2020 59th IEEE Conference on Decision and Control (CDC)
Jeju Island, Republic of Korea, December 14-18, 2020

Privacy-Preserving Policy Synthesis in Markov Decision Processes

Parham Gohari, Matthew Hale and Ufuk Topcu

Abstract— **In decision-making problems, the actions of an agent may reveal sensitive information that drives its decisions. For instance, a corporation's investment decisions may reveal its sensitive knowledge about market dynamics. To prevent this type of information leakage, we introduce a policy synthesis algorithm that protects the privacy of the transition probabilities in a Markov decision process. We use differential privacy as the mathematical definition of privacy. The algorithm first perturbs the transition probabilities using a mechanism that provides differential privacy. Then, based on the privatized transition probabilities, we synthesize a policy using dynamic programming. Our main contribution is to bound the "cost of privacy," *i.e.*, the difference between the expected total rewards with privacy and the expected total rewards without privacy. We also show that computing the cost of privacy has time complexity that is polynomial in the parameters of the problem. Moreover, we establish that the cost of privacy increases with the strength of differential privacy protections, and we quantify this increase. Finally, numerical experiments on two example environments validate the established relationship between the cost of privacy and the strength of data privacy protections.**

I. INTRODUCTION

In many decision-making problems, agents desire to protect sensitive information that drives their actions from eavesdroppers and adversaries, such as applications in autonomous driving or smart power grids [1], [2]. In these applications, as well as in many other sequential decision-making problems, choosing actions can be cast as a policy-synthesis problem wherein the environment is modeled as a Markov decision process (MDP) [3], [4]. The goal in a policy-synthesis problem is to find a reward-maximizing control policy based on the transition probabilities of the underlying MDP. In this work, we study the problem of synthesizing a policy that protects the privacy of the transition probabilities.

Transition probabilities in an MDP govern the dynamics of the environment and may carry information that should be protected during policy synthesis. For example, suppose that through market research, a corporation discovers a niche in the market and decides to invest. Such an investment may alert competitors to the discovered niche and leads to other firms making similar decisions. Previous works in economics have associated higher market shares with profitability [5], [6]. Therefore, competitors' entrance to the market may be harmful to the investing corporation. As a result, it is often crucial for a decision-maker to choose actions that do not reveal its knowledge about its environment dynamics.

We use differential privacy as the definition of privacy for an MDP's transition probabilities. Differential privacy, first introduced in [7], is a property of an algorithm and has been used in the computer science literature as a quantitative definition of privacy for databases [8], [9]. It has also recently been used in control theory [10], [11]. Differential privacy makes it unlikely that the output of a differentially private algorithm will reveal any useful information about the individual entries of the input dataset; however, it may still pass on information about the aggregate statistics of the input dataset that are useful in down-stream analytics.

The main contribution of this paper is to develop a policy-synthesis algorithm that enforces differential privacy for transition probabilities with adjustable privacy and utility. We define the utility of a privacy-preserving policy synthesis algorithm to be the value function associated with the policy, which in an MDP is its expected total reward [12]. Utility loss due to privacy is a common phenomenon, and we follow the convention in the differential privacy literature to analyze the utility of the privacy-preserving algorithm by comparing it to its non-private counterpart [13], [14].

In order to show that the algorithm enforces differential privacy, we exploit the fact that differential privacy is immune to post-processing [13]. By immunity to post-processing, we mean that arbitrary functions of the output of a differentially private algorithm do not weaken its privacy guarantees. The algorithm first privatizes the transition probabilities via the Dirichlet mechanism [15]. We then use dynamic programming to synthesize a policy based on the privatized transition probabilities. Since the dynamic programming stage is an act of post-processing on the output of a differentially private mechanism, its output preserves the differential privacy provided to transition probabilities.

We employ the Dirichlet mechanism for privatization because it preserves the unique structure of the transition probabilities, *i.e.*, vectors with non-negative components that sum to one. Using traditional differentially private mechanisms that add infinite-support noise to transition probabilities are ill-suited to this work as they break their structure. For example, they can result in a transition probability vector with negative components. Although normalization may seem a fitting solution in order to project the perturbed vector back onto the unit simplex, we avoid normalization because it makes it difficult to quantify utility.

We introduce the "cost of privacy" as a measure of the utility of the algorithm. We define the cost of privacy

P. Gohari is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX. U. Topcu is with Faculty of Oden Institute for Computational Engineering, The University of Texas at Austin, Austin, TX. email: {pgohari, utopcu}@utexas.edu. M. Hale is with Faculty of Mechanical and Aerospace Engineering at the University of Florida, Gainseville, FL. email: matthewhale@ufl.edu.

All authors were supported by AFRL FA9550-19-1-0169. PG and UT were also supported by DARPA D19AP00004 and ARL ACC-APG-RTP W911NF. MH was also supported by NSF CAREER Grant 1943275.

978-1-7281-7447-1/20/$31.00 ©2020 IEEE

to be the difference between the expected total rewards of the policy with privacy and that of the same policy without privacy. Since we perturb the transition probabilities to enforce differential privacy, the output of the dynamic-programming stage is susceptible to suboptimality, which the cost of privacy quantifies.

We bound the cost of privacy for both finite- and infinite-horizon MDPs, thereby enabling a decision-maker to control the level of privacy based on the utility loss that they tolerate. For finite-horizon MDPs, we show that we can compute the cost of privacy in polynomial time via a backward-in-time recursive algorithm. For the case of infinite-horizon MDPs, we show that an algorithm similar to policy evaluation converges to the cost of privacy asymptotically. We show that the number of iterations required to approximate the cost of privacy is polynomial in problem parameters.

In order to empirically validate the expressions that we introduce for the cost of privacy, we run the algorithm on two example MPDs. The first example is a small MDP that models a corporation's investment. The second example is an MDP with a larger state and action space, with its transition probabilities generated randomly. We run the algorithm at a range of privacy levels and visualize our results by plotting the cost of privacy versus privacy level. The results illustrate the trade-off that we establish between the strength of data privacy and utility. Furthermore, we observe that the bounds we provide for the cost of privacy are meaningful in the sense that they empirically provide a close approximation to the cost of privacy.

Related work. The works in [16]–[18] study the problem of learning a policy in an MDP while enforcing differential privacy. The key difference between this paper and the works above is that we protect the transition probabilities which belong to the probability simplex, whereas the other works protect the sensory data that are scalars. We emphasize that although scalars can be readily privatized using traditional differentially private mechanisms, transition probabilities need to be treated specially to ensure that they remain non-negative and sum to one.

The problems of robust and distributionally robust MDPs are related to this paper. Robust policy synthesis in an MDP is the problem of synthesizing a policy that mitigates uncertainties present in transition probabilities [19], [20]. Distributionally robust MDPs assume that the planner has access to a probability measure over the uncertainty sets [21].

We base the cost of privacy bounds on a concentration bound that we derive for the output of the Dirichlet mechanism. Finding the worst-case cost of privacy coincides with lower bounding the value of a distributionally robust policy where the uncertainties in the transition probabilities adhere to the concentration bound of the Dirichlet mechanism.

II. PRELIMINARIES

In this section we set the notation and definitions used throughout the paper.

A. Notation

We denote the set of real numbers by $\mathbb{R}$. Let $(\cdot)^T$ denote the transpose of a vector. We define the unit simplex to be $\Delta(n) := \{x \in \mathbb{R}^n \mid \mathbf{1}^T x = 1, x \geq 0\}$, where $\mathbf{1}$ is the vector of all ones in $\mathbb{R}^n$ and the inequality is evaluated element wise. We use the notation $\Delta^\circ(n)$ to denote the interior of $\Delta(n)$. For a finite set $\mathcal{A}$, its cardinality is denoted by $|\mathcal{A}|$. $\mathbf{E}[\cdot]$ and $\mathrm{Var}(\cdot)$ denote the expectation and the variance of a random variable, respectively. $\|\cdot\|_1$ and $\|\cdot\|_\infty$ denote the one and infinity norm of a vector, respectively. For a vector p, we use the notation p_i to denote the i^{th} component of p. We use the gamma function

$$\Gamma(z) := \int_0^\infty x^{z-1} \exp(-x) \mathrm{d}x.$$

B. Markov decision processes

An MDP is a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$ where $\mathcal{S}$ is the set of states, $\mathcal{A}_s$ is the set of available actions at state $s \in \mathcal{S}$, and $r : \mathcal{S} \times \mathcal{A}_s \mapsto \mathbb{R}$ is the reward function that indicates the one-step reward for taking action a at state s. $\mathcal{P} := \{P(s,a) \in \Delta(|\mathcal{S}|) \mid (s,a) \in \mathcal{S} \times \mathcal{A}_s\}$ is the set of transition probabilities. Finally, T is the time horizon and γ is the discount factor.

We now define a policy, that is, a rule for making a sequence of decisions in an MDP. In particular, let $h_t := \{s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_t\}$ be a history until stage t, and let $\mathcal{H}_t(s_t)$ denote the set of all possible histories that end in state s_t. A policy $\pi : \mathcal{H}_t(s_t) \mapsto \Delta(|\mathcal{A}_{s_t}|)$ maps a history h_t to a probability distribution over the set of actions, $\mathcal{A}_{s_t}$.

A policy π is evaluated by its value function $V_t^\pi : \mathcal{S} \mapsto \mathbb{R}$, that is defined as

$$V_t^\pi(s) := \mathbf{E}\left[\sum_{i=t}^{T} \gamma^{i-t} r_i \,\middle|\, s_t = s\right].$$

The expectation is taken over the stochasticity of the policy π and transition probabilities $\mathcal{P}$. We study the problem of privacy-preserving policy synthesis, and in a synthesis problem, the goal is to find an optimal policy with the highest value function beginning at initial state s_0.

In this paper, we restrict our attention to Markovian policies, *i.e.*, the class of policies that only depend on the most recent state of the history. Markovian policies are shown to be optimal under some mild conditions [22]. We use the notation $\pi_t(a \mid s)$ to show the probability of taking action a at state s and stage t.

C. Differential privacy

For an algorithm that satisfies differential privacy, it is unlikely to tell apart *nearby* input datasets based on observations of the algorithm's output. Nearby datasets are defined formally by an adjacency relationship. We first state the adjacency relationship used in this paper.

Definition 1 (From [15], Definition 1)**.** *For a constant $b \in (0,1]$, two vectors $p, q \in \Delta(n)$ are said to be b-adjacent if there exist indices i, j such that $p_{-(i,j)} = q_{-(i,j)}$ and $\|p - q\|_1 \leq b$.*

The above definition considers two vectors in the unit simplex adjacent if they only differ in two indices, i, j, by no more than b in their 1-norm. Note that the usual adjacency relationship in the differential privacy literature considers two input datasets adjacent if they only differ in one entry [13]; however, it is not possible for the elements of the unit simplex to differ in only one entry because their components must sum to one.

Definition 2 (Probabilistic differential privacy [15])**.** *Fix a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and $b \in (0, 1]$. A mechanism $\mathcal{M} : \Delta(n) \times \Omega \mapsto \Delta(n)$ is said to be probabilistically (ϵ, δ)-differentially private if, for all $p \in \Delta(n)$, we can partition the output space $\Delta(n)$ into two disjoint sets, Ω_1, Ω_2, such that $\mathbb{P}[\mathcal{M}(p) \in \Omega_2] \leq \delta$, and for all $q \in \Delta(n)$ b-adjacent to p, we have that*

$$\log\left(\frac{\mathbb{P}[\mathcal{M}(p) = x]}{\mathbb{P}[\mathcal{M}(q) = x]}\right) \leq \epsilon, \forall x \in \Omega_1.$$

Probabilistic (ϵ, δ)-differential privacy is known to imply ordinary (ϵ, δ)-differential privacy [23].

D. The Dirichlet mechanism

A Dirichlet mechanism with parameter $k > 0$ takes as input a vector $p \in \Delta_n^\circ$ and outputs $x \in \Delta(n)$ according to a Dirichlet probability distribution. Fix k and let $\mathcal{M}_D^{(k)}$ denote the Dirichlet mechanism. Then

$$\mathbb{P}\left[\mathcal{M}_D^{(k)}(p) = x\right] = \frac{1}{\mathrm{B}(kp)} \prod_{i=1}^{n-1} x_i^{kp_i - 1} \left(1 - \sum_{i=1}^{n-1} x_i\right)^{kp_n - 1},$$

where $\mathrm{B}(kp) := \prod_{i=1}^{n} \Gamma(kp_i) / \Gamma\left(k \sum_{i=1}^{n} p_i\right)$ is the multi-variate beta function.

The Dirichlet mechanism satisfies probabilistic (ϵ, δ)-differential privacy [15], and has the following properties. The expected value of the output is equal to the input vector, *i.e.*, $\mathbf{E}\left[\mathcal{M}_D^{(k)}(p)\right] = p$. An increase in k results in weaker differential privacy protections, and in particular it increases ϵ. However, as k increases, the output becomes more concentrated around the input vector p.

III. Privacy-preserving synthesis algorithm

In this section, we first present the proposed privacy-preserving synthesis algorithm. Then, we show the differential privacy of the algorithm.

A. Algorithm

The algorithm takes as input an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$ and the value of k that is the parameter for the Dirichlet mechanism. It then outputs a policy $\bar{\pi}$ and its value function $\bar{V}^{\bar{\pi}}$. The algorithm comprises two stages. The first stage privatizes the transition probabilities by applying the Dirichlet mechanism independently on each transition probability vector in $\mathcal{P}$. Let $\bar{\mathcal{P}} := \left\{\bar{P}(s,a) = \mathcal{M}_D^{(k)}\left(P(s,a)\right) \mid P(s,a) \in \mathcal{P}\right\}$ be the set of transition probabilities after privatization. The second stage finds an optimal policy and the optimal value of the privatized MDP $\bar{\mathcal{M}} := \left(\mathcal{S}, \mathcal{A}_s, r, \bar{\mathcal{P}}, T, \gamma\right)$. An optimal policy is one that satisfies the Bellman condition of optimality, and the optimal value is the value of such policies [22]. In the case of a finite-horizon MDP, the second stage finds $(\bar{\pi}, \bar{V}_t)$ such that for all $t \in \{0, \ldots, T-1\}$ and all $s \in \mathcal{S}$,

$$\bar{V}_t(s) = \max_{\pi} \sum_{a \in \mathcal{A}_s} \pi(a \mid s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s,a,s') \bar{V}_{t+1}(s')\right),$$

$$\bar{\pi}_t \in \arg\max_{\pi} \sum_{a \in \mathcal{A}_s} \pi(a \mid s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s,a,s') \bar{V}_{t+1}(s')\right),$$

where $\bar{P}(s,a,s')$ denotes the privatized probability that taking action a at state s takes the agent to state s'. We assume that the terminal values are given by a known function $R_T : \mathcal{S} \mapsto \mathbb{R}$, *i.e.*, $\bar{V}_T(s) = R_T(s)$, for all $s \in \mathcal{S}$.

For an infinite-horizon discounted MDP, it can be shown that the optimal policy is a stationary policy, *i.e.*, a policy that adopts the same decision rule at all stages [22]. Let $\bar{V}_\infty$ denote the optimal value of $\bar{\mathcal{M}}$. Then, for an infinite-horizon MDP, the second stage of Algorithm 1 computes $(\bar{\pi}, \bar{V}_\infty)$ such that for all $s \in \mathcal{S}$,

$$\bar{V}_\infty(s) = \max_{\pi} \sum_{a \in \mathcal{A}_s} \pi(a \mid s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s,a,s') \bar{V}_\infty(s')\right),$$

$$\bar{\pi} \in \arg\max_{\pi} \sum_{a \in \mathcal{A}_s} \pi(a \mid s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s,a,s') \bar{V}_\infty(s')\right).$$

There are various methods suggested to efficiently compute $\bar{\pi}$ and its value function, such as dynamic programming or linear programming [22]. The third and the fourth step of Algorithm 1 may adopt any of these methods to synthesize and evaluate an optimal policy for the privatized MDP $\bar{\mathcal{M}}$.

Algorithm 1: Privacy-preserving synthesis algorithm

Input: $(\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$, k
Output: $\bar{\pi}$, $\bar{V}^{\bar{\pi}}$
1 Construct the set of privatized transition probabilities $\bar{\mathcal{P}} := \left\{\bar{P}(s,a) = \mathcal{M}_D^{(k)}\left(P(s,a)\right) \mid P(s,a) \in \mathcal{P}\right\}$.
2 Replace $\mathcal{M}$ with its privatized version $\bar{\mathcal{M}} := \left(\mathcal{S}, \mathcal{A}_s, r, \bar{\mathcal{P}}, T, \gamma\right)$
3 Synthesize policy $\bar{\pi}$ for $\bar{\mathcal{M}}$.
4 Compute the value function of $\bar{\pi}$, $\bar{V}^{\bar{\pi}}$.

B. Proof of differential privacy

We prove that Algorithm 1 is (ϵ, δ)-differentially private by differential privacy's immunity to post-processing.

Lemma 1 (From [13], Proposition 2.1)**.** *Let $\mathcal{M} : \Delta(n) \mapsto \Delta(n)$ be a mechanism that is (ϵ, δ)-differentially private. Let $f : \Delta(n) \mapsto \mathbb{R}$ be an arbitrary mapping. Then, $f \circ \mathcal{M} : \Delta(n) \mapsto \mathbb{R}$ is (ϵ, δ)-differentially private.*

Recall that probabilistic (ϵ, δ)-differential privacy implies ordinary (ϵ, δ)-differential privacy. Let $(\hat{\epsilon}, \hat{\delta})$ denote the level of the probabilistic differential privacy of the Dirichlet mechanism employed in Algorithm 1. By Lemma 1, the algorithm is $(\hat{\epsilon}, \hat{\delta})$-differentially private because the synthesis step is an instance of a post-processing mapping f.

IV. UTILITY ANALYSIS

Algorithm 1 synthesizes a policy $\bar{\pi}$ based on privatized transition probabilities in $\bar{\mathcal{P}}$. It then computes the value function of $\bar{\pi}$, $\bar{V}_t^{\bar{\pi}}$, using $\bar{\mathcal{P}}$. Let $V_t^{\bar{\pi}} : \mathcal{S} \mapsto \mathbb{R}$ be the value function that the non-private transition probabilities in $\mathcal{P}$ assign to $\bar{\pi}$. The utility of Algorithm 1 is equal to $V^{\bar{\pi}}(s_0)$.

We assume that after the privatization stage, the algorithm loses access to the non-private transition probabilities in $\mathcal{P}$. The reason is that in many real-world applications, a central cloud is used to compute the policy, and agents submit their data to the cloud [24], [25]. For agents to preserve their data privacy, they privatize their data prior to any submission to the cloud [26].

We start off the utility analysis of Algorithm 1 with introducing a concentration bound on the output of the Dirichlet mechanism. Due to space restrictions, the proofs of the subsequent lemmas and theorems are omitted and can be found in [27].

Lemma 2. *Let $\mathcal{M}_D^{(k)}$ denote a Dirichlet mechanism with parameter $k \in \mathbb{R}_+$. Then, for all $\beta > 0$, and all $p \in \Delta^\circ(n)$,*

$$\mathbb{P}\left(\left\|\mathcal{M}_D^{(k)}(p) - p\right\|_\infty \geq \sqrt{\frac{\log(1/\beta)}{2(k+1)}}\right) \leq \beta.$$

The above lemma enables us to evaluate $V^{\bar{\pi}}(s_0)$, *i.e.*, the conditional expectation of the value function without privacy, based on the privatized transition probabilities $\bar{\mathcal{P}}$ and k. In particular, for a finite-horizon MDP we provide an upper bound on $\left|\mathbf{E}\left[V_0^{\bar{\pi}}(s_0) \mid \bar{\mathcal{P}}, k\right] - \bar{V}_0^{\bar{\pi}}(s_0)\right|$. For an infinite-horizon MDP, we upper bound $\left|\mathbf{E}\left[V_\infty^{\bar{\pi}}(s_0) \mid \bar{\mathcal{P}}, k\right] - \bar{V}_\infty^{\bar{\pi}}(s_0)\right|$. We refer to both expressions as the "cost of privacy."

The bounds are based on the pessimistic and optimistic value functions that possible transition-probability vectors generate. Let $\alpha := \sqrt{\log(1/\beta)/2(k+1)}$, then Lemma 2 implies that for all $P(s,a) \in \mathcal{P}$ and the corresponding $\bar{P}(s,a) \in \bar{\mathcal{P}}$, $\mathbb{P}\left(||\bar{P}(s,a) - P(s,a)||_\infty \leq \alpha\right) \geq 1 - \beta$. For all $(s,a) \in \mathcal{S} \times \mathcal{A}$, we define

$$\hat{\mathcal{P}}_{\alpha,\beta}(s,a) := \{\beta P_1(s,a) + (1-\beta) P_2(s,a) \mid \|P_2(s,a) - \bar{P}(s,a)\|_\infty \leq \alpha, P_1(s,a) \in \Delta(|\mathcal{S}|)\}. \quad (1)$$

We use the sets $\hat{\mathcal{P}}_{\alpha,\beta}$ to compute a pessimistic and optimistic value function to bound the cost of privacy.

A. Finite-horizon MDPs

We bound the cost of privacy for a finite-horizon MDP by establishing a common upper and lower bound for both $\mathbf{E}\left[V_0^{\bar{\pi}}(s_0) \mid \bar{\mathcal{P}}, k\right]$ and $\bar{V}_0^{\bar{\pi}}(s_0)$. We first state a technical lemma that we later use to prove the theorems of this section.

Lemma 3. *Fix k and a set of transition probabilities $\mathcal{P}$, and let $\bar{\mathcal{P}} := \left\{\bar{P}(s,a) = \mathcal{M}_D^{(k)}(P(s,a)) \mid P(s,a) \in \mathcal{P}\right\}$. For any $\beta > 0$, let $\alpha := \sqrt{\log(1/\beta)/2(k+1)}$. Then,*

$$\bar{P}(s,a) \in \hat{\mathcal{P}}_{\alpha,\beta}(s,a), \quad \forall \bar{P}(s,a) \in \bar{\mathcal{P}},$$

$$\mathbf{E}\left[P(s,a) \mid \bar{\mathcal{P}}, k\right] \in \hat{\mathcal{P}}_{\alpha,\beta}(s,a), \quad \forall P(s,a) \in \mathcal{P}.$$

Theorem 1. *Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$ and k be the input, and $(\bar{\pi}, \bar{V}_t^{\bar{\pi}})$ be the output of Algorithm 1, and let $T < \infty$. Fix $\beta > 0$, and let $\alpha := \sqrt{\log(1/\beta)/2(k+1)}$. Let $R_T : \mathcal{S} \mapsto \mathbb{R}$ denote the terminal value function of $\mathcal{M}$. Define $\underline{v}_t^{\bar{\pi}} : \mathcal{S} \mapsto \mathbb{R}$ and $\bar{v}_t^{\bar{\pi}} : \mathcal{S} \mapsto \mathbb{R}$ as follows. For all $s \in \mathcal{S}$, let $\bar{v}_T^{\bar{\pi}}(s) := R_T(s)$, $\underline{v}_T^{\bar{\pi}}(s) := R_T(s)$, and for all $t \in \{0, \ldots, T-1\}$, let*

$$\underline{v}_t^{\bar{\pi}}(s) := \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \min_{p \in \hat{\mathcal{P}}_{\alpha,\beta}} \sum_{s' \in \mathcal{S}} p(s,a,s') \underline{v}_{t+1}^{\bar{\pi}}(s')\right),$$

$$\bar{v}_t^{\bar{\pi}}(s) := \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \max_{p \in \hat{\mathcal{P}}_{\alpha,\beta}} \sum_{s' \in \mathcal{S}} p(s,a,s') \bar{v}_{t+1}^{\bar{\pi}}(s')\right).$$

Then, we have that

$$\left|\mathbf{E}\left[V_0^{\bar{\pi}}(s_0) \mid \bar{\mathcal{P}}, k\right] - \bar{V}_0^{\bar{\pi}}(s_0)\right| \leq \bar{v}_0^{\bar{\pi}}(s_0) - \underline{v}_0^{\bar{\pi}}(s_0).$$

B. Infinite-horizon MDPs

In this section, we bound the cost of privacy for an infinite-horizon MDP. We first state a technical lemma, which we later use to bound the cost of privacy for infinite-horizon MDPs.

Lemma 4. *Fix k and an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$, and let $\bar{\mathcal{P}} := \left\{\bar{P}(s,a) = \mathcal{M}_D^{(k)}(P(s,a)) \mid P(s,a) \in \mathcal{P}\right\}$. For any $\beta > 0$, let $\alpha := \sqrt{\log(1/\beta)/2(k+1)}$. Define mappings $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3 : \mathbb{R}^{|\mathcal{S}|} \times \mathbb{R}^{|\mathcal{S}|}$ as*

$$\mathcal{L}_1 \mathbf{v} := \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \min_{p \in \hat{\mathcal{P}}_{\alpha,\beta}} \sum_{s' \in \mathcal{S}} p(s,a,s') \mathbf{v}(s')\right),$$

$$\mathcal{L}_2 \mathbf{v} := \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \bar{P}(s,a,s') \mathbf{v}(s')\right),$$

$$\mathcal{L}_3 \mathbf{v} := \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathbf{E}\left[P(s,a,s') \mathbf{v}(s') \middle| \bar{\mathcal{P}}, k\right]\right).$$

Then, mappings $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ are γ-contraction mappings, i.e., for all $\mathbf{v_1}, \mathbf{v_2} \in \mathbb{R}^{|\mathcal{S}|}$ and $i \in \{1,2,3\}$,

$$\|\mathcal{L}_i \mathbf{v_1} - \mathcal{L}_i \mathbf{v_2}\|_\infty \leq \gamma \|\mathbf{v_1} - \mathbf{v_2}\|_\infty.$$

Theorem 2. *Let $\mathcal{M} = (\mathcal{S}, \mathcal{A}_s, r, \mathcal{P}, T, \gamma)$ and k be the input, and $(\bar{\pi}, \bar{V}_\infty^{\bar{\pi}})$ be the output of Algorithm 1, and let $T = \infty$. Fix $\beta > 0$, and let $\alpha := \sqrt{\log(1/\beta)/2(k+1)}$. For all $s \in \mathcal{S}$, let $\underline{v}_\infty^{\bar{\pi}} : \mathcal{S} \mapsto \mathbb{R}$ and $\bar{v}_\infty^{\bar{\pi}} : \mathcal{S} \mapsto \mathbb{R}$ satisfy*

$$\underline{v}_\infty^{\bar{\pi}}(s) = \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \min_{p \in \hat{\mathcal{P}}_{\alpha,\beta}} \sum_{s' \in \mathcal{S}} p(s,a,s') \underline{v}_\infty^{\bar{\pi}}(s')\right),$$

$$\bar{v}_\infty^{\bar{\pi}}(s) = \sum_{a \in \mathcal{A}_s} \bar{\pi}(a \mid s)\left(r(s,a) + \gamma \max_{p \in \hat{\mathcal{P}}_{\alpha,\beta}} \sum_{s' \in \mathcal{S}} p(s,a,s') \bar{v}_\infty^{\bar{\pi}}(s')\right).$$

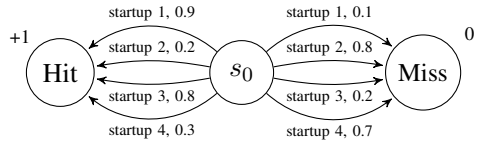


Fig. 1: The corporation's investment model with four possible startups to acquire, given as an MDP.

Then, we have that

$$\left|\mathbf{E}\left[V_\infty^{\bar{\pi}}(s_0) \mid \bar{\mathcal{P}}, k\right] - \bar{V}_\infty^{\bar{\pi}}(s_0)\right| \leq \bar{v}_\infty^{\bar{\pi}}(s_0) - \underline{v}_\infty^{\bar{\pi}}(s_0).$$

V. COMPUTATIONAL COMPLEXITY

In the previous section, we introduced expressions that bound the cost of privacy for both finite- and infinite-horizon MDPs. The bounds do not take a closed form, and they are computed by iterative methods. In this section, we show that the computational complexity of computing the cost of privacy is polynomial in problem parameters for both cases.

A. Finite-horizon MDPs

Revisiting the definition of $\underline{v}_t^{\bar{\pi}}$ and $\bar{v}_t^{\bar{\pi}}$ in Theorem 1, the inner minimization or maximization problem must be solved for each of T stages and $|\mathcal{S}|$ states. We first consider computing the lower bound $\underline{v}_t^{\bar{\pi}}$. Fix $(s,a) \in \mathcal{S} \times \mathcal{A}_s$ and $t \in \{0,\dots,T-1\}$. Then the inner minimization problem can be recast as

$$\min_{P_1,P_2,p\in\mathbb{R}^{|\mathcal{S}|}} \quad \sum_{s'\in\mathcal{S}} p(s,a,s')\underline{v}_t^{\bar{\pi}}(s') \tag{P}$$

$$\begin{aligned}
\text{subject to} \quad & \mathbf{1}^T P_1(s,a) = 1, P_1(s,a,s') \geq 0, && \forall s' \in \mathcal{S},\\
& \mathbf{1}^T P_2(s,a) = 1, P_2(s,a,s') \geq 0, && \forall s' \in \mathcal{S},\\
& \mathbf{1}^T p(s,a) = 1, p(s,a,s') \geq 0, && \forall s' \in \mathcal{S},\\
& P_2(s,a,s') - \bar{P}(s,a,s') \leq \alpha, && \forall s' \in \mathcal{S},\\
& P_2(s,a,s') - \bar{P}(s,a,s') \geq -\alpha, && \forall s' \in \mathcal{S},\\
& \beta P_1(s,a) + (1-\beta)P_2(s,a) = p(s,a).
\end{aligned}$$

The above optimization problem is a linear program (LP) with $3|\mathcal{S}|$ variables and $6|\mathcal{S}|+3$ constraints. Similarly, the inner maximization problem in $\bar{v}_t^{\bar{\pi}}$ can be cast as an LP by negating the objective function in (P).

Considering the interior-point method that is known to solve an LP in $\mathcal{O}(n^{3.5})$ time, where n is the number of variables [28], the computational complexity of computing the cost of privacy for a finite-horizon MDP is $\mathcal{O}\left(T|\mathcal{S}|^{4.5}|\mathcal{A}_s|\right)$.

B. Infinite-horizon MDPs

in Lemma 4, we introduced $\mathcal{L}_1$ that is a γ-contraction mapping. Suppose there exists $R \in \mathbb{R}_+$ such that the reward function of the underlying MDP satisfies $|r(s,a)| \leq R$, for all $(s,a) \in \mathcal{S} \times \mathcal{A}_s$. Then, all the value functions including the private, non-private, optimistic, and the pessimistic value function must be bounded above by a constant $v_{\max}$. Let $v_1^{(k)}$ be the value of the k^{th} iteration corresponding to $\mathcal{L}_1$, and v_1^∞ be the limiting value. We can write

$$\left\|v_1^{(k)} - v_1^\infty\right\|_\infty \leq \frac{\gamma^k}{1-\gamma}\left\|v_1^{(1)} - v_1^{(0)}\right\|_\infty \leq 2v_{\max}\frac{\gamma^k}{1-\gamma}.$$

The above inequality indicates that in order to reach an ϵ-approximation of the limit, $\mathcal{O}(\log(1/\epsilon))$ iterations are required. The inner minimization problem is identical to the finite-horizon case, which we reformulated as an LP in (P). Combining the above arguments together, we conclude that the required number of iterations such that $\|v_1^{(k)} - \underline{v}_\infty^{\bar{\pi}}\|_\infty \leq \epsilon$, is $\mathcal{O}\left(|\mathcal{S}|^{4.5}|\mathcal{A}_s|\log(1/\epsilon)\right)$. The same computational complexity holds for the upper bound $\bar{v}_\infty^{\bar{\pi}}$.

VI. NUMERICAL RESULTS

In this section, we empirically validate the developments of previous sections, wherein we introduced the expressions that compute the cost of privacy and their corresponding computational complexity. We apply Algorithm 1 to two example MDPs at a range of k values, which represent a range of privacy protection levels. The first is a small-sized MDP that represents a simple model for a corporation's investment planning. The second example is an MDP with a larger state and action space, which has transition probabilities, reward function, and terminal reward function generated randomly. For both examples, the algorithm is run 50 times, and Figure 2, which depicts the results, shows the mean values alongside their standard deviations that appear as error bars.

Example 1. Suppose a corporation has been tracking four startups, and it has to decide which startup to acquire. Assume that the corporation's model of each of the startup's probability of success is given by the MDP in Figure 1.

The first empirical result of this section corresponds to applying the algorithm to the scenario described in Example 1, and is depicted in Figure 2a.

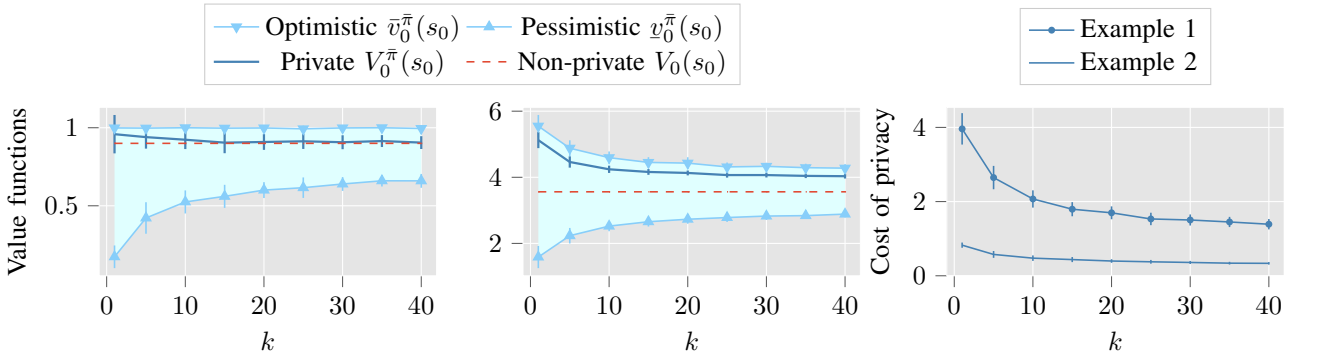
Example 2. In this example, we apply the algorithm to a larger MDP in order to test its scalability. In particular, the MDP has 20 states, 5 actions available at each state, and a time horizon of 10.

Figures 2a and 2b indicate that an increase in k improves the approximations of the private and non-private value by $\bar{v}_0^{\bar{\pi}}(s_0)$ and $\underline{v}_0^{\bar{\pi}}(s_0)$. Therefore the cost of privacy decreases with k, which Figure 2c confirms.

For both examples, the negative correlation between k and the cost of privacy is consistent with Lemma 2. An increase in k results in a tighter concentration bound on the output of the Dirichlet mechanism, and it lowers α in Theorems 1 and 2. A smaller α further restricts the inner optimization problem in (P); thus, it helps the optimistic and the pessimistic value functions to provide better approximations, which leads to a lower cost of privacy.

VII. CONCLUSION

We introduced a privacy-preserving policy synthesis algorithm that protects the privacy of the transition probabilities of its input MDP. The algorithm employs a Dirichlet mechanism to privatize the transition probabilities. We



a: Example 1 b: Example 2 c: The cost of privacy for both Examples

Fig. 2: Plots a and b show all the value functions that are used to compute and validate the cost of privacy for Examples 1 and 2. Plot c shows the cost of privacy itself for both examples.

established a concentration bound on the output of the Dirichlet mechanism based on its scaling parameter k. We used the concentration bound to bound the cost of privacy imposed by privatizing the transition probabilities. We further showed that the cost of privacy can be computed efficiently by establishing that the computational complexity of the algorithm is polynomial in problem parameters. Finally, the simulation results validated the developments in both the soundness of the expressions we introduced for the cost of privacy and the computational complexity associated with computing them.

REFERENCES

[1] D. J. Glancy, "Privacy in autonomous vehicles," *Santa Clara L. Rev.*, vol. 52, p. 1171, 2012.

[2] Z. Guan, G. Si, X. Zhang, L. Wu, N. Guizani, X. Du, and Y. Ma, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 82–88, 2018.

[3] S. Brechtel, T. Gindele, and R. Dillmann, "Probabilistic mdp-behavior planning for cars," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1537–1542.

[4] S. Misra, A. Mondal, S. Banik, M. Khatua, S. Bera, and M. S. Obaidat, "Residential energy management in smart grid: A markov decision process-based approach," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of things and IEEE Cyber, Physical and Social Computing*, pp. 1152–1157.

[5] D. M. Szymanski, S. G. Bharadwaj, and P. R. Varadarajan, "An analysis of the market share-profitability relationship," *Journal of marketing*, vol. 57, no. 3, pp. 1–18, 1993.

[6] J. E. Prescott, A. K. Kohli, and N. Venkatraman, "The market share-profitability relationship: An empirical assessment of major assertions and contradictions," *Strategic Management Journal*, vol. 7, no. 4, pp. 377–394, 1986.

[7] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*. Springer, 2006, pp. 265–284.

[8] F. K. Dankar and K. El Emam, "The application of differential privacy to health data," in *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, 2012, pp. 158–166.

[9] S.-S. Ho and S. Ruan, "Differential privacy for location pattern mining," in *Proceedings of the 4th ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS*, 2011, pp. 17–24.

[10] J. Cortés, G. E. Dullerud, S. Han, J. Le Ny, S. Mitra, and G. J. Pappas, "Differential privacy in control and network systems," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 4252–4272.

[11] S. Han and G. J. Pappas, "Privacy in control and dynamical systems," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 309–332, 2018.

[12] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[13] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[14] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*. IEEE, 2007, pp. 94–103.

[15] P. Gohari, B. Wu, M. Hale, and U. Topcu, "The dirichlet mechanism for differential privacy on the unit simplex," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1253–1258.

[16] M. Zhang, J. Chen, L. Yang, and J. Zhang, "Dynamic pricing for privacy-preserving mobile crowdsensing: A reinforcement learning approach," *IEEE Network*, vol. 33, no. 2, pp. 160–165, 2019.

[17] P. Venkitasubramaniam, "Privacy in stochastic control: A markov decision process perspective," in *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2013, pp. 381–388.

[18] B. Wang and N. Hegde, "Privacy-preserving q-learning with functional noise in continuous spaces," in *Advances in Neural Information Processing Systems*, 2019, pp. 11 323–11 333.

[19] A. Nilim and L. El Ghaoui, "Robust markov decision processes with uncertain transition matrices," Ph.D. dissertation, University of California, Berkeley, 2004.

[20] G. N. Iyengar, "Robust dynamic programming," *Mathematics of Operations Research*, vol. 30, no. 2, pp. 257–280, 2005.

[21] H. Xu and S. Mannor, "Distributionally robust markov decision processes," in *Advances in Neural Information Processing Systems*, 2010, pp. 2505–2513.

[22] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[23] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber, "Privacy: Theory meets practice on the map," in *2008 IEEE 24th international conference on data engineering*, pp. 277–286.

[24] L. Kong, M. K. Khan, F. Wu, G. Chen, and P. Zeng, "Millimeter-wave wireless communications for iot-cloud supported autonomous vehicles: Overview, design, and challenges," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 62–68, 2017.

[25] Z. Li, C. Chen, and K. Wang, "Cloud computing for agent-based urban transportation systems," *IEEE Intelligent Systems*, vol. 26, no. 1, pp. 73–79, 2011.

[26] J. Liu, C. Zhang, and Y. Fang, "Epic: A differential privacy framework to defend smart homes against internet traffic analysis," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1206–1217, 2018.

[27] P. Gohari, M. Hale, and U. Topcu, "Privacy-preserving policy synthesis in markov decision processes," *arXiv preprint arXiv:2004.07778*.

[28] N. Karmarkar, "A new polynomial-time algorithm for linear programming," in *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, 1984, pp. 302–311.

[29] O. Marchal, J. Arbel *et al.*, "On the sub-gaussianity of the beta and dirichlet distributions," *Electronic Communications in Probability*, vol. 22, 2017.