Two-Phase Private Information Retrieval with Constraint on Computation Cost

Jeng-Chi Weng

Abstract

For the task of Private Information Retrieval (PIR), a user wishes to retrieve a message from N non-colluding databases, each storing K messages, without revealing any information about which is the desired message to each of the databases. To address this challenge, we explore the potential benefits of leveraging a locally equipped cache to optimize resource utilization. In addition to the cache ratio and the communication cost required, which have long been intensively discussed in the literature, we present a general two-phase scheme that takes into account a constraint on computation cost. This constraint may be associated with the time required to obtain the desired information from a database. Our analysis focuses on delineating the trade-off relationship between the cache ratio and the online communication cost while operating within a specified computational cost constraint.

1 Introduction

The problem of private information retrieval was first introduced and addressed by Chor *et al.* [1] in 1995. In this problem, a user wishes to retrieve a message from a network of databases, each of which stores the same set of messages, without revealing the identity of the desired message. While the intuitive solution would involve downloading the entire message set from one of the databases, this approach proves highly inefficient and incurs substantial communication costs. Thus, the goal of PIR is to discover a more efficient solution.

Chor *et al.* proposed a basic scheme that requires only two databases. Since then, many improved schemes have been developed and in 2017, the capacity of PIR, which is the maximum number of bits of desired information that is privately retrieved per downloaded information bit, was derived by Sun and Jafar [2] from the information-theoretic perspective. Subsequently, an improved scheme [3] was proposed and it maintains the same download cost performance while minimizing message size compared to all other capacity-achieving codes.

However, these schemes primarily aimed to minimize communication costs and establish fundamental limits within various problem settings. They did not take into account a critical practical performance metric for PIR schemes: computation cost. Modern databases contain a significant number of messages, and if capacity-achieving PIR schemes exhibit linear growth in computation cost with the number of messages, they can demand substantial time and resources to retrieve information from a database.

On the other hand, in the computer science community where PIR problem is studied in the setting with each message being one bit in length, Corrigan-Gibbs and Kogan [4] addressed the abovementioned problem by proposing a two-phase scheme. In the offline phase, which takes place before the user has decided which message (bit) of the database it wants to retrieve, the user fetches some ”hints” from a database. Subsequently, in the online phase, which takes place after the user has selected the desired message, the user sends queries to the databases and retrieves answers from them. Combined with the ”hints” fetched in the offline phase, the user then recovers the desired message. In terms of computation cost, they define it as the number of messages (bits) involved in the computation of an answer and their scheme only requires computation costs that are sublinear to the number of messages.

In the information theory community, two-phase schemes are also extensively explored. Wei *et al.* [5] considered the problem of PIR when the user is equipped with a cache that holds an uncoded fraction from each of the messages with databases unaware of the cached contents. They also aimed to characterize the optimal download cost under such problem setting. Seo *et al.* [6] further delved into cases where the cache contents are coded and their scheme achieves the optimal download cost with a certain cache size.

Therefore, we aim to integrate these concepts, and propose a comprehensive scheme under different computation constraints with these schemes as its special case.

This report is organized as follows. In Section 2, we formally define our problem. Section 3 presents our proposed scheme and we characterize its performance. Section 4 delves into the privacy aspects of our scheme. Finally, in Section 5, we draw conclusions regarding the problem and our approach.

2 Problem Statement

Most of the problem setups are similar to the problem setup defined in [2]. The main differences are that our schemes work in two phases, and we further define computation cost.

We consider K independent messages $W_1, \cdots, W_K$, each of size L_W.

$$H(W_1, \cdots, W_K) = H(W_1) + \cdots + H(W_K),$$
$$H(W_1) = \cdots = H(W_K) = L_W$$

2.1 Offline Phase

In the offline phase, where the user hasn't decided which message is of interest. The user generates an offline query Q_{off} to an offline database. In addition, since the user has no information of the contents of the messages, the query must be independent of them.

$$I(W_1, \cdots, W_K; Q_{\text{off}}) = 0$$

Upon receiving the query, the offline database responds an answer to the user, which will be stored in the user's cache Z of size L_Z. Also, the answer is fully determined by the offline query the user sent and the messages it stored.

$$H(Z \mid Q_{\text{off}}, W_1, \cdots, W_K) = 0$$

Note that the cache contents are unknown to the online databases, i.e. the offline database and the online databases don't communicate with each other.

2.2 Online Phase

In the online phase, after the user has decided which message is of interest, the user wishes to retrieve message W_θ privately, where $\theta \in \{1, \cdots, K\}$ is a chosen message index. There exist N non-colluding online databases, each stores all the K messages. Therefore, the user generates N queries $Q_1, \cdots, Q_N$, which is dependent on the index of the desired message θ and the offline query Q_{off}, and sends $Q_n^{(\theta)}$ to the n-th database. Once again, the queries are independent of the messages.

$$\forall \theta \in \{1, \cdots, K\}, \quad I(W_1, \cdots, W_K; Q_1^{(\theta)}, \cdots, Q_N^{(\theta)}) = 0$$

Also, upon receiving the query $Q_n^{(\theta)}$, the n-th database generates the answer $A_n^{(\theta)}$, which is a function of the query $Q_n^{(\theta)}$ and the messages it stored.

$$\forall \theta \in \{1, \cdots, K\}, \forall n \in \{1, \cdots, N\}, \quad H(A_n^{(\theta)} \mid Q_n^{(\theta)}, W_1, \cdots, W_K) = 0$$

2.2.1 Correctness

With the answers received $A_1^{(\theta)}, \cdots, A_N^{(\theta)}$ and the cached contents and the query information, the user must be able to recover the desired message W_θ with probability of error P. The correctness of the scheme is examined by the probability of error P_e, and it must approach zero as the message size L_W approaches infinity.

$$\frac{1}{L_W} H(W_K \mid Z, A_1^{(\theta)}, \cdots, A_N^{(\theta)}, Q_{\text{off}}, Q_1^{(\theta)}, \cdots, Q_N^{(\theta)}) = o(L_W)$$

where $o(L_W)$ represents any term that approaches zero as L approach infinity.

2.2.2 Privacy

To protect privacy while retrieving the desire message, the query $Q_n^{(\theta)}$ sent to the n-th database must be indistinguishable. In other words, the distribution of the query $Q_n^{(\theta)}$ is independent of the index of the desired message.

$$\Pr(Q_n^{(\theta)} = q \mid \theta = k) = \Pr(Q_n^{(\theta)} = q \mid \theta = l)$$
$$k, l \in \{1, \cdots, K\}, n \in \{1, \cdots, N\}$$

2.3 Metric

2.3.1 Normalized Online Communication Cost

The online communication cost should be the sum of the length of the queries and the answers. However, in the information-theoretic setting, where message size can be arbitrarily large, upload cost (the length of the queries) is negligible compared to the download cost (the length of the answers). Thus, in terms of online communication cost, we only consider the download cost.

Accordingly, we define the online communication cost, denoted D_{on}, as the expected value of the total number of bits downloaded from all the online databases in the online phase. In addition, in order to properly characterize the performance of our schemes regardless of the message size L_W, we further normalized our download cost by the message size.

$$\text{Normalized Online Communication Cost} \triangleq \frac{D_{\text{on}}}{L_W}$$

2.3.2 Cache Ratio

Similarly, we define the cache size L_Z as the expected value of the total number of bits downloaded from the offline databases and cache ratio is defined as the cache size that is normalized by the message size L_W.

$$\text{Cache Ratio} \triangleq \frac{L_Z}{L_W}$$

2.3.3 Normalized Online Computation Cost

In Corrigan-Gibbs and Kogan's work [4], they define computation cost as the number of variables involved in the computation of an answer. For instance, the computation cost of $W_{1,1} + W_{2,1} + W_{3,1}$ will be 3, since there are 3 variables involved in the computation of the answer.

In our setting, we also define the online computation cost as the total number of variables involved in the computation of the all the answers. For instance, if the user requests $W_{1,1} + W_{2,1} + W_{3,1}$ and $W_{1,1} + W_{2,1}$, then the computation cost will be 5. Once again, we normalize the online computation cost by the message size L_W.

$$\text{Normalized Online Computation Cost} \triangleq \frac{\text{num of variables involved in the computation of answers}}{L_W}$$

Our goal is to optimize the achievable bound of tradeoff between cache ratio and normalized online communication cost under certain constraints on normalized online computation cost based on the existing schemes.

3 Proposed Scheme

Our scheme comprises two sub-schemes, each targeting specific optimization goals: one focuses on reducing the cache ratio, while the other concentrates on minimizing the normalized online communication cost. The cache-ratio-oriented scheme's objective is the reduction of the cache ratio, while the communication-cost-oriented scheme aims to decrease the normalized online communication cost. Notably, we demonstrate that each of these schemes operates efficiently within certain cache ratio ranges. Moreover, for cache ratios falling in between these established bounds, the achievable normalized online communication cost can be reached by memory-sharing the proposed schemes with neighboring cache ratios.

Furthermore, like in [2], where they partition each message into blocks of N^K bits, and like in [3], where they segment each message into blocks of $N-1$ bits, we adopt a similar approach of dividing the message into specific bit-count segments. Subsequently, we apply the same scheme individually to each of these message blocks.

3.1 Cache-Ratio-Oriented Scheme

3.1.1 Idea

Our inspiration stems from the work of Corrigan-Gibbs and Kogan [4]. In their work, they successfully achieved an online computation cost that exhibits sublinear growth concerning the number of messages, denoted as K. It is evident that when the message size is not constrained to one bit, a straightforward approach would involve replicating the scheme for each message bit. However, Sun and Jafar, in their information-theoretic formulation of PIR as presented in [2], demonstrated the existence of more sophisticated schemes for the PIR problem, some of which have been proven to excel in certain metrics, particularly communication cost. Therefore, our objective is to explore whether the scheme proposed by Corrigan-Gibbs and Kogan in [4] can be enhanced to accommodate messages of arbitrary size.

To this end, we introduce a scheme that shares several similarities with the basic framework mentioned in Corrigan-Gibbs and Kogan's paper. Ultimately, our investigation reveals that certain aspects of the information-theoretic formulation of messages can be harnessed to improve the efficiency of the scheme

3.1.2 Settings

Basic Settings

- We divide each message into blocks of N bits. For each block, we randomly permute the message bits within a message with bijection functions $f_i : \{1, \cdots, N\} \to \{1, \cdots, N\}, \forall i \in \{1, \cdots, K\}$. We represent the permuted messages as the following.

$$W_{1,1}, W_{1,2}, \cdots, W_{1,N-1}, W_{1,N}$$

$$W_{2,1}, W_{2,2}, \cdots, W_{2,N-1}, W_{2,N}$$

$$\cdots \quad \cdots \qquad\quad \cdots \quad \cdots$$

$$W_{K,1}, W_{K,2}, \cdots, W_{K,N-1}, W_{K,N}$$

  where $W_{i,j}$ corresponds to an actual message bit $W_{i,f_i(j)}$.

- s is a predetermined value that denotes the number of variables involved in an online sub-query.
- $s^{'}$ is another predetermined value that denotes the number of online sub-queries in a online query that is sent to an online database.
- Constraints on s,$s^{'}$:
  - $ss^{'} < K$

Setup

(1) Randomly divide K indices into sets of $ss^{'} + 1$ indices.
$\Rightarrow$ We get $M - 1$ sets of $ss^{'} + 1$ indices, denoted $G_1, G_2, \cdots, G_{M-1}$, and one set of $K - (M-1)(ss^{'} + 1)$ indices, denoted G_M, where $M = \lceil \frac{K}{ss^{'}+1} \rceil$.

(2) For G_M, we randomly choose $M(ss^{'} + 1) - K$ indices from $G_1, G_2, \cdots, G_{M-1}$ and add them to G_M so that the cardinality of G_M is $ss^{'} + 1$ as well.

(3) Construct $G_{1,n}, G_{2,n}, \cdots, G_{M,n}$ from $G_1, G_2, \cdots, G_M$ for all $n \in \{1, \cdots, N\}$, where $G_{j,n} = \{W_{i,n} \mid i \in G_j\}$.

3.1.3 Offline Phase

Algorithm Offline phase

```
$S = \{\}$
for $n \leftarrow 1$ to $N$ do
    • with prob $1 - \frac{ss^{'}}{K}$, cache the parity of $G_{1,n}, G_{2,n}, \cdots, G_{M,n}$
    • with prob $\frac{ss^{'}}{K}$, add all the bits $W_{1,n}, W_{2,n}, \cdots, W_{K,n}$, to $S$
end for
if $s \leq \frac{K}{2} + 1$:
    • Create 2 disjoint sets of size $s - 1$ from $\{1, 2, \cdots, K\}$
    • For every set, choose $s - 1$ random bits from different messages in $S$, where the message indices are in
      this set, and cache the parity of them.
else
    • Divide $K$ indices into $\lceil \frac{K}{K-(s-1)} \rceil$ sets.
    • For every set, choose $s - 1$ random bits from different files in $S$, where the message indices aren't in
      this set, and cache the parity of them.
⇒ We get $S_1, S_2, \cdots, S_{\lceil \frac{K}{K-(s-1)} \rceil}$, where for every message $W_i$, we can find a set $S_j$ such that $S_j$ doesn't
contain any bit of $W_i$.
```

3.1.4 Online Phase

Suppose the desired message is W_θ.

Algorithm Online phase

```
(1) Query $Q_n^{(\theta)}$ to the $n$-th database:
    if the parities of $G_{1,n}, G_{2,n}, \cdots, G_{M,n}$ is cached:
        (1) Find $G_{i,n}$ such that $W_{\theta,n} \in G_{i,n}$
        (2) Divide $G_{i,n} \setminus \{W_{\theta,n}\}$ into $s^{'}$ disjoint sets $S_{i,1}, S_{i,2}, \cdots, S_{i,s^{'}}$.
        (3) $Q_n^{(\theta)} = \{S_{i,1}, S_{i,2}, \cdots, S_{i,s^{'}}\}$
    else
        (1) Find $S_j$ such that it doesn't contain any bit of $F_\theta$.
        (2) Randomly select $s(s^{'} - 1)$ bits from different files whose bits aren't in $S_j \cup \{F_{\theta,n}\}$ and divide them
            into $s^{'} - 1$ disjoint sets $S_{i,1}, S_{i,2}, \cdots, S_{i,s^{'}-1}$.
        (3) $Q_n^{(\theta)} = \{S_j \cup \{W_{\theta,n}\}, S_{i,1}, S_{i,2}, \cdots, S_{i,s^{'}-1}\}$
(2) Send query $Q_n^{(\theta)}$ to the $n$-th database, and receive the corresponding answer $A_n^{(\theta)}$, which are the parities
    of the sets.
(3) Reconstruct $W_\theta$ from the answers $A_1^{(\theta)}, A_2^{(\theta)}, \cdots, A_N^{(\theta)}$ and the cache contents.
```

3.2 Communication-Cost-Oriented Scheme

3.2.1 Idea

The idea of this scheme originates from the scheme proposed in [3]. In that work, they partition each message into blocks, each consisting of $N-1$ bits, and introduce an additional dummy variable set to zero. This arrangement can be thought of as each online database, totaling $N-1$ in number, providing information of a bit of the desired message. However, to ensure privacy, each query must include a bit from every message.

Building upon this foundation, we pose a question: What if we augment the number of dummy variables? Could this approach lead to a reduction in computation cost? At a fundamental level, introducing more dummy variables has the potential to significantly decrease computation costs since the values of dummy variables are known and hence there is no need to include them in queries. However, this adjustment also necessitates downloading a greater number of bits. Consequently, we explore strategies for efficiently caching message information to ultimately lower online communication costs.

In this section, we present a scheme that requires the user to cache a larger volume of content but ultimately achieves a lower overall computation cost and online communication cost.

3.2.2 Settings

Basic Settings

- We divide each message into blocks of w bits.
- For each block, we pre-pend p zero-variables for each message. Since we already know the content of these dummy variables, which is zero, these zero-variables won't be sent in actual implementation. For example, if the scheme asks for $P_{1,1}+W_{1,1}+W_{2,1}$, we only send $W_{1,1}+W_{2,1}$ to the database. Hence these pre-pend variables won't be counted as variables for the cost of computation. We form these zero-variables for convenience of the construction of our scheme.

$$\begin{gathered}P_{1,1}, P_{1,2}, \cdots, P_{1,p}, W_{1,1}, W_{1,2}, \cdots, W_{1,w} \\ P_{2,1}, P_{2,2}, \cdots, P_{1,p}, W_{2,1}, W_{2,2}, \cdots, W_{2,w} \\ \cdots \quad \cdots \quad \cdots \qquad\qquad \cdots \quad \cdots \\ P_{K,1}, P_{K,2}, \cdots, P_{K,p}, W_{K,1}, W_{K,2}, \cdots, W_{K,w}\end{gathered}$$

- Constraints on the values of p,w:
  - $p+w \geq N+1$
  - $1 \leq w \leq N$.
- We denote $P_{1,i}, P_{2,i}, \cdots, P_{K,i}$ as the i-th dummy column, and $W_{1,j}, W_{2,j}, \cdots, W_{K,j}$ as the j-th non-dummy column.

3.2.3 Offline Phase

Algorithm Offline phase

```
(1) $S = \{\}$
(2) Uniformly randomly choose a bit from each message, can be either a zero-variable or a true message bit, and add it to $S$.
(3) Cache the parity of $S$.
(4) Select a non-dummy column first and uniformly randomly select $N$ columns, either dummy or non-dummy, from the remaining $p + w - 1$ columns.
(5) Cache every message bit of unselected non-dummy columns. For instance, if the first to third non-dummy columns aren't selected, we further cache $W_{1,1}, \cdots, W_{K,1}, W_{1,2}, \cdots, W_{K,2}, W_{1,3}, \cdots, W_{K,3}$.
```

3.2.4 Online Phase

Suppose the desired message is W_θ.

- Let the bit related to W_θ that is in S, either $P_{\theta,i}$ or $W_{\theta,j}$, be b_θ.
- For all selected columns, either dummy or non-dummy, the corresponding message bits of the desired message form a set D, called "*DesiredSet*", of size $N + 1$ (number of selected columns).

Algorithm Online phase

```
(1) Set $D$ is modified for the following two conditions.
      • $b_\theta$ is in $D$, i.e. $b_\theta$ is a bit of a selected column.
        $\Rightarrow D \leftarrow D \setminus \{b_\theta\}$
      • $b_\theta$ isn't in $D$.
        $\Rightarrow$ Uniformly randomly select a random bit of a selected dummy column, denoted $P_{\theta,j}$. Note that there must be one, since $1 \leq w \leq N$.
        $\Rightarrow D \leftarrow D \setminus \{P_{\theta,j}\}$
(2) Queries $Q^{(\theta)} = \{S \setminus \{b_\theta\} \cup d \mid d \in D\}$, which is of size $N$, since $D$ is of size $N$.
(3) Send each of the $N$ queries in $Q^{(\theta)}$ to each of the online database and receive the answers $A^{(\theta)}$
(4) Reconstruct the desired message $W_\theta$ from the answers $A^{(\theta)}$ and the cache contents.
```

3.3 Performance

	Cache-Ratio-Oriented Scheme	Communication-Cost-Oriented Scheme
Cache ratio $\frac{L_Z}{L_W}$	$\begin{cases}(1-\frac{ss^{'}}{K})\lceil\frac{K}{ss^{'}+1}\rceil+\frac{\lceil\frac{K}{K-(s-1)}\rceil}{N} & ,s\neq 1\\(1-\frac{s^{'}}{K})\lceil\frac{K}{s^{'}+1}\rceil & ,s=1\end{cases}$	$\frac{1}{w}[1-(\frac{p}{p+w})^K]+\frac{K[p+w-(N+1)]}{(p+w-1)}$ $\approx\frac{1}{w}+\frac{K[p+w-(N+1)]}{(p+w-1)}$
Normalized online communication cost $\frac{D}{L_W}$	$s^{'}$	$\frac{N}{w}[1-(\frac{p}{p+w})^K]$ $\approx\frac{N}{w}$
Number of variables involved in a query	s	$K\frac{p}{p+w}$
Total computation	$ss^{'}$	$\frac{KN}{p+w}[1-(\frac{p}{p+w})^K]$

3.3.1 Tests & Results

We test two cases, with a small and bigger number of messages respectively.

- $\begin{cases}K: & 15\\N: & 13\end{cases}$

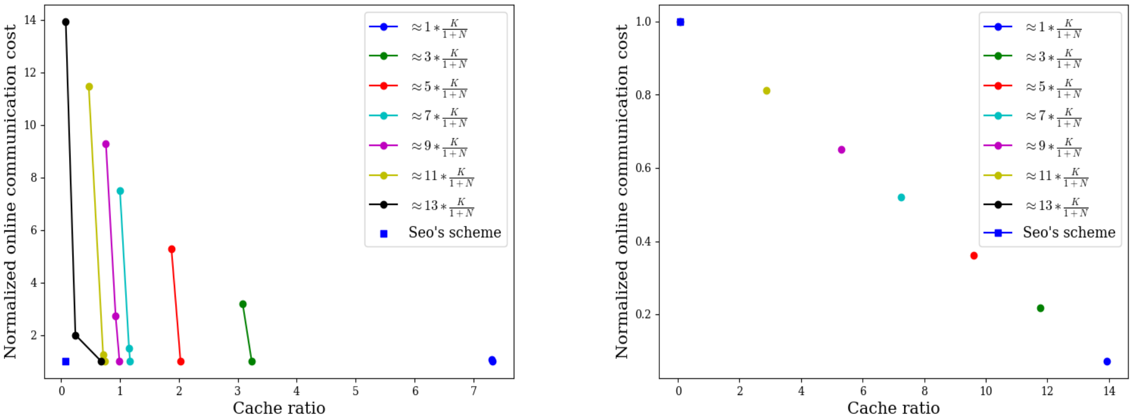

Figure 1: left: Cache-Ratio-Oriented Scheme, right: Communication-Cost-Oriented Scheme

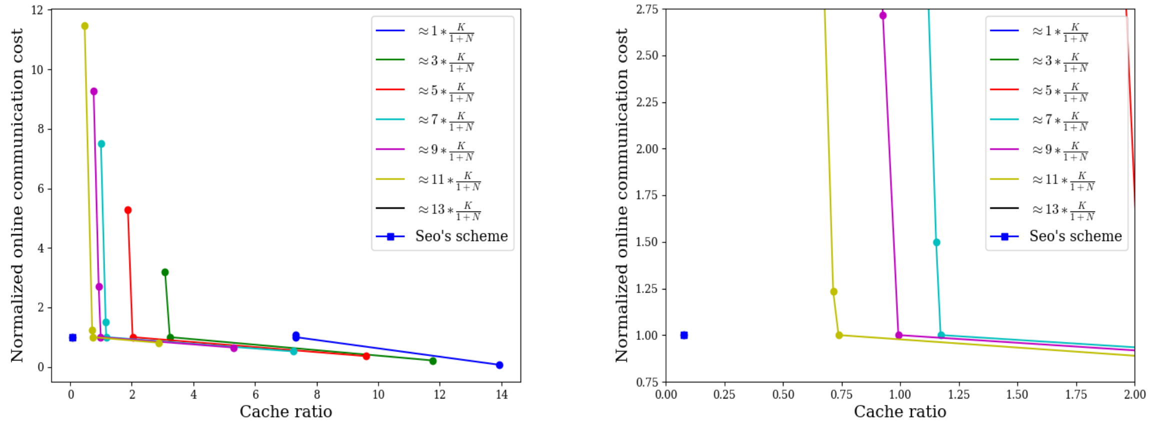

Figure 2: Combined Scheme (right: region around 0)

- $\begin{cases} K: & 1000 \\ N: & 13 \end{cases}$

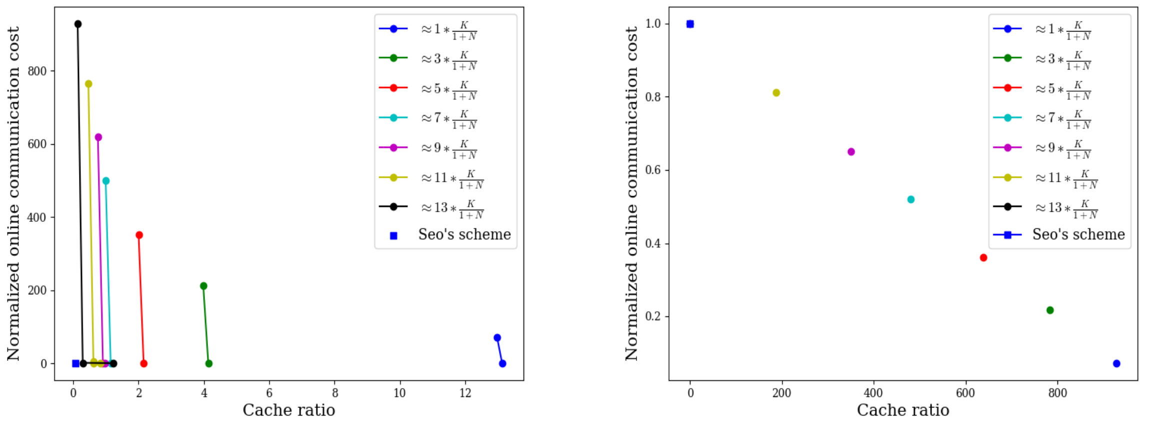

Figure 3: left: Cache-Ratio-Oriented Scheme, right: Communication-Cost-Oriented Scheme

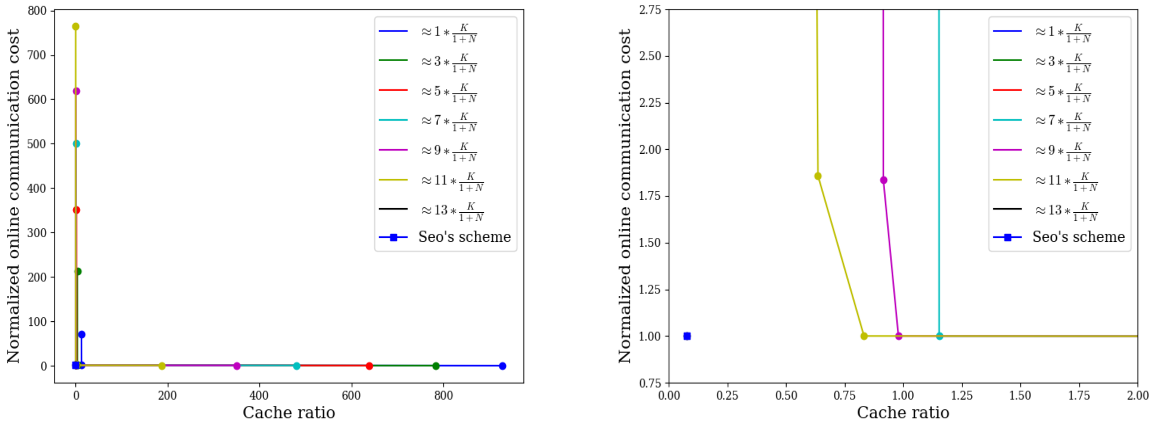

Figure 4: Combined Scheme (right: region around 0)

Analyzing the results, it becomes evident that in the cache-ratio-oriented scheme, optimal performance frequently occurs when s' equals 1, which is quite intuitive. Given the constraint imposed by the product of s and s', reducing s' effectively decreases the normalized online communication cost, while increasing the value of s does not yield a significant impact.

Nevertheless, the value of s' does come into play if the computation cost calculation is defined differently. For instance, in cases where the addition of variables is considered costly in comparison to simply obtaining a bit, these additions can constitute a crucial component of the computation cost. In such scenarios, the variation in the value of s' assumes importance, as it entails a trade-off wherein online communication cost is sacrificed in exchange for a reduction in the need for additions.

4 Privacy

4.1 Cache-Ratio-Oriented Scheme

WLOG, we assume the query sent to the n-th database $Q_n^{(\theta)}$ is as the following.

$$Q_n^{(\theta)} = q = \{\{W_{2,1}, W_{3,1}, \cdots, W_{1+s,1}\}, \{W_{2+s,1}, \cdots, W_{1+2s,1}\}, \cdots, \{W_{2+(s'-1)s,1}, \cdots, W_{1+s's,1}\}\}$$

Also, from the query construction mentioned in the above, it's obvious that

$$\Pr(Q_n^{(\theta)} = q \mid \theta = 2) = \Pr(Q_n^{(\theta)} = q \mid \theta = 3) = \cdots = \Pr(Q_n^{(\theta)} = q \mid \theta = 1 + ss^{'})$$

since a bit of these messages exists in the query. Therefore, what we have to compare is the probability that $Q_n^{(\theta)} = q$ under the condition that the bit of the desired message isn't in q. In order to guarantee privacy, the equivalence that the following equation must hold.

$$\Pr(Q_n^{(\theta)} = q \mid \theta = 1) = \Pr(Q_n^{(\theta)} = q \mid \theta = 2)$$

Furthermore, since the message bits are randomly permuted with a message and a bit of a message only exist once in an online query, we only consider the message indicies. To be more specific, we simply the query $Q_n^{(\theta)}$ to be a set of sets of message indicies.

$$Q_n^{(\theta)} = q = \{\{2, 3, \cdots, 1+s\}, \{2+s, \cdots, 1+2s\}, \cdots, \{2+(s^{'}-1)s, \cdots, 1+s^{'}s\}\}$$

We consider the equation that need to hold in the following subsections.

- $\Pr(Q_n^{(\theta)} = q \mid \theta = 1)$

  If the desired message is W_1, and it isn't in the online query, it must be the case that its information is requested in the offline phase, i.e. 1 belongs to one of the sets in the query. In the online phase, it is excluded from the set, and other sets are uniformly randomly generated from the remaining indices.

  Also, the other indices in the set that includes 1 in the offline phase are also uniformly randomly generated. Therefore, the whole procedure can be seen as we choose $ss^{'}$ indices from $K-1$ (excluding 1), and the divide them into groups.

$$\begin{aligned}
&\Pr(Q_n^{(\theta)} = q \mid \theta = 1)\\
&= (1 - \frac{ss^{'}}{K})\frac{1}{\binom{K-1}{ss^{'}} * (\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}\\
&= (\frac{K-ss^{'}}{K})\frac{(ss^{'})!(K-ss^{'}-1)!}{(K-1)!}\frac{1}{(\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}\\
&= \frac{1}{\binom{K}{ss^{'}}}\frac{1}{(\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}
\end{aligned}$$

- $\Pr(Q_n^{(\theta)} = q \mid \theta = 2)$

  On the other hand, if the desired message is W_2, and it is in the online query (suppose that it is in the set $S_{\text{sub-query}}$), it must be the case that the parity of $S_{\text{sub-query}} \setminus \{1\}$ is cached in the offline phase, where its elements are uniformly randomly chosen form the all the indicies excluding 2 in the offline phase.

  In the online phase, the elements in the other sets (sub-queries) in the online query are also uniformly randomly chosen from the remaining indices.

$$\begin{aligned}
&\Pr(Q_n^{(\theta)} = q \mid \theta = 2)\\
&= (\frac{ss^{'}}{K})\frac{1}{\binom{K-1}{s-1}\binom{K-s}{s(s^{'}-1)} * (\text{num of combinations to divide } s(s^{'}-1) \text{ indices into groups of } s \text{ indicies})}\\
&= (\frac{ss^{'}}{K})\frac{1}{\binom{K-1}{ss^{'}-1} * (\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}\\
&= \frac{ss^{'}}{K}\frac{(ss^{'}-1)!(K-ss^{'})!}{(K-1)!}\frac{1}{(\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}\\
&= \frac{1}{\binom{K}{ss^{'}}}\frac{1}{(\text{num of combinations to divide } ss^{'} \text{ indices into groups of } s \text{ indicies})}
\end{aligned}$$

Therefore, from the above derivation, we see that the equation $\Pr(Q_n^{(\theta)} = q \mid \theta = 1) = \Pr(Q_n^{(\theta)} = q \mid \theta = 2)$ holds and thus the privacy of this scheme is preserved.

4.2 Communication-Cost-Oriented Scheme

Regarding the communication-cost-oriented scheme, an examination of the online query construction reveals that each online query contains a bit from every message, whether it's a dummy bit or not. These queries exhibit a high correlation with set S, the parity of which is stored during the offline phase. To be more precise, the online query dispatched to an online database differs from set S in only one bit, specifically, a bit belonging to the desired message.

Moreover, given that the bits of undesired messages are preselected during the offline phase and are uniformly and randomly drawn from each message, we investigate whether the same randomness applies to the desired message. In other words, if a non-dummy bit of the desired message is more likely to appear in the online query, the database can infer that the desired message is more likely to originate from the messages it has observed in the query (as dummy variables won't be sent in actual implementation). Hence,

$$\Pr(Q_n^{(i)} = q \mid \theta = i) \neq \Pr(Q_n^{(j)} = q \mid \theta = j), \quad \forall i, j \in \{1, \cdots, K\}$$

Therefore, if

$$\Pr(P_{\theta,i} \text{ in an online query}) = \Pr(W_{\theta,j} \text{ in an online query}) = \frac{1}{p+w}, \quad \forall i \in \{1, \cdots, p\}, j \in \{1, \cdots, w\}$$

then

$$\Pr(Q_n^{(i)} = q \mid \theta = i) = \Pr(Q_n^{(j)} = q \mid \theta = j) = (\frac{1}{p+w})^K, \quad \forall i, j \in \{1, \cdots, K\}.$$

The privacy is preserved. We derive $\Pr(P_{\theta,i}$ in an online query), $\forall i \in \{1, \cdots, p\}$ in the following section.

- $\Pr(P_{\theta,i}$ in an online query)

$$\begin{gathered} P_{1,1}, P_{1,2}, \cdots, P_{1,p}, W_{1,1}, W_{1,2}, \cdots, W_{1,w} \\ P_{2,1}, P_{2,2}, \cdots, P_{1,p}, W_{2,1}, W_{2,2}, \cdots, W_{2,w} \\ \cdots \quad \cdots \quad \cdots \qquad\qquad \cdots \quad \cdots \\ P_{K,1}, P_{K,2}, \cdots, P_{K,p}, W_{K,1}, W_{K,2}, \cdots, W_{K,w} \end{gathered}$$

  - We first define the number of selected columns.

    X : the num of selected dummy columns

    Y : the num of selected non-dummy columns

    From the scheme, we know that $X + Y = N + 1$ and $1 \leq Y \leq N$.

  - $\Pr(P_{\theta,i}$ in an online query $\mid X = x, Y = y)$

    Suppose the offline-selected bit of the desired message is $B_{\theta,s}$ and our scheme can be categorized into the following 3 different cases.

    (1) column s is selected and $B_{\theta,s}$ is a dummy variable.

$$\Pr(P_{\theta,i} \text{ in an online query} \mid X = x, Y = y, \text{Case 1})$$

$$
\begin{aligned}
&= \Pr(\text{column } i \text{ selected}) * \Pr(P_{\theta,i} \text{ not in } S \mid \text{column } i \text{ selected}) * \frac{1}{N} \\
&= \frac{x}{p} * \frac{x-1}{x} * \frac{1}{N} \\
&= \frac{x-1}{p} * \frac{1}{N}
\end{aligned}
$$

(2) column s is selected and $B_{\theta,s}$ is a true message bit.

$$
\begin{aligned}
&\Pr(P_{\theta,i} \text{ in an online query} \mid X = x, Y = y, \text{Case 2}) \\
&= \Pr(\text{column } i \text{ selected}) * \frac{1}{N} \\
&= \frac{x}{p} * \frac{1}{N}
\end{aligned}
$$

(3) column s is not selected.

$$
\begin{aligned}
&\Pr(P_{\theta,i} \text{ in an online query} \mid X = x, Y = y, \text{Case 3}) \\
&= \Pr(\text{column } i \text{ selected}) * \Pr(P_{\theta,i} \text{ not excluded from } D \mid \text{column } i \text{ selected}) * \frac{1}{N} \\
&= \frac{x}{p} * \frac{x-1}{x} * \frac{1}{N} \\
&= \frac{x-1}{p} * \frac{1}{N}
\end{aligned}
$$

And the probability of the above 3 cases is as follows.

$\Pr(\text{Case 1} \mid X = x, Y = y)$	$\Pr(\text{Case 2} \mid X = x, Y = y)$	$\Pr(\text{Case 3} \mid X = x, Y = y)$
$\frac{x}{p+w}$	$\frac{y}{p+w}$	$1 - \frac{x+y}{p+w}$

Therefore,

$$
\begin{aligned}
&\Pr(P_{\theta,i} \text{ in an online query} \mid X = x, Y = y) \\
&= \sum_{i=1}^{3} \Pr(\text{Case } i \mid X = x, Y = y) * \Pr(P_{\theta,i} \text{ in an online query} \mid X = x, Y = y, \text{Case } i) \\
&= \frac{x}{p+w} * \frac{x-1}{p} * \frac{1}{N} + \frac{y}{p+w} * \frac{x}{p} * \frac{1}{N} + (1 - \frac{x+y}{p+w}) * \frac{x-1}{p} * \frac{1}{N} \\
&= \frac{(x-1)(p+w)+y}{Np(p+w)}
\end{aligned}
$$

- $\Pr(X = x, Y = y)$

In the offline phase of the scheme, a dummy column is uniformly randomly chosen, then the other N columns are uniformly randomly chosen from the remaining columns. Therefore, the probability of $X = x, Y = Y$ is as follows.

$$
\Pr(X = x, Y = y) = \frac{\binom{p-1}{x-1}\binom{w}{y}}{\binom{p+w-1}{N}}
$$

- $\Pr(P_{\theta,i}$ in an online query$)$

$$\begin{aligned}\Pr(P_{\theta,i} \text{ in an online query}) &= \sum_{\substack{x,y \\ x+y=N+1, x\geq 1}} \frac{\binom{p-1}{x-1}\binom{w}{y}}{\binom{p+w-1}{N}} * \frac{(x-1)(p+w)+y}{Np(p+w)} \\ &= \frac{1}{p+w}\end{aligned}$$

  We calculate it using the technique of generating function and the full derivation can be seen in Appendix A.

From this result and the fact that $\Pr(W_{\theta,j} \text{ in an online query}) = \Pr(W_{\theta,k} \text{ in an online query}) \quad \forall j,k \in \{1,\cdots,w\}$ due to the symmetry of the true message bits. we have come to the result that

$$\Pr(P_{\theta,i} \text{ in an online query}) = \Pr(W_{\theta,j} \text{ in an online query}) = \frac{1}{p+w}, \quad \forall i \in \{1,\cdots,p\}, j \in \{1,\cdots,w\}$$

Therefore, the privacy of this scheme is preserved.

5 Conclusion

In this report, we have conducted a comprehensive review of prior research addressing the problem of private information retrieval (PIR). The inception of this problem in the literature can be attributed to the pioneering work of Chor *et al.* [1]. Subsequently, Sun and Jafar [2] introduced the information-theoretic formulation of PIR, including the derivation of its capacity, which represents the maximum number of bits of desired information that can be privately retrieved per downloaded bit. Tian *et al.* [3] then presented an improved capacity-achieving scheme, which served as a significant source of inspiration for our work. In the computer science community, particularly when message size is predominantly defined as one bit, Corrigan-Gibbs and Kogan's two-phase scheme [4] aimed to reduce online computation cost through the use of cached contents in the offline phase. This scheme's framework also played a pivotal role in influencing our approach. Our generalized scheme finds its starting point in the optimal download cost derived by Seo *et al.* [6] under a specific cache size constraint, which can be viewed as a two-phase extension of the scheme proposed by Tian *et al.* [3].

Subsequently, we have formulated our problem, incorporating concepts from previous work and integrating them into a cohesive framework. Additionally, we have introduced a scheme that comprises two sub-schemes, with each sub-scheme focusing on reducing either the cache ratio or the normalized online communication cost. We have presented the performance of our scheme and elucidated the tradeoff curve between cache ratio and normalized online communication cost.

The cache-ratio-oriented scheme draws inspiration from the work of Corrigan-Gibbs and Kogan, but in a more generalized setting where we consider message sizes of arbitrary lengths rather than just one bit as discussed in their work. We have further enhanced their scheme to ensure that the likelihood of failure in our scheme is reduced to zero. We have introduced two predetermined variables, namely s and s', which are adjusted for the online computation cost constraint and the online communication cost, respectively.

On the other hand, the communication-cost oriented scheme takes cues from both Tian *et al.* and Seo *et al.* In essence, our interpretation of Seo *et al.*'s scheme involves moving the query from one of the databases in Tian *et al.*'s scheme to the offline phase. Furthermore, we have extended the number of dummy variables, initially introduced in Tian *et al.*'s scheme with a fixed value of one, to any positive integer. Consequently, we can reduce the computation cost while accepting a higher cache size

Lastly, in line with all PIR schemes, we have provided a rigorous proof demonstrating that our proposed scheme effectively preserves the defined privacy requirements.

Appendix A

To derive the term

$$\Pr(P_{\theta,i}\text{ in an online query}) = \sum_{\substack{x,y\\ x+y=N+1, 1\leq y\leq N}} \frac{\binom{p-1}{x-1}\binom{w}{y}}{\binom{p+w-1}{N}} * \frac{(x-1)(p+w)+y}{Np(p+w)}$$

we have to first the determine the possible values of x and y. Accordingly, we can discuss the term in the following two cases.

- $1 \leq p \leq N$ In this case, we track the value of x. Accordingly, $y = N + 1 - x$.

$$\begin{aligned}\Pr(P_{\theta,i}\text{ in an online query}) &= \sum_{\substack{x,y\\ x+y=N+1, 1\leq y\leq N}} \frac{\binom{p-1}{x-1}\binom{w}{y}}{\binom{p+w-1}{N}} * \frac{(x-1)(p+w)+y}{Np(p+w)} \\ &= \frac{1}{Np(p+w)\binom{p+w-1}{N}} \sum_{x=1}^{p} \binom{p-1}{x-1}\binom{w}{N+1-x}[(x-1)(p+w)+(N+1-x)] \\ &\overset{(\text{let } z=x-1)}{=} \frac{1}{Np(p+w)\binom{p+w-1}{N}} \sum_{z=0}^{p-1} \binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N]\end{aligned}$$

We specifically focus on the term $\sum_{z=0}^{p-1}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N]$.

$$\begin{aligned}&\sum_{z=0}^{p-1}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \\ &= (p-1)(p+w-1)\sum_{z=1}^{p-1}\binom{p-2}{z-1}\binom{w}{N-z} + N\sum_{z=0}^{p-1}\binom{p-1}{z}\binom{w}{N-z}\end{aligned}$$

We define generating functions to derive the term.

- Let $f_1(\alpha) = \sum_{z=1}^{p-1}\binom{p-2}{z-1}\alpha^{z-1} = (1+\alpha)^{p-2}$
- Let $g_1(\alpha) = \sum_{z=N-w}^{N}\binom{w}{N-z}\alpha^{N-z} = (1+\alpha)^{w}$

$$\begin{aligned}&f_1(\alpha)\ g_1(\alpha) = (1+\alpha)^{p+w-2} \\ &\rightarrow \sum_{z=1}^{p-1}\binom{p-2}{z-1}\binom{w}{N-z} = \binom{p+w-2}{N-1}\end{aligned}$$

- Let $k_1(\alpha) = \sum_{z=0}^{p-1} \binom{p-1}{z} \alpha^z = (1+\alpha)^{p-1}$
- Let $l_1(\alpha) = \sum_{z=N-w}^{N} \binom{w}{N-z} \alpha^{N-z} = (1+\alpha)^w$

$$\begin{aligned} & k_1(\alpha)\ l_1(\alpha) = (1+\alpha)^{p+w-1} \\ & \to \sum_{z=0}^{p-1} \binom{p-1}{z}\binom{w}{N-z} = \binom{p+w-1}{N} \end{aligned}$$

Therefore,

$$\begin{aligned} & \sum_{z=0}^{p-1} \binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \\ & = (p-1)(p+w-1)\sum_{z=1}^{p-1}\binom{p-2}{z-1}\binom{w}{N-z} + N\sum_{z=0}^{p-1}\binom{p-1}{z}\binom{w}{N-z} \\ & = (p-1)(p+w-1)\binom{p+w-2}{N-1} + N\binom{p+w-1}{N} \\ & = Np\binom{p+w-1}{N} \end{aligned}$$

so

$$\begin{aligned} \Pr(P_{\theta,i} \text{ in an online query}) &= \frac{1}{Np(p+w)\binom{p+w-1}{N}} \sum_{z=0}^{p-1}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \\ &= \frac{1}{p+w} \end{aligned}$$

- $p \geq N+1$ In this case, we track the value of y. Accordingly, $x = N+1-y$.

$$\begin{aligned} \Pr(P_{\theta,i} \text{ in an online query}) &= \sum_{\substack{x,y \\ x+y=N+1, 1\leq y\leq N}} \frac{\binom{p-1}{x-1}\binom{w}{y}}{\binom{p+w-1}{N}} * \frac{(x-1)(p+w)+y}{Np(p+w)} \\ &= \frac{1}{Np(p+w)\binom{p+w-1}{N}} \sum_{y=0}^{w}\binom{p-1}{N-y}\binom{w}{y}[(N-y)(p+w)+b] \\ &= \frac{1}{Np(p+w)\binom{p+w-1}{N}} \sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \end{aligned}$$

We specifically focus on the term $\sum_{z=N-w}^{N} \binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N]$.

$$\sum_{z=N-w}^{N} \binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N]$$

$$= \begin{cases} (p-1)(p+w-1)\sum_{z=1}^{N}\binom{p-2}{z-1}\binom{w}{N-z} + N\sum_{z=0}^{N}\binom{p-1}{z}\binom{w}{N-z} & \text{if } w = N \\ (p-1)(p+w-1)\sum_{z=N-w}^{N}\binom{p-2}{z-1}\binom{w}{N-z} + N\sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z} & \text{if } w < N \end{cases}$$

From the above derivation using generating functions, we know that

$$\begin{cases} \sum_{z=1}^{N}\binom{p-2}{z-1}\binom{w}{N-z} = \binom{p+w-2}{N-1} & \text{if } w = N \\ \sum_{z=N-w}^{N}\binom{p-2}{z-1}\binom{w}{N-z} = \binom{p+w-2}{N-1} & \text{if } w < N \end{cases}$$

$$\begin{cases} \sum_{z=0}^{N}\binom{p-1}{z}\binom{w}{N-z} = \binom{p+w-1}{N} & \text{if } w = N \\ \sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z} = \binom{p+w-1}{N} & \text{if } w < N \end{cases}$$

Therefore,

$$\begin{aligned} &\sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \\ &= (p-1)(p+w-1)\sum_{z=N-w}^{N}\binom{p-2}{z-1}\binom{w}{N-z} + N\sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z} \\ &= (p-1)(p+w-1)\binom{p+w-2}{N-1} + N\binom{p+w-1}{N} \\ &= Np\binom{p+w-1}{N} \end{aligned}$$

so

$$\begin{aligned} \Pr(P_{\theta,i} \text{ in an online query}) &= \frac{1}{Np(p+w)\binom{p+w-1}{N}}\sum_{z=N-w}^{N}\binom{p-1}{z}\binom{w}{N-z}[z(p+w-1)+N] \\ &= \frac{1}{p+w} \end{aligned}$$

References

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

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

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

[4] H. Corrigan-Gibbs and D. Kogan, "Private information retrieval with sublinear online time," Cryptology ePrint Archive, Paper 2019/1075, 2019, https://eprint.iacr.org/2019/1075. [Online]. Available: https://eprint.iacr.org/2019/1075

[5] 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.

[6] H. Seo, H. Lee, and W. Choi, "Fundamental limits of private information retrieval with unknown cache prefetching," *IEEE Transactions on Communications*, vol. 69, no. 12, pp. 8132–8144, 2021.