Privacy-Aware Location Sharing with Deep Reinforcement Learning

Ecenaz Erdemir, Pier Luigi Dragotti and Deniz Gündüz
Imperial College London
Department of Electrical and Electronic Engineering
Email: {e.erdemir17, p.dragotti, d.gunduz}@imperial.ac.uk

***Abstract*—Location-based services (LBSs) have become widely popular. Despite their utility, these services raise concerns for privacy since they require sharing location information with untrusted third parties. In this work, we study privacy-utility trade-off in location sharing mechanisms. Existing approaches are mainly focused on privacy of sharing a single location or myopic location trace privacy; neither of them taking into account the temporal correlations between the past and current locations. Although these methods preserve the privacy for the current time, they may leak significant amount of information at the trace level as the adversary can exploit temporal correlations in a trace. We propose an information theoretically optimal privacy preserving location release mechanism that takes temporal correlations into account. We measure the privacy leakage by the mutual information between the user's true and released location traces. To tackle the history-dependent mutual information minimization, we reformulate the problem as a Markov decision process (MDP), and solve it using asynchronous actor-critic deep reinforcement learning (RL).**

I. Introduction

Fast advances in mobile devices and positioning technologies have fostered the development of many location-based services (LBSs), such as Google Maps, Uber, Forsquare and Tripadvisor. These services provide users with useful information about their surroundings, transportation services, friends' activities, or nearby attraction points. Moreover, the integration of LBSs with social networks, such as Facebook, Twitter, YouTube, has rapidly increased indirect location sharing, e.g., via image or video sharing. However, location is one of the most sensitive private information for users, since a malicious adversary can use this information to derive users' habits, health condition, social relationships, or religion. Therefore, location trace privacy has been an important concern in LBSs, and there is an increasing pressure from consumers to keep their traces private against malicious attackers or untrusted service providers (SPs), while preserving the utility obtained from these applications.

A large body of research has focused on location-privacy protection mechanisms (LPPMs) against an untrusted service provider [1]. These methods can be categorized as spatial-location and temporal-location privacy preserving methods [2]. While the former focuses on protecting a single location data [3]–[7], the latter aims at providing location trace privacy [8]–[10]. Individual locations on a trace are highly correlated, and the strategies focusing on the current location privacy might reveal sensitive information about the past or future locations.

Differential privacy, k-anonymity and information theoretic metrics are commonly used as privacy measures [3]–[10]. By definition, differential privacy prevents the service provider from inferring the current location of the user, even if the SP has the knowledge of all the remaining locations. K-anonymity ensures that a location is indistinguishable from at least $k-1$ other location points. However, differential privacy and k-anonymity are meant to ensure the privacy of a single location, and they are shown not to be appropriate measures for location privacy in [11]. Instead, we treat the true and released location traces as random sequences, and measure the privacy leakage by mutual information [12].

In [7], the authors introduce location distortion mechanisms to keep the user's trajectory private. Privacy is measured by mutual information between the true and released traces and constrained by the average distortion for a specific distortion measure. The true trajectory is assumed to form a Markov chain. Due to the computational complexity of history-dependent mutual information optimization, authors propose bounds which take only the current and one step past locations into account. However, due to temporal correlations in the trajectory, the optimal distortion introduced at each time instance depends on the entire distortion and location history. Hence, the proposed bounds do not guarantee optimality.

In this work, we consider the scenario in which the user follows a trajectory generated by a first-order Markov process, and periodically reports a distorted version of her location to an untrusted service provider. We assume that the true locations become available to the user in an online manner. We use the mutual information between the true and distorted location traces as a measure of privacy loss. For the privacy-utility trade-off, we introduce an online LPPM minimizing the mutual information while keeping the distortion below a certain threshold. Unlike [7], we consider location release policies which take the entire released location history into account, and show its optimality. To tackle the complexity, we exploit the Markovity of the true user trajectory, and recast the problem as a Markov decision process (MDP). After identifying the structure of the optimal policy, we use advantage actor-critic (A2C) deep reinforcement learning (RL) framework as a tool to evaluate our continuous state and action space MDP numerically.

This work was partially supported by the European Research Council (ERC) through project BEACON (No. 677854).

II. Problem Statement

We consider a user who shares her location with a service provider to gain utility through some LBS. We denote the true location of the user at time t by $X_t \in \mathcal{W}$, where $\mathcal{W}$ is the finite set of possible locations. We assume that the user trajectory $\{X_t\}_{t\geq 1}$ follows a first-order time-homogeneous Markov chain with transition probabilities $q_x(x_{t+1}|x_t)$, and initial probability distribution p_{x_1}. At time t, the user shares a distorted version of her current location, denoted by $Y_t \in \mathcal{W}$, with the untrusted service provider due to privacy concerns. We assume that the user shares the distorted location in an online manner; that is, the released location at time t does not depend on the future true locations; i.e., for any $1 < t < n$, $Y_t \to (X^t, Y^{t-1}) \to (X_{t+1}^n, Y_{t+1}^n)$ form a Markov chain, where we have denoted the sequence $(X_{t+1}, \ldots, X_n)$ by X_{t+1}^n, and the sequence $(X_1, \ldots, X_t)$ by X^t.

Our goal is to characterize the trade-off between the privacy and utility. We quantify privacy by the information leaked to the untrusted service provider, measured by the mutual information between the true and released location trajectories. The information leakage of the user's location release strategy for a time period n is given by

$$I(X^n;Y^n) = \sum_{t=1}^{n} I(X^n;Y_t|Y^{t-1}) = \sum_{t=1}^{n} I(X^t;Y_t|Y^{t-1}), \quad (1)$$

where the first equality follows from the chain rule of mutual information, and the second from the Markov chain $Y^t \to (X_t, Y^{t-1}) \to X_{t+1}^n$.

Releasing distorted locations also reduces the utility received from the service provider. Therefore, the distortion applied by the user should be limited. The distortion between the true location X_t and the released location Y_t is measured by a specified distortion measure $d(X_t, Y_t)$ (e.g., Manhattan distance or Euclidean distance).

Our goal is to minimize the information leakage rate to the service provider while keeping the average distortion below a specified level for utility. The infinite-horizon optimization problem can be written as:

$$\min_{\{q_t(y_t|x^t,y^{t-1})\}_{t=1}^{\infty}} \lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} I^{\boldsymbol{q}}(X^t;Y_t|Y^{t-1}) \quad (2)$$

$$\text{such that } \lim_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\sum_{t=1}^{n} d(X_t,Y_t)\right] \leq \bar{D}, \quad (3)$$

where $\bar{D}$ is the specified average distortion constraint on the utility loss, x_t and y_t represent the realizations of X_t and Y_t, $q_t(y_t|x^t, y^{t-1})$ is a conditional probability distribution which represents the user's randomized *location release policy* at time t. The expectation in (3) is taken over the joint probabilities of X_t and Y_t, where the randomness stems from both the Markov process generating the true trajectory, and the random release mechanism $q_t(y_t|x^t, y^{t-1})$. The mutual information induced by policy $q_t(y_t|x^t, y^{t-1})$ is calculated using the joint probability distribution

$$P^{\boldsymbol{q}}(X^n = x^n, Y^n = y^n) = p_{x_1} q_1(y_1|x_1) \times \prod_{t=2}^{n} \left[q_x(x_t|x_{t-1}) q_t(y_t|x^t, y^{t-1})\right], \quad (4)$$

where $\boldsymbol{q} = \{q_t(y_t|x^t, y^{t-1})\}_{t=1}^n$. In the next section, we characterize the structure of the optimal location release policy, and using this structure recast the problem as an MDP, and finally evaluate the optimal trade-off numerically using deep RL.

III. Privacy-utility trade-off for online location sharing

In this section, we analyze the optimal privacy-utility trade-off achievable by an LPPM under the notion of mutual information minimization with a distortion constraint. Moreover, we propose simplified location release policies that still preserve the optimality.

By the definition of mutual information, the objective in (2) depends on the entire history of X and Y. Therefore, the user must follow a history-dependent location release policy $q_t^h(y_t|x^t, y^{t-1})$, where a feasible set $\mathcal{Q}_H$ satisfies $\sum_{y_t\in\mathcal{W}} q_t^h(y_t|x^t, y^{t-1}) = 1$. As a result of strong history dependence, computational complexity of the minimization problem increases exponentially with the increasing length of user trajectory. To tackle this problem, we introduce a class of simplified policies.

A. Simplified Location Release Policies

In this section we introduce a set of policies $\mathcal{Q}_S \subseteq \mathcal{Q}_H$ of the form $q_t^s(y_t|x_t, x_{t-1}, y^{t-1})$, which samples the distorted location only by considering the last two true locations and the entire released location history. Hence, the joint distribution (4) induced by $\boldsymbol{q}_s \in \mathcal{Q}_S$, where $\boldsymbol{q}_s = \{q_t^s(y_t|x_t, x_{t-1}, y^{t-1})\}_{t=1}^n$ can be written as

$$P^{\boldsymbol{q}_s}(X^n = x^n, Y^n = y^n) = p_{x_1} q_1^s(y_1|x_1) \times \prod_{t=2}^{n} \left[q_x(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1})\right]. \quad (5)$$

Next, we show that considering location release policies in set $\mathcal{Q}_S$ is without loss of optimality.

Theorem 1. *In the minimization problem (2), there is no loss of optimality in restricting the location release policies to the set of policies $\boldsymbol{q}_s \in \mathcal{Q}_S$. Furthermore, information leakage induced by any $\boldsymbol{q}_s \in \mathcal{Q}_S$ can be written as:*

$$I^{\boldsymbol{q}_s}(X^n, Y^n) = \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1}) \quad (6)$$

$$= \sum_{t=1}^{n} \sum_{\substack{y^t\in\mathcal{W}^t \\ (x_t,x_{t-1})\in\mathcal{W}}} P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^t) \log \frac{q_t^s(y_t|x_t, x_{t-1}, y^{t-1})}{P^{\boldsymbol{q}_s}(y_t|y^{t-1})}. \quad (7)$$

The proof of Theorem 1 relies on the following lemmas and will be presented later.

Lemma 1. *For any* $\boldsymbol{q} \in \mathcal{Q}_H$,

$$I^{\boldsymbol{q}}(X^n;Y^n) \geq \sum_{t=1}^{n} I^{\boldsymbol{q}}(X_t, X_{t-1}; Y_t|Y^{t-1}) \tag{8}$$

with equality if and only if $\boldsymbol{q} \in \mathcal{Q}_S$.

Proof: For any $\boldsymbol{q} \in \mathcal{Q}_H$,

$$I^{\boldsymbol{q}}(X^n;Y^n) = \sum_{t=1}^{n} I^{\boldsymbol{q}}(X^t; Y_t|Y^{t-1}) \tag{9}$$

$$\geq \sum_{t=1}^{n} I^{\boldsymbol{q}}(X_t, X_{t-1}; Y_t|Y^{t-1}), \tag{10}$$

where (9) follows from (1), and (10) from the non-negativity of mutual information. ∎

Lemma 2. *For any* $\boldsymbol{q}_h \in \mathcal{Q}_H$*, there exists a* $\boldsymbol{q}_s \in \mathcal{Q}_S$ *such that*

$$\sum_{t=1}^{n} I^{\boldsymbol{q}_h}(X_t, X_{t-1}; Y_t|Y^{t-1}) = \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1}). \tag{11}$$

Proof: For any $\boldsymbol{q}_h \in \mathcal{Q}_H$, we choose the policy $\boldsymbol{q}_s \in \mathcal{Q}_S$ such that

$$q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) = P^{\boldsymbol{q}_h}_{Y_t|X_t, X_{t-1}, Y^{t-1}}(y_t|x_t, x_{t-1}, y^{t-1}), \tag{12}$$

and we show that $P^{\boldsymbol{q}_h}_{X_t, X_{t-1}, Y^t} = P^{\boldsymbol{q}_s}_{X_t, X_{t-1}, Y^t}$. Then, $I^{\boldsymbol{q}_h}(X_t, X_{t-1}; Y_t|Y^{t-1}) = I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1})$ holds, which proves the statement in Lemma 2. The proof of $P^{\boldsymbol{q}_h}_{X_t, X_{t-1}, Y^t} = P^{\boldsymbol{q}_s}_{X_t, X_{t-1}, Y^t}$ is derived by induction as follows,

$$\begin{aligned}
&P^{\boldsymbol{q}_h}(x_{t+1}, x_t, y^t) \\
&= \sum_{x_{t-1} \in \mathcal{W}} q_x(x_{t+1}|x_t) q_t^h(y_t|x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_h}(x_t, x_{t-1}, y^{t-1}) \\
&= \sum_{x_{t-1} \in \mathcal{W}} q_x(x_{t+1}|x_t) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) P^{\boldsymbol{q}_s}(x_t, x_{t-1}, y^{t-1}) \\
&= P^{\boldsymbol{q}_s}(x_{t+1}, x_t, y^t),
\end{aligned} \tag{13}$$

where (12) holds, and $P^{\boldsymbol{q}_h}_{X_1}(x) = p_{x_1}(x) = P^{\boldsymbol{q}_s}_{X_1}(x)$ is for the initialization of the induction. ∎

Proof of Theorem 1: Following Lemmas 1 and 2, for any $\boldsymbol{q}_h \in \mathcal{Q}_H$, there exists a $\boldsymbol{q}_s \in \mathcal{Q}_S$ such that

$$I^{\boldsymbol{q}_h}(X^n;Y^n) \geq I^{\boldsymbol{q}_s}(X^n;Y^n). \tag{14}$$

Hence, there is no loss of optimality in using the location release policies of the form $q_t^s(y_t, |x_t, x_{t-1}, y^{t-1})$, and information leakage reduces to (7). ∎

Restricting our attention to the user location release policies $\boldsymbol{q}_s \in \mathcal{Q}_S$, we can write the minimization problem (2) as

$$\min_{\boldsymbol{q}_s : \{\mathbb{E}^{\boldsymbol{q}_s}[d(x_t,y_t)] \leq \bar{D}\}_{i=1}^{n}} \frac{1}{n} \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1}). \tag{15}$$

The location release strategy followed by the user is illustrated by the Markov chain in Fig. 1, where H_t denotes the released location history, i.e., $H_t = Y^t$. That is, the user samples a distorted location, Y_t, at time t by considering the current and previous true locations, (X_t, X_{t-1}), and released location history, (H_{t-2}, Y_{t-1}).

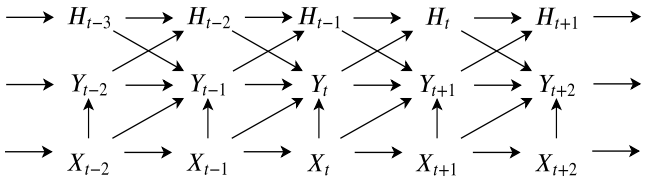

Fig. 1: Markov chain for the simplified location release policy.

Minimization of the mutual information subject to a utility constraint can be converted into an unconstrained minimization problem using Lagrange multipliers. Since the distortion constraint is memoryless, we can integrate it into the additive objective function easily. Hence, the unconstrained minimization problem for online location release privacy-utility trade-off can be rewritten as

$$\min_{\boldsymbol{q}_s \in \boldsymbol{Q}_s} \frac{1}{n} \sum_{t=1}^{n} I^{\boldsymbol{q}_s}(X_t, X_{t-1}; Y_t|Y^{t-1}) + \lambda(\mathbb{E}^{\boldsymbol{q}_s}[d(x_t, y_t)] - \bar{D}). \tag{16}$$

B. MDP Formulation

Markovity of the user's true location trace and the additive objective function in (16) allow us to represent the problem as an MDP with state X_t. However, the information leakage at time t depends on Y^{t-1}, resulting in a growing state space in time. Therefore, for a given policy $\boldsymbol{q}_s$ and any realization y^{t-1} of Y^{t-1}, we define a belief state $\beta_t \in \mathcal{P}_X$ as a probability distribution over the state space:

$$\beta_t(x_{t-1}) = P^{\boldsymbol{q}_s}(X_{t-1} = x_{t-1}|Y^{t-1} = y^{t-1}). \tag{17}$$

This represents the service provider's belief on the user's true location at the beginning of time instance t, i.e., after receiving the distorted location y_{t-1} at the end of the previous time instance $t-1$. The MDP actions are defined as the probability distributions sampling the released location $Y_t = y_t$ at time t, and determined by the randomized location release policies. The user's action induced by a policy $\boldsymbol{q}_s$ can be denoted by $a_t(y_t|x_t, x_{t-1}) = P^{\boldsymbol{q}_s}(Y_t = y_t|X_t = x_t, X_{t-1}, \beta_t)$ [13]–[15]. At each time t, the service provider updates its belief on the true location $\beta_{t+1}(x_t)$ after observing the distorted location y_t by

$$\begin{aligned}
\beta_{t+1}(x_t) &= \frac{p(x_t, y_t|y^{t-1})}{p(y_t|y^{t-1})} = \frac{\sum_{x_{t-1}} p(x_t, x_{t-1}, y_t|y^{t-1})}{\sum_{x_t, x_{t-1}} p(x_t, x_{t-1}, y_t|y^{t-1})} \\
&= \frac{\sum_{x_{t-1}} p(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) p(x_{t-1}|y^{t-1})}{\sum_{x_t, x_{t-1}} p(x_t|x_{t-1}) q_t^s(y_t|x_t, x_{t-1}, y^{t-1}) p(x_{t-1}|y^{t-1})} \\
&= \frac{\sum_{x_{t-1}} q_x(x_t|x_{t-1}) a(y_t|x_t, x_{t-1}) \beta_t(x_{t-1})}{\sum_{x_t, x_{t-1}} q_x(x_t|x_{t-1}) a(y_t|x_t, x_{t-1}) \beta_t(x_{t-1})}.
\end{aligned} \tag{18}$$

We define per-step information leakage of the user due to taking action $a_t(y_t|x_t, x_{t-1})$ at time t as,

$$l_t(x_t, x_{t-1}, a_t, y^t; \boldsymbol{q}_s) := \log \frac{a_t(y_t|x_t, x_{t-1})}{P^{\boldsymbol{q}_s}(y_t|y^{t-1})}. \quad (19)$$

The expectation of n step sum of (19) over the joint probability $P^{\boldsymbol{q}_s}(X_t, X_{t-1}, Y^t)$ is equal to the mutual information expression in the original problem (15). Therefore, given belief and action probabilities, average information leakage at time t can be formulated as,

$$\begin{aligned}\mathbb{E}^{\boldsymbol{q}_s}[l_t(x_{t-1}^t, a_t, y^t)] =& \sum_{x_t, x_{t-1}, y_t \in \mathcal{W}} \beta_t(x_{t-1}) a_t(y_t|x_t, x_{t-1}) q_x(x_t|x_{t-1}) \\ & \times \log \frac{a_t(y_t|x_t, x_{t-1})}{\sum_{\hat{x}_t, \hat{x}_{t-1} \in \mathcal{W}} \beta_t(\hat{x}_{t-1}) a_t(y_t|\hat{x}_t, \hat{x}_{t-1}) q_x(\hat{x}_t|\hat{x}_{t-1})} \\ :=& \mathcal{L}(\beta_t, a_t). \end{aligned} \quad (20)$$

We remark that the representation of average distortion in terms of belief and action probabilities is straightforward due to its additive form. Similarly to (20), average distortion at time t can be written as,

$$\begin{aligned}\mathbb{E}^{\boldsymbol{q}_s}[d(x_t, y_t)] =& \sum_{x_t, x_{t-1}, y_t \in \mathcal{W}} \beta_t(x_{t-1}) a_t(y_t|x_t, x_{t-1}) q_x(x_t|x_{t-1}) d(x_t, y_t) \\ :=& \mathcal{D}(\beta_t, a_t) \end{aligned} \quad (21)$$

Finally, we can recast the original problem in (16) as a continuous state and action space MDP. Evaluation of the MDP relies on minimizing the objective

$$\mathcal{C}(\beta_t, a_t) = \mathcal{L}(\beta_t, a_t) + \lambda(\mathcal{D}(\beta_t, a_t) - \bar{D}) \quad (22)$$

at each time step t for a trajectory of length n.

Finding optimal policies for continuous state and action space MDPs is a PSPACE-hard problem [16]. In practice, they can be solved by various finite-state MDP evaluation methods, e.g., value iteration, policy iteration and gradient-based methods. These are based on the discretization of the continuous belief states to obtain a finite state MDP [17]. While finer discretization of the belief reduces the loss from the optimal solution, it causes an increase in the state space; hence, in the complexity of the problem. Therefore, we use a deep learning based method as a tool to numerically solve our continuous state and action space MDP problem.

C. Advantage Actor-Critic (A2C) Deep RL

In RL, an agent discovers the best action to take in a particular state by receiving instantaneous rewards/costs from the environment [18]. On the other hand, in our problem, we have the knowledge of state transitions and the cost for every state-action pair without a need for interacting with the environment. We use A2C-deep RL as a computational tool to numerically evaluate the optimal location release policies for our continuous state and action space MDP.

To integrate RL framework into our problem, we create an artificial environment which inputs the user's current action, $a_t(y_t|x_t, x_{t-1})$, samples an observation y_t, and calculates the next state, β_{t+1}, using Bayesian belief update (18). Instantaneous cost revealed by the environment is calculated by (22).

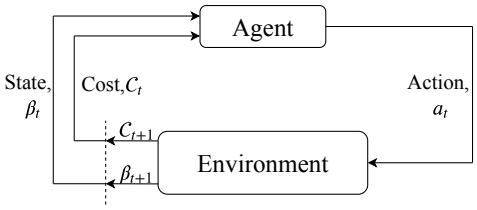

Fig. 2: RL for a known model.

The user receives the experience tuple $(\beta_t, a_t, y_t, \beta_{t+1}, \mathcal{C}_t)$ from the environment, and refines her policy accordingly. Fig. 2 illustrates the interaction between the artificial environment and the user, which is represented by the RL agent. The corresponding Bellman equation induced by the location release policy $\boldsymbol{q}_s$ can be written as

$$V^{\boldsymbol{q}_s}(\beta) + J(\boldsymbol{q}_s) = \min_a \Big\{ \mathcal{C}(\beta, a) + V^{\boldsymbol{q}_s}(\beta') \Big\}, \quad (23)$$

where $V^{\boldsymbol{q}_s}(\beta)$ is the state-value function, β' is the updated belief state according to (18), a represents action probability distributions, and $J(\boldsymbol{q}_s)$ is the cost-to-go function, i.e., the expected future cost induced by policy $\boldsymbol{q}_s$ [19].

RL methods can be divided into three groups: value-based, policy-based, and actor-critic [20]. Actor-critic methods combine the advantages of value-based (critic-only) and policy-based (actor-only) methods, such as low variance and continuous action producing capability. The actor represents the policy structure, while the critic estimates the value function [18]. In our setting, we parameterize the value function by the parameter vector $\theta \in \Theta$ as $V_\theta(\beta)$, and the stochastic policy by $\xi \in \Xi$ as $q_{\boldsymbol{\xi}}$. The difference between the right and the left hand side of (23) is called temporal difference (TD) error, which represents the error between the critic's estimate and the target differing by one-step in time [21]. The TD error for the experience tuple $(\beta_t, a_t, y_t, \beta_{t+1}, \mathcal{C}_t)$ is estimated as

$$\delta_t = \mathcal{C}_t(\beta_t) + \gamma V_{\theta_t}(\beta_{t+1}) - V_{\theta_t}(\beta_t), \quad (24)$$

where $\mathcal{C}_t(\beta_t) + \gamma V_{\theta_t}(\beta_{t+1})$ is called the TD target, and γ is a discount factor that we choose very close to 1 to approximate the Bellman equation in (23) for our infinite-horizon average cost MDP. To implement RL in the infinite-horizon problem, we take sample averages over independent finite trajectories, which are generated by experience tuples at each time t via Monte-Carlo roll-outs.

Instead of using value functions in actor and critic updates, we use advantage function to reduce the variance in policy gradient methods. The advantage can be approximated by TD error. Hence, the critic is updated by gradient descent as:

$$\theta_{t+1} = \theta_t + \alpha_t^c \nabla_\theta \ell_c(\theta_t), \quad (25)$$

where $\ell_c(\theta_t) = \delta_t^2$ is the critic loss and α_t^c is the learning rate of the critic at time t. The actor is updated similarly as,

$$\xi_{t+1} = \xi_t - \alpha_t^a \nabla_\xi \ell_a(\xi_t), \quad (26)$$

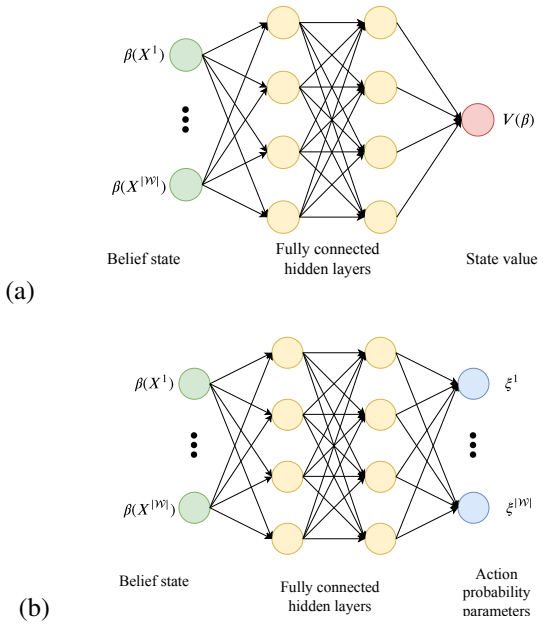

Fig. 3: Critic (a) and actor (b) neural network structures.

Algorithm 1: A2C-deep RL algorithm for online LPPM

Initialize the DNNs with random weights ξ and θ
Initialize environment E
for *episode*=$1, N$ **do**
  Initialize belief state β_0;
  for $t = 0, n$ **do**
    Sample action probability vector $a_t \sim Dirichlet(a|\xi)$ from the current policy;
    Perform the action and calculate cost $\mathcal{C}_{\xi_t}$ in E;
    Sample an observation y_t and calculate the next belief state β_{t+1} in E;
    Set TD target $\mathcal{C}_{\xi_t} + \gamma V^{\xi}_{\theta_t}(\beta_{t+1})$;
    Minimize the loss $\ell_c(\theta) = \delta^2 = (\mathcal{C}_{\xi_t} + \gamma V^{\xi}_{\theta_t}(\beta_{t+1}) - V^{\xi}_{\theta_t}(\beta_t))^2$;
    Update the critic $\theta \leftarrow \theta + \alpha^c \nabla_\theta \delta^2$;
    Minimize the loss $\ell_a(\xi_t) = (\ln(Dirichlet(a|\xi_t))\delta_t)$;
    Update the actor $\xi \leftarrow \xi - \alpha^a \nabla_\xi \ell_a(\xi_t)$;
    Update the belief state $\beta_{t+1} \leftarrow \beta_t$
  end
end

where $\ell_a(\xi_t) = -\ln(q_s(y_t|\beta_t, \xi_t))\delta_t$ is the actor loss and α^a_t is the actor's learning rate. This method is called *advantage actor-critic RL*.

In our A2C-deep RL implementation, we represent the actor and critic mechanisms by fully connected feed-forward deep neural networks (DNNs) with two hidden layers as illustrated in Fig. 3. The critic DNN takes the current belief state $\beta(\boldsymbol{X})$ as input, where $\boldsymbol{X}$ is the location vector of size $|\mathcal{W}|$, and outputs the value of the belief state for the current action probabilities $V^{\xi}_{\theta}(\beta)$. The actor takes the belief state as input, and outputs the parameters used for determining the action probabilities of the corresponding belief. Here, $\{\xi^1, \ldots, \xi^{|\mathcal{W}|}\}$ are the concentration parameters of a Dirichlet distribution which represent the action probabilities. The overall A2C deep RL algorithm for online LPPM is described in Algorithm 1.

IV. Numerical Results

In this section, we evaluate the performance of the proposed LPPM policy for a simple grid-world example, and compare the results with the myopic Markovian location release mechanism proposed in [7]. In [7], an upper bound on the privacy-utility trade-off is given by a myopic policy as follows:

$$\sum_{t=1}^{n} \min_{\substack{q(y_t|x_t,x_{t-1},y_{t-1}):\\ \mathbb{E}^q[d(x_t,y_t)]\leq \bar{D}}} I^q(X_t, X_{t-1}; Y_t|Y_{t-1}). \tag{27}$$

Exploiting the fact that (27) is similar to the rate-distortion function, Blahut-Arimoto algorithm is used in [7] to minimize the conditional mutual information at each time step. Finite-horizon solution of the objective function (27) is obtained by applying alternating minimization sequentially. In our simulations, we obtained the average information leakage and distortion for this approach by normalizing for $n = 300$.

We consider a simple 4×4 grid-world, where $|\mathcal{W}|$=16. The cells are numbered such that the first and the last rows of the grid-world are represented by $\{1, 2, 3, 4\}$ and $\{13, 14, 15, 16\}$, respectively. User's trajectory forms a first-order Markov chain with the transition probability matrix $\boldsymbol{Q_x}$. The user can start its movement at any square with equal probability $p_{x_1} = \frac{1}{16}$. The Lagrangian multiplier $\lambda \in [0, 20]$ denotes the user's choice of privacy-utility balance. We train two fully connected feed-forward DNNs, representing the actor and critic, by utilizing ADAM optimizer [22]. Both networks contain two hidden layers with leaky-ReLU activation [23]. Distortion is measured by the Manhattan distance between x_t and y_t. We obtain the corresponding privacy-utility trade-off by averaging the total information leakage and distortion over a horizon of $n = 300$.

In Fig. 4, privacy-distortion trade-off curves are obtained assuming that $\boldsymbol{Q}^0_x$, $\boldsymbol{Q}^1_x$ and $\boldsymbol{Q}^2_x$ are 16×16 Markov transition matrices with different correlation levels. In all three cases, the user can move from any square to any square at each step. While all the transition probabilities are equal, i.e. $\frac{1}{|\mathcal{W}|}$, for $\boldsymbol{Q}^0_x$, the probability of the user moving to a closer square is greater than taking a larger step to a more distant one for $\boldsymbol{Q}^1_x$ and $\boldsymbol{Q}^2_x$. Moreover, $\boldsymbol{Q}^1_x$ represents a more uniform trajectory, where the agent moves to equidistant cells with equal probability, while with $\boldsymbol{Q}^2_x$ the agent is more likely to follow a certain path, i.e., the random trajectory generated by $\boldsymbol{Q}^2_x$ has lower entropy. The transition probabilities with $\boldsymbol{Q}^1_x$ are given by:

$$q_x(x_t|x_{t+1}) = \frac{r_{d(x_t,x_{t+1})}/d(x_t, x_{t+1})}{\sum_{x_{t+1}} r_{d(x_t,x_{t+1})}/d(x_t, x_{t+1})}, \tag{28}$$

where $d(x_t, x_{t+1})$ is the Manhattan distance between the user positions at time t and $t+1$; $r_{d(x_t,x_{t+1})}$ is a scalar which determines the probability of the user moving from one grid to the

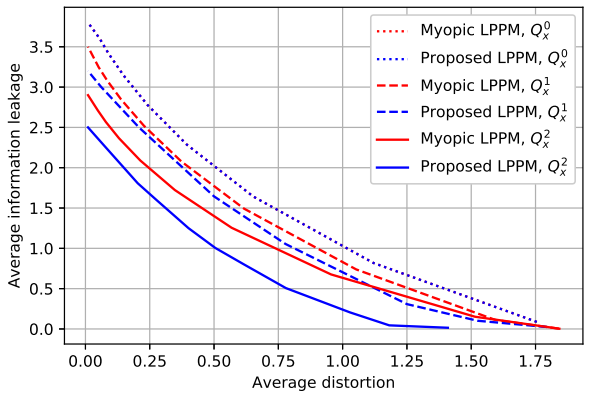

Fig. 4: Average information leakage as a function of the average distortion for the myopic and proposed LPPM policies.

equidistant grids in the next step. Fig. 4 is obtained by setting $r_0 = 1$ and $r_i = 7 - i$, $i = 1, \ldots, 6$. Furthermore, we generate $\boldsymbol{Q}_x^2$ such that $q_x(x_t|x_{t+1}) = \frac{u(x_t,x_{t+1})/d(x_t,x_{t+1})}{\sum_{x_{t+1}} u(x_t,x_{t+1})/d(x_t,x_{t+1})}$, where, for $x_t \in \{1, 2, \ldots, 15\}$,

$$u(x_t, x_{t+1}) = \begin{cases} r_1, & \text{for } mod(x_t, 4) \neq 0,\ x_{t+1} = x_t + 1 \\ r_1, & \text{for } mod(x_t, 4) = 0,\ x_{t+1} = x_t + 4, \\ r_0, & \text{otherwise,} \end{cases}$$

$u(16, x_{t+1}) = r_0$ for $x_{t+1} \in \{1, \ldots, 15\}$, and $u(16, 16) = r_1$. As a result, temporal correlations in the location history increase in the order $\boldsymbol{Q}_x^0$, $\boldsymbol{Q}_x^1$, $\boldsymbol{Q}_x^2$.

We train our DNNs for a time horizon of $n = 300$ in each episode, and over 5000 Monte Carlo roll-outs. Fig. 4 shows that, for $\boldsymbol{Q}_x^2$ the proposed LPPM obtained through deep RL leaks much less information than the myopic LPPM for the same distortion level, indicating the benefits of considering all the history when taking actions at each time instant. This difference is less for $\boldsymbol{Q}_x^1$, since the temporal correlations in the location history is much less than $\boldsymbol{Q}_x^2$. Finally, both proposed and myopic LPPMs performances are the same for $\boldsymbol{Q}_x^0$, since the user movement with uniform distribution does not have temporal memory, and therefore, taking the history into account does not help.

V. Conclusions

We have studied the privacy-utility trade-off in LPPMs using mutual information as a privacy measure. Having identified some properties of the optimal policy, we recast the problem as an MDP. Due to continuous state and action spaces, it is challenging to characterize or even numerically compute the optimal policy. We overcome this difficulty by employing advantage actor-critic deep RL as a computational tool. Utilizing DNNs, we numerically evaluated the privacy-utility trade-off curve of the proposed location release policy. We compared the results with a myopic LPPM, and observed the effect of considering temporal correlations on information leakage-distortion performance. According to the simulation results, we have seen that the proposed LPPM policy provides significant privacy advantage, especially when the user trajectory has higher temporal correlations.

References

[1] V. Primault, A. Boutet, S. B. Mokhtar, and L. Brunie., "The long road to computational location privacy: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2772–2793, Oct. 2018.

[2] C.-Y. Chow and M. F. Mokbel, "Trajectory privacy in location-based services and data publication," *SIGKDD Explorations*, vol. 13, pp. 19–29, Aug. 2011.

[3] K. P. N. Puttaswamy, S. Wang, T. Steinbauer, D. Agrawal, A. E. Abbadi, C. Kruegel, and B. Y. Zhao, "Preserving location privacy in geosocial applications," *IEEE Transactions on Mobile Computing*, vol. 13, no. 1, pp. 159–173, Jan 2014.

[4] R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, and J.-P. Hubaux, "Unraveling an old cloak: k-anonymity for location privacy," in *ACM Conference on Computer and Communications Security*, Sep. 2010.

[5] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y. Le Boudec, "Protecting location privacy: Optimal strategy against localization attacks," in *ACM Conference on Computer and Communications Security*, Oct. 2012, pp. 617–627.

[6] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *IEEE Symposium on Foundations of Computer Science*, Oct 2013, pp. 429–438.

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

[8] V. Bindschaedler and R. Shokri, "Synthesizing plausible privacy-preserving location traces," in *IEEE Symposium on Security and Privacy (SP)*, May 2016, pp. 546–563.

[9] W. Luo, Y. Lu, D. Zhao, and H. Jiang, "On location and trace privacy of the moving object using the negative survey," *IEEE Trans. on Emerging Topics in Comput. Intelligence*, vol. 1, no. 2, pp. 125–134, April 2017.

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

[11] R. Shokri, G. Theodorakopoulos, J. Le Boudec, and J. Hubaux, "Quantifying location privacy," in *IEEE Symposium on Security and Privacy*, May 2011, pp. 247–262.

[12] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey," *ACM Comput. Surv.*, vol. 51, no. 3, pp. 57:1–57:38, Jun. 2018.

[13] S. Li, A. Khisti, and A. Mahajan, "Information-theoretic privacy for smart metering systems with a rechargeable battery," *IEEE Transactions on Information Theory*, vol. 64, no. 5, pp. 3679–3695, May 2018.

[14] G. Giaconi and D. Gündüz, "Smart meter privacy with renewable energy and a finite capacity battery," in *IEEE Int. Workshop on Sig. Proc. Advances in Wireless Communications (SPAWC)*, July 2016, pp. 1–5.

[15] E. Erdemir, P. L. Dragotti, and D. Gündüz, "Privacy-cost trade-off in a smart meter system with a renewable energy source and a rechargeable battery," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brighton, UK, May 2019, pp. 2687–2691.

[16] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of markov decision processes," *Mathematics of Operations Research*, vol. 12, no. 3, pp. 441–450, 1987.

[17] N. Saldi, T. Linder, and S. Yüksel, *Approximations for Partially Observed Markov Decision Processes*. Cham: Springer International Publishing, 2018, pp. 99–123.

[18] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018.

[19] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II*, 3rd ed. Athena Scientific, 2007.

[20] V. R. Konda and J. N. Tsitsiklis, "On actor-critic algorithms," *SIAM J. Control Optim.*, vol. 42, no. 4, pp. 1143–1166, Apr. 2003.

[21] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1291–1307, Nov 2012.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[23] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.