The Economics of Privacy and Utility: Investment Strategies

Chandra Sharma, George Amariucai, and Shuangqing Wei, Senior Member, IEEE

Abstract—The inevitable leakage of privacy as a result of unrestrained disclosure of personal information has motivated extensive research on robust privacy-preserving mechanisms. However, existing research is mostly limited to solving the problem in a static setting with disregard for the privacy leakage over time. Unfortunately, this treatment of privacy is insufficient in practical settings where users continuously disclose their personal information over time resulting in an accumulated leakage of the users’ sensitive information. In this paper, we consider privacy leakage over a finite time horizon and investigate optimal strategies to maximize the utility of the disclosed data while limiting the finite-horizon privacy leakage. We consider a simple privacy mechanism that involves compressing the user’s data before each disclosure to meet the desired constraint on future privacy. We further motivate several algorithms to optimize the dynamic privacy-utility tradeoff and evaluate their performance via extensive synthetic performance tests.

Index Terms—Dynamic privacy, utility, finite horizon, Kalman filter, Bellman equation.

I. INTRODUCTION

The unprecedented growth of big-data applications suggests that there is a growing competition in the technological world to collect and harness tremendous amounts of user information. Tech companies and other online service providers are always seeking to enhance the quality of their products and services by collecting massive amounts of information from their user base. Users often disclose their personal information either by directly engaging with the service providers, such as in the case of social media and online shopping, or indirectly, and often inadvertently, by simply possessing IoT and other smart devices, such as location trackers.

Users are often oblivious to the actual scale and nature of the information disclosure. The disclosed information often contains sensitive information about the users such as their current location, income, religious beliefs and sexual orientation.

Further, service providers often share, and even sell, their customers’ information with third parties, which makes protecting the users’ private information ever so critical. In light of this, there have been increasing efforts to devise privacy-preserving mechanisms that make it difficult for external entities to infer a user’s private information from the disclosed data. Such mechanisms protect the user’s information often via means of distortion [1], [2], [3], [4], [5], [6], [7], [8] and/or compression [9], [10], [11], [12], [13] before disclosure. Unfortunately, the introduction of distortion (and/or compression) entails a loss of some useful information from the disclosed data which can otherwise be utilized by the service provider to provide a customized service to the user. The challenge, therefore, is to find an optimal tradeoff between protecting the user’s privacy and enhancing the utility of the disclosed data.

Existing works that address the tradeoff between privacy and utility mostly treat privacy differently based on the context. In the context of data analysis, privacy is related to an adversary’s ability to identify a user or the user’s sensitive attribute values in a database (for instance, [14], [15], [16], [17], [18], [19]), whereas in the context of information flow (over a noisy channel), privacy is related to an adversary’s increase in knowledge about a user’s sensitive attribute values given some observable data (for instance, [2], [14], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31]). While existing works try to capture different notions of privacy and derive theoretical bounds on the privacy leakage, they do so in a static setting which either assumes that a user discloses their information only once, or it treats each disclosure of the user’s information independently of the previous disclosures. This model of privacy does not accurately reflect real world settings in which users continuously disclose their personal information over time (as in social media) and each disclosure is temporally correlated with the previous disclosures. Therefore, privacy models under static settings are not only incomplete but also inaccurate for many practical applications.

In this paper, we consider a dynamic setting where users continuously disclose their personal information, resulting in a gradual release of sensitive data. Within this context, we introduce a privacy-conscious user who carefully shares her personal information with a data analyst over a finite time period. The user’s primary objective is to optimize her immediate benefits, which the data analyst generates by extracting valuable insights from the disclosed information at each step. Simultaneously, the user aims to restrict the extent of sensitive data leakage at the conclusion of this time...
horizon. Our model is primarily tailored for scenarios where users need to engage in specific activities for a predetermined duration and require a strategic approach to minimize privacy leakage while optimizing utility. As an illustrative example, let’s consider a user contemplating the use of a navigation service for an upcoming trip. During the course of using the service, the user may be required to disclose real-time location and other pertinent information. To make an informed decision regarding the potential privacy risks associated with employing the service, the user must carefully evaluate the extent to which her privacy may be compromised. Notably, the assessment of privacy exposure at the conclusion of the usage period holds particular significance. This is because private attributes might undergo changes during the engagement period. For instance, the user’s location (deemed a sensitive feature in this context) is likely to change during the course of navigation. From the user’s perspective, ensuring the confidentiality of her final location is of utmost importance.

Some other practical applications of our privacy model are highlighted below.

1) **Fitness Trackers**: Many popular fitness trackers are known to continuously collect user information for analysis. Such practices raise concerns related to user privacy, specifically, how the information collected at each timestamp contributes to the additional leakage of sensitive information about the user. With the aid of our privacy model, users have the ability to define the specific duration for which they plan to participate in a particular activity (for instance, a month of running) and are willing to share their activity data for analytical purposes. The model then formulates effective strategies that protect their sensitive information at the end of the activity (for instance, their final weight) while also delivering immediate benefits, such as precise daily tracking of calorie expenditure.

2) **Financial Services**: Users may wish to share financial data with institutions such as banks and investment firms for services like credit scoring or investment advice. Our model enables users to set specific timeframes during which their financial data is collected and utilized to enhance service quality. Subsequently, our model can generate the most effective strategies that align with their privacy and utility requirements. For example, it can formulate strategies that help keep their wealth (sensitive data) confidential at the end of the period while also ensuring they receive the best financial advice (utility) during that period.

3) **Location Tracking and Geospatial Data**: Location services, such as Find my Device, often require users to share their real-time location with apps and services. Users may be concerned about the extent of location data collected and how it could be used. Using our privacy model, users have the ability to protect their privacy while utilizing the benefits of such services. For instance, a user may want to find their phone if they lose it during a morning walk, but they may not wish to allow location tracking to be exploited to infer their home location. By specifying the duration of location tracking (e.g., the duration of the walk), our model can generate strategies that can be employed to conceal the device’s true location at the end of this duration while simultaneously providing immediate benefits, such as being able to locate a lost phone during the walk.

It is important to note that in the context of location tracking, the initial and final locations of a device could coincide. Our privacy model is specifically designed to consider all previously shared data and the temporal correlations between these data points to minimize privacy breaches at the conclusion of an engagement period. All data disclosed during this period are intentionally distorted to both uphold privacy requirements and deliver immediate benefits to the user.

4) **Education and Online Learning**: Students may have concerns about how their data, such as submitted assignments and discussion posts, are used by online learning platforms. Before fully engaging with a learning platform, they may carefully weigh the potential risks against the benefits and choose to selectively disclose information. With the assistance of our privacy model, they can proactively develop strategies to optimize their benefits from the platform while safeguarding sensitive information, such as their opinions on politically sensitive topics.

In each of the aforementioned scenarios and beyond, our privacy model is designed to provide users with immediate utilities during the data collection period while ensuring that their privacy remains protected at the conclusion of the collection period.

The privacy model discussed in this paper has a conceptual connection to the investment problem in economics. In both cases, individuals are faced with decision-making scenarios where they must balance competing interests. Just as investors seek to optimize their financial returns while managing risks, users of the privacy model aim to maximize the utility and benefits derived from utilizing data-based services while minimizing the potential privacy risks and exposure. The privacy model acts as a strategic tool for users to invest their data wisely, much like an investor allocates resources to different assets to achieve specific financial goals. Both scenarios involve the need for thoughtful and strategic decision-making to achieve a balance between benefits and risks, highlighting the parallel between risk management and maximizing financial returns in investment, and managing privacy leakage while maximizing utility in the context of data sharing.

Overall, the main contributions of the paper are as follows:

1) We formulate a novel privacy-utility tradeoff problem capturing the dynamics of privacy leakage over a finite time horizon. Our dynamic model of privacy also captures a practical setting in which a user’s perception of their own privacy may change over time.

2) Under our privacy-utility tradeoff model, we examine a privacy mechanism that involves compressing the user’s data before each disclosure and investigate different strategies that allow the user to maximize their net utility subject to certain privacy requirements.
3) We discuss challenges associated with finding optimal strategies for real world problems and motivate sub-optimal algorithms to solve the tradeoff problem.

4) We extensively evaluate the performance of the sub-optimal algorithms on synthetic datasets and demonstrate that despite being sub-optimal, the proposed algorithms perform extremely well in achieving a good privacy-utility tradeoff.

5) We formulate a simpler dynamic privacy problem that is computationally less intensive to solve but conserves the essence of the original problem.

The rest of the paper is organized as follows. In Section II, we briefly review existing works closely related to this paper. In Section III, we discuss the problem setting, explore the privacy and utility requirements and motivate the finite-horizon privacy-utility tradeoff optimization problem. In Section IV, we formulate the finite-horizon privacy-utility tradeoff problem as a Markov Decision Process problem. In Section V, we discuss the challenges associated with solving the optimization problem and motivate sub-optimal algorithms. In Section VI we discuss a simpler model problem, highlight its advantages and present a computationally less intensive algorithm to solve the simplified problem. In Section VII, we evaluate the performance of all algorithms on synthetic datasets. Finally, in Section VIII, we sum up our work with closing remarks.

II. RELATED WORK

The problem of optimizing the privacy-utility tradeoff in a static setting has been widely studied under different notions of privacy. Existing works mostly focus on precisely defining privacy based on the context and deriving bounds on the privacy leakage using metrics such as KL-divergence [32], Differential Privacy [14], [15], [16], Correlated Differential Privacy [17], [18], Blowfish Privacy [19], Mutual Information [2], [6], [20], [21], [33], Changes in min-entropy [14], [23], [24], Fisher Information [25], [26], [27], Maximal Leakage [28], [29], \( \chi^2 \)-information [34] and Maximal Correlation [20], [30], [31]). However, most existing works do not model a continuous and correlated disclosure of information and therefore, do not capture the privacy leakage over time.

There are but a few works that model the continuous disclosure of a user’s information and capture the temporal correlation between subsequent disclosures. In [35], the authors investigate the privacy leakage resulting from a continuous release of a time-series data that is correlated, both temporally and spatially, with a user’s sensitive data (also considered to be time-series but non-disclosable). The privacy mechanism seeks to distort the time series data before each disclosure to impede inference attacks on the sensitive data while preserving the utility of the disclosed data. This model seeks to limit the privacy leakage at the present time using the information from the past disclosures and by carefully regulating the current disclosure (other similar models can be found in [36], [37], [38]). In contrast, our model seeks to limit the privacy leakage in the future using the information from the past disclosures and by carefully regulating the present and all future disclosures. Due to the uncertainties around the future observations, our model is more complex, but also more general, and easily simplifies to a model similar to that in [35] under a particular instantiation (namely, \( n = k \) where \( n \) represents the finite time horizon and \( k \) represents the current time step).

Recently, researchers have started to explore the privacy issues in a dynamic setting with regard to future privacy leakage. In [39], the authors investigate the privacy issues related with the continuous disclosure of sensor measurements containing some private and some public information. They formulate the problem as a filtering problem in which they seek to find the optimal compression that maximizes the variance of the estimation error associated with the estimation of the private data while minimizing the variance of the estimation error associated with the estimation of the public data. Under their model, they investigate the privacy-utility tradeoff at the current time step and one time step into the future. The same work is further extended in [13] where the authors investigate the tradeoff multiple time steps into the future. In their formulation, the authors make predictions about the system’s future state using the observations available up until the current time step. This resembles a setting in which a user’s past and present actions are considered to estimate their future privacy leakage. However, the user’s future actions are not considered in making the prediction; therefore, while the model is relatively simple, it is not quite complete as it does not consider the impact of a user’s future actions on their future privacy leakage. In contrast, our dynamic privacy models explicitly account for a user’s past, present as well as future actions in evaluating their future privacy leakage. Further, as a direct consequence of incorporating inherently uncertain future observations in the problem formulation, our problem formulation features probabilistic models of privacy and utility constraints as opposed to the deterministic models featured in [13].

III. PROBLEM DESCRIPTION

A. Problem Setup

We consider a setting where a privacy-aware user seeks to cautiously disclose her personal information to a data analyst over a finite time horizon. The objective of the user is to maximize her instantaneous utilities, which the data analyst provides by extracting useful information from the disclosed data at each time step, while limiting the amount of leakage about her sensitive information at the end of the finite time horizon. Due to the compounding nature of privacy leakage, at each time step from zero to the finite time horizon, the privacy leakage about the private value, as observed at the finite time horizon, can only increase. Hence, the constraint on the privacy leakage at the end of the finite time horizon also implicitly imposes a constraint on the privacy leakage at each time step before the end of the finite horizon.

In contrast to the static setting which models the information disclosure at a single time step, the dynamic setting under consideration models the incremental disclosure of information at every time step until the end of the finite time horizon.
The solution to this dynamic privacy problem involves finding an optimal strategy that maximizes the sum of instantaneous utilities while ensuring that the privacy leakage at the end of the finite horizon remains below a pre-specified threshold with high probability.

In the dynamic privacy setting, we assume that each user has a set of features, represented by the random vector $X$, which evolves over time. We use the subscript $k$ to denote the feature vector at time step $k$. At any given time step $k$, the feature vector consists of the user’s private features represented by the random vector $X^p$ and the user’s public features represented by the random vector $X^u$ such that $X^u = X^P \cup X^U$. We consider a general setting where $X^P$ and $X^U$ are correlated. In many real world settings, the user’s observations of her own feature vectors are only available as noisy measurements (for instance, heart-rate readings from a smart watch). To model this, we assume that the true values of $X^u$ (and consequently, $X^P$ and $X^U$) may not be directly observable; instead, there is an observable process $Z_k$ that carries information about $X^u$ such that $Z_k = f(X^u)$. Now for some (stochastic) function $f(\cdot)$. The user’s instantaneous privacy and utility are directly associated with $X^P$ and $X^U$, respectively. Next, we assume that the user is willing to disclose $Z_k$ in return for some utility. However, as $Z_k$ contains information about both $X^P$ and $X^U$, disclosing $Z_k$ inevitably leaks some information about $X^P$, and this leakage carries over to the future time-steps which the user seeks to avoid. To address this, we consider a privacy mechanism which perturbs $Z_k$ before disclosure. The privacy mechanism involves transforming the entire observation vector, $Z_k$, into a lower-dimensional noisy vector, $	ilde{Z}_k$. The transformation is essentially non-invertible and therefore, certain information about $Z_k$ (and consequently, $X_k$) is lost due to the compression. An ideal transformation function maximizes the information loss regarding $X^P$ while minimizing the information loss regarding $X^u$. However, due to the correlation between $X^P$ and $X^U$, this may not always be possible.

A Linear Dynamical System (LDS), which is a continuous state-space generalization of a Hidden Markov Model, can be used to model the evolution of the user’s feature vectors over time as well as the observation process. LDS has been widely used to model the underlying system in the context of privacy-preserving information disclosure [13], [39], [40], [41], [42]. We consider a first-order LDS model in which $X_k$ evolves according to the linear equation:

$$X_k = F_k X_{k-1} + W_k,$$

where $F_k$ is the state-transition matrix and $W_k$ is a zero-mean Gaussian process noise with covariance $Q_k$. Similarly, the observation process can be represented by the linear equation:

$$Z_k = H_k X_k + V_k,$$

where $H_k$ denotes the observation matrix and $V_k$ denotes another zero-mean Gaussian measurement noise with covariance $R_k$. We consider both $F_k$ and $H_k$ to be full-ranked square matrices and assume that all system parameters are publicly known. For quick reference, the description of all system parameters, system vectors and other symbols used throughout this paper can be found in Table I.

The privacy mechanism involves mapping the observation vector $Z_k$ to a lower-dimensional vector $\tilde{Z}_k$ using a compression matrix $C_k$ such that $\tilde{Z}_k = C_k^T Z_k$. Although most of the recent privacy work relies on randomization-based privacy mechanisms, we should note that compression is still the most widely-used privacy mechanism in practice, especially in its simplified information-withholding form. We assume that an adversary has complete knowledge about the system dynamics. The goal of the data-owner (user) is to prudently select the compression matrices, $C_1, C_2, \cdots, C_n$, that maximize the sum of instantaneous utilities while limiting the amount of information leaked about the private features at the end of the finite time horizon, $n$. Note that the sequence $C_1, C_2, \cdots, C_n$ constitutes the strategy for the data-owner. The data analyst is tasked with inferring $X^u$ from the disclosed sequence $\tilde{Z}_1, \tilde{Z}_2, \cdots, \tilde{Z}_k$ at each time step $k$ while the future adversary seeks to infer $X^u$ from the disclosed sequence $\tilde{Z}_1, \tilde{Z}_2, \cdots, \tilde{Z}_n$. The problem, therefore, naturally relates to an estimation problem. The dynamics of the problem are depicted in Figure 1.

Before discussing the formal models of privacy and utility, we first focus on the problem of estimating the latent system states, $X^P_k$ and $X^U_k$, given a series of observations. This estimation problem can be solved using the Kalman filter which is an optimal linear filter in terms of minimizing the Mean Squared Error of the estimates [43]. Estimation using the Kalman filter involves two steps: the prediction step in which the system states are predicted a priori and the update step in which the

| Symbol | Description |
|--------|-------------|
| $n$    | Finite time horizon |
| $N_p$  | Number of private features |
| $N_u$  | Number of utility features |
| $N$    | Total number of features ($N = N_p + N_u$) |
| $M$    | Compression size ($M < N$) |
| $Z_k$  | Random vector representing the user’s features ($X_k \in \mathbb{R}^{N \times 1}$) |
| $\tilde{Z}_k$ | Observation vector ($\tilde{Z}_k \in \mathbb{R}^{M \times 1}$) |
| $\hat{Z}_k$ | Compressed observation vector ($\hat{Z}_k \in \mathbb{R}^{M \times 1}$) |
| $F_k$  | State-transition matrix ($F_k \in \mathbb{R}^{N \times N}$) |
| $H_k$  | Observation matrix ($H_k \in \mathbb{R}^{N \times N}$) |
| $Q_k$  | The covariance of the process noise ($Q_k \in \mathbb{R}^{N \times N}$) |
| $R_k$  | The covariance of the measurement noise ($R_k \in \mathbb{R}^{N \times N}$) |
| $C_k$  | Compression matrix ($C_k \in \mathbb{R}^{N \times M}$) |
| $X^P_k$ | Random vector representing the user’s true private feature at time $k$ ($X^P_k \in \mathbb{R}^{N_p \times 1}$) |
| $X^U_k$ | Random vector representing the user’s true public (utility) feature at time $k$ ($X^U_k \in \mathbb{R}^{N_u \times 1}$) |
| $X^P_{ij}$ | Data owner's estimation of $X^P_{ij}$ using observations $Z_1, Z_2, \cdots, Z_i$ ($k = j$) |
| $X^U_{ij}$ | Data owner's estimation of $X^U_{ij}$ using observations $Z_1, Z_2, \cdots, Z_i$ ($k = j$) |
| $\tilde{X}^P_{ij}$ | Adversary's estimation of $X^P_{ij}$ using observations $\tilde{Z}_1, \tilde{Z}_2, \cdots, \tilde{Z}_i$ ($k = j$) |
| $\tilde{X}^U_{ij}$ | Data analyst's estimation of $X^U_{ij}$ using observations $\tilde{Z}_1, \tilde{Z}_2, \cdots, \tilde{Z}_i$ ($k = j$) |
| $P_{ij}$ | Error covariance associated with the data owner's estimate of $X^P_{ij}$ |
| $P^2_{ij}$ | Error covariance associated with the data analyst and adversary's estimate of $X^U_{ij}$ |
| $P(X|Y)$ | The conditional probability of $X$ given $Y$ |
estimates, \( \hat{X}_{k|k}^u \) and \( \hat{X}_{k|k}^Z \), is achievable if the data owner discloses her true observations, \( Z_1, Z_2, \ldots, Z_k \), with no privacy mechanisms. Similarly, from the future privacy point of view, it is desirable that \( d(\hat{X}_{k|k}^u, \hat{X}_{k|k}^Z) \) is as large as possible. Here, the distance function is chosen to quantify privacy over other metrics, such as error covariance, because in many practical applications, it offers a more intuitive measure of privacy. For example, in the context of location tracking, which involves monitoring the movements of individuals or devices over time, a distance metric provides a quantitative means to assess the accuracy of an attacker’s location estimate. By measuring the distance between the estimated location and the true location, it provides a clear numerical value that represents the level of privacy risk. Smaller distances imply lower privacy, while larger distances indicate greater privacy.

Due to the correlation between \( X_k^u \) and \( X_k^Z \), in general, it is not feasible to both minimize the instantaneous utility losses and maximize the perceived future privacy. The problem, therefore, naturally manifests as a privacy-utility trade-off optimization problem. Intuitively, the optimization problem involves finding an optimal strategy that minimizes the sum of instantaneous utility losses, \( \sum_{n=1}^{n} d(\hat{X}_{n|n}^u, \hat{X}_{n|n}^Z) \), under the constraint that the privacy leakage at the end of the finite horizon, \( \frac{1}{d(\hat{X}_{n|n}^u, \hat{X}_{n|n}^Z)} \), must not exceed a pre-specified threshold \( \frac{1}{\delta} \). Formulating the optimization problem, however, exposes several challenges. For one, \( \sum_{n=1}^{n} d(\hat{X}_{n|n}^u, \hat{X}_{n|n}^Z) \) and \( d(\hat{X}_{n|n}^Z, \hat{X}_{n|n}^{-\delta}) \) are both random variables due to the uncertainties in the future observations, \( Z_{k+1}, Z_{k+2}, \ldots, Z_n \). Further, without the knowledge of the future observations, it is difficult to devise an optimal strategy that satisfies the constraint on future privacy leakage. In fact, given the Gaussian assumption for both the process noise and the measurement noise, it may not even be possible to ensure a non-trivial constraint on the future privacy leakage using any feasible sequence of actions, \( C_1, C_2, \ldots, C_n \), it is, therefore, more appropriate to characterize strategies in terms of the probability of privacy outage, \( P(d(\hat{X}_{n|n}^u, \hat{X}_{n|n}^Z) < \delta) \), in addition to the total utility loss. The probability of privacy outage reflects the probability that the desired privacy constraint \( d(\hat{X}_{n|n}^u, \hat{X}_{n|n}^Z) < \delta \) is not satisfied in the future, at time \( n \).

**IV. FORMULATION AS A MARKOV DECISION PROCESS PROBLEM**

The finite horizon privacy-utility trade-off problem fits nicely with a Markov Decision Process (MDP). In a discrete time continuous state MDP model, at every time step \( k \), an agent observes the current state of some Markov process \( S_k \), takes an action \( a_k \) and receives a reward \( R_k \). The state of the Markov process at time step \( k \), in general, depends on the state and the action at time step \( k-1 \) and some stochastic process noise \( o_k \). The reward received by the agent at time step \( k \) depends on the current state of the Markov process, the current action taken by the agent and the next state of the Markov process.

Formally, a discrete time continuous state continuous action Markov Decision Process is a tuple \((S, A, P, R, \gamma)\) where \( S \in \)
\( \mathbb{R}^J \) represents the state space, \( A \in \mathbb{R}^k \) represents the action space, \( P : \mathbb{R} \times A \times \mathbb{R} \rightarrow [0, 1] \) represents the state-transition function such that \( P(s_{k+1}|a_k, s_k) \) gives the probability of transitioning to the next state \( s_{k+1} \) from the current state \( s_k \) by taking an action \( a_k \). Let the random vectors \( \tilde{S}_k, A_k \) and \( S_{k+1} \) represent the current state, the current action and the next state, respectively. As the state space is continuous, \( P \) is specified as a probability density function such that \( \int_{\mathbb{R}} P(s_{k+1}|a_k, s_k) ds_{k+1} = P(s_k + 1 \in \Psi|s_k = a_k, A_k = a_k) \), where \( \Psi \subseteq \mathbb{R} \) with \( \mathbb{R} \) the space of \( S_k \). Similarly, \( R : \mathbb{R} \times A \times \mathbb{R} \rightarrow \mathbb{R} \) represents the reward function such that \( R_k(s_k, a_k, s_{k+1}) \) gives the reward received by the agent at time step \( k \) by taking an action \( a_k \) when the current and the next states of the Markov process are \( s_k \) and \( s_{k+1} \), respectively. The discount factor \( \gamma \in [0, 1] \) captures how the agent values her future rewards compared to her current reward – if \( \gamma = 1 \), the agent values all her future rewards equally to her current reward and if \( \gamma = 0 \), the agent only values her current reward and disregards all her future rewards.

The goal of the agent is to maximize the expected sum of her current and future discounted rewards, \( \mathbb{E} \sum_k \gamma^k R_k(S_k, A_k, S_{k+1}) \). The agent seeks to find the optimal sequence of actions that allows her to optimize the expected sum of discounted rewards. In this regard, it is useful to define a function, called the optimal state-value function, that provides a measure of the maximum achievable sum of expected rewards from a particular state. Let \( V^*_k(s_k) \) denote the optimal state-value function at time step \( k \) given the current state \( s_k \). Then, the optimal state-value function can be written as a recursive equation using Bellman’s Principle of Optimality as:

\[
V^*_k(s_k) = \max_{a_k} \mathbb{E} \sum_{s_{k+1}} P(s_{k+1}|a_k, s_k) \left( R_k(s_k, a_k, s_{k+1}) + \gamma V^*_{k+1}(s_{k+1}) \right) ds_{k+1}.
\]

The Bellman equation formulation offers a dynamic programming approach to solve the resulting optimization problem.

The finite-horizon privacy-utility problem can be directly translated to a finite-horizon discrete time continuous state continuous action MDP problem. Recall that in the finite-horizon privacy setting, the user seeks to find an optimal sequence of actions that allows her to maximize the sum of the instantaneous utilities while ensuring that the privacy leakage at the end of the finite horizon remains below a pre-specified threshold with high probability. This is inherently a decision problem that incorporates a meaningful notion of reward captured as the expected sum of instantaneous utilities and future privacy leakage.

Let \( \hat{X}_{k|k}^Z \) represent the data owner’s estimate of \( X \), given the series of observations, \( Z_1, Z_2, \ldots, Z_k \) and \( P^Z_{k|k} \) represent the error covariance associated with the estimate.\(^1\) Similarly, let \( \hat{X}_{k|k}^Z \) represent the data analyst’s (or adversary’s) estimate of \( X \), given the series of observations, \( \tilde{Z}_1, \tilde{Z}_2, \ldots, \tilde{Z}_k \) and \( P^\tilde{Z}_{k|k} \) represent the error covariance associated with the estimate.\(^1\) Since \( P^Z_{k|k} \) is not a function of \( Z_k \). The superscript \( Z \) is used as a convention to imply that the symbol being defined directly concerns the data owner, who has observations \( \{Z_k\} \), rather than the adversary or data analyst, who have observations \( \{\tilde{Z}_k\} \).

Now, define

\[
d^p(k, k) \triangleq d(\hat{X}_{k|k}^Z, \hat{X}_{k|k}^\tilde{Z}),
\]

\[
d^p(j, j) \triangleq d(\hat{X}_{j|k}^Z, \hat{X}_{j|k}^\tilde{Z}) \quad (n \geq j \geq k),
\]

\[
S_k = \{Z_k, \hat{X}_{k|k}^Z, \tilde{X}_{k|k}^\tilde{Z}, p^Z_{k|k}, p^\tilde{Z}_{k|k}, P^Z_{k|k}, P^\tilde{Z}_{k|k}\},
\]

where \( S_k \) represents the state of the MDP (which is fundamentally different from the state of the LDS, \( X_k \)). The state variables in \( S_k \) can be recursively computed using the following sequence of Kalman filter equations:

\[
\hat{X}_{k|k-1}^Z = F_k \hat{X}_{k-1|k-1}^Z,
\]

\[
\hat{X}_{k|k-1}^\tilde{Z} = F_k \hat{X}_{k-1|k-1}^\tilde{Z},
\]

\[
p^Z_{k|k-1} = F_k p^Z_{k-1|k-1}|F_k^T + Q_k,
\]

\[
p^\tilde{Z}_{k|k-1} = F_k p^\tilde{Z}_{k-1|k-1}|F_k^T + Q_k,
\]

\[
K_k = P^Z_{k|k-1} H_k^T (H_k P^Z_{k|k-1} H_k^T + R_k)^{-1},
\]

\[
K_k = P^\tilde{Z}_{k|k-1} H_k^T (H_k P^\tilde{Z}_{k|k-1} H_k^T + C_k^T R_k C_k)^{-1},
\]

\[
\hat{X}_{k|k}^Z = \hat{X}_{k|k-1}^Z + K_k (Z_k - H_k \hat{X}_{k|k-1}^\tilde{Z}),
\]

\[
\hat{X}_{k|k}^\tilde{Z} = \hat{X}_{k|k-1}^\tilde{Z} + K_k (Z_k - C_k H_k \hat{X}_{k|k-1}^Z),
\]

\[
P^Z_{k|k} = P^Z_{k|k-1} - K_k H_k P^Z_{k|k-1},
\]

\[
P^\tilde{Z}_{k|k} = P^\tilde{Z}_{k|k-1} - K_k C_k H_k P^\tilde{Z}_{k|k-1}.
\]

Initially, \( \hat{X}_{0|0}^Z = \hat{X}_{0|0}^\tilde{Z} = \mathbb{E}[X_0] \) and \( P^Z_{0|0} = P^\tilde{Z}_{0|0} = \text{Cov}(X_0) \). We now define the reward function, \( R_k \), as:

\[
R_k(S_k, C_k, S_{k+1}) = \begin{cases} 
\alpha(d^p(n, k + 1) - d^p(n, k)) - d^u(k, k) & \text{when } k < n, \\
\alpha d^p(n, k) - d^u(k, k) & \text{when } k = n,
\end{cases}
\]

where \( \alpha \) is the privacy-utility tradeoff parameter.

At any given time \( k \), the user’s goal is to choose an action \( C_k = f(S_k) \) that allows her to maximize the expected sum of the current and future rewards, \( \mathbb{E} \sum_{n=k}^{\infty} R(S_k, C_k, S_{k+1}) \).

Let \( C^* = \{C_1^*, C_2^*, \ldots, C_n^*\} \) be the set of optimal actions. Notice that at the beginning of the finite time horizon, the sum of the rewards can be expressed as

\[
\sum_{k=0}^{n} R_k(S_k, C_k, S_{k+1}) = \alpha(2d^p(n, n) - d^p(n, 0)) - \sum_{k=0}^{n} d^u(k, k).
\]

Since \( d^p(n, 0) \) and \( d^u(0, 0) \) are both zero (which follows from the assumption that \( \hat{X}_{0|0}^Z = \hat{X}_{0|0}^\tilde{Z} \) and \( P^Z_{0|0} = P^\tilde{Z}_{0|0} \)), substituting \( 2\alpha = \beta \), we get

\[
\sum_{k=0}^{n} R_k(S_k, C_k, S_{k+1}) = \beta d^p(n, n) - \sum_{k=1}^{n} d^u(k, k). \quad (4)
\]

As the reader may have noticed by now, the reward function is defined such that the sum of rewards captures both the privacy and the utility aspects of the problem in a single
expression given in (4). The value of $\gamma$ is taken to be 1 for the same reason. Note that the parameter $\beta$ in (4) directly relates to the probability of privacy outage $P(d(\hat{X}_{n|n}, \tilde{Z}_{n|p}) < \delta)$ and the resulting privacy-utility tradeoff: larger values of $\beta$ are expected to result in higher utility losses with lower probabilities of privacy outage while smaller values of $\beta$ are expected to result in lower utility losses with higher probabilities of privacy outage. The privacy-utility tradeoff region corresponding to different values of $\beta$ will be determined experimentally.

We now use the Bellman equation to formulate the finite horizon privacy-utility trade-off optimization problem. Let $V_k$ denote the state-value function at timestep $k$ and $V^*_k(s_k)$ denote the optimal state-value function given the state $s_k = \{\tilde{z}_k, \tilde{X}_{k-1|k-1}, \tilde{X}_{k|k-1}^z, P^z_{k-1|k-1}, P^z_{k-1|k-1}\}$. Then, using the Bellman equation of optimality, the optimization problem can be formulated as:

$$ V^*_k(s_k) = \max_{C_k} \int_\mathcal{Z} P(s_{k+1}|s_k, C_k) \left( R_k(s_k, C_k, s_{k+1}) + V^*_k(s_{k+1}) \right) ds_{k+1} $$

$$ = \max_{C_k} \int_\mathcal{Z} P(s_{k+1}|s_k, C_k) \left( \sum_{z_k} \mathbf{P}(\tilde{X}_{k|k}|s_k, C_k) \int_A \mathbf{P}(\tilde{X}_{k|k}^z|s_k, C_k) \right) ds_{k+1} $$

$$ \cdot \left( R_k(s_k, C_k, s_{k+1}) + V^*_k(s_{k+1}) \right) $$

$$ \cdot d\tilde{z}_{k+1} $$

where $\mathcal{Z}$, $A$, $\Delta$, $\Phi$ and $\Omega$ are the feasible spaces of $z_{k+1}, \tilde{z}_{k|k}, \tilde{X}_{k|k}, P^z_{k|k}$ and $P^z_{k|k}$, respectively.

Given $s_k$ and $C_k$, the state variables, $\tilde{z}_{k|k}, \tilde{X}_{k|k}, P^z_{k|k}$ and $P^z_{k|k}$ are all deterministic (which directly follows from the application of the Kalman filter equations). Therefore,

$$ V^*_k(s_k) = \max_{C_k} \int_\mathcal{Z} P(s_{k+1}|s_k, C_k) \left( R_k(s_k, C_k, s_{k+1}) + V^*_k(s_{k+1}) \right) ds_{k+1} $$

$$ = \max_{C_k} \int_\mathcal{Z} P(s_{k+1}|s_k, C_k) \left( R_k(s_k, C_k, s_{k+1}) + V^*_k(s_{k+1}) \right) ds_{k+1} $$

If $X_k$ is a Gaussian process and $V_k$ is a Gaussian white noise process, then, $Z_{k+1}|Z_k \sim N(\tilde{\mu}, \tilde{\Sigma})$ where

$$ \tilde{\mu} = \mathbb{E}[Z_{k+1}|Z_k] $$

$$ = \mathbb{E}[H X_{k+1} + V_{k+1}|Z_k] $$

$$ = \mathbb{E}[H X_{k+1}|Z_k] + \mathbb{E}[V_{k+1}|Z_k] $$

$$ = H \mathbb{E}[X_{k+1}|Z_k] $$

$$ = H \tilde{X}_{k+1} $$

$$ = H \tilde{X}_{k+1}^z $$

and

$$ \tilde{\Sigma} = \text{Cov}(Z_{k+1}|Z_k) $$

$$ = \text{Cov}(H X_{k+1} + V_{k+1}|Z_k) $$

$$ = \text{Cov}(H X_{k+1}|Z_k) + \text{Cov}(V_{k+1}|Z_k) $$

$$ = H \text{Cov}(X_{k+1}|Z_k) H^T + R $$

$$ = H P^z_{k+1|k} H^T + R $$

$$ = H (P^z_{k+1|k} F^T + Q) H^T + R. $$

The optimization problem in (6) reflects the user’s objective of maximizing the expected sum of the current and future rewards starting at a particular state $s_k$. The optimizing argument $C^*_k(s_k) = \arg \max_{C_k} V^*_k(s_k)$ constitutes the best action taken toward the goal of maximizing the expected sum of rewards.

V. SUB-OPTIMAL ALGORITHMS

The optimization problem formulated in (6) suffers from the curse of dimensionality as both the state space and the action space are continuous. Analytical methods to solve the problem are infeasible for practical problems as they do not yield closed-form solutions for higher dimensional problems. Numerical algorithms, such as the value iteration algorithm and the policy iteration algorithm which advance by sweeping through all possible states at each time step, also fail as there are infinitely many states to sweep through. The problem is therefore not easily tractable without further assumptions about the state space or the action space, or both.

A popular approach to solving similar optimization problems involves discretizing the state space (see, for instance, [45], [46], [47], [48], [49]). This approach is typically suboptimal, however, it can still yield promising solutions. In what follows, we highlight different algorithms that are based on the discretization of the state space to solve the optimization problem formulated in (6).

A. Value Iteration With Discretization

The value iteration approach to solving a finite-horizon discrete state space MDP problem involves solving the Bellman equation to find the optimal values for every possible state at every time step, starting at the end of the finite time horizon and working backwards. At time step $n$, the optimal value of each state is computed using the terminal reward given by $R_n(S_n, C_n) = \beta d^P(n, n) - d^P(n, n)$. The algorithm then iteratively calculates the optimal values at previous time steps as given in Algorithm 1.

The major characteristic of the value iteration algorithm is that it calculates optimal values and the optimal actions associated with the states at all time steps before the actual observations are available. The calculated values are optimal for the discrete MDP; however, due to the additional discretization step (which is not intrinsic to the value iteration algorithm itself), they are typically sub-optimal for the original MDP.

B. Pessimistic Algorithm

The pessimistic algorithm is a customized algorithm to solve the finite-horizon discrete state space MDP problem. The pessimistic algorithm captures an agent who always expects to transition to the worst possible state at every time step. The pessimistic algorithm seeks to optimize the value and
SHARMA et al.: ECONOMICS OF PRIVACY AND UTILITY: INVESTMENT STRATEGIES 1751

Algorithm 1 Value Iteration Algorithm With Discretization
1: Define the feasible state space, S.
2: Select a discretization rule, D, and discretize S according to D.
3: procedure VALUE ITERATION
4: Initialize $V^*_n(s_n)$ for all $s_n \in S$ with terminal rewards.
5: for $k = n - 1$ to 1 do
6: for each $s_k \in S$ do
7: $V^*_k(s_k) = \max_{C_k} \sum_{z_{k+1}} P(z_{k+1}|z_k) \left( R_k(s_k, C_k, s_{k+1}) + V^*_k(s_{k+1}) \right)$
8: $C^*_k(s_k) = \arg \max_{C_k} V^*_k(s_k)$
9: end for
10: end for
11: end procedure

Algorithm 2 Pessimistic Algorithm
1: Define the feasible state space, S.
2: Select a discretization rule, D, and discretize S according to D.
3: Initialize $V^*_n(s_n)$ for all $s_n \in S$ with terminal rewards.
4: $v^*_n = \min\{V^*_n(s_n) : s_n \in S\}$
5: $s^*_n = \arg\min_{s_n} \{V^*_n(s_n) : s_n \in S\}$
6: for $k = n - 1$ to 1 do
7: for each $s_k \in S$ do
8: $V^*_k(s_k) = \max_{C_k} \left( R_k(s_k, C_k, s_{k+1}) + v^*_k \right)$
9: $C^*_k(s_k) = \arg\max_{C_k} V^*_k(s_k)$
10: end for
11: $v^*_k = \min\{V^*_k(s_k) : s_k \in S\}$
12: $s^*_k = \arg\min_{s_k} \{V^*_k(s_k) : s_k \in S\}$
13: end for

the action associated with a state while assuming that the next transition leads to the state with the least value. The advantage of using the algorithm is that it is computationally less intensive as the state transition in the underlying model is assumed to be deterministic. The pessimistic algorithm is highlighted in Algorithm 2.

C. Optimistic Algorithm
In contrast to the pessimistic algorithm, the optimistic algorithm captures an agent who always expects to transition to the best possible state at every time step. The optimistic algorithm seeks to optimize the value and the action associated with a state while assuming that the next transition leads to the state with the highest value. The optimistic algorithm is highlighted in Algorithm 3.

VI. PRIVACY-UTILITY TRADEOFF UNDER ESTIMATED PRIVACY LEAKAGE
The finite-horizon privacy-utility tradeoff problem can also be formulated with the constraint on estimated privacy leakage instead of the actual privacy leakage at the end of the finite time horizon. The resulting optimization problem is much simpler to solve, nevertheless, the dynamic privacy requirements are still captured into the problem formulation. To this end, we consider a user who seeks to maximize her instantaneous utility while limiting the estimated future leakage about her sensitive information. The resulting optimization problem is:

$$\min_{C_k} d(\hat{X}_{k|k}^u, \hat{Z}_{k|k}^u)$$
subject to: $$d(\hat{X}_{n|k}^p, \hat{Z}_{n|k}^p) \geq \delta$$

Intuitively, the user seeks to disclose as much as possible (allowed by the privacy constraint) at the current time step so as to maximize her current utility with a complete disregard for her future utilities. This strategy is, therefore, referred to as a maximum disclosure strategy. In contrast, the optimization problem formulated in (6) captures a user who seeks to cautiously disclose her personal information piecewise.

In a dynamic setting, a user following the maximum disclosure strategy needs to solve the optimization problem at every time step as new observations are made. As the user approaches the end of the finite time horizon, the privacy constraint is more restrictive due to the accumulated leakage resulting from the past disclosures. In some cases, the actual leakage may already exceed the estimated leakage and therefore, no choice of $C_k$ may satisfy the constraint, especially closer to the end of the finite time horizon. Therefore, it is more appropriate to formulate an unconstrained optimization problem that captures the semantics of the constrained problem. This leads us to the following optimization problem:

$$\min_{C_k} d(\hat{X}_{k|k}^u, \hat{Z}_{k|k}^u) - \beta \left( d(\hat{X}_{n|k}^p, \hat{Z}_{n|k}^p) \right)$$
$$= \max_{C_k} \beta \left( d(\hat{X}_{n|k}^p, \hat{Z}_{n|k}^p) \right) - d(\hat{X}_{k|k}^u, \hat{Z}_{k|k}^u)$$

The parameter, $\beta$, in (9) directly relates to the constraint in (8) and influences the probability of privacy outage at the end of the finite time horizon, $P(d(\hat{X}_{n|k}^p, \hat{Z}_{n|k}^p) < \delta)$. Although the optimization problem formulated in (9) can easily be solved without further transformation, it is
nevertheless possible to transform the optimization problem into an equivalent MDP formulation. First, define

$$d^u(k, k) \triangleq d(\hat{X}_{k|k}^Z, \hat{X}_{k|k}^u),$$
$$d^p(n, k) \triangleq d(\hat{X}_{n|k}^Z, \hat{X}_{n|k}^p),$$
$$S_k \triangleq (Z_k),$$

where $S_k$ represents the state of the MDP. We now define the reward function, $R_k$, as

$$R_k(S_k, C_k) = \beta d^p(n, k) - d^u(k, k)$$

Notice that the reward function is independent of the future observations and therefore, deterministic. At any given time $k$, the user’s goal is to chose an action $C_k = f(S_k)$ that allows her to maximize the instantaneous reward, $R_k(S_k, C_k)$. Since the user is oblivious to future rewards, we set $\gamma = 0$. The MDP equivalent of the optimization problem in (9) can then be expressed as:

$$V_k^*(s_k) = \max_{C_k} R_k(s_k, C_k)$$

(10)

The optimization problem in (10) reflects the user’s objective of maximizing her instantaneous reward at a particular state $s_k$. The argument of the optimization $C_k^*(s_k) = \arg \max_{C_k} V_k^*(s_k)$ constitutes the best action taken toward the goal of maximizing the instantaneous reward.

The main advantage of the optimization problem formulated in (10) (and equivalently, in 9) is that it does not require sweeping through all possible states at each time step (which would otherwise be required if the current reward depended on future states) and therefore, computationally much less intensive to solve. Further, the optimization problem is solved forwards as new observations become available. Algorithm 4 highlights the steps involved in solving the finite-horizon privacy-utility tradeoff optimization problem under estimated privacy leakage using the maximum disclosure strategy.

**Algorithm 4** Maximum Disclosure Algorithm

1: At each time step $k$, do
2: $V_k^*(s_k) = \max_{C_k} R_k(s_k, C_k)$
3: $C_k^*(s_k) = \arg \max_{C_k} V_k^*(s_k)$
4: end

VII. SIMULATIONS

In this section, we evaluate the performance of the value iteration algorithm, the pessimistic algorithm, the optimistic algorithm and the maximum disclosure algorithm via synthetic simulations. For our simulations, we consider an LDS with $N_p = 1$ and $N_u = 2$. We assume that $F_k$ and $H_k$ are time invariant such that $F_1 = F_2 = \cdots = F_n = F$ and $H_1 = H_2 = \cdots = H_n = H$. Further, we assume that $X_k$ is a zero mean Gaussian process and $W_k$ and $V_k$ are independent and identically distributed standard Gaussian random vectors.

In practical applications, the matrices $F$ and $H$ are determined by learning the underlying system model. Learning the parameters of a Linear Dynamical system, commonly referred to as system identification, is a well-investigated problem [50], [51], [52], [53], [54], [55], [56]. For the purpose of simulations, we sample the elements of $F$ and $H$ independently from a uniform distribution in the unit interval. $F$ is further normalized such that its eigenvalues lie within a unit circle which ensures that the LDS is stable.

As $P^Z_{k | k - 1}$ is not a function of the observations or the actions, it is computed offline. Similarly, $P^Z_{k - 1 | k - 1}$ is estimated with $P^Z_{k - 1 | k - 1}$. For the value iteration, pessimistic and optimistic algorithms, a discretization function, $\mathcal{D}$, is used to approximate the components of $Z_k$, $\hat{X}^Z_{k - 1 | k - 1}$, and $\hat{X}^Z_{k - 1 | k - 1}$ as binary variables. As a simplest discretization strategy, we chose the function $\mathcal{D}$ such that:

$$\mathcal{D}(y) = \begin{cases} E[y] - 0.1 & \text{when } y < E[y], \\ E[y] + 0.1 & \text{when } y \geq E[y]. \end{cases}$$

(11)

The choice of 0.1 as the distance to the quantization points from the mean is arbitrary.

The performances of the four algorithms are evaluated in terms of the probability of privacy outage and the average utility loss. First, each algorithm, with the exception of the maximum disclosure algorithm, is run in turn to determine the optimal actions associated with every discretized state of the Markov Decision Process. Next, 10,000 Monte-Carlo simulations of the LDS are carried out. In each simulation of the LDS, a sequence of observations, $z_1, z_2, \cdots, z_n$ are generated. When an observation $z_k$ is generated, the Kalman filter equations are used to compute the actual state, $s_k$. For the value iteration, the pessimistic and the optimistic algorithms, the Bellman equation (6) is solved to determine the optimal action, $C_k^*$, associated with the state. For the maximum disclosure algorithm, the non-recursive equation (10) is solved to determine the optimal action, $C_k^*$, associated with the state. This process is repeated until the end of the finite time horizon. At the end of the finite time horizon, any violation of the privacy constraint: $d(\hat{X}_{n|n}^Z, \hat{X}_{n|n}^p) < \delta$, is checked, which concludes one simulation. After all simulations have been completed, the probability of privacy outage and the average utility loss are calculated using

$$P(\text{outage}) = \frac{\text{number of constraint violations}}{\text{total number of simulations}}$$

and

$$\text{Average utility loss} = \frac{\sum_{k=1}^{n} d(\hat{X}_{k|k}^Z, \hat{X}_{k|k}^u)}{\text{total number of simulations}},$$

respectively. The experiment is repeated multiple times for different randomly generated samples of $H$ and $F$.

From among multiple system models used in the simulations, three representative system models are selected that provide various insights on the performances of the four algorithms:
Fig. 2 shows the performances of the four algorithms in terms of the probability of privacy outage and the average utility loss across different values of $\beta$ for system model 1. For reference, the performance of a naive strategy in which the user randomly selects $C_k$ from a uniform distribution in the unit interval at each time step $k$ is also included. In Figure 2a and Figure 2b, we observe that all four algorithms consistently outperform the random action strategy across all values of $\beta$. Also, for all four algorithms, we observe a decrease in the probability of privacy outage, and an increase in the average utility loss, as $\beta$ increases. This observation is consistent with the intuition that larger values of $\beta$ put more weight on the privacy requirement than the utility requirement and therefore, result in a decrease in the probability of privacy outage and an increase in the utility loss. For the random action strategy, however, we observe that the probability of privacy outage (Figure 2a) and the average utility loss (Figure 2b) are both virtually constant across all values of $\beta$. This is expected as the random action strategy does not account for $\beta$ in the selection of $C_k$.

In Figure 2a and Figure 2b, we also observe that the performances of the value iteration, the pessimistic and the optimistic algorithms are similar across different values of $\beta$. We consistently observed similar performances of the three algorithms for different random samples of $H$ and $F$ and for different values of $n$. In light of this, we conclude that the average performances of the three algorithms in terms of the probability of outage and the average utility loss are similar. Consequently, it may be desirable to use the pessimistic or the optimistic algorithm over the value iteration algorithm for speed benefits. However, it should be noted that the algorithms may perform differently for a more robust discretization strategy.
The quantization points in pursuit of a better discretization than the maximum disclosure strategy. However, increasing strategy, we expect the three algorithms to perform better than the binary discretization strategy. For a more robust discretization, system models can be attributed to the choice of the value iteration, pessimistic, and optimistic algorithms. Similarly, the dynamic range for the privacy-utility tradeoff achieved by the maximum disclosure strategy is significantly higher than the other strategies—the higher dynamic range translates to more room for tuning the privacy-utility tradeoff.

Figure 3, Figure 4 and Figure 5 highlight the privacy-utility tradeoff achieved by the maximum disclosure strategy/algorithm against the value iteration, pessimistic and optimistic algorithms for the three system models. For system model 1 (Figure 3), we observe that the maximum disclosure strategy significantly outperforms the other algorithms and achieves much better privacy-utility tradeoff. For system model 2 (Figure 4), we observe that the performance of the maximum disclosure strategy is similar to that of the value iteration, pessimistic and optimistic algorithms. Similarly, for system model 3 (Figure 5), the dynamic range for the privacy-utility tradeoff achieved by the maximum disclosure strategy is significantly higher than the other strategies—the higher dynamic range translates to more room for tuning the privacy-utility tradeoff.

The performance of the four algorithms, in general, depends on the system model. The relatively poor performances of the value iteration, the pessimistic and the optimistic algorithms against the maximum disclosure strategy across all representative system models can be attributed to the choice of the binary discretization strategy. For a more robust discretization strategy, we expect the three algorithms to perform better than the maximum disclosure strategy. However, increasing the quantization points in pursuit of a better discretization strategy significantly increases the computational requirements and may not be feasible for all systems. For high dimensional practical problems, maximum disclosure strategy is therefore the only computationally feasible option to solve the dynamic privacy problem.

VIII. CONCLUSION AND FUTURE WORK

The increasing privacy concerns associated with disclosing personal data have guided the state-of-the-art privacy mechanisms that afford theoretical privacy guarantees. However, existing works mostly consider the problem in a static setting which either assumes that a user discloses her information only once, or treats each disclosure of the user’s information independently to the previous disclosures. In this paper, we considered a dynamic model of privacy in which a privacy-aware user cautiously discloses her personal information to a service provider over a finite time horizon. We investigated different strategies that allow a user to maximize their utility over time while limiting the privacy leakage at the end of the finite horizon. Using experimental evaluations on synthetic datasets, we showed that these strategies, although sub-optimal, can yield promising tradeoff region for the finite-horizon privacy-utility tradeoff problem. Finally, we demonstrated that there exists a simpler strategy corresponding to a simplified version of the problem, that has significant computational benefits with encouraging performance.

Future work will focus on the practical implementation of the privacy model in real-world applications. In line with expanding its practical use, we are gearing up to integrate the privacy model into a fitness app, providing tangible benefits to users in the realm of health and wellness. Furthermore, we are exploring the synergistic integration of the compression-based privacy mechanism with a distortion-based mechanism, anticipating that this combined approach will enhance the privacy-utility tradeoff, fostering an even more robust protection of user privacy. These strategic initiatives perfectly align with the ever-evolving landscape of privacy solutions in real-world, dynamic settings.

REFERENCES

[1] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, “Our data, ourselves: Privacy via distributed noise generation,” in Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn. Cham, Switzerland: Springer, 2006, pp. 486–503.
[2] 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.
[3] X. He, X. Zhang, and C.-C. J. Kuo, “A distortion-based approach to privacy-preserving metering in smart grids,” IEEE Access, vol. 1, pp. 67–78, 2013.
[4] Q. Geng and P. Viswanath, “The optimal noise-adding mechanism in differential privacy,” IEEE Trans. Inf. Theory, vol. 62, no. 2, pp. 925–951, Feb. 2016.
[5] W. Wang, L. Ying, and J. Zhang, “On the relation between identifiability, differential privacy, and mutual-information privacy,” IEEE Trans. Inf. Theory, vol. 62, no. 9, pp. 5018–5029, Sep. 2016.
[6] K. Kalantari, L. Sankar, and O. Kosut, “On information-theoretic privacy with general distortion cost functions,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2017, pp. 2865–2869.
[7] J. Liao, O. Kosut, L. Sankar, and F. P. Calmon, “Privacy under hard distortion constraints,” in Proc. IEEE Inf. Theory Workshop (ITW), Nov. 2018, pp. 1–5.
S. Y. Kung, "Compressive privacy: From information/estimation theory to machine learning [lecture notes]," IEEE Signal Process. Mag., vol. 34, no. 1, pp. 94–112, Jan. 2017.

S.-Y. Kung, T. Chanyaswad, J. M. Chang, and P. Wu, “Collaborative PCA/DCA learning methods for compressive privacy,” ACM Trans. Embedded Comput. Syst., vol. 16, no. 3, p. 1–18, Aug. 2017.

S. Y. Kung, “A compressive privacy approach to generalized information bottleneck and privacy funnel problems,” J. Franklin Inst., vol. 355, no. 4, pp. 1846–1872, Mar. 2018.

Y. Song, C. X. Wang, and W. P. Tay, “Compressive privacy for a linear dynamical system,” IEEE Trans. Inf. Forensics Security, vol. 15, pp. 895–910, 2020.

M. S. Alvim, M. E. Andres, K. Chatzikokolakis, P. Degano, and C. Palamidessi, “Differential privacy: On the trade-off between utility and information dissemination,” in Proc. Int. Workshop Formal Aspects Secur. Trust, Cham, Switzerland: Springer, 2011, pp. 39–54.

C. Yin, J. Xi, R. Sun, and J. Wang, “Location privacy protection based on differential privacy strategy for big data in Internet of Things,” IEEE Trans. Inf. Informat., vol. 14, no. 8, pp. 3628–3636, Aug. 2018.

Y. Yang, Z. Zhang, G. Miklau, M. Winslett, and X. Xiao, “Differential privacy in data publication and analysis,” in Proc. ACM SIGMOD Int. Conf. Management Data, May 2012, pp. 1118–1129, 2012.

T. Zhu, P. Xiong, G. Li, and W. Zhou, “Correlated differential privacy: Hiding information in non-IID data set,” IEEE Trans. Inf. Forensics Security, vol. 10, no. 2, pp. 229–242, Feb. 2015.

X. Wu, T. Wu, M. Khan, Q. Ni, and W. Dou, “Game theory based correlated privacy preserving analysis in big data,” IEEE Trans. Big Data, vol. 7, no. 4, pp. 643–656, Oct. 2021.

X. He, A. Machanavajjhala, and B. Ding, “Blowfish privacy: Tuning privacy to utility trade-offs using policies,” in Proc. ACM SIGMOD Int. Conf. Management Data, Jun. 2014, pp. 1447–1458.

A. Makhdoumi and N. Fawaz, “Privacy-utility tradeoff under statistical uncertainty,” in Proc. 51st Annu. Allerton Conf. Commun., Control, Comput. (Allerton), Oct. 2013, pp. 1627–1634.

S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor, “Smart meter privacy: A utility-privacy framework,” in Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm), Oct. 2011, pp. 190–195.

C. Sharma, B. Mandal, and G. Amarucci, “A practical approach to navigating the tradeoff between privacy and precise utility,” in Proc. IEEE Int. Conf. Communi., Jun. 2021, pp. 1–6.

C. Braun, K. Chatzikokolakis, and C. Palamidessi, “Quantitative notions of leakage for one-way attacks,” Electron. Notes Theor. Comput. Sci., vol. 249, pp. 75–89, Aug. 2009.

G. Smith, “On the foundations of quantitative information flow,” in Proc. Int. Conf. Found. Softw. Comput. Struct. Cham, Switzerland: Springer, 2009, pp. 288–302.

F. Farokhi and H. Sandberg, “Fisher information as a measure of privacy: Preserving privacy of households with smart meters using batteries,” IEEE Trans. Smart Grid, vol. 9, no. 5, pp. 4726–4734, Sep. 2018.

F. Farokhi and H. Sandberg, “Optimal privacy-preserving policy using constrained additive noise to minimize the Fisher information,” in Proc. IEEE 56th Annu. Conf. Decis. Control (CDC), Dec. 2017, pp. 2692–2697.

C. X. Wang, Y. Song, and W. P. Tay, “Preserving parameter privacy in sensor networks,” in Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP), Nov. 2018, pp. 1316–1320.

I. Issa, S. Kamath, and A. B. Wagner, “An operational measure of information leakage,” in Proc. Ann. Conf. Inf. Sci. Syst. (CISS), Mar. 2016, pp. 234–239.

J. Liao, L. Sankar, F. P. Calmon, and V. Y. F. Tan, “Hypothesis testing under maximal leakage privacy constraints,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), Jun. 2017, pp. 779–783.

C. T. Li and A. El Gamal, “Maximal correlation secrecy,” IEEE Trans. Inf. Theory, vol. 64, no. 5, pp. 3916–3926, May 2018.

S. Assehdeh, F. Alajaji, and T. Linder, “On maximal correlation, mutual information and data privacy,” in Proc. IEEE 14th Can. Workshop Inf. Theory (CWIT), Jul. 2015, pp. 27–31.