On the Capacity of Private Nonlinear Computation for Replicated Databases

Sarah A. Obead[†], Hsuan-Yin Lin[‡], Eirik Rosnes[‡], and Jörg Kliewer[†]
[†]Helen and John C. Hartmann Department of Electrical and Computer Engineering
New Jersey Institute of Technology, Newark, New Jersey 07102, USA
[‡]Simula UiB, N–5008 Bergen, Norway

***Abstract*—We consider the problem of private computation (PC) in a distributed storage system. In such a setting a user wishes to compute a function of f messages replicated across n noncolluding databases, while revealing no information about the desired function to the databases. We provide an information-theoretically accurate achievable PC rate, which is the ratio of the smallest desired amount of information and the total amount of downloaded information, for the scenario of nonlinear computation. For a large message size the rate equals the PC capacity, i.e., the maximum achievable PC rate, when the candidate functions are the f independent messages and one arbitrary nonlinear function of these. When the number of messages grows, the PC rate approaches an outer bound on the PC capacity. As a special case, we consider private monomial computation (PMC) and numerically compare the achievable PMC rate to the outer bound for a finite number of messages.**

I. Introduction

The problem of private information retrieval (PIR) from public databases, introduced in [1], has been the focus of attention for several decades in the computer science community (see, e.g., [2], [3]). In PIR, the goal is to privately access an arbitrary message stored in a database without revealing any information of the identity of the desired message. If the users do not have any side information on the data stored in the database, the best strategy is to store the messages in at least two databases while ensuring PIR. Hence, the design of PIR protocols has focused on the case when multiple databases, i.e., distributed storage systems (DSSs), store the messages. Recently, the aspect of minimizing the communication cost, e.g., the required rate or bandwidth of privately querying the databases with the desired requests and downloading the corresponding information has attracted a great deal of attention in the information theory and coding communities. Thus, the renewed interest in PIR primarily focused on the study and design of efficient PIR protocols for DSSs. For example, [4], [5], presented fundamental limits of the PIR rate when data is replicated over noncolluding and colluding databases, respectively.

Motivated by privacy concerns in distributed computing, a generalization of the PIR problem has emerged recently [6]–[13] to address the *private computation (PC)* of arbitrary functions over the stored messages. In PC a user intends to compute a function of the messages stored at multiple databases while keeping the identity of the function private from each database, as they may be under the control of an adversary. In [6], [7], the scenario of private linear computation (PLC) is considered for noncolluding replicated databases. In these works, the capacity and achievable rates for the communication overhead needed to privately compute a given *linear* function were derived as a function of the number of messages and the number of databases, respectively. Interestingly, the PLC capacity is equal to the PIR capacity of [4]. The extension to the coded case is addressed in [9], [10] and [11]–[13] for PLC and private polynomial computation (PPC), respectively.

In contrast to our previous work in [13] (and also [11], [12]), which considered PPC schemes for coded storage for polynomials of degree at most g, for some fixed integer g, and only a simplified rate definition, in this work we extend these considerations to general private nonlinear computation for replication-based storage and an *exact* information-theoretic definition of the PC rate. This complicates the analysis. We also include a converse result which is absent from [13]. We provide a general achievable scheme for the scenario of nonlinear computation with rate equal to the PC capacity, i.e., the maximum achievable PC rate, when the message size is large and the candidate functions are the independent messages and one arbitrary nonlinear function of these. When the number of messages grows, the PC rate approaches an outer bound on the PC capacity derived from [8, Thm. 1] and thus becomes the capacity itself. A similar result was stated in [6, Thm. 2], however for a simplified definition of the PC rate that does not take into account that the candidate functions may have different amount of information, referred to as the function size. Moreover, we discuss how a PC scheme should be designed to achieve the PC capacity. As a special case, we consider private monomial computation (PMC) and numerically compare the achievable PMC rate to the outer bound for a finite number of messages.

II. Preliminaries

A. Notation

We denote by $\mathbb{N}$ the set of all positive integers, $[a] \triangleq \{1, 2, \ldots, a\}$, and $[a : b] \triangleq \{a, a+1, \ldots, b\}$ for $a, b \in \mathbb{N}$, $a \leq b$. Random and deterministic quantities are carefully distinguished as follows. A random variable is denoted by a capital Roman letter, e.g., X, while its realization is denoted by the corresponding small Roman letter, e.g., x. Vectors are boldfaced, e.g., $\boldsymbol{X}$ denotes a random vector and $\boldsymbol{x}$ denotes

This work is supported by US NSF grant CNS-1526547.

978-1-5386-6900-6/19/$31.00 ©2019 IEEE

a deterministic vector. In addition, sets are denoted by calligraphic uppercase letters, e.g., $\mathcal{X}$. The notation $\boldsymbol{X} \sim \boldsymbol{Y}$ is used to indicate that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are identically distributed. For a given index set $\mathcal{S}$, we also write $\boldsymbol{X}^{\mathcal{S}}$ to represent $\bigl\{\boldsymbol{X}^{(v)}\colon v \in \mathcal{S}\bigr\}$. Furthermore, some constants and functions are also depicted by Greek letters or a special font, e.g., X. The function $\mathsf{H}(X)$ represents the entropy of X, and $\mathrm{I}(X\,;Y)$ the mutual information between the random variables X and Y. The binomial coefficient of a over b is denoted by $\binom{a}{b}$.

A monomial $\boldsymbol{z}^{\boldsymbol{i}}$ in m variables $z_1,\ldots,z_m$ with degree g is written as $\boldsymbol{z}^{\boldsymbol{i}} = z^{i_1}\cdots z^{i_m}$, where $\boldsymbol{i} \triangleq (i_1,\ldots,i_m) \in (\{0\}\cup\mathbb{N})^m$ is the exponent vector with $\mathsf{wt}(\boldsymbol{i}) \triangleq \sum_{j=1}^{m} i_j = g$. The set $\{\boldsymbol{z}^{\boldsymbol{i}} : \boldsymbol{i} \in (\{0\}\cup\mathbb{N})^m,\ 1 \leq \mathsf{wt}(\boldsymbol{i}) \leq g\}$ of all monomials in m variables of degree at most g has size

$$\mathsf{M}(m,g) \triangleq \sum_{h=1}^{g} \binom{h+m-1}{h} = \binom{g+m}{g} - 1.$$

B. Problem Statement

The PC problem for replicated DSSs is described as follows. We consider a DSS that stores in total f independent messages $\boldsymbol{W}^{(1)},\ldots,\boldsymbol{W}^{(f)}$, where each message $\boldsymbol{W}^{(m)} = \bigl(W_1^{(m)},\ldots,W_{\beta\mathsf{L}}^{(m)}\bigr)$, $m \in [f]$, is a random length-$\beta\mathsf{L}$ vector with independent and identically distributed symbols that are chosen at random from the field $\mathbb{F}_q$ for some $\beta,\mathsf{L} \in \mathbb{N}$. The messages are replicated and stored on the j-th database, $j \in [n]$. Without loss of generality, we assume that the symbols of each message are selected uniformly over the field $\mathbb{F}_q$. Thus,

$$\mathsf{H}\bigl(\boldsymbol{W}^{(m)}\bigr) = \beta\mathsf{L},\ \forall\, m \in [f],$$
$$\mathsf{H}\bigl(\boldsymbol{W}^{(1)},\ldots,\boldsymbol{W}^{(f)}\bigr) = f\beta\mathsf{L} \quad \text{(in q-ary units)}.$$

We consider the case of n noncolluding databases. In PC, a user wishes to privately compute exactly one function image $X_i^{(v)} \triangleq \phi^{(v)}(W_i^{(1)},\ldots,W_i^{(f)})$, $\forall\, i \in [\beta\mathsf{L}]$, out of μ arbitrary *candidate* functions $\phi^{(1)},\ldots,\phi^{(\mu)}\colon (\mathbb{F}_q)^f \to \mathbb{F}_q$, where $X_1^{(v)},\ldots,X_{\beta\mathsf{L}}^{(v)}$ are independent and identically distributed according to a prototype random variable $X^{(v)}$ with probability mass function $P_{X^{(v)}}$. Now, let $\boldsymbol{X}^{(v)} \triangleq \bigl(X_1^{(v)},\ldots,X_{\beta\mathsf{L}}^{(v)}\bigr)$. With some abuse of language, in the following, we often refer to the image $\boldsymbol{X}^{(v)}$ as the function $\phi^{(v)}$. Without loss of generality, we assume that the candidate functions are ordered descendingly with respect to their entropy, i.e., $\mathsf{H}\bigl(X^{(1)}\bigr) = \max_{v\in[\mu]} \mathsf{H}\bigl(X^{(v)}\bigr) \triangleq \mathsf{H}_{\mathsf{max}}$ and $\mathsf{H}\bigl(X^{(\mu)}\bigr) = \min_{v\in[\mu]} \mathsf{H}\bigl(X^{(v)}\bigr) \triangleq \mathsf{H}_{\mathsf{min}}$. Thus, in q-ary units, we have

$$\mathsf{H}(\boldsymbol{X}^{(v)}) = \beta\mathsf{L}\,\mathsf{H}\bigl(X^{(v)}\bigr),\ \forall\, v \in [\mu],$$
$$\mathsf{H}(\boldsymbol{X}^{(1)},\ldots,\boldsymbol{X}^{(\mu)}) = \beta\mathsf{L}\,\mathsf{H}\Bigl(X^{(1)},\ldots,X^{(\mu)}\Bigr),$$
$$\mathsf{H}(X^{(1)}) \geq \mathsf{H}(X^{(2)}) \geq \cdots \geq \mathsf{H}(X^{(\mu)}) \geq 0.$$

The user privately selects an index $v \in [\mu]$ and wishes to compute the v-th function while keeping the requested function index v private from each database. In order to retrieve the desired function $\boldsymbol{X}^{(v)}$, $v \in [\mu]$, from the DSS, the user sends a random query $Q_j^{(v)}$ to the j-th database for all $j \in [n]$. The queries are generated by the user without any prior knowledge of the realizations of the stored messages, and they are independent of the candidate functions. In other words, $\mathrm{I}\Bigl(\boldsymbol{X}^{(1)},\ldots,\boldsymbol{X}^{(\mu)}\,;Q_1^{(v)},\ldots,Q_n^{(v)}\Bigr) = 0$, $\forall\, v \in [\mu]$.

In response to the received query, the j-th database sends the answer $A_j^{(v)}$ back to the user, where $A_j^{(v)}$ is a deterministic function of $Q_j^{(v)}$ and the data stored in the database. Thus, $\mathsf{H}\bigl(A_j^{(v)} \bigm| Q_j^{(v)},\boldsymbol{W}^{[f]}\bigr) = 0$, $\forall\, v \in [\mu]$ and $\forall\, j \in [n]$.

To maintain user privacy, the query-answer function must be identically distributed for all possible function indices $v \in [\mu]$ from the perspective of each database. In other words, the scheme's queries and answer strings must be independent from the desired function index. Moreover, the user must be able to reliably decode the desired function $\boldsymbol{X}^{(v)}$ from the received database answers.

Consider a DSS with n noncolluding replicated databases storing f messages. The user wishes to retrieve the v-th function $\boldsymbol{X}^{(v)}$, $v \in [\mu]$, from the queries $Q_j^{(v)}$ and answers $A_j^{(v)}$, $j \in [n]$. For a PC protocol, the following conditions must be satisfied $\forall\, v,v' \in [\mu]$, $v \neq v'$, and $\forall\, j \in [n]$,

[Privacy]

$$(Q_j^{(v)},A_j^{(v)},\boldsymbol{X}^{[\mu]}) \sim (Q_j^{(v')},A_j^{(v')},\boldsymbol{X}^{[\mu]}),$$

[Recovery]

$$\mathsf{H}\bigl(\boldsymbol{X}^{(v)} \bigm| A_1^{(v)},\ldots,A_n^{(v)},Q_1^{(v)},\ldots,Q_n^{(v)}\bigr) = o(\mathsf{L}),$$

where any function of L, say $\lambda(\mathsf{L})$, is said to be $o(\mathsf{L})$ if $\lim_{\mathsf{L}\to\infty} \lambda(\mathsf{L})/\mathsf{L} = 0$.

To measure the efficiency of a PC protocol, we consider the required number of downloaded symbols for retrieving the $\beta\mathsf{L}$ symbols of the desired function.

Definition 1 (PC rate and capacity for replicated DSSs)**.** *The rate of a PC protocol, denoted by R, is defined as the ratio of the smallest desired function size $\beta\mathsf{L}\,\mathsf{H}_{\mathsf{min}}$ to the total required download cost D, i.e.,*[1]

$$\mathsf{R} \triangleq \frac{\beta\mathsf{L}\,\mathsf{H}_{\mathsf{min}}}{\mathsf{D}}.$$

The PC capacity*, denoted by C_{PC}, is the maximum achievable PC rate over all possible PC protocols.*

III. A Converse Bound and an Achievable Scheme

In this section, we first derive an outer bound on the PC rate of any PC protocol from [8, Thm. 1] (Theorem 1 below) and then an achievable rate for the special case of large message sizes (Theorem 2 below).

A. Converse Bound

Theorem 1. *Consider a DSS with n noncolluding replicated databases storing f messages, where the number of arbitrary candidate functions to be computed is $\mu \geq 1$. Then, the PC capacity C_{PC} is upperbounded as*

$$\mathsf{C}_{\mathsf{PC}} \leq \frac{n^{\mu}\,\mathsf{H}_{\mathsf{min}}}{\sum_{v=1}^{\mu} n^{\mu-v+1}\bigl[\mathsf{H}(X^{[v]}) - \mathsf{H}(X^{[v-1]})\bigr]}, \qquad (1)$$

[1] We adopt the rate definition of the dependent PIR (DPIR) problem [8].

where $X^{[0]}$ is the empty set and $\mathsf{H}(\emptyset)=0$.

Proof: From the converse proof of either [6] or [8], it is not difficult to see that the total download cost D of a PC protocol is lowerbounded as

$$\mathsf{D} \geq \mathsf{H}\bigl(\boldsymbol{X}^{(1)}\bigr) + \frac{\mathsf{H}\bigl(\boldsymbol{X}^{(2)} \,\big|\, \boldsymbol{X}^{(1)}\bigr)}{n} + \frac{\mathsf{H}\bigl(\boldsymbol{X}^{(3)} \,\big|\, \boldsymbol{X}^{(1)}, \boldsymbol{X}^{(2)}\bigr)}{n^2} + \cdots + \frac{1}{n^{\mu-1}}\,\mathsf{H}\bigl(\boldsymbol{X}^{(\mu)} \,\big|\, \boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(\mu-1)}\bigr),$$

from which the result follows directly from Definition 1. ■

Corollary 1. *The outer bound from* (1) *equals*

$$\mathsf{H}_{\min}\frac{1-\frac{1}{n}}{1-\left(\frac{1}{n}\right)^f} \triangleq \mathsf{H}_{\min}\mathsf{C}_{\mathsf{PIR}} \tag{2}$$

when $\mu \geq f$ and the candidate functions include the f independent messages $\boldsymbol{W}^{(1)}, \ldots, \boldsymbol{W}^{(f)}$, where $\mathsf{C}_{\mathsf{PIR}} = \frac{1-\frac{1}{n}}{1-(\frac{1}{n})^f}$ is the PIR capacity for a DSS with n noncolluding replicated databases storing f messages [4].

B. Achievability

Theorem 2. *Consider a DSS with n noncolluding replicated databases storing f messages of length $\beta\mathsf{L}$, where the number of arbitrary candidate functions to be computed is $\mu \geq 1$. Then, as $\mathsf{L} \to \infty$, the PC rate*

$$\mathsf{R} = \frac{\mathsf{H}_{\min}}{\sum\limits_{v=1}^{\mu-1} \frac{1}{n^{v-1}}\,\mathsf{H}(X^{(v)}) + \frac{1}{n^{\mu-1}}\Bigl[\mathsf{H}(X^{[\mu]}) - \sum\limits_{v=1}^{\mu-1}\mathsf{H}(X^{(v)})\Bigr]} \tag{3}$$

is achievable.

Corollary 2. *The PC rate R from* (3) *is lowerbounded as*

$$\mathsf{R} \geq \frac{\mathsf{H}_{\min}}{\mathsf{H}_{\max}}\,\frac{1-\frac{1}{n}}{1-\left(\frac{1}{n}\right)^{\mu}}.$$

Corollary 3. *Consider a DSS with n noncolluding replicated databases storing f messages of length $\beta\mathsf{L}$. Then, as $\mathsf{L}\to\infty$, the PC rate*

$$\mathsf{R} = \begin{cases} \mathsf{H}_{\min}\frac{1-\frac{1}{n}}{1-\left(\frac{1}{n}\right)^f} = \mathsf{H}_{\min}\mathsf{C}_{\mathsf{PIR}}, & \\ & \text{if } \mu = f+1, \\ \frac{\mathsf{H}_{\min}(1-\frac{1}{n})}{1-\left(\frac{1}{n}\right)^f + \left(1-\frac{1}{n}\right)\sum_{v=f+1}^{\mu-1}\mathsf{H}(X^{(v)})\left[\frac{1}{n^{v-1}}-\frac{1}{n^{\mu-1}}\right]}, & \\ & \text{if } \mu \geq f+2 \end{cases} \tag{4}$$

is achievable when the candidate functions include the f independent messages $\boldsymbol{W}^{(1)}, \ldots, \boldsymbol{W}^{(f)}$.

Remark 1.

- *For $\mu = f+1$ the PC rate from Corollary 3 equals the outer bound from Corollary 1. Thus, the proposed scheme is capacity-achieving.*
- *The PC rate from Corollary 3 and the outer bound from Corollary 1 converge to $\mathsf{H}_{\min}(1-1/n)$ as $f\to\infty$. A similar result was stated in [6, Thm. 2], however for a simplified definition of the PC rate.*
- *The rate of* (3) *extends the elementary capacity result for the case of two arbitrary correlated functions [6, Sec. VII], while the lower bound from Corollary 2 matches the lower bound on the capacity of DPIR [8, Sec. III-B].*
- *If all the μ functions are uniformly distributed, $\mathsf{H}_{\min} = \mathsf{H}_{\max}$ and we obtain the PC rate*

$$\mathsf{R} = \frac{1-\frac{1}{n}}{1-\left(\frac{1}{n}\right)^{\mu}}.$$

A PMC problem is a PC problem where the candidate functions to be computed are restricted to a subset of all possible multivariate monomials in f variables (or messages) with degree at most g which includes $\boldsymbol{W}^{(1)}, \ldots, \boldsymbol{W}^{(f)}$, where $f \leq \mu \leq \mathsf{M}(f,g)$, $g\in\mathbb{N}$. The goal here is to find a scheme that achieves the outer bound in (2). Towards this goal, we state the following remark.

Remark 2.

- *For multivariate monomials in f variables with degree at most g, it can be seen that the PMC rate*

$$\frac{1-\frac{1}{n}}{1-\left(\frac{1}{n}\right)^{\mu}} \tag{5}$$

  can be achieved via the PIR protocol from [4] by considering each candidate monomial as a virtual *message.*
- *In the case of monomials with degree at most $g=1$, $\mu = f$ (since $\mathsf{M}(f,g)=f$) and $\mathsf{H}_{\min}=\mathsf{H}_{\max}$, and the PMC rate reduces to the PIR capacity $\mathsf{C}_{\mathsf{PIR}}$.*
- *Finally, for monomials with higher degree, i.e., $g\geq 2$, we can achieve a PMC rate R strictly larger than* (5) *by Corollary 3, using a similar approach of redundancy elimination as in the schemes in [13, Sec. III-C]. Moreover, the gap between the achievable PMC rate and the outer bound from* (2) *decreases with the degree of the monomials and the number of messages (see Section V).*

C. Achievable Scheme for Theorem 2

We start with a PIR query scheme for μ *virtual* messages, where the μ arbitrary candidate functions of the PC problem are considered as μ arbitrary correlated messages. Given that μ virtual messages are replicated over n noncolluding databases, we require the length of each message to be $\beta\mathsf{L} = n^{\mu}\mathsf{L}$ with a sufficiently large L. Let $\boldsymbol{X}^{(v)} = (\boldsymbol{X}_1^{(v)}, \ldots, \boldsymbol{X}_{\beta}^{(v)})$, where each segment $\boldsymbol{X}_i^{(v)}$, $i\in[\beta]$, contains L symbols. For $\tau\in[\mu]$, a sum $\boldsymbol{X}_{i_1}^{(v_1)} + \cdots + \boldsymbol{X}_{i_\tau}^{(v_\tau)}$ of τ distinct candidate function segments is called a τ-sum for any $(i_1,\ldots,i_\tau)\in[\beta]^{\tau}$, and $\{v_1,\ldots,v_\tau\}\subseteq[\mu]$ determines the type of the τ-sum.

Here, we rely on lossless data compression of large-enough message segments to achieve the PC rate presented in Theorem 2. However, due to possible dependency across message symbols associated with the same subindex, we follow similar index assignment and message symmetry principles as for the PLC schemes in [6], [9], [10].

TABLE I

QUERY SETS FOR A DSS WITH n NONCOLLUDING REPLICATED DATABASES STORING f MESSAGES AND WHERE THE FIRST ($v = 1$) OUT OF μ CANDIDATE FUNCTIONS IS PRIVATELY COMPUTED. FOR SIMPLICITY, $\boldsymbol{U}_*^{(v)}$ INDICATES THAT THE EXACT REQUESTED SUBINDEX $t \in [\beta]$ IS OMITTED.

j	1	$\dots$	n
$Q_j^{(1)}(\mathcal{D};1)$	$\boldsymbol{U}_1^{(1)}$	$\dots$	$\boldsymbol{U}_n^{(1)}$
$Q_j^{(1)}(\mathcal{U};1)$	$\boldsymbol{U}_1^{(2)},\dots,\boldsymbol{U}_1^{(\mu)}$	$\dots$	$\boldsymbol{U}_n^{(2)},\dots,\boldsymbol{U}_n^{(\mu)}$
$Q_j^{(1)}(\mathcal{D};2)$	$\boldsymbol{U}_{n+1}^{(1)}+\boldsymbol{U}_2^{(2)}$ $\vdots$ $\boldsymbol{U}_{n+\mu-1}^{(1)}+\boldsymbol{U}_2^{(\mu)}$ $\vdots$ $\boldsymbol{U}_{n+(\mu-1)(n-1)}^{(1)}+\boldsymbol{U}_n^{(\mu)}$	$\dots$ $\vdots$ $\dots$ $\vdots$ $\dots$	$\boldsymbol{U}_{n+(\mu-1)(n-1)^2+1}^{(1)}+\boldsymbol{U}_1^{(2)}$ $\vdots$ $\boldsymbol{U}_{n+(\mu-1)(n-1)^2+(\mu-1)}^{(1)}+\boldsymbol{U}_1^{(\mu)}$ $\vdots$ $\boldsymbol{U}_{n+n(\mu-1)(n-1)}^{(1)}+\boldsymbol{U}_{n-1}^{(\mu)}$
$Q_j^{(1)}(\mathcal{U};2)$	$\boldsymbol{U}_{n+2}^{(2)}+\boldsymbol{U}_{n+1}^{(3)}$ $\vdots$ $\boldsymbol{U}_{n+(\mu-1)(n-1)}^{(\mu-1)}+\boldsymbol{U}_*^{(\mu)}$	$\dots$ $\vdots$ $\dots$	$\boldsymbol{U}_*^{(2)}+\boldsymbol{U}_{n+(\mu-1)(n-1)^2+1}^{(3)}$ $\vdots$ $\boldsymbol{U}_{n+n(\mu-1)(n-1)}^{(\mu-1)}+\boldsymbol{U}_*^{(\mu)}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$Q_j^{(1)}(\mathcal{D};\mu)$	$\boldsymbol{U}_*^{(1)}+\cdots+\boldsymbol{U}_*^{(\mu)}$ $\vdots$ $\boldsymbol{U}_*^{(1)}+\cdots+\boldsymbol{U}_*^{(\mu)}$	$\dots$ $\vdots$ $\dots$	$\boldsymbol{U}_*^{(1)}+\cdots+\boldsymbol{U}_*^{(\mu)}$ $\vdots$ $\boldsymbol{U}_{n^\mu}^{(1)}+\cdots+\boldsymbol{U}_*^{(\mu)}$

The overall protocol is composed of μ rounds. For a desired function indexed by $v \in [\mu]$, a query set $Q_j^{(v)}$, $j \in [n]$, is composed of μ disjoint subsets, one generated by each round $\tau \in [\mu]$. For each round τ the query subset is further subdivided into two subsets. The first subset $Q_j^{(v)}(\mathcal{D};\tau)$ consists of τ-sums with a single symbol from the *desired* message and $\tau - 1$ symbols from *undesired* messages, while the second subset $Q_j^{(v)}(\mathcal{U};\tau)$ contains τ-sums with symbols only from undesired messages.[2] We let π be a random permutation over the β message segments. For $v \in [\mu]$,

$$\boldsymbol{U}_t^{(v)} \triangleq \boldsymbol{X}_{\pi(t)}^{(v)}, \quad t \in [\beta],$$

denotes a permuted segment from the virtual message $\boldsymbol{X}^{(v)}$, where the permutation π is selected privately by the user and is applied as a one-time pad to all messages. Without loss of generality, let the desired virtual message be $\boldsymbol{X}^{(1)}$. The construction of the queries for arbitrary n and μ is done round-wise for each round $\tau \in [\mu]$ and each database as shown in Table I. The answer string of each database is generated as follows.

- For the first round ($\tau = 1$), optimally compress the length-L segments $\{\boldsymbol{U}_t^{(1)},\boldsymbol{U}_t^{(2)},\dots,\boldsymbol{U}_t^{(\mu)}\}$, $t \in [\beta]$, jointly, which results in $\mathsf{L}\,\mathsf{H}(X^{[\mu]}) + o(\mathsf{L})$ units.
- In the second round ($\tau = 2$), for the 2-sum $\boldsymbol{U}_t^{(v)} + \boldsymbol{U}_{t'}^{(v')}$, $\forall\, v, v' \in [\mu]$, $v < v'$, and $t, t' \in [\beta]$, compress each message segment independently based on $\max\{\mathsf{H}(X^{(v)}), \mathsf{H}(X^{(v')})\}$ and then return the sum of the two compressed segments, which results in $\mathsf{L}\max\{\mathsf{H}(X^{(v)}), \mathsf{H}(X^{(v')})\} + o(\mathsf{L})$ units. For this round, one can show that in total $(n-1)\sum_{v=1}^{\mu-1}(\mu - v)\mathsf{L}\,\mathsf{H}(X^{(v)}) + o(\mathsf{L})$ units are downloaded.
- For the following rounds ($\tau > 2$), each database compresses the segments of each queried τ-sum $\sum_{l=1}^{\tau}\boldsymbol{U}_{t_l}^{(v_l)}$, where $\{v_1,\dots,v_\tau\} \subseteq [\mu]$ and $(t_1,\dots,t_\tau) \in [\beta]^\tau$, separately based on $\max\{\mathsf{H}(X^{(v_1)}),\dots,\mathsf{H}(X^{(v_\tau)})\}$. Each database then returns the sum of the compressed segments in $\mathsf{L}\max\{\mathsf{H}(X^{(v_1)}),\dots,\mathsf{H}(X^{(v_\tau)})\} + o(\mathsf{L})$ units. By the end of each round, one can show that in total $(n-1)^{\tau-1}\sum_{v=1}^{\mu-(\tau-1)}\binom{\mu-v}{\tau-1}\mathsf{L}\,\mathsf{H}(X^{(v)}) + o(\mathsf{L})$ units are downloaded for each $\tau \in [3:\mu]$.

1) Recovery and Privacy: The scheme inherently satisfies the recovery and privacy conditions stated in Section II-B. Privacy is guaranteed by satisfying the index, message, and database symmetry principles as for the PLC schemes in [6], [9], [10]. As for the recovery, one can easily see from the PIR query structure that the user is able to obtain all β segments of the desired function based on the answers received from the n databases. Then, each segment is decoded (or optimally decompressed) to obtain in total $\beta\mathsf{L}$ symbols with a probability of decoding error that is arbitrarily close to zero for a sufficiently large L.

2) Achievable Rate: The PC rate of the scheme, assuming $\mathsf{L} \to \infty$, is given by

$$\begin{aligned}
\mathsf{R} &\overset{(a)}{=} \frac{\beta\mathsf{L}\,\mathsf{H}_{\text{min}}}{\mathsf{D}} \\
&= \frac{n^\mu \mathsf{L}\,\mathsf{H}_{\text{min}}}{n\mathsf{L}\Bigl[\mathsf{H}(X^{[\mu]}) + \sum\limits_{\tau=2}^{\mu}(n-1)^{\tau-1}\sum\limits_{v=1}^{\mu-(\tau-1)}\binom{\mu-v}{\tau-1}\mathsf{H}(X^{(v)})\Bigr]} \\
&= \frac{n^\mu \mathsf{H}_{\text{min}}}{n\Bigl[\mathsf{H}(X^{[\mu]}) + \sum\limits_{\tau=2}^{\mu}(n-1)^{\tau-1}\sum\limits_{v=1}^{\mu-(\tau-1)}\binom{\mu-v}{\tau-1}\mathsf{H}(X^{(v)})\Bigr]} \qquad (6)\\
&\overset{(b)}{=} \frac{n^{\mu-1}\mathsf{H}_{\text{min}}}{\mathsf{H}(X^{[\mu]}) + \sum\limits_{v=1}^{\mu-1}\sum\limits_{\tau=2}^{\mu-(v-1)}(n-1)^{\tau-1}\binom{\mu-v}{\tau-1}\mathsf{H}(X^{(v)})} \\
&\overset{(c)}{=} \frac{n^{\mu-1}\mathsf{H}_{\text{min}}}{\mathsf{H}(X^{[\mu]}) + \sum\limits_{v=1}^{\mu-1}\mathsf{H}(X^{(v)})\sum\limits_{\tau'=1}^{\mu-v}\binom{\mu-v}{\tau'}(n-1)^{\tau'}} \\
&\overset{(d)}{=} \frac{n^{\mu-1}\mathsf{H}_{\text{min}}}{\mathsf{H}(X^{[\mu]}) + \sum\limits_{v=1}^{\mu-1}\mathsf{H}(X^{(v)})(n^{\mu-v}-1)} \\
&= \frac{\mathsf{H}_{\text{min}}}{\sum\limits_{v=1}^{\mu-1}\frac{1}{n^{v-1}}\mathsf{H}(X^{(v)}) + \frac{1}{n^{\mu-1}}\Bigl[\mathsf{H}(X^{[\mu]}) - \sum\limits_{v=1}^{\mu-1}\mathsf{H}(X^{(v)})\Bigr]},
\end{aligned}$$

where (a) follows from Definition 1, (b) follows from changing the order of the two summations, (c) results by defining $\tau' = \tau - 1$ of the second summation term, and (d) follows from the binomial identity.

For the scenario of Corollary 3, by a similar approach of redundancy elimination as in the schemes in [13, Sec. III-C], the PC scheme above can be modified by removing the

[2] With some abuse of notation, the generated queries are sets containing their answers.

redundant 1-sums. Using [13, Lem. 1] and $\mathsf{H}(X^{(v)}) = \mathsf{H}_{\mathsf{max}} = 1,\ \forall\, v \in [f]$, the PC rate can be shown to be equal to (4).

IV. Discussion of the Outer Bound of Theorem 1

By expanding the denominator of (1), denoted by $\mathsf{D}_{\mathsf{opt}}$, we get

$$\begin{aligned}\mathsf{D}_{\mathsf{opt}} &= \sum_{v=1}^{\mu} n^{\mu-v+1}\bigl[\mathsf{H}(X^{[v]}) - \mathsf{H}(X^{[v-1]})\bigr]\\ &= n\,\mathsf{H}\bigl(X^{[\mu]}\bigr) + n(n-1)\,\mathsf{H}\bigl(X^{[\mu-1]}\bigr)\\ &\quad + n(n-1)\cdot n\,\mathsf{H}\bigl(X^{[\mu-2]}\bigr) + \cdots\\ &\quad + n(n-1)\cdot n^{\mu-2}\,\mathsf{H}(X^{(1)}).\end{aligned}$$

Next, consider the total download cost of the achievable scheme for Theorem 2 divided by L, i.e., the denominator of (6), and denote it by D_1. We have

$$\begin{aligned}\mathsf{D}_1 &= n\,\mathsf{H}(X^{[\mu]}) + \sum_{\tau=2}^{\mu} n(n-1)^{\tau-1} \sum_{v=1}^{\mu-(\tau-1)} \tbinom{\mu-v}{\tau-1}\,\mathsf{H}(X^{(v)})\\ &= n\,\mathsf{H}(X^{[\mu]}) + n(n-1)\sum_{v=1}^{\mu-1} \tbinom{\mu-v}{1}\,\mathsf{H}(X^{(v)})\\ &\quad + n(n-1)\sum_{v=1}^{\mu-2}(n-1)\tbinom{\mu-v}{2}\,\mathsf{H}(X^{(v)}) + \cdots\\ &\quad + n(n-1)\cdot(n-1)^{\mu-2}\,\mathsf{H}(X^{(1)}).\end{aligned}$$

By comparing $\mathsf{D}_{\mathsf{opt}}$ with D_1, it can be seen that because joint compression of the virtual message segments is not utilized, the outer bound of Theorem 1 is not achieved. An open question is to design an optimal scheme that achieves a download cost of $\mathsf{D}_{\mathsf{opt}}$.

V. Special Case: Private Monomial Computation

In this section, we consider the special case of PMC. One can easily see that the assumption of Corollary 3 covers the scenario of PMC, which includes the f independent messages as candidate functions. Hence, as $\mathsf{L} \to \infty$, the rate in (4) is achievable for PMC.

In Fig. 1, for the field $\mathbb{F}_3$ and $n = 3$ and 5, we plot the PMC rate computed from (4) and the outer bound from (2) as a function of the number of messages f for $\mu = \widetilde{\mathsf{M}}(f, g)$ with $g = 2$ and $g = 3$, where $\widetilde{\mathsf{M}}(f, g)$ denotes the number of *nonparallel* monomials [13, Sec. III-E]. Note that the PMC rate is close to the outer bound even for a small number of messages. As $f \to \infty$, it follows from Remark 1 that the PMC rate approaches $\mathsf{H}_{\mathsf{min}}(1 - 1/n)$.

VI. Conclusion

We presented a novel PC scheme for noncolluding replicated databases and the scenario of nonlinear computation and showed that the resulting PC rate equals the PC capacity as the message size grows for the case when the candidate functions are the independent messages and one arbitrary nonlinear function of these. Moreover, the PC rate approaches an outer bound on the PC capacity and thus becomes the capacity itself when the number of messages grows. Finally, we compared the outer bound and the achievable rate for the special case of PMC.

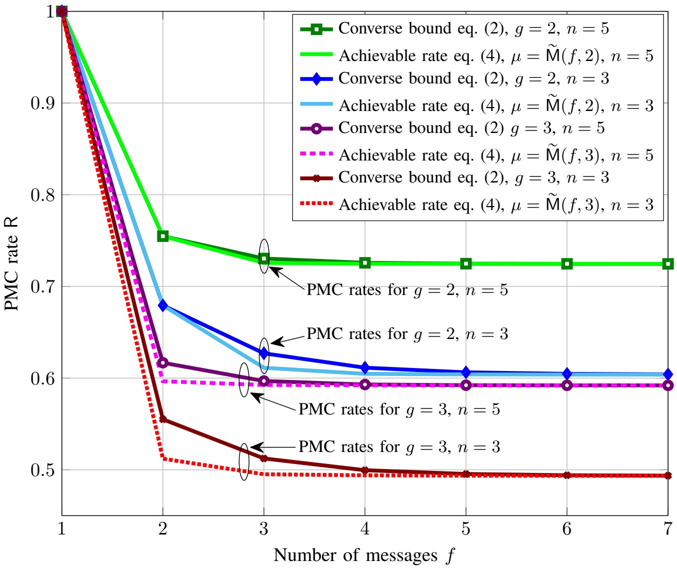

Fig. 1. PMC rate R versus the number of messages f for the retrieval of nonparallel monomials over the field $\mathbb{F}_3$.

References

[1] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," in *Proc. 36th Annu. IEEE Symp. Found. Comp. Sci. (FOCS)*, Milwaukee, WI, USA, Oct. 23–25, 1995, pp. 41–50.

[2] W. Gasarch, "A survey on private information retrieval," *Bull. Eur. Assoc. Theor. Comput. Sci. (EATCS)*, vol. 82, pp. 72–107, Feb. 2004.

[3] S. Yekhanin, "Private information retrieval," *Commun. ACM*, vol. 53, no. 4, pp. 68–73, Apr. 2010.

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

[5] ——, "The capacity of robust private information retrieval with colluding databases," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2361–2370, Apr. 2018.

[6] ——, "The capacity of private computation," *IEEE Trans. Inf. Theory*, vol. 65, no. 6, pp. 3880–3897, Jun. 2019.

[7] M. Mirmohseni and M. A. Maddah-Ali, "Private function retrieval," in *Proc. Iran Workshop Commun. Inf. Theory (IWCIT)*, Tehran, Iran, Apr. 25–26, 2018, pp. 1–6.

[8] Z. Chen, Z. Wang, and S. Jafar, "The asymptotic capacity of private search," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Vail, CO, USA, Jun. 17–22, 2018, pp. 2122–2126.

[9] S. A. Obead and J. Kliewer, "Achievable rate of private function retrieval from MDS coded databases," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Vail, CO, USA, Jun. 17–22, 2018, pp. 2117–2121.

[10] S. A. Obead, H.-Y. Lin, E. Rosnes, and J. Kliewer, "Capacity of private linear computation for coded databases," in *Proc. 56th Allerton Conf. Commun., Control, Comput.*, Monticello, IL, USA, Oct. 2–5, 2018.

[11] D. Karpuk, "Private computation of systematically encoded data with colluding servers," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Vail, CO, USA, Jun. 17–22, 2018, pp. 2112–2116.

[12] N. Raviv and D. A. Karpuk, "Private polynomial computation from Lagrange encoding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France, Jul. 7–12, 2019.

[13] S. A. Obead, H.-Y. Lin, E. Rosnes, and J. Kliewer, "Private polynomial computation for noncolluding coded databases," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France, Jul. 7–12, 2019.