Parameterized MDPs and Reinforcement Learning Problems - A Maximum Entropy Principle Based Framework

Amber Srivastava and Srinivasa M Salapaka, *Member, IEEE*

Abstract—We present a framework to address a class of sequential decision making problems. Our framework features learning the optimal control policy with robustness to noisy data, determining the unknown state and action parameters, and performing sensitivity analysis with respect to problem parameters. We consider two broad categories of sequential decision making problems modelled as infinite horizon Markov Decision Processes (MDPs) with (and without) an absorbing state. The central idea underlying our framework is to quantify exploration in terms of the Shannon Entropy of the trajectories under the MDP and determine the stochastic policy that maximizes it while guaranteeing a low value of the expected cost along a trajectory. This resulting policy enhances the quality of exploration early on in the learning process, and consequently allows faster convergence rates and robust solutions even in the presence of noisy data as demonstrated in our comparisons to popular algorithms such as Q-learning, Double Q-learning and entropy regularized Soft Q-learning. The framework extends to the class of parameterized MDP and RL problems, where states and actions are parameter dependent, and the objective is to determine the optimal parameters along with the corresponding optimal policy. Here, the associated cost function can possibly be non-convex with multiple poor local minima. Simulation results applied to a 5G small cell network problem demonstrate successful determination of communication routes and the small cell locations. We also obtain sensitivity measures to problem parameters and robustness to noisy environment data.

I. INTRODUCTION

MARKOV Decision Processes (MDPs) model sequential decision making problems which arise in many application areas such as robotics, automatic control, sensor networks, economics, and manufacturing. These models are characterized by the state-evolution dynamics $s_{t+1} = f(s_t, a_t)$, a control policy $\mu(a_t|s_t)$ that allocates an *action* a_t from a control set to each state s_t, and a cost $c(s_t, a_t, s_{t+1})$ associated with the transition from s_t to s_{t+1}. The goal in these applications is to determine the optimal control policy that results in a *path*, a sequence of actions and states, with minimum cumulative cost. There are many variants of this problem [1], where the dynamics can be defined over finite or infinite horizons; where the state-dynamics f can be stochastic in which case expected cost functions are minimized; and reinforcement learning (RL) problems where the models for the state dynamics may be partially or completely unknown,

and cost function is not known a priori, albeit the cost at each step is revealed at the end of each transition. Some of the most common methodologies that address MDPs include dynamic programming, value and policy iterations [2], linear programming [3]–[5], and Q-learning [6].

In this article, we view MDPs and their variants as combinatorial optimization problems, and develop a framework based on maximum entropy principle (MEP) [7] to address them. MEP has proved successful in addressing a variety of combinatorial optimization problems such as facility location problems [8], combinatorial drug discovery [9], traveling salesman problem and its variants [8], impage processing [10], graph and markov chain aggregation [11], and protein structure alignment [12]. MDPs, too, can be viewed as *combinatorial* optimization problems - due to the combinatorially large number of paths (sequence of consecutive states and actions) that it may take based on the control policy and its inherent stochasticity. In our MEP framework, we determine a probability distribution defined on *the space of paths* [13], such that (a) it is the *fairest* distribution - the one with the maximum Shannon Entropy H, and (b) it satisfies the constraint that the expected cumulative cost J attains a prespecified feasible value J_0. The framework results in an iterative scheme, an *annealing* scheme, where probability distributions are improved upon by successively lowering the prespecified values J_0. In fact, the lagrange multiplier β corresponding to the cost constraint ($J = J_0$) in the unconstrained lagrangian is increased from small values (near 0) to large values to effect annealing. Higher values of multipliers correspond to lower values of the expected cost. We show that as multiplier value increases, the corresponding probability distributions become more localized, finally converging to a deterministic policy.

This framework is applicable to all the classes of MDPs and its variants described above. Our MEP based approach inherits the flexibility of algorithms such as deterministic annealing [8] developed in the context of combinatorial resource allocation, which include adding capacity, communication, and dynamic constraints. The added advantage of our approach is that we can draw close parallels to existing algorithms for MDPs and RL (e.g. Q-Learning) thus enabling us to exploit their algorithmic insights. Below we highlight some features and advantages of our approach.

Exploration and Unbiased Policy: In the context of model-free RL setting, the algorithms interact with the *environment* via *agents* and rely upon the instantaneous cost (or reward) generated by the environment to learn the optimal policy. Some

of the popular algorithms include Q-learning [6], Double Q-learning [14], Soft Q-learning (entropy regularized Q-learning) [15]–[21] in discrete state and action spaces, and TRPO [22], and SAC [23] in continuous spaces. It is commonly known that for the above algorithms to perform well, all *relevant* states and actions should be explored. In fact, under the assumption that each state-action pair is visited multiple times during the learning process it is guaranteed that the above discrete space algorithms [6], [14]–[16] will converge to the optimal policy. Thus, the adequate *exploration* of the state and action spaces becomes incumbent to the success of these algorithms in determining the optimal policy. Often the instantaneous cost is noisy [15] which hinders with the learning process and demands for an enhanced quality exploration.

In our MEP based approach, the Shannon Entropy of the probability distribution over the paths in the MDP explicitly characterizes the *exploration*. The framework results in a *distribution over the paths* that is as *unbiased* as possible under the given cost constraint. The corresponding stochastic policy is maximally noncommittal to any particular path in the MDP that achieves the constraint; this results in better (unbiased) exploration. The policy starts from being entirely explorative, when multiplier value is small ($\beta \approx 0$), and becomes increasingly *exploitative* as the multiplier value is increased. In fact, simulations show that the unbiased exploration scheme during the early stages of the learning process precipitates into a faster convergence in comparison to several benchmark algorithms. An additional feature of our framework is that the exploration performed by the stochastic policy accounts for the time window T beyond which the instantaneous costs $c_t(s_t, a_t, s_{t+1})$ do not contribute significantly to the cumulative cost (i.e., $c_t < \epsilon \; \forall \; t \geq T$). More precisely, the stochastic policy *induces* higher exploration when T is large (or equivalently, when there are many possible paths of the MDP) and lesser exploration when T is small (i.e., lesser number of possible paths) thereby inherently optimizing the exploration-exploitation trade-off.

In this article, we also address the class of MDPs where the associated *maximum* Shannon Entropy is infinite (such as MDPs without a termination state). In this case, we use a discounted Shannon Entropy term [24], [25]. However, unlike the existing literature [25], we do not necessitate the discounting factor in the entropy term to be equal to the one in the cost function of the MDP. In contrast, simulations from our approach show that choosing a larger discount factor ($\alpha \lesssim 1$) for the entropy term than the discount factors ($\gamma < 1$) in the value function J results in better convergence rates.

Parameterized MDPs: The class of optimization problems are not even necessarily MDPs which contributes significantly to their inherent complexities. However, we model them in a specific way to retain the Markov property without any loss of generality, thereby making these problems tractable. Scenarios such as self organizing networks [26], 5G small cell network design [27], [28], supply chain networks, and last mile delivery problems [29] pose a *parameterized* MDP with a two-fold objective of determining simultaneously (a) the optimal control policy for the underlying stochastic process, and (b) the unknown parameters that the state and action variables depend upon such that the cumulative cost is minimized. The latter
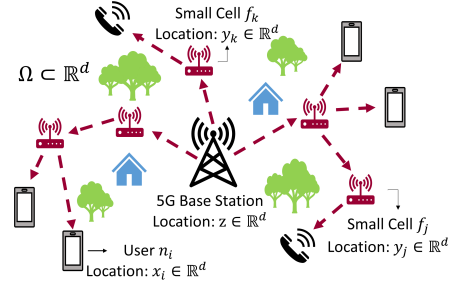

Fig. 1. Illustrates the 5G Small Cell Network. The 5G Base Station δ is located at $z \in \mathbb{R}^d$ and the users $\{n_i\}$ are located at $\{x_i \in \mathbb{R}^d\}$. The objective is to determine the small cell $\{f_j\}$ location $\{y_j \in \mathbb{R}^d\}$ and the communication routes (control policy) from the Base Station δ to each user $\{n_i\}$ via the network of the small cells such that the total cost of communication gets minimized.

objective is akin to NP-hard facility location problem [30], where the associated cost function is riddled with multiple poor local minima. For instance, Figure 1 illustrates a 5G small cell network, where the objective is to simultaneously determine the locations of the small cells $\{f_j\}$ and design the communication paths (control policy) between the user nodes $\{n_i\}$ and base station δ via network of small cells $\{f_j\}$. Here, the state space $\mathcal{S}$ of the underlying MDP is parameterized by unknown locations $\{y_j\}$ of small cells $\{f_j\}$.

The existing solution approaches [2]–[5] can be extended to the parameterized MDPs by discretizing the parameter domain. However, the resulting problem is not necessarily an MDP as every transition from one state to another is dependent on the path (and the parameter values) taken till the current state. Other related approaches for parameterized MDPs are case specific; for instance, [31] presents action-based parameterization of state space with application to service rate control in closed Jackson networks, and [32]–[37] incorporate parameterized actions that is applicable in the domain of RoboCup soccer where at each step the agent must select both the discrete action it wishes to execute as well as continuously valued parameters required by that action. On the other hand, the parameterized MDPs that we address in this article predominantly originate in network based applications that involves simultaneous routing and resource allocations and pose additional challenges of non-convexity and NP-hardness. We address these MDPs in both the scenarios, where the underlying model is known as well as unknown.

Algebraic structure and Sensitivity Analysis: In our framework, maximization of Shannon entropy of the distribution over the paths under constraint on the cost function value results in an unconstrained Lagrangian - the *free-energy* function. This function is a *smooth* approximation of the cumulative cost function of the MDP, which enables the use of calculus. We exploit this distinctive feature of our framework to determine the unknown state and action parameters in case of parameterized MDPs and perform sensitivity analysis for various problem parameters. Also, the framework easily accommodates stochastic models that describe uncertainties in the instantaneous cost and parameter values.

Algorithmic Guarantees and Innovation: For the classes of MDPs that we consider, our MEP-based framework results into non-trivial derivations of the recursive Bellman equation

for the associated Lagrangian. We show that these Bellman operators are contraction maps and use their several properties to guarantee the convergence to the optimal policy and as well as to a local minima in the case of parameterized MDPs.

Through simulations, we compared our proposed framework with benchmark algorithms Q, Double Q and entropy regularized Soft Q-learning. Our algorithms demonstrate a smaller variance (an average of over 55% of variance in Q and Soft-Q learning) on our test cases with multiple runs, converge at a faster rate (as high as 2 times) than the benchmark algorithms even in the case of noisy environments [15]. We implemented simulations on the parameterized MDPs that models small-cell network design problem in 5G communication. Upon comparison with the sequential method of first determining the unknown parameters (small cell locations) and then the control policy (equivalently, the communication paths), we show that our algorithms result into costs that are as low as 65% of the former. The efficacy of our algorithms can be assessed from the fact that the solutions in the model-based and model-free cases are approximately the same (differ by $< 2\%$).

Prior Work on Entropy Regularization: Some of the previous works in RL literature [15]–[21], [38], [39] either use entropy as a regularization term $(-\log \mu(a_t|s_t))$ [15], [16] to the instantaneous cost function $c(s_t, a_t, s_{t+1})$ or maximize the entropy $(-\sum_a \mu(a|s) \log \mu(a|s))$ [17]–[19] associated *only* to the stochastic policy under constraints on the cost J. This results into benefits such as better exploration, overcoming the effect of noise w_t in the instantaneous cost c_t and obtaining faster convergence. However, even though the resulting stochastic policy and soft-max approximation of J provides appreciable improvements over benchmark algorithms such as Q, Double Q-learning [15], and policy gradient methods [17], it is not in compliance with the Maximum Entropy Principle applied to the distribution over the paths of MDP. Thus, the resulting stochastic policy is inherently *biased* in its exploration *over the paths of the MDP*. Simulations demonstrate the benefit of *unbiased* exploration (in our framework) in terms of faster convergence, better performance in noisy environment, and lesser variance in comparison to the biased exploration of the benchmark algorithms.

II. MAXIMUM ENTROPY PRINCIPLE

We briefly review the Maximum Entropy Principle (MEP) [7] since our framework relies heavily upon it. MEP states that for a random variable $\mathcal{X}$ with a given prior information, the most unbiased probability distribution given prior data is the one that maximizes the Shannon entropy. More specifically, let the known prior information of the random variable $\mathcal{X}$ be given as constraints on the expectation of the functions $f_k : \mathcal{X} \to \mathbb{R}$, $1 \leq k \leq m$. Then, the most unbiased probability distribution $p_{\mathcal{X}}(\cdot)$ solves the optimization problem

$$\max_{\{p_{\mathcal{X}}(x_i)\}} \quad H(\mathcal{X}) = -\sum_{i=1}^{n} p_{\mathcal{X}}(x_i) \ln p_{\mathcal{X}}(x_i)$$
$$\text{subject to} \quad \sum_{i=1}^{n} p_{\mathcal{X}}(x_i) f_k(x_i) = F_k \quad \forall \, 1 \leq k \leq m, \tag{1}$$

where F_k, $1 \leq k \leq m$, are known expected values of the functions f_k. The above optimization problem results into the Gibbs' distribution [40] $p_{\mathcal{X}}(x_i) = \frac{\exp\{-\sum_k \lambda_k f_k(x_i)\}}{\sum_{j=1}^{n} \exp\{-\sum_k \lambda_k f_k(x_j)\}}$, where λ_k, $1 \leq k \leq m$, are the Lagrange multipliers corresponding to the inequality constraints in (1).

III. MDPs WITH FINITE SHANNON ENTROPY

A. Problem Formulation

We consider an infinite horizon discounted MDP that comprises of a *cost-free termination* state δ. We formally define this MDP as a tuple $\langle \mathcal{S}, \mathcal{A}, c, p, \gamma \rangle$ where $\mathcal{S} = \{s_1, \ldots, s_N = \delta\}$, $\mathcal{A} = \{a_1, \ldots, a_M\}$, and $c : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ respectively denote the state space, action space, and cost function; $p : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to [0,1]$ is the state transition probability function and $0 < \gamma \leq 1$ is the discounting factor. A control policy $\mu : \mathcal{A} \times \mathcal{S} \to \{0,1\}$ determines the action taken at each state $s \in \mathcal{S}$, where $\mu(a|s) = 1$ implies that action $a \in \mathcal{A}$ is taken when the system is in the state $s \in \mathcal{S}$ and $\mu(a|s) = 0$ indicates otherwise. For every initial state $x_0 = s$, the MDP induces a stochastic process, whose realization is a *path* ω - an infinite sequence of actions and states, that is

$$\omega = (u_0, x_1, u_1, x_2, u_2, \ldots, x_T, u_T, x_{T+1}, \ldots), \tag{2}$$

where $u_t \in \mathcal{A}$, $x_t \in \mathcal{S}$ for all $t \in \mathbb{Z}_{\geq 0}$ and $x_t = \delta$ for all $t \geq k$ if and when the system reaches the termination state $\delta \in \mathcal{S}$ in k steps. The objective is to determine the optimal policy μ^* that minimizes the state value function

$$J^{\mu}(s) = \mathbb{E}_{p_{\mu}}\left[\sum_{t=0}^{\infty} \gamma^t c(x_t, u_t, x_{t+1}) \big| x_0 = s\right], \quad \forall \, s \in \mathcal{S} \tag{3}$$

where the expectation is with respect to the probability distribution $p_{\mu}(\cdot|s) : \omega \to [0,1]$ on the space of all possible paths $\omega \in \Omega := \{(u_t, x_{t+1})_{t \in \mathbb{Z}_{\geq 0}} : u_t \in \mathcal{A}, x_t \in \mathcal{S}\}$. In order to ensure that the system reaches the cost-free termination state in finite steps and the optimal state value function $J^{\mu}(s)$ is finite, we make the following assumption throughout this section.

Assumption 1. There exists atleast one deterministic proper policy $\bar{\mu}(a|s) \in \{0,1\} \, \forall \, a \in \mathcal{A}, s \in \mathcal{S}$ such that $\min_{s \in \mathcal{S}} p_{\bar{\mu}}(x_{|\mathcal{S}|} = \delta | x_0 = s) > 0$. In other words, under the policy $\bar{\mu}$ there is a non-zero probability to reach the cost-free termination state δ, when starting from any state s.

In our MEP-based framework we consider the following set of stochastic policies μ

$$\Gamma := \{\pi : 0 < \pi(a|s) < 1 \, \forall \, a \in \mathcal{A}, s \in \mathcal{S}\}, \tag{4}$$

and the following lemma ensures that under the Assumption 1 all the policies $\mu \in \Gamma$ are *proper*; that is,

Lemma 1. *For any policy $\mu \in \Gamma$ as defined in (4), $\min_{s \in \mathcal{S}} p_{\mu}(x_{|\mathcal{S}|} = \delta | x_0 = s) > 0$, i.e., under each policy $\mu \in \Gamma$ the probability to reach the termination state δ in $|\mathcal{S}| = N$ steps beginning from any $s \in \mathcal{S}$, is strictly positive.*

Proof. Please refer to the Appendix A.

We use the Maximum Entropy Principle to determine the policy $\mu \in \Gamma$ such that the Shannon Entropy of the corresponding distribution p_{μ} is maximized and the state value function

$J^\mu(s)$ attains a specified value J_0. More specifically, we pose the following optimization problem

$$\max_{\{p_\mu(\cdot|s)\}:\mu\in\Gamma} H^\mu(s) = -\sum_{\omega\in\Omega} p_\mu(\omega|s)\log p_\mu(\omega|s)$$

$$\text{subject to} \quad J^\mu(s) = J_0. \tag{5}$$

Well posedness: For the class of proper policy $\mu \in \Gamma$ the maximum entropy $H^\mu(s) \ \forall \ s \in \mathcal{S}$ is finite as shown in [41], [42]. In short, the existence of a cost-free termination state δ and a non-zero probability to reach it from any state $s \in \mathcal{S}$ ensures that the maximum entropy is finite. Please refer to the Theorem 1 in [41] or Proposition 2 in [42] for further details.

Remark. Though the optimization problem in (5) considers the stochastic policies $\mu \in \Gamma$, our algorithms presented in the later sections are designed in such that the resulting stochastic policy asymptotically converges to a deterministic policy.

B. Problem Solution

The probability $p_\mu(\omega|s)$ of taking the path ω in (2) can be determined from the underlying policy μ by exploiting the Markov property that dissociates $p_\mu(\omega|s)$ in terms of the policy μ and the state transition probability p as

$$p_\mu(\omega|x_0) = \prod_{t=0}^{\infty} \mu(u_t|x_t)p(x_{t+1}|x_t,u_t). \tag{6}$$

Thus, in our framework we prudently work with the policy μ which is defined over finite action and state spaces as against the distribution $p_\mu(\omega|s)$ defined over infinitely many paths $\omega \in \Omega$. The Lagrangian corresponding to the above optimization problem in (5) is $V_\beta^\mu(s) = \big(J^\mu(s) - J_0\big) - \frac{1}{\beta}H^\mu(s) =$

$$\mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t c_{x_t x_{t+1}}^{u_t} + \frac{1}{\beta}(\log\mu_{u_t|x_t} + \log p_{x_t x_{t+1}}^{u_t})\Big|x_0 = s\Big], \tag{7}$$

where β is the Lagrange parameter. We refer to the above Lagrangian $V_\beta^\mu(s)$ (7) as the *free-energy* function and $\frac{1}{\beta}$ as *temperature* due to their close analogies with statistical physics (where free energy is enthalpy (E) minus the temperature times entropy (TH)). To determine the optimal policy μ_β^* that minimizes the Lagrangian $V_\beta^\mu(s)$ in (7), we first derive the Bellman equation corresponding to $V_\beta^\mu(s)$.

Theorem 1. *The free-energy function $V_\beta^\mu(s)$ in (7) satisfies the following recursive Bellman equation*

$$V_\beta^\mu(s) = \sum_{\substack{a\in\mathcal{A}\\s'\in\mathcal{S}}} \mu_{a|s}p_{ss'}^a\big(\bar{c}_{ss'}^a + \frac{\gamma}{\beta}\log\mu_{a|s} + \gamma V_\beta^\mu(s')\big), \tag{8}$$

where $\mu_{a|s} = \mu(a|s)$, $p_{ss'}^a = p(s'|s,a)$, $\bar{c}_{ss'}^a = c(s,a,s') + \frac{\gamma}{\beta}\log p(s'|s,a)$ for simplicity in notation.

Proof. Please refer to the Appendix A for details. It must be noted that this derivation shows and exploits the algebraic structure $\sum_{s'} p_{ss'}^a H^\mu(s') = \sum_{s'} p_{ss'}^a \log p_{ss'}^a + \log\mu_{a|s} + \lambda_s$ as detailed in Lemma 2 in the appendix.

Now the optimal policy satisfies $\frac{\partial V_\beta^\mu(s)}{\partial\mu(a|s)} = 0$, which results into the Gibbs distribution

$$\mu_\beta^*(a|s) = \frac{\exp\{-(\beta/\gamma)\Lambda_\beta(s,a)\}}{\sum_{a'\in\mathcal{A}}\exp\{-(\beta/\gamma)\Lambda_\beta(s,a')\}}, \quad \text{where} \tag{9}$$

$$\Lambda_\beta(s,a) = \sum_{s'\in\mathcal{S}} p_{ss'}^a\big(\bar{c}_{ss'}^a + \gamma V_\beta^*(s')\big), \tag{10}$$

is the state-action value function, $p_{ss'}^a = p(s'|s,a)$, $c_{ss'}^a = c(s,a,s')$, $\bar{c}_{ss'}^a = c_{ss'}^a + \frac{\gamma}{\beta}\log p_{ss'}^a$ and $V_\beta^*(= V_\beta^{\mu_\beta^*})$ is the value function corresponding to the policy μ_β^*. To avoid notional clutter we use the above notations wherever it is clear from the context. Substituting the policy μ_β^* in (9) back into the Bellman equation (8) we obtain the *implicit* equation

$$V_\beta^*(s) = -\frac{\gamma}{\beta}\log\Big(\sum_{a\in\mathcal{A}}\exp\Big\{-\frac{\beta}{\gamma}\Lambda_\beta(s,a)\Big\}\Big). \tag{11}$$

To solve for the state-action value function $\Lambda_\beta(s,a)$ and free-energy function $V_\beta^*(s)$ we substitute the expression of $V_\beta^*(s)$ in (11) into the expression of $\Lambda_\beta(s,a)$ in (10) to obtain the implicit equation $\Lambda_\beta(s,a) =: [T\Lambda_\beta](s,a)$, where

$$[T\Lambda_\beta](s,a) = \sum_{s'\in\mathcal{S}} p_{ss'}^a\big(c_{ss'}^a + \frac{\gamma}{\beta}\log p_{ss'}^a\big)$$
$$- \frac{\gamma^2}{\beta}\sum_{s'\in\mathcal{S}} p_{ss'}^a\log\sum_{a'\in\mathcal{A}}\exp\Big\{-\frac{\beta}{\gamma}\Lambda_\beta(s',a')\Big\}. \tag{12}$$

To solve the above implicit equation, we show that the map T in (12) is a contraction map and therefore Λ_β can be obtained using fixed point iterations, which guarantee converging to the unique fixed point. Consequently, the global minimum V_β^* in (11) and the optimal policy μ_β^* in (9) can be obtained.

Theorem 2. *The map $[T\Lambda_\beta](s,a)$ as defined in (12) is a contraction mapping with respect to a weighted maximum norm, i.e. $\exists$ a vector $\xi = (\xi_s) \in \mathbb{R}^{|\mathcal{S}|}$ with $\xi_s > 0 \ \forall \ s \in \mathcal{S}$ and a scalar $\alpha < 1$ such that*

$$\|T\Lambda_\beta - T\Lambda_\beta'\|_\xi \le \alpha\|\Lambda_\beta - \Lambda_\beta'\|_\xi \tag{13}$$

where $\|\Lambda_\beta\|_\xi = \max_{s\in\mathcal{S},a\in\mathcal{A}} \frac{|\Lambda_\beta(s,a)|}{\xi_s}$

Proof. Please refer to Appendix B for details.

Remark. It is known from the sensitivity analysis [40] that the value of the Lagrange parameter β in (7), (11) is inversely proportional to the constant J_0 in (5). It is clear from (7) that for small values of $\beta \approx 0$ (equivalently large J_0), we are mainly maximizing the Shannon Entropy $H^\mu(s)$ and the resultant policy in (9) naturally encourages exploration along the paths of the MDP. As β increases (J_0 decreases), more and more weight is given to the state value function $J^\mu(s)$ in (7) and the policy in (9) goes from being exploratory to being exploitative. As $\beta \to \infty$ the exploration is completely eliminated and we converge to a deterministic policy $\to \mu^*$ that minimizes $J^\mu(s)$ in (3).

Remark. We briefly draw readers' attention to the value function $Y(s) = \mathbb{E}[\sum_{t=0}^{\infty}\gamma^t(c_{x_t x_{t+1}}^{u_t} + (1/\beta)\log\mu_{u_t|x_t})]$ considered in the entropy regularized methods [15]. Note that in $Y(s)$ the discounting γ^t is multiplied to both the cost term $c_{x_t x_{t+1}}^{u_t}$ as well as the entropy term $(1/\beta)\log\mu_{u_t|x_t}$. However, in our MEP-based method, the Lagrangian $V_\beta^\mu(s)$ in (7) comprises of discounting γ^t *only* over the cost term $c_{x_t x_{t+1}}^{u_t}$ and not on the entropy terms $(1/\beta)\log\mu_{u_t|x_t}$ and $(1/\beta)\log p_{x_t x_{t+1}}^{u_t}$. Therefore, the policy in [15] does not satisfy MEP applied

over the distribution p_μ; consequently their exploration along the paths is not as unbiased as our algorithm (as demonstrated in Section VI).

C. Model-free Reinforcement Learning Problems

In these problems, the cost function $c(s, a, s')$ and the state-transition probability $p(s'|s, a)$ are not known apriori; however, at each discrete time instant t the *agent* takes an action u_t under a policy μ and the *environment* (underlying stochastic process) results into an instantaneous cost $c^{u_t}_{x_t x_{t+1}}$ and the subsequently moves to the state $x_{t+1} \sim p(\cdot|x_t, u_t)$. Motivated by the iterative updates of Q-learning algorithm [2] we consider the following stochastic updates in our Algorithm 1 to learn the state-action value function in our methodology

$$\Psi_{t+1}(x_t, u_t) = (1 - \nu_t(x_t, u_t))\Psi_t(x_t, u_t) +$$
$$\nu_t(x_t, u_t)\Big[c^{u_t}_{x_t x_{t+1}} - \frac{\gamma^2}{\beta} \log \sum_{a' \in \mathcal{A}} \exp\Big\{\frac{-\beta}{\gamma}\Psi_t(x_{t+1}, a')\Big\}\Big], \quad (14)$$

with the stepsize parameter $\nu_t(x_t, u_t) \in (0, 1]$, and show that under appropriate conditions on ν_t (as illustrated shortly), Ψ_t will converge to the fixed point $\bar{\Lambda}^*_\beta$ of the implicit equation

$$\bar{\Lambda}_\beta(s, a) = \sum_{s' \in \mathcal{S}} p^a_{ss'}\Big(c^a_{ss'} - \frac{\gamma^2}{\beta} \log \sum_{a'} \exp\big(\frac{-\beta}{\gamma}\bar{\Lambda}_\beta(s', a')\big)\Big)$$
$$=: [\bar{T}\bar{\Lambda}_\beta](s, a). \quad (15)$$

The above equation comprises of a minor change from the equation $\Lambda_\beta(s, a) = [T\Lambda_\beta](s, a)$ in (12). The latter has an additional term $\frac{\gamma}{\beta}\sum_{s'} p^a_{ss'} \log p^a_{ss'}$ which makes it difficult to *learn* its fixed point Λ^*_β in the absence of the state transition probability $p^a_{ss'}$ itself. Since in this work we do not attempt to determine (or learn) either the distribution $p^a_{ss'}$ (as in [43]) from the interactions of the agent with the environment, we work with the approximate state-action value function $\bar{\Lambda}_\beta$ in (15) where $\bar{\Lambda}_\beta \to \Lambda_\beta$ for large β values (since $\frac{\gamma}{\beta}(\sum_{s'} p^a_{ss'} \log p^a_{ss'}) \to 0$ as $\beta \to \infty$). The following proposition elucidates the conditions under which the updates Ψ_t in (14) converge to the fixed point $\bar{\Lambda}^*_\beta$.

Proposition 1. *Consider the class of MDPs illustrated in Section III-A. Given that*

$$\sum_{t=0}^{\infty} \nu_t(s, a) = \infty, \sum_{t=0}^{\infty} \nu_t^2(s, a) < \infty \ \forall \ s \in \mathcal{S}, a \in \mathcal{A},$$

the update $\Psi_t(s, a)$ in (14) converges to the fixed point $\bar{\Lambda}^*_\beta$ of the map $\bar{T}\bar{\Lambda}_\beta \to \bar{\Lambda}_\beta$ in (15) with probability 1.

Proof. Please refer to the Appendix D.

Remark. Note that the *stochasticity* of the optimal policy $\mu^*_\beta(a|s)$ (9) depends on γ value which allows it to incorporate for the effect of the discount factor on its exploration strategy. More precisely, in the case of large discount factors the time window T, in which instantaneous costs $\gamma^t c(s_t, a_t, s_{t+1})$ is considerable (i.e., $\gamma^t c^{a_t}_{s_t s_{t+1}} > \epsilon \ \forall \ t \leq T$), is large and thus, the stochastic policy (9) performs higher exploration along the paths. On the other hand, for small discount factors this time window T is relatively smaller and thus, the stochastic policy (9) inherently performs lesser exploration. As illustrated in

---

Algorithm 1: Model-free Reinforcement Learning

Input: $\beta_{\min}$, $\beta_{\max}$, N, $\gamma_t(\cdot, \cdot)$, τ; **Output:** μ^*, $\bar{\Lambda}^*$
Initialize: $t = 0$, $\beta = \beta_{\min}$, $\Psi_0 = 0$, $\mu_0(a|s) = 1/|\mathcal{A}|$.
for $episode = 1$ *to* N **do**
  $\beta \leftarrow \nu\beta$; reset environment at state x_t
  while *True* **do**
    sample $u_t \sim \mu_t(\cdot|x_t)$; obtain cost c_t and x_{t+1}
    update $\Psi_t(x_t, u_t)$, $\mu_{t+1}(u_t|x_t)$ in (14) and (9)
    break if $x_{t+1} = \delta$; $t \leftarrow t + 1$

---

the simulations, this characteristic of the policy in (9) results into even faster convergence rates in comparison to benchmark algorithms as the discount factor γ decreases.

IV. MDPs WITH INFINITE SHANNON ENTROPY

Here we consider the MDPs where the Shannon Entropy $H^\mu(s)$ of the distribution $\{p_\mu(\omega|s)\}$ over the paths $\omega \in \Omega$ is not necessarily finite (for instance, due to absence of absorption state). To ensure the finiteness of the objective in (5) we consider the *discounted* Shannon Entropy [24], [25]

$$H^\mu_d(s) = -\mathbb{E}\Big[\sum_{t=0}^{\infty} \alpha^t (\log \mu_{u_t|x_t} + \log p^{u_t}_{x_t x_{t+1}})\big|x_0 = s\Big] \quad (16)$$

with a discount factor of $\alpha \in (0, 1)$ which we chose to be independent of the discount factor γ in the value function $J^\mu(s)$. The free-energy function (or, the Lagrangian) resulting from the optimization problem in (5) with the alternate objective function $H^\mu_d(s)$ in (16) is given by

$$V^\mu_{\beta,I}(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \hat{c}^{u_t}_{x_t x_{t+1}} + \frac{\alpha^t}{\beta} \log \mu(u_t|x_t)\big|x_0 = s\Big], \quad (17)$$

where $\hat{c}^{u_t}_{x_t x_{t+1}} = c^{u_t}_{x_t x_{t+1}} + \frac{\gamma^t}{\beta \alpha^t} \log p^{u_t}_{x_t x_{t+1}}$, and the subscript I stands for 'infinite entropy' case. Note that the free-energy functions (7) and (17) differ only with regards to the discount factor α and thus, our solution methodology in this section is similar to the one in Section III-B.

Theorem 3. *The free-energy function $V^\mu_{\beta,I}(s)$ in (17) satisfies the recursive Bellman equation*

$$V^\mu_{\beta,I}(s) = \sum_{a,s'} \mu_{a|s} p^a_{ss'}(\check{c}^a_{ss'} + \frac{\gamma}{\alpha\beta} \log \mu_{a|s} + \gamma V^\mu_{\beta,I}(s')) \quad (18)$$

where $\check{c}^a_{ss'} = c^a_{ss'} + \frac{\gamma}{\alpha\beta} \log p^a_{ss'}$.

Proof. Please see Appendix C. The above derivation shows and exploits algebraic structure $\alpha \sum_{s'} p^a_{ss'} H^\mu_d(s') = \sum_{s'} p^a_{ss'} \log p^a_{ss'} + \log \alpha\mu(a|s) + \lambda_s$ (Lemma 4).

The optimal policy satisfies $\frac{\partial V^\mu_{\beta,I}(s)}{\partial \mu(a|s)} = 0$, which results into the Gibbs distribution

$$\mu^*_{\beta,I}(a|s) = \frac{\exp\big\{-\frac{\beta\alpha}{\gamma}\Phi_\beta(s, a)\big\}}{\sum_{a' \in \mathcal{A}} \exp\big\{-\frac{\beta\alpha}{\gamma}\Phi_\beta(s, a')\big\}}, \text{ where} \quad (19)$$

$$\Phi_\beta(s, a) = \sum_{s' \in \mathcal{S}} p^a_{ss'}(\check{c}^a_{ss'} + \gamma V^*_{\beta,I}(s')), \quad (20)$$

is the corresponding state-action value function. Substituting the $\mu_{\beta,I}^*$ in (19) in the Bellman equation (18) results into the following optimal free-energy function $V_{\beta,I}^*(s)(:= V_{\beta,i}^{\mu_{\beta,i}^*}(s))$.

$$V_{\beta,i}^*(s) = -\frac{\gamma}{\alpha\beta} \log \sum_{a' \in \mathcal{A}} \exp\left(\frac{-\alpha\beta}{\gamma} \Phi_\beta(s, a')\right) \quad (21)$$

Remark. The subsequent steps to learn the optimal policy $\mu_{\beta,I}^*$ in (19) are similar to the steps demonstrated in the Section III-C. We forego the similar analysis here.

Remark. When $\alpha = \gamma$ the policy $\mu_{\beta,I}^*$ in (19), state-action value function Φ_β in (20) and the free-energy function $V_{\beta,I}^*$ in (21) corresponds to the similar expressions that are obtained in the Entropy regularized methods [15]. However, in this paper we do not require that $\alpha = \gamma$. On the other hand, we propose that α should take up large values. In fact, our simulations in the Section VI demonstrate better convergence rates that are obtained when $\gamma < \alpha = (1 - \epsilon)$ as compared to the scenarios where $\gamma = \alpha < 1$.

V. PARAMETERIZED MDPS

A. Problem Formulation

As illustrated in Section I, many application areas such as battlefield surveillance, self organizing networks and 5G small cell networks (see Figure 1) pose a parmaterized MDP that requires simultaneously determining the (a) optimal policy μ^* to reach the termination state $\delta \in \mathcal{S}$, and (b) the unknown state and action parameters $\zeta = \{\zeta_s\}$ and $\eta = \{\eta_a\}$ such that the state value function

$$J_{\zeta\eta}^\mu(s) = \mathbb{E}_{p_\mu}\left[\sum_{t=0}^\infty \gamma^t c(x_t(\zeta), u_t(\eta), x_{t+1}(\zeta)) \Big| x_0 = s\right] \quad (22)$$

is minimized $\forall\ s \in \mathcal{S}$ where $x_t(\zeta)$ denotes the state $x_t \in \mathcal{S}$ with the associated parameter ζ_{x_t} and $u_t(\eta)$ denotes the action $u_t \in \mathcal{A}$ with the associated action parameter value η_{u_t}. As in Section III-A, we assume that the parameterized MDPs exhibit atleast one deterministic proper policy (Assumption 1) to ensure the finiteness of the value function $J_{\zeta\eta}^\mu(s)$ and the Shannon Entropy $H^\mu(s)$ of the MDP for all $\mu \in \Gamma$. We further assume that the state-transition probability $\{p_{ss'}^a\}$ is independent of the state and action parameters ζ, η.

B. Problem Solution

This problem was solved in Section III-B, where the states and actions were not parameterized, or equivalently can be viewed as if parameters ζ, η were known and fixed. For the parameterized case, we apply the same solution methodology, which results in the same optimal policy $\mu_{\beta,\zeta\eta}^*$ as in (9) as well as the corresponding free-energy function $V_{\beta,\zeta\eta}^*(s)$ in (11) except that now they are characterized by the parameters ζ, η. To determine the optimal (local) parameters ζ, η, we set

$$\sum_{s' \in \mathcal{S}} \frac{\partial V_{\beta,\zeta\eta}^*(s')}{\partial \zeta_s} = 0 \forall\ s \text{ and } \sum_{s' \in \mathcal{S}} \frac{\partial V_{\beta,\zeta\eta}^*(s')}{\partial \eta_a} = 0 \forall\ a, \quad (23)$$

which we implement by using the gradient descent steps

$$\zeta_s^+ = \zeta_s - \eta \sum_{s' \in \mathcal{S}} G_{\zeta_s}^\beta(s'), \quad \eta_a^+ = \eta_a - \bar{\eta} \sum_{s' \in \mathcal{S}} G_{\eta_a}^\beta(s'). \quad (24)$$

Here $G_{\zeta_s}^\beta(s') := \partial V_{\beta,\zeta\eta}^*(s')/\partial \zeta_s$ and $G_{\eta_a}^\beta(s') := \partial V_{\beta,\zeta\eta}^*(s')/\partial \eta_a$. The derivatives $G_{\zeta_s}^\beta$ and $G_{\eta_a}^\beta$ are assumed to be bounded (see constraint (b)-(c) in the Proposition 2). We compute these derivatives as $G_{\zeta_s}^\beta(s') = \sum_{a'} \mu_{a'|s'} K_{\zeta_s}^\beta(s', a')$ and $G_{\eta_a}^\beta(s') = \sum_{a'} \mu_{a'|s'} L_{\eta_a}^\beta(s', a')\ \forall\ s' \in \mathcal{S}$ where $K_{\zeta_s}^\beta(s', a')$ and $L_{\eta_a}^\beta(s', a')$ are the fixed points of their corresponding recursive Bellman equations $K_{\zeta_s}^\beta(s', a') = [T_1 K_{\zeta_s}^\beta](s, a)$ and $L_{\eta_a}^\beta(s', a') = [T_2 L_{\eta_a}^\beta](s', a')$ where

$$\begin{aligned}
[T_1 K_{\zeta_s}^\beta](s', a') &= \sum_{s''} p_{s's''}^{a'}\left[\frac{\partial c_{s's''}^{a'}}{\partial \zeta_s} + \gamma G_{\zeta_s}^\beta(s'')\right], \\
[T_2 L_{\eta_a}^\beta](s', a') &= \sum_{s''} p_{s's''}^{a'}\left[\frac{\partial c_{s's''}^{a'}}{\partial \eta_a} + \gamma G_{\eta_a}^\beta(s'')\right].
\end{aligned} \quad (25)$$

Note that in the above equations we have suppressed the dependence of the instantaneous cost function $c_{s's''}^{a'}$ on the parameters ζ and η to avoid notational clutter.

Theorem 4. *The operators $[T_1 K_{\zeta_s}^\beta](s', a')$ and $[T_2 L_{\eta_a}^\beta](s', a')$ defined in (25) are contraction maps with respect to a weighted maximum norm $\|\cdot\|_\xi$, where $\|X\|_\xi = \max_{s',a'} \frac{X(s',a')}{\xi_{s'}}$ and $\xi \in \mathbb{R}^{|\mathcal{S}|}$ is a vector of positive components ξ_s.*

Proof. Please refer the Appendix D for details.

As previously stated in Section I, the state value function $J_{\zeta\eta}^\mu(\cdot)$ in (22) is generally non-convex function of the parameters ζ, η and riddled with multiple poor local minima with the resulting optimization problem being possibly NP-hard [30]. In our proposed algorithm for parameterized MDPs we anneal β from $\beta_{\min}$ to $\beta_{\max}$, similar to our approach for non-parameterized MDPs in Section III-B, where the solution from the current β iteration is used to initialize the subsequent β iteration. However, in addition to facilitating a steady transition from an exploratory policy to an exploitative policy (that is beneficial in the model-free RL setting), annealing facilitates a gradual homotopy from the convex negative Shannon entropy function to the non-convex state value function $J_{\zeta\eta}^\mu$ which prevents our algorithm from getting stuck at a poor local minimum. The underlying idea of our heuristic is to track the optimal as the initial convex function deforms to the actual non-convex cost function. Additionally, minimizing the Lagrangian $V_\beta^*(s)$ at $\beta = \beta_{\min} \approx 0$ determines the global minimum thereby making our algorithm insensitive to initialization. In Algorithm 2 we illustrate the steps to determine the policy and parameter values in a parameterized MDP setting.

C. Parameterized Reinforcement Learning

In many applications, formulated as parameterized MDPs, the explicit knowledge of the cost function $c_{ss'}^a$, its dependence on the parameters ζ, η, and the state-transition probabilities $\{p_{ss'}^a\}$ are not known. However, for each action a the environment results into an instantaneous cost based on its current x_t, next state x_{t+1} and the parameter ζ, η values which can subsequently be used to simultaneously learn the policy $\mu_{\beta,\zeta\eta}^*$ and the unknown state and action parameters ζ, η via stochastic iterative updates. At each β iteration in our learning algorithm, we employ the stochastic iterative updates in (14)

Algorithm 2: Parameterized Markov Decision Process

Input: $\beta_{\min}$, $\beta_{\max}$, τ; **Output:** μ^*, ζ and η.
Initialize: $\beta = \beta_{\min}$, $\mu_{a|s} = \frac{1}{|\mathcal{A}|}$, and $\zeta, \eta, G_\zeta^\beta, G_\eta^\beta$,
$\quad K_\zeta^\beta, L_\eta^\beta \Lambda_\beta$ to 0
while $\beta \leq \beta_{\max}$ **do**
$\quad$ **while** *True* **do**
$\quad\quad$ **while** *until convergence* **do**
$\quad\quad\quad$ update $\Lambda_\beta, \mu_\beta, G_{\zeta_s}^\beta, G_{\eta_a}^\beta$ in (10), (9) and (25)
$\quad\quad$ update ζ, η using gradient descent in (24)
$\quad\quad$ **if** $\|G_{\zeta_s}\| < \epsilon$, $\|G_{\eta_a}\| < \epsilon$ **then** break
$\quad$ $\beta \leftarrow \tau\beta$

to determine the optimal policy $\mu_{\beta,\zeta\eta}^*$ for given ζ, η values and subsequently employ the stochastic iterative updates

$$K_{\zeta_s}^{t+1}(x_t, u_t) = (1 - \nu_t(x_t, u_t))K_{\zeta_s}^t(x_t, u_t) +$$
$$\nu_t(x_t, u_t)\Big[\frac{\partial c_{x_t x_{t+1}}^{u_t}}{\partial \zeta_s} + \gamma G_{\zeta_s}^t(x_{t+1})\Big], \quad (26)$$

where $G_{\zeta_s}^t(x_{t+1}) = \sum_a \mu_{a|x_{t+1}} K_{\zeta_s}^t(x_{t+1}, a)$ to learn the derivative $G_{\zeta_s}^{\beta*}(\cdot)$. Similar updates are used to learn $G_{\eta_a}^{\beta*}(\cdot)$. The parameter values ζ, η are then updated using the gradient descent step in (24). The following proposition formalizes the convergence of the updates in (26) to the fixed point $G_{\zeta_s}^{\beta*}$.

Proposition 2. *For the class of parameterized MDPs considered in Section V-A given that*

(a) $\sum_{t=0}^\infty \nu_t(s,a) = \infty$, $\sum_{t=0}^\infty \nu_t^2(s,a) < \infty \ \forall s \in \mathcal{S}, a \in \mathcal{A}$,
(b) $\exists B > 0$ *such that* $\left|\frac{\partial c(s',a',s'')}{\partial \zeta_s}\right| \leq B \ \forall \ s, s', a', s''$,
(c) $\exists C > 0$ *such that* $\left|\frac{\partial c(s',a',s'')}{\partial \eta_a}\right| \leq C \ \forall \ a, s', a', s''$,

the updates in (26) converge to the unique fixed point $G_{\zeta_s}^{\beta*}(s')$ of the map $T_1 : G_{\zeta_s} \to G_{\zeta_s}$ in (25).

Proof. Please refer to the Appendix D for details.

Algorithmic Details: Please refer to the Algorithm 3 for a complete implementation. Unlike the scenario in Section III-C where the agent acts upon the environment by taking an action $u_t \in \mathcal{A}$ and learns *only* the policy μ^*, here the agent interacts with the environment by (a) taking an action $u_t \in \mathcal{A}$ and *also* providing (b) estimated parameter ζ, η values to the environment; subsequently, the environment results into an instantaneous cost and the next state. In our Algorithm 3, we first learn the policy μ_β^* at a given value of the parameters ζ, η using the iterations (14) and then learn the fixed points $G_{\zeta_s}^{\beta*}$, $G_{\zeta_a}^{\beta*}$ using the iterations in (26) to update the parameters ζ, η using (24). Note that the iterations (26) require the derivatives $\partial c(s', a', s'')/\partial \zeta_s$ and $\partial c(s', a', s'')/\partial \eta_a$ which we determine using the instantaneous costs resulting from two ϵ-*distinct environments* and finite difference method. Here the ϵ-distinct environments essentially represent the same underlying MDP but are distinct only in one of the parameter values. However, if two ϵ-distinct environments are not feasible one can work with a single environment where the algorithm stores the instantaneous costs and the corresponding parameter values upon each interaction with the environment which can later

Algorithm 3: Parameterized Reinforcement Learning

Input: $\beta_{\min}$, $\beta_{\max}$, τ, T, ν_t; **Output:** μ^*, ζ, η
Initialize: $\beta = \beta_{\min}$, $\mu_t = \frac{1}{|\mathcal{A}|}$, and $\zeta, \eta, G_\zeta^\beta, G_\eta^\beta$,
$\quad K_\zeta^\beta, L_\eta^\beta, \bar\Lambda_\beta$ to 0.
while $\beta \leq \beta_{\max}$ **do**
$\quad$ Use Algorithm 1 to obtain $\mu_{\beta,\zeta\eta}^*$ at given ζ, η, β.
$\quad$ Consider *env1(ζ,η)*, *env2(ζ',η')*; set $\zeta' = \zeta$, $\eta' = \eta$
$\quad$ **while** $\{\zeta_s\}$, $\{\eta_a\}$ *converge* **do**
$\quad\quad$ **for** $\forall s \in \mathcal{S}$ **do**
$\quad\quad\quad$ **for** *episode* $= 1$ *to* T **do**
$\quad\quad\quad\quad$ reset *env1, env2* at state x_t,
$\quad\quad\quad\quad$ **while** *True* **do**
$\quad\quad\quad\quad\quad$ sample action $u_t \sim \mu^*(\cdot|x_t)$.
$\quad\quad\quad\quad\quad$ *env1:* obtain c_t, x_{t+1}.
$\quad\quad\quad\quad\quad$ *env2:* set $\zeta_s' = \zeta_s + \Delta\zeta_s$, get c_t', x_{t+1}.
$\quad\quad\quad\quad\quad$ update $G_{\zeta_s}^{t+1}(x_t)$ with $\frac{\partial c_{x_t x_{t+1}}^{u_t}}{\partial \zeta_s} \approx \frac{c_t' - c_t}{\Delta\zeta_s}$.
$\quad\quad\quad\quad\quad$ break if $x_{t+1} = \delta$; $t \leftarrow t+1$.
$\quad$ Use same approach as above to learn $G_{\eta_a}^\beta$ update $\{\zeta_s\}$, $\{\eta_a\}$ via gradient descent in (24).
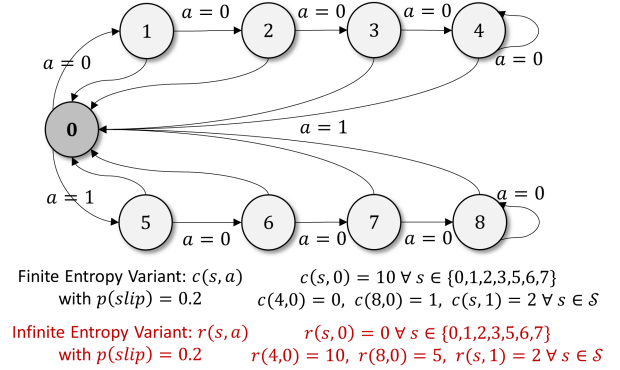$\quad$ $\beta \leftarrow \tau\beta$

Fig. 2. Finite and infinite entropy variants (respectively, cost and reward variants) of Double Chain (DC) environment. State space $\mathcal{S} = \{0, 1, \ldots, 8\}$ and action space $\mathcal{A} = \{0, 1\}$. Cost $c(s, a)$ for version (a) is denoted in black and reward $r(s, a)$ for version (b) is in red. The agent slips to the state $s' = 0$ with probability 0.2 every time the action $a = 0$ is taken at the state $s \in \mathcal{S}\backslash\{4\}$ in the finite entropy variant (and $s \in \mathcal{S}$ for the infinite entropy version (b)). The state $s = 4$ is the cost-free termination state for version (a).

be used to compute the derivatives using finite difference methods.

Remark. Parameterized MDPs with infinite Shannon entropy H^μ can be analogously addressed using above methodology.

VI. SIMULATIONS

In this section we demonstrate our algorithms to (a) determine the control policy μ^* for the finite and infinite Shannon entropy variants of the double chain (DC) environments in Figure 2, the gridworld environment in Figure 3(c3), and (b) solve the parameterized MDP associated to the design of 5G small cell network illustrated in Figure 4. We compare our MEP-based Algorithm 1 with the benchmark algorithms Soft Q-learning [15], Q-learning [6] and Double Q-learning [14]. Note that our choice of the benchmark algorithm Soft Q-learning (entropy regularized RL presented in [15]) is based
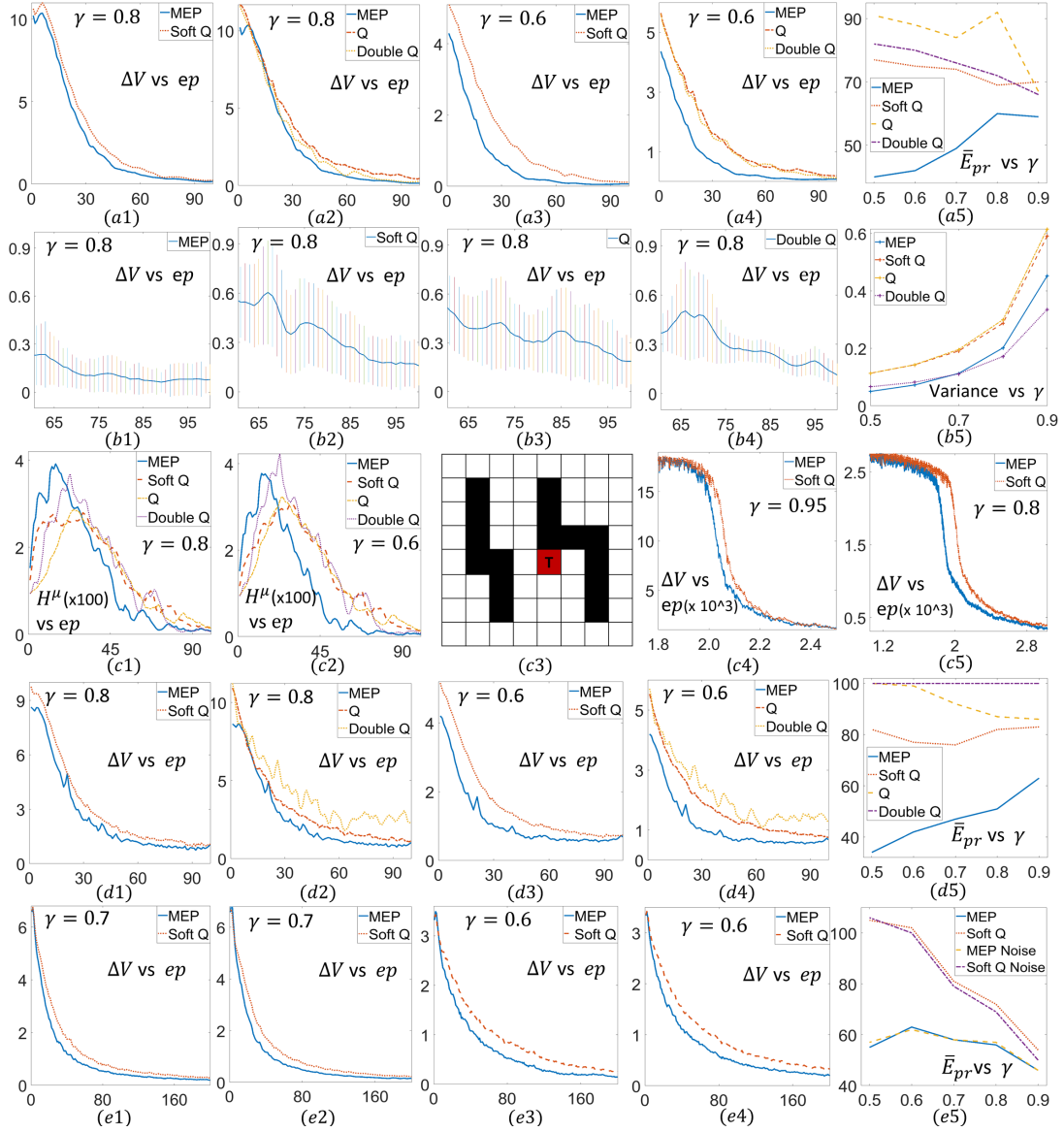
Fig. 3. Performance of MEP-based algorithm: Illustrations on Double Chain Environment in Figure 2. (a1)-(a4) Finite Entropy variant: Illustrates faster convergence of Algorithm 1 (MEP) at different γ values. (a5) Demonstrates faster and increasing rates of convergence of Algorithm 1 (MEP) with decreasing γ values. (b1)-(b4) Illustrates higher variance of Soft Q and Q learning in comparison to Double Q and MEP. (b5) Illustrates increasing *average* variance in learning with increasing γ; Soft Q and Q exhibit higher variances over MEP and Double Q. (c1)-(c2) MEP exhibits larger Shannon Entropy (equivalently, exploration) during early episodes and smaller Shannon Entropy (equivalently, more exploitation) in later episodes; together these are responsible for faster convergence of MEP as illustrated in Fig. 3(a1)-(a5). (c3) Gridworld in [15]: Black cells are inaccessible, red cell 'T' is terminal state, $\mathcal{S}$ includes each cell, $\mathcal{A}$ comprises of 9 actions - vertical, horizontal, diagonal movements and no movement. Each action incurs a cost of 1, and agent slips vertically and horizontally with $p = 0.15$ and diagonally with $p = 0.05$ incurring additional cost of 1. (c4)-(c5) Illustrates faster convergence of MEP on Gridworld environment in [15] for different γ. (d1)-(d5) Cost Version (with added Gaussian noise) : Similar observations as in (a1)-(a5) with significantly higher instability in learning with Double Q algorithm. (e1)-(e5) *Infinite Entropy Variant* : Demonstrates faster convergence of Algorithm 1 (MEP) to J^* without noise in (e1),(e3) and with added noise in (e2),(e4). (e5) Illustrates the consistent faster convergence rates of MEP across different γ values.

on its commonality to our framework as discussed in the Section III-B, and the choice of algorithms Q-learning and Double Q-learning is based on their widespread utility in the literature. Also, note that the work done in [15] already establishes the efficacy of the Soft Q-learning algorithm over the following algorithms in literature Q-learning, Double Q-learning, Ψ-learning [44], Speedy Q-learning [45], and the consistent Bellman Operator $\mathcal{T}_C$ of [46]. Below we highlight features and advantages of our MEP-based Algorithm 1.

Faster Convergence to Optimal J^*: Figures 3(a1)-(a4) (finite entropy variant of DC) and Figures 3 (e1)-(e4) (infinite

entropy variant of DC) illustrate the faster convergence of our MEP-based Algorithm 1 for different discount factor γ values. Here the learning error is given by ΔV at each episode between the value function V_β^μ corresponding to policy $\mu = \mu(ep)$ in the episode ep and the optimal value function J^*; that is,

$$\Delta V(ep) = \frac{1}{N} \sum_{i=1}^{N} \sum_{s \in \mathcal{S}} \left| V_{\beta,i}^{\mu(ep)}(s) - J^*(s) \right|, \quad (27)$$

where N denotes the total experimental runs and i indexes the value function $V_{\beta,i}^\mu$ for each run. As observed in Figures
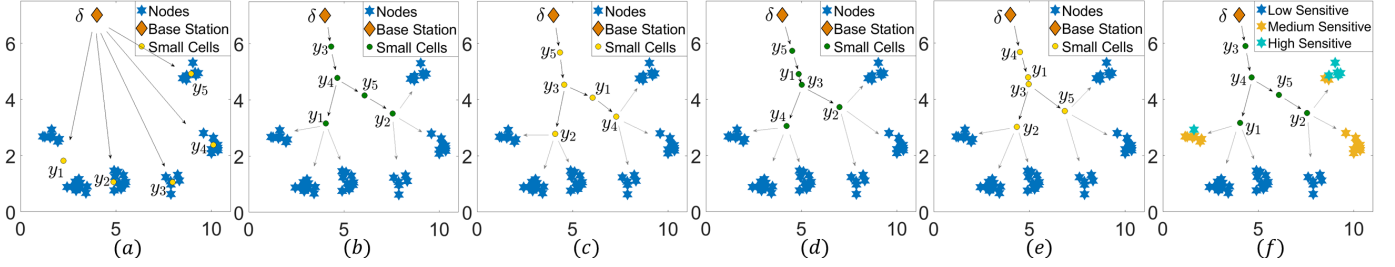
Fig. 4. Parameterized MDPs and RL (a) Illustrates user nodes $\{n_i\}$, base state station δ, and small cell locations $\{y_j\}$ and communication routes determined using a straightforward sequential methodology. (b)-(c) demonstrate small cells at $\{y_j\}$ and communication routes (as illustrated by arrows) resulting from policy obtained from Algorithm 2 (model-based) and Algorithm 3 (model-free), respectively when the $p^a_{ss'}$ is deterministic. (d)-(e) solutions obtained using Algorithm 2 and Algorithm 3, respectively when $p^a_{ss'}$ is probabilistic. (f) sensitivity analysis of the solutions with respect to user node locations $\{x_i\}$.

3(a1) and 3(a3), and Figures 3(e1) and 3(e3), our Algorithm 1 converges even faster as the discount factor γ decreases. We characterize the faster convergence rates also in terms of the convergence time - more precisely the *critical* episode $\bar{E}_{pr}$ beyond which learning error ΔV is within 5% of the optimal J^* (see Figures 3(a5) and 3(e5)). As is observed in the figures, the performance of our (MEP-based) algorithms gets even better with decreasing γ values as the stochastic policy μ^*_β optimizes the amount of exploration required along the paths based on γ values, whereas the other algorithms show deterioration in performance as γ decreases. These results persist in problems with larger state and action spaces, and with more stochasticity in terms of $p^a_{ss'}$, where learning the optimal policy becomes more challenging. For instance, our results with the Gridworld example [15], as shown in Figures 4(c3)-(c5), demonstrate that our algorithm converges faster than the Soft Q learning algorithm proposed in [15] for each distinct value of discount factor γ.

Smaller Variance: Figures 3(b1)-(b4) compare the variance observed in the learning process during the last 40 epsiodes of the N experimental runs. As demonstrated, the Soft Q (Fig. 3(b)) and Q learning (Fig. 3(c)) algorithms exhibit higher variances when compared to to the MEP-based (Fig. 3(a)) and Double Q learning (Fig. 3(d)) algorithms. The Figure 3(b5) plots the *average* variances of all the algorithms in the last 40 episodes of the learning process with respect to different values of γ. As illustrated, Soft Q and Q learning exhibit higher variances across all γ values in comparison to our and Double Q algorithms. Between our and Double Q, the variance in our algorithm is smaller at lower values of γ and vice-versa.

Inherently better exploration and exploitation: The Shannon entropy H^μ corresponding to our algorithm is higher during the initial episodes indicating a better exploration under the learned policy μ, when compared to other algorithms as seen in Figures 3(c1)-(c2). Additionally, it exhibits smaller H^μ in the later episodes indicating a more exploitative nature of the learned policy μ. Unbiased policies resulting from our algorithm along with enhanced exploration-exploitation trade-off results into the faster convergence rates, and smaller variance as discussed above.

Robustness to noise in data: Figures 3(d1)-(d5) demonstrate robustness to noisy environments; here the instantaneous cost $c(s, a)$ in the finite horizon variant of DC is noisy. For the purpose of simulations, we add Gaussian noise $\mathcal{N}(0, \sigma^2)$ with $\sigma = 5$ when $a = 1, s \in \mathcal{S}$, $a = 0, s = 8$, otherwise $\sigma = 10$.

Similar to our observations and conclusions in Figures 3(a1)-(a4), Figures 3(d1)-(d2) ($\gamma = 0.8$) and Figures 3(d3)-(d4) ($\gamma = 0.6$) illustrate faster convergence of our MEP-based algorithm. Also, similar to the Figure 3(a5), Figure 3(d5) exhibits higher convergence rates (i.e. lower $\bar{E}_{pr}$) $\forall \gamma$ values in comparison to benchmark algorithms. In the infinite entropy variant of DC, results of similar nature are obtained upon adding the noise $\mathcal{N}(0, \sigma^2)$ as illustrated in Figure 3(e2) and 3(e4).

Simultaneously determining the unknown parameters and policy in Parameterized MDPs: We design the 5G Small Cell Network (see Figure 1) both when the underlying model ($c^a_{ss'}$ and $p^a_{ss'}$) is known (using Algorithm 2) and as well as unknown (using Algorithm 3). In our simulations we randomly distribute 46 user nodes $\{n_i\}$ at $\{x_i\}$ and the base station δ at z in the domain $\Omega \subset \mathbb{R}^2$ as shown in Figure 4(a). The objective is to allocate $\{f_j\}^5_{j=1}$ small cells and determine the corresponding communication routes. Here, the parameterized MDP comprises of $\mathcal{S} = \{n_1, \ldots, n_{46}, f_1, \ldots, f_5\}$, $\mathcal{A} = \{f_1, \ldots, f_5\}$, and the cost function $c(s, a) = \|\rho(s) - \rho(a)\|^2_2$ where $\rho(\cdot)$ denotes the spatial location. We consider two cases where (a) $p^a_{ss'}$ is deterministic, i.e. an action a at the state s results into $s' = a$ with probability 1, and (b) $p^a_{ss'}$ is probabilistic such that action a at the state s results into $s' = a$ with probability 0.9 or to the state $s' = f_1$ with probability 0.1. Additionally, due to absence of any prior work in literature on network design problems modelled as parameterized MDPs, we compare our results with the solution resulting from a straightforward sequential methodology (as shown in Figure 4(a)) where we first partition the user nodes into 5 distinct clusters to allocate a small cells in each cluster, and then determine optimal communication routes in the network.

Deterministic $p(s, a, s')$: Figure 4(b) illustrates the allocation of small cells and the corresponding communication routes (ascertained using the resulting optimal policy μ^*) as determined by the Algorithm 2. For instance, the route $\delta \rightarrow y_3 \rightarrow y_4 \rightarrow y_1 \rightarrow n_i$ carries the communication packet from the base station δ to the respective user nodes n_i as indicated by the grey arrow from y_1. The cost incurred by the network design in Figure 4(b) is approximately 180% lesser than that in Figure 4(a) clearly indicating the advantage obtained from simultaneously determining the parameters and policy over a sequential methodology. In the model-free RL setting where the functions $c(s, a)$, $p^a_{ss'}$, and the locations $\{x_i\}$ of the user nodes $\{n_i\}$ are not known, we employ our Algorithm 3 to determine the small cell locations $\{y_j\}^5_{j=1}$

and as well as the optimal policy $\{\mu^*(a|s)\}$ as demonstrated in the Figure 4(c). It is evident from Figures 4(b) and (c) that the solutions obtained when the model is completely known and unknown are approximately same. In fact, the solutions obtained differ only by 1.9% in terms of the total cost $\sum_{s\in\mathcal{S}} J^\mu_{\zeta\eta}(s)$ (22) incurred, clearly indicating the efficacy of our model-free learning Algorithm 3.

Probabilistic $p(s, a, s')$: Figure 4(d) illustrates the solution as obtained by our Algorithm 2 when the underlying model ($c(s, a)$, $p^a_{ss'}$, and $\{x_i\}$) is known. Here, the cost associated to the network design is approximately 127% lesser than in Figure 4(a). Figure 4(e) illustrates the solution as obtained by the Algorithm 3 for the model-free case ($c(s, a)$, $p^a_{ss'}$, and $\{x_i\}$ are unknown). Similar to the above scenario, the solutions obtained for this case using the Algorithms 2 and 3 are also approximately the same and differ only by 0.3% in terms of the total cost $\sum_s J^\mu_{\zeta\eta}(s)$ incurred; thereby, substantiating the efficacy of our proposed model-free learning Algorithm 3.

Sensitivity Analysis: Our algorithms enable categorizing the user nodes $\{n_i\}$ in Figure 4(b) into the categories of (i) low, (ii) medium, and (iii) high sensitiveness such that the final solution is least susceptible to the user nodes in (i) and most susceptible to the nodes in (iii). Note that the above sensitivity analysis requires to compute the derivative $\sum_{s'} \partial V^\mu_\beta(s')/\partial\zeta_s$, and we determine it by solving for the fixed point of the Bellman equation in (25). The derivative $\sum_{s'} \partial V^\mu_\beta(s')/\partial\zeta_s$ computed at $\beta \to \infty$ is a measure of sensitivity of the solution to the cost function $\sum_s J^\mu_{\zeta\eta}(s)$ in (22) since V^μ_β in (11) is a smooth approximation of $J^\mu_{\zeta\eta}(s)$ in (22) and $V^\mu_\beta \to J^\mu_{\zeta\eta}(s)$ as $\beta \to \infty$. A similar analysis for Figure 4(c)-(e) can be done if the locations $\{x_i\}$ of the user nodes $\{n_i\}$ are known to the *agent*. The sensitivity of the final solution to the locations $\{y_j\}$, z of the small cells and the base station can also be determined in a similar manner.

VII. ANALYSIS AND DISCUSSION

1) Mutual Information Minimization: The optimization problem (5) maximizes the Shannon entropy $H^\mu(s)$ under a given constraint on the value function J^μ. We can similarly pose and solve the mutual information minimization problem that requires to determine the distribution $p_{\mu^*}(\mathcal{P}|s)$ (with control policy μ^*) over the paths of the MDP that is close to some given prior distribution $q(\mathcal{P}|s)$ [16], [17]. Here, the objective is to minimize the KL-divergence $D_{KL}(p_\mu\|q)$ under the constraint $J = J_0$ (as in (5)).

2) Computational complexity: Our MEP-based Algorithm 1 performs exactly the same number of computations as the Soft Q-learning algorithm [15] for each *epoch* (or, iteration) within an episode. In comparison to the Q and Double Q learning algorithms, our proposed algorithm, apart from performing the additional minor computations of explicitly determining μ^* in (9), exhibits the similar number of computational steps.

3) Scheduling β and Phase Transition: In our Algorithms 1, 2, and 3 we geometrically anneal β (i.e. $\beta_{k+1} = \tau\beta_k$, $\tau > 1$) from a small value $\beta_{\min}$ in the first episode to a large value $\beta_{\max}$ in the last episode. Under this scheme, the interval $\beta_{k+1} - \beta_k$ is smaller in the initial episodes; i.e., β_{k+1} is closer

to β_k initially that encourages significant exploration under policy μ^*_β (9). Further, the change from exploratory nature of μ^*_β to exploitative nature occurs drastically in the later episodes due to a larger $\beta_{k+1} - \beta_k$ interval. Several other MEP-based algorithms such as Deterministic Annealing [8], incorporate geometric annealing of β. The underlying idea in [8] is that the solution undergoes significant changes only at certain *critical β_{cr}* (*phase transition*) and shows insignificant changes between two consecutive critical β_{cr}'s. Thus, for all practical purposes geometric annealing of β works well. Similar to [8] our Algorithms 2 and 3 also undergo the phase transition and we are working on its analytical expression.

4) Capacity and Exclusion Constraints: Certain parameterized MDPs may pose capacity or dynamical constraints over its parameters. For instance, each small cell f_j allocated in the Figure 4 can be constrained in capacity to cater to maximum c_j fraction of user nodes in the network. Our framework allows to model such a constraint as $q_\mu(f_j) := \sum_{a,n_i} \mu(a|n_i)p(f_j|a, n_i) \le c_j$ where $q_\mu(f_j)$ measures fraction of user nodes $\{n_i\}$ that connect to f_j. In another scenario, the locations $\{x_i\}$ of the user nodes could be dynamically varying as $\dot{x}_i = f(x, t)$. Subsequently, the resulting policy μ^*_β and the small cells $\{y_j\}$ will also be time varying. Similar to [47], in our framework we can treat the free-energy function V^μ_β in (11) as a *control-Lyapunov* function and determine time varying μ^*_β and $\{y_j\}$ such that $\dot{V}^\mu_\beta \le 0$.

5) Uncertainty in Parameters: Many application areas comprise of states and actions where the associated parameters are uncertain with a known distribution over the set of their possible values. For instance, a user nodes n_i in Figure 4 may have an associated uncertainty in its location x_i owing to measurement errors. Our proposed framework easily incorporates such uncertainties in parameter values. For example, the above uncertainty will result into replacing $c(n_i, s', a)$ with $c'(n_i, s', a) = \sum_{x_i \in X_i} p(x_i|n_i)c(n_i, s', a)$ where $p(x_i|n_i)$ is the distribution over the set X_i of location x_i. The subsequent solution approach remains the same as in Section V-B.

APPENDIX A

Proof of Lemma 1: Let $\bar{x}_0 = s$. By Assumption 1 $\exists$ a path $\omega = (\bar{u}_0, \bar{x}_1, \ldots, \bar{x}_N = \delta)$ such that $p_{\bar{\mu}}(\omega|x_0 = s) > 0$ which implies $p(x_{k+1} = \bar{x}_{k+1}|x_k = \bar{x}_k, u_k = \bar{u}_k) > 0$ by (6). Then, probability $p_\mu(\omega|x_0 = s)$ of taking path ω under the stochastic policy $\mu \in \Gamma$ in (4) is also positive.

Proof of Theorem 1: The following Lemma is needed

Lemma 2. *The Shannon Entropy $H^\mu(\cdot)$ corresponding to the MDP illustrated in Section III-A satisfies the algebraic expression $\sum_{s'} p^a_{ss'} H^\mu(s') = \sum_{s'} p^a_{ss'} \log p^a_{ss'} + \log \mu_{a|s} + \lambda_s$.*

Proof. $H^\mu(\cdot)$ in (5) satisfies the recursive Bellman equation

$$H^\mu(s') = \sum_{a's''} \mu_{a'|s'} p^{a'}_{s's''} \left[-\log p^{a'}_{s's''} - \log \mu_{a'|s'} + H^\mu(s'') \right]$$

On the right side of the above Bellman equation, we subtract a zero term $\sum_s \lambda_s(\sum_a \mu_{a|s} - 1)$ that accounts for normalization constraint $\sum_a \mu_{a|s} = 1 \ \forall \ s$ and λ_s are some constants. Taking

the derivative of the resulting expression with respect to $\mu_{a|s}$ we obtain

$$\frac{\partial H^\mu(s')}{\partial \mu_{a|s}} = \rho(s,a)\delta_{ss'} + \sum_{a',s''} \mu_{a'|s'} p^{a'}_{s's''} \frac{\partial H^\mu(s'')}{\partial \mu_{a|s}} - \lambda_s, \quad (28)$$

where $\rho(s,a) = -\sum_{s''} p^a_{ss''}(\log p^a_{ss''} - H^\mu(s'')) - \log \mu_{a|s}$. The subsequent steps in the proof involve algebraic manipulations and makes use of the quantity $p_\mu(s') := \sum_s p_\mu(s'|s)$ where $p_\mu(s'|s) = \sum_a p^a_{ss'} \mu_{a|s}$. Under the trivial assumption that for each state s' there exists a state-action pair (s,a) such that the probability of the system to enter the state s' upon taking action a in the state s is non-zero (i.e. $p^a_{ss'} > 0$) we have that $p_\mu(s') > 0$. Now, we multiply equation (28) by $p_\mu(s')$ and add over all $s' \in \mathcal{S}$ to obtain

$$\sum_{s'} p_\mu(s') \frac{\partial H^\mu(s')}{\partial \mu_{a|s}} = p_\mu(s')\rho(s,a) + \sum_{s''} p_\mu(s'')\frac{\partial H^\mu(s'')}{\partial \mu_{a|s}} - \lambda_s,$$

where $p_\mu(s'') = \sum_{s'} p_\mu(s')p_\mu(s''|s')$. The derivative terms on both sides cancel to give $p_\mu(s')\rho(s,a) - \lambda_s = 0$ which implies $\sum_{s'} p^a_{ss'} H^\mu(s') = \sum_{s'} p^a_{ss'} \log p^a_{ss'} + \log \mu_{a|s} + \lambda_s$. □

Now consider the free energy function $V^\mu_\beta(s)$ in (7) and separate out the $t = 0$ term in its infinite summation to obtain

$$V^\mu_\beta(s) = \sum_{a,s'} \mu_{a|s} p^a_{ss'}\left[\hat{c}^a_{ss'} + \frac{1}{\beta}\log \mu_{a|s} + \gamma V^\mu_{\gamma\beta}(s')\right], \quad (29)$$

where $\hat{c}^a_{ss'} = c^a_{ss'} + \frac{1}{\beta}\log p^a_{ss'}$ and $V^\mu_{\gamma\beta}(s') = V^\mu_\beta(s') - \frac{1-\gamma}{\gamma\beta}H(s')$. Substituting $V_{\gamma\beta}(s')$ and the algebraic expression obtained in Lemma 2 in the above equation (29) we obtain

$$V^\mu_\beta(s) = \sum_{a,s'} \mu_{a|s} p^a_{ss'}\left[\bar{c}^a_{ss'} + \frac{\gamma}{\beta}\log \mu_{a|s} + \gamma V^\mu_\beta(s')\right].$$

Appendix B

Proof of Theorem 2: The following lemma will be used.

Lemma 3. *For every policy $\mu \in \Gamma$ defined in (4) there exists a vector $\xi = (\xi_s) \in \mathbb{R}^{|\mathcal{S}|}_+$ with positive components and a scalar $\lambda < 1$ such that $\sum_{s'} p^a_{ss'}\xi_{s'} \le \lambda \xi_s$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$.*

Proof. Consider a new MDP with state transition probabilities similar to the original MDP and the transition costs $c^a_{ss'} = -1 - \frac{1}{\beta}\log(|\mathcal{A}||\mathcal{S}|)$ except when $s = \delta$. Thus, the free-energy function $V^\mu_\beta(s)$ in (7) for the new MDP is less than or equal to -1. We define $-\xi_s \triangleq V^*_\beta(s)$ (as given in 11)) and use LogSumExp [48] inequality to obtain $-\xi_s \le \min_a \Lambda_\beta(s,a) \le \Lambda_\beta(s,a) \; \forall \; a \in \mathcal{A}$ where $\Lambda_\beta(s,a)$ is the state action value function in (12). Thus, $-\xi_s \le \sum_{s'} p^a_{ss'}\left(c^a_{ss'} + \frac{\gamma}{\beta}\log p^a_{ss'} - \gamma\xi_{s'}\right)$ and upon substituting $c^a_{ss'}$ we obtain $-\xi_s \le -1 - \gamma\sum_{s'} p^a_{ss'}\xi_{s'} \le -1 - \sum_{s'} p^a_{ss'}\xi_{s'}$.

$$\Rightarrow \sum_{s'\in\mathcal{S}} p^a_{ss'}\xi_{s'} \le \xi_s - 1 \le \left[\max_s \frac{\xi_s - 1}{\xi_s}\right]\xi_s =: \lambda\xi_s.$$

Since $V^*_\beta(s) \le -1 \Rightarrow \xi_s - 1 \ge 0$ and thus $\lambda < 1$. □

Next we show that $T : \Lambda_\beta \to \Lambda_\beta$ in (12) is a contraction map. For any $\hat{\Lambda}_\beta$ and $\check{\Lambda}_\beta$ we have that $[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)$

$$= -\frac{\gamma^2}{\beta}\sum_{s'\in\mathcal{S}} p^a_{ss'}\log\frac{\sum_a \exp\left(-\frac{\beta}{\gamma}\hat{\Lambda}_\beta(s',a)\right)}{\sum_{a'}\exp\left(-\frac{\beta}{\gamma}\check{\Lambda}_\beta(s',a')\right)}$$

$$\ge \gamma\sum_{s',a'} p^a_{ss'}\hat{\mu}_{a'|s'}(\hat{\Lambda}_\beta(s',a') - \check{\Lambda}_\beta(s',a')) =: \gamma\Delta_{\hat{\mu}}, \quad (30)$$

where we use the Log sum inequality to obtain (30), and $\hat{\mu}_{a|s}$ is the stochastic policy in (9) corresponding to $\hat{\Lambda}_\beta(s,a)$. Similarly, we obtain $[T\check{\Lambda}_\beta - T\hat{\Lambda}_\beta](s,a) \ge -\gamma\sum_{s',a'} p^a_{ss'}\check{\mu}_{a'|s'}(\hat{\Lambda}_\beta(s',a') - \check{\Lambda}_\beta(s',a')) =: -\gamma\Delta_{\check{\mu}}$ where $\check{\mu}_{a|s}$ is the policy in (9) corresponding to $\check{\Lambda}_\beta(s,a)$. Now from $\gamma\Delta_{\hat{\mu}} \le [T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a) \le \gamma\Delta_{\check{\mu}}$ we conclude that $|[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)| \le \gamma\Delta_{\bar{\mu}}(s,a)$ where $\Delta_{\bar{\mu}}(s,a) = \max\{|\Delta_{\hat{\mu}}(s,a)|,|\Delta_{\check{\mu}}(s,a)|\}$ and we have $|[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)|$

$$\le \gamma\sum_{s',a'} p^a_{ss'}\bar{\mu}_{a'|s'}|\hat{\Lambda}_\beta(s',a') - \check{\Lambda}_\beta(s',a')| \quad (31)$$

$$\le \gamma\sum_{s',a'} p^a_{ss'}\xi_{s'}\bar{\mu}_{a'|s'}\|\hat{\Lambda}_\beta - \check{\Lambda}_\beta\|_\xi \quad (32)$$

where $\|\Lambda_\beta\|_\xi = \max_{s,a}\frac{\Lambda_\beta(s,a)}{\xi_s}$ and $\xi \in \mathbb{R}^\mathcal{S}$ is as given in Lemma 3. Further, from the same Lemma we obtain

$$|[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)| \le \gamma\lambda\xi_s\sum_{a'\in\mathcal{A}}\bar{\mu}_{a'|s'}\|\hat{\Lambda}_\beta - \check{\Lambda}_\beta\|_\xi \quad (33)$$

$$\Rightarrow \|T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta\|_\xi \le \gamma\lambda\|\hat{\Lambda}_\beta - \check{\Lambda}_\beta\|_\xi \text{ with } \gamma\lambda < 1 \quad (34)$$

Appendix C

Proof of Theorem 3: The proof follows the similar idea as the proof for Theorem 1 in Appendix A and thus, we do not explain it in detail to avoid redundancy except the following Lemma that illustrates the algebraic structure of the discounted Shannon entropy $H^\mu_d(\cdot)$ in (16) which is different from that in Lemma 2 and also required in our proof of the said theorem.

Lemma 4. *The discounted Shannon entropy $H^\mu_d(\cdot)$ corresponding to the MDP in Section IV satisfies the algebraic term $\alpha\sum_{s'} p^a_{ss'}H^\mu_d(s') = \sum_{s'} p^a_{ss'}\log p^a_{ss'} + \log\alpha\mu(a|s) + \lambda_s$.*

Proof. Define a new MDP that augments the action and state spaces $(\mathcal{A},\mathcal{S})$ of the original MDP with an additional action a_e and state s_e, respectively, and derives its state-transition probability $\{q^a_{ss'}\}$ and policy $\{\zeta_{a|s}\}$ from original MDP as

$$q^a_{ss'} = \begin{cases} p^a_{ss'} & \forall s,s' \in \mathcal{S}, a \in \mathcal{A} \\ 1 & \text{if } s', a = s_e, a_e \\ 1 & \text{if } s' = s = s_e \\ 0 & \text{otherwise} \end{cases} \quad \zeta_{a|s} = \begin{cases} \alpha\mu_{a|s} & \forall(s,a) \in (\mathcal{S},\mathcal{A}) \\ 1-\alpha & \text{if } a = a_e, s \in \mathcal{S} \\ 0 & \text{if } a \in \mathcal{A}, s = s_e \\ 1 & \text{if } a = a_e, s = s_e \end{cases}$$

Next, we define $T^\mu := \alpha H^\mu_d$ that satisfies $T^\mu(s') = \sum_{a's''} \eta_{a'|s'} p^{a'}_{s's''}\left[-\log p^a_{s's''} - \log\eta_{a'|s'} + T^\mu(s'')\right]$ derived using (16) where $\eta_{a'|s'} = \alpha\mu_{a'|s'}$. The subsequent steps of the proof are same as the proof of Lemma 2. □

Appendix D

Proof of Proposition 1: The proof in this section is analogous to the proof of Proposition 5.5 in [2]. Let $\bar{T}$ be the map in (15). The stochastic iterative updates in (14) can be re-written as $\bar{\Psi}_{t+1}(x_t, u_t) =$

$(1-\nu_t(x_t,u_t))\bar{\Psi}_t(x_t,u_t)+\nu_t(x_t,u_t)\left(\left[\bar{T}\bar{\Psi}_t\right](x_t,u_t)+w_t(x_t,u_t)\right)$ where $w_t(x_t,u_t)=c^{u_t}_{x_t x_{t+1}}-\frac{\gamma^2}{\beta}\log\sum_a\exp(-\frac{\beta}{\gamma}\bar{\Psi}_t(s_{t+1},a))-\bar{T}\bar{\Psi}_t(x_t,u_t)$. Let $\mathcal{F}_t$ represent the history of the stochastic updates, i.e., $\mathcal{F}_t=\{\bar{\Psi}_0,...,\bar{\Psi}_t,w_0,...,w_{t-1},\nu_0,...,\nu_t\}$, then $\mathbb{E}[w_t(x_t,u_t)|\mathcal{F}_t]=0$ and $\mathbb{E}[w_t^2(x_t,u_t)|\mathcal{F}_t]\leq K(1+\max_{s,a}\bar{\Psi}_t^2(s,a))$, where K is a constant. These expressions satisfy the conditions on the expected value and the variance of $w_t(x_t,u_t)$ that along with the contraction property of $\bar{T}$ guarantees the convergence of the stochastic updates (14) as illustrated in the Proposition 4.4 in [2].

Proof of Theorem 4: We show that the map T_1 in (25) is a contraction map. For any $K^{\beta}_{\zeta_s}$ and $\bar{K}^{\beta}_{\zeta_s}$ we obtain that $|[T_1 K^{\beta}_{\zeta_s} - T_1\bar{K}^{\beta}_{\zeta_s}](s')| \leq \gamma\sum_{a,s''}p^a_{s's''}\mu_{a|s'}|K^{\beta}_{\zeta_s}(s'',a) - \bar{K}^{\beta}_{\zeta_s}(s'',a)|$. Note that this inequality is similar to the one in (31); thus, we follow the exact same steps from (31) to (34) to show that $\|T_1 K^{\beta}_{\zeta_s}-T_1\bar{K}^{\beta}_{\zeta_s}\|_\xi \leq \gamma\lambda\|K^{\beta}_{\zeta_s}-\bar{K}^{\beta}_{\zeta_s}\|_\xi$ and $\gamma\lambda < 1$.

Proof of Proposition 2: The proof in this section is similar to the proof of Proposition 1 in Appendix D. Additional conditions on the boundedness of the derivatives $\left|\frac{\partial c^a_{ss'}}{\partial \zeta_l}\right|$ and $\left|\frac{\partial c^a_{ss'}}{\partial \eta_k}\right|$ are required to bound the variance $\mathbb{E}[w_t^2|\mathcal{F}_t]$.

References

[1] E. A. Feinberg and A. Shwartz, *Handbook of Markov decision processes: methods and applications*. Springer Science & Business Media, 2012, vol. 40.

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-dynamic programming*. Athena Scientific Belmont, MA, 1996, vol. 5.

[3] A. Hordijk and L. Kallenberg, "Linear programming and markov decision chains," *Management Science*, vol. 25, no. 4, pp. 352–362, 1979.

[4] Y. Abbasi-Yadkori, P. L. Bartlett, and A. Malek, "Linear programming for large-scale markov decision problems," in *JMLR Workshop and Conference Proceedings*, no. 32. MIT Press, 2014, pp. 496–504.

[5] D. P. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations research*, vol. 51, no. 6, pp. 850–865, 2003.

[6] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[7] E. T. Jaynes, "Information theory and statistical mechanics," *Physical review*, vol. 106, no. 4, p. 620, 1957.

[8] K. Rose, "Deterministic annealing, clustering, and optimization," Ph.D. dissertation, California Institute of Technology, 1991.

[9] P. Sharma, S. Salapaka, and C. Beck, "A scalable approach to combinatorial library design for drug discovery," *Journal of chemical information and modeling*, vol. 48, no. 1, pp. 27–41, 2008.

[10] J.-G. Yu, J. Zhao, J. Tian, and Y. Tan, "Maximal entropy random walk for region-based visual saliency," *IEEE transactions on cybernetics*, vol. 44, no. 9, pp. 1661–1672, 2013.

[11] Y. Xu, S. M. Salapaka, and C. L. Beck, "Aggregation of graph models and markov chains by deterministic annealing," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2807–2812, 2014.

[12] L. Chen, T. Zhou, and Y. Tang, "Protein structure alignment by deterministic annealing," *Bioinformatics*, vol. 21, no. 1, pp. 51–62, 2005.

[13] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning." in *Aaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.

[14] H. V. Hasselt, "Double q-learning," in *Advances in Neural Information Processing Systems*, 2010, pp. 2613–2621.

[15] R. Fox, A. Pakman, and N. Tishby, "Taming the noise in reinforcement learning via soft updates," *arXiv preprint arXiv:1512.08562*, 2015.

[16] J. Grau-Moya, F. Leibfried, and P. Vrancx, "Soft q-learning with mutual-information regularization," 2018.

[17] J. Peters, K. Mulling, and Y. Altun, "Relative entropy policy search," in *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[18] G. Neu, A. Jonsson, and V. Gómez, "A unified view of entropy-regularized markov decision processes," *arXiv preprint arXiv:1705.07798*, 2017.

[19] K. Asadi and M. L. Littman, "An alternative softmax operator for reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 243–252.

[20] O. Nachum, M. Norouzi, K. Xu, and D. Schuurmans, "Bridging the gap between value and policy based reinforcement learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 2775–2785.

[21] B. Dai, A. Shaw, L. Li, L. Xiao, N. He, Z. Liu, J. Chen, and L. Song, "Sbeed: Convergent reinforcement learning with nonlinear function approximation," *arXiv preprint arXiv:1712.10285*, 2017.

[22] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*, 2015, pp. 1889–1897.

[23] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *arXiv preprint arXiv:1801.01290*, 2018.

[24] L. P. Hansen, T. J. Sargent, G. Turmuhambetova, and N. Williams, "Robust control and model misspecification," *Journal of Economic Theory*, vol. 128, no. 1, pp. 45–90, 2006.

[25] Z. Zhou, M. Bloem, and N. Bambos, "Infinite time horizon maximum causal entropy inverse reinforcement learning," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 2787–2802, 2017.

[26] A. Aguilar-Garcia, S. Fortes, M. Molina-García, J. Calle-Sánchez, J. I. Alonso, A. Garrido, A. Fernández-Durán, and R. Barco, "Location-aware self-organizing methods in femtocell networks," *Computer Networks*, vol. 93, pp. 125–140, 2015.

[27] U. Siddique, H. Tabassum, E. Hossain, and D. I. Kim, "Wireless backhauling of 5g small cells: Challenges and solution approaches," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 22–31, 2015.

[28] G. Manganini, M. Pirotta, M. Restelli, L. Piroddi, and M. Prandini, "Policy search for the optimal control of markov decision processes: A novel particle-based iterative scheme," *IEEE transactions on cybernetics*, vol. 46, no. 11, pp. 2643–2655, 2015.

[29] A. Srivastava and S. M. Salapaka, "Simultaneous facility location and path optimization in static and dynamic networks," *IEEE Transactions on Control of Network Systems*, pp. 1–1, 2020.

[30] M. Mahajan, P. Nimbhorkar, and K. Varadarajan, "The planar k-means problem is np-hard," in *International Workshop on Algorithms and Computation*. Springer, 2009, pp. 274–285.

[31] L. Xia and Q.-S. Jia, "Parameterized markov decision process and its application to service rate control," *Automatica*, vol. 54, pp. 29–35, 2015.

[32] M. Hausknecht and P. Stone, "Deep reinforcement learning in parameterized action space," *arXiv preprint arXiv:1511.04143*, 2015.

[33] W. Masson, P. Ranchod, and G. Konidaris, "Reinforcement learning with parameterized actions," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[34] E. Wei, D. Wicke, and S. Luke, "Hierarchical approaches for reinforcement learning in parameterized action space," in *2018 AAAI Spring Symposium Series*, 2018.

[35] J. Xiong, Q. Wang, Z. Yang, P. Sun, L. Han, Y. Zheng, H. Fu, T. Zhang, J. Liu, and H. Liu, "Parametrized deep q-networks learning: Reinforcement learning with discrete-continuous hybrid action space," *arXiv preprint arXiv:1810.06394*, 2018.

[36] V. Narayanan and S. Jagannathan, "Event-triggered distributed control of nonlinear interconnected systems using online reinforcement learning with exploration," *IEEE transactions on cybernetics*, vol. 48, no. 9, pp. 2510–2519, 2017.

[37] E. Çilden and F. Polat, "Toward generalization of automated temporal abstraction to partially observable reinforcement learning," *IEEE transactions on cybernetics*, vol. 45, no. 8, pp. 1414–1425, 2014.

[38] W. Shi, S. Song, and C. Wu, "Soft policy gradient method for maximum entropy deep reinforcement learning," *arXiv preprint arXiv:1909.03198*, 2019.

[39] G. Xiang and J. Su, "Task-oriented deep reinforcement learning for robotic skill acquisition and control," *IEEE transactions on cybernetics*, 2019.

[40] E. T. Jaynes, *Probability theory: The logic of science*. Cambridge university press, 2003.

[41] F. Biondi, A. Legay, B. F. Nielsen, and A. Wkasowski, "Maximizing entropy over markov processes," *Journal of Logical and Algebraic Methods in Programming*, vol. 83, no. 5-6, pp. 384–399, 2014.

[42] Y. Savas, M. Ornik, M. Cubuktepe, and U. Topcu, "Entropy maximization for constrained markov decision processes," in *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2018, pp. 911–918.

[43] S. Ross, J. Pineau, B. Chaib-draa, and P. Kreitmann, "A bayesian approach for learning and planning in partially observable markov decision processes," *Journal of Machine Learning Research*, vol. 12, no. May, pp. 1729–1770, 2011.

[44] K. Rawlik, M. Toussaint, and S. Vijayakumar, "Approximate inference and stochastic optimal control," *arXiv preprint arXiv:1009.3958*, 2010.

[45] M. Ghavamzadeh, H. J. Kappen, M. G. Azar, and R. Munos, "Speedy q-learning," in *Advances in neural information processing systems*, 2011, pp. 2411–2419.

[46] M. G. Bellemare, G. Ostrovski, A. Guez, P. S. Thomas, and R. Munos, "Increasing the action gap: New operators for reinforcement learning," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[47] P. Sharma, S. M. Salapaka, and C. L. Beck, "Entropy-based framework for dynamic coverage and clustering problems," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 135–150, 2011.

[48] K. Kobayashi *et al.*, *Mathematics of information and coding*. American Mathematical Soc., 2007, vol. 203.