthfully obey the retrieval process, but the databases are curious to learn the identity of the desired file). The user also has access to a private cache Z containing ℓ linear combinations of each message, with $r = \frac{\ell}{L}$ being the caching ratio. The structure of such linear combinations is pre-specified to facilitate the retrieval procedure. By jointly designing the prefetching (i.e., the structure of the aforementioned cache contents) and the updating procedures, 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 for arbitrary K, N, f, L, and r.

To that end, we propose a novel achievable scheme that is based on the *syndrome decoding* idea introduced in [32], 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 [33]. This transformation, in effect, maps the messages into their corresponding syndromes. Thus, the problem is reduced to retrieving the syndrome representation of the messages (i.e., the auxiliary messages) that comprises $\lceil \bar{L} \rceil = \left\lceil \log_2 \left(\sum_{i=0}^{f} \binom{L}{i} \right) \right\rceil \leq L$ bits, where L is the original message length.

In the case of $r = 0$, we directly apply the PIR scheme in [34] to the auxiliary messages of length $\lceil \bar{L} \rceil$, which is optimal under the message length constraints. In the case where r satisfies $0 < r \leq \frac{1}{1+N+N^2+\cdots+N^{K-1}}$ (denoted as very low r), $\frac{1}{1+N} \leq r \leq 1$ (denoted as very high r), we extend the PIR scheme in [34] to the cache-aided setting in [29], and develop a novel *cache-aided arbitrary message length* PIR scheme to solve our problem. We also present an achievable scheme for the mid-range r, satisfying $\frac{1}{1+N+N^2+\cdots+N^{K-1}} < r < \frac{1}{1+N}$, tailored for the case of $K = 3$ messages, and discuss possible extensions for arbitrary K afterwards. 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 $\tilde{r} = \frac{\ell}{\lceil \bar{L} \rceil}$. This is in effect a higher caching ratio than r, which in turns lead to a lower download cost as in [29]. 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 [29], 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 is perfectly characterized for very high caching ratios, and is characterized within a maximum gap of only 2 bits otherwise. Notably, such a gap is 0 if $\bar{L}$ is an integer. 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 $f < \frac{L}{2}$.

The rest of the paper is organized as follows. Our system model is described in Section 2. The main results are presented in Section 3, with the main converse proof following in Section 4, and the achievability proofs in Sections 5 and 6. Section 7 includes a discussion on extending our achievability results, and the paper is concluded in Section 8.

2. System Model

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

$$H(W_i) = L, \quad 1 \leq i \leq K, \tag{2}$$
$$H(W_1, \ldots, W_K) = H(W_1) + \ldots + H(W_K). \tag{3}$$

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 ($[K]$ denotes the set $\{1, 2, \ldots, K\}$) but not the database (this is true if message θ, for example, has been previously obtained in a private manner). 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 are denoted by a random variable Z. The cache is populated through a *prefetching phase* in which the user caches pre-specified linear combinations from each message W_i, $i \in [K]$, with $\ell < L$ bits (specifically, we consider the case when the prefetching and retrieval strategies can be jointly designed, i.e., we assume that the information source performing the prefetching may provide a linear combination of its content with any desired structure to assist the user in minimizing the download cost in the retrieval phase). 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

$$Z = [W_1 R_1,\ W_2 R_2,\ \cdots,\ W_K R_K]. \tag{4}$$

The explicit design of R_i, $i \in [K]$ is specified along the lines of the achievability proof. We assume that the contents of the cache are *unknown* to the databases, as in, e.g., [21,27,29]. We define the *caching ratio* as

$$r = \frac{\ell}{L}. \tag{5}$$

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

$$H(Z) = \sum_{i=1}^{K} H(W_i R_i) \leq KLr, \tag{6}$$
$$I(W_i; Z) = H(W_i R_i) \leq Lr, \quad 1 \leq i \leq K. \tag{7}$$

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 $\hat{W}_\theta$ (clearly, $f \geq 1$ must hold; otherwise, there is no need to update $\hat{W}_\theta$). 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:

$$H(W_\theta) = H(\hat{W}_\theta) = L, \tag{8}$$

$$H(\bar{W}_\theta) = \log_2\left(\sum_{i=0}^{f}\binom{L}{i}\right) \triangleq \bar{L}, \tag{9}$$

$$H(W_\theta|\hat{W}_\theta) = H(\bar{W}_\theta|\hat{W}_\theta) = \bar{L}, \tag{10}$$

$$H(\bar{W}_\theta|\hat{W}_\theta, W_\theta) = 0, \tag{11}$$

$$|\bar{W}_\theta| \leq f \leq L, \tag{12}$$

where $|\cdot|$ denotes cardinality. We assume that the maximum Hamming distance f between the outdated and updated message is known to the user. By (9), 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 follows (in case the number of cached bits is greater than this bound in (13), the extra bits can be ignored by the user):

$$\ell \leq \lceil \bar{L} \rceil. \tag{13}$$

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 (we use the notation x_S to denote the collection of $\{x_i,\ i \in S\}$)

$$I\left(W_{[K]\backslash\{\theta\}}, \bar{W}_\theta; Q_{1:N}^{[\theta]} \middle| \hat{W}_\theta, Z\right) = 0. \tag{14}$$

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

$$H\left(A_n^{[\theta]} \middle| Q_n^{[\theta]}, W_{1:K}\right) = 0. \tag{15}$$

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

$$\left(Q_n^{[1]}, A_n^{[1]}, \hat{W}_1, W_{1:K}\right) \sim \left(Q_n^{[k]}, A_n^{[k]}, \hat{W}_k, W_{1:K}\right), \tag{16}$$

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*:

$$H\left(W_\theta \middle| A_{1:N}^{[\theta]}, Q_{1:N}^{[\theta]}, \hat{W}_\theta, Z\right) = 0. \tag{17}$$

The overall system model is depicted in Figure 1. We also include a list of notation with their definitions in Table 1 for ease of presentation.

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 (16) and the correctness constraint (17). 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.,

$$\bar{D} = \sum_{n=1}^{N} H\left(A_n^{[\theta]}\right). \tag{18}$$

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, we solve for

$$\bar{D}_L = \min\{\bar{D} : (\bar{D}, L) \text{ is achievable}\}. \tag{19}$$

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 [2,29]. Our main result, however, shows that we can use $\hat{W}_\theta$ to do strictly better.

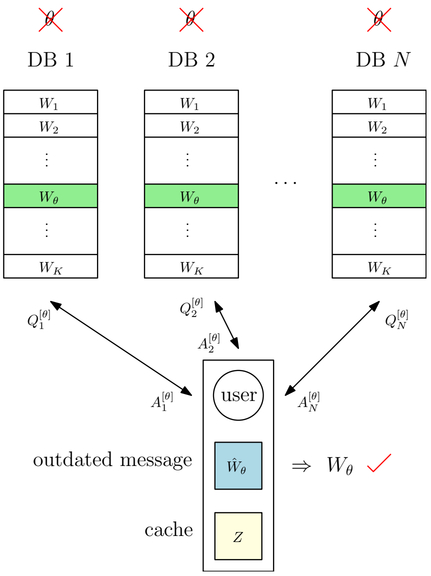

Figure 1. Cache-aided private updating with unknown prefetching system model.

Table 1. Key notations and system parameters.

Symbol	**Definition**
K	number of messages
N	number of databases
L	message length
θ	index of the required message
$\hat{W}_\theta$	outdated message
W_θ	current message
f	upper bound on differences between outdated and current messages
Z	cache content
ℓ	number of linearly-combined bits cached from each message
r	caching ratio: ℓ/L
$\bar{L}$	number of bits sufficient to update the message: $\log_2(\sum_{i=0}^{f}\binom{L}{i})$

3. Main Results

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*

$$\bar{D}_L \geq \left\lceil \max_{i\in\{2,\ldots,K+1\}} (\bar{L}-Lr)\sum_{j=0}^{K+1-i}\frac{1}{N^j} - Lr\sum_{j=0}^{K-i}\frac{K+1-i-j}{N^j} \right\rceil, \tag{20}$$

with $\bar{L}$ *defined in* (9)*.*

The proof of Theorem 1 is provided in Section 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 [29], for $s \in \{1,2,\ldots,K-1\}$, we define a caching ratio r_s as

$$r_s = \frac{\binom{K-2}{s-1}}{\binom{K-2}{s-1} + \sum_{i=0}^{K-1-s}\binom{K-1}{s+i}(N-1)^i N}. \tag{21}$$

Now, we say that a caching ratio r is *very low* if $0 \leq r \leq r_1 = \frac{1}{1+N+N^2+\cdots+N^{K-1}}$, *very high* if $r_{K-1} = \frac{1}{1+N} \leq r \leq 1$, and *mid-range* otherwise. We are now ready to present our first achievability result.

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

$$\bar{D}_L \leq \left\lceil (\lceil\bar{L}\rceil - Lr)\cdot\sum_{i=0}^{K-1}\frac{1}{N^i} - Lr\cdot\sum_{i=0}^{K-2}\frac{K-1-i}{N^i} \right\rceil, \tag{22}$$

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

$$\bar{D}_L \leq \lceil\bar{L}\rceil - Lr, \tag{23}$$

with $\bar{L}$ *defined in* (9)*.*

The proof of Theorem 2 is provided in Section 5.

Combining the achievability bounds in Theorem 2 with the converse bound in Theorem 1, we obtain a fairly tight, up to a ceiling difference of $\bar{L}$, characterization of the optimal download cost $\bar{D}_L$ for very low and very high 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 ratios, we have*

$$\left\lceil (\bar{L}-Lr)\sum_{j=0}^{K-1}\frac{1}{N^j} - Lr\sum_{j=0}^{K-2}\frac{K-1-j}{N^j} \right\rceil \leq \bar{D}_L \leq \left\lceil (\lceil\bar{L}\rceil-Lr)\sum_{j=0}^{K-1}\frac{1}{N^j} - Lr\sum_{j=0}^{K-2}\frac{K-1-j}{N^j} \right\rceil, \tag{24}$$

and for very high caching ratios, we have

$$\bar{D}_L = \lceil\bar{L}\rceil - Lr \tag{25}$$

Proof. The right-hand side inequality of (24) is given directly by Theorem 2. By choosing $i = 2$ in (20), we obtain the left-hand side inequality in (24). Similarly, by choosing $i = K - 1$ in (20), we obtain the result in (25) (note that Lr is an integer, and so in this case, the converse and achievability bounds match). This concludes the proof. □

We now have the following remarks.

Remark 1. *The result in Corollary 1 generalizes our preliminary work on the private updating problem with no caching involved [35]. Specifically, plugging in $r = 0$ in Corollary 1 directly gives ([35], Theorem 1).*

Remark 2. *Consider the result in (24). From (9) and (12), 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 $f < \frac{L}{2}$ (this can be readily shown using the binomial theorem; details are in Appendix A). Combining this with the results in ([29], 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 2, 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.*

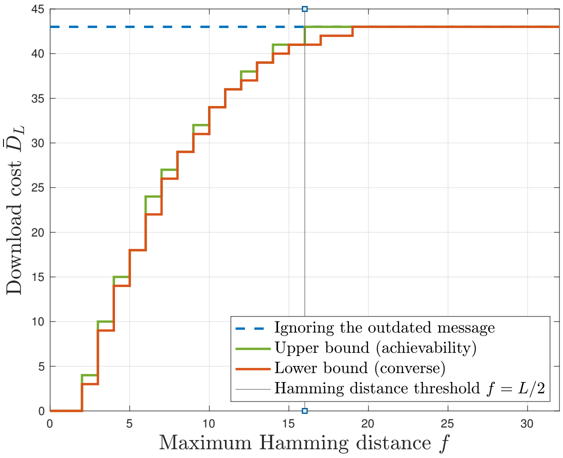

Figure 2. Download cost of cache-aided private updating with unknown prefetching with $L = 32$ bits, $N = 2$ databases, $K = 3$ messages, and $r = \frac{1}{10}$ caching ratio (Corollary 1's results for the very low caching ratio).

Remark 3. *If L and f are such that $\bar{L} = \lceil \bar{L} \rceil$, then the upper and lower bounds in (24) match. We will see that this holds if a* perfect code *(a code that attains the Hamming bound with equality [33]) by which the queries are sent exists (cf. Section 5). Otherwise, if $\bar{L} < \lceil \bar{L} \rceil$, one can show using similar arguments as in ([34], Section 7.2) that the two bounds are within 2 bits for $N \geq 2$ databases.*

Next, we have the following achievability result regarding mid-range caching ratios.

Theorem 3 (Mid-Range Achievability)**.** *In the cache-aided private updating problem with unknown prefetching with $K = 3$ messages, for mid-range effective caching ratios, the optimal download cost is upper bounded by*

$$\bar{D}_L \leq \left\lceil (\lceil \bar{L} \rceil - Lr)\left(1 + \frac{1}{N}\right) - Lr \right\rceil \tag{26}$$

with $\bar{L}$ defined in (9).

The proof of Theorem 3 is provided in Section 6. In Section 7, we include a discussion on extending the above achievability result for arbitrary K.

Combining the mid-range achievability bound in Theorem 3 and the converse bound in Theorem 1 for $i = K$, we characterize the optimal download cost for $\bar{D}_L$ for mid-range caching ratios when $K = 3$. Furthermore, combining this characterization with the result of Corollary 1 gives a complete characterization of $\bar{D}_L$ when $K = 3$ for *any* caching ratio. To this end, we define the $K = 3$ *converse* bound $\bar{D}_{K=3}(r)$ and the $K = 3$ *achievability* bound $\bar{D}^{K=3}(r)$ to express this characterization:

$$\bar{D}_{K=3}(r) = \begin{cases} \left\lceil (\bar{L} - Lr) \cdot \sum_{i=0}^{2} \frac{1}{N^i} - Lr \cdot \sum_{i=0}^{1} \frac{2-i}{N^i} \right\rceil, & \text{if } 0 \le r \le r_1; \\ \left\lceil (\bar{L} - Lr)\left(1 + \frac{1}{N}\right) - Lr \right\rceil, & \text{if } r_1 \le r \le r_2; \\ \lceil \bar{L} \rceil - Lr, & \text{if } r_2 \le r \le 1. \end{cases} \tag{27}$$

$$\bar{D}^{K=3}(r) = \begin{cases} \left\lceil (\lceil \bar{L} \rceil - Lr) \cdot \sum_{i=0}^{2} \frac{1}{N^i} - Lr \cdot \sum_{i=0}^{1} \frac{2-i}{N^i} \right\rceil, & \text{if } 0 \le r \le r_1; \\ \left\lceil (\lceil \bar{L} \rceil - Lr)\left(1 + \frac{1}{N}\right) - Lr \right\rceil, & \text{if } r_1 \le r \le r_2; \\ \lceil \bar{L} \rceil - Lr, & \text{if } r_2 \le r \le 1. \end{cases} \tag{28}$$

We have now proved the following corollary.

Corollary 2 ($K = 3$ Characterization)**.** *In the cache-aided private updating problem with unknown prefetching where $K = 3$, for any caching ratio, we have*

$$\bar{D}_{K=3}(r) \le \bar{D}_L \le \bar{D}^{K=3}(r) \tag{29}$$

4. Proof of Theorem 1: Converse

In this section, we derive the general (converse) lower bound for the download cost in Theorem 1. To do so, we prove two useful lemmas, analogues to their counterparts in the cache-aided PIR setting of [29], 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 [29] 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*

$$\bar{D} - (\bar{L} - Lr) \ge I\left(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} \middle| W_{1:k-1}, \hat{W}_{k-1}, Z\right) \tag{30}$$

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

Proof. We start with the right-hand side of (30),

$$\begin{aligned} & I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-1}, \hat{W}_{k-1}, Z) \\ &= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}, W_{k-1} | W_{1:k-2}, \hat{W}_{k-1}, Z) - I(W_{k:K}; W_{k-1} | W_{1:k-2}, \hat{W}_{k-1}, Z) && (31) \\ &= I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-2}, \hat{W}_{k-1}, Z) + I(W_{k:K}; W_{k-1} | Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]}, W_{1:k-2}, \hat{W}_{k-1}, Z) \\ &\overset{(17)}{=} I(W_{k:K}; Q_{1:N}^{[k-1]}, A_{1:N}^{[k-1]} | W_{1:k-2}, \hat{W}_{k-1}, Z) && (32) \end{aligned}$$

$$\overset{(14)}{=} I(W_{k:K}; A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, \hat{W}_{k-1}, Z) \tag{33}$$

$$= H(A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{34}$$

$$\overset{(17)}{=} H(A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(A^{[k-1]}_{1:N}, W_{k-1} | Q^{[k-1]}_{1:N}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{35}$$

$$\leq H(A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(W_{k-1} | Q^{[k-1]}_{1:N}, W_{1:k-2}, W_{k:K}, \hat{W}_{k-1}, Z) \tag{36}$$

$$\overset{(14)}{=} H(A^{[k-1]}_{1:N} | Q^{[k-1]}_{1:N}, W_{1:k-2}, \hat{W}_{k-1}, Z) - H(W_{k-1} | \hat{W}_{k-1}, Z) \tag{37}$$

$$\overset{(18),(3)}{\leq} \bar{D} - H(W_{k-1} | \hat{W}_{k-1}, W_{k-1}R_{k-1}) \tag{38}$$

$$= \bar{D} - \Big(H(W_{k-1}, W_{k-1}R_{k-1} | \hat{W}_{k-1}) - H(W_{k-1}R_{k-1} | \hat{W}_{k-1}) \Big) \tag{39}$$

$$= \bar{D} - \Big(H(W_{k-1} | \hat{W}_{k-1}) + H(W_{k-1}R_{k-1} | \hat{W}_{k-1}, W_{k-1}) - H(W_{k-1}R_{k-1} | \hat{W}_{k-1}) \Big) \tag{40}$$

$$\overset{(10),(7)}{\leq} \bar{D} - (\bar{L} - Lr). \tag{41}$$

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 $k \in \{2, \ldots, K\}$, the mutual information term in Lemma 1 can be inductively lower bounded as*

$$\begin{aligned} &I\Big(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N} \Big| W_{1:k-1}, \hat{W}_{k-1}, Z\Big) \\ &\quad \geq \frac{1}{N} I\Big(W_{k+1:K}; Q^{[k]}_{1:N}, A^{[k]}_{1:N} \Big| W_{1:k}, \hat{W}_k, Z\Big) + \frac{\bar{L} - Lr}{N} - (K - k + 1)Lr. \end{aligned} \tag{42}$$

Proof. We start with the left-hand side of (42),

$$\begin{aligned} &I(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N} | W_{1:k-1}, \hat{W}_{k-1}, Z) \\ &= I(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N}, Z, \hat{W}_{k-1} | W_{1:k-1}) - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}) &(43) \\ &= I(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N} | W_{1:k-1}) + I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}, Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N}) \\ &\quad - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}) &(44) \\ &\geq I(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N} | W_{1:k-1}) - I(W_{k:K}; Z, \hat{W}_{k-1} | W_{1:k-1}). &(45) \end{aligned}$$

Now, for the first term in (45), we have

$$I(W_{k:K}; Q^{[k-1]}_{1:N}, A^{[k-1]}_{1:N} | W_{1:k-1}) \tag{46}$$

$$\geq \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; Q^{[k-1]}_n, A^{[k-1]}_n | W_{1:k-1}) \tag{47}$$

$$\overset{(16)}{=} \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; Q^{[k]}_n, A^{[k]}_n | W_{1:k-1}) \tag{48}$$

$$= \frac{1}{N} \sum_{n=1}^{N} I(W_{k:K}; A^{[k]}_n | W_{1:k-1}, Q^{[k]}_n) \tag{49}$$

$$\overset{(15)}{=} \frac{1}{N} \sum_{n=1}^{N} H(A^{[k]}_n | W_{1:k-1}, Q^{[k]}_n) \tag{50}$$

$$
\begin{aligned}
&\geq \frac{1}{N}\sum_{n=1}^{N} H(A_n^{[k]}|W_{1:k-1},\hat{W}_k,Z,Q_{1:N}^{[k]},A_{1:n-1}^{[k]}) && (51)\\
&\overset{(15)}{=} \frac{1}{N}\sum_{n=1}^{N} I(W_{k:K};A_n^{[k]}|W_{1:k-1},\hat{W}_k,Z,Q_{1:N}^{[k]},A_{1:n-1}^{[k]}) && (52)\\
&= \frac{1}{N} I(W_{k:K};A_{1:N}^{[k]}|W_{1:k-1},\hat{W}_k,Z,Q_{1:N}^{[k]}) && (53)\\
&\overset{(14)}{=} \frac{1}{N} I(W_{k:K};Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k-1},\hat{W}_k,Z) && (54)\\
&\overset{(17)}{=} \frac{1}{N} I(W_{k:K};W_k,Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k-1},\hat{W}_k,Z) && (55)\\
&= \frac{1}{N} I(W_{k:K};Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k},\hat{W}_k,Z) + \frac{1}{N} I(W_{k:K};W_k|W_{1:k-1},\hat{W}_k,Z) && (56)\\
&= \frac{1}{N} I(W_{k:K};Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k},\hat{W}_k,Z) + \frac{1}{N} H(W_k|\hat{W}_k,Z) && (57)\\
&\overset{(10),(7)}{\geq} \frac{1}{N} I(W_{k+1:K};Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k},\hat{W}_k,Z) + \frac{\bar{L}-Lr}{N}. && (58)
\end{aligned}
$$

Note that (58) follows from a similar argument in Lemma 1 starting at (37). Next, for the second term in (45), we have

$$
\begin{aligned}
&I(W_{k:K};Z,\hat{W}_{k-1}|W_{1:k-1})\\
&= H(W_{k:K}|W_{1:k-1}) - H(W_{k:K}|W_{k-1},Z,\hat{W}_{k-1}) && (59)\\
&= (K-k+1)L - (K-k+1)L(1-r) && (60)\\
&= (K-k+1)Lr && (61)
\end{aligned}
$$

Combining the above results concludes the proof. □

We now apply the result of Lemma 2 recursively on that of Lemma 1 to get the general lower bound through the following series of inequalities:

$$
\begin{aligned}
\bar{D} &\overset{(30)}{\geq} (\bar{L}-Lr) + I(W_{k:K};Q_{1:N}^{[k-1]},A_{1:N}^{[k-1]}|W_{1:k-1},\hat{W}_1,Z) && (62)\\
&\overset{(42)}{\geq} (\bar{L}-Lr) + \frac{\bar{L}-Lr}{N}\\
&\quad + \frac{1}{N} I(W_{k+1:K};Q_{1:N}^{[k]},A_{1:N}^{[k]}|W_{1:k},\hat{W}_k,Z)\\
&\quad - (K-k+1)Lr && (63)\\
&\overset{(42)}{\geq} (\bar{L}-Lr) + \frac{\bar{L}-Lr}{N} + \frac{\bar{L}-Lr}{N^2}\\
&\quad + \frac{1}{N^2} I(W_{k+2:K};Q_{1:N}^{[k+1]},A_{1:N}^{[k+1]}|W_{1:k+1},\hat{W}_{k+1},Z)\\
&\quad - (K-k+1)Lr + \frac{(K-k+2)Lr}{N} && (64)\\
&\overset{(42)}{\geq} \dots && (65)\\
&= (\bar{L}-Lr)\sum_{j=0}^{K+1-k}\frac{1}{N^j} - Lr\sum_{j=0}^{K-k}\frac{K+1-k-j}{N^j} && (66)
\end{aligned}
$$

Next, since the bound in (66) is valid for arbitrary k, it is still valid for k corresponding to the maximum possible lower bound, i.e., (66) gives K intersecting line segments, therefore, the download cost $\bar{D}$ is lower bounded by their maximum value

$$\bar{D} \geq \max_{i \in \{2,\ldots,K+1\}} (\bar{L} - Lr) \sum_{j=0}^{K+1-i} \frac{1}{N^j} - Lr \sum_{j=0}^{K-i} \frac{K+1-i-j}{N^j}. \tag{67}$$

Since (67) 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 (67) to get (20).

This concludes the converse proof.

5. Proof of Theorem 2: Achievability for Very Low and Very High Caching Ratios

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 [32] (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 [36], 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 a 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 an arbitrary message length in [34], 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 [35].

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 $C = 2/3$ (from (1)). Setting $r = 0$ in (22), 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 generator matrix G and parity check matrix H given by

$$\mathsf{G} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \quad \mathsf{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \tag{68}$$

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

$$\mathsf{s}_\theta = W_\theta \mathsf{H}^T. \tag{69}$$

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 [32]. 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

$$\begin{aligned} \mathsf{s}_1 = W_1 \mathsf{H}^T &= \begin{bmatrix} a_1 + a_2 & a_1 + a_3 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{a}_1 & \bar{a}_2 \end{bmatrix}, \end{aligned} \tag{70}$$

$$\begin{aligned} \mathsf{s}_2 = W_2 \mathsf{H}^T &= \begin{bmatrix} b_1 + b_2 & b_1 + b_3 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{b}_1 & \bar{b}_2 \end{bmatrix}. \end{aligned} \tag{71}$$

Assume $\theta = 1$. Since $\lceil \bar{L} \rceil = N^{K-1}$, we can apply a *non-symmetric* PIR scheme [34] to decode s_1. This scheme is shown in Table 2, and has a download cost of $\bar{D} = 3$ bits, which is optimal in this case since it meets the converse bound.

Table 2. Query table for $N = K = 2$, $L = 3$, $f = 1$, and $r = 0$.

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

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

$$\mathsf{G} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}, \quad \mathsf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{72}$$

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

$$\begin{aligned} \mathsf{s}_1 = W_1 \mathsf{H}^T &= \begin{bmatrix} a_1 + a_2 + a_3 & a_1 + a_2 + a_4 & a_1 + a_5 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{a}_1 & \bar{a}_2 & \bar{a}_3 \end{bmatrix}, \end{aligned} \tag{73}$$

$$\begin{aligned} \mathsf{s}_2 = W_2 \mathsf{H}^T &= \begin{bmatrix} b_1 + b_2 + b_3 & b_1 + b_2 + b_4 & b_1 + b_5 \end{bmatrix} \\ &\triangleq \begin{bmatrix} \bar{b}_1 & \bar{b}_2 & \bar{b}_3 \end{bmatrix}. \end{aligned} \tag{74}$$

Since $\lceil \bar{L} \rceil = N^{K-1} + 1$, we follow the methodology in [34]; 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 [37]. The technique in [37] 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 h, and the same from database 2 yet with $h' = h + e_\theta$, where e_i is the ith standard basis vector. The full PIR scheme is shown in Table 3, and it 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.

Table 3. Query table for $N = K = 2$, $L = 5$, $f = 1$, and $r = 0$.

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

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 [33], we know that such a code exists if

$$2^{\lceil \bar{L} \rceil} \le \sum_{j=0}^{2f} \binom{L}{j}. \tag{75}$$

In addition, such a code must satisfy the Hamming bound [33]:

$$\sum_{j=0}^{f} \binom{L}{j} \le 2^{\lceil \bar{L} \rceil}. \tag{76}$$

By the definition of $\bar{L}$ in (9), both (75) and (76) 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 ([34], 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 $\hat{W}_\theta$ [32]. This shows that (22) 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

$$R_i = \mathsf{H}^T, \quad \forall i \in [K], \tag{77}$$

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 an arbitrary message length that utilize the results from [29]. In particular, for all $s \in \{1, 2, \ldots, K-1\}$, we define the message length of a cache-aided PIR scheme from [29] with caching ratio r_s as

$$L_r(s) = \binom{K-2}{s-1} + \sum_{i=0}^{K-1-s} \binom{K-1}{s+i} (N-1)^i N, \tag{78}$$

and the normalized download cost of such a scheme as

$$D_r(s) = \frac{\sum_{i=0}^{K-1-s} \binom{K}{s+1+i} (N-1)^i N}{\binom{K-2}{s-1} + \sum_{i=0}^{K-1-s} \binom{K-1}{s+i} (N-1)^i N}. \tag{79}$$

For very low caching ratio r, we recall from [29] that the optimal normalized download cost of a cache-aided PIR scheme is

$$D^*(r) = (1-r) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i}, \tag{80}$$

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

$$D^*(r) = (1-r). \tag{81}$$

With these tools in hand, in the remainder of this section, 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 (22)

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-aided 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

$$L = G_0 L_r(1) + L_0, \tag{82}$$

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} (without caching). 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

$$L_0 = G_1 N^{K-1} + L_1, \tag{83}$$

where $G_1 = \left\lfloor \frac{L_0}{N^{k-1}} \right\rfloor$ 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 [37] 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

$$L_1 = G_2(N-1) + L_2, \tag{84}$$

where $G_2 = \left\lfloor \frac{L_1}{N-1} \right\rfloor$ 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 [37] with $L_2 + 1$ databases and message size L_2. Therefore, the message size and the achievable download cost are

$$L = G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) + L_2, \tag{85}$$

$$D = \begin{cases} G_0 L_r(1) D^*(r_1) + G_1 \frac{N^{K-1}}{C} + G_2 N, & \text{if } L_2 = 0, \\ G_0 L_r(1) D^*(r_1) + G_1 \frac{N^{K-1}}{C} + G_2 N + L_2 + 1, & \text{otherwise.} \end{cases} \tag{86}$$

We next show that the achievable download cost in (86) satisfies $D \leq \lceil D^*(r) \cdot L \rceil$. To this end, we have the following lemma.

Lemma 3. *For two very low caching ratios r_a and r_b with $0 \le r_a \le r_b \le r_1$, we have*

$$D^*(r_a) - D^*(r_b) = (r_b - r_a) \cdot D_c, \tag{87}$$

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

Proof. We begin from the left-hand side of (87) and use (80) to write

$$\begin{aligned} & D^*(r_a) - D^*(r_b) \\ &= \left((1-r_a) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r_a \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right) - \left((1-r_b) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - r_b \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right) && (88) \\ &= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} + (r_b - r_a) \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} && (89) \\ &= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{1 + (K-1-i)}{N^i} && (90) \\ &= (r_b - r_a) \cdot \sum_{i=0}^{K-1} \frac{K-i}{N^i}. && (91) \end{aligned}$$

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

Now towards proving $D \le \lceil D^*(r) \cdot L \rceil$, 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

$$\begin{aligned} & G_0 L_r(1) D^*(r_1) + G_1 \frac{N^{K-1}}{C} + G_2 N + L_2 \\ & < D^*(r) \cdot \Big(G_0 L_r(1) + G_1 N^{K-1} + G_2(N-1) + L_2 \Big) + 1. \end{aligned} \tag{92}$$

First, we group the terms in (92); we need to show that

$$\begin{aligned} & - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) + G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r) \right) - (G_2(N-1) + L_2) D^*(r) \\ & < 1 - G_2 N - L_2. \end{aligned} \tag{93}$$

Focusing on the left-hand side of (93), we use Lemma 3 to simplify the expression, while noting that $D^*(0) = \frac{1}{C}$, as follows:

$$\begin{aligned} & - G_0 L_r(1) \cdot (D^*(r) - D^*(r_1)) + G_1 N^{K-1} \cdot \left(\frac{1}{C} - D^*(r) \right) - (G_2(N-1) + L_2) D^*(r) \\ &= -G_0 L_r(1) D_c (r_1 - r) + G_1 N^{K-1} D_c r - (G_2(N-1) + L_2) \left(\frac{1}{C} - D_c r \right) && (94) \\ &= D_c \cdot \big(- G_0 L_r(1) r_1 + G_0 L_r(1) r + G_1 N^{K-1} r + G_2(N-1) r + L_2 r \big) - \frac{G_2(N-1) + L_2}{C} && (95) \\ &= D_c \cdot (-G_0 L_r(1) r_1 + Lr) - \frac{G_2(N-1) + L_2}{C} && (96) \\ &= D_c \cdot (-G_0 + Lr) - \frac{G_2(N-1) + L_2}{C}. && (97) \end{aligned}$$

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

$$D_c \cdot (Lr - G_0) - \frac{G_2(N-1) + L_2}{C} = -\frac{G_2(N-1) + L_2}{C}. \tag{98}$$

Now, substituting (98) back into (93), we now need to show

$$0 < 1 - G_2N - L_2 + \frac{G_2(N-1) + L_2}{C}. \tag{99}$$

If $N = 1$, then $G_2 = 0$, and so (99) 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 (99) gives

$$\begin{align} & 1 - G_2N - L_2 + \frac{G_2(N-1) + L_2}{C} \tag{100} \\ &= 1 - G_2N + G_2\frac{N^K - 1}{N^{K-1}} + L_2\left(\frac{N^K - 1}{N^{K-1}(N-1)} - 1\right) \\ &= 1 - G_2\frac{1}{N^{K-1}} + L_2\left(\frac{N^{K-1} - 1}{N^{K-1}(N-1)}\right). \tag{101} \end{align}$$

We wish to find a lower bound for (101). 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 $G_2(N-1) < L_1 \leq N^{K-1} - 1$ from (84). Plugging these values into (101) gives

$$\begin{align} & 1 - G_2\frac{1}{N^{K-1}} + L_2\left(\frac{N^{K-1} - 1}{N^{K-1}(N-1)}\right) \\ &\geq 1 - \frac{G_2(N-1)}{N^{K-1}(N-1)} + \frac{N^{K-1} - 1}{N^{K-1}(N-1)} \tag{102} \\ &> 1 - \frac{N^{K-1} - 1}{N^{K-1}(N-1)} + \frac{N^{K-1} - 1}{N^{K-1}(N-1)} = 1. \tag{103} \end{align}$$

and so (99) holds for $N \geq 2$.

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

$$\begin{align} & G_0L_r(1)D^*(r_1) + G_1\frac{N^{K-1}}{C} + G_2N + L_2 + 1 \\ &< D^*(r) \cdot \Big(G_0L_r(1) + G_1N^{K-1} + G_2(N-1) + L_2\Big) + 1. \tag{104} \end{align}$$

First, we group the terms in (104); we need to show that

$$\begin{align} & G_1N^{K-1} \cdot \left(\frac{1}{C} - D^*(r)\right) - G_0L_r(1) \cdot (D^*(r) - D^*(r_1)) - (G_2(N-1) + L_2)D^*(r) \\ &< 1 - G_2N - L_2 - 1. \tag{105} \end{align}$$

By (98), we substitute the left-hand side of (105) so that we have

$$0 < 1 - G_2N - L_2 + \frac{G_2(N-1) + L_2}{C} - 1. \tag{106}$$

Since $L_2 \geq 1$, we have $N \geq 2$, and so (106) holds by (103). 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 ([34], Theorem 4), the above scheme is both correct and private. To conclude our proof, we define a normalized version of r:

$$\tilde{r} = \frac{Lr}{\lceil L \rceil}, \tag{107}$$

as the *effective* caching ratio. Clearly, by (13), $0 \leq \tilde{r} \leq 1$. Now, 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 be 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

$$\begin{aligned} \bar{D} &\leq \lceil D^*(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil && (108)\\ &= \left\lceil \lceil \bar{L} \rceil (1-\tilde{r}) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - \lceil \bar{L} \rceil \tilde{r} \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil && (109)\\ &= \left\lceil (\lceil \bar{L} \rceil - Lr) \cdot \sum_{i=0}^{K-1} \frac{1}{N^i} - Lr \cdot \sum_{i=0}^{K-2} \frac{K-1-i}{N^i} \right\rceil, && (110) \end{aligned}$$

which is precisely (22).

5.4. Very High Caching Ratio: Proof of (23)

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

$$L = G_0 L_r(K-1) + L_0, \tag{111}$$

where $G_0 = \left\lfloor \frac{L-Lr}{N} \right\rfloor$, 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 obtain 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

$$\begin{aligned} L &= G_0 L_r(K-1) + L_0, && (112)\\ D &= G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0. && (113) \end{aligned}$$

We next show that the achievable download cost in (113) 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

$$G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0 < D^*(r) \cdot L + 1. \tag{114}$$

First, we rearrange the terms in (114) as

$$G_0 L_r(K-1) D^*(r_{K-1}) + L_0 - C_0 - D^*(r) \cdot L < 1, \tag{115}$$

and then we reduce the left-hand side of (115) as follows

$$
\begin{aligned}
&G_0L_r(K-1)D^*(r_{K-1})+L_0-C_0-D^*(r)\cdot L \\
&= G_0(1+N)(1-\frac{1}{1+N})+L_0-C_0-(1-r)\cdot L \quad (116)\\
&= G_0N+L_0-C_0-L+Lr \quad (117)\\
&= -C_0-G_0+Lr=0. \quad (118)
\end{aligned}
$$

Thus, (114) 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 ([34], 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 be 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

$$
\begin{aligned}
\bar{D} &\leq \lceil D^*(\tilde{r})\cdot\lceil\bar{L}\rceil\rceil \quad (119)\\
&= \lceil (1-\tilde{r})\cdot\lceil\bar{L}\rceil\rceil \quad (120)\\
&= \lceil\lceil\bar{L}\rceil - Lr\rceil = \lceil\bar{L}\rceil - Lr, \quad (121)
\end{aligned}
$$

which is precisely (23).

6. Proof of Theorem 3: Achievability for $K = 3$ with Mid-Range Caching Ratios

What follows is a cache-aided achievable scheme for retrieving an arbitrary L bits, for mid-range caching ratios given fixed $K = 3$ setting $\left(\frac{1}{1+N+N^2} = r_1 < r < r_2 = \frac{1}{1+N}\right)$. This scheme leverages cache-aided PIR schemes for very high and very low caching ratios but within an asymmetric PIR setting instead.

First, consider the asymmetric cache-aided PIR scheme with $N = 3$ and $L = 3$ in Table 4. This scheme does not utilize all of the databases, nor does it utilize the cache in full. This scheme downloads one useful bit privately at a cost of 1 bit, and it is an asymmetric version of the cache-aided PIR scheme for very high caching ratios. This scheme can be repeated up to five more times to get up to five more useful bits, and each additional bit is obtained privately.

Table 4. Asymmetric query table with $N = K = L = 3$ and very high r. Here, we have $Z = \{a_1, a_2, b_1, b_2, c_1, c_2\}$.

Database 1	Database 2	Database 3
$a_3 + b_1 + c_1$		

Next, consider the asymmetric cache-aided PIR scheme with $N = 3$ and $L = 6$ in Table 5. While this scheme does utilize all of the databases, it has asymmetric traffic between the databases, and it also does not utilize the cache in full. This scheme downloads $1 + N$ useful bits at a cost of $2 + N$, and it is an asymmetric version of the cache-aided PIR scheme for very low caching ratios. Once again, this scheme can be repeated up to five more times to get up to $5\cdot(1+N)$ more useful bits, and each additional set of $1 + N$ bits is obtained privately.

In these examples, we see that each scheme can be used a total of $N \cdot Lr = 6$ times. Now, note that these two schemes can be used *in conjunction* with one another, and that rather than repeating the same scheme over and over again, we can just use them interchangeably to suit our needs.

Table 5. Asymmetric query table with $N = K = 3$, $L = 6$ and very low r. Here, we have $Z = \{a_1, a_2, b_1, b_2, c_1, c_2\}$.

Database 1	Database 2	Database 3
$a_3 + b_1$		
$a_4 + c_1$		
$b_3 + c_3$		
	$a_5 + b_3 + c_3$	$a_6 + b_3 + c_3$

Consider a cache-aided PIR example where $N = 3$, $L = 14$, and $r = \frac{2}{14}$. Note that r is now mid-range. We can use a combination of the asymmetric very high caching ratio scheme and very low caching ratio scheme to download the remaining 12 useful bits as shown in Table 6. First, we use the asymmetric very high caching ratio scheme four times to obtain four useful bits at a cost of 4 bits total. Then, we use the the asymmetric very low caching ratio scheme two times to download the remaining $2 \cdot (1 + N) = 8$ useful bits at a cost of $2 \cdot (2 + N) = 10$, and so the total download cost is 14.

Table 6. Query table for $N = K = 3$, $L = 14$, and mid-range caching ratio $r = \frac{2}{14}$. Here, we have $Z = \{a_1, a_2, b_1, b_2, c_1, c_2\}$.

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

It is also worth noting that in the same scenario, but with $L = 13$ and $r = \frac{2}{13}$, we can use almost the almost the same query structure as in Table 6. The only difference is that we *truncate* the given scheme by not making the query for a_{14}. In this particular case, this truncation strategy can be performed again to obtain an $L = 12$, $r = \frac{2}{12}$ query structure.

In general, one can use a combination of $N \cdot Lr - 1$ very high and very low caching ratio schemes, and then if the remaining number of useful bits left to download is some ℓ with $1 < \ell < N + 1$, use a truncated very low caching ratio scheme. Otherwise, just a normal very high or very low caching ratio scheme can be used.

In order to determine the number of times these very high and very low schemes are used, along with the number of bits that are downloaded via the truncation strategy, we define three terms as follows:

$$G_1 = \left\lfloor \frac{L_r(1) \cdot Lr - L}{N} \right\rfloor, \tag{122}$$

$$G_2 = \left\lfloor \frac{L - L_r(2) \cdot Lr}{N} \right\rfloor, \tag{123}$$

$$L_3 = L - (G_1 + G_2(1+N)) - Lr. \tag{124}$$

The G_1 term is the number of times a very high caching ratio scheme is used, while G_2 is the number of times a very low caching ratio scheme is used. The L_3 term is the number of bits obtained from the truncation strategy when it is used. According to these terms, it follows that the message size and the achievable download cost are

$$L = G_1 + G_2(1+N) + L_3 + Lr, \tag{125}$$

$$D = \begin{cases} G_1 + G_2(2+N), & \text{if } L_3 = 0, \\ G_1 + G_2(2+N) + L_3 + 1, & \text{otherwise.} \end{cases} \tag{126}$$

Lastly, for mid-range caching ratios with $K = 3$, we recall from [29] that the optimal normalized download cost of a cache-aided PIR scheme is

$$D^*(r) = (1-r)\left(1 + \frac{1}{N}\right) - r. \tag{127}$$

We next show that the achievable download cost in (126) satisfies $D \le \lceil D^*(r) \cdot L \rceil$. To this end, it suffices to show that $D < D^*(r) \cdot L + 1$ for two cases. For the first case, let $L_3 = 0$. We wish to show that

$$G_1 + G_2(2+N) + L_3 - D^*(r) \cdot L < 1. \tag{128}$$

Reducing the left-hand side of (128), we have

$$\begin{aligned} & G_1 + G_2(2+N) + L_3 - D^*(r) \cdot L \\ & = G_1 + G_2(2+N) + L_3 - ((1-r) \cdot (1 + \frac{1}{N}) - r) \cdot L \qquad (129) \\ & = G_1 + G_2(2+N) + L_3 - (1 - 2r + \frac{1-r}{N}) \cdot (G_1 + G_2(1+N) + L_3 + Lr) \qquad (130) \\ & = G_2 - Lr + (2r - \frac{1-r}{N}) \cdot L \qquad (131) \\ & = G_2 + Lr - \frac{L - Lr}{N} \qquad (132) \\ & = G_2 - \frac{L - (1+N) \cdot Lr}{N} \qquad (133) \\ & = G_2 - \frac{L - L_r(2) \cdot Lr}{N}. \qquad (134) \end{aligned}$$

Substituting (134) into (128), we need to show that

$$G_2 - \frac{L - L_r(2) \cdot Lr}{N} < 1, \tag{135}$$

which clearly holds by (123). It follows that (128) holds when $L_3 = 0$. To show that this is also the case when $L_3 \ge 1$, we use a lemma.

Lemma 4. *In the $K = 3$ setting, for any caching ratio r with $\frac{1}{1+N+N^2} = r_1 < r < r_2 = \frac{1}{1+N}$, we have*

$$L_3 = 0 \Longleftrightarrow \frac{L - L_r(2) \cdot Lr}{N} \in \mathbb{Z} \tag{136}$$

The proof of Lemma 4 can be found in Appendix B.

Now, for the second case, let $L_3 \geq 1$. We wish to show that

$$G_1 + G_2(2+N) + L_3 - D^*(r) \cdot L < 0. \tag{137}$$

By (134), we substitute the left-hand side of (137) so that we have

$$G_2 - \frac{L - L_r(2) \cdot Lr}{N} < 0, \tag{138}$$

which holds by Lemma 4. Thus, this completes the proof that $D \leq \lceil D^*(r) \cdot L \rceil$ for mid-range caching ratios in the $K = 3$ setting.

Since the above PIR scheme is constructed as a concatenation of several PIR schemes that are both correct and private (by [34], 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 be 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

$$\begin{aligned} \bar{D} &\leq \lceil D^*(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil \qquad (139) \\ &= \left\lceil \lceil \bar{L} \rceil (1-\tilde{r}) \left(1 + \frac{1}{N}\right) - \lceil \bar{L} \rceil \tilde{r} \right\rceil \qquad (140) \\ &= \left\lceil (\lceil \bar{L} \rceil - Lr) \left(1 + \frac{1}{N}\right) - Lr \right\rceil \qquad (141) \end{aligned}$$

which is precisely (26).

7. 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 perform the same for an effective caching ratio $\tilde{r}$, defined in (107), with $\frac{1}{1+N+N^2+\cdots+N^{K-1}} = r_1 \leq \tilde{r} \leq r_{K-1} = \frac{1}{1+N}$, i.e., such caching ratios that are *mid-range*. With Theorem 3 and Corollary 2, this has been performed for the $K = 3$ case. However, this is still an open question for when K is arbitrary.

Our approach for our achievability 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 performed in [34] for the classical PIR setting.

From [29], for $r_s < r < r_{s+1}$ and $\alpha \in [0,1]$ with $r = \alpha r_s + (1-\alpha) r_{s+1}$, we define

$$\bar{D}(r) = \alpha D_r(s) + (1-\alpha) D_r(s+1). \tag{142}$$

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. This is likewise the case for mid-range caching ratios r when $K = 3$ in Theorem 3. 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 section is to present a motivating example that shows what these results may look like. Consider the following example setting: $N = 3$, $K = 4$, and $r_{K-2} \leq r \leq r_{K-1}$. 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, let us consider some scenarios with a caching ratio $r = \frac{1}{7}$ starting with the case when the number of cached bits is 3, and so the total message length is 21. Using the methods found in this work, we have a scheme satisfying $D \leq \lceil \bar{D}(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$ given in Table 7.

Using these same methods, if there are two cached bits with a total message length of 14, then we also have a scheme satisfying $D \leq \lceil \bar{D}(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$ using a subset of the queries in Table 7. However, for the case when there is only one cached bit with a total message length of seven, we have no scheme satisfying $D \leq \lceil \bar{D}(\tilde{r}) \cdot \lceil \bar{L} \rceil \rceil$, not with using the methods in this work at least. It is worth noting that for some other mid-range caching ratios with this setting, the scheme from [37] can be used to produce some satisfactory results ($r = \frac{1}{6}$ for example) but not for the case when $r = \frac{1}{7}$ in this setting. This is discussed in more detail in [38].

The question remains: *why does this pattern break, and why it is difficult to find an alternative query structure?* The answer we have come to is that it 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.

Table 7. Query table for $N = 3$, $K = 4$, $L = 21$, and $r = \frac{3}{21}$. Here, we have $Z = \{a_1, a_2, a_3, b_1, b_2, b_3, c_1, c_2, c_3, d_1, d_2, d_3\}$

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

8. Conclusions

In this work, we introduce 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. To solve this problem, we develop novel *arbitrary message length cache-aided* PIR schemes for different caching ratios. These schemes are then combined with syndrome decoding techniques to guarantee privacy and efficiency. Such schemes are 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.

Outside of the issues discussed in Section 7, 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. Imposing less control over the prefetching phase is one direction to be pursued in the research line of cache-aided private updating.

Author Contributions: Conceptualization, A.A. and K.B.; formal analysis, B.S.; writing—original draft preparation, B.S.; writing—review and editing, A.A. and K.B.; supervision, A.A.; funding acquisition, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the U.S. National Science Foundation under Grant CNS 21-14537 and Grant ECCS 21-46099.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: This work was done when B. Stark was a Master's student at UNC Charlotte.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Bound on Effective Value of f

For completeness, we show that

$$f < \frac{L}{2} \iff \lceil \bar{L} \rceil < L, \tag{A1}$$

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 it follows that

$$\sum_{i=0}^{L} \binom{L}{i} = 2 \cdot \sum_{i=0}^{\frac{L-1}{2}} \binom{L}{i} = 2^{L} \Leftrightarrow \sum_{i=0}^{f} \binom{L}{i} = 2^{L-1}. \tag{A2}$$

So for odd L, we have $\bar{L} = \log_2\left(\sum_{i=0}^{f} \binom{L}{i}\right) = L - 1$, 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

$$\sum_{i=0}^{L} \binom{L}{i} = 2 \cdot \sum_{i=0}^{\left\lfloor \frac{L-1}{2} \right\rfloor} \binom{L}{i} + \binom{L}{\frac{L}{2}} = 2^{L} \Leftrightarrow \sum_{i=0}^{f} \binom{L}{i} < 2^{L-1}. \tag{A3}$$

So for even L, we have $\bar{L} = \log_2\left(\sum_{i=0}^{f} \binom{L}{i}\right) < L - 1$. Also, note that for even L,

$$\sum_{i=0}^{\frac{L}{2}} \binom{L}{i} = \sum_{i=0}^{\left\lfloor \frac{L-1}{2} \right\rfloor} \binom{L}{i} + \binom{L}{\frac{L}{2}} > 2^{L-1}. \tag{A4}$$

This means that $\left\lfloor \frac{L-1}{2} \right\rfloor$ is the maximum value of f satisfying $\lceil \bar{L} \rceil < L$ when L is even.

Therefore, for any message length L, we have the result in Remark 2. This completes the proof.

Appendix B. Proof of Lemma 4

First, we note that

$$\frac{L_r(1) \cdot Lr - L}{N} + \frac{L - L_r(2) \cdot Lr}{N} = N \cdot Lr, \tag{A5}$$

and so it follows that $G_1 + G_2 \in \{N \cdot Lr,\ N \cdot Lr - 1\}$.

Consider the case when $G_1 + G_2 = N \cdot Lr$. Plugging this into (124), it can be shown that

$$G_2 = \frac{L - L_r(2) \cdot Lr}{N} - \frac{L_3}{N}. \tag{A6}$$

Substituting (A6) back into (124), it can be shown that

$$G_1 = \frac{L_r(1) \cdot Lr - L}{N} + \frac{L_3}{N}. \tag{A7}$$

If $L_3 > 0$, then substituting such a value into (A7) would contradict (122). Likewise, if $L_3 < 0$, then substituting such a value into (A6) would contradict (123). Therefore,

$$G_1 + G_2 = N \cdot Lr \Rightarrow L_3 = 0 \Rightarrow \frac{L - L_r(2) \cdot Lr}{N} \in \mathbb{Z}. \tag{A8}$$

Now consider the case when $G_1 + G_2 = N \cdot Lr - 1$. Plugging this into (124), it can be shown that

$$G_2 = \frac{L - L_r(2) \cdot Lr}{N} - \frac{L_3 - 1}{N}. \tag{A9}$$

If $L_3 \leq 0$, then substituting such a value into (A9) would contradict (123). Likewise, if $L_3 \geq N + 1$, then substituting such a value into (A9) would also contradict (123). Therefore, we have

$$G_1 + G_2 = N \cdot Lr - 1 \Rightarrow L_3 \neq 0 \Rightarrow \frac{L - L_r(2) \cdot Lr}{N} \notin \mathbb{Z}. \tag{A10}$$

Finally, by combining (A8) and (A10), we can obtain the result in the lemma. This completes the proof.

References

1. Chor, B.; Kushilevitz, E.; Goldreich, O.; Sudan, M. Private Information Retrieval. *J. ACM* **1998**, *45*, 965–981. [CrossRef]
2. Sun, H.; Jafar, S.A. The Capacity of Private Information Retrieval. *IEEE Trans. Inf. Theory* **2017**, *63*, 4075–4088. [CrossRef]
3. Banawan, K.; Ulukus, S. The Capacity of Private Information Retrieval from Coded Databases. *IEEE Trans. Inf. Theory* **2018**, *64*, 1945–1956. [CrossRef]
4. Sun, H.; Jafar, S.A. The Capacity of Symmetric Private Information Retrieval. *IEEE Trans. Inf. Theory* **2019**, *65*, 322–329. [CrossRef]
5. Banawan, K.; Ulukus, S. Multi-Message Private Information Retrieval: Capacity Results and Near-Optimal Schemes. *IEEE Trans. Inf. Theory* **2018**, *64*, 6842–6862. [CrossRef]
6. Tajeddine, R.; Gnilke, O.W.; Karpuk, D.; Freij-Hollanti, R.; Hollanti, C.; Rouayheb, S.E. Private Information Retrieval Schemes for Coded Data with Arbitrary Collusion Patterns. In Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany, 25–30 June 2017.
7. Wang, Q.; Skoglund, M. On PIR and Symmetric PIR from Colluding Databases with Adversaries and Eavesdroppers. *IEEE Trans. Inf. Theory* **2019**, *65*, 3183–3197. [CrossRef]
8. Tian, C.; Sun, H.; Chen, J. Capacity-Achieving Private Information Retrieval Codes with Optimal Message Size and Upload Cost. *IEEE Trans. Inf. Theory* **2019**, *65*, 7613–7627. [CrossRef]
9. Guo, T.; Zhou, R.; Tian, C. On the Information Leakage in Private Information Retrieval Systems. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 2999–3012. [CrossRef]
10. Banawan, K.; Ulukus, S. The Capacity of Private Information Retrieval from Byzantine and Colluding Databases. *IEEE Trans. Inf. Theory* **2019**, *65*, 1206–1219. [CrossRef]
11. Attia, M.A.; Kumar, D.; Tandon, R. The Capacity of Private Information Retrieval from Uncoded Storage Constrained Databases. *IEEE Trans. Inf. Theory* **2020**, *66*, 6617–6634. [CrossRef]
12. Sun, H.; Jafar, S.A. The Capacity of Private Computation. *IEEE Trans. Inf. Theory* **2019**, *65*, 3880–3897. [CrossRef]
13. Kumar, S.; i Amat, A.G.; Rosnes, E.; Senigagliesi, L. Private Information Retrieval from a Cellular Network with Caching at the Edge. *IEEE Trans. Commun.* **2019**, *67*, 4900–4912. [CrossRef]

14. Raviv, N.; Tamo, I.; Yaakobi, E. Private Information Retrieval in Graph-Based Replication Systems. *IEEE Trans. Inf. Theory* **2020**, *66*, 3590–3602. [CrossRef]
15. Yao, X.; Liu, N.; Kang, W. The Capacity of Multi-round Private Information Retrieval from Byzantine Databases. In Proceedings of the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 7–12 July 2019.
16. Samy, I.; Tandon, R.; Lazos, L. On the Capacity of Leaky Private Information Retrieval. In Proceedings of the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 7–12 July 2019.
17. D'Oliveira, R.G.L.; El Rouayheb, S. One-Shot PIR: Refinement and Lifting. *IEEE Trans. Inf. Theory* **2020**, *66*, 2443–2455. [CrossRef]
18. Ulukus, S.; Avestimehr, S.; Gastpar, M.; Jafar, S.A.; Tandon, R.; Tian, C. Private retrieval, computing, and learning: Recent progress and future challenges. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 729–748. [CrossRef]
19. Vithana, S.; Wang, Z.; Ulukus, S. Private information retrieval and its extensions: An introduction, open problems, future directions. *IEEE BITS Inf. Theory Mag.* **2023**, *3*, 67–85. [CrossRef]
20. Jia, Z.; Jafar, S. X-Secure T-Private Federated Submodel Learning. In Proceedings of the 2021 IEEE International Conference on Communications (ICC 2021), Montreal, QC, Canada, 14–23 June 2021.
21. Chen, Z.; Wang, Z.; Jafar, S.A. The Capacity of T-Private Information Retrieval with Private Side Information. *IEEE Trans. Inf. Theory* **2020**, *66*, 4761–4773. [CrossRef]
22. Wei, Y.P.; Banawan, K.; Ulukus, S. The Capacity of Private Information Retrieval with Partially Known Private Side Information. *IEEE Trans. Inf. Theory* **2019**, *65*, 8222–8231. [CrossRef]
23. Wei, Y.P.; Ulukus, S. The Capacity of Private Information Retrieval with Private Side Information Under Storage Constraints. *IEEE Trans. Inf. Theory* **2019**, *66*, 2023–2031. [CrossRef]
24. Shariatpanahi, S.P.; Siavoshani, M.J.; Maddah-Ali, M.A. Multi-Message Private Information Retrieval with Private Side Information. In Proceedings of the 2018 IEEE Information Theory Workshop (ITW), Guangzhou, China, 25–29 November 2018.
25. Heidarzadeh, A.; Garcia, B.; Kadhe, S.; Rouayheb, S.E.; Sprintson, A. On the Capacity of Single-Server Multi-Message Private Information Retrieval with Side Information. In Proceedings of the 2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 2–5 October 2018.
26. Li, S.; Gastpar, M. Single-server Multi-message Private Information Retrieval with Side Information. In Proceedings of the 2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 2–5 October 2018.
27. Kadhe, S.; Garcia, B.; Heidarzadeh, A.; El Rouayheb, S.; Sprintson, A. Private Information Retrieval with Side Information. *IEEE Trans. Inf. Theory* **2020**, *66*, 2032–2043. [CrossRef]
28. Tandon, R. The capacity of cache aided private information retrieval. In Proceedings of the 2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 3–6 October 2017; pp. 1078–1082. [CrossRef]
29. Wei, Y.P.; Banawan, K.; Ulukus, S. Fundamental Limits of Cache-Aided Private Information Retrieval with Unknown and Uncoded Prefetching. *IEEE Trans. Inf. Theory* **2019**, *65*, 3215–3232. [CrossRef]
30. Chen, Z.; Wang, Z.; Jafar, S.A. The Asymptotic Capacity of Private Search. *IEEE Trans. Inf. Theory* **2020**, *66*, 4709–4721. [CrossRef]
31. Wang, Z.; Banawan, K.; Ulukus, S. Private Set Intersection: A Multi-Message Symmetric Private Information Retrieval Perspective. *arXiv* **2020**, arXiv:1912.13501. [CrossRef]
32. Pradhan, S.S.; Ramchandran, K. Distributed source coding using syndromes (DISCUS): Design and construction. *IEEE Trans. Inf. Theory* **2003**, *49*, 626–643. [CrossRef]
33. Blahut, R.E. *Algebraic Codes for Data Transmission*; Cambridge University Press: Cambridge, UK, 2003.
34. Sun, H.; Jafar, S.A. Optimal Download Cost of Private Information Retrieval for Arbitrary Message Length. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 2920–2932. [CrossRef]
35. Herren, B.; Arafa, A.; Banawan, K. Download Cost of Private Updating. In Proceedings of the ICC 2021—IEEE International Conference on Communications, Montreal, QC, Canada, 14–23 June 2021; pp. 1–6. [CrossRef]
36. Slepian, D.; Wolf, J.K. Noiseless coding of correlated information sources. *IEEE Trans. Inf. Theory* **1973**, *IT-19*, 471–480. [CrossRef]
37. Shah, N.B.; Rashmi, K.V.; Ramchandran, K. One extra bit of download ensures perfectly private information retrieval. In Proceedings of the 2014 IEEE International Symposium on Information Theory, Honolulu, HI, USA, 29 June–4 July 2014. [CrossRef]
38. Herren, B. Download Cost of Cache-Aided Private Updating with Unknown Prefetching. Master's Thesis, University of North Carolina at Charlotte, Charlotte, NC, USA, 2022.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.