ON-OFF Privacy Against Correlation Over Time

Fangwei Ye, *Member, IEEE*, Carolina Naim, *Graduate Student Member, IEEE*,
and Salim El Rouayheb, *Member, IEEE*

***Abstract*— We consider the problem of ON-OFF privacy in which a user is interested in the latest message generated by one of *n* sources available at a server. The user has the choice to turn privacy ON or OFF depending on whether he wants to hide his interest at the time or not. The challenge of allowing the privacy to be toggled between ON and OFF is that the user's online behavior is correlated over time. Therefore, the user cannot simply ignore the privacy requirement when privacy is OFF. We represent the user's correlated requests by an *n*-state Markov chain. Our goal is to design ON-OFF privacy schemes with optimal download rate that ensure privacy for past and future requests. We devise a polynomial-time algorithm to construct an ON-OFF privacy scheme. Moreover, we present an upper bound on the achievable rate. We show that the proposed scheme is optimal and the upper bound is tight for some special families of Markov chains. We also give an implicit characterization of the optimal achievable rate as a linear programming (LP).**

***Index Terms*— Information-theoretic privacy, private information retrieval, Markov chains.**

I. Introduction

A. Motivation

In THE current data-driven world, users' information is always being collected online, and its privacy has become a significant concern. Many users wish to keep private their personal information, such as their age, sex, political views, health disorders, etc. Significant research has been devoted to study algorithms that preserve users' privacy. Some of the proposed approaches include applying anonymization techniques [1], differential privacy algorithms [2], and private information retrieval methods [3].

Privacy, however, comes at a cost. Privacy-preserving algorithms typically incur higher overheads in terms of computation, memory, and delay. These incurred costs motivate one to think of privacy as an expensive commodity and, therefore, to allow the user to request it, *i.e.,* turn privacy ON, only when needed; otherwise, turn it OFF. The user may choose to switch between privacy being ON and OFF depending on several criteria, such as location (country, workplace vs. home, etc.), network connection (public or private network), devices (shared vs. personal machines) being used, or service quality (privacy-preserving algorithms typically induce more overheads), to name a few.

At a conceptual level, ON-OFF privacy algorithms enable privacy to be switched between ON and OFF whenever desired. One of the main challenges in designing such algorithms is correlation. For instance, a user's online behavior is personal which creates correlation over time. That is, by monitoring the user's behavior when his privacy is OFF, one may learn about the user's behavior when his privacy was ON. Therefore, the user cannot simply ignore the privacy requirement when privacy is OFF.

Take for example a user who is subscribed to two political online video channels, one is pro-right, and the other is pro-left. The user is interested in watching the latest videos posted by one of these channels. Correlation over time here is due to the fact that a typical user is more likely to keep watching videos from the same channel. One may think of a scenario where the user is more likely to watch the top item in his recommended list that depends on the previously watched videos. Therefore, when the user switches his privacy from ON to OFF, the user cannot openly request the video he is interested in because this leaks information on what he was watching right before (when privacy was ON).

In this work, we abstract the previous example into the information retrieval setting, i.e., downloading messages from a server. The user may choose to turn privacy ON or OFF at each instant. When privacy is ON, the user wants to completely hide, in an information-theoretic sense, his interest from the server. Otherwise, when privacy is OFF, the user does not worry about the privacy of his interest at that particular instant. Nevertheless, he must be careful not to leak information about his previous or future interests that he wants to keep private. Our objective is to construct ON-OFF privacy schemes that:

1) Deliver to the user his request while ensuring perfect information-theoretic privacy against the server. That is, the observations of the server must be statistically independent of the user's interests when Privacy is ON.
2) Maximize the download rate or equivalently minimize the amount of downloaded information.

B. Related Work

The study of information-theoretic measures for privacy has received significant interest in the literature (see for e.g. [4]–[8]). The closest problem to the ON-OFF privacy problem studied in this paper is the private information retrieval (PIR) from a single server [3], which can be viewed as a special

Manuscript received July 28, 2020; revised November 10, 2020 and December 27, 2020; accepted December 27, 2020. Date of publication January 8, 2021; date of current version February 1, 2021. This work was supported by NSF under Grant CCF 1817635. This article was presented in part at the IEEE Information Theory Workshop (ITW), Visby, Gotland, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Aris Gkoulalas Divanis. *(Corresponding author: Fangwei Ye.)*

The authors are with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey, Piscataway, NJ 08854 USA (e-mail: fangwei.ye@rutgers.edu; carolina.naim@rutgers.edu; salim.elrouayheb@rutgers.edu).

Digital Object Identifier 10.1109/TIFS.2021.3050068

case (when privacy is always ON) of the ON-OFF privacy problem. In this case, it is known that to achieve information-theoretic privacy, the user must download all the messages, except in the case when the user has some side information [9], [10]. Recently, there has been significant progress on PIR with multiple servers with a focus on download rate and coded data (*e.g.,* [11]–[15] and references therein).

A related problem that considers privacy with correlation, namely location privacy, was studied in [16]–[24]. The privacy notions studied therein include k-anonymity [16], [17], (extended) differential privacy [18]–[20], and distortion privacy [21], [22], which all differ from the information-theoretic privacy measure studied in this paper. The works of [23], [24] recently studied the information-theoretic privacy measure in location-privacy protection mechanisms, and their privacy metric was defined by the mutual information between the released data and the true traces. In this paper's language, it can be viewed as the case when privacy is always ON. However, in this paper, we want to prevent the adversary from inferring a selective part of the requests specified by an ON or OFF privacy status, and the simple time-sharing (switching between a private and a non-private scheme according to the privacy status) approach is not permissible due to the correlation.

The ON-OFF privacy problem was studied by the authors first in [25]. The focus was on preserving the privacy of past requests for which privacy was ON. This paper is based on the setting studied later in [26] which requires privacy of both past and future requests. The work in [26] studied the special case when $n = 2$ sources, and an optimal scheme and a tight upper bound on the rate were presented therein. The concept of ON-OFF privacy was also applied to preserve privacy of sensitive genotypes in genomics in [27].

C. Contributions

To study how correlation affects privacy, we focus in this paper on the simplest non-trivial correlation model given by a Markov chain. That is, we assume that the user's requests to the server are correlated in time according to a Markov chain. We also assume that the user knows his future requests into a window of size ω[1].

Under this model, our main result is summarized in Theorem 1 which: (i) gives a general upper bound on the download rate; and (ii) gives an achievable rate obtained by an ON-OFF privacy scheme having polynomial time complexity in n.

We show that our proposed scheme is optimal, *i.e.,* the upper bound is tight, for a family of Markov chains for $n > 2$.

For $n = 2$ sources, this scheme is equivalent to the one in [26] and therefore is always optimal. Therefore, the results in this paper can be viewed as a generalization of the earlier results in [26] on $n = 2$ sources to any $n \geq 2$ sources.

We also give an implicit characterization of the optimal achievable rate, which relies on solving a linear program (LP) with an exponential number (in n) of variables and constraints. Thus, it is intractable to tackle it using standard LP solvers (*e.g.,* [29]). From that perspective, our results can be viewed as leveraging the special structure of the problem to provide an efficiently computable upper bound and a polynomial time scheme.

D. Organization

The rest of the paper is organized as follows. In Section II, we describe the formulation of the ON-OFF privacy problem. We present our main result, Theorem 1, in Section III, and its corollaries in Section IV. In Sections V and VI, we propose an efficient ON-OFF privacy scheme that gives the achievable bound in Theorem 1. In Section VII, we derive the upper bound, in Theorem 1, on the achievable rate. Finally, we present an implicit characterization of the optimal achievable rate in Section VIII. We conclude in Section IX.

II. Problem Formulation

A. System Model

A single server stores n information sources $\{\mathscr{W}_i : i \in \mathcal{N}\}$, where $\mathcal{N} := \{1, 2, \ldots, n\}$. The system is time-varying, and the time index t is assumed to be discrete, *i.e.,* $t \in \mathbb{N}$, throughout this paper. At each time t, each source $\mathscr{W}_i$ generates a new message $W_{i,t}$ of length L, which is independent of previously generated messages $\left\{W_{i,j} : j = 0, \ldots, t-1\right\}$. Without loss of generality, we assume that $W_{i,t}$ for $i \in \mathcal{N}$ and $t \in \mathbb{N}$ are independently and identically drawn from the uniform distribution over $\{0, 1\}^L$.

At time t, the user is interested in retrieving the latest message generated by a desired source, *i.e.,* one of the messages from $\{W_{i,t} : i \in \mathcal{N}\}$. In particular, let X_t be the source of interest at time t, which takes values in $\mathcal{N}$. In the sequel, we will call X_t the *user's request* at time t. For notational simplicity, we drop t from $W_{i,t}$ when the time index t is clear from context, *i.e.,* $W_{i,t}$ will be denoted by W_i. To retrieve the desired message, the user is allowed to construct a query Q_t and send this query to the server. Upon receiving the query, the server responds to the user by producing an answer A_t. After receiving the answer, the user should be able to recover the message W_{X_t} that he is interested in.

Meanwhile, the user may wish to hide the identity of his source of interest at time t. Specifically, the user may choose the *privacy status* F_t to be ON or OFF. When F_t is ON, the user wishes to keep X_t private and when F_t is OFF, the user is not concerned with hiding X_t. We assume that the privacy status $\{F_t : t \in \mathbb{N}\}$ is independent of the user's requests, and the user's privacy status $\{F_i : i \leq t\}$ is known and recorded by both the server and the user at time t.

We assume in our model that the privacy status is independent of the user's requests because as mentioned in Section I-A, the user may choose privacy to be ON or OFF depending on many factors such as location, network connection, devices or service quality etc, and in general these factors are independent of the user's requests.

In this paper, we are particularly interested in the case where the requests X_t form a Markov chain, *i.e.,* $\{X_t : t \in \mathbb{N}\}$ is generated by a (discrete) Markov source. The transition matrix

[1] This can happen in applications where the user places his requests in a queue of size ω. For example, the user may know what he will be watching next since it is the next item in a playlist or the top recommendation in a recommended list.

P of the Markov chain is known by both the server and the user, and the transition probability from state i to state j is denoted by $P_{i,j}$.

Moreover, we assume that the user knows his future requests in a window of positive size ω[2], which means that at time t, the user knows the future requests $\{X_{t+1}, \ldots, X_{t+\omega}\}$ in addition to the current and all past requests $\{X_0, \ldots, X_t\}$. This models several scenarios where user's requests are in a queue. One can think of the situation where the user places his requests in a playlist when watching videos.

The system mainly consists of two encoding functions, which we describe below. Let $[t]$ denote $\{0, 1, \ldots, t\}$ and $X_{[t]}$ denote $\{X_0, X_1, \ldots, X_t\}$ for $t \in \mathbb{N}$ in the sequel.

1) Query Encoding Function: The query Q_t, at time t, is generated by a query encoding function ϕ_t. Given the assumptions that the messages $W_{i,t}$ (as well as the answers A_t) are independent over time and the privacy status $F_{[t]}$ are known by both the user and the server, we suppose that ϕ_t is a probabilistic function of the user's known requests[3] $X_{[t+\omega]}$ for some $\omega \in \mathbb{N}^+$, *i.e.,*

$$Q_t = \phi_t\left(X_{[t+\omega]}, \mathsf{K}\right), \tag{1}$$

where K is the random key to generate a probabilistic query.

2) Answer Encoding Function: Accordingly, the answer A_t from the server is given by the answer encoding function ρ_t, which is assumed to be a deterministic function of the query Q_t and the latest messages, *i.e.,*

$$A_t = \rho_t\left(Q_t, W_1, \ldots, W_n\right). \tag{2}$$

In particular, the length of answer A_t is assumed to be a function of the query Q_t, and we denote this length by $\ell(Q_t)$. Then, the average length of the answer A_t is given by

$$\ell_t = \mathbb{E}_{Q_t}[\ell\left(Q_t\right)], \tag{3}$$

where $\mathbb{E}[\cdot]$ is the expectation operator.

After receiving the answer A_t, the user should be able to recover the desired message from the answer with zero-error probability. This is referred to as the *decodability* condition.

B. Adversary Model

The adversary is the untrusted server and is assumed to have full statistical knowledge of user's requests and the querying mechanism, that is, the Markov chain's transition probabilities modeling the user's requests and the querying mechanism that generates the queries for information retrieval, respectively.

We assume that the server has no memory constraint, so the server can use all the queries it received up to time t, represented by $Q_{[t]}$, and the statistical knowledge of user's requests and the querying mechanism, to infer user's private requests, *i.e.,* all previous requests of which privacy was ON and all future requests. We also assume that the adversary has unbounded computational power and can launch any attack to infer any of the user's private requests.

Privacy is quantified by the mutual information between the user's private requests and the queries released to the server. It is worth noting that the information-theoretic privacy measure is preferable in this paper, since it is independent of specific attacking strategies. We consider the most stringent privacy constraint, namely information-theoretic perfect privacy, which requires that absolutely zero information, measured by the mutual information, about the user's private requests is leaked to the server. Formally, it can be written as

$$I\left(X_{\mathcal{B}_t}; Q_{[t]}\right) = 0, \quad \forall t \in \mathbb{N}, \tag{4}$$

where $\mathcal{B}_t := \{i : i \le t, F_i = \mathrm{ON}\} \cup \{i : i \ge t+1\}$ denotes the user's private requests, *i.e.,* all previous requests of which privacy was ON and all future requests, and $I(\cdot)$ denotes the mutual information. We refer to (4) as the *privacy* condition.

Remark 1: The privacy requirement in (4) *implies that at time t, only the previous privacy status $\{F_i : i \le t\}$ is known, and the user may not know whether he will choose privacy to be ON or OFF in the future. For this reason, we have adopted a worst-case formulation in the privacy constraint by assuming that privacy is always ON in the future. In other words, at time t, all previous requests when privacy was ON, as well as all future requests need to be protected. This is characterized by the set $\mathcal{B}_t := \{i : i \le t, F_i = ON\} \cup \{i : i \ge t+1\}$ in* (4).

For large messages, the upload cost is negligible relative to the download cost, so in this paper, we are interested in minimizing the download cost of the answer at each time, *i.e.,* the average length ℓ_t at time t. By convention, we measure the efficiency by the download rate $R_t = L/\ell_t$, and define the achievable rate region as follows.

Definition 1 (Achievable Rate): The rate tuple $(R_t : t \in \mathbb{N})$ is achievable if there exists a scheme with average download cost ℓ_t such that $R_t \le L/\ell_t$.

In the rest of this paper, we will study the achievable region of $(R_t : t \in \mathbb{N})$. In particular, the focus of this paper is the characterization of R_t for each $t \in \mathbb{N}$.

III. Main Result

Before stating the main result, we introduce some necessary notation. Let $\tau(t)$ be the last time privacy was ON, *i.e.,*

$$\tau(t) := \max\{i : i \le t, F_i = \mathrm{ON}\}. \tag{5}$$

Without loss of generality, we assume that $F_0 = \mathrm{ON}$, so $\tau(t)$ is always well-defined. Also, when the time index t is clear from context, we drop t from the notation and write $\tau(t)$ as τ for simplicity. For our analysis, it is convenient to define

$$U_t := \left(X_\tau, X_{t+1}\right), \tag{6}$$

which represents the last request when privacy was ON and the next request of the user at time t, so the alphabet size of U_t is $\mathcal{N}^2$.

Stating our main results calls for the following notation, which we summarize in Figure 1. The inherent value of this notation will be apparent when we give the proofs of our main

[2] If the window size $\omega = 0$, *i.e.,* no future requests are known, we have to relax the the stringent privacy requirement (4) defined in this work to a weaker sense where only past requests are protected. This falls into a different model studied in [28].

[3] One may also take all previous queries $Q_{[t-1]}$ as variables of the function. However, since $Q_{[t-1]}$ is also a probabilistic function of $X_{[t+\omega]}$, the variables of the function can be written as in (1).

$$\begin{array}{c|cccc|} & 1 & 2 & \cdots & n \\ u_{x,1} & p(1|u_{1,1}) & p(2|u_{2,1}) & \cdots & p(n|u_{n,1}) \\ u_{x,2} & p(1|u_{1,2}) & p(2|u_{2,2}) & \cdots & p(n|u_{n,2}) \\ \vdots & \vdots & \ddots & & \vdots \\ u_{x,m} & p(1|u_{1,m}) & p(2|u_{2,m}) & \cdots & p(n|u_{n,m}) \end{array}$$

Fig. 1. For a given x, sort the probabilities $p\left(X_t = x|U_t = u\right)$ for $u \in \mathcal{N}^2$ in an ascending order, and store the values in column x where $m = n^2$. λ_i is the sum of the i^{th} row.

results in later sections. For any given $x \in \mathcal{N}$, we can order the likelihood probabilities $p\left(X_t = x|U_t = u_{x,i}\right)$ such that

$$p\left(X_t = x|U_t = u_{x,1}\right) \leq p\left(X_t = x|U_t = u_{x,2}\right) \leq \cdots \leq p\left(X_t = x|U_t = u_{x,m}\right), \quad (7)$$

where $m = n^2$ and $u_{x,i}$ for $i = 1, \ldots, m$ are distinct elements in $\mathcal{N}^2$. Note that probabilities $p\left(X_t = x|U_t = u\right)$ for $x \in \mathcal{N}$ and $u \in \mathcal{N}^2$ can be determined by the given Markov chain. These ordered probabilities can be stored in the columns of a matrix, as shown in Figure 1. Then, for $x \in \mathcal{N}$ and $i = 1, \ldots, m$, let $\lambda_i(t)$ be the summation of row i of this matrix, more formally,

$$\lambda_i\left(t\right) = \sum_{x \in \mathcal{N}} p\left(X_t = x|U_t = u_{x,i}\right). \quad (8)$$

Also, for $i = 1, \ldots, n$, let

$$\theta_i(t) = \begin{cases} \lambda_i(t) - \lambda_{i-1}(t), & i < n, \\ 1 - \lambda_{i-1}(t), & i = n, \end{cases} \quad (9)$$

where $\lambda_0\left(t\right)$ is assumed to be 0. For notational simplicity, let

$$\frac{1}{R_t^I} := \sum_{i=1}^{n} i\, \theta_i(t), \quad (10)$$

and

$$\frac{1}{R_t^O} := \lambda_m(t) = \sum_{x \in \mathcal{N}} \max_{u \in \mathcal{N}^2} p(X_t = x|U_t = u), \quad (11)$$

where $\lambda_m(t)$ is defined in (8). We may drop the time index t when it is clear from context, that is, we will write $\lambda_i\left(t\right)$ and $\theta_i\left(t\right)$ as λ_i and θ_i, respectively. With this notation, we are ready to state the main theorem.

Theorem 1: Suppose that $\{X_t : t \in \mathbb{N}\}$ *is a Markov process with the transition matrix* P. *The rate tuple* $(R_t : t \in \mathbb{N})$ *is achievable if*

$$\frac{1}{R_t} \geq \frac{1}{R_t^I}, \quad (12)$$

where $1/R_t^I$ *is defined in* (10). *On the other hand, any achievable rate tuple* $(R_t : t \in \mathbb{N})$ *must satisfy*

$$\frac{1}{R_t} \geq \frac{1}{R_t^O}, \quad (13)$$

where $1/R_t^O$ *is defined in* (11).

Remark 2 (Single Server PIR): As mentioned earlier, the single server private information retrieval problem can be viewed as a special case of this setting where privacy is always ON. As a sanity check, if $F_t = ON$ *for all* $t \in \mathbb{N}$, *we have* $U_t = (X_t, X_{t+1})$ *by the definition* (6), *and then we can easily see that* $\max_{u_t} p\left(x_t|u_t\right) = 1$, *for all* $x_t \in \mathcal{N}$. *Thus, we know from Theorem 1 that* R_t *is achievable only if*

$$R_t \leq R_t^O = \frac{1}{\lambda_m(t)} = \frac{1}{n},$$

which implies that it is necessary to download all messages when the privacy is ON. This is consistent with the well- known result in the literature on PIR [3].

The rest of the paper is dedicated to proving Theorem 1. In particular, we propose a polynomial-time querying scheme that achieves R_t^I in sections V and VI. As discussed in the previous remark, the user has to query for all the messages when privacy is ON, so our focus will be on the instances when privacy is OFF. Roughly speaking, in our proposed probabilistic querying scheme, the user asks for a subset of the messages containing the message in which he is interested. The user generates his query based on his knowledge of his previous requests when privacy was ON, his current request, and his next request. Moreover, the proof of the upper bound R_t^O will be presented in Section VII.

IV. Optimality for Special Families of Markov Chains

Before we proceed to prove Theorem 1, we give two corollaries that characterize two special classes of Markov chains for which the bounds in Theorem 1 are tight, *i.e.,* $R_t^I = R_t^O$, which means that our proposed scheme is optimal for these special cases.

Corollary 1 (Optimality for $n = 2$*): For the case* $n = 2$, *the two bounds* (12) *and* (13) *match, i.e.,*

$$\frac{1}{R_t^I} = \frac{1}{R_t^O} = \lambda_m(t). \quad (14)$$

In other words, the rate tuple $(R_t : t \in \mathbb{N})$ *is achievable if and only if*

$$R_t \leq \frac{1}{\lambda_m(t)}. \quad (15)$$

Definition 2 (Symmetric Markov Chain): A Markov chain is symmetric if its transition matrix P *is given by*

$$P_{i,j} = \begin{cases} \alpha, & \text{if } i = j, \\ \dfrac{1-\alpha}{n-1}, & \text{if } i \neq j, \end{cases} \quad (16)$$

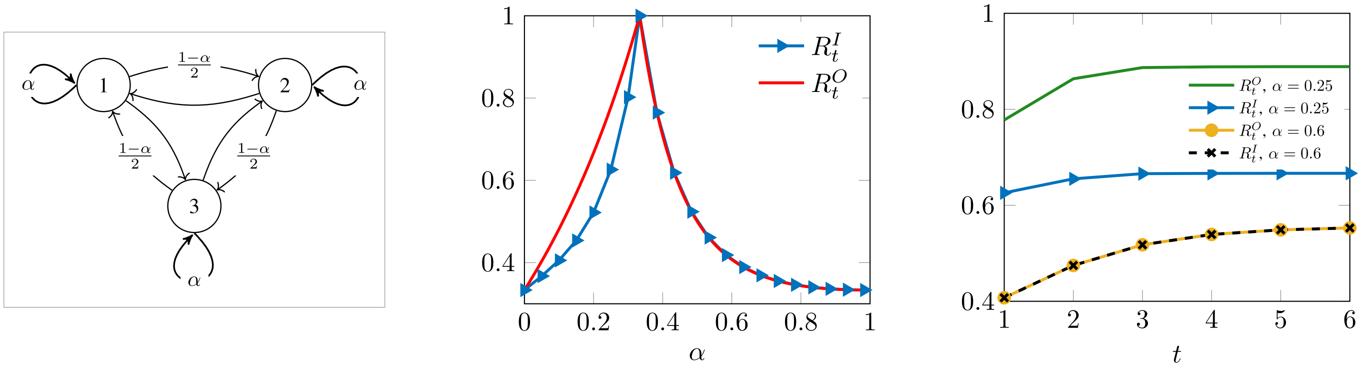
where $0 \leq \alpha \leq 1$ *and* $P_{i,j}$ *denotes the transition probability from state* i *to state* j.

Corollary 2: (Optimality for Symmetric Markov Chain): For the symmetric Markov chain such that $\frac{1}{n} \leq \alpha \leq 1$, *the two bounds* (12) *and* (13) *match. In particular,*

$$\frac{1}{R_t^I} = \frac{1}{R_t^O} = \alpha n \frac{(n-1)^{t-\tau} + (n-1)(n\alpha - 1)^{\tau}}{(n-1)^{t-\tau} + (n\alpha - 1)^{t-\tau+1}}. \quad (17)$$

In other words, the rate tuple $(R_t : t \in \mathbb{N})$ *is achievable if and only if*

$$\frac{1}{R_t} \geq \alpha n \frac{(n-1)^{t-\tau} + (n-1)(n\alpha - 1)^{\tau}}{(n-1)^{t-\tau} + (n\alpha - 1)^{t-\tau+1}}. \quad (18)$$



(a) A Symmetric Markov Chain.

(b) R_t^I and R_t^O as a function of α.

(c) R_t^I and R_t^O as a function of time (t).

Fig. 2. In Figure 2a, we graphically represent the 3-state symmetric Markov chain used in Example 1, where $0 \le \alpha \le 1$. In Figure 2b, we plot the achievable rate R_t^I (c.f.(12)) and the upper bound R_t^O (c.f.(13)), as a function of α, when $\tau = 0$ and $t = 1$. In Figure 2c, we plot R_t^I and R_t^O as a function of time t for both $\alpha = 0.25$ and $\alpha = 0.6$.

The proofs of Corollaries 1 and 2 can be found in Appendix A and Appendix B, respectively.

Example 1: We study a special case described in Corollary 2. Suppose that we are given $\tau = 0$, and a 3-state Markov chain, as represented in Figure 2a, where $0 \le \alpha \le 1$.

In this case, we have two regimes, one for $\alpha < \frac{1}{3}$ and the other for $\alpha \ge \frac{1}{3}$. This is because the ordering of probabilities (c.f.(7)) changes at $\alpha = \frac{1}{3}$.

For $\alpha \ge \frac{1}{3}$, the bounds (12) *and* (13) *match, e.g., for $t = 1$,*

$$\frac{1}{R_1^O} = \frac{1}{R_1^I} = \frac{6\alpha^2}{3\alpha^2 - 2\alpha + 1}. \tag{19}$$

However, for $\alpha < \frac{1}{3}$ and $t = 1$, we have

$$\frac{1}{R_1^O} = \frac{3-3\alpha}{3\alpha+1} \le \frac{1}{R_1^I} = \frac{2}{3\alpha+1} - \frac{4\alpha-2}{3\alpha^2-2\alpha+1} - 1. \tag{20}$$

We illustrate (19) *and* (20) *in Figure 2b.*

In Figure 2c, we analyze the rate over time for $\alpha = 0.25$, and $\alpha = 0.6$. It is notable that as t grows, the correlation between X_t (the current request) and X_τ (the request when privacy was last ON) decreases, which leads to an increase in the download rate R_t.

V. Achievability: Linear Programming Formulation

Towards finding an ON-OFF privacy scheme, we consider uncoded queries for retrieving messages, *i.e.,* the query Q_t at time t takes values in the power set of $\mathcal{N}$, denoted by $\mathscr{P}(\mathcal{N})$. In other words, the user will query for a subset of the messages $W_{\mathcal{N}}$ at each time. Later in this section, we will see that designing an uncoded query scheme is equivalent to solving a linear programming problem.

Upon receiving the query $Q_t \subseteq \mathcal{N}$, the server generates a corresponding answer $A_t = W_{Q_t} \subseteq W_{\mathcal{N}}$. The length of the answer can be written as

$$\ell(Q_t) = |Q_t|\, L,$$

where L is the length of a message. Therefore, the average length ℓ_t is

$$\ell_t = \mathbb{E}[|Q_t|]\, L. \tag{21}$$

Next, we describe how to construct the query Q_t for each time t. The query Q_t is a probabilistic function of the current request X_t and $U_t = (X_\tau, X_{t+1})$(c.f.(6)). Therefore, the encoding of the query Q_t can be equivalently denoted by the probability distribution $w(q_t|x_t, u_t)$, where $x_t \in \mathcal{N}$, $u_t = (x_\tau, x_{t+1}) \in \mathcal{N}^2$ and $q_t \in \mathscr{P}(\mathcal{N})$. In other words, given $x_t \in \mathcal{N}$ and $u_t = (x_\tau, x_{t+1}) \in \mathcal{N}^2$, the user will send $q_t \in \mathscr{P}(\mathcal{N})$ with probability $w(q_t|x_t, u_t)$.

Since $A_t = W_{Q_t}$, if $X_t \in Q_t$, then the retrieved answer contains the desired message W_{X_t}. Therefore, if

$$p(q_t, x_t|u_t) = 0, \quad \forall x_t \notin q_t, \tag{22}$$

then decodability is guaranteed. Note that $p(q_t, x_t|u_t)$ can be written as

$$p(q_t, x_t|u_t) = p(x_t|u_t)\, w(q_t|x_t, u_t),$$

where $p(x_t|u_t)$ is given by the Markov chain, so $p(q_t, x_t|u_t)$ is completely determined by $w(q_t|x_t, u_t)$.

To guarantee the privacy(c.f.(4)), we introduce the following lemma. It states that if we design the encoding function $w(q_t|x_t, u_t)$ such that

$$p(q_t|u_t) = p(q_t), \quad \forall u_t \in \mathcal{N}^2,\ q_t \in \mathscr{P}(\mathcal{N}), \tag{23}$$

for all $t \in \mathbb{N}$, then the scheme satisfies the required privacy constraint (4).

Lemma 1: If Q_i is a probabilistic function of U_i and X_i, and Q_i is independent of U_i for $i = 0, 1, \ldots, t$, then $Q_{[t]}$ is independent of $X_{\mathcal{B}_t}$, where $\mathcal{B}_t = \{i : i \le t, F_i = ON\} \cup \{i : i \ge t+1\}$.

Proof: See Appendix C. □

Since the download cost of the scheme is as given in (21), *i.e.,* $\ell_t = \mathbb{E}[|Q_t|]\, L$, and we desire a scheme with low download cost (high rate), we would like to design an encoding function $w(q_t|x_t, u_t)$ that minimizes $\mathbb{E}[|Q_t|]$.

Hence, it remains to design the distribution $w(q_t|x_t, u_t)$ that minimizes $\mathbb{E}[|Q_t|]$ under the constraints (22) and (23).

As such, any feasible solution to the following optimization problem corresponds to an admissible encoding function $w\left(q_t|x_t,u_t\right)$ as desired.

$$\begin{aligned} \underset{w(q_t|x_t,u_t)}{\text{minimize}}\ & \mathbb{E}\left[|Q_t|\right]=\sum_{q_t} p\left(q_t\right)|q_t| \\ \text{subject to}\ & p\left(x_t,q_t|u_t\right)=0, \quad x_t\notin q_t, \\ & p\left(q_t|u_t\right)=p\left(q_t\right). \end{aligned} \tag{24}$$

Note that the problem is always feasible, as

$$w\left(q_t=\mathcal{N}|x_t,u_t\right)=1, \quad \forall u_t,x_t, \tag{25}$$

is a feasible solution to (24).

One may also notice that if we treat each probability $w\left(q_t|x_t,u_t\right)$ for $x_t\in\mathcal{N}$, $u_t\in\mathcal{N}^2$ and $q_t\in\mathscr{P}\left(\mathcal{N}\right)$ as a decision variable, then both the objective function and two constraints are linear, and hence the optimization problem (24) is indeed a linear programming instance. However, this linear programming problem has $n^3\,2^n$ variables and $n\,2^{n-1}+n^2\,2^n$ constraints. The scale of the problem is intractable in complexity with any generic linear programming solver. For example, using the techniques presented in [30], the complexity of this linear programming is $\mathcal{O}\left(\left(n^2\,2^n\right)^{2.5}\right)$. This makes the numerical solution impossible when n is large.

Therefore, in the following section, we present a polynomial time algorithm that gives a feasible solution that might not always be optimal.

VI. Efficient ON-OFF Privacy Query Scheme

Instead of attempting to solve the linear programming problem (24) numerically, we are going to identify a feasible solution $w^*\left(q_t|x_t,u_t\right)$ to the problem *efficiently*, and bound the objective $\mathbb{E}\left[|Q_t|\right]$ *analytically*, *i.e.,* a feasible solution attains an objective such that

$$\mathbb{E}\left[|Q_t^*|\right]\le 1/R_t^I=\sum_{i=1}^{n} i\,\theta_i(t), \tag{26}$$

which means there exists a scheme such that the download cost ℓ_t is less than or equal to L/R_t^I, or R_t^I is achievable.

A key observation on (24) is that any tractable solution $w\left(q_t|x_t,u_t\right)$ must be sparse, *i.e.,* a few non-zero valued probabilities $w\left(q_t|x_t,u_t\right)$ for $x_t\in\mathcal{N}$, $u_t\in\mathcal{N}^2$ and $q_t\in\mathscr{P}(\mathcal{N})$. Otherwise, simply initializing or outputting the solution $w\left(q_t|x_t,u_t\right)$ introduces an exponential overhead in complexity. This observation motivates our algorithm, which admits a sparse $w\left(q_t|x_t,u_t\right)$.

Since the time index t will be clear from context, in the sequel we drop it from the subscripts. For any given $p\left(x|u\right)$, we recall the optimization problem we are interested in,

$$\begin{aligned} \underset{w(q|x,u)}{\text{minimize}}\ & \mathbb{E}\left[|Q|\right]=\sum_{q} p\left(q\right)|q| \\ \text{subject to}\ & p\left(x,q|u\right)=0, \quad x\notin q, \\ & p\left(q|u\right)=p\left(q\right), \end{aligned} \tag{27}$$

where $x\in\mathcal{N}$, $u\in\mathcal{N}^2$ and $q\in\mathscr{P}(\mathcal{N})$.

Instead of finding a feasible solution to (27) directly, we introduce an auxiliary random variable Z. Let Z be a multiset $(\mathcal{N},f)$, where $\mathcal{N}$ is the ground set and f is the multiplicity function. The cardinality of the multiset Z is the summation of multiplicities of all its element, *i.e.,*

$$|Z|=\sum_{x\in\mathcal{N}} f(x).$$

Let $\mathcal{Z}$ be the collection of all multisets such that cardinality is bounded by n, *i.e.,*

$$\mathcal{Z}=\left\{Z:Z\in\left(\mathcal{N},f\right),|Z|\le n\right\}. \tag{28}$$

Then for any given $p\left(x|u\right)$, we can define an alternative optimization problem:

$$\begin{aligned} \underset{w(z|x,u)}{\text{minimize}}\ & \mathbb{E}\left[|Z|\right]=\sum_{z} p\left(z\right)|z| \\ \text{subject to}\ & p\left(x,z|u\right)=0, \quad x\notin z, \\ & p\left(z|u\right)=p\left(z\right), \end{aligned} \tag{29}$$

where $x\in\mathcal{N}$, $u\in\mathcal{N}^2$ and $z\in\mathcal{Z}$.

One can easily check that any feasible solution to (29) can be easily transformed to be a feasible solution to (27) by simply letting $Q=\text{Set}(Z)$, *i.e.,* forcing the multiplicity of elements in Z to be 1. Moreover, the corresponding solution to (27) attains a better objective value, *i.e.,* if $w\left(z|x,u\right)$ is a feasible solution to (29) and $w\left(q|x,u\right)$ is a feasible solution to (27), then $\mathbb{E}\left[|Q|\right]\le\mathbb{E}\left[|Z|\right]$, where $\mathbb{E}\left[|Z|\right]$ is the objective value attained by $w\left(z|x,u\right)$ and $\mathbb{E}\left[|Q|\right]$ is the objective value attained by $w\left(q|x,u\right)$, respectively. Therefore, we will study the feasible region of (29) instead. In particular, we will find a feasible solution $w^*\left(z|x,u\right)$ such that

$$\mathbb{E}\left[|Z^*|\right]=\sum_{i=1}^{n} i\,\theta_i. \tag{30}$$

Then there exists a corresponding feasible solution $w^*\left(q|x,u\right)$, by simply letting $Q=\text{Set}(Z)$, to the original problem (27) such that

$$\mathbb{E}\left[|Q^*|\right]\le\sum_{i=1}^{n} i\,\theta_i, \tag{31}$$

which is the same as (26) and is to be proved.

In the remainder of this section, we start by describing the algorithm in Subsection VI-A. We then analyze its complexity in Subsection VI-B, and finally in Subsection VI-C, we verify that the algorithm outputs a feasible solution as desired.

A. Algorithm Description

In this section, we describe the algorithm to construct a feasible solution $w\left(z|x,u\right)$ to (29), *i.e.,* for any given distribution $p\left(x|u\right)$, we will give a constructive proof of some Z, satisfying that

$$p\left(z,x|u\right)=0, \quad \forall x\notin z, \tag{32}$$

and

$$p\left(z|u\right)=p\left(z|u'\right), \quad \forall z\in\mathcal{Z} \text{ and } u,\ u'\in\mathcal{N}^2. \tag{33}$$

In particular, we will show that the feasible solution $w(z|x,u)$ gives

$$p(|Z|=\ell)=\theta_\ell, \quad \ell=1,\ldots,n. \tag{34}$$

Note that $\theta_i \geq 0$ for all $i=1,\ldots,n$ by the definition (9), which is stated in the following proposition.

Proposition 1: For any given Markov chain and time index t, $\theta_i \geq 0$, for $i=1,\ldots,n$.

Proof: See Appendix D. □

One can see that the objective value attained by this feasible solution is

$$\mathbb{E}[|Z|]=\sum_{i=1}^{n} i\,\theta_i.$$

Before describing the steps of the algorithm we give an intuitive explanation and overview of the algorithm. In order to minimize $\mathbb{E}[|Z|]$, we would like to construct some $w(z|x,u)$ that makes the probability $p(|Z|=\ell)$ larger for smaller ℓ, *i.e.,* a greedy-like algorithmic approach is appealing. As a result of the two constraints (32) and (33), one can easily check that the maximum value of $p(|Z|=1)$ is θ_1, and the solution gives

$$p(z,x|u)=\min_{u'\in\mathcal{N}^2} p\left(x|u'\right), \quad \forall u \text{ and } z=x.$$

We would like to keep this greedy manner to manage the probabilities $p(z,x|u)$ for $|z|=2,\ldots,n$. However, when $|z|\geq 2$, it becomes more complicated. For instance, when $|z|=2$, one of the two elements of the set z has to be x, in order to satisfy (32), which corresponds to the decodability constraint. Roughly speaking, we aim to use the second element of z to obfuscate each x with another x' in order to satisfy (33), which corresponds to the privacy constraint. The challenging part of this algorithm is this choice of x', and the corresponding probability $p(z,x|u)$, where $z=\{x,x'\}$.

The following algorithm, consisting of five main steps, rigorously describes how we design this obfuscation. In Step 1, we calculate preliminaries from the given probability distribution $p(x|u)$ and initialize the algorithm. In Step 2, we describe how to properly obfuscate each x with the other $\ell-1$ elements for a given ℓ, and in Step 3, we describe how to design a common obfuscation (obtain some common sets z of cardinality ℓ and some proper values) for all $x\in\mathcal{N}$ simultaneously. Then, in Step 4, we augment the configurations to the initialized variables, and finally in Step 5, we output the configurations and the values. Details are given as follows:

- **Step 1: Preliminaries**

For any given distribution $p(x|u)$, by sorting $p(x|u)$ for each $x\in\mathcal{N}$, we can easily obtain parameters

$$\left\{u_{x,i}: x\in\mathcal{N}, i\ =1,\ldots,m\right\},$$

where $u_{x,i}$ is as defined in (7) and $m=|\mathcal{N}^2|=n^2$. For notational simplicity, let

$$\lambda_{x,i}=p\left(X_t=x|U_t=u_{x,i}\right).$$

Let M be an auxiliary $m\times n$ matrix determined by the given $p(x|u)$. In particular, we initialize M by

$$M_{i,j}=\max\left\{p(X=j|U=i)-\lambda_{j,n-1},0\right\}. \tag{35}$$

for $i=1,\ldots,m$, and $j=1,\ldots,n$. This matrix will be updated during the following procedure. For the ease of notation, let $M_{i,j}^{-}=a$ denote $M_{i,j}=M_{i,j}-a$, *i.e.,* subtracting a from $M_{i,j}$.

For $\ell=1,\ldots,n-1$ and $x=1,\ldots,n$, we access to $\{u_{x,i}: i=1,\ldots,\ell-1\}$. For ease of notation, let

$$\mathcal{U}_{\ell,x}^{-}=\left\{u_{x,i}: i=1,\ldots,\ell-1\right\},$$

and

$$\mathcal{U}_{\ell,x}^{+}=\left\{u_{x,i}: i=\ell,\ldots,m\right\}.$$

- **Step 2:**

For each i, or precisely $u_{x,i}$, we choose a collection of pairs

$$I_{\ell,x,i}\times V_{\ell,x,i}=\left\{\left(e_{\ell,x,i,j},v_{\ell,x,i,j}\right): j=1,2,\ldots,c_{\ell,x,i}\right\} \tag{36}$$

such that

$$0\leq v_{\ell,x,i,j}\leq M_{u_{x,i},e_{\ell,x,i,j}}, \tag{37}$$

and

$$\sum_{j=1}^{c_{\ell,x,i}} v_{\ell,x,i,j}=\lambda_{x,\ell}-\lambda_{x,\ell-1}. \tag{38}$$

where $e_{\ell,x,i,j}$ for $j=1,\ldots,c_{\ell,x,i}$ are distinct indices belonging to $\{1,\ldots,n\}$, and clearly we have $c_{\ell,x,i}\leq n$.

Then, we update the matrix M by

$$M_{u_{x,i},e_{\ell,x,i,j}}^{-}=v_{\ell,x,i,j}, \tag{39}$$

for all $e_{\ell,x,i,j}\in I_{\ell,x,i}$. We slightly abuse the notation here by using the same notation M to denote the matrix at different points. Nevertheless, the underlying ℓ, x and i we are dealing with will be clear from context.

Roughly speaking, we extract $\lambda_{x,\ell}-\lambda_{x,\ell-1}$ from the $u_{x,i}$-th row of the non-negative matrix M for given ℓ and x, where $e_{\ell,x,i,j}$ and $v_{\ell,x,i,j}$ specify the column indices and values extracted from each position of $u_{x,i}$-th row. The matrix M is always non-negative during the update from (37) and (39), so the existence of such a collection of $I_{\ell,x,i}\times V_{\ell,x,i}$ can be guaranteed if the summation of the $u_{x,i}$-th row of the initialized matrix M(c.f.(35)) is greater than or equal to the summation of the subtracted values (the right-hand side of (40)) for all x and ℓ during the process, which is given by the following proposition.

Proposition 2: For any $u=1,\ldots,m$,

$$\sum_{x=1}^{n}\max\left\{p(X=x|U=u)-\lambda_{x,n-1},0\right\} \geq \sum_{\ell=1}^{n-1}\sum_{x:u\in\mathcal{U}_{\ell,x}^{-}}\left(\lambda_{x,\ell}-\lambda_{x,\ell-1}\right). \tag{40}$$

Proof: See Appendix E. □

- **Step 3:**

For fixed ℓ and x, after finishing the above process for all $i=1,\ldots,\ell-1$, we obtain $I_{\ell,x,i}$ and $V_{\ell,x,i}$ for $i=1,\ldots,\ell-1$. Provided $I_{\ell,x,i}$ and $V_{\ell,x,i}$ for $i=1,\ldots,\ell-1$, we pick a collection of pairs

$$\left\{(\zeta_{\ell,x,k},\nu_{\ell,x,k}): k=1,2,\ldots,c_{\ell,x}\right\}$$

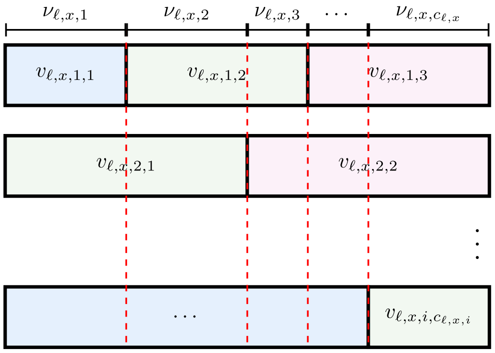


Fig. 3. The rows represents $V_{\ell,x,1},\dots,V_{\ell,x,\ell-1}$ for given ℓ and x. Each block represents an element $\upsilon_{\ell,x,i,j}$ in the set $V_{\ell,x,i}$, where $j = 1,\dots,c_{\ell,x,i}$. Each $\nu_{\ell,x,k}$ can be chosen to be the value of the difference between two consecutive boundaries of blocks, *e.g.*, $\nu_{\ell,x,1} = \upsilon_{\ell,x,1,1}$ and $\nu_{\ell,x,2} = \upsilon_{\ell,x,2,1} - \upsilon_{\ell,x,1,1}$ etc. The corresponding $\zeta_{\ell,x,k}$ can be chosen to be $\zeta_{\ell,x,1} = \left(e_{\ell,x,1,1}, e_{\ell,x,2,1},\cdots\right)$ and $\zeta_{\ell,x,2} = \left(e_{\ell,x,1,2}, e_{\ell,x,2,1},\cdots\right)$ etc.

such that

$$\zeta_{\ell,x,k} \in I_{\ell,x,1} \times I_{\ell,x,2} \times \cdots \times I_{\ell,x,\ell-1},$$

and

$$\sum_{k:\zeta_{\ell,x,k}(i)=e_{\ell,x,i,j}} \nu_{\ell,x,k} = \upsilon_{\ell,x,i,j}, \tag{41}$$

for all $i = 1,\dots,\ell-1$ and $j = 1,\dots,c_{\ell,x,i}$, where $\zeta_{\ell,x,k}(i)$ is the i-th element of $\zeta_{\ell,x,k}$, *i.e.*, $\zeta_{\ell,x,k}(i) \in I_{\ell,x,i}$.

A simple deterministic approach of picking such a collection of $(\zeta_{\ell,x,k}, \nu_{\ell,x,k})$ can be basically illustrated by Figure 3. Roughly speaking, there is a buffer tracking the front of the sets $V_{\ell,x,i}$ for $i = 1,\dots,\ell-1$. Each time, the buffer pushes the minimal value among them *i.e.*, $\nu_{\ell,x,k}$, minus the value from the front, and adds one more value from the same set $V_{\ell,x,i}$ which has been pushed out. The corresponding positions of values in the buffer form the set $\zeta_{\ell,x,k}$. As such, we can easily see that

$$\sum_{k=1}^{c_{\ell,x}} \nu_{\ell,x,k} = \lambda_{x,\ell} - \lambda_{x,\ell-1}. \tag{42}$$

Also, one can easily check that this process returns

$$c_{\ell,x} = \sum_{i=1}^{\ell-1} c_{\ell,x,i} \le n(\ell-1). \tag{43}$$

- **Step 4: Augment**

For each $k = 1,\dots,c_{\ell,x}$, let

$$z_{\ell,x,k} = \{\zeta_{\ell,x,k}, x\}, \tag{44}$$

and

$$\begin{aligned}&\mathcal{F}_{\ell,x,k}\\ &= \left\{(\bar{z},\bar{x},\bar{u}) : \bar{z} = z_{\ell,x,k}, \bar{x} = x, \bar{u} \in \mathcal{U}^{+}_{\ell,x}\right\}\\ &\bigcup \left\{(\bar{z},\bar{x},\bar{u}) : \bar{z}=z_{\ell,x,k}, \bar{x}=\zeta_{\ell,x,k}(i), \bar{u}=u_{x,i} \in\mathcal{U}^{-}_{\ell,x}\right\}.\end{aligned} \tag{45}$$

For each ℓ and x, we can obtain $\mathcal{F}_{\ell,x,k}$ and $\nu_{\ell,x,k}$ for $k = 1,\dots,c_{\ell,x}$. The tuple $(\bar{z},\bar{x},\bar{u})$ in $\mathcal{F}_{\ell,x,k}$ is indeed the non-zero valued position and $\nu_{\ell,x,k}$ is the value that we will assign to the probability $p\,(\bar{z},\bar{x}|\bar{u})$. However, since there may exist duplicated tuples in $\mathcal{F}_{\ell,x,k}$ for different x, we augment the value $\nu_{\ell,x,k}$ corresponding to the same tuple $(\bar{z},\bar{x},\bar{u})$, *i.e.*,

$$\mathcal{F}_{\ell} = \bigcup_{x=1}^{n} \bigcup_{k=1}^{c_{\ell,x}} \mathcal{F}_{\ell,x,k}, \tag{46}$$

and for any $(\bar{z},\bar{x},\bar{u}) \in \mathcal{F}_{\ell}$,

$$g\,(\bar{z},\bar{x},\bar{u}) = \sum_{x=1}^{n} \sum_{k:(\bar{z},\bar{x},\bar{u})\in\mathcal{F}_{\ell,x,k}} \nu_{\ell,x,k}. \tag{47}$$

After obtaining $\mathcal{F}_{\ell}$ for $\ell = 1,\dots,n-1$, for $\ell = n$, let

$$\mathcal{F}_{n} = \left\{(\bar{z},\bar{x},\bar{u}) : \bar{z} = \mathcal{N}, M_{\bar{u},\bar{x}} > 0\right\}, \tag{48}$$

and

$$g(\bar{z},\bar{x},\bar{u}) = M_{\bar{u},\bar{x}}, \tag{49}$$

for any $(\bar{z},\bar{x},\bar{u}) \in \mathcal{F}_{n}$.

- **Step 5: Output**

The output of the algorithm is $\{\mathcal{F}, g(\mathcal{F})\}$, where

$$\mathcal{F} = \{\mathcal{F}_{\ell} : \ell = 1,\dots,n\}.$$

stores the non-zero valued positions of an admissible distribution $p\,(z,x|u)$ for $x \in \mathcal{N}$, $u \in \mathcal{N}^2$ and $z \in \mathcal{Z}$, and $g(\mathcal{F})$ stores the corresponding probabilities.

B. Complexity

For the sake of completeness, we discuss the complexity of the algorithm. As said, the bottleneck is to represent the solution $w\,(z|x,u)$ for $z \in \mathcal{Z}$, $x \in \mathcal{N}$ and $u \in \mathcal{N}^2$, which has exponential number of values, so the complexity is indeed dominated by the size of $\mathcal{F}$, *i.e.*, the non-zero valued positions of the output distribution $p\,(z,x|u)$.

It is notable that $|\mathcal{F}_{\ell,x,k}| = m$ and $c_{\ell,x} \le n^2$ from (43), so we have

$$|\mathcal{F}| \le \sum_{\ell=1}^{n} \sum_{x=1}^{n} \sum_{k=1}^{c_{\ell,x}} |\mathcal{F}_{\ell,x,k}| \le mn^4 = n^6, \tag{50}$$

i.e., the complexity of the algorithm is $\mathcal{O}(n^6)$.

The purpose of the complexity analysis here is to justify that the proposed algorithm is with poly(n) complexity. One may possibly reduce the complexity by orders by utilizing some data structures, which is beyond the interest of this paper.

C. Algorithm Verification

In this subsection, we will verify the algorithm, *i.e.*, we will prove that it outputs a distribution $p\,(z,x|u)$ satisfying (32), (33) and (34) for any given distribution $p\,(x|u)$.

First, we show that the algorithm described in VI-A outputs a distribution $p\,(z,x|u)$ satisfying (32) and (33) for any given distribution $p\,(x|u)$.

Proposition 3: For any given $p\,(x|u)$ for $u \in \mathcal{N}^2$ and $x \in \mathcal{N}$, $\{\mathcal{F}, g(\mathcal{F})\}$ returns non-zero valued positions and values of some distribution $p\,(z,x|u)$ such that $p\,(z,x|u) = 0$ for all $x \notin z$ and $p\,(z|u) = p\left(z|u'\right)$ for all $z \in \mathcal{Z}$ and $u, u' \in \mathcal{N}^2$.

Proof:

As claimed, $\mathcal{F}$ and $g(\mathcal{F})$ store the non-zero valued positions and values of $p(z, x|u)$, so it is equivalent for us to show that

1) For any $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}$, we have

$$\bar{x} \in \bar{z}. \tag{51}$$

2) For any given $\bar{z}$, $\bar{u}$ and $\bar{u}'$, we have

$$\sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}} g(\bar{z}, x, \bar{u}) = \sum_{x:(\bar{z},x,\bar{u}')\in\mathcal{F}} g\left(\bar{z}, x, \bar{u}'\right). \tag{52}$$

3) For any given $\bar{x}$ and $\bar{u}$, we have

$$\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}} g(z, \bar{x}, \bar{u}) = p(\bar{x}|\bar{u}). \tag{53}$$

Details can be found in Appendix F. □

Next, we show that the algorithm described in VI-A returns $p(z, x|u)$ satisfying (34).

Proposition 4: For any given $p(x|u)$ for $u \in \mathcal{N}^2$ and $x \in \mathcal{N}$, the algorithm returns some distribution $p(z, x|u)$ such that

$$p(|Z| = \ell) = \theta_\ell, \quad \ell = 1, \ldots, n. \tag{54}$$

Proof: See Appendix G. □

VII. An Outer Bound

In this section, we will show that any ON-OFF privacy scheme must satisfy $R_t \leq R_t^O$.

First, we define an auxiliary random variable Y_t taking values in $\mathscr{P}(\mathcal{N})$ based on the decodability of the subset of messages. Specifically, let Y_t be a function of Q_t such that $Y_t = \mathcal{D}$ for $\mathcal{D} \in \mathscr{P}(\mathcal{N})$ if the user may decode the messages $W_{\mathcal{D}}$ but not any message W_i for $i \in \mathcal{N}\backslash\mathcal{D}$ from the answer A_t. Roughly speaking, Y_t represents the capability of decoding messages from the query Q_t. Note that since the query Q_t and messages $W_{\mathcal{N}}$ are independent, the decodability of any message is known by the server only through Q_t, that is, Y_t is a function of Q_t. In this way, the alphabet $\mathcal{Q}$ (may be infinite if the query is coded) of the query is partitioned into 2^n classes based on the decodability of the subset of the messages. Clearly, from the definition of Y_t, we have that the length of the answer $\ell(Q_t)$ satisfies

$$\ell(Q_t) \geq |Y_t|\, L,$$

since the answer A_t is at least of length $|Y_t|\, L$ if the user can decode $|Y_t|$ messages from the answer A_t. Hence, the download cost ℓ_t is bounded by

$$\ell_t \geq \mathbb{E}[|Y_t|]\, L. \tag{55}$$

Next, we start to reinterpret the privacy and the decodability constraints in terms of the auxiliary variable Y_t. By the definition of Y_t, the decodability can be written as

$$p(x_t, y_t) = 0, \quad \forall x_t \notin y_t, \tag{56}$$

where $x_t \in \mathcal{N}$ and $y_t \in \mathscr{P}(\mathcal{N})$.

Recall the privacy constraint

$$I\left(X_{\mathcal{B}_t}; Q_{[t]}\right) = 0,$$

and we must have

$$I\left(X_{\mathcal{B}_t}; Q_{[t]}\right) \overset{\text{(a)}}{\geq} I(U_t; Q_t) \overset{\text{(b)}}{\geq} I(U_t; Y_t),$$

where (a) follows from $U_t = (X_\tau, X_{t+1}) \subset X_{\mathcal{B}_t}$ and (b) follows because Y_t is a function of Q_t.

Thus, we can relax the privacy constraint by

$$I(U_t; Y_t) = 0. \tag{57}$$

For any given $p(x_t|u_t)$, if Y_t takes values in $\mathscr{P}(\mathcal{N})$ and satisfies (56) and (57), then $\mathbb{E}[|Y_t|]$ is lower bounded by the following lemma.

Lemma 2: For any random variables U, X and Y, taking values in the alphabet $\mathcal{N}^2$, $\mathcal{N}$ and $\mathscr{P}(\mathcal{N})$ respectively, if Y is independent of U, and $p(x, y|u) = 0$ for $x \notin y$, then

$$\mathbb{E}[|Y|] \geq \sum_{x\in\mathcal{N}} \max_{u\in\mathcal{N}^2} p(x|u). \tag{58}$$

Proof: See Appendix H. □

By substituting (58) in (55), we have

$$\ell_t \geq L \sum_{x_t\in\mathcal{N}} \max_{u_t\in\mathcal{N}^2} p(x_t|u_t). \tag{59}$$

Therefore, for any ON-OFF privacy scheme satisfying the decobability and privacy constraint, we know that the download cost is lower bounded by the right-hand side of (59). In other words, any ON-OFF privacy scheme must satisfy

$$\frac{1}{R_t} \geq \frac{1}{R_t^O} = \sum_{x_t\in\mathcal{N}} \max_{u_t\in\mathcal{N}^2} p(x_t|u_t) = \lambda_m(t).$$

VIII. LP Formulation of Optimal Achievable Rate

In this section, we present an implicit characterization of the optimal rate, which is formulated by a linear program with an exponential number (in n) of variables and constraints.

As discussed in Section V, the query design relies on solving the following linear program:

$$\begin{aligned} \underset{w(q|x,u)}{\text{minimize}} \quad & \mathbb{E}[|Q|] = \sum_q p(q)\,|q| \\ \text{subject to} \quad & p(x, q|u) = 0, \quad x \notin q, \\ & p(q|u) = p(q), \end{aligned} \tag{60}$$

where $x \in \mathcal{N}$, $u \in \mathcal{N}^2$, $q \in \mathscr{P}(\mathcal{N})$ and probabilities $p(x|u)$ are given. We know that any feasible solution to the above problem yields an achievable scheme. In other words, the rate R_t is achievable if

$$\frac{1}{R_t} \geq C_1^*,$$

where C_1^* is the optimal value to (60).

On the other hand, one may notice that the key lemma, *i.e.,* Lemma 2, to show the outer bound, indeed indicates that any achievable scheme must satisfy that

$$\frac{1}{R_t} \geq C_2^*,$$

where C_2^* is the optimal value to the following problem:

$$\begin{aligned} \underset{w(y|x,u)}{\text{minimize}} \quad & \mathbb{E}[|Y|] = \sum_y p(y)\,|y| \\ \text{subject to} \quad & p(x, y|u) = 0, \quad x \notin y, \\ & p(y|u) = p(y), \end{aligned} \tag{61}$$

where $x \in \mathcal{N}$, $u \in \mathcal{N}^2$, $y \in \mathscr{P}(\mathcal{N})$ and probabilities $p(x|u)$ are given by the Markov chain.

Although problems (60) and (61) have different physical meanings, it is easy to see that they have the same optimal value, *i.e.,* $C_1^* = C_2^*$. Therefore, by letting C_t be the optimal value to both problems, the achievable region can be fully characterized by

Corollary 3: The rate tuple $(R_t : t \in \mathbb{N})$ *is achievable if and only if* $R_t \leq C_t$.

However, it is notable that Corollary 3 is an *implicit* characterization, because as we discussed, the exponential blow-up of the number of variables and constraints makes the linear programming problem intractable.

Remark 3 (Window Size ω): From our earlier discussion, we know that the feasible region of (60) *denotes schemes that only require a window of size* $\omega = 1$. *Although we have assumed that the user knows the future requests within a window of positive size ω, increasing the window size into the future beyond* $\omega = 1$ *does not, in fact, increase the rate. Intuitively, this phenomenon stems from the Markov assumption we use to model the user's requests. If the window size* $\omega = 0$, *i.e., no future requests are known, the privacy defined in* (4) *has to be relaxed, and only past requests can be protected, which was studied in [28].*

In this paper, we actually proposed an explicit scheme to the ON-OFF privacy problem by finding a feasible solution that might not be optimal, to (60), which is of polynomial time complexity. Moreover, we show that our scheme is optimal for some cases in Corollary 1 and Corollary 2.

IX. Conclusion and Future Directions

In this paper, we continue to look at the problem of turning privacy ON and OFF in an information retrieval setting when the user's interests are correlated over time. We model this correlation by a Markov chain with n states. Our previous work in [25] focused on privacy for past interests. Our work in [26] studied privacy for the past and the future, albeit for the special case of Markov chains with $n = 2$ states. In this paper, we generalize the work in [26] to Markov chains with $n \geq 2$ states. We give a new achievable scheme with polynomial time complexity and a general upper bound on the achievable rate. We prove the optimality of our scheme for special cases, namely, a family of symmetric Markov chains, and two-state Markov chains.

Future directions of this work include finding tighter outer bounds on the rate and efficient constructions of ON-OFF privacy schemes that would achieve them. Also, it is worthy to investigate settings in which the user's requests follow a different model than the Markov chain, or the user's requests and desired privacy status are correlated.

Appendix A
Optimality for $n = 2$

The special case when $n = 2$ was first studied in [26]. For $n = 2$, the two bounds (12) and (13) match, *i.e.,* $R_t^I = R_t^O$. To see this we write R_t^I by

$$\frac{1}{R_t^I} = \sum_{i=1}^{m} i\,\theta_i(t) = 2 - \lambda_1(t).$$

For a given x_t, *e.g.,* $x_t = 1$, suppose that $u^* = \arg\min_{u_t} p(x_t|u_t)$. Then we can see that, for $\bar{x}_t = 2$, $u^* = \arg\max_{u_t} p(\bar{x}_t|u_t)$ since $p(x_t|u_t) + p(\bar{x}_t|u_t) = 1$, for any u_t when $n = 2$. Thus, we have

$$\min_{u_t} p(x_t|u_t) + \max_{u_t} p(\bar{x}_t|u_t) = 1,$$

for any x_t and $\bar{x}_t = \mathcal{N} \backslash \{x_t\}$, which implies that

$$\begin{aligned}\lambda_1(t) &= \sum_{x_t=1}^{2} \min_{u_t} p(x_t|u_t) = \sum_{x_t=1}^{2} \left(1 - \max_{u_t} p(\bar{x}_t|u_t)\right)\\ &= 2 - \sum_{\bar{x}_t=1}^{2} \max_{u_t} p(\bar{x}_t|u_t) = 2 - \lambda_m(t).\end{aligned}$$

Therefore, we can obtain that

$$\frac{1}{R_t^I} = 2 - \lambda_1(t) = \lambda_m(t) = \frac{1}{R_t^O}.$$

Appendix B
Proof of Corollary 2

We first take the transition matrix P to the power of t, *i.e.,*

$$(P^t)_{i,j} = \begin{cases} \dfrac{(n-1)^{t-1} - (n\alpha-1)^t}{n(n-1)^{t-1}}, & \text{if} \quad i = j, \\ \dfrac{(n-1)^t - (n\alpha-1)^t}{n(n-1)^t}, & \text{if} \quad i \neq j, \end{cases}$$

for all $i, j \in \{1, \ldots, n\}$.

Then, the probabilities $p(x_t|u_t)$ can be written as

$$p(X_t = j | X_\tau = i, X_{t+1} = k) = \frac{P_{j,k}(P^\delta)_{i,j}}{(P^{\delta+1})_{i,k}}, \tag{62}$$

where $\delta = t - \tau$ and $i, j \in \{1, \ldots, n\}$. By invoking the symmetry of the given Markov chain, we notice that the right-hand side of (62) can only have a few of expressions depending on the choices of i, j and k, *i.e.,*

$$\begin{aligned}&p(X_t = j | X_\tau = i, X_{t+1} = k)\\ &= \begin{cases} \sigma_1 := \dfrac{\alpha\left((n-1)^\delta + (n\alpha-1)^\delta(n-1)\right)}{(n-1)^\delta + (n\alpha-1)^{\delta+1}}, & \text{if } i = j = k, \\ \sigma_2 := \dfrac{(1-\alpha)\left((n-1)^\delta + (n\alpha-1)^\delta(n-1)\right)}{(n-1)^{\delta+1} - (n\alpha-1)^{\delta+1}}, & \text{if } i = j \neq k, \\ \sigma_3 := \dfrac{\alpha\left((n-1)^{\delta+1} - (n\alpha-1)^\delta(n-1)\right)}{(n-1)^{\delta+1} - (n\alpha-1)^{\delta+1}}, & \text{if } i \neq j = k, \\ \sigma_4 := \dfrac{(1-\alpha)\left((n-1)^\delta - (n\alpha-1)^\delta\right)}{\left((n-1)^\delta + (n\alpha-1)^{\delta+1}\right)(n-1)}, & \text{if } i = k \neq j, \\ \sigma_5 := \dfrac{(1-\alpha)\big((n-1)^\delta - (n\alpha-1)^\delta))}{(n-1)^{\delta+1} - (n\alpha-1)^{\delta+1}}, & \text{if } i \neq j \neq k. \end{cases}\end{aligned} \tag{63}$$

By examining σ_1 to σ_5 in (63), we have $\sigma_1 \geq \sigma_3 \geq \sigma_2 \geq \sigma_5 \geq \sigma_4$, for $\frac{1}{n} \leq \alpha \leq 1$.

For a fixed $j \in \{1, \ldots, n\}$, by counting the number of times each condition of (63), *e.g.,* $i = j = k$, $i = j \neq k$, etc., is satisfied for $i, k \in \mathcal{N}$, we can get the following ordering of n^2 probabilities $p\left(X_t = j | X_\tau = i, X_{t+1} = k\right)$ (for a fixed j):

$$\underbrace{\sigma_4 \leq \cdots \leq \sigma_4}_{n-1} \leq \underbrace{\sigma_5 \leq \cdots \leq \sigma_5}_{(n-1)(n-2)} \geq \underbrace{\sigma_2 \leq \cdots \leq \sigma_2}_{n-1} \leq \underbrace{\sigma_3 \leq \cdots \leq \sigma_3}_{n-1} \leq \sigma_1.$$

Due to the symmetry, this ordering remains the same for all $j \in \{1, \ldots, n\}$. Given this ordering for any fixed j, we can check

$$\frac{1}{R_t^O} = \sum_{x \in \mathcal{N}} p\left(X_t = x | U_t = u_{x,n^2}\right) = \sum_{x \in \mathcal{N}} \sigma_1 = n\sigma_1,$$

where $p\left(X_t = x | U_t = u_{x,n^2}\right)$ is defined in (7).

Also, from (9), we can check that $\theta_1 = n\sigma_4$, $\theta_i = 0$ for $i = 2, \ldots, n-1$, and $\theta_n = 1 - n\sigma_4$, so we have

$$\frac{1}{R_t^I} = \sum_{i=1}^{n} i\theta_i = n\sigma_4 + n - n^2\sigma_4.$$

By substituting the expression of σ_1 and σ_4 defined in (63), one can verify that $n\sigma_4 + n - n^2\sigma_4 = n\sigma_1$, which implies in $R_t^I = R_t^O$. This completes the proof of Corollary 2.

Remark 4: When $0 \leq \alpha < \frac{1}{n}$, we may follow the same steps as we did but divide the discussion into two cases: $\delta = t - \tau$ is even or odd. When δ is even, we have $\sigma_2 \geq \sigma_4 \geq \sigma_5 \geq \sigma_1 \geq \sigma_3$, and

$$\frac{1}{R_t^O} = n\sigma_2 \leq \frac{1}{R_t^I} = n\sigma_3 + n - n^2\sigma_3.$$

Similarly when δ is odd, we have $\sigma_5 \geq \sigma_4 \geq \sigma_2 \geq \sigma_3 \geq \sigma_1$, and

$$\frac{1}{R_t^O} = n\sigma_5 \leq \frac{1}{R_t^I} = \sigma_3(2n - n^2) - n\sigma_1 + n.$$

In both cases, we can see a gap between R_t^O and R_t^I, which is as per our observation in Example 1.

Appendix C
Proof of Lemma 1

First, let us recall that $\mathcal{B}_t = \{i : i \leq t, F_i = \text{ON}\} \cup \{i : i \geq t+1\}$ and $U_t = (X_\tau, X_{t+1})$ where $\tau = \max\{i : i \leq t, F_i = \text{ON}\}$.

We prove the statement by induction on t. Consider the base case $t = 0$. From the assumption $F_0 = \text{ON}$ (assumption of this paper), we know that $U_0 = \{X_0, X_1\}$ and $\mathcal{B}_0 = \{i : i = 0, 1, \ldots\}$. If Q_0 is a stochastic function of X_0 and X_1, and Q_0 is independent of X_0 and X_1, then we have

$$I\left(Q_0; X_{\mathcal{B}_0}\right) = I\left(Q_0; X_0, X_1\right) + I\left(Q_0; X_{\mathcal{B}_0} | X_1, X_0\right) = 0,$$

i.e., Q_0 is independent of $X_{\mathcal{B}_0}$. The last equality follows because Q_0 is independent of X_0 and X_1, and Q_0 is a stochastic function of X_0 and X_1.

Now, we start the inductive step. Assume that the statement is true for some $t-1$, *i.e.,* if Q_i is a stochastic function of U_i and X_i, and Q_i is independent of U_i for $i = 0, 1, \ldots, t-1$, then $Q_{[t-1]}$ is independent of $X_{\mathcal{B}_{t-1}}$.

Next, for the case t, if Q_i is a stochastic function of U_i and X_i, and Q_i is independent of U_i for $i = 0, 1, \ldots, t$, then we know from the inductive assumption that $Q_{[t-1]}$ is independent of $X_{\mathcal{B}_{t-1}}$, *i.e.,*

$$I\left(Q_{[t-1]}; X_{\mathcal{B}_{t-1}}\right) = 0. \tag{64}$$

Then consider

$$\begin{aligned}
I&\left(Q_{[t]}; X_{\mathcal{B}_t}\right) \\
&= I\left(Q_{[t-1]}; X_{\mathcal{B}_t}\right) + I\left(Q_t; U_t | Q_{[t-1]}\right) \\
&\quad + I\left(Q_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | Q_{[t-1]}, U_t\right) \\
&\overset{\text{(a)}}{\leq} I\left(Q_{[t-1]}; X_{\mathcal{B}_{t-1}}\right) + I\left(Q_t; U_t | Q_{[t-1]}\right) \\
&\quad + I\left(Q_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | Q_{[t-1]}, U_t\right) \\
&\overset{\text{(b)}}{=} I\left(Q_t; U_t | Q_{[t-1]}\right) + I\left(Q_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | Q_{[t-1]}, U_t\right) \\
&= I\left(Q_t; U_t\right) + I\left(Q_t; Q_{[t-1]} | U_t\right) - I\left(Q_t; Q_{[t-1]}\right) \\
&\quad + I\left(Q_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | Q_{[t-1]}, U_t\right) \\
&\leq I\left(Q_t; U_t\right) + I\left(Q_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}}, Q_{[t-1]} | U_t\right) \\
&\overset{\text{(c)}}{=} I\left(Q_t; U_t\right) + I\left(X_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}}, Q_{[t-1]} | U_t\right) \\
&= I\left(Q_t; U_t\right) + I\left(X_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | U_t\right) \\
&\quad + I\left(X_t; Q_{[t-1]} | X_{\mathcal{B}_t \setminus \{\tau, t+1\}}, U_t\right) \\
&\overset{\text{(d)}}{=} I\left(Q_t; U_t\right) + I\left(X_t; X_{\mathcal{B}_t \setminus \{\tau, t+1\}} | U_t\right) \\
&\overset{\text{(e)}}{=} I\left(Q_t; U_t\right) = 0,
\end{aligned}$$

where (a) follows from $\mathcal{B}_t \subseteq \mathcal{B}_{t-1}$ by inspecting the definition of $\mathcal{B}_t$, (b) follows from (64), (c) follows because Q_t is a stochastic function of U_t and X_t, (d) follows from $X_{\mathcal{B}_{t-1}} = \left\{X_{\mathcal{B}_t \setminus \{\tau, t+1\}}, U_t, X_t\right\}$ and (64), and (e) follows from the Markovity of X_t.

Appendix D
Proof of Proposition 1

From the definitions in (8) and (9), we can easily see that λ_i is non-decreasing with i, so $\theta_i \geq 0$ for all i if and only if $\lambda_{n-1} \leq 1$. It is sufficient for us to show that $\lambda_n \leq 1$.

For any given distribution $p\left(x_t | u_t\right)$ where $x_t \in \mathcal{N}$ and $u_t \in \mathcal{N}^2$, we claim that there exists some u such that

$$p\left(x_t = x | u_t = u\right) \geq p\left(x_t = x | u_t = u_{x,n}\right), \quad \forall x \in \mathcal{N}. \tag{65}$$

To see this, one can choose any $u \in \mathcal{N}^2 \setminus \{u_{x,i} : x \in \mathcal{N}, i = 1, \ldots, n-1\}$. Note that since $|\mathcal{N}^2| = n^2$ and $|\{u_{x,i} : x \in \mathcal{N}, i = 1, \ldots, n-1\}| = n(n-1)$, the set $\mathcal{N}^2 \setminus \{u_{x,i} : x \in \mathcal{N}, i = 1, \ldots, n-1\}$ is non-empty.

By summing (65) over all x, we have

$$\sum_{x \in \mathcal{N}} p\left(x_t = x | u_t = u\right) \geq \sum_{x \in \mathcal{N}} p\left(x_t = x | u_t = u_{x,n}\right) = \lambda_n.$$

Since $\sum_{x \in \mathcal{N}} p\left(x_t = x | u_t = u\right) = 1$ for a fixed u, we complete showing that $\lambda_n \leq 1$.

APPENDIX E
PROOF OF PROPOSITION 2

Recall that we need to show that for any $u = 1, \ldots, m$,

$$\sum_{x=1}^{n} \max\left\{p\left(X = x|U = u\right) - \lambda_{x,n-1}, 0\right\} \geq \sum_{\ell=1}^{n-1} \sum_{x:u\in\mathcal{U}^{-}_{\ell,x}} \left(\lambda_{x,\ell} - \lambda_{x,\ell-1}\right). \tag{66}$$

Assume without loss of generality that $u = u_{1,\alpha_1} = \cdots = u_{n,\alpha_n}$. Then, the left-hand side of (66) can be written as

$$\begin{aligned}\sum_{x=1}^{n} &\max\left\{p\left(X = x|U = u\right) - \lambda_{x,n-1}, 0\right\}\\ &= \sum_{x:\alpha_x\geq n-1} p\left(X = x|U = u\right) - \lambda_{x,n-1}\\ &= \sum_{x:\alpha_x\geq n-1} \left(\lambda_{x,\alpha_x} - \lambda_{x,n-1}\right),\end{aligned}$$

and the right-hand side of (66) can be written as

$$\begin{aligned}\sum_{\ell=1}^{n-1} \sum_{x:u\in\mathcal{U}^{-}_{\ell,x}} \lambda_{x,\ell} - \lambda_{x,\ell-1} &= \sum_{x:\alpha_x\leq n-2} \sum_{\ell=\alpha_x+1}^{n-1} \lambda_{x,\ell} - \lambda_{x,\ell-1}\\ &= \sum_{x:\alpha_x\leq n-2} \left(\lambda_{x,n-1} - \lambda_{x,\alpha_x}\right).\end{aligned}$$

Since

$$\begin{aligned}\sum_{x:\alpha_x\geq n-1} &\left(\lambda_{x,\alpha_x} - \lambda_{x,n-1}\right) - \sum_{x:\alpha_x\leq n-2} \left(\lambda_{x,n-1} - \lambda_{x,\alpha_x}\right)\\ &= \sum_{x=1}^{n} \left(\lambda_{x,\alpha_x} - \lambda_{x,n-1}\right) = 1 - \sum_{x=1}^{n} \lambda_{x,n-1} = \theta_n, \end{aligned}\tag{67}$$

we can see that (66) is established if and only if $\theta_n \geq 0$, which is given by Proposition 1. This completes the proof of Proposition 2.

Remark 5: To benefit the following proof, we give an immediate implication of (67) *here. As described, the right-hand side of* (66) *is the total values assigned for $\ell = 1, \ldots, n-1$ and the left-hand side of* (66) *is the initialization of the matrix M, so the remaining values will be assigned for $\ell = n$ as described in* (48) *and* (49)*. As such, we know from* (67) *that*

$$\sum_{x:(z,x,u)\in\mathcal{F}_n} g(z, x, u) = \theta_n, \tag{68}$$

for any $u \in \{1, \ldots, m\}$ and $z = \{1, \ldots, n\}$.

APPENDIX F
PROOF OF PROPOSITION 3

Before proving the proposition, we provide some observations of $\mathcal{F}_{\ell,x,k}$ for some x, k and $\ell = 1, \ldots, n-1$ by examining (45). Let $\mathbb{1}\{\cdot\}$ be the indicator function.

- If $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$ for some ℓ, x and k, then $\bar{z} = z_{\ell,x,k}$ is uniquely determined, *i.e.,* if $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$, then
$$\left(z', x', u'\right) \notin \mathcal{F}_{\ell,x,k}, \quad \forall z' \neq \bar{z}. \tag{69}$$
- For any $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$, $|\bar{z}| = \ell$, and $\bar{x} \in \bar{z}$.
- The cardinality of each $\mathcal{F}_{\ell,x,k}$ is $|\mathcal{F}_{\ell,x,k}| = m$. In particular, all tuples $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$ have distinct values of $\bar{u}$. In other words, let $(\bar{z}, \cdot, \cdot) \in \mathcal{F}_{\ell,x,k}$ denote that there exists some $\bar{x}, \bar{u}$ such that $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$, and then
$$\sum_{x'=1}^{n} \mathbb{1}\left\{\left(\bar{z}, x', u'\right) \in \mathcal{F}_{\ell,x,k}\right\} = \mathbb{1}\left\{\left(\bar{z}, \cdot, \cdot\right) \in \mathcal{F}_{\ell,x,k}\right\}, \tag{70}$$
for any $u' \in \{1, \ldots, m\}$.

1) The first statement is straightforward. Suppose that $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_\ell$ for some ℓ. If $\ell = 1, \ldots, n-1$, we know from (46) that $\mathcal{F}_\ell$ is the union of $\mathcal{F}_{\ell,x,k}$. For each $\mathcal{F}_{\ell,x,k}$, we know that $\bar{x} \in \bar{z}$ for any $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_{\ell,x,k}$. If $\ell = n$, $\bar{z} = \mathcal{N}$ for all $(\bar{z}, \bar{x}, \bar{u}) \in \mathcal{F}_n$, so $\bar{x} \in \bar{z}$.

2) For the second statement, when $|\bar{z}| = \ell \in \{1, \ldots, n-1\}$,
$$\begin{aligned}\sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}} &g\left(\bar{z}, x, \bar{u}\right)\\ &= \sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}_\ell} g\left(\bar{z}, x, \bar{u}\right)\\ &\overset{\text{(a)}}{=} \sum_{x} \sum_{x'=1}^{n} \sum_{k:(\bar{z},x,\bar{u})\in\mathcal{F}_{\ell,x',k}} \nu_{\ell,x',k}\\ &\overset{\text{(b)}}{=} \sum_{x'=1}^{n} \sum_{k=1}^{c_{\ell,x'}} \nu_{\ell,x',k} \cdot \mathbb{1}\left\{\left(\bar{z}, \cdot, \cdot\right) \in \mathcal{F}_{\ell,x',k}\right\}, \end{aligned}\tag{71}$$
where (a) follows from (47) and (b) follows from (70). When $|\bar{z}| = \ell = n$, *i.e.,* $\bar{z} = \{1, \ldots, n\}$, we have
$$\sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}} g\left(\bar{z}, x, \bar{u}\right) = \sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}_n} g\left(\bar{z}, x, \bar{u}\right) \overset{\text{(c)}}{=} \theta_n, \tag{72}$$
where (c) follows from (68).
Since both the right-hand sides of (71) and (72) are independent of $\bar{u}$, for any given $\bar{z}$, $\bar{u}$ and $\bar{u}'$
$$\sum_{x:(\bar{z},x,\bar{u})\in\mathcal{F}} g\left(\bar{z}, x, \bar{u}\right) = \sum_{x:(\bar{z},x,\bar{u}')\in\mathcal{F}} g\left(\bar{z}, x, \bar{u}'\right).$$

3) As for the third statement, for any given $\bar{x}$ and $\bar{u}$,
$$\begin{aligned}\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}} &g\left(z, \bar{x}, \bar{u}\right)\\ &= \sum_{\ell=1}^{n} \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z, \bar{x}, \bar{u}\right)\\ &= \sum_{\ell=1}^{n-1} \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z, \bar{x}, \bar{u}\right) + \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_n} g\left(z, \bar{x}, \bar{u}\right). \end{aligned}\tag{73}$$

For the first term of (73), we have
$$\begin{aligned}\sum_{\ell=1}^{n-1} &\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z, \bar{x}, \bar{u}\right)\\ &\overset{\text{(a)}}{=} \sum_{\ell=1}^{n-1} \sum_{z} \sum_{x=1}^{n} \sum_{k:(z,\bar{x},\bar{u})\in\mathcal{F}_{\ell,x,k}} \nu_{\ell,x,k}\end{aligned}$$

$$\overset{(b)}{=} \sum_{\ell=1}^{n-1}\sum_{x=1}^{n}\sum_{k:(\cdot,\bar{x},\bar{u})\in\mathcal{F}_{\ell,x,k}} \nu_{\ell,x,k}, \tag{74}$$

where (a) follows by substituting (47), and (b) follows from (69).

By examining $\mathcal{F}_{\ell,x,k}$, we can see two disjoint subsets,

$$\mathcal{F}^{-}_{\ell,x,k} := \{(\bar{z},\bar{x},\bar{u}) : \bar{z} = z_{\ell,x,k}, \bar{x} = \zeta_{\ell,x,k}(i), \\ \bar{u} = u_{x,i} \in \mathcal{U}^{-}_{\ell,x}\},$$

and

$$\mathcal{F}^{+}_{\ell,x,k} := \left\{(\bar{z},\bar{x},\bar{u}) : \bar{z} = z_{\ell,x,k}, \bar{x} = x, \bar{u} \in \mathcal{U}^{+}_{\ell,x}\right\},$$

For a fixed $\bar{u}$, assume that $\bar{u} = u_{1,\alpha_1} = \cdots = u_{n,\alpha_n}$. Then, we write (74) as

$$\begin{aligned}
&\sum_{\ell=1}^{n-1}\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z,\bar{x},\bar{u}\right)\\
&= \sum_{\ell=1}^{n-1}\sum_{x=1}^{n}\sum_{k:(\cdot,\bar{x},\bar{u})\in\mathcal{F}_{\ell,x,k}} \nu_{\ell,x,k}\\
&= \sum_{\ell=1}^{n-1}\sum_{x:\alpha_x\le\ell-1}\sum_{k:\bar{x}=\zeta_{\ell,x,k}(\alpha_x)} \nu_{\ell,x,k} + \sum_{\ell=1}^{n-1}\sum_{k:\alpha_{\bar{x}}\ge\ell} \nu_{\ell,\bar{x},k}\\
&= \sum_{\ell=1}^{n-1}\sum_{x:\alpha_x\le\ell-1}\sum_{k:\bar{x}=\zeta_{\ell,x,k}(\alpha_x)} \nu_{\ell,x,k} + \sum_{\ell=1}^{\min\{n-1,\alpha_{\bar{x}}\}}\sum_{k} \nu_{\ell,\bar{x},k}.
\end{aligned} \tag{75}$$

From (42), we know that $\sum_k \nu_{\ell,\bar{x},k} = \lambda_{\bar{x},\ell} - \lambda_{\bar{x},\ell-1}$, and hence the second term of (75) can be written as

$$\sum_{\ell=1}^{\min\{n-1,\alpha_{\bar{x}}\}}\sum_{k} \nu_{\ell,\bar{x},k} = \lambda_{\bar{x},\min\{n-1,\alpha_{\bar{x}}\}}. \tag{76}$$

For the first term of (75), we know from (41) that

$$\sum_{k:\zeta_{\ell,x,k}(\alpha_x)=\bar{x}} \nu_{\ell,x,k} = \upsilon_{\ell,x,\alpha_x,j},$$

and $\bar{x} = e_{\ell,x,i,j}$ for some j, where $\upsilon_{\ell,x,\alpha_x,j}$ and $e_{\ell,x,i,j}$ are defined in (36). Then, we know from (39) that

$$\sum_{k:\zeta_{\ell,x,k}(\alpha_x)=\bar{x}} \nu_{\ell,x,k} = \upsilon_{\ell,x,\alpha_x,j} = M^{-}_{u_{x,\alpha_x},\bar{x}} = M^{-}_{\bar{u},\bar{x}},$$

i.e., the value subtracted from $M_{\bar{u},\bar{x}}$ for given ℓ and x. Thus, we have

$$\begin{aligned}
&\sum_{\ell=1}^{n-1}\sum_{x:\alpha_x\le\ell-1}\sum_{k:\bar{x}=\zeta_{\ell,x,k}(\alpha_x)} \nu_{\ell,x,k}\\
&= \sum_{\ell=1}^{n-1}\sum_{x:\alpha_x\le\ell-1} M^{-}_{\bar{u},\bar{x}}\\
&= \sum_{\ell=1}^{n-1}\sum_{x:\bar{u}\in\mathcal{U}^{-}_{\ell,x}} M^{-}_{\bar{u},\bar{x}},
\end{aligned} \tag{77}$$

i.e., all values subtracted from $M_{\bar{u},\bar{x}}$ for $\ell = 1,\ldots,n-1$ and all x. By substituting (77) and (76) in (75), we have

$$\begin{aligned}
&\sum_{\ell=1}^{n-1}\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z,\bar{x},\bar{u}\right)\\
&\qquad= \sum_{\ell=1}^{n-1}\sum_{x:\bar{u}\in\mathcal{U}^{-}_{\ell,x}} M^{-}_{\bar{u},\bar{x}} + \lambda_{\bar{x},\min\{n-1,\alpha_{\bar{x}}\}}.
\end{aligned} \tag{78}$$

Then, substituting (78) in (73), we have

$$\begin{aligned}
&\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}} g\left(z,\bar{x},\bar{u}\right)\\
&\quad= \sum_{\ell=1}^{n-1}\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_\ell} g\left(z,\bar{x},\bar{u}\right) + \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_n} g\left(z,\bar{x},\bar{u}\right)\\
&\quad= \sum_{\ell=1}^{n-1}\sum_{x:\bar{u}\in\mathcal{U}^{-}_{\ell,x}} M^{-}_{\bar{u},\bar{x}} + \lambda_{\bar{x},\min\{n-1,\alpha_{\bar{x}}\}} +\\
&\qquad\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_n} g\left(z,\bar{x},\bar{u}\right).
\end{aligned}$$

Recalling that we assign all the remaining values $M_{\bar{u},\bar{x}}$ to $g\left(z,\bar{x},\bar{u}\right)$ in (48) and (49) when $\ell = n$, we know that

$$\begin{aligned}
&\sum_{\ell=1}^{n-1}\sum_{x:\bar{u}\in\mathcal{U}^{-}_{\ell,x}} M^{-}_{\bar{u},\bar{x}} + \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_n} g\left(z,\bar{x},\bar{u}\right)\\
&\qquad= \max\left\{p\left(X=\bar{x}|U=\bar{u}\right) - \lambda_{\bar{x},n-1}, 0\right\},
\end{aligned}$$

i.e., the initial value of $M_{\bar{u},\bar{x}}$ defined in (35). Therefore,

$$\begin{aligned}
&\sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}} g\left(z,\bar{x},\bar{u}\right)\\
&\quad= \sum_{\ell=1}^{n-1}\sum_{x:\bar{u}\in\mathcal{U}^{-}_{\ell,x}} M^{-}_{\bar{u},\bar{x}} + \lambda_{\bar{x},\min\{n-1,\alpha_{\bar{x}}\}}\\
&\qquad+ \sum_{z:(z,\bar{x},\bar{u})\in\mathcal{F}_n} g\left(z,\bar{x},\bar{u}\right)\\
&\quad= \max\left\{p\left(X=\bar{x}|U=\bar{u}\right) - \lambda_{\bar{x},n-1}, 0\right\}\\
&\qquad+ \lambda_{\bar{x},\min\{n-1,\alpha_{\bar{x}}\}}\\
&\quad= \max\left\{p\left(\bar{x}|\bar{u}\right) - \lambda_{\bar{x},n-1}, 0\right\} + \min\left\{\lambda_{\bar{x},n-1}, p\left(\bar{x}|\bar{u}\right)\right\}\\
&\quad= p\left(\bar{x}|\bar{u}\right),
\end{aligned}$$

which completes the proof.

Appendix G
Proof of Proposition 4

The proof is quite straightforward from previous intermediate steps. As we know that $p\left(z|u\right) = p\left(z|u'\right)$ for any $u, u' \in \mathcal{N}^2$ and $z \in \mathcal{Z}$ from Proposition 3, for any given $\ell \in \{1,\ldots,n-1\}$, we have

$$\sum_{z:|z|=\ell} p\left(z\right) = \sum_{z:|z|=\ell} p\left(z|u\right) = \sum_{z:|z|=\ell}\sum_{x:(z,x,u)\in\mathcal{F}} g\left(z,x,u\right)$$

$$
\begin{aligned}
&\overset{(a)}{=} \sum_{z:|z|=\ell}\sum_{x'=1}^{n}\sum_{k=1}^{c_{\ell,x'}} \nu_{\ell,x',k}\cdot \mathbb{1}\left\{(z,\cdot,\cdot)\in\mathcal{F}_{\ell,x',k}\right\}\\
&= \sum_{x'=1}^{n}\sum_{k=1}^{c_{\ell,x'}} \nu_{\ell,x',k}\cdot \sum_{z:|z|=\ell}\mathbb{1}\left\{(z,\cdot,\cdot)\in\mathcal{F}_{\ell,x',k}\right\}\\
&\overset{(b)}{=} \sum_{x'=1}^{n}\sum_{k=1}^{c_{\ell,x'}} \nu_{\ell,x',k} \overset{(c)}{=} \sum_{x'=1}^{n}\lambda_{x',\ell}-\lambda_{x',\ell-1}=\theta_\ell,
\end{aligned}
$$

where (a) follows from (71), (b) follows from (69), and (c) follows from (42). For $\ell = n$, we have

$$
\sum_{z:|z|=n} p(z) = 1-\sum_{\ell=1}^{n-1}\sum_{z:|z|=\ell} p(z) = 1-\sum_{\ell=1}^{n-1}\theta_\ell = \theta_n
$$

by definition, then $p(|Z|=\ell)=\theta_\ell$, for all $\ell = 1,\ldots,n$.

Appendix H
Proof of Lemma 2

Consider

$$
\begin{aligned}
\max_{u\in\mathcal{N}^2} p(x|u) &\overset{(a)}{=} \max_{u\in\mathcal{N}^2}\sum_{y:x\in y} p(x,y|u)\\
&= \max_{u\in\mathcal{N}^2}\sum_{y:x\in y} p(y|u)\,p(x|y,u)\\
&\overset{(b)}{=} \max_{u\in\mathcal{N}^2}\sum_{y:x\in y} p(y)\,p(x|y,u)\\
&\le \sum_{y:x\in y} p(y)\max_{u\in\mathcal{N}^2} p(x|y,u)\\
&\le \sum_{y:x\in y} p(y),
\end{aligned}
$$

where (a) follows from $p(x,y|u)=0$ for $x\notin y$, and (b) follows because Y is independent of U. Thus, we obtain that

$$
\begin{aligned}
\sum_{x\in\mathcal{N}}\max_{u\in\mathcal{N}^2} p(x|u) &\le \sum_{x\in\mathcal{N}}\sum_{y:x\in y} p(y) = \sum_{y\in\mathscr{P}(\mathcal{N})}\sum_{x:x\in y} p(y)\\
&= \sum_{y\in\mathscr{P}(\mathcal{N})} p(y)\sum_{x:x\in y} 1 = \mathbb{E}\left[|Y|\right],
\end{aligned}
$$

which completes the proof.

References

[1] L. Sweeney, "K-Anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, Oct. 2002.

[2] C. Dwork, "Differential privacy," in *Proc. 33rd Int. Colloq. Automata, Lang. Program. (ICALP)*, Venice, Italy, 2006, pp. 1–12.

[3] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," in *Proc. IEEE Symp. Found. Comput. Sci. (FOCS)*, Oct. 1995, pp. 41–50.

[4] L. Sankar, S. R. Rajagopalan, and H. V. Poor, "Utility-privacy tradeoffs in databases: An information-theoretic approach," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 6, pp. 838–852, Jun. 2013.

[5] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, "Protecting data privacy in private information retrieval schemes," *J. Comput. Syst. Sci.*, vol. 60, no. 3, pp. 592–629, Jun. 2000.

[6] M. Bezzi, "An information theoretic approach for privacy metrics," *Trans. Data Privacy*, vol. 3, no. 3, pp. 199–215, Dec. 2010.

[7] E. Nekouei, T. Tanaka, M. Skoglund, and K. H. Johansson, "Information-theoretic approaches to privacy in estimation and control," *Annu. Rev. Control*, vol. 47, pp. 412–422, Jan. 2019.

[8] F. du Pin Calmon and N. Fawaz, "Privacy against statistical inference," in *Proc. 50th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Oct. 2012, pp. 1401–1408.

[9] S. Kadhe, B. Garcia, A. Heidarzadeh, S. El Rouayheb, and A. Sprintson, "Private information retrieval with side information," *IEEE Trans. Inf. Theory*, vol. 66, no. 4, pp. 2032–2043, Apr. 2020.

[10] S. Li and M. Gastpar, "Single-server multi-message private information retrieval with side information," in *Proc. 56th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Oct. 2018, pp. 173–179.

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

[12] N. B. Shah, K. V. Rashmi, and K. Ramchandran, "One extra bit of download ensures perfectly private information retrieval," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2014, pp. 856–860.

[13] R. Tajeddine, O. W. Gnilke, and S. El Rouayheb, "Private information retrieval from MDS coded data in distributed storage systems," *IEEE Trans. Inf. Theory*, vol. 64, no. 11, pp. 7081–7093, Nov. 2018.

[14] R. Freij-Hollanti, O. W. Gnilke, C. Hollanti, and D. A. Karpuk, "Private information retrieval from coded databases with colluding servers," *SIAM J. Appl. Algebra Geometry*, vol. 1, no. 1, pp. 647–664, Jan. 2017.

[15] K. Banawan and S. Ulukus, "The capacity of private information retrieval from coded databases," *IEEE Trans. Inf. Theory*, vol. 64, no. 3, pp. 1945–1956, Mar. 2018.

[16] R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, and J.-P. Hubaux, "Unraveling an old cloak: K-anonymity for location privacy," in *Proc. 9th Annu. ACM Workshop Privacy Electron. Soc. (WPES)*, Oct. 2010, pp. 115–118.

[17] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 1–18, Jan. 2008.

[18] J. Hua, W. Tong, F. Xu, and S. Zhong, "A geo-indistinguishable location perturbation mechanism for location-based services supporting frequent queries," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 5, pp. 1155–1168, May 2018.

[19] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "A predictive differentially-private mechanism for mobility traces," in *Privacy Enhancing Technologies*. Amsterdam, The Netherlands: Springer, 2014, pp. 21–41.

[20] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, Denver, CO, USA, Oct. 2015, pp. 1298–1309.

[21] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *Proc. IEEE Symp. Secur. Privacy*, Berkeley, CA, USA, May 2011, pp. 247–262.

[22] R. Shokri, G. Theodorakopoulos, and C. Troncoso, "Privacy games along location traces," *ACM Trans. Privacy Secur.*, vol. 19, no. 4, pp. 1–31, Feb. 2017.

[23] E. Erdemir, P. L. Dragotti, and D. Gunduz, "Privacy-aware time-series data sharing with deep reinforcement learning," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 389–401, 2021.

[24] W. Zhang, M. Li, R. Tandon, and H. Li, "Online location trace privacy: An information theoretic approach," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 1, pp. 235–250, Jan. 2019.

[25] C. Naim, F. Ye, and S. E. Rouayheb, "ON-OFF privacy with correlated requests," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France Jul. 2019, pp. 817–821.

[26] F. Ye, C. Naim, and S. E. Rouayheb, "Preserving ON-OFF privacy for past and future requests," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Visby, Sweden, Aug. 2019, pp. 1–5.

[27] F. Ye, H. Cho, and S. El Rouayheb, "Mechanisms for hiding sensitive genotypes with information-theoretic privacy," 2020, *arXiv:2007.05139*. [Online]. Available: http://arxiv.org/abs/2007.05139

[28] F. Ye, C. Naim, and S. El Rouayheb, "ON-OFF privacy in the presence of correlation," 2020, *arXiv:2004.04186*. [Online]. Available: http://arxiv.org/abs/2004.04186

[29] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*. Hoboken, NJ, USA: Wiley, 2011.

[30] P. M. Vaidya, "Speeding-up linear programming using fast matrix multiplication," in *Proc. 30th Annu. Symp. Found. Comput. Sci.*, Research Triangle Park, NC, USA, 1989, pp. 332–337.