Exploiting Partial Observability for Optimal Deception

Mustafa O. Karabag[1], Melkior Ornik[2], and Ufuk Topcu[3]

***Abstract*— Deception is a useful tool in situations where an agent operates in the presence of its adversaries. We consider a setting where a supervisor provides a reference policy to an agent, expects the agent to operate in an environment by following the reference policy, and partially observes the agent's behavior. The agent instead follows a different, deceptive policy to achieve a different task. We model the environment with a Markov decision process and study the synthesis of optimal deceptive policies under partial observability. We formalize the notion of deception as a hypothesis testing problem and show that the synthesis of optimal deceptive policies is NP-hard. As an approximation, we consider the class of mixture policies, which provides a convex optimization formulation of the deception problem. We give an algorithm that converges to the optimal mixture policy. We also consider a special class of Markov decision processes where the transition and observation functions are deterministic. For this case, we give a randomized algorithm for path planning that generates a path for the agent in polynomial time and achieves the optimal value for the considered objective function.**

***Index Terms*— Markov decision processes, deception under partial observations, computational complexity**

I. Introduction

Deception naturally emerges in adversarial environments where an agent is performing a task that is undesirable for others. Deception is present, for example, in cyber-physical systems [1], [2], physical and information warfare [3], [4], and robotics [5]. We consider a deception problem in a setting consisting of a supervisor and an agent. In this framework, the supervisor provides a reference policy to the agent, expects the agent to perform a task by following the reference policy, and partially observes the agent's behavior. The agent, on the other hand, aims to perform a different task. For this purpose, the agent follows a different, deceptive policy.

The goal of the supervisor is to distinguish a deceptive agent from a well-intentioned agent, i.e., decide whether the agent followed the reference policy. The supervisor receives partial observations of the agent's state for detection. The goal of the deceptive agent is to perform its task while not being detected. For this reason, the agent's deceptive policy should achieve its task and generate observations, which are indistinguishable from those of the reference policy.

In this setting, we model the environment with a Markov decision process (MDP), and the agent's task as a reachability specification. The supervisor receives partial observations of the agent's state via an observation function. The agent, on the other hand, has full observability of its own state and knows the observation function of the supervisor. Given the MDP and the observation function, the agent's policy induces a hidden Markov model (HMM). The supervisor receives observation sequences from this HMM and uses them to decide whether the agent followed the reference policy.

We use Kullback-Leibler (KL) divergence to measure the deceptiveness. In detail, we use the KL divergence between the distribution of observation sequences under the agent's policy and the distribution of observation sequences under the reference policy. The value of the KL divergence is the expectation of the log-likelihood ratio between the HMM generated by the agent's policy and the HMM generated by the reference policy for a random observation sequence. The agent's problem is to find a policy that would minimize the KL divergence, making, in that sense, the two HMMs indistinguishable.

The minimization of KL divergence between two HMMs is a computationally challenging task. When the observation function is a one-to-one mapping, i.e., HMM is a Markov chain, this problem can be reduced to a convex optimization problem and solved in polynomial time [6]. The agent's partial observability provides greater opportunities for deception because the optimal value for the KL divergence objective function for the partially observable setting is lower than the fully observable case. However, exploiting the partial observability is computationally challenging. We show that the 3-SAT problem [7] can be reduced to an instance of the deception problem in the partially observable setting. Consequently, the agent's problem is NP-hard. Furthermore, we show that there is no polynomial time approximation scheme for it unless P = NP.

The computational hardness of the agent's problem is due to the large size of the policy space, the large number of observation sequences, and the stochasticity of the MDP or observation function. One can synthesize an optimal deceptive policy by solving a convex optimization problem that considers the class of history-dependent policies. However, this optimization problem would have exponentially many variables in the length of the time horizon.

We consider a smaller policy space as an approximation to the agent's problem. A mixture policy [8] is a weighted set of basis policies. We use mixture policies as the search space for the agent's problem. Since the KL objective function is a convex function of the weight vector, one can find the best mixture of any given set of policies by solving a convex optimization problem. On the other hand, the construction of the optimization problem still requires a parameter for each observation sequence. Since the number of observation sequences is potentially large, the full construction of the optimization problem is impractical. Instead, we propose to use stochastic optimization to solve this problem. We give an iterative algorithm that asymptotically converges to the optimal value and outputs a near-optimal mixture of a given set of policies. The advantage of the algorithm is that the full construction is not required and every iteration takes polynomial time in the size of the problem.

When the transition and observation functions are deterministic, one can synthesize the optimal policy by directly optimizing the probabilities of the observation sequences. However, synthesizing an explicit policy is generally infeasible since the number of observation sequences is large. Instead of synthesizing an explicit policy, we propose a randomized algorithm that generates a single path. The algorithm boosts the probabilities of the observation sequences for which there is a path that satisfies the agent's task. The algorithm induces the optimal distribution of observation sequences and generates a path for the agent in polynomial time.

*This work was supported in part by ONR N00014-21-1-2502, DARPA D19AP00004, and AFRL FA9550-19-1-0169.

[1]M. O. Karabag is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78705 USA (e-mail:karabag@utexas.edu).

[2]M. Ornik is with the Department of Aerospace Engineering and the Coordinated Science Laboratory, The University of Illinois Urbana-Champaign, Urbana, IL 61801, USA (e-mail:mornik@illinois.edu).

[3]U. Topcu is with the Department of Aerospace Engineering and Engineering Mechanics and the Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78705 USA (e-mail:utopcu@utexas.edu).

Related Work: Deception has been studied in the game theory framework, e.g., [2], [9], [10], as a game between deceiving and deceived players where there is an information asymmetry or capability difference between the players. Different from existing works that consider static games or state-dependent utility functions, we consider a sequential setting with an observation sequence-dependent utility function motivated by hypothesis testing.

Paper [6] studies the synthesis of deceptive policies when the supervisor receives full observations of the agent's state. In this case, optimal policies can be synthesized in polynomial time via a convex optimization problem. Different from [6], we consider that the supervisor has partial observations of the agent's state.

The concept of opacity [11]–[13] is closely related to the notion of deception: Hiding properties of the system from an outside observer. Probabilistic system opacity [12] is the problem of determining the source of an observation sequence given a set of HMMs. Two HMMs are pairwise probabilistically opaque if the misclassification rate is a positive constant for any observation sequence. Under strong assumptions, e.g., HMMs can start from any initial state with nonzero probability, [12] shows that probabilistic opacity can be verified in polynomial time. In the course of our paper, we show that, when the initial state distribution is not strictly positive, the verification is NP-hard. We also consider the optimization problem of finding an HMM that is closest to a target HMM which is not studied in [12].

Partially observable MDPs (POMDPs) are commonly used to model the environment of an agent with partial observability of its state. While there are existing results on the hardness of policy synthesis for POMDPs [14]–[16], these results do not apply to the problem studied in this paper since we consider partial observability for an outside observer, i.e., the supervisor, and not for the agent. For example, planning in deterministic POMDPs [16] is provably hard due to the initial state ambiguity whereas we provide an efficient algorithm for this case thanks to the full observability of the agent.

Decision problems for regular languages [17]–[19] are closely related to the deception problem due to the objective function that we consider. In the course of our paper, we show that the language containment problem [18] is equivalent to deciding the finiteness of the KL divergence. We use the results from automata theory to establish the computational hardness of the deception problem. In addition to the qualitative analysis, we quantitatively optimize the closeness of two languages using KL divergence.

II. Preliminaries

Set $\left\{x = (x_1, \ldots, x_n) | \sum_{i=1}^{n} x_i = 1, x_i \geq b\right\}$ is denoted by Δ_b^n. $|C|$ denotes the size of set C. The power set of C is denoted by 2^C. $Proj_W(x)$ is the L_2 projection of x onto W. $Ber(p)$ is the distribution of a Bernoulli random variable with parameter p. Set $\{1, \ldots, n\}$ is denoted by $[n]$.

Let Q_1 and Q_2 be discrete probability distributions with a support $\mathcal{X}$. The *Kullback–Leibler (KL) divergence* between Q_1 and Q_2 is

$$KL(Q_1||Q_2) = \sum_{x \in \mathcal{X}} Q_1(x) \log \left(\frac{Q_1(x)}{Q_2(x)} \right).$$

We define $Q_1(x) \log \left(\frac{Q_1(x)}{Q_2(x)} \right)$ to be 0 if $Q_1(x) = 0$, and ∞ if $Q_1(x) > 0$ and $Q_2(x) = 0$. KL divergence is a jointly convex function in its arguments.

Let $p(y|x)$ be a conditional probability mass function. Let W_1 and W_2 be discrete probability distributions with a support $\mathcal{Y}$ such that for every $y \in \mathcal{Y}$, $W_1(y) = \sum_{x \in \mathcal{X}} Q_1(x) p(y|x)$ and $W_2(y) = \sum_{x \in \mathcal{X}} Q_2(x) p(y|x)$. *Data processing inequality* states that any (potentially) stochastic transformation $p(x|y)$ satisfies

$$KL(Q_1||Q_2) \geq KL(W_1||W_2). \tag{1}$$

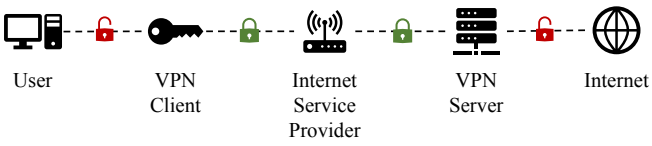


Fig. 1: Internet access via a virtual private network (VPN). VPN client encrypts the user data, and the internet service provider (ISP) cannot observe the user's traffic.

A. Markov Decision Processes

A *Markov decision process* (MDP) is a tuple $\mathcal{M} = (S, A, P, s_0)$ where S is a finite set of states, A is a finite set of actions, $P : S \times A \times S \to [0, 1]$ is the transition probability function, and s_0 is the initial state. $A(s)$ denotes the set of available actions at state s where $\sum_{q \in S} P(s, a, q) = 1$ for all $a \in A(s)$. State s is *absorbing* if $P(s, a, s) = 1$ for all $a \in A(s)$.

The *history* h_t at time t is a sequence of states and actions such that $h_t = s_0 a_0 s_1 \ldots s_{t-1} a_{t-1} s_t$. The set of all possible histories at time t is $\mathcal{H}_t$. A *path* $\xi = s_0 s_1 \ldots$ is an infinite sequence of states. The set of paths is $Paths(\mathcal{M})$. A *(history-dependent) policy* for $\mathcal{M}$ is a sequence $\pi = \mu_0 \mu_1 \ldots$ where each $\mu_t : \mathcal{H}_t \times A \to [0, 1]$ is a function such that $\sum_{a \in A(s_t)} \mu_t(h_t, a) = 1$ for all $h_t \in \mathcal{H}_t$. The set of all policies for $\mathcal{M}$ is denoted by $\Pi(\mathcal{M})$. A *Markovian policy* is a sequence $\pi = \mu_1 \mu_2 \ldots$ where $\mu_t : S \times A \to [0, 1]$ is a function such that $\sum_{a \in A(s)} \mu_t(s, a) = 1$ for every $s \in S$ and $t \geq 1$. A *stationary policy* is a sequence $\pi = \mu\mu \ldots$ where $\mu : S \times A \to [0, 1]$ is a function such that $\sum_{a \in A(s)} \mu(s, a) = 1$ for every $s \in S$. For notational simplicity, we use $\pi(s, a)$ for $\mu(s, a)$ if π is stationary. A *deterministic* policy is a sequence $\pi = \mu_0 \mu_1 \ldots$ such that $\mu_t(\cdot, a) = 0$ or 1 where $\cdot$ is a state or a history. We use $\Pi^{D,H}(\mathcal{M})$ to denote the set of deterministic, history-dependent policies.

The event of reaching set R is denoted with $\Diamond R$. A path $\xi = s_0 s_1 \ldots$ satisfies $\Diamond R$, i.e., $\xi \models \Diamond R$, if and only if $s_i \in R$ for some $i \geq 0$. The event of reaching set R in T steps is denoted with $\Diamond_{\leq T} R$. A path $\xi = s_0 s_1 \ldots$ satisfies $\Diamond_{\leq T} R$, i.e., $\xi \models \Diamond_{\leq T} R$, if and only if $s_i \in R$ for some $0 \leq i \leq T$.

A *nondeterministic finite automaton* (NFA) is a tuple $N = (Q, \Sigma, \Delta, q_0, F)$ where Q is a finite set of states, Σ is a finite set of input symbols, $\Delta : Q \times \Sigma \to 2^Q$ is a transition function, q_0 is an initial state, and F is a set of accepting states such that $F \subseteq Q$.

III. Deception Under Partial Observability

We consider a setting where an *agent* operates in a stochastic environment modeled by an MDP $\mathcal{M}$. A *supervisor* provides a *reference policy* $\pi^{\mathcal{S}}$ to the agent and expects the agent to perform a task by following the reference policy. While the agent operates in the environment, the supervisor receives a *partial observation* of the agent's state at every time step. The agent's goal is to perform a different task, that is, to reach a set $R^{\mathcal{A}}$ with a probability of at least ν. For this purpose, the agent may deviate from the reference policy and follow another policy $\pi^{\mathcal{A}}$. The agent's *deceptive policy* should accomplish the agent's task with high probability and generate an observation sequence that minimizes the chance of being detected.

The supervisor observes the agent's state via an *observation function* $O : S \times \Omega \to [0, 1]$ where Ω is a finite set of observations and $\sum_{o \in \Omega} O(s, o) = 1$ for all $s \in S$. The agent has full observability of its state and knows the observation function of the supervisor. For full generality, we assume that the agent does not know the observations received by the supervisor, because in the case when the agent knows the observations received by the supervisor, we can add auxiliary states to represent the observations received by the supervisor.

Example. We consider the virtual private network (VPN) example given in Figure 1 to demonstrate the effects of partial observability. If a user accesses the internet without a VPN, then the internet service provider (ISP) can observe the user's unencrypted traffic. In this case, the ISP can detect users with undesirable traffic. If the user accesses the internet via a VPN client, then ISP observes the user's encrypted data. The encryption makes the user's traffic partially observable; encrypted data for different types of traffic looks effectively the same for the ISP. When a VPN is used, i.e., when partial observability is exploited, ISP cannot distinguish the users with undesirable traffic.

We remark that the setting we consider is different from partially observable MDPs (POMDPs). In POMDPs, the agent has partial observability of its state, and the goal is to find a policy that uses observations whereas in our setting the agent has full observability of its state and the goal is to shape the observation sequence.

A policy induces probability measures over paths and observation sequences. With an abuse of notation, we denote the probability measures induced by policy π with $\Pr^{\pi}$. For simplicity, we use $\Pr^{\mathcal{S}}$ and $\Pr^{\mathcal{A}}$ for $\pi^{\mathcal{S}}$ and $\pi^{\mathcal{A}}$, respectively. We assume that $R^{\mathcal{A}}$ is a set of absorbing states, and the reference policy eventually reaches an absorbing state, i.e., $\Pr^{\mathcal{S}}(\Diamond_{\leq T} S^{end}) = 1$ for T-step finite horizon, and $\Pr^{\mathcal{S}}(\Diamond S^{end}) = 1$ for infinite horizon where S^{end} is the set of all absorbing states. All absorbing states share a unique observation ε indicating an absorbing state has been reached. Formally, $O(s,\varepsilon) = 1$ for all $s \in S^{end}$, and $O(s,\varepsilon) = 0$ for all $s \in S \setminus S^{end}$. When $R^{\mathcal{A}}$ is not a set of absorbing state one can use a finite automaton to represent the event of reaching $R^{\mathcal{A}}$ and synthesize the policy in the product MDP of the automaton and the original MDP.

We propose the following problems for the synthesis of optimal deceptive policies under partial observability in finite and infinite horizon settings. We use KL divergence as a proxy for the closeness of observations induced by the agent's policy and the reference policy. We note that KL divergences in Problems 1 and 2 are over the distributions of observation sequences.

Problem 1 (Finite Horizon)**.** *Given a Markovian reference policy $\pi^{\mathcal{S}}$, solve*

$$\min_{\pi^{\mathcal{A}} \in \Pi(\mathcal{M})} \quad KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) \tag{2a}$$

$$\textit{subject to} \quad \Pr^{\mathcal{A}}\left(\Diamond_{\leq T} R^{\mathcal{A}}\right) \geq \nu \tag{2b}$$

where $\Theta^{\mathcal{A}}_{0:T}$ and $\Theta^{\mathcal{S}}_{0:T}$ are the probability distributions of $(T+1)-$length observation sequences under $\pi^{\mathcal{A}}$ and $\pi^{\mathcal{S}}$, respectively.

Problem 2 (Infinite Horizon)**.** *Given a stationary reference policy $\pi^{\mathcal{S}}$, solve*

$$\min_{\pi^{\mathcal{A}} \in \Pi(\mathcal{M})} \quad KL\left(\Theta^{\mathcal{A}} || \Theta^{\mathcal{S}}\right) \tag{3a}$$

$$\textit{subject to} \quad \Pr^{\mathcal{A}}\left(\Diamond R^{\mathcal{A}}\right) \geq \nu \tag{3b}$$

where $\Theta^{\mathcal{A}}$ and $\Theta^{\mathcal{S}}$ are the probability distributions of infinite length observation sequences under $\pi^{\mathcal{A}}$ and $\pi^{\mathcal{S}}$, respectively.

The minimization of the KL objective function is a proxy for minimizing the chance of being detected when the supervisor uses the observation sequences with likelihood-ratio test [20], which is the most powerful test for a given significance level. We refer the interested readers to [6] on the relationship between the KL divergence and likelihood-ratio test. Suppose the supervisor aims to decide whether the agent followed $\pi^{\mathcal{A}}$ or $\pi^{\mathcal{S}}$ given an observation sequence θ using the likelihood-ratio test. The KL divergence between the distributions $\Theta^{\mathcal{A}}$ and $\Theta^{\mathcal{S}}$ is the expectation of the log-likelihood difference between the hypotheses $\pi^{\mathcal{A}}$ and $\pi^{\mathcal{S}}$. Hence, minimizing the KL objective function increases the false negative rate, i.e., the probability of the supervisor believing that the agent followed $\pi^{\mathcal{S}}$ whereas the agent followed $\pi^{\mathcal{A}}$. If the KL divergence is zero, then the distribution $\Theta^{\mathcal{A}}$ of observation sequences under $\pi^{\mathcal{A}}$ is equal to the distribution $\Theta^{\mathcal{S}}$ under $\pi^{\mathcal{S}}$, i.e., the maximum of false positive and negative rates is greater than or equal to $1/2$ for any statistical test. If the objective function is infinite, then with a positive probability the supervisor is almost sure that the agent did not follow $\pi^{\mathcal{S}}$.

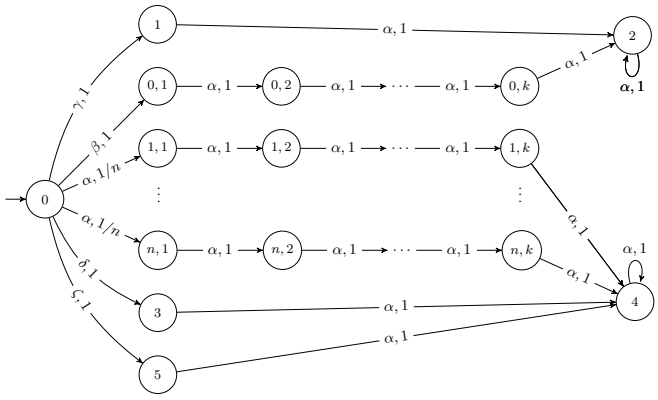


Fig. 2: MDP for the proof of Proposition 1 where nodes are the states. A label a, p of an edge between nodes s and q refers to the transition that happens with probability p under action a, i.e., $P(s,a,q) = p$.

Remark. *In our problem setting, the supervisor knows the behavioral model, i.e., reference policy, of the well-intentioned agents. If the supervisor does not know the behavioral model, it can first infer a model, e.g., as in [21] using an n-Gram model, and perform detection using the learned model. We also consider that the supervisor receives an observation sequence for detection. Instead of using the sequences directly, one may consider using some features of the sequences, such as symbol frequencies or every other symbol. However, by the data processing inequality* (1)*, using features can only lower the KL divergence, thereby worsening the detection rate. Overall, we consider a setting that is the worst-case scenario for the deceptive agent since the supervisor knows the model of the well-intentioned agents and uses complete observation sequences.*

IV. The Complexity of Optimal Deception Under Partial Observability

In this section, we discuss the complexity of optimal deception under partial observability. Under full observability, i.e., there is a one-to-one mapping between states and observations, the synthesis of optimal deceptive policies can be achieved in polynomial time by solving a convex optimization problem [6]. It is easy to see that, by (1), the optimal values of (2) and (3) under partial observability are upper-bounded by the optimal values of (2) and (3) under full observability, respectively. This intuitively implies that the chance of being detected is lower under partial observability.

While partial observability provides a better opportunity for deception, i.e., lower objective values, and the agent still has full observability of its own state, exploiting partial observability is a hard problem. We can synthesize optimal deceptive policies under partial observability by solving a convex optimization problem with exponentially many parameters in the time horizon. The exponential complexity is due to the number of possible histories and observation sequences. Proposition 1 shows that Problem 1 is a provably hard problem, and there is no polynomial-time algorithm unless $P = NP$. The proof is due to a reduction from the 3-SAT problem [7].

Remark. *Determining whether an observation sequence is possible for an HMM is equivalent to determining whether a word is accepted by a NFA. Formally, for a stationary policy π and a set C of end states, we can construct a NFA $N = (Q, \Sigma, \Delta, q_0, F)$ such that a word θ is accepted by N if an only if there exists a path ξ for MDP $\mathcal{M}$ that reach C satisfying $\Pr^{\pi}(\theta|\xi)$. NFA N is constructed such that $Q = S$, $q_0 = s_0$, $\Sigma = \Omega$, $F = C$, and $s' \in \Delta(s, o)$ if and only if $o \in O(s)$ and $\sum_{a \in A(s)} \pi(s, a)P(s, a, s') > 0$.*

Proposition 1. *Let v^* be the optimal value of* (2)*. Deciding whether $v^* = \infty$ is NP-hard. If P $\neq$ NP, there is no polynomial-time approximation scheme for* (2) *that guarantees a value lower than or equal to $(v^* + \epsilon)$ or $(1 + \epsilon)v^*$.*

Proof of Proposition 1. We use the MDP given in Fig. 2 to prove the hardness of (2). This MDP shares a similar structure with the NFA used to prove the hardness of language universality and containment problems for NFAs [22], [23]. The task of the agent is to reach state 2 with probability 1, i.e., $R^{\mathcal{A}} = \{2\}$ and $\nu = 1$. The observation function is defined using a 3-SAT formula. A 3-SAT formula [7] is a conjunctive normal formula with n clauses where each clause has three literals from a set $l_1, \ldots, l_k, \neg l_1, \ldots, \neg l_k$ of $2k$ literals. Let ϕ be an arbitrary instance of 3-SAT and $T = k + 3$. The observation function $O : S \times \Omega \to [0, 1]$ is defined such that

- $O(0, x) = 1$, $O(1, y) = 1$, $O(2, \varepsilon) = 1$, $O(3, y) = 1$, $O(4, \varepsilon) = 1$, $O(5, z) = 1$,
- $O((0, j), \top) = 0.5$ and $O((0, j), \bot) = 0.5$,
- $O((i, j), \top) = 0.5$ and $O((i, j), \bot) = 0.5$ if i-th clause of ϕ does not contain l_j and $\neg l_j$,
- $O((i, j), \top) = 1$ if i-th clause of ϕ contains $\neg l_j$, and
- $O((i, j), \bot) = 1$ if i-th clause of ϕ contains l_j.

To show that deciding whether the optimal value of (2) is ∞ is NP-hard, consider a reference policy such that $\pi^{\mathcal{S}}(0, \alpha) = 1$. Note that the decision at state 0 is sufficient to describe the policy since there is only one action for the other states. If $\pi^{\mathcal{A}}(0, \beta) \neq 1$ the agent violates the task constraint, i.e., $\Pr^{\mathcal{A}}\left(\Diamond_{\leq T} R^{\mathcal{A}}\right) \geq \nu$, or $KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) = \infty$ since there is a positive probability that the agent generates an observation sequence θ such that $\Pr^{\mathcal{S}}(\theta) = 0$.

If $\pi^{\mathcal{A}}(0, \beta) = 1$, we have $KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) < \infty$ if and only if $\Pr^{\mathcal{S}}(\theta) > 0$ for all $\theta \in \{x\{\top, \bot\}^k\}$ since $\Pr^{\mathcal{A}}(\theta) > 0$ for all $\theta \in \{x\{\top, \bot\}^k\}$. Note that by construction of the observation function, $\Pr^{\mathcal{S}}(\theta = o_1 \ldots o_{k+1}) > 0$ if and only if $o_2 \ldots o_{k+1}$ is an assignment for $l_1 \ldots l_k$ that satisfies $\neg\phi$. Consequently, $\Pr^{\mathcal{S}}(\theta) > 0$ for all $\theta \in \{x\{\top, \bot\}^k\}$ if and only if $\neg\phi$ is true for all $\theta \in \{x\{\top, \bot\}^k\}$. Hence, $\Pr^{\mathcal{S}}(\theta) > 0$ for all $\theta \in \{x\{\top, \bot\}^k\}$ if and only if ϕ is not satisfiable. Since 3-SAT problem is NP-hard [7], and the size of the MDP is polynomial in the size of the 3-SAT instance, deciding whether the optimal value of (2) is ∞ is NP-hard.

To show that there is no polynomial-time ϵ-approximation scheme for (2) unless P $\neq$ NP, we consider a reference policy such that $\pi^{\mathcal{S}}(0, \alpha) = 0.5$, $\pi^{\mathcal{S}}(0, \delta) = b$, and $\pi^{\mathcal{S}}(0, \zeta) = 0.5 - b$. If ϕ is not satisfiable, the optimal value of (2) is $\log(1/b)$, which is achieved when $\pi^{\mathcal{A}}(0, \gamma) = 1$. If ϕ is satisfiable, every $\theta \in \{x\{\top, \bot\}^k\}$ has $\Pr^{\mathcal{S}}(\theta) \geq 2^{-k+2}/n$ by construction of the observation function. If ϕ is satisfiable and $\pi^{\mathcal{A}}(0, \beta) = 1$, we have

$$KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) = \sum_{\theta \in \{x\{\top,\bot\}^k\}} \Pr^{\mathcal{A}}(\theta) \log\left(\frac{\Pr^{\mathcal{A}}(\theta)}{\Pr^{\mathcal{S}}(\theta)}\right)$$
$$= \sum_{\theta \in \{x\{\top,\bot\}^k\}} 2^{-k} \log\left(\frac{2^{-k}}{\Pr^{\mathcal{S}}(\theta)}\right) \leq \log(n/4).$$

Hence, the optimal value of (2) is lower than or equal to $\log(n/4)$ if ϕ is satisfiable. Let $1/b < n/4$. If an approximation scheme assigns $\pi^{\mathcal{A}}(0, \beta) > 0$ and ϕ is not satisfiable, then $KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) - \log(1/b) = \infty$. If an approximation scheme assigns $\pi^{\mathcal{A}}(0, \beta) = 0$ and ϕ is satisfiable, then $KL\left(\Theta^{\mathcal{A}}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) - \log(n/4)$ is a constant not depending on the input parameter ϵ of the approximation algorithm. Therefore, any approximation algorithm has to solve the 3-SAT problem to achieve ϵ-optimality, and there is no polynomial-time ϵ-approximation scheme for (2) unless P $\neq$ NP. ∎

Proposition 1 also applies to the infinite horizon optimization problem given in (3) as the reference policy in the proof is stationary.

We remark that the paper [12] showed that for two hidden Markov models (HMMs) deciding whether the likelihood-ratio of any observation sequence converges to a positive number is possible in polynomial time assuming that both HMMs start from any initial state with a nonzero probability, i.e., the initial state distribution is strictly positive. The proof of Proposition 1 shows that when the probability distribution of the initial state is not strictly positive, there is no polynomial-time algorithm for this problem unless P = NP.

We also remark that optimal deception under partial observability is a hard problem even for the simplest observation functions. For example, consider an observation function such that all transient states emit the same observation with probability 1 and all absorbing states emit another observation with probability 1. The deciding whether the optimal value of (3) is ∞ correspond to the language containment problem of unary NFA, which is shown to be coNP-complete [19].

V. Synthesis of Deceptive Policies

In this section we discuss the synthesis of deceptive policies under partial observability. In detail, we consider the synthesis for finite horizon using mixture policies as an alternative to optimal policies. We also consider a special class of MDPs where optimal distribution of paths can be induced in polynomial time for infinite horizon.

A. Mixture Policies for Finite Horizon

We consider a special class of policies for the finite horizon case since the synthesis of optimal deceptive policies is computationally challenging due to the size of history-dependent policies. A *mixture policy* [8] is a convex combination of a finite set of policies. In detail, a mixture policy is a tuple $([\pi^1, \ldots, \pi^N], [\alpha_1, \ldots, \alpha_N])$ where $[\pi^1, \ldots, \pi^N]$ is a vector of basis policies and $[\alpha_1, \ldots, \alpha_N] \in \Delta_0^N$ is mixing probabilities. At time 0, the agent chooses policy π^i with probability α_i and follows the selected policy for the whole path.

The class of mixture policies has the following useful property: The probability distribution over paths (and over observation sequences) induced by the mixture policy is a linear combination of the probability distribution induced by each basis policy. This property provides a convex representation of the deception problem. The KL divergence between the distributions of observation sequences is a nonconvex funtion of the parameters of policies $\pi^1, \ldots, \pi^N$. On the other hand, the KL objective function is a convex function of the mixing probabilities $\alpha_1, \ldots, \alpha_N$. Hence, for a given set of basis policies $\pi^1, \ldots, \pi^N$, our goal is to optimize a convex function of the mixing probabilities $\alpha_1, \ldots, \alpha_N$ and find the best mixture policy.

The straightforward approach to find the optimal mixing probabilities is the following. First, enumerate the possible observation sequences under the reference policy for the given time horizon. Second, find the probabilities of the observation sequences under the basis policies. Finally, optimize the KL objective function. However, the enumeration of possible observation sequences is a challenging problem. Counting the possible observation sequences

Algorithm 1: Mixing algorithm

Input: A set $C^{(0)}$ of basis policies.
$\alpha^{(0,1)} \leftarrow [1/|C^{(0)}|, \ldots, 1/|C^{(0)}|]$. ▷ Initial uniform mixing
$\beta^{(0)} \leftarrow 1$, $b^{(0)} \leftarrow 1/(2|C^{(0)}|)$, $N^{(0)} \leftarrow 1$. ▷ Opt. parameters
for $k = 1, \ldots$ **do**
  $C^{(k)} \leftarrow C^{(k-1)}$, $\alpha^{(k,0)} \leftarrow \alpha^{(k-1,N^{(k-1)})}$.
  $\beta^{(k)} \leftarrow \beta^{(k-1)}/2$, $b^{(k)} \leftarrow b^{(k-1)}/\sqrt{2}$, $N^{(k)} \leftarrow 4N^{(k-1)}$.
  for $i = 1, \ldots, N^{(k)}$ **do**
    Uniformly sample an observation sequence θ from Ω^T.
    if $\Pr^{\mathcal{S}}(\theta) = 0$ **then**
      for $\pi \in C^{(k)}$ *such that* $\Pr^{\pi}(\theta) \neq 0$ **do**
        $C^{(k)} \leftarrow C^{(k)} \setminus \{\pi\}$.
        Remove the mixing probability that correspond to π from $\alpha^{(k,i-1)}$ and normalize $\alpha^{(k,i-1)}$.
      $f(\alpha^{(k,i-1)}, \theta) \leftarrow 0$ ▷ No gradient step if $\Pr^{\pi}(\theta) = 0$
    else
      $f(\alpha^{(k,i-1)}, \theta) \leftarrow \Pr^{(C^{(k)}, \alpha^{(k,i-1)})}(\theta) \log\left(\frac{\Pr^{(C^{(k)}, \alpha^{(k,i-1)})}(\theta)}{\Pr^{\mathcal{S}}(\theta)}\right)$
    $TC = \{\alpha | \Pr^{(C^{(k)}, \alpha)}(\Diamond_{\leq T} R^{\mathcal{A}}) \geq \nu\}$ ▷ Task constraint
    $\alpha^{(k,i)} \leftarrow \alpha^{(k,i-1)} - \beta^{(k)} \nabla f_{\alpha^{(k,i-1)}}(\alpha^{(k,i-1)}, \theta)$
    $\alpha^{(k,i)} \leftarrow Proj_{\Delta_{b^{(k)}} \cap TC}(\alpha^{(k,i)})$
  $\alpha^{(k)} \leftarrow \sum_{i=1}^{N^{(k)}} \alpha^{(k,i)}/N^{(k)}$ ▷ Mixing vector after itr. k

under the reference policy is ♯P-complete due to a reduction from the problem of counting the number of satisfying assignments to a Boolean formula [24]. Hence, even the construction of the problem is computationally hard.

To avoid the complete construction, we propose to use stochastic optimization for the synthesis of optimal mixture policies. Algorithm 1 uses projected stochastic gradient descent method. It samples an observation sequence uniformly randomly and adjusts the mixing probabilities accordingly. If the likelihood of an observation sequence is positive under a basis policy and 0 under the reference policy, then the basis policy is removed. Since computing the probability of an observation sequence for an HMM can be performed in polynomial time, every iteration in the inner **for** loop takes polynomial time in the size of the basis policies. We have

$$\frac{\partial f(\alpha^{(k,i-1)})}{\partial \alpha_j^{(k,i-1)}} = \Pr^{\pi^j}(\theta)\left(\log\left(\frac{\Pr^{(C^{(k)}, \alpha^{(k,i-1)})}(\theta)}{\Pr^{\mathcal{S}}(\theta)}\right) + 1\right).$$

The output of Algorithm 1 almost surely asymptotically converges to a set of optimal mixture parameters.

Proposition 2. *Let α^* be a set of optimal mixing probabilities for the set $C^{(0)}$ of basis policies. Assume that there exists $\pi^i \in C^{(0)}$ with $KL\left(\Theta_{0:T}^{\pi^i} || \Theta_{0:T}^{\mathcal{S}}\right) < \infty$. In Algorithm 1, with probability* 1,

$$\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] = KL\left(\Theta_{0:T}^{(C^{(0)}, \alpha^*)} || \Theta_{0:T}^{\mathcal{S}}\right).$$

Let v^* be the optimal value of (2)*. If $C^{(0)} = \Pi^{D,H}(\mathcal{M})$ in Algorithm 1, with probability* 1,

$$\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] = v^*.$$

Proof of Proposition 2. We first show that with probability 1, $\lim_{k\to\infty} KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right) < \infty$.

Note that $KL\left(\Theta_{0:T}^{\pi^i} || \Theta_{0:T}^{\mathcal{S}}\right) < \infty$ if and only if $\Pr^{\pi^i}(\theta) = 0$ for all $\theta \in \Omega^T$ such that $\Pr^{\mathcal{S}}(\theta) = 0$. Let $\pi^i \in C^{(k)}$ be a policy such that $\Pr^{\mathcal{S}}(\theta) = 0$ and $\Pr^{\pi^i}(\theta) \neq 0$ for some $\theta \in \Omega^T$. We have $\pi^i \in C^{(k+1)}$ with probability at most $(1 - 1/|\Omega^T|)^{N^{(k)}} \leq \exp(-N^{(k)}/|\Omega^T|)$. After K rounds, we have $\pi^i \in C^{(K+1)}$ with probability at most $\exp(-4^K N^{(0)}/|\Omega^T|)$. Hence, $\pi^i \notin C^{(k)}$ with probability 1 as $k \to \infty$. By the union bound and the convexity of the KL divergence, $\lim_{k\to\infty} KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right) \leq \sum_{i=1}^{N^{(k)}} \lim_{k\to\infty} KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k,i)})} || \Theta_{0:T}^{\mathcal{S}}\right)/N^{(k)} < \infty$.

We now show that with probability 1,

$$\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] = KL\left(\Theta_{0:T}^{(C, \alpha^*)} || \Theta_{0:T}^{\mathcal{S}}\right).$$

Let $v^{(k),*} = \min_{\alpha \in \Delta_{b^{(k)}}} KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha)} || \Theta_{0:T}^{\mathcal{S}}\right)$. Assume that $KL\left(\Theta_{0:T}^{\pi} || \Theta_{0:T}^{\mathcal{S}}\right) < \infty$ for all $\pi \in C^k$. Let $M^{(k)} = \sup_{\alpha^{(k)} \in \Delta_{b^{(k)}}} \max_{\theta \in \Omega^T} \left\|\nabla f(\alpha^{(k)}, \theta)\right\|^2$. By Equation 2.19 of [25], we have

$$\mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right) - v^{(k),*}\right] \leq \frac{4 + (M^{(k)})^2 N^{(k)} (\beta^{(k)})^2}{2|\Omega^T| N^{(k)} \beta^{(k)}}.$$

Let θ be an arbitrary observation sequence, and $\theta^* = \arg\min_{\theta' \in \Omega^T} \Pr^{\mathcal{S}}(\theta')$ such that $\Pr^{\mathcal{S}}(\theta') > 0$. For large enough k, we have $\log(b^{(k)}/\Pr^{\mathcal{S}}(\theta)) \leq \partial f(\alpha^{(k)}, \theta)/\partial \alpha_j^{(k)} \leq 2/\Pr^{\mathcal{S}}(\theta)$. Similarly, for large k, we have $|\partial f(\alpha^{(k,i-1)}, \theta)\ /\partial \alpha_j^{(k)}| \leq -\log(b^{(k)}/\Pr^{\mathcal{S}}(\theta))$ since $b^{(k)} \to 0$. Hence,

$$\left\|\nabla f(\alpha^{(k)}, \theta)\right\|^2 \leq |C^{(k)}| \log\left(\frac{b^{(k)}}{\Pr^{\mathcal{S}}(\theta^*)}\right)^2$$

for all $\alpha^{(k)} \in \Delta_{b^{(k)}}$ and $\theta \in \Omega^T$ since $\alpha^{(k,i-1)}$ has $|C^{(k)}|$ elements. Define $L^{(k)} = |C^{(k)}| \log\left(\frac{b^{(k)}}{\Pr^{\mathcal{S}}(\theta^*)}\right)^2$. There exists $k' \geq 0$ such that

$$\mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right) - v^{(k),*}\right] \leq \frac{4 + (L^{(k)})^2 N^{(k)} (\beta^{(k)})^2}{2|\Omega^T| N^{(k)} \beta^{(k)}}.$$

for all $k > k'$.

Since $\lim_{k\to\infty} N^{(k)} \beta^{(k)} = \infty$, we only need to show $\lim_{k\to\infty} (L^{(k)})^2 \beta^{(k)} = 0$ in order to show that the term on the right hand side goes to zero as $k \to \infty$.

Since $b^{(k+1)} = \sqrt{2} b^{(k)}$, we have

$$\lim_{k\to\infty} \log\left(\frac{b^{(k+1)}}{\Pr^{\mathcal{S}}(\theta^*)}\right) / \log\left(\frac{b^{(k)}}{\Pr^{\mathcal{S}}(\theta^*)}\right) \leq \sqrt{3}.$$

Consequently, $\lim_{k\to\infty} L^{(k+1)}/L^{(k)} \leq \sqrt{3}$. Since $\beta^{(k+1)}/\beta^{(k)} = 1/2$, we have $\lim_{k\to\infty} (L^{(k)})^2 \beta^{(k)} = 0$, which implies $\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] - v^{(k),*} = 0$.

Since $KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)$ is a bounded, continuous function, we have $\lim_{k\to\infty} v^{(k),*} = KL\left(\Theta_{0:T}^{(C, \alpha^*)} || \Theta_{0:T}^{\mathcal{S}}\right)$. This property trivially holds when α^* is an interior point and holds due to the continuity and boundedness when α^* is a boundary point. Consequently, with probability 1,

$$\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] = KL\left(\Theta_{0:T}^{(C, \alpha^*)} || \Theta_{0:T}^{\mathcal{S}}\right).$$

We now show that if $C^{(0)} = \Pi^{D,H}(\mathcal{M})$ in Algorithm 1, then $\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)}, \alpha^{(k)})} || \Theta_{0:T}^{\mathcal{S}}\right)\right] = v^*$ with probability 1.

Theorem 3.1 of [8] shows that every final state distribution of a finite horizon MDP achieved by a Markovian, randomized policy can be achieved with a mixture of Markovian, deterministic policies. Consider an MDP $\mathcal{M}'$ whose states are possible histories from $t = 0$ to T of the MDP $\mathcal{M}$. The possible transitions between the states of $\mathcal{M}'$ are defined via the state-action histories on $\mathcal{M}$. A Markovian policy on $\mathcal{M}'$ is a history dependent policy on $\mathcal{M}$, and the final state distribution of $\mathcal{M}'$ is the distribution of histories of $\mathcal{M}$. By Theorem 3.1 of [8], the mixture of Markovian, deterministic policies achieve every final state distribution of $\mathcal{M}'$, which implies that the mixture of history dependent, deterministic policies achieve every history distribution of $\mathcal{M}$. Consequently, if $C^{(0)} = \Pi^{D,H}(\mathcal{M})$,

$$\lim_{k\to\infty} \mathbb{E}\left[KL\left(\Theta_{0:T}^{(C^{(k)},\alpha^{(k)})}||\Theta_{0:T}^{\mathcal{S}}\right)\right] = KL\left(\Theta_{0:T}^{(C^{(0)},\alpha^{*})}||\Theta_{0:T}^{\mathcal{S}}\right) = v^*$$

with probability 1. ■

We note that we use an iterative scheme to adjust the parameters of projected stochastic gradient descent. Such a scheme is needed because the Lipschitz constant of the objective function is unknown.

When the set of basis policies is the set of deterministic, history dependent policies, the output mixture policy is an optimal solution for Problem 1 in the limit since there exists a mixture of deterministic, history dependent policies that induces the same distribution of observation sequences with the optimal history-dependent policy.

Stochastic gradient descent method requires uniform sampling of the feasible observation sequences. In Algorithm 1, we sample observation sequences uniformly randomly from Ω^T for simplicity. However, some observation sequences may be infeasible under the basis policies, i.e., $\max_i \Pr^{\pi^i}(\theta) = 0$. Algorithm 1 relies on rejection sampling and ignores such observation sequences. The convergence of Algorithm 1 might be slow in practice due to rejection sampling and the large size of Ω^T. Direct sampling from the set $\{\theta|\max_i \Pr^{\pi^i}(\theta) > 0\}$ in polynomial time is possible using the method given in [26] since this set defines a regular language. Another way to potential overcome this problem is to employ importance sampling, i.e., sample observation sequences using the current mixture policy $(C^{(k)}, \alpha^{(k,i-1)})$. In this way, we can get an unbiased estimate of the gradient. Formally, we have

$$\mathbb{E}_{\theta\sim Uniform(\{\theta|\max_i \Pr^{\pi^i}(\theta)>0\})}\left[\nabla f(\alpha^{(k,i-1)},\theta)\right] = \frac{\mathbb{E}_{\theta\sim(C^{(k)},\alpha^{(k,i-1)})}\left[\frac{\nabla f(\alpha^{(k,i-1)},\theta)}{\Pr^{(C^{(k)},\alpha^{(k,i-1)})}(\theta)}\right]}{|\{\theta|\max_i \Pr^{\pi^i}(\theta) > 0\}|}.$$

B. Optimal Path Distributions for Infinite Horizon Deterministic MDPs and Observation Functions

In this section, we give a path planing algorithm for infinite horizon deterministic MDPs and observation functions. For a deterministic MDP, the transition probability and observation functions are deterministic. In other words, the environment is a graph, and every path has a fixed observation sequence. We remark that while we consider deterministic MDPs, the reference policy can still be randomized.

Consider an agent that aims to follow predetermined path and thereby induce a predetermined observation sequence, as can be done in MDPs with deterministic transitions and observation functions. In this case, the agent can set the probability of any observation sequence to a desired value and achieve optimality for Problem 2.

As discussed in Section V-A, the straightforward approach is to enumerate observation sequences and solve an optimization problem that minimizes the KL divergence to the reference policy's observation distribution. However, this requires solving an optimization problem with infinitely many variables since the number of observation sequences is infinite. Instead, we give a randomized algorithm for path planning that runs in polynomial time and finds a random path for the agent such that the path satisfies the task constraint in expectation, and the objective value is optimal in expectation.

Algorithm 2: Path planning algorithm for deterministic MDPs

```
Sample c from Uniform([0, 1]).
while True do
    Sample a path ξ from Γ^S.
    if (O(ξ) ∈ L^A) = (c ≤ ν) then
        Find a path ξ' such that (ξ' ⊨ ◇R^A) XOR (c > ν) is true
          and O(ξ') = O(ξ).
        break
Output ξ'
```

Algorithm 2 increases the probabilities of the observation sequences for which there is a path reaching $R^{\mathcal{A}}$. With an abuse of notation, let $O(\xi)$ be the corresponding observation sequence of path ξ. Also, let $\mathcal{L}^{\mathcal{A}}$ be the set of observation sequences such that for every $\theta \in \mathcal{L}^{\mathcal{A}}$ there exists a path ξ satisfying $O(\xi) = \theta$ and $\xi \models \Diamond R^{\mathcal{A}}$. The algorithm relies on rejection sampling and works as follows. First, it samples a path ξ using $\pi^{\mathcal{S}}$. With probability ν, the algorithm accepts ξ if and only if there is a path ξ' that reaches $R^{\mathcal{A}}$ and $O(\xi) = O(\xi')$, i.e., $O(\xi) \in \mathcal{L}^{\mathcal{A}}$. With probability $1 - \nu$, the algorithm accepts ξ if and only if there is no path ξ' that reaches $R^{\mathcal{A}}$ with $O(\xi) = O(\xi')$, i.e., $O(\xi) \notin \mathcal{L}^{\mathcal{A}}$. At the end, the algorithm outputs path ξ'.

Proposition 3. *Assume that $\nu \geq \Pr^{\mathcal{S}}(\xi|O(\xi) \in \mathcal{L}^{\mathcal{A}})$. Let μ be the probability measure induced by Algorithm 2 over the paths of $\mathcal{M}$ and v^* be the optimal value of* (3)*. Algorithm 2 satisfies*

$$\Pr(\xi \models \Diamond R^{\mathcal{A}}|\xi \sim \mu) \geq \nu \text{ and } KL(\mu||\Theta^{\mathcal{S}}) = v^*,$$

and it has an expected time complexity of

$$\mathcal{O}\left(\frac{\nu|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi))]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}})^2} + \frac{(1-\nu)|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi)]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\notin\mathcal{L}^{\mathcal{A}})^2}\right)$$

where $len(\xi = s_0 s_1 \ldots) = \min\{i|s_i \in S^{end}\}$.

Proof of Proposition 3. We first show that $KL(\mu||\Theta^{\mathcal{S}}) = v^*$.

Let θ be an arbitrary observation sequence. If $\theta \in \mathcal{L}^{\mathcal{A}}$, we have

$$\Pr^{\mathcal{A}}(\theta) = \nu \sum_{\substack{\xi\in Paths(\mathcal{M})\\ O(\xi)=\theta}} \frac{\Pr^{\mathcal{S}}(\xi)}{\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}})} = \frac{\nu\Pr^{\mathcal{S}}(\theta)}{\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}})}.$$

Similarly, if $\theta \notin \mathcal{L}^{\mathcal{A}}$, $\Pr^{\mathcal{A}}(\theta) = (1-\nu)\Pr^{\mathcal{S}}(\theta)/\Pr^{\mathcal{S}}(\xi|O(\xi) \notin \mathcal{L}^{\mathcal{A}})$.

The KL divergence is equal to

$$KL(\mu||\Theta^{\mathcal{S}}) = KL\left(Ber(\nu)||Ber(\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}}))\right).$$

We now show that the optimal value of (3) is lower bounded by $KL(\mu||\Theta^{\mathcal{S}})$. Consider a binary clustering C of the observation sequences such that an observation sequence θ is in C if and only if $\theta \in \mathcal{L}^{\mathcal{A}}$. By definition $\Pr^{\mathcal{S}}(C) = \Pr^{\mathcal{S}}(\xi|O(\xi) \in \mathcal{L}^{\mathcal{A}})$.

Let μ^* be an optimal distribution of observation sequences for (3). If $\Pr(C|\mu^*) < \nu$ then $\Pr^{\mathcal{A}}(\Diamond R^{\mathcal{A}}) < \nu$. Hence, $\Pr(C|\mu^*) \geq \nu$. Using (1) and the binary clustering, we have

$$KL(\mu^*||\Theta^{\mathcal{S}}) = v^* \geq KL\left(Ber(\nu)||Ber(\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}}))\right).$$

Since v^* is the optimal value of (3), we have $KL(\mu||\Theta^{\mathcal{S}}) = v^*$.

Note that $\Pr(\xi \models \Diamond R^{\mathcal{A}}|\xi \sim \mu) = \nu$ due to the acceptance condition in the **if** statement.

We now derive the time complexity of Algorithm 2. Sampling a path under a stationary reference policy $\pi^{\mathcal{S}}$ takes $\mathcal{O}(|S||A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi))])$ time in expectation. For a given random observation sequence $O(\xi)$, determining whether $O(\xi) \in \mathcal{L}^{\mathcal{A}}$ is equivalent to the string acceptance problem for NFAs, which has a time complexity of $\mathcal{O}(|S|^2\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi)])$ in expectation. If $c \leq \nu$, sampling a path ξ such that $O(\xi) \in \mathcal{L}^{\mathcal{A}}$ takes $\Pr^{\mathcal{S}}(\xi|O(\xi) \in \mathcal{L}^{\mathcal{A}})^{-1}$ time in expectation. Otherwise, sampling a path ξ such that $O(\xi) \notin \mathcal{L}^{\mathcal{A}}$ takes $\Pr^{\mathcal{S}}(\xi|O(\xi) \notin \mathcal{L}^{\mathcal{A}})^{-1}$ time in expectation. If $c \leq \nu$, finding a path ξ' such that $\xi' \models \Diamond R^{\mathcal{A}}$ and $O(\xi') = O(\xi)$ is equivalent to finding an accepting trace for a given string in NFAs, which has a time complexity of $\mathcal{O}(|S|^2\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi)|O(\xi) \in \mathcal{L}^{\mathcal{A}}])$ in expectation. Similarly, if $c > \nu$, finding a path ξ' such that $\xi' \not\models \Diamond R^{\mathcal{A}}$ and $O(\xi') = O(\xi)$ has a time complexity of $\mathcal{O}(|S|^2\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi)|O(\xi) \notin \mathcal{L}^{\mathcal{A}}])$ in expectation. Overall, the expected time complexity of Algorithm 2 is at the order of

$$\frac{\nu|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}\left[len(\xi)|O(\xi)\in\mathcal{L}^{\mathcal{A}}\right]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}})} + \frac{(1-\nu)|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}\left[len(\xi)|O(\xi)\notin\mathcal{L}^{\mathcal{A}}\right]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\notin\mathcal{L}^{\mathcal{A}})}.$$

Since $len(\xi) \geq 0$, the expected time complexity is bounded by

$$\mathcal{O}\left(\frac{\nu|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}\left[len(\xi)\right]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\in\mathcal{L}^{\mathcal{A}})^2} + \frac{(1-\nu)|S|^2|A|\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}\left[len(\xi)\right]}{\Pr^{\mathcal{S}}(\xi|O(\xi)\notin\mathcal{L}^{\mathcal{A}})^2}\right).$$

■

Proposition 3 shows that likelihood ratio for the observation sequence of the output path is optimal in expectation, as Algorithm 2 boosts the probabilities of all observation sequences for which there is a path that reaches $R^{\mathcal{A}}$ at the same ratio.

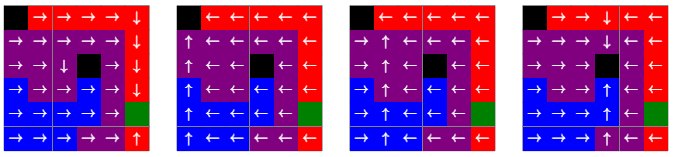
The running time of Algorithm 2 depends both on some properties of the reference policy as well as the size of the MDP. The dependencies on ν and $\Pr^{\mathcal{S}}(\xi|O(\xi) \in \mathcal{L}^{\mathcal{A}})$ are due to rejection sampling. The dependencies on $|S|$, $|A|$ and $\mathbb{E}_{\xi\sim\pi^{\mathcal{S}}}[len(\xi)]$ are due to checking whether the sampled observation sequence is in $\mathcal{L}^{\mathcal{A}}$.

VI. Numerical Example

We demonstrate the synthesis of optimal mixture policies in a grid-world environment shown in Fig. 3. At every state, there are 4 available actions: up, down, left, right, and stay. With probability 0.9, the agent moves in the chosen direction or stays if action stay is chosen. With probability 0.1, the agent moves in the other directions or stays. If a transition is not possible because the agent is at the boundary of the grid, the transition probability is proportionally distributed among the other transitions. The observation function represents a binary temperature sensor that has two levels, Low and $High$. Blue cells are more likely to emit observation Low and red cells are more likely to emit observation $High$. Purple cells emit observations Low and $High$ with equal probabilities.

We consider the mixture of three basis policies shown in Fig. 3b–3d. Policy π^1 reaches the black cell on left in minimum time, and policy π^3 reaches the black cell on right in minimum time. The length of the time horizon is 8. There are 480 observation sequences such that $\max_i \Pr^{\pi^i}(\theta) > 0$. The basis policies π^1, π^2, and π^3, reach the target black cells with probabilities 0.93, 0.54, and 0.75, respectively. We set $\nu = 0.5$. Hence, every mixture of the basis policies is feasible. We initialize the mixing probabilities with a uniform distribution.

Policies π^2 and π^3 are advantageous over policy π^1. Under $\pi^{\mathcal{S}}$, the agent reaches an end state after 6 transitions with high probability



(a) $\pi^{\mathcal{S}}$ (b) π^1 (c) π^2 (d) π^3

Fig. 3: The environment is a 6×6 grid world. The initial state is the bottom left cell. Black cells are the target set of states for the agent. The reference policy and the basis policies are shown in Fig. 3a–3d. Blue cells emit observation Low with probability $1/8$ and H with probability $7/8$. Purple cells emit observation Low with probability $1/2$ and $High$ with probability $1/2$. Red cells emit observation Low with probability $1/8$ and $High$ with probability $7/8$. Black and green are the end states, and they emit observation ε with probability 1.

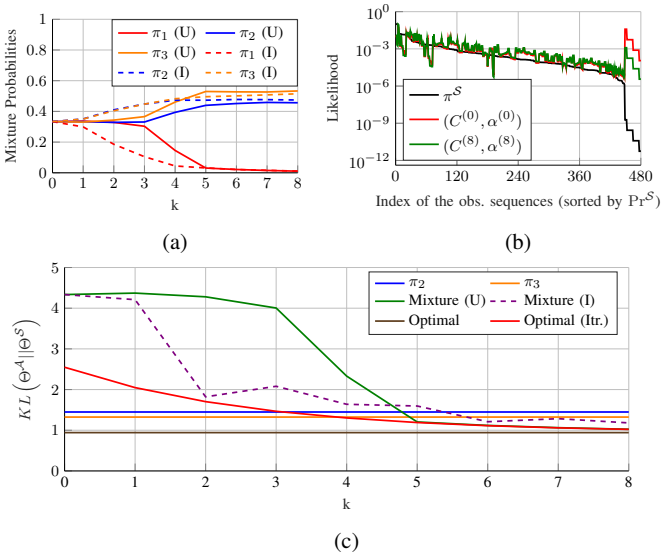


Fig. 4: (a) The mixing probabilities for different values of k in Algorithm 1. 'U' refers to uniform sampling, and 'I' refers to importance sampling of observation sequences using $(C^{(k)}, \alpha^{k,i-1})$. (b) The likelihood values for the observation sequences under the initial and final mixture policies compared to the reference policy. (c) The objective value for different values of k in Algorithm 1. 'Optimal' is the value for the optimal mixture policy. 'Optimal (Itr.)' is the value for the optimal mixture policy at iteration k subject to the constraint $\alpha^{(k)} \in \Delta_{b(k)}$. The objective value for π^1 is 12.23.

(w.h.p.) Policy π^3 is advantageous since it also causes the agent to reach to an end state after 6 transitions w.h.p. The stochasticity in the observation function might lead to the same observation sequences for $\pi^{\mathcal{S}}$ and π^3. Policy π^2 is advantageous, since the observation sequences generated by π^2 resemble the observation sequences generated by $\pi^{\mathcal{S}}$. For example, $Low, Low, Low, High, High, High, \varepsilon, \varepsilon$ is among the most likely observation sequences under $\pi^{\mathcal{S}}$, and $Low, Low, Low, High, High, High, High, \varepsilon$ is among the most likely observation sequences under π^2. The stochasticity in the environment might lead to the same observation sequences for $\pi^{\mathcal{S}}$ and π^2. Policy π^1 is intuitively dissimilar to $\pi^{\mathcal{S}}$ in terms of the induced observation distributions. For example, under π^1, the agent reaches an end state after 5 transitions w.h.p. On the other hand, reaching an end state after 5 transitions is unlikely under $\pi^{\mathcal{S}}$. Overall, we expect the weights of π^2 and π^3 to be higher than the weight of π^1.

We run Algorithm 1 for $k = 1, \ldots, 8$. For uniform sam-

pling, we directly sample from the observation sequences such that $\max_i \Pr^{\pi^i}(\theta) > 0$. In addition to the uniform sampling of observation sequences, we use the importance sampling method, i.e., at iteration k, we sample paths using policy $(C^{(k)}, \alpha^{(k,i-1)})$.

The mixture probabilities are shown in Fig. 4a, and the expected log-likelihood ratios are shown in Fig. 4c. As explained above, both sampling methods assign high weights to π^2 and π^3 and a low weight to π^1. Fig. 4b shows that the final mixture downweights the observation sequences that are unlikely under the reference policy. In Fig. 4c, we observe that as the convexity of the objective function suggests, mixture policies outperform the basis policies and converge to the optimal mixture: When the optimal mixture policy is used, the supervisor needs $\approx 40\%$ more observation sequences to achieve the same detection rate for likelihood-ratio test since

$$\min_i KL\left(\Theta^{\pi^i}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) / \min_{\alpha \in \Delta_0} KL\left(\Theta^{(C^{(0)},\alpha)}_{0:T} || \Theta^{\mathcal{S}}_{0:T}\right) = 1.40.$$

Importance sampling quickly improves value of the objective function. After a single iteration, i.e., 4 sample observation sequences, importance sampling downweights π^1. Uniform sampling outperforms importance sampling after 5 iterations, i.e, 1364 sample observation sequences. The performance of importance sampling is better than uniform sampling for the iterations where the number of samples is lower than the number of possible observation sequences. This property holds because importance sampling creates a bias towards observation sequences that have high $\Pr^{(C^{(k)},\alpha^{(k)})}(\theta)$. In detail, for any θ, the value of the objective function is proportional to $\Pr^{(C^{(k)},\alpha^{(k)})}(\theta)$. Hence, using importance sampling creates a bias towards the observation sequences that highly affect the objective function. These observation sequences are sampled w.h.p. even with a small number of samples. On the other hand, when uniform sampling is used, it is more likely to sample an observation sequence with a low $\Pr^{(C^{(k)},\alpha^{(k)})}(\theta)$. Such observation sequences has a low impact on the objective function. When the number of samples is lower than the number of possible observation sequences, uniform sampling fails to sample observation sequences that have high $\Pr^{(C^{(k)},\alpha^{(k)})}(\theta)$, and performs worse than importance sampling. When the number of samples is high, uniform sampling outperforms importance sampling since importance sampling suffers from high variance.

VII. Conclusions

We considered the problem of deception by an agent under partial observations received by its supervisor. We modeled this problem as hypothesis testing problem in MDPs, and used KL divergence as a proxy of deceptiveness. We showed that finding optimal deceptive policies, while possible, is computationally intractable, and there is no polynomial-time approximation algorithm. As an alternative to the synthesis of optimal policies, we considered special classes of policies where deceptive policies can be synthesized efficiently.

In some settings, the system manager, i.e., the supervisor, may not know the behavioral models of the well-intentioned and deceptive agents. For example, in cyber settings, the users show various behavior. Learning and detection are performed together in these settings [21], [27], [28]. It would be interesting to explore the synthesis deceptive strategies that exploit the vulnerabilities of the learning modules or worsens the detection rate by showing a different behavior to induce a wrong prior for the system manager.

References

[1] C. Kwon, W. Liu, and I. Hwang, "Security analysis for cyber-physical systems against stealthy deception attacks," in *2013 American Control Conference*, 2013, pp. 3344–3349.

[2] M. H. Almeshekah and E. H. Spafford, "Cyber security deception," in *Cyber Deception*. Springer, 2016, pp. 23–50.

[3] W. Hutchinson, "Information warfare and deception." *Informing Science*, vol. 9, 2006.

[4] M. Lloyd, *The Art of Military Deception*. Pen and Sword, 2003.

[5] J. Shim and R. C. Arkin, "A taxonomy of robot deception and its benefits in HRI," in *International Conference on Systems, Man, and Cybernetics*, 2013, pp. 2328–2335.

[6] M. O. Karabag, M. Ornik, and U. Topcu, "Deception in supervisory control," *IEEE Transactions on Automatic Control*, vol. 67, no. 2, pp. 738–753, 2021.

[7] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[8] E. J. Collins and J. M. McNamara, "Finite-horizon dynamic optimisation when the terminal reward is a concave functional of the distribution of the final state," *Advances in Applied Probability*, vol. 30, no. 1, pp. 122–136, 1998.

[9] D. Li and J. B. Cruz Jr, "Information, decision-making and deception in games," *Decision Support Systems*, vol. 47, no. 4, pp. 518–527, 2009.

[10] T. Zhang and Q. Zhu, "Hypothesis testing game for cyber deception," in *International Conference on Decision and Game Theory for Security*. Springer, 2018, pp. 540–555.

[11] A. Saboori and C. N. Hadjicostis, "Current-state opacity formulations in probabilistic finite automata," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 120–133, 2013.

[12] C. Keroglou and C. N. Hadjicostis, "Probabilistic system opacity in discrete event systems," *Discrete Event Dynamic Systems*, vol. 28, no. 2, pp. 289–314, 2018.

[13] B. Bérard, K. Chatterjee, and N. Sznajder, "Probabilistic opacity for Markov decision processes," *Information Processing Letters*, vol. 115, no. 1, pp. 52–59, 2015.

[14] 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.

[15] O. Madani, S. Hanks, and A. Condon, "On the undecidability of probabilistic planning and infinite-horizon partially observable Markov decision problems," in *AAAI/IAAI*, 1999, pp. 541–548.

[16] B. Bonet, "Deterministic POMDPs revisited," in *25th Conference on Uncertainty in Artificial Intelligence*, 2009, pp. 59–66.

[17] D. C. Kozen, *Automata and computability*. Springer Science & Business Media, 2012.

[18] R. E. Stearns and H. B. Hunt III, "On the equivalence and containment problems for unambiguous regular expressions, regular grammars and finite automata," *SIAM Journal on Computing*, vol. 14, no. 3, pp. 598–611, 1985.

[19] L. J. Stockmeyer and A. R. Meyer, "Word problems requiring exponential time (preliminary report)," in *5th Annual ACM Symposium on Theory of Computing*, 1973, pp. 1–9.

[20] J. Neyman and E. S. Pearson, "On the problem of the most efficient tests of statistical hypotheses," *Philosophical Transactions of the Royal Society of London*, vol. 231, no. 694-706, pp. 289–337, 1933.

[21] C. Wressnegger, G. Schwenk, D. Arp, and K. Rieck, "A close look on n-grams in intrusion detection: anomaly detection vs. classification," in *ACM workshop on Artificial intelligence and security*, 2013, pp. 67–76.

[22] J. Burghardt, "Example to demonstrate that the subset property for regular languages is NP-hard," https://en.wikipedia.org/wiki/File:RegSubsetNP.pdf, 2016, accessed Aug 5, 2021.

[23] M. Krötzsch, T. Masopust, and M. Thomazo, "Complexity of universality and related problems for partially ordered NFAs," *Information and Computation*, vol. 255, pp. 177–192, 2017.

[24] L. G. Valiant, "The complexity of enumeration and reliability problems," *SIAM Journal on Computing*, vol. 8, no. 3, pp. 410–421, 1979.

[25] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[26] O. Bernardi and O. Giménez, "A linear algorithm for the random sampling from regular languages," *Algorithmica*, vol. 62, no. 1, pp. 130–145, 2012.

[27] J. Zhang, L. Pan, Q.-L. Han, C. Chen, S. Wen, and Y. Xiang, "Deep learning based attack detection for cyber-physical system cybersecurity: A survey," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 377–391, 2021.

[28] I. Rosenberg, A. Shabtai, Y. Elovici, and L. Rokach, "Adversarial machine learning attacks and defense methods in the cyber security domain," *ACM Computing Surveys (CSUR)*, vol. 54, no. 5, pp. 1–36, 2021.