Entropy Maximization for Partially Observable Markov Decision Processes

Yagiz Savas*, Michael Hibbard*, Bo Wu, Takashi Tanaka, and Ufuk Topcu

***Abstract*—We study the problem of synthesizing a controller that maximizes the entropy of a partially observable Markov decision process (POMDP) subject to a constraint on the expected total reward. Such a controller minimizes the predictability of an agent's trajectories to an outside observer while guaranteeing the completion of a task expressed by a reward function. We first prove that an agent with partial observations can achieve an entropy at most as well as an agent with perfect observations. Then, focusing on finite-state controllers (FSCs) with deterministic memory transitions, we show that the maximum entropy of a POMDP is lower bounded by the maximum entropy of the parametric Markov chain (pMC) induced by such FCSs. This relationship allows us to recast the entropy maximization problem as a so-called parameter synthesis problem for the induced pMC. We then present an algorithm to synthesize an FSC that locally maximizes the entropy of a POMDP over FSCs with the same number of memory states. In numerical examples, we illustrate the relationship between the maximum entropy, the number of memory states in the FSC, and the expected reward.**

I. Introduction

Entropy [1] is an information-theoretic measure to quantify the unpredictability of outcomes in a random variable. In this paper, we consider a sequential decision-making framework of partially observable Markov decision processes (POMDPs) in which a reward in terms of the entropy is introduced in addition to the classical state-dependent reward. More specifically, in the POMDP formulation that we consider, we look for a *controller* that maximizes the entropy reward while making sure that the expected state-dependent reward is above a given threshold. Intuitively, the entropy reward plays a role to promote the unpredictability of the controlled process to an outside observer. Therefore, the considered POMDP formulation provides a meaningful framework for sequential decision-making in stochastic environments with imperfect information and nondeterministic choices, where a given task should be accomplished in the most unpredictable way.

A controller in a POMDP is a decision rule that resolves the nondeterminism and induces a stochastic process. Following [2], [3], we quantify the unpredictability of realizations in an induced stochastic process by defining the entropy of the process as the joint entropy of a sequence of random variables. We then mathematically show that the maximum entropy of a POMDP is upper bounded by the entropy of its corresponding fully observable counterpart, which is a Markov decision process (MDP).

For a given POMDP, the main objective of this paper is to synthesize a controller that induces a process whose realizations accumulate rewards in the most unpredictable way to an outside observer. Controller synthesis problems for POMDPs are notoriously hard to solve. The optimal controllers are often required to take the full observation history into account which makes searching for them undecidable in the infinite horizon case and PSPACE-complete in the finite horizon case [4], [5]. For computational tractability, POMDP controllers are often restricted to have finite states that represent finite observation memory [6]. Furthermore, in contrast to classical POMDP problems in which the optimal controllers are deterministic, problems adopting information-theoretic performance criteria such as entropy like in this paper, admit randomized controllers that specify probability distributions over action selection.

In this paper, we synthesize a randomized finite-state controller (FSC) for a POMDP that specifies a probability distribution over actions for each of its memory states [7]. In particular, we consider the POMDP entropy maximization problem over all FSCs with a fixed number of memory states. A key observation is that one can use a parametric Markov chain (pMC) to succinctly represent the product between a POMDP and the set of all FSCs with a fixed number of memory states [8], [9]. By restricting our attention to FSCs with deterministic memory transitions, we recast the POMDP controller synthesis problem as a so-called parameter synthesis problem for a pMC whose entropy we aim to maximize. To build a connection between the entropy of the POMDP and that of the corresponding pMC, we first prove that the maximum entropy of a pMC induced from a POMDP by FSCs with deterministic memory transitions is a lower bound on the maximum entropy of the POMDP. Furthermore, for some specific memory transition functions in FSCs, we show that one can monotonically obtain stochastic processes induced from a POMDP with higher entropy by increasing the number of memory states in the FSCs. Finally, we present a computation algorithm, based on a nonlinear optimization problem, to synthesize parameters in an FSC to maximize the entropy of a pMC subject to expected reward constraints.

One application of this theoretical framework is the synthesis of a controller for an autonomous agent carrying out a mission in an adversarial environment. In particular, if the agent's sensor measurements are noisy and the mission is defined in terms of a reward function, the synthesized controller leaks the minimum information about the agent's trajectories to an

All authors are with the Department of Aerospace Engineering and Engineering Mechanics, and the Oden Institute for Computational Engineering and Sciences, University of Texas, Austin, 201 E 24th St, Austin, TX 78712. email: {yagiz.savas, mwhibbard, bwu3, ttanaka, utopcu}@utexas.edu

* Y. Savas and M. Hibbard contributed equally to this work.

outside observer while guaranteeing the accumulation of an expected total reward above a desired threshold. Furthermore, the proposed methods could also be applied to distribute traffic assignments over a network with possibly noisy traffic information, which is known as stochastic traffic assignment [10], as a higher entropy in this scenario promotes the use of different paths.

Related Work. A preliminary version of this paper has appeared in [11], where we present solutions for entropy maximization over FSCs with a *specific memory transition function* and the same number of memory states. This considerably extended version includes detailed proofs for all theoretical results, a nonlinear optimization problem formulating the entropy maximization over all deterministic FSCs with the same number of memory states, and an extended numerical examples section.

A recent study [3] showed that an entropy-maximizing controller for an MDP could be synthesized efficiently by solving a convex optimization problem. In POMDPs, entropy has often been used for active sensing applications [12]–[14], where an agent seeks to select actions that maximize its information gain from the environment. These applications differ from our own as we seek to maximize the entropy of the trajectories an agent follows rather than maximizing its knowledge of the environment.

In the reinforcement learning literature, the entropy of a controller has been used as a regularization term in an agent's objective to balance the trade-off between exploration and exploitation [15]. As discussed in [16], using a controller with high entropy, an agent can learn a greater variety of admissible methods to complete a task, leading to a greater robustness when subsequently fine-tuned to specific scenarios. In imitation learning [17], a controller with high entropy similarly yields greater robustness when the provided demonstrations are imperfect. Unlike the aforementioned work, here we aim to synthesize a controller that maximizes the entropy of the induced stochastic process, rather than synthesizing a controller with high entropy.

A range of solution techniques exist for POMDP controller synthesis using FSCs. For deterministic FSCs, existing approaches include branch-and-bound method [6], automaton learning-based method [18], and expectation-maximization [19]. They mainly target for finding an optimal transition structure of the FSC, As for randomized FSCs, in addition to the transition structure, one also needs to optimize the probabilistic transition probabilities between FSC states and the action selection probabilities. To this end, researchers propose solutions using policy iteration [7], [20], gradient descent [21], and nonlinear optimization [22], [23]. However, the results mentioned above only consider state-dependent reward optimization or the satisfaction of a given specification. In contrast, we consider the synthesis of FSCs for entropy maximization, which is a nonlinear objective that requires a new optimization formulation as well as solution techniques.

Contribution. The contributions of this paper are four-fold. First, we prove that the maximum entropy for a POMDP is bounded by the maximum entropy of its underlying fully observable MDP. Secondly, by restricting the scope of the FSC synthesis problem to finite-state controllers with deterministic memory transitions, we prove that the maximum entropy of the induced pMC is a lower bound on the maximum entropy of the POMDP. Thirdly, we present a nonlinear optimization problem whose solution provides a controller that maximizes the entropy of the POMDP over all deterministic FSCs with the same number of memory states. Lastly, for deterministic FSCs, we propose a specific memory transition function which increases the entropy of the induced stochastic process respect to an increasing number of memory states.

Organization. We provide the modeling framework and preliminary definitions in Section II. We then formally state the entropy maximization problem for the finite and infinite horizons in Section III. We show that the maximum entropy of a POMDP is upper bounded by that of its underlying MDP in Section IV. We then focus on FSCs in Section V, and prove that the maximum entropy of the pMC induced by a deterministic FSC is a lower bound for the maximum entropy of the POMDP. We then present a procedure to synthesize a local optimal FSC to maximize the POMDP entropy subject to reward constraints. We provide numerical examples in Section VI and conclude with possible future directions in Section VII. Proofs for all technical results are provided in Appendix A.

II. Preliminaries

For a set $\mathcal{S}$, we denote its power set and cardinality by $2^{\mathcal{S}}$ and $|\mathcal{S}|$, respectively. The set of all probability distributions on a finite set $\mathcal{S}$, i.e., all functions $f{:}\mathcal{S}{\to}[0,1]$ such that $\sum_{s\in\mathcal{S}} f(s){=}1$, is denoted by $\Delta(\mathcal{S})$. The set $\{1,2,3,\ldots\}$ of natural numbers is denoted by $\mathbb{N}$. For a sequence $\{X_t, t{\in}\mathbb{N}\}$, a subsequence $(X_k, X_{k+1}, \ldots, X_l)$ is denoted by X_k^l. The subsequence $(X_1, X_2, \ldots, X_l)$ is simply denoted by X^l.

A. Partially Observable Markov Decision Processes

Definition 1: A *partially observable Markov decision process* (POMDP) is a tuple $\mathcal{M} = (\mathcal{S}, s_I, \mathcal{A}, \mathcal{P}, \mathcal{Z}, \mathcal{O}, \mathcal{R})$ where $\mathcal{S}$ is a finite set of states, $s_I{\in}\mathcal{S}$ is a unique initial state, $\mathcal{A}$ is a finite set of actions, $\mathcal{P}{:}\mathcal{S}{\times}\mathcal{A}{\to}\Delta(\mathcal{S})$ is a transition function, $\mathcal{Z}$ is a finite set of observations, $\mathcal{O}{:}\mathcal{S}{\to}\Delta(\mathcal{Z})$ is an observation function, and $\mathcal{R} : \mathcal{S}{\times}\mathcal{A}{\to}\mathbb{R}$ is a reward function.

For simplicity, we assume that all actions $a{\in}\mathcal{A}$ are available in all states $s{\in}\mathcal{S}$. Additionally, we assume that only a single observation is available from the initial state, i.e., $|\mathcal{O}(s_I)|{=}1$. For the ease of notation, we denote the transition probability $\mathcal{P}(s'|s,a)$ and the observation probability $\mathcal{O}(z|s)$ by $\mathcal{P}_{s,a,s'}$ and $\mathcal{O}_{s,z}$, respectively.

For a POMDP $\mathcal{M}$, the *corresponding fully observable* MDP $\mathcal{M}_{fo}$ is obtained by setting $\mathcal{Z}{=}\mathcal{S}$ and $\mathcal{O}_{s,s}{=}1$ for all $s{\in}\mathcal{S}$. A *Markov chain* (MC) is a fully observable MDP such that $|\mathcal{A}|{=}1$.

A *system history* of length $t{\in}\mathbb{N}$ for a POMDP $\mathcal{M}$ is a sequence $h^t{=}(s_I, a_1, s_2, a_2, s_3, \ldots, s_t)$ of states and actions such that $\mathcal{P}_{s_k,a_k,s_{k+1}}{>}0$ for all $k{\geq}1$. We denote the set of all system histories of length t by $\mathcal{H}^t$.

For any system history $h^t{=}(s_I, a_1, s_2, \ldots, s_t)$ of length t, there is an associated *observation history* $o^t{=}(z_I, a_1, z_2, \ldots, z_t)$ of length $t{\in}\mathbb{N}$ where $\mathcal{O}_{s_k,z_k}{>}0$

for all $k\geq 1$. Note that there are, in general, multiple observation histories that are admissible for a given system history. For a POMDP $\mathcal{M}$, we denote the collection of all observation histories of length t by $Obs^t_{\mathcal{M}}$ and define the set of all observation histories as $Obs_{\mathcal{M}}:=\cup_{t\in\mathbb{N}} Obs^t_{\mathcal{M}}$.

Definition 2: For a POMDP $\mathcal{M}$, a *controller* π is a mapping $\pi : Obs_{\mathcal{M}}\to\Delta(\mathcal{A})$. We denote the set of all controllers by $\Pi(\mathcal{M})$.

The probability that the controller π takes the action $a\in\mathcal{A}$ upon receiving the observation history $o^t\in Obs^t_{\mathcal{M}}$ is denoted by $\pi(a|o^t)$.

B. Entropy of Stochastic Processes

The *entropy of a random variable* X with a countable support $\mathcal{X}$ and probability mass function (pmf) $p(x)$ is

$$H(X) := -\sum_{x\in\mathcal{X}} p(x)\log p(x). \tag{1}$$

We use the convention that $0\log 0=0$. Let (X_1, X_2) be a pair of random variables with the joint pmf $p(x_1, x_2)$ and the support $\mathcal{X}\times\mathcal{X}$. The *joint entropy* of (X_1, X_2) is

$$H(X_1, X_2) := -\sum_{x_1\in\mathcal{X}}\sum_{x_2\in\mathcal{X}} p(x_1, x_2)\log p(x_1, x_2), \tag{2}$$

and the *conditional entropy* of X_2 given X_1 is

$$H(X_2|X_1) := -\sum_{x_1\in\mathcal{X}}\sum_{x_2\in\mathcal{X}} p(x_1, x_2)\log p(x_2|x_1). \tag{3}$$

The definitions of the joint and conditional entropies extend to collections of k random variables as shown in [1]. A discrete *stochastic process* $\mathbb{X}$ is a discrete time-indexed sequence of random variables, i.e., $\mathbb{X}=\{X_k\in\mathcal{X}:k\in\mathbb{N}\}$.

Definition 3: (Entropy of a stochastic process) [24] The *entropy of a stochastic process* $\mathbb{X}$ is defined as

$$H(\mathbb{X}) := \lim_{k\to\infty} H(X_1, X_2, \ldots, X_k). \tag{4}$$

The above definition is different from the *entropy rate* of a stochastic process, which is defined as $\lim_{k\to\infty}\frac{1}{k}H(X^k)$ when the limit exists [1]. The limit in (4) either converges to a non-negative real number or diverges to positive infinity [24].

For a POMDP $\mathcal{M}$, a controller $\pi\in\Pi(\mathcal{M})$ induces a discrete stochastic process $\{S_k\in\mathcal{S}:k\in\mathbb{N}\}$ in which each S_k is a random variable over the state space $\mathcal{S}$. We denote the entropy of a POMDP $\mathcal{M}$ under a controller $\pi\in\Pi(\mathcal{M})$ by $H^\pi(\mathcal{M})$.

III. Problem Statement

We consider an *agent* whose behavior is modeled as a POMDP and an *outside observer* whose objective is to infer the states occupied by the agent in the future from the states occupied in the past. Being aware of the observer's objective, the agent aims to synthesize a controller that minimizes the predictability of its future states while ensuring that the expected total reward it collects exceeds a specified threshold.

We measure the predictability of the agent's future states by the entropy of the underlying stochastic process. The rationale behind this choice can be better understood by recalling (see, e.g., Theorem 2.5.1 in [1]) that, for an arbitrary controller $\pi\in\Pi(\mathcal{M})$, the identity

$$H^\pi(S_1, S_2, \ldots, S_N) = H^\pi(S_t^N|S^{t-1}) + H^\pi(S^{t-1}) \tag{5}$$

holds for any $N\in\mathbb{N}$ and $t\leq N$. Therefore, by maximizing the value of the left hand side of (5), one maximizes the entropy of all future sequences $(S_t, \ldots, S_N)$ for any given history of sequence $(S_1, \ldots, S_{t-1})$.

We first focus on an agent with a finite decision horizon and define the finite horizon entropy maximization problem as follows:

Problem 1 (Finite horizon entropy maximization): For a POMDP $\mathcal{M}$, a finite decision horizon $N\in\mathbb{N}$, and a reward threshold $\Gamma\in\mathbb{R}$, synthesize a controller $\pi^\star\in\Pi(\mathcal{M})$ that solves the following problem:

$$\underset{\pi\in\Pi(\mathcal{M})}{\text{maximize}} \quad H^\pi(S_1, S_2, \ldots, S_N) \tag{6a}$$

$$\text{subject to:} \quad \mathbb{E}^\pi\Big[\sum_{t=1}^{N}\mathcal{R}(S_t, A_t)\Big] \geq \Gamma. \tag{6b}$$

In the finite horizon entropy maximization problem, we seek a controller that randomizes the agent's *finite length state trajectories* by using only the *observation history* information.

Next, we consider an agent with infinite decision horizon whose objective is to randomize its infinite length state trajectories. Noting that

$$\lim_{t\to\infty} H^\pi(S_1, S_2, \ldots, S_t) = H^\pi(S_1) + \sum_{t=2}^{\infty} H^\pi(S_t|S^{t-1}), \tag{7}$$

we treat each term $H^\pi(S_t|S^{t-1})$ as a virtual local reward for the agent. Note that $H^\pi(S_1)=0$ for any $\pi\in\Pi(\mathcal{M})$ since we assume $|\mathcal{O}(s_I)|=1$. Allowing the agent to discount its future rewards $\mathcal{R}(S_t, A_t)$ as well as its virtual entropy reward $H^\pi(S_t|S^{t-1})$ to ensure the finiteness of the solution, we define the infinite horizon entropy maximization problem as follows:

Problem 2 (Infinite horizon entropy maximization): For a POMDP $\mathcal{M}$, a discount factor $\beta\in[0, 1)$, and a reward threshold $\Gamma\in\mathbb{R}$, synthesize a controller $\pi^\star\in\Pi(\mathcal{M})$ that solves the following problem:

$$\underset{\pi\in\Pi(\mathcal{M})}{\text{maximize}} \quad \sum_{t=2}^{\infty}\beta^{t-2}H^\pi(S_t|S^{t-1}) \tag{8a}$$

$$\text{subject to:} \quad \mathbb{E}^\pi\Big[\sum_{t=1}^{\infty}\beta^{t-1}\mathcal{R}(S_t, A_t)\Big] \geq \Gamma. \tag{8b}$$

For a reward function $\mathcal{R}:S\times\mathcal{A}\to\mathbb{R}$, it is known [25] that

$$\sup_{\pi\in\Pi(\mathcal{M})} \mathbb{E}^\pi\Big[\sum_{t=1}^{N}\mathcal{R}(S_t, A_t)\Big] \leq \sup_{\pi\in\Pi(\mathcal{M}_{fo})} \mathbb{E}^\pi\Big[\sum_{t=1}^{N}\mathcal{R}(S_t, A_t)\Big].$$

The above inequality implies that an agent with perfect observations can collect an expected total reward that is at least as high as the expected total reward collected by an agent with imperfect observations. Since the objective functions in the entropy maximization problems are quite different from the classical expected total reward objective, it is not obvious whether a similar claim holds for the entropy maximization problems. In the next section, we establish that an agent with

perfect observations can indeed randomize its trajectories at least as well as an agent with imperfect observations.

It is known that deciding the existence of a controller that satisfies the constraint (6b) is, in general, PSPACE-complete [26]. Moreover, for $\beta\in[0,1)$, the existence of a policy that satisfies the constraint (8b) is, in general, undecidable [4]. Therefore, the synthesis of globally optimal controllers that solve the entropy maximization problems is, in general, intractable. In the second part of the paper, we restrict our attention to a special class of controllers, namely finite-state controllers (FSCs). We present a method to synthesize FSCs that are local optimal solutions to entropy maximization problems among all FSCs with fixed number of memory states and fixed memory transition functions.

IV. An Upper Bound on Maximum Entropy

In this section, we establish that an agent with perfect observations can randomize its trajectories at least as well as an agent with imperfect observations. Formally, we show that, for any $N\in\mathbb{N}\cup\{\infty\}$,

$$\sup_{\pi\in\Pi(\mathcal{M})} H^{\pi}(S_1,S_2,\dots,S_N)\leq \sup_{\pi\in\Pi(\mathcal{M}_{fo})} H^{\pi}(S_1,S_2,\dots,S_N).$$

We first prove that the above inequality holds for $N\in\mathbb{N}$. Then, using a monotonocity argument, we show that the inequality still holds as $N\to\infty$.

Recall that for a POMDP $\mathcal{M}$ under a controller $\pi\in\mathcal{M}$, we have the identity

$$H(S_1,S_2,\dots,S_N)=\sum_{t=2}^{N} H(S_t|S^{t-1}).$$

For a given system history $h^t=(s_I,a_1,s_2,a_2,s_3,\dots,s_t)$, let the sequences $s^t=(s_I,s_2,s_3,\dots,s_t)$ and $a^t=(a_1,a_2,a_3,\dots,a_t)$ be the corresponding state and action histories of length t, respectively. We denote the set of all state and action histories of length t by $\mathcal{SH}^t$ and $\mathcal{AH}^t$. Additionally, we define the set of all possible state and action histories as $\mathcal{SH}:=\cup_{t\in\mathbb{N}}\mathcal{SH}^t$ and $\mathcal{AH}:=\cup_{t\in\mathbb{N}}\mathcal{AH}^t$.

It can be shown that, for a POMDP $\mathcal{M}$ under the controller $\pi\in\Pi(\mathcal{M})$, the realization probability $Pr^{\pi}(s^{t+1}|s^t)$ of the state history $s^{t+1}\in\mathcal{SH}^{t+1}$ for a given $s^t\in\mathcal{SH}^t$ is

$$Pr^{\pi}(s^{t+1}|s^t)=\sum_{a^t\in\mathcal{AH}^t}\prod_{k=1}^{t}\mu_k(a_k|h^k)\mathcal{P}_{s_t,a_t,s_{t+1}}. \quad (9)$$

In the above equation, h^k are prefixes of h^t from which the state sequence s^t is obtained, and $\mu_t:\mathcal{H}^t\to\Delta(\mathcal{A})$ is a mapping such that

$$\mu_t(a|h^t):=\sum_{o^t\in Obs^t_{\mathcal{M}}}\pi(a|o^t)Pr(o^t|h^t). \quad (10)$$

Note that the realization probability $Pr(o^t|h^t)$ of the observation history o^t for a given h^t can be recursively written as

$$Pr(o^t|h^t)=\mathcal{O}_{s_t,z_t}\mathcal{P}_{s_{t-1},a_{t-1},s_t}Pr(o^{t-1}|h^{t-1}) \quad (11)$$

for all $t>1$, since, by assumption, $o^1=z_I$ with probability 1.

Now, for a given controller $\pi\in\Pi(\mathcal{M})$ and a finite constant $N\in\mathbb{N}$, let $\mathcal{V}^{\pi}_{t,N}:\mathcal{SH}^t\to\mathbb{R}$ be the *value function* such that

$$\mathcal{V}^{\pi}_{t,N}(s^t):=H(S_{t+1}|S^t=s^t)+\sum_{k=t+1}^{N}H^{\pi}(S_{k+1}|S^k_t,S^t=s^t). \quad (12)$$

Lemma 1: For a POMDP $\mathcal{M}$, a controller $\pi\in\Pi(\mathcal{M})$ and a finite constant $N\in\mathbb{N}$, the value function $\mathcal{V}^{\pi}_{t,N}$, defined in (12), satisfies the equality

$$\mathcal{V}^{\pi}_{t,N}(s^t)=H^{\pi}(S_{t+1}|S^t=s^t)+\sum_{s^{t+1}\in\mathcal{SH}^{t+1}}Pr^{\pi}(s^{t+1}|s^t)\mathcal{V}^{\pi}_{t+1,N}(s^{t+1}) \quad (13)$$

for all $t<N$ and $s^t\in\mathcal{SH}^t$.

We remind the reader that the proof of all technical results, including the proof of Lemma 1, are provided in Appendix A. For $t\leq N$, let $\mathcal{V}^{\star}_{t,N}:\mathcal{SH}^t\to\mathbb{R}$ be a function such that

$$\mathcal{V}^{\star}_{t,N}(s^t):=\sup_{\pi\in\Pi(\mathcal{M})}\mathcal{V}^{\pi}_{t,N}(s^t). \quad (14)$$

Then, using Lemma 1 together with *the principal of optimality* [27, Chapter 4], we conclude that, for all $t<N$ and $s^t\in\mathcal{SH}^t$,

$$\mathcal{V}^{\star}_{t,N}(s^t)=\sup_{\pi\in\Pi(\mathcal{M})}\Big[H^{\pi}(S_{t+1}|S^t=s^t)+\sum_{s^{t+1}\in\mathcal{SH}^{t+1}}Pr^{\pi}(s^{t+1}|s^t)\mathcal{V}^{\star}_{t+1,N}(s^{t+1})\Big]. \quad (15)$$

Note that, by construction, for any $N\in\mathbb{N}$, we have

$$\sup_{\pi\in\Pi(\mathcal{M})} H^{\pi}(S_1,S_2,\dots,S_N)=\mathcal{V}^{\star}_{1,N}(s_I).$$

Recall that for any given controller $\pi\in\Pi(\mathcal{M})$ on a POMDP $\mathcal{M}$, we can construct, through (10), a controller $\pi'\in\Pi(\mathcal{M}_{fo})$ on the corresponding MDP $\mathcal{M}_{fo}$ which satisfies $Pr^{\pi}(s^{t+1}|s^t)=Pr^{\pi'}(s^{t+1}|s^t)$ for all $s^t,s^{t+1}\in\mathcal{SH}$. Then, for all $s^t\in\mathcal{SH}$, we have

$$\sup_{\pi\in\Pi(\mathcal{M})} H^{\pi}(S_{t+1}|S^t=s^t)\leq \sup_{\pi\in\Pi(\mathcal{M}_{fo})} H^{\pi}(S_{t+1}|S^t=s^t).$$

Informally, having access to the state history s^t, a controller $\pi'\in\Pi(\mathcal{M}_{fo})$ can attain an immediate reward $H^{\pi'}(S_{t+1}|S^t=s^t)$ in (15) that is at least as high as the immediate reward achieved by a controller $\pi\in\Pi(\mathcal{M})$. Then, we have the following result as a consequence of Lemma 1.

Theorem 1: For a POMDP $\mathcal{M}$ and a finite constant $N\in\mathbb{N}$,

$$\sup_{\pi\in\Pi(\mathcal{M})} H^{\pi}(S_1,S_2,\dots,S_N)\leq \sup_{\pi\in\Pi(\mathcal{M}_{fo})} H^{\pi}(S_1,S_2,\dots,S_N). \quad (16)$$

The extension of Theorem 1 to infinite state sequences, i.e., to the case where $N\to\infty$, is rather straightforward. Since $\mathcal{V}^{\pi}_{t,N}$, defined in (12), is monotonically non-decreasing in N for all $\pi\in\Pi(\mathcal{M})$, i.e., $\mathcal{V}^{\pi}_{t,N+1}\geq\mathcal{V}^{\pi}_{t,N}$, we have

$$\sup_{\pi\in\Pi(\mathcal{M})}\lim_{N\to\infty}\mathcal{V}^{\pi}_{t,N}(s^t)=\lim_{N\to\infty}\sup_{\pi\in\Pi(\mathcal{M})}\mathcal{V}^{\pi}_{t,N}(s^t) \quad (17)$$

for all $s^t\in\mathcal{SH}^t$. Therefore, by taking the limits of both sides in (16), we conclude that the inequality between supremums still

holds as $N\to\infty$. Finally, we conclude this section by noting that, with a slight modification of the statement of Lemma 1, it can be shown that all results presented in this section hold even if the future entropy rewards are discounted as in (8a).

V. Entropy Maximization Over Finite-State Controllers

The synthesis of controllers that are global optimal solutions of the entropy maximization problems is, in general, intractable due to the constraints (6b) and (8b). For tractability, we restrict our attention to a subset of general controller space and develop methods to synthesize local optimal controllers within this restricted domain. Specifically, in this section, we focus on finite-state controllers (FSCs) with a fixed number of memory states.

Definition 4: For a POMDP $\mathcal{M}$, a *k-finite-state controller* (k-FSC) is a tuple $\mathcal{C}=(Q, q_1, \gamma, \delta)$, where $Q=\{q_1, q_2, \dots, q_k\}$ is a finite set of memory states, $q_1\in Q$ is the initial memory state, $\gamma:Q\times\mathcal{Z}\to\Delta(\mathcal{A})$ is a decision function and $\delta:Q\times\mathcal{Z}\times\mathcal{A}\to\Delta(Q)$ is a memory transition function. We denote the collection of all k-FSCs by $\mathcal{F}_k(\mathcal{M})$.

For a POMDP, a k-FSC induces a Markov chain (MC). It is shown in [23] that the set of all MCs that can be induced by a k-FSC is the set of all well-defined instantiations of a specific parametric Markov chain (pMC). Therefore, without loss of generality, one can work on that specific pMC to synthesize an instantiation which corresponds to the MC induced by an entropy-maximizing FSC. In the following sections, we first provide formal definitions of pMCs and their instantiations. We then reformulate the entropy maximization problem over k-FSCs as another optimization problem over pMCs.

Remark 1: Any given instance of the finite horizon entropy maximization problem can be reduced to an instance of the infinite horizon entropy maximization problem in time polynomial in the size of the POMDP and the decision horizon N. For brevity, we establish our results only for the infinite horizon entropy maximization problem and provide the details of the aforementioned reduction in Appendix B.

A. Parametric Markov Chains

We develop solutions to entropy maximization problems through the use of parametric Markov chains.

Definition 5: For a POMDP $\mathcal{M}$ and a constant $k>0$, the *induced parametric Markov chain* (pMC) is a tuple $\mathcal{D}_{\mathcal{M},k}=(S_{\mathcal{M},k}, s_{I,\mathcal{M},k}, V_{\mathcal{M},k}, P_{\mathcal{M},k}, \mathcal{R}_{\mathcal{M},k})$ where entries are as follows;

- $S_{\mathcal{M},k} = \mathcal{S}\times\{1,2,...,k\}$ is the finite set of states.
- $s_{I,\mathcal{M},k} = \langle s_I, 1\rangle$ is the initial state.
- $V_{\mathcal{M},k} = \{\gamma_a^{q,z} | z\in\mathcal{Z}, q\in Q, a\in\mathcal{A}\}$
  $\cup\{\delta_{q'}^{q,z,a} | z\in\mathcal{Z}, q, q'\in Q, a\in\mathcal{A}\}$
  is the finite set of parameters.
- $P_{\mathcal{M},k} : S_{\mathcal{M},k}\to\Delta(S_{\mathcal{M},k})$ is a transition function such that $P_{\mathcal{M},k}(\langle s',q'\rangle \mid \langle s,q\rangle) := \sum_{a\in A}\overline{P}(\langle s',q'\rangle \mid \langle s,q\rangle, a)$ for all $\langle s,q\rangle, \langle s',q'\rangle\in S_{\mathcal{M},k}$ where $\overline{P} : S_{\mathcal{M},k}\times\mathcal{A}\to\Delta(S_{\mathcal{M},k})$ is a mapping such that

$$\overline{P}(\langle s',q'\rangle \mid \langle s,q\rangle, a) := \sum_{z\in\mathcal{Z}} \mathcal{O}_{s,z}\,\mathcal{P}_{s,a,s'}\,\gamma_a^{q,z}\,\delta_{q'}^{q,z,a}. \quad (18)$$

- $\mathcal{R}_{\mathcal{M},k}(\langle s,q\rangle, a) := \mathcal{R}(s,a)$ for all $s\in\mathcal{S}$, $q\in Q$, and $a\in\mathcal{A}$.

Note that when defining the parametric transition probabilities $P_{\mathcal{M},k}$ of the induced pMC, we suppose that the observations are obtained before selecting actions $a\in\mathcal{A}$. We also remark that different definitions of the induced pMC can be used to reduce the number of parameters in $V_{\mathcal{M},k}$ [23].

An MC can now be obtained from an induced pMC by instantiating the parameters $V_{\mathcal{M},k}$ in a way that the resulting transition function is well-defined. Formally, a *well-defined instantiation* for $V_{\mathcal{M},k}$ is a function $u : V_{\mathcal{M},k}\to[0,1]$ such that, for all $a\in\mathcal{A}$, $q\in Q$, and $z\in\mathcal{Z}$,

$$\sum_{a\in\mathcal{A}} u(\gamma_a^{q,z}) = 1 \quad \text{and} \quad \sum_{q'\in Q} u(\delta_{q'}^{q,z,a}) = 1.$$

We note that the definition of an instantiation provided in [23] is different from the definition used here. However, as explained in [23], it is possible to construct a new pMC in time polynomial in the size of the induced pMC, on which both definitions become equivalent. Therefore, the definition of an instantiation provided here is as expressive as the definition provided in [23].

Applying a well-defined instantiation u to the induced pMC $\mathcal{D}_{\mathcal{M},k}$, denoted $\mathcal{D}_{\mathcal{M},k}[u]$, replaces each parametric transition probability $P_{\mathcal{M},k}$ by $P^u_{\mathcal{M},k}$. It is straightforward to verify that $\mathcal{D}_{\mathcal{M},k}[u]$ is a Markov chain. For a pMC $\mathcal{D}_{\mathcal{M},k}$, we denote the set of all well-defined instantiations by $\Upsilon_{\mathcal{M},k}$.

For an induced pMC $\mathcal{D}_{\mathcal{M},k}$, every well-defined instantiation $u\in\Upsilon_{\mathcal{M},k}$ describes a k-FSC $\mathcal{C}_u\in\mathcal{F}_k(\mathcal{M})$ [23]. Thus, we can synthesize all admissible MCs that can be induced from a POMDP $\mathcal{M}$ by a k-FSC $\mathcal{C}\in\mathcal{F}_k(\mathcal{M})$ through well-defined instantiations u over $V_{\mathcal{M},k}$. As a result, it is possible to reduce the entropy maximization problems over k-FSCs to a parameter synthesis problem for the induced pMC.

B. A Reformulation of Entropy Maximization Problem over Parametric Markov Chains

Recall that we are interested in maximizing the entropy of the state sequence of a given POMDP, as can be seen in (8a). It can be shown that the entropy of the state sequence of the induced pMC is an inseparable function of the entropy of the state sequence of the POMDP and the entropy of the state sequence of the k-FSC. Therefore, an entropy-maximizing k-FSC for a POMDP cannot be synthesized using the induced pMC unless we introduce further restrictions on the memory transition function of the k-FSCs. To this aim, in this section, we focus on k-FSCs with deterministic memory transitions.

Definition 6: A *deterministic k-FSC* $\mathcal{C}=(Q, q_1, \gamma, \delta)$ is a k-FSC such that for all $q\in Q$, $|Succ(q)|= 1$. We denote the collection of all deterministic k-FSCs by $\mathcal{F}_k^{det}(\mathcal{M})$.

For a given k-FSC $\mathcal{C}\in\mathcal{F}_k^{det}(\mathcal{M})$, let $u_{\mathcal{C}}:V_{\mathcal{M},k}\to\mathbb{R}$ be the corresponding instantiation of the induced pMC $\mathcal{D}_{\mathcal{M},k}$ such that $u_{\mathcal{C}}(\gamma_a^{q,z}):=\gamma(a|q,z)$ and $u_{\mathcal{C}}(\delta_{q'}^{q,z,a}):=\delta(q'|q,z,a)$. Noting that $\mathcal{D}_{\mathcal{M},k}[u_{\mathcal{C}}]$ is a stochastic process, we denote its sequence of states by $(S_{\mathcal{M},k,1}, S_{\mathcal{M},k,2}, \dots)$. Moreover, for a given state history $S^{t-1}_{\mathcal{M},k}$, we denote the one-step entropy of $\mathcal{D}_{\mathcal{M},k}[u_{\mathcal{C}}]$ by $H^{\mathcal{C}}(S_{\mathcal{M},k,t}|S^{t-1}_{\mathcal{M},k})$.

Proposition 1: For a given POMDP $\mathcal{M}$, a controller $\mathcal{C}\in\mathcal{F}_k^{det}(\mathcal{M})$, and constants $t\in\mathbb{N}$ and $k>0$, we have

$$H^{\mathcal{C}}(S_t|S^{t-1}) = H^{\mathcal{C}}(S_{\mathcal{M},k,t}|S^{t-1}_{\mathcal{M},k}). \tag{19}$$

Proposition 1 shows that the local (one-step) entropy that is gained in a POMDP under a deterministic k-FSC is equal to the local entropy that is gained in a pMC instantiated by the corresponding k-FSC. Then, together with the definition of the induced pMC, Proposition 1 implies that, for any $\mathcal{C}\in\mathcal{F}_k^{det}(\mathcal{M})$,

$$\mathcal{C} \in \arg\max_{\mathcal{C}\in\mathcal{F}_k^{det}(\mathcal{M})} \sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}}(S_t|S^{t-1}) \tag{20a}$$

$$\text{subject to: } \mathbb{E}^{\mathcal{C}}\Big[\sum_{t=1}^{\infty}\beta^{t-1}\mathcal{R}(S_t,A_t)\Big] \geq \Gamma \tag{20b}$$

if and only if

$$\mathcal{C} \in \arg\max_{\mathcal{C}\in\mathcal{F}_k^{det}(\mathcal{M})} \sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}}(S_{\mathcal{M},k,t}|S^{t-1}_{\mathcal{M},k}) \tag{21a}$$

$$\text{subject to: } \mathbb{E}^{\mathcal{C}}\Big[\sum_{t=1}^{\infty}\beta^{t-1}\mathcal{R}(S_{\mathcal{M},k,t},A_t)\Big] \geq \Gamma. \tag{21b}$$

Remark 2: The equivalence between the problems (20a)-(20b) and (21a)-(21b) does not hold if we consider the entropy maximization problem over the set $\mathcal{F}_k(\mathcal{M})$ instead of the set $\mathcal{F}_k^{det}(\mathcal{M})$. Note that for a POMDP $\mathcal{M}$ under a controller $\mathcal{C}\in\mathcal{F}_k(\mathcal{M})$, we have $|Succ(s)|\leq|S|$ for all $s\in S$, whereas on the corresponding induced pMC, it is possible to have $|Succ(\langle s,q\rangle)|>|S|$ due to stochastic memory transitions. Moreover, the entropy of a distribution with a support of size $|Succ(\langle s,q\rangle)|$ can be made larger than the entropy of a distribution with a support of size $|Succ(s)|$. Hence, if one performs the maximization over the set $\mathcal{F}_k(\mathcal{M})$, the maximum entropy of the induced pMC $\mathcal{D}_{\mathcal{M},k}$ is not necessarily equal to the maximum entropy of the POMDP $\mathcal{M}$.

Corollary 1: Let $G_1^\star$ and $G_2^\star$ be the optimal values of the problems given in (8a)-(8b) and (21a)-(21b), respectively. Then, we have $G_1^\star\geq G_2^\star$.

The above result follows from $\mathcal{F}_k^{det}(\mathcal{M})\subseteq\Pi(\mathcal{M})$ and implies that the maximum entropy of the pMC induced from FSCs with deterministic memory transitions is upper bounded by the maximum entropy of the corresponding POMDP.

C. Finite-State Controller Synthesis: Optimization Problem

We now present a nonlinear optimization problem to synthesize a deterministic k-FSC that maximizes the entropy of a POMDP over all deterministic k-FSCs.

Recall that for a POMDP $\mathcal{M}$ and a constant $k>0$, the induced pMC $\mathcal{D}_{\mathcal{M},k}$ represents all possible MCs that can be induced from $\mathcal{M}$ by a k-FSC. Moreover, Proposition 1 implies that the maximum entropy of $\mathcal{D}_{\mathcal{M},k}$ is equal to the maximum entropy of $\mathcal{M}$ *if one restricts attention to k-FSCs with deterministic memory transitions*. In what follows, we formulate an optimization problem to synthesize a well-defined instantiation u for the pMC $\mathcal{D}_{\mathcal{M},k}$ such that the entropy of the MC $\mathcal{D}_{\mathcal{M},k}[u]$ is maximized over all MCs $\mathcal{D}_{\mathcal{M},k}[u']$ for which $P^{u'}_{\mathcal{M},k}(\langle s',q'\rangle \mid \langle s,q\rangle)>0$ for a single $q'\in Q$.

To restrict the search space to FSCs with deterministic memory transitions, we introduce the following constraints

$$u(\delta^{q,z,a}_{q'}) \in \{0,1\} \;\text{ and }\; \sum_{z\in\mathcal{Z}}\sum_{a\in\mathcal{A}} u(\delta^{q,z,a}_{q'}) \in \{0,|\mathcal{Z}||\mathcal{A}|\}.$$

Intuitively, the above constraints enforce the FSC to transition to a single successor memory state regardless of the received observation and the taken action. We note that, for a given memory state pair (q,q'), the second integer constraint can be implemented as $|\mathcal{Z}||\mathcal{A}|$ equality constraints. Finally, the above constraints do not prevent the agent from randomizing its actions. The agent can still randomize its actions at a given state $s\in S$ by instantiating the parameters $\gamma_a^{q,z}$ appropriately.

For notational simplicity, let $\mathbf{s}$ denote an arbitrary state $\langle s,t\rangle\in S_{\mathcal{M},k}$. For a well-defined instantiation $\mathcal{D}_{\mathcal{M},k}[u]$ of induced pMC $\mathcal{D}_{\mathcal{M},k}$, let $L^u{:}S_{\mathcal{M},k}\to\mathbb{R}$ be the *local entropy* function such that

$$L^u(\mathbf{s}) := -\sum_{\mathbf{s}'\in S_{\mathcal{M},k}} P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})\log P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s}) \tag{22}$$

for all $\mathbf{s}\in S_{\mathcal{M},k}$. By slightly modifying the statement of Lemma 1 and defining variables $\nu\in\mathbb{R}^{|S_{\mathcal{M},k}|}$, it can be shown that the maximum entropy (21a) of $\mathcal{D}_{\mathcal{M},k}$ is the unique fixed-point of the system of equations

$$\nu(\mathbf{s}) = \max_{u\in\Upsilon}\left\{L^u(\mathbf{s}) + \beta\sum_{\mathbf{s}'\in S_{\mathcal{M},k}} P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})\nu(\mathbf{s}')\right\} \tag{23}$$

and equal to $\nu(\mathbf{s}_I){:=}\nu(s_{I,\mathcal{M},k})$. Hence, the maximum entropy (21a) of $\mathcal{D}_{\mathcal{M},k}$ can be computed by finding the maximum $\nu(\mathbf{s}_I)$ that satisfies

$$\nu(\mathbf{s}) \leq L^u(\mathbf{s}) + \beta\sum_{\mathbf{s}'\in S_{\mathcal{M},k}} P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})\nu(\mathbf{s}')\;\forall\mathbf{s}\in S_{\mathcal{M},k}.$$

Note that in the above inequality both $P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})$ and $\nu(\mathbf{s}')$ are variables. Hence, standard methods, e.g., value iteration, cannot be used to compute $\nu(\mathbf{s}_I)$; instead, one needs to solve a nonlinear optimization problem, which we present shortly, for the computation of $\nu(\mathbf{s}_I)$.

For the expected total reward constraint, let $\mathcal{R}^u{:}\mathcal{S}_{\mathcal{M},k}\to\mathbb{R}$ define the expected immediate rewards on $\mathcal{D}_{\mathcal{M},k}$ such that, for all $\mathbf{s}\in S_{\mathcal{M},k}$,

$$\mathcal{R}^u(\mathbf{s}) := \sum_{\mathbf{s}'\in S_{\mathcal{M},k}}\sum_{a\in\mathcal{A}} \overline{P}^u(\mathbf{s}'|\mathbf{s}',a)\mathcal{R}(\mathbf{s}',a) \tag{24}$$

where $\overline{P}^u{:}S_{\mathcal{M},k}\times\mathcal{A}\to\Delta(S_{\mathcal{M},k})$ is defined by replacing parameters $\gamma_a^{q,z}$ and $\delta^{q,z,a}_{q'}$ in (18) with their corresponding instantiations $u(\gamma_a^{q,z})$ and $u(\delta^{q,z,a}_{q'})$. Then, the problem in

(21a)-(21b) can be formulated as a nonlinear optimization problem (NLP) as follows:

$$\underset{\nu,u,\eta}{\text{maximize}} \qquad \nu(\mathbf{s}_I) \tag{25a}$$

subject to:

$$\nu(\mathbf{s}) \leq L^u(\mathbf{s}) + \beta \sum_{\mathbf{s}' \in S_{\mathcal{M},k}} P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})\nu(\mathbf{s}') \;\forall \mathbf{s} \in S_{\mathcal{M},k} \tag{25b}$$

$$\eta(\mathbf{s}) \leq \mathcal{R}^u(\mathbf{s}) + \beta \sum_{\mathbf{s}' \in S_{\mathcal{M},k}} P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})\eta(\mathbf{s}') \;\forall \mathbf{s} \in S_{\mathcal{M},k} \tag{25c}$$

$$\eta(\mathbf{s}_I) \geq \Gamma \tag{25d}$$

$$u(\delta^{q,z,a}_{q'}) \in \{0,1\}, \quad \sum_{q' \in Q} u(\delta^{q,z,a}_{q'}) = 1 \tag{25e}$$

$$0 \leq u(\gamma^{q,z}_a) \leq 1, \quad \sum_{a \in \mathcal{A}} u(\gamma^{q,z}_a) = 1 \tag{25f}$$

$$\sum_{z \in \mathcal{Z}} \sum_{a \in \mathcal{A}} u(\delta^{q,z,a}_{q'}) \in \{0, |\mathcal{Z}||\mathcal{A}|\}. \tag{25g}$$

In the above optimization problem, the variable $\eta(\mathbf{s})$ denotes the expected reward collected by starting from the state $\mathbf{s}\in S_{\mathcal{M},k}$. From the classical theory of Markov decision processes [27], it follows that the constraint (25d) ensures that a solution $u^\star$ to the above problem collects an expected total reward that exceeds the threshold Γ.

Recall that the transition function $P^u_{\mathcal{M},k}$ of the Markov chain $\mathcal{D}_{\mathcal{M},k}[u]$, which results from the instantiation u of the pMC $\mathcal{D}_{\mathcal{M},k}$, is given by $P^u_{\mathcal{M},k}(\mathbf{s}'|\mathbf{s})=\sum_{a\in\mathcal{A}} \overline{P}^u(\mathbf{s}'|\mathbf{s},a)$ where

$$\overline{P}^u(\mathbf{s}'|\mathbf{s},a) := \sum_{z \in \mathcal{Z}} \mathcal{O}_{s,z}\, \mathcal{P}_{s,a,s'}\, u(\gamma^{q,z}_a)\, u(\delta^{q,z,a}_{q'}), \tag{26}$$

$\mathbf{s}=\langle s,q\rangle$, and $\mathbf{s}'=\langle s',q'\rangle$. Therefore, the problem in (25a)-(25g) involves nonlinear constraints (25b) where three variables are multiplied with each other. Even though certain relaxation techniques, *e.g.*, McCormick envelopes [28], can be used to replace the constraints (25b) with specific bilinear constraints, finding an optimal solution to the resulting NLP remains as a challenge due to binary constraints (25e). One can employ branch-and-bound algorithms [29] to obtain a solution to (25a)-(25g), but unfortunately, such algorithms scale poorly with the size of the problem instances.

For practical purposes, instead of computing a global optimal solution, one can aim to obtain a local optimal solution to the problem in (25a)-(25g) *after setting the instantiation* $u(\delta^{q,z,a}_{q'})$ *of memory transitions to a constant*. In the next section, we provide a technique to obtain a local optimal solution to the problem (8a)-(8b) over all k-FSCs with a specific deterministic memory transition function.

D. Finite-State Controller Synthesis: A Solution Approach

In this section, we consider the entropy maximization problems over k-FSCs with a specific deterministic transition function and present an algorithm to synthesize a controller which locally maximizes the entropy of a given POMDP.

We first set the variables $u(\delta^{q,z,a}_{q'})$ in the problem (25a)-(25g) to constants such that they satisfy the constraint (25e). Note that this operation is equivalent to restricting the search space in (8a)-(8b) to k-FSCs with a specific deterministic transition function, where the transition function satisfies $\delta(q'|q,z,a)=u(\delta^{q,z,a}_{q'})$. The resulting optimization problem has decision variables $\nu(\mathbf{s})$, $\eta(\mathbf{s})$, and $u(\gamma^{q,z}_a)$, i.e., $u(\delta^{q,z,a}_{q'})$ is not a variable anymore. The resulting problem has bilinear constraints (25b) and (25c), and hence, it is not a convex optimization problem. However, we can obtain a local optimal solution to the resulting problem using a variant of the convex-concave-procedure (CCP) [30]. In particular, we employ *penalty CCP* which is introduced in [31] and used in the context of pMCs in [8]. The algorithm described below follows closely from the one proposed in [8].

Penalty CCP algorithm takes five inputs: a threshold constant $\epsilon>0$, an initial penalty constant $\tau_0>0$, a multiplication factor $\mu>1$, a maximum penalty constant $\tau_{\max}$, and initial estimates $\hat{\nu}_0(\mathbf{s})$, $\hat{\eta}_0(\mathbf{s})$, and $\hat{u}_0(\gamma^{q,z}_a)$ for the variables $\nu(\mathbf{s})$, $\eta(\mathbf{s})$, and $u(\gamma^{q,z}_a)$, respectively.

Let $\mathbf{v}$ denote an arbitrary tuple $(\mathbf{s}',q,z,a)\in S_{\mathcal{M},k}\times Q\times\mathcal{Z}\times\mathcal{A}$. For each $\mathbf{v}$, we introduce two new variables $\Phi_{\nu,\mathbf{v}}>0$ and $\Phi_{\eta,\mathbf{v}}>0$. Moreover, for each iteration $k\in\mathbb{Z}_+$ of the algorithm, we recursively define $\tau_{k+1}:=\min\{\mu\tau_k,\tau_{\max}\}$.

At each iteration $k\in\mathbb{Z}_+$ of the algorithm, we first convexify the constraints (25b) and (25c) (explained below). We then solve the resulting convex optimization problem by replacing the objective function (25a) with

$$\underset{\nu,u,\eta}{\text{maximize}} \qquad \nu(\mathbf{s}_I) - \tau_k \sum_{\mathbf{v}} (\Phi_{\eta,\mathbf{v}} + \Phi_{\nu,\mathbf{v}}).$$

If the optimal value Val_k of this problem satisfies $|Val_k - Val_{k-1}|<\epsilon$, and the optimal solution satisfies $\sum_{\mathbf{v}}(\Phi_{\eta,\mathbf{v}} + \Phi_{\nu,\mathbf{v}})=0$, we terminate the algorithm and conclude that the solution is local optimal [31]. Otherwise, we set the optimal decision variables $\nu^\star(\mathbf{s})$, $\eta^\star(\mathbf{s})$, and $u^\star(\gamma^{q,z}_a)$ for the current iteration as the estimates $\hat{\nu}_{k+1}(\mathbf{s})$, $\hat{\eta}_{k+1}(\mathbf{s})$, and $\hat{u}_{k+1}(\gamma^{q,z}_a)$ for the successive iteration, and solve the resulting optimization problem. The procedure explained above has no theoretical convergence guarantees to a feasible solution [31], i.e., a solution that satisfies $\sum_{\mathbf{v}}(\Phi_{\eta,\mathbf{v}}+\Phi_{\nu,\mathbf{v}})=0$. However, in practice, we observe that it usually converges to a feasible solution.

In what follows, we explain the convexification procedure for the constraint (25b); the convexification of (25c) is performed by following the same procedure. Note that the last term on the right hand side of (25b) is the summation of bilinear terms $c(s,s',a,z,u)\nu(\mathbf{s}')u(\gamma^{q,z}_a)$ where $c(s,s',a,z,u)$ is a constant such that

$$c(s,s',a,z,u) := \mathcal{O}_{s,z}\, \mathcal{P}_{s,a,s'}\, u(\delta^{q,z,a}_{q'}).$$

With some abuse of notation, we denote $c(s,s',a,z,u)$ by c. As explained in [8], a bilinear function $f(x,y)=2Cxy$, where C is a constant, can be written as a difference of convex functions $f(x,y)=f_1(x,y)-f_2(x,y)$ where $f_1(x,y)=C(x+y)^2$ and $f_2(x,y)=C(x^2+y^2)$. Since we have a constraint of the form $0\leq L^u(\mathbf{s})+f(x,y)$ in (25b), we linearize the function $f_1(x,y)$ around the point $\hat{\nu}_k(\mathbf{s})$ and $\hat{u}_k(\gamma^{q,z}_a)$. Specifically, at iteration $k\in\mathbb{Z}_+$, we replace each bilinear term $c\nu(\mathbf{s}')u(\gamma^{q,z}_a)$

in (25b) with

$$c\Big(\hat{\nu}_k(\mathbf{s}')+\hat{u}_k(\gamma_a^{q,z})\Big)^2-\frac{c}{2}\Big((\nu(\mathbf{s}'))^2+(u(\gamma_a^{q,z}))^2\Big)$$
$$+c\Big(\hat{\nu}_k(\mathbf{s}')+\hat{u}_k(\gamma_a^{q,z})\Big)\Big(\nu(\mathbf{s}')-\hat{\nu}_k(\mathbf{s}')\Big)$$
$$+c\Big(\hat{\nu}_k(\mathbf{s}')+\hat{u}_k(\gamma_a^{q,z})\Big)\Big(u(\gamma_a^{q,z})-\hat{u}_k(\gamma_a^{q,z})\Big)+\Phi_{\nu,\mathbf{v}}.$$

Note that the above expression is concave in the variables $\nu(\mathbf{s}')$ and $u(\gamma_a^{q,z})$. Therefore, the problem resulting from the replacement of the bilinear terms with the above expression is a convex optimization problem.

E. Finite-State Controller Synthesis: A Monotonocity Result

In the previous section, we presented an algorithm to solve the problem in (25a)-(25g), which requires one to set the variables $u(\delta_{q'}^{q,z,a})$ to constants that satisfy the constraint (25c). As discussed earlier, the choice of these constants establishes the memory transition structure of the k-FSCs over which the entropy maximization is performed. In this section, we present a particular memory transition function which has a monotonocity property. That is, when this memory transition function is used, by increasing the number of memory states, one can only increase the optimal value of the optimization problem (8a)-(8b).

For a POMDP $\mathcal{M}$, consider a k-FSC $\mathcal{C}=(Q,q_1,\gamma,\delta)$ with the memory transition function $\overline{\delta}:Q\times\mathcal{Z}\times\mathcal{A}\to\Delta(Q)$ such that

$$\begin{cases}\overline{\delta}(q_{i+1}|q_i,z,a)=1 & \forall z\in\mathcal{Z},a\in\mathcal{A},1\leq i<k\\ \overline{\delta}(q_k|q_k,z,a)=1 & \forall z\in\mathcal{Z},a\in\mathcal{A}\\ \overline{\delta}(q_i|q_j,z,a)=0 & \text{otherwise.}\end{cases}\tag{27}$$

Let $\overline{\mathcal{F}}_k(\mathcal{M})\subset\mathcal{F}_k^{det}(\mathcal{M})$ be the set of k-FSCs whose memory transition function is given in (27). Additionally, let

$$E_{k,\max}:=\max_{\mathcal{C}\in\overline{\mathcal{F}}_k(\mathcal{M})}\ \sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}}(S_{\mathcal{M},k,t}|S_{\mathcal{M},k}^{t-1}).$$

Lemma 2: For all $j\leq k$, we have $E_{j,\max}\leq E_{k,\max}$.

Using the above result, we can obtain a practical algorithm to synthesize an entropy-maximizing controller as follows. First, fix the number of memory states to $k=1$. Then, by setting $u(\delta_{q'}^{q,z,a})=\overline{\delta}(q'|q,z,a)$, find a local optimal solution to the problem in (25a)-(25g). Next, increase the number of memory states to $k=2$, solve the resulting problem, and compare the optimal value of the problem with the previous result. Repeat this procedure until there is no significant change in the optimal value of the optimization problem. We note that since the algorithm described in the previous section computes a local optimal solution for the problem in (25a)-(25g), the procedure described above has no theoretical guarantees to yield improving solutions for the problem (8a)-(8b). However, we observe that, in practice, the described procedure works considerably well.

VI. Numerical Examples

We now provide several numerical examples to demonstrate the relation between the infinite-horizon maximum entropy of a POMDP with the threshold Γ on the expected total reward, as well as with the number of memory states in the FSC. Furthermore, in the finite horizon case, we provide an example to demonstrate the relation between the maximum entropy of a POMDP with respect to the number of time steps in the finite horizon, as well as an example studying the relation between the maximum entropy of a POMDP to the expected total reward for a fixed number of time steps. Finally, in the discounted case, we provide an example studying the relation between the maximum entropy of a POMDP and the discount factor. We use the MOSEK [32] solver with the CVX [33] interface to solve the convex optimization problems obtained from the convex-concave procedure. To improve the approximation of exponential cone constraints, we use the CVXQUAD [34] package.

A. The Relationship Between the Maximum Entropy and the Expected Total Reward Threshold

We first consider a POMDP with 6 states shown in Fig. 1. There is a single observation $\mathcal{Z}=\{z_1\}$, yielding $\mathcal{O}_{s,z_1}=1$ for all states s. We use a deterministic 2-FSC whose memory transition function δ is given in (27). Because there is only one observation, the synthesized controller is an open-loop controller. We suppose that the agent aims to reach state s_4 and encode this objective by defining a reward function $\mathcal{R}$ such that $\mathcal{R}(s_2,a_1)=\mathcal{R}(s_3,a_1)=1$ and $\mathcal{R}(s,a)=0$ otherwise.

We investigate the effect of the threshold Γ in (8b) on the maximum entropy by synthesizing controllers for values between $\Gamma=0.5$ and $\Gamma=1$. For each value of Γ, we use the memory transition function (27) and solve the optimization problem given in Section V-C 10 times by randomly initializing the convex-concave procedure. For each Γ, we pick the best result of the 10 trials, and plot the maximum entropy of the stochastic process induced by the synthesized controllers in Fig. 2. For comparison, we synthesize controllers by solving a feasibility problem given in [8]. We obtain the feasibility problem from (25a)-(25g) by removing the entropy constraint (25b) and replacing the objective function (25a) with a constant value.

The proposed approach yields the globally optimal controller by attaining a tight bound on Γ. The global optimality of the controller is evident in Fig. 2, as the entropy of the proposed approach exactly matches that of the underlying MDP for each value of Γ. Because the feasibility problem only seeks to find a feasible instantiation of the parameters that satisfy the expected total reward constraints (25c)-(25d), the entropy of the induced stochastic processes is less than the maximum attainable entropy.

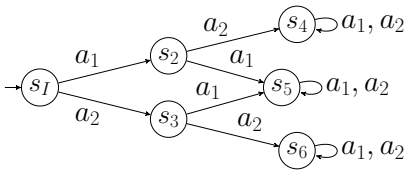


Fig. 1: POMDP illustrating the relation between the maximum entropy and the expected total reward Γ.

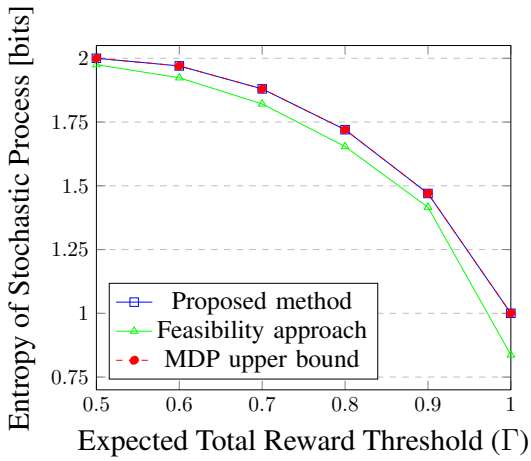


Fig. 2: The trade-off between the maximum entropy and the expected total rewards.

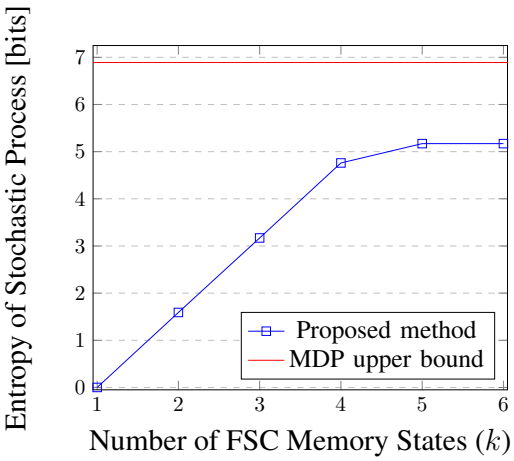


Fig. 4: Comparison between the maximum entropy of the induced stochastic process for varying values of k.

B. The Relationship Between the Maximum Entropy and the Number of Memory States

We now consider the 15-state POMDP shown in Fig. 3. As in the previous example, there is only a single observation $\mathcal{Z}=\{z_1\}$ yielding $\mathcal{O}_{s,z_1}=1$ for all states s. We suppose that the agent aims to reach s_{14} with probability 1. To encode this objective, we set $\Gamma=1$ with $\mathcal{R}(s_{10},a_2)=1$, $\mathcal{R}(s_{11},a_2)=1$, $\mathcal{R}(s_{12},a_2)=1$, and $\mathcal{R}(s,a)=0$ otherwise.

We study the relation between the number of memory states and the maximum entropy of the induced pMC by synthesizing controllers for $k=1,\ldots,6$ memory states. We again run the optimization problem given in Section V-C 10 times while randomly initializing the convex-concave procedure. In Fig. 4, we plot the maximum entropy of the induced stochastic process for each value of k. Furthermore, Fig. 5 shows the entropy-maximizing controller for the POMDP, where edge weights correspond to the probability of action selection.

From Fig. 4, we see that the 1-FSC achieves a maximum entropy of 0. A 1-FSC selecting any action besides a_2 cannot reach state s_{14} while collecting an expected total reward of 1. An additional memory state allows the agent to randomize its action selection for one more time step. After 5 memory states, however, additional memory states do not affect the maximum entropy of the induced stochastic process.

Any controller with at least 5 memory states achieves an optimal action distribution shown in Fig. 5. Unlike the previous example, a gap between the maximum entropy of the MDP and that of the induced pMC remains. The maximum entropy of the POMDP must lie within this gap. This example

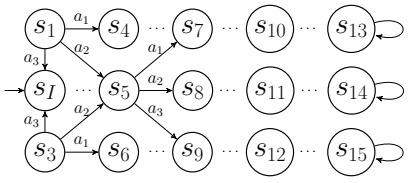


Fig. 3: POMDP illustrating the relation between the maximum entropy and the number of memory states in the FSC.

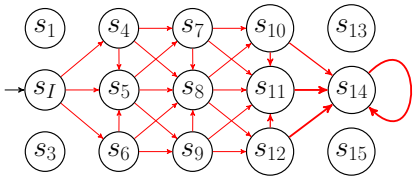


Fig. 5: Trajectories of synthesized entropy maximizing controller. Edge thicknesses indicate the transition probabilities.

demonstrates the monotonicity of the maximum entropy with the number of states in the controller.

C. The Relationship Between the Maximum Entropy and the Time Horizon

In this example, we examine the relation between the maximum entropy of a POMDP and the number of time steps in the finite-horizon problem. We consider the POMDP whose state-space is given by the 4×4 gridworld shown in Fig. 6 (left). The yellow state is the unique initial state of the agent, the red states are error states to be avoided, and the blue state is the target state. In each state, the agent can select one of four possible actions: move left, move right, move up, or move down. Whenever the agent selects an action, it transitions to its intended state with probability $0.95-\epsilon/3$, slips to the left or to the right with probability $0.025-\epsilon/3$, and slips backwards with probability ϵ. In this example, we use a value of $\epsilon=0.005$. If the agent were to transition off the gridworld, it instead remains in its current state.

There are nine possible observations that the agent can make: *error (target) state to the left, error (target) state to the right, error (target) state above, error (target) state below*, and *"no observation"*. The observations in each state are deterministic; e.g., if the agent were in the state to the left of the target state, then it makes the observation *"target state to the right"* with probability 1.

We consider how the variation of the finite time horizon T affects the maximum entropy of the POMDP. We use values of T ranging from $T=16$ to $T=30$ time steps.

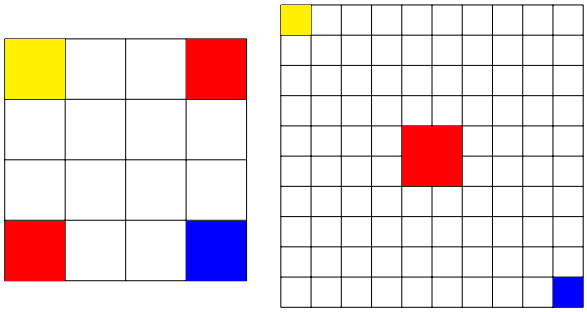


Fig. 6: (Left) Gridworld considered in Section VI-C and Section VI-D. (Right) Gridworld considered in Section VI-E.

The minimum expected total reward threshold is set to $\Gamma{=}0.9$, which we encode by setting the values of the reward function $\mathcal{R}$ as $\mathcal{R}((3,1),\text{right}){=}\mathcal{R}((4,2),\text{down}){=}0.95{-}\epsilon/3$, $\mathcal{R}((3,1),\text{up}){=}\mathcal{R}((4,2),\text{left}){=}0.025{-}\epsilon/3$, and $\mathcal{R}((3,1),$ left$)$ $=\mathcal{R}((4,2),\text{up}){=}\epsilon$, and 0 otherwise. We use a deterministic 1-FSC and run the optimization problem given in Section V-C a total of 5 times for each value of T. In Fig. 7, we plot the maximum entropy of the induced stochastic process observed over the set of trials as a function of the finite time horizon T.

From Fig. 7, we see that the maximum entropy of the POMDP increases with the size of the finite time horizon. Intuitively, a longer time horizon allows the agent to more uniformly distribute its actions. With a shorter time horizon, the agent must allocate more probability mass towards selecting actions that lead it down and to the right in order to reach the target state within the time horizon. For longer time horizons, the agent is able to more uniformly distribute its actions, yielding a higher maximum entropy.

D. The Relationship Between the Maximum Entropy and the Finite-Horizon Expected Total Reward

In this example, we again consider the $4{\times}4$ gridworld shown in Fig. 6 (left) with identical transition, observation, and reward functions as those used in Section VI-C. We now suppose that the agent must collect an expected total reward above some minimum threshold Γ within a finite time horizon of $T{=}16$ time steps. We use values of Γ varying from $\Gamma{=}0.5$ to $\Gamma{=}0.975$. Using a deterministic 1-FSC, we run the optimization problem given in Section V-C a total of 5 times for each value of Γ and store the largest value of the maximum entropy observed. Fig. 8 plots the relation between the maximum entropy of the induced stochastic process as a function of the lower bound Γ on the total expected reward.

Lower values of Γ allow the agent to more uniformly distribute its probabilities for action selection, as the agent need not reach the blue state with high probability. For increasingly large values of Γ, the agent must select its actions such that it drives itself towards the blue state, reducing the maximum entropy of the induced stochastic process. As the value of Γ approaches 1, the maximum entropy of the induced stochastic process begins to level off. For large values of Γ, and for $\Gamma{=}0.95$ and $\Gamma{=}0.975$ in particular, the synthesized controllers only slightly deviate from one another. Because the synthesized policies are nearly identical, the resulting maximum entropies of their respective induced stochastic processes are likewise nearly identical.

E. The Relationship Between the Infinite-Horizon Maximum Entropy and the Discount Factor

In the final example, we study the relation between the maximum entropy of the POMDP and the value of the discount factor, β. We consider the POMDP with the state space given by the $10{\times}10$ gridworld shown in Fig. 6 (right). As in the previous examples, the yellow state is the unique initial state of the agent, the red states are error states to be avoided, and the blue state is the target state. The construction of the transition, observation, and reward functions is identical to those of the previous two examples. In this example, we set the minimum expected total reward threshold as $\Gamma{=}0.9$.

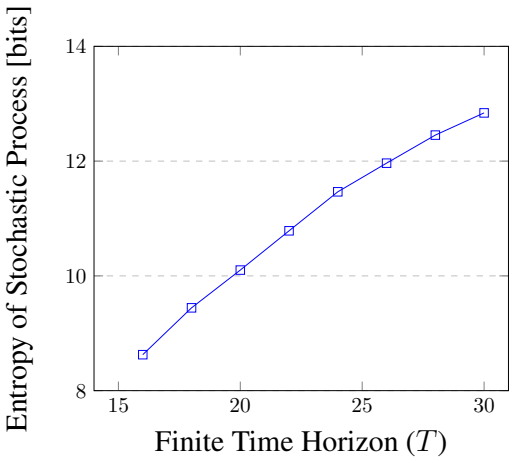


Fig. 7: Relation between the maximum entropy of the induced stochastic process and the time horizon T.

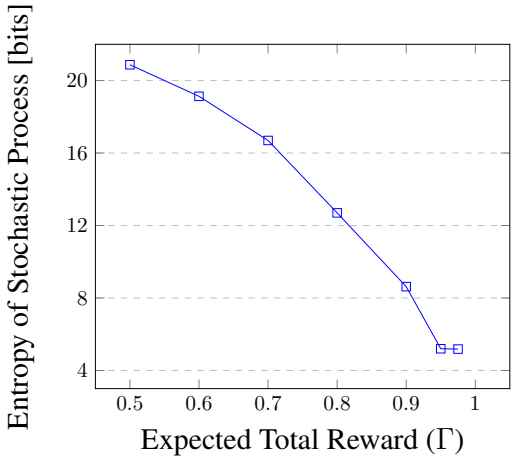


Fig. 8: Relation between the maximum entropy of the induced stochastic process and the expected total reward Γ for finite time horizon, $T{=}16$.

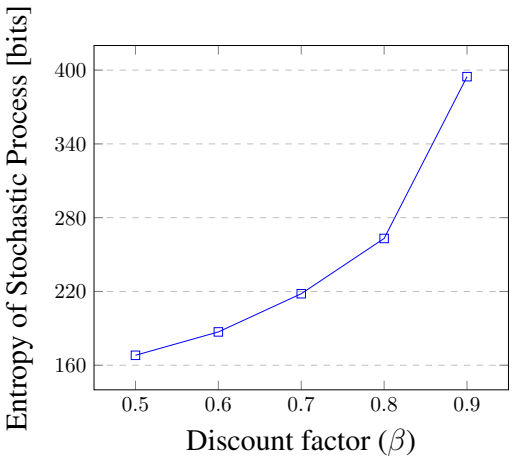


Fig. 9: Relation between the maximum entropy of the induced stochastic process and the discount factor, β.

The agent operates over an infinite horizon where its reward function for the entropy is discounted by varying values of β ranging from β=0.5 to β=0.9. For each value of β, we use a deterministic 1-FSC and run the optimization problem given in Section V-C a total of 5 times for each value of β. We subsequently compute the undiscounted entropy of the induced stochastic process for each synthesized controller. We compute the undiscounted entropy since this value accurately reflects the entropy of the induced stochastic process on the *physical* state space as viewed by an observer.

Fig. 9 plots the maximum observed undiscounted entropies as a function of the discount factor β. For low values of β, the agent is unconcerned with the entropy of its future trajectories and focuses its action selection towards reaching the blue state. As the value of β increases, the agent begins to more heavily weight the expected entropy of its future trajectories. For these values of β, the agent selects its actions to avoid the red and blue absorbing states, thereby allowing itself to remain operating in the environment for a longer period of time, further increasing the expected entropy of its future trajectories. Through this example, we see that by increasing the discount factor, the agent can achieve a higher maximum entropy, albeit at the cost of requiring an increasingly large time horizon to collect an expected total reward of at least Γ=0.9.

VII. Conclusions

We studied the synthesis of a controller which, from a given POMDP, induces a stochastic process with maximum entropy among the ones whose realizations accumulate a certain level of expected reward. Since the entropy maximization objective is considerably different than the traditionally used expected reward maximization objective, we first showed that the maximum entropy of a POMDP is upper bounded by the maximum entropy of its corresponding fully observable counterpart. Then, by restricting our attention to FSCs with deterministic memory transitions, we recast the entropy maximization problem as a so-called parameter synthesis problem for pMCs. We present a nonlinear optimization problem for the synthesis of an FSC that maximizes the entropy of a POMDP over all FSCs with the same number of memory states and deterministic memory transitions. Considering the intractability of finding a global optimal solution to the presented optimization problem, we proposed a convex-concave procedure approach to obtain a local optimal solution after setting the memory transition of FSCs to a fixed structure.

Even though finding a solution to the constrained entropy optimization problem is at least PSPACE-hard due to expected reward constraints, the computational complexity of the unconstrained entropy maximization problem is still an open problem. Additionally, developing the computational methods to synthesize a controller that maximizes the entropy of a POMDP over all FSCs with the same number of memory states may be a fruitful research direction.

References

[1] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley & Sons, Inc., 2006.

[2] F. Biondi, A. Legay, B. F. Nielsen, and A. Wasowski, "Maximizing entropy over Markov processes," *Journal of Logical and Algebraic Methods in Programming*, vol. 83, no. 5, pp. 384 – 399, 2014.

[3] Y. Savas, M. Ornik, M. Cubuktepe, M. O. Karabag, and U. Topcu, "Entropy maximization for Markov decision processes under temporal logic constraints," *IEEE Transactions on Automatic Control*, 2019.

[4] 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.

[5] K. Chatterjee, M. Chmelík, and M. Tracol, "What is decidable about partially observable Markov decision processes with ω-regular objectives," *Journal of Computer and System Sciences*, vol. 82, no. 5, pp. 878–911, 2016.

[6] N. Meuleau, K.-E. Kim, L. P. Kaelbling, and A. R. Cassandra, "Solving POMDPs by searching the space of finite policies," in *Conference on Uncertainty in artificial intelligence*, 1999, pp. 417–426.

[7] P. Poupart and C. Boutilier, "Bounded finite-state controllers," in *Advances in Neural Information Processing Systems*, 2004, pp. 823–830.

[8] M. Cubuktepe, N. Jansen, S. Junges, J.-P. Katoen, and U. Topcu, "Synthesis in pMDPs: A tale of 1001 parameters," in *Automated Technology for Verification and Analysis*, 2018, pp. 160–176.

[9] L. Hutschenreiter, C. Baier, and J. Klein, "Parametric Markov chains: PCTL complexity and fraction-free gaussian elimination," *arXiv preprint arXiv:1709.02093*, 2017.

[10] T. Akamatsu, "Cyclic flows, Markov process and stochastic traffic assignment," *Transportation Research Part B: Methodological*, vol. 30, no. 5, pp. 369–386, 1996.

[11] M. Hibbard, Y. Savas, B. Wu, T. Tanaka, and U. Topcu, "Unpredictable planning under partial observability," in *Conference on Decision and Control*, 2019.

[12] C. Kreucher, K. Kastella, and A. O. Hero Iii, "Sensor management using an active sensing approach," *Signal Processing*, vol. 85, no. 3, pp. 607–624, 2005.

[13] N. Roy, G. Gordon, and S. Thrun, "Finding approximate POMDP solutions through belief compression," *Journal of artificial intelligence research*, vol. 23, pp. 1–40, 2005.

[14] R. Eidenberger and J. Scharinger, "Active perception and scene modeling by planning with probabilistic 6D object poses," in *International Conference on Intelligent Robots and Systems*, 2010, pp. 1036–1043.

[15] T. Haarnoja, V. Pong, A. Zhou, M. Dalal, P. Abbeel, and S. Levine, "Composable deep reinforcement learning for robotic manipulation," in *International Conference on Robotics and Automation*, 2018, pp. 6244–6251.

[16] T. Haarnoja, H. Tang, P. Abbeel, and S. Levine, "Reinforcement learning with deep energy-based policies," in *International Conference on Machine Learning*, 2017, pp. 1352–1361.

[17] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, 2008, pp. 1433–1438.

[18] X. Zhang, B. Wu, and H. Lin, "Learning based supervisor synthesis of POMDP for PCTL specifications," in *Conference on Decision and Control*, 2015, pp. 7470–7475.

[19] J. K. Pajarinen and J. Peltonen, "Periodic finite state controllers for efficient POMDP and DEC-POMDP planning," in *Advances in Neural Information Processing Systems*, 2011, pp. 2636–2644.
[20] E. A. Hansen, "Solving POMDPs by searching in policy space," in *Conference on Uncertainty in artificial intelligence*, 1998, pp. 211–219.
[21] N. Meuleau, L. Peshkin, K.-E. Kim, and L. P. Kaelbling, "Learning finite-state controllers for partially observable environments," in *Conference on Uncertainty in Artificial Intelligence*, 1999, pp. 427–436.
[22] C. Amato, D. S. Bernstein, and S. Zilberstein, "Optimizing fixed-size stochastic controllers for POMDPs and decentralized POMDPs," *Autonomous Agents and Multi-Agent Systems*, pp. 293–320, 2010.
[23] S. Junges, N. Jansen, R. Wimmer, T. Quatmann, L. Winterer, J. Katoen, and B. Becker, "Finite-state controllers of POMDPs using parameter synthesis," in *Conference on Uncertainty in Artifical Intelligence*, 2018, pp. 519–529.
[24] F. Biondi, *Markovian Processes for Quantitative Information Leakage*. PhD thesis, IT University of Copenhagen, 2014.
[25] K. Astrom, "Optimal control of Markov processes with incomplete state information," *Journal of Mathematical Analysis and Applications*, vol. 10, no. 1, pp. 174 – 205, 1965.
[26] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of Markov decision processes." *Mathematics of Operations Research*, 1987.
[27] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
[28] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems," *Mathematical Programming*, 1976.
[29] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.
[30] A. L. Yuille and A. Rangarajan, "The concave-convex procedure (cccp)," in *Advances in Neural Information Processing Systems*, 2002, pp. 1033–1040.
[31] T. Lipp and S. Boyd, "Variations and extension of the convex-concave procedure," *Optimization and Engineering*, vol. 17, no. 2, pp. 263–287, 2016.
[32] M. ApS, *MOSEK Optimizer API for Python. Version 8.1.*, 2019. [Online]. Available: https://docs.mosek.com/8.1/pythonapi/index.html
[33] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.
[34] H. Fawzi, J. Saunderson, and P. A. Parrilo, "Semidefinite approximations of the matrix logarithm," *Foundations of Computational Mathematics*, 2018, package cvxquad at https://github.com/hfawzi/cvxquad.

APPENDIX A

In this appendix, we provide proofs for all theoretical results presented in the paper.

Proof of Lemma 1. We prove the claim by strong induction on t. For the base case, we have

$$\begin{aligned}
\mathcal{V}^{\pi}_{T-1,T}(s^{T-1}) &= H^{\pi}(S_T|S_{T-1}, S^{T-1}=s^{T-1}) \\
&\quad + H^{\pi}(S_{T+1}|S_T, S^{T-1}=s^{T-1}) && \text{(28a)}\\
&= H^{\pi}(S_T|S^{T-1}=s^{T-1}) \\
&\quad + H^{\pi}(S_{T+1}|S_T, S^{T-1}=s^{T-1}) && \text{(28b)}\\
&= H^{\pi}(S_T|S^{T-1}=s^{T-1}) \\
&\quad + \sum_{s^T\in\mathcal{SH}^T} Pr^{\pi}(s^T|s^{T-1}) H^{\pi}(S_{T+1}|S^T=s^T) && \text{(28c)}\\
&= H^{\pi}(S_T|S^{T-1}=s^{T-1}) \\
&\quad + \sum_{s^T\in\mathcal{SH}^T} Pr^{\pi}(s^T|s^{T-1}) \mathcal{V}^{\pi}_{T,T}(s^T). && \text{(28d)}
\end{aligned}$$

where (28b) follows from (28a) by the fact that S_{T-1} is a component of S^{T-1}. By the law of total probability and the definition of the state history, we obtain (28c) from (28b). Lastly, (28d) holds by the definition of the value function defined in (13). We now assume that the equality in (13) holds for time steps $T-2, T-3, \ldots, t+1$, and show that the equality holds for t.

$$\begin{aligned}
\mathcal{V}^{\pi}_{t,T}(s^t) &= \sum_{k=t}^{T} H^{\pi}(S_{k+1}|S^k_t, S^t=s^t) && \text{(29a)}\\
&= H^{\pi}(S_{t+1}|S^t=s^t) \\
&\quad + \sum_{k=t+1}^{T} H^{\pi}(S_{k+1}|S^k_{t+1}, S^t=s^t) && \text{(29b)}\\
&= H^{\pi}(S_{t+1}|S^t=s^t) + \sum_{s^{t+1}\in\mathcal{SH}^t} \sum_{k=t+1}^{T} \ldots \\
&\quad Pr^{\pi}(s^{t+1}|s^t) H^{\pi}(S_{k+1}|S^k_t, S^{t+1}=s^{t+1}) && \text{(29c)}\\
&= H^{\pi}(S_{t+1}|S^t=s^t) \\
&\quad + \sum_{s^{t+1}\in\mathcal{SH}^{t+1}} Pr^{\pi}(s^{t+1}|s^t) \mathcal{V}^{\pi}_{t+1,T}(s^{t+1}). && \text{(29d)}
\end{aligned}$$

As in the base case, (29b) follows from (29a) by the fact that S_t is a component of s^t. We then obtain (29c) from (29b) by the law of total probability and the definition of the state history. Lastly, (29d) holds by the definition of the value function defined in (13). The equality holds for a general t, completing the induction. We may thus write the total expected entropy in this recursive form.□

Proof of Theorem 1. We prove the claim by strong induction on t. Denote the value function for $\pi \in \Pi(\mathcal{M})$ as $\mathcal{V}^{\pi}_{t,T}(s^t)$ and the value function for $\pi' \in \Pi(\mathcal{M}_{fo})$ constructed according to (10) as $\mathcal{V}^{\pi'}_{t,T}(s^t)$, respectively. Starting with the base case $t = T$, we have

$$\begin{aligned}
\mathcal{V}^{\pi}_{T,T}(s^T) &= H^{\pi}(S_{T+1}|S^T=s^T) && \text{(30a)}\\
&= H^{\pi'}(S_{T+1}|S^T=s^T) && \text{(30b)}
\end{aligned}$$

by definition of π'. In particular, the equality in (30b) follows from the fact that we can construct an equivalent history-dependent controller on the underlying MDP that achieves the same transition probabilities for any observation-based controller. Taking the supremum of left hand side over $\Pi(\mathcal{M})$ and right hand side over $\Pi(\mathcal{M}_{fo})$, and noting that $\Pi(\mathcal{M}) \subseteq \Pi(\mathcal{M}_{fo})$, we have

$$\begin{aligned}
\sup_{\pi\in\Pi(\mathcal{M})} \mathcal{V}^{\pi}_{T,T}(s^T) &\leq \sup_{\pi'\in\Pi(\mathcal{M}_{fo})} H^{\pi'}(S_{T+1}|S^T=s^T) && \text{(31a)}\\
&= \mathcal{V}^{\pi'}_{T,T}(s^T). && \text{(31b)}
\end{aligned}$$

Now assume that the inequality holds for time steps $T-1, ..., t+1$. We show that it also holds for t as follows. Note first that

$$\begin{aligned}
\mathcal{V}^{\pi}_{t,T}(s^t) &= H^{\pi}(S_{t+1}|S^t=s^t) \\
&\quad + \sum_{s^{t+1}\in\mathcal{SH}^{t+1}} Pr^{\pi}(s^{t+1}|s^t) \mathcal{V}^{\pi}_{t+1,T}(s^{t+1}) && \text{(32a)}\\
&\leq H^{\pi}(S_{t+1}|S^t=s^t) \\
&\quad + \sum_{s^{t+1}\in\mathcal{SH}^{t+1}} Pr^{\pi}(s^{t+1}|s^t) \mathcal{V}^{\pi'}_{t+1,T}(s^{t+1}) && \text{(32b)}\\
&= H^{\pi'}(S_{t+1}|S^t=s^t) \\
&\quad + \sum_{s^{t+1}\in\mathcal{SH}^{t+1}} Pr^{\pi'}(s^{t+1}|s^t) \mathcal{V}^{\pi'}_{t+1,T}(s^{t+1}). && \text{(32c)}
\end{aligned}$$

By Lemma 1, we can write the value function recursively in (32a). The equality in (32b) then follows by the induction hypothesis. By (10), we can construct an equivalent controller on the underlying MDP that has the same transition probabilities. Doing so yields (32c). Then, we have

$$\sup_{\pi\in\Pi(\mathcal{M})} \mathcal{V}^{\pi}_{t,T}(s^t) \leq \sup_{\pi'\in\Pi(\mathcal{M}_{fo})} H^{\pi'}(S_{t+1}|S^t=s^t) + \sum_{s^{t+1}\in\mathcal{SH}^{t+1}} Pr^{\pi'}(s^{t+1}|s^t)\mathcal{V}^{\pi'}_{t+1,T}(s^{t+1}) \tag{33a}$$

$$= \mathcal{V}^{\pi'}_{t,T}(s^t). \tag{33b}$$

where inequality in (33a) is due to the fact that $\Pi(\mathcal{M})\subseteq\Pi(\mathcal{M}_{fo})$ and (33b) follows by the definition of the value function in (13). Thus the induction holds for t. Since the claim holds for all t, we have $\mathcal{V}^{\pi}_{1,T}(s_I)\leq\mathcal{V}^{\pi'}_{1,T}(s_I)$, which concludes the proof. □

Proof of Proposition 1: The result follows from the fact that the controller $\mathcal{C}$ only allows deterministic memory transitions. Note that, we have

$$H^{\mathcal{C}}(S_t|S^{t-1}) = \sum_{s^t\in\mathcal{SH}^t} Pr^{\mathcal{C}}(s_t,s^{t-1})\log Pr^{\mathcal{C}}(s_t|s^{t-1}). \tag{34}$$

Let q_t be the current memory state and q^{t-1} be the history of memory states. By the law of total probability, we have

$$Pr^{\mathcal{C}}(s_t|s^{t-1}) = \sum_{q^t} Pr^{\mathcal{C}}(s_t,q_t|q^{t-1},s^{t-1})Pr^{\mathcal{C}}(q^{t-1}|s^{t-1}).$$

Since the memory transitions are deterministic under $\mathcal{C}\in\mathcal{F}^{det}_k(\mathcal{M})$, by recursively expanding the right hand side of the above equality, it can be observed that $Pr^{\mathcal{C}}(q^{t-1}|s^{t-1})=1$ for a given state history realization s^{t-1}. Then, the result follows from the definition of states $S_{\mathcal{M},k}$. □

Proof of Lemma 2. We prove the claim by induction on the number of memory states k. We start with the base case $n=1$. Consider an instantiated pMC $\mathcal{D}_{\mathcal{M},1}[u_{\mathcal{C}}]$ for which there exists a corresponding deterministic 1-FSC $\mathcal{C}\in\bar{\mathcal{F}}_1(\mathcal{M})$ whose decision function γ satisfies $\gamma(a|q_1,z)=u_{\mathcal{C}}(\gamma_a^{q_1,z})$. Now, construct a deterministic 2-FSC $\mathcal{C}'$ whose decision function γ' satisfies $\gamma'(a|q_1,z)=\gamma'(a|q_2,z)=u_{\mathcal{C}}(\gamma_a^{q_1,z})$. Then, since the memory transitions of both FSCs satisfy (27), there is a one to one correspondence between the state histories of $\mathcal{D}_{\mathcal{M},1}[u_{\mathcal{C}}]$ and $\mathcal{D}_{\mathcal{M},2}[u_{\mathcal{C}'}]$. Using Lemma 1, it can be shown that

$$\sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}}(S_{\mathcal{M},1,t}|S^{t-1}_{\mathcal{M},1}) = \sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}'}(S_{\mathcal{M},2,t}|S^{t-1}_{\mathcal{M},2}).$$

Since we choose $\mathcal{C}$ arbitrarily, the maximum entropy of $\mathcal{D}_{\mathcal{M},2}$ cannot be lower than that of $\mathcal{D}_{\mathcal{M},1}$, i.e.,

$$E_{1,\max}\leq E_{2,\max}. \tag{35}$$

We assume that the claim holds for $n=1,2,...,k-1$, and show that it also holds for $k=n$. Consider an instantiated pMC $\mathcal{D}_{\mathcal{M},k-1}[u_{\mathcal{C}}]$ for which there exists a corresponding deterministic $(k-1)$-FSC $\mathcal{C}\in\bar{\mathcal{F}}_{k-1}(\mathcal{M})$ whose decision function γ satisfies $\gamma(a|q_i,z)=u_{\mathcal{C}}(\gamma_a^{q_i,z})$ for $i=1,\dots,k-1$. Then, we can construct an k-FSC $\mathcal{C}'$ whose decision function γ' satisfies $\gamma'(a|q_i,z) := \gamma(a|q_i,z)$ for $i=1,\dots,k-1$, and $\gamma'(a|q_k,z) := \gamma(a|q_{k-1},z)$. Since the memory transitions of both FSCs satisfy (27), there is a one to one correspondence between the state histories of $\mathcal{D}_{\mathcal{M},k-1}[u_{\mathcal{C}}]$ and $\mathcal{D}_{\mathcal{M},k}[u_{\mathcal{C}'}]$. Using Lemma 1, it can be shown that

$$\sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}}(S_{\mathcal{M},k-1,t}|S^{t-1}_{\mathcal{M},k-1}) = \sum_{t=2}^{\infty}\beta^{t-2}H^{\mathcal{C}'}(S_{\mathcal{M},k,t}|S^{t-1}_{\mathcal{M},k}).$$

Then, since $\mathcal{C}$ is chosen arbitrarily, using the induction hypothesis, we obtain $E_{j,\max}\leq E_{k,\max}$ for all $j\leq k$. This completes the proof. □

APPENDIX B

In this appendix, we describe a method to transform the finite horizon entropy maximization problem to infinite horizon entropy maximization problem with discount factor $\beta=1$. Even though the transformation requires us to use the discount factor $\beta=1$, the resulting POMDP includes a "sink state" which allows us to extend all theoretical results provided in the paper to finite horizon entropy maximization problem.

For a given POMDP $\mathcal{M}$ and a finite decision horizon $N\in\mathbb{N}$, we append the time as an additional state to the underlying transition system. In particular, instead of using $\mathcal{S}$, we use $(\mathcal{S}\times[N])\cup Sink$ as the state space, where $[N]=\{1,2,\dots,N\}$. The initial state of the resulting POMDP is $s_I\times 1$, and the set of actions are the same as $\mathcal{M}$. The state transition function $\overline{\mathcal{P}}$ is given as

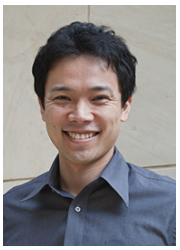
$$\overline{\mathcal{P}}((s',t')|(s,t),a) = \begin{cases} \mathcal{P}(s'|s,a) & \text{if } t'=t+1\wedge t'<N \\ 1 & \text{if } (s',t')=Sink\wedge t=N \\ 1 & \text{if } (s',t')=(s,t)=Sink \\ 0 & \text{otherwise.} \end{cases}$$

In other words, the process moves forward in time and get absorbed in the $Sink$ state after N-th stage. Now, on the resulting POMDP, all results provided for infinite horizon entropy maximization problem can be extended to the finite horizon setting.

Yagiz Savas joined the Department of Aerospace Engineering at the University of Texas at Austin as a Ph.D. student in Fall 2017. He received his B.S. degree in Mechanical Engineering from Bogazici University in 2017. His research focuses on developing theory and algorithms that guarantee desirable behavior of autonomous systems operating in uncertain and adversarial environments.

Michael Hibbard joined the Department of Aerospace Engineering at the University of Texas at Austin as a Ph.D. student in Fall 2018. He received his B.S. degree in Engineering Mechanics and Astronautics from the University of Wisconsin-Madison in Spring 2018. His research interests lie in the development of theory and algorithms providing formal guarantees for the mission success of autonomous agents acting in adversarial environments.

Bo Wu received a B.S. degree from Harbin Institute of Technology, China, in 2008, an M.S. degree from Lund University, Sweden, in 2011 and a Ph.D. degree from the University of Notre Dame, USA, in 2018, all in electrical engineering. He is currently a postdoctoral researcher at the Oden Institute for Computational Engineering and Sciences at the University of Texas at Austin. His research interest is to apply formal methods, learning, and control in autonomous systems, such as robotic systems, communication systems, and human-in-the-loop systems, to provide privacy, security, and performance guarantees.



Takashi Tanaka received the B.S. degree from the University of Tokyo, Tokyo, Japan, in 2006, and the M.S. and Ph.D. degrees in Aerospace Engineering (automatic control) from the University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA, in 2009 and 2012, respectively. He was a Postdoctoral Associate with the Laboratory for Information and Decision Systems (LIDS) at the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from 2012 to 2015, and a postdoctoral researcher at KTH Royal Institute of Technology, Stockholm, Sweden, from 2015 to 2017. Currently, he is an Assistant Professor in the Department of Aerospace Engineering and Engineering Mechanics at the University of Texas at Austin. His research interests include control theory and its applications; most recently the information-theoretic perspectives of optimal control problems. Dr. Tanaka was a recipient of the IEEE Conference on Decision and Control (CDC) best student paper award in 2011.

Ufuk Topcu joined the Department of Aerospace Engineering at the University of Texas at Austin as an assistant professor in Fall 2015. He received his Ph.D. degree from the University of California at Berkeley in 2008. He held research positions at the University of Pennsylvania and California Institute of Technology. His research focuses on the theoretical, algorithmic and computational aspects of design and verification of autonomous systems through novel connections between formal methods, learning theory and controls.