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

Amber Srivastava and Srinivasa M. Salapaka, *Senior 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 modeled 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 nonconvex with multiple poor local minima. Simulation results applied to a 5G small cell network problem demonstrate the successful determination of communication routes and the small cell locations. We also obtain sensitivity measures to problem parameters and robustness to noisy environment data.**

***Index Terms*—Markov decision processes (MDPs), maximum entropy principle (MEP), network design, parameterized sequential decision making, reinforcement learning.**

## I. Introduction

MARKOV decision processes (MDPs) model sequential decision-making problems which arise in many application areas, such as robotics, 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; where the models for the state dynamics may be partially or completely unknown, and the 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], [4]; and $Q$-learning [5].

In this article, we view MDPs and their variants as combinatorial optimization problems and develop a framework based on the maximum entropy principle (MEP) [6] to address them. MEP has proved successful in addressing a variety of combinatorial optimization problems, such as facility location problems [7], combinatorial drug discovery [8], traveling salesman problem and its variants [7], image processing [9], graph and Markov chain aggregation [10], and protein structure alignment [11]. 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* [12], such that 1) it is the *fairest* distribution—the one with the maximum Shannon entropy $H$ and 2) 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 $\beta$ 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 the 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

Manuscript received 27 May 2020; revised 7 January 2021 and 13 May 2021; accepted 22 July 2021. Date of publication 18 August 2021; date of current version 18 August 2022. This work was supported in part by the National Science Foundation under Grant ECCS (NRI) 18-30639; in part by the U.S. Department of Energy under Award DE-EE0009125; and in part by the Dynamic Research Enterprise for Multidisciplinary Engineering Sciences (DREMES)—collaboration between Zhejiang University and the University of Illinois at Urbana–Champaign. This article was recommended by Associate Editor D. Liu. *(Corresponding author: Amber Srivastava.)*

The authors are with the Mechanical Science and Engineering Department and Coordinated Science Laboratory, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: asrvstv6@illinois.edu; salapaka@illinois.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TCYB.2021.3102510.

Digital Object Identifier 10.1109/TCYB.2021.3102510

the flexibility of algorithms such as deterministic annealing [7] 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 the main contributions 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 [5], Double $Q$-learning [13], Soft $Q$-learning [entropy regularized (ER) $Q$-learning] [14]–[20] in discrete state and action spaces, and trust region policy optimization (TRPO) [21], and soft actor–critic (SAC) [22] 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 [5], [13]–[15] 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 [14] which hinders the learning process and demands 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 the multiplier value is small ($\beta \approx 0$), and becomes increasingly *exploitative* as the multiplier value increases.

*Parameterized MDPs and RL:* These classes of optimization problems are not even necessarily MDPs which contribute 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 [23]; 5G small cell network design [24], [25]; supply chain networks; and last mile delivery problems [26], pose a *parameterized* MDP with a two-fold objective of determining simultaneously 1) the optimal control policy for the underlying stochastic process and 2) the unknown parameters that the state and action variables depend upon such that the cumulative cost is minimized. The latter objective is akin to facility location problem [27]–[29], that is shown to be NP-hard [27], and where the associated cost function (nonconvex) is riddled with multiple poor local minima.

For instance, Fig. 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 $\delta$ via a network of small cells. Here, the state space $\mathcal{S}$ of the underlying MDP is parameterized by the locations $\{y_j\}$ of small cells $\{f_j\}$.

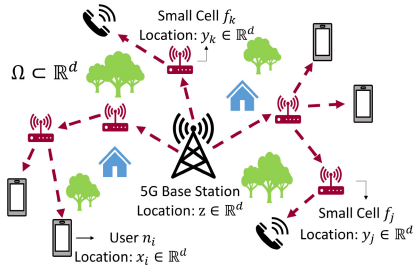


Fig. 1. 5G small cell network. The objective is to determine the small cell location $\{y_j \in \mathbb{R}^d\}$ and the communication routes from the base station $\delta$ to each user $\{n_i\}$ via the network of the small cells.

*Algebraic Structure and Sensitivity Analysis:* In our framework, maximization of Shannon entropy of the distribution over the paths under a 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 the 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 nontrivial 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 local minima in the case of parameterized MDPs.

In the context of model-free RL, we provide comparisons with the benchmark algorithms Q, Double Q, and ER G-learning [14] (also referred to as Soft $Q$-learning). Our algorithms converge at a faster rate (as fast as 1.5 times) than the benchmark algorithms across various values of the discount factor, and even in the case of noisy environments. In the context of parameterized MDPs and RL, we address the small-cell network design problem in 5G communication. Here, the parameters are the unknown locations of the small cells and the control policy determines the routing of the communication packet. 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 nearly the same. We also demonstrate sensitivity analysis, benefits of annealing, and considering entropy of distribution over the paths in our simulations on parameterized MDPs and RL.

This article is organized as follows. We briefly review the related work and MEP [6] in Section II. In Sections III and IV, we develop the MEP-based framework for MDPs. Section V

builds upon Section III to address the case of parameterized MDPs and RL problems. Simulations on a variety of scenarios are presented in Section VI. We discuss the generality of our framework, its capabilities, and future directions of the work in Section VII. For the ease of reading, we provide a comprehensive list of symbols in Section F of the supplementary material.

## II. Preliminaries

*Related Work in Entropy Regularization:* Some of the previous works in RL literature [14]–[20], [30], [31] either use entropy as a regularization term $(-\log \mu(a_t|s_t))$ [14], [15] to the instantaneous cost function $c(s_t, a_t, s_{t+1})$ or maximize the entropy $(-\sum_a \mu(a|s)\log\mu(a|s))$ [16]–[18] associated *only* with the stochastic policy under constraints on the cost $J$. This results in benefits, such as better exploration, overcoming the effect of noise $w_t$ in the instantaneous cost $c_t$, and obtaining faster convergence. However, the resulting stochastic policy and soft-max approximation of the value function $J$ are not in compliance with the MEP applied to the distribution over the paths of MDP. Thus, the resulting stochastic policy is biased in its exploration *over the paths of the MDP*. Our simulations demonstrate the benefit of *unbiased* exploration (in our framework) in terms of faster convergence and better performance in the noisy environment in comparison to the ER benchmark algorithm.

*Related Work in Parameterized MDPs and RL:* The existing solution approaches [2]–[4] 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 to the current state. Other related approaches for parameterized MDPs are case specific; for instance, [32] presents action-based parameterization of state space with applications to service rate control in closed Jackson networks, and [33]–[38] incorporate parameterized actions that are 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 class of parameterized MDPs that we address in this article predominantly originate in network-based applications that involve simultaneous routing and resource allocations and pose additional challenges of nonconvexity and NP-hardness. We address these MDPs in both the scenarios, where the underlying model is known as well as unknown.

*Maximum Entropy Principle:* We briefly review the MEP [6] 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 $f_k : \mathcal{X} \to \mathbb{R}$, $1 \le k \le m$. Then, the most unbiased probability distribution $p_{\mathcal{X}}(\cdot)$ solves

$$\max_{\{p_{\mathcal{X}}(x_i)\}} \; H(\mathcal{X}) = -\sum_{i=1}^{n} p_{\mathcal{X}}(x_i)\ln p_{\mathcal{X}}(x_i)$$

$$\text{subject to } \sum_{i=1}^{n} p_{\mathcal{X}}(x_i) f_k(x_i) = F_k \quad \forall\ 1 \le k \le m \tag{1}$$

where $F_k$, $1 \le k \le m$, are known expected values of the functions $f_k$. The above optimization problem results into Gibbs' distribution [39] $p_{\mathcal{X}}(x_i) = ([\exp\{-\sum_k \lambda_k f_k(x_i)\}]/[\sum_{j=1}^{n}\exp\{-\sum_k \lambda_k f_k(x_j)\}])$, where $\lambda_k$, $1 \le k \le m$, 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 $\delta$. 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 $0 < \gamma \le 1$ 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* $\omega$—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 $t \in \mathbb{Z}_{\ge 0}$ and $x_t = \delta$ for all $t \ge k$ if and when the system reaches the termination state $\delta \in \mathcal{S}$ in $k$ steps. The objective is to determine the optimal policy $\mu^*$ 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 $\omega \in \Omega := \{(u_t, x_{t+1})_{t\in\mathbb{Z}_{\ge 0}} : u_t \in \mathcal{A}, x_t \in \mathcal{S}\}$. 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 nonzero probability to reach the cost-free termination state $\delta$, when starting from any state $s$.

We consider the following set of stochastic policies $\mu$:

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

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

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

*Proof:* Refer to Appendix A. ∎

We use the MEP to determine the policy $\mu \in \Gamma$ such that the Shannon entropy of the corresponding distribution $p_\mu$ is maximized and the state value function $J^{\mu}(s)$ attains a specified

value $J_0$. More specifically, we pose the following optimization problem:

$$\max_{\{p_\mu(\cdot|s)\}:\mu\in\Gamma} \quad H^\mu(s) = -\sum_{\omega\in\Omega} p_\mu(\omega|s)\log p_\mu(\omega|s)$$
$$\text{subject to} \quad J^\mu(s) = J_0. \tag{5}$$

*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 [40] and [41]. In short, the existence of a cost-free termination state $\delta$ and a nonzero probability to reach it from any state $s \in \mathcal{S}$ ensures that the maximum entropy is finite. Refer to [40, Th. 1] or [41, Proposition 2] for further details.

*Remark 1:* Though the optimization problem in (5) considers the stochastic policies $\mu \in \Gamma$, our algorithms presented in the later sections are designed 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 $\omega$ in (2) can be determined from the underlying policy $\mu$ by exploiting the Markov property that dissociates $p_\mu(\omega|s)$ in terms of the policy $\mu$ 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 $\mu$ 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 $V^\mu_\beta(s) = J^\mu(s) - 1/\beta H^\mu(s) =$

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

where $\beta$ is the Lagrange parameter. Here, we have not included the constant value $J_0$ in the cost Lagrangian $V^\mu_\beta(s)$ for simplicity. We refer to the above Lagrangian $V^\mu_\beta(s)$ (7) as the *free-energy* function and $1/\beta$ 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 $\mu^*_\beta$ that minimizes the Lagrangian $V^\mu_\beta(s)$ in (7), we first derive the Bellman equation for $V^\mu_\beta(s)$.

*Theorem 1:* The free-energy function $V^\mu_\beta(s)$ in (7) satisfies the following recursive Bellman equation:

$$V^\mu_\beta(s) = \sum_{a,s'\in\mathcal{A},\mathcal{S}} \mu_{a|s}p^a_{ss'}\left(\bar{c}^a_{ss'} + \frac{\gamma}{\beta}\log\mu_{a|s} + \gamma V^\mu_\beta(s')\right) \tag{8}$$

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

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

Now, the optimal policy satisfies $[\partial V^\mu_\beta(s)/\partial\mu(a|s)] = 0$, which results into Gibb's distribution

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

$$\Lambda_\beta(s,a) = \sum_{s'\in\mathcal{S}} p^a_{ss'}\Big(\bar{c}^a_{ss'} + \gamma V^*_\beta(s')\Big) \tag{10}$$

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

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

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

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

To solve the above implicit equation, we show that the map $T$ in (12) is a contraction map and, therefore, $\Lambda_\beta$ can be obtained using fixed-point iterations, which guarantee converging to the unique fixed point. Consequently, the global minimum $V^*_\beta$ in (11) and the optimal policy $\mu^*_\beta$ 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, that is, $\exists$ 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

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

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

*Proof:* Refer to Appendix B for details. ■

*Remark 2:* It is known from the sensitivity analysis [39] that the value of the Lagrange parameter $\beta$ in (7) is inversely proportional to the constant $J_0$ in (5). Thus, at 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) encourages exploration along the paths of the MDP. As $\beta$ 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 3:* We briefly draw readers' attention to the value function $Y(s) = \mathbb{E}[\sum_{t=0}^{\infty}\gamma^t(c^{u_t}_{x_t x_{t+1}} + (1/\beta)\log\mu_{u_t|x_t})]$ considered in the ER methods [14]. Note that in $Y(s)$ the discounting $\gamma^t$ is multiplied to both the cost term $c^{u_t}_{x_t x_{t+1}}$ 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 $\gamma^t$ *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 [14] does not satisfy MEP applied over the distribution $p_\mu$; consequently their exploration along the paths is not as unbiased as our algorithm.

### 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 *a priori*; however, at each discrete-time instant $t$, the *agent* takes an action $u_t$ under a policy $\mu$ and the *environment* (underlying stochastic process) results into an instantaneous cost $c_{x_t x_{t+1}}^{u_t}$ and the subsequently moves to the state $x_{t+1} \sim p(\cdot|x_t,u_t)$. Motivated by the iterative updates of the $Q$-learning algorithm [2], we consider the following stochastic updates in our Algorithm 1 to learn the state–action value function in our methodology:

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

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

$$\bar\Lambda_\beta(s,a) = \sum_{s'\in\mathcal{S}} p_{ss'}^a\left(c_{ss'}^a - \frac{\gamma^2}{\beta}\log\sum_{a'}\exp\left(\frac{-\beta}{\gamma}\bar\Lambda_\beta(s',a')\right)\right) =: \left[\bar T\bar\Lambda_\beta\right](s,a). \tag{15}$$

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 $\gamma/\beta\sum_{s'}p_{ss'}^a\log p_{ss'}^a$ which makes it difficult to *learn* its fixed point $\Lambda_\beta^*$ in the absence of the state transition probability $p_{ss'}^a$ itself. Since in this work we do not attempt to determine (or learn) either the distribution $p_{ss'}^a$ (as in [42]) 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 $\beta$ values [since $\frac{\gamma}{\beta}(\sum_{s'}p_{ss'}^a\log p_{ss'}^a)\to 0$ as $\beta\to\infty$]. The following proposition elucidates the conditions under which the updates $\Psi_t$ 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:* Refer to Appendix D. ∎

*Remark 4:* Note that the *stochasticity* of the optimal policy $\mu_\beta^*(a|s)$ (9) depends on $\gamma$ 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 $\gamma^t c(s_t,a_t,s_{t+1})$ are considerable (i.e., $\gamma^t c_{s_t s_{t+1}}^{a_t} > \epsilon\ \forall\ t\le T$), 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 the simulations, this characteristic of the policy in (9) results into even faster convergence rates in comparison to benchmark algorithms as the discount factor $\gamma$ decreases.

**Algorithm 1:** Model-Free Reinforcement Learning

**Input:** $N$, $\nu_t(\cdot,\cdot)$, $\sigma$; **Output:** $\mu^*$, $\bar\Lambda^*$
**Initialize:** $t=0$, $\Psi_0=0$, $\mu_0(a|s)=1/|\mathcal{A}|$.
**for** *episode* $=1$ *to* $N$ **do**
  $\beta=\sigma\times$ *epsiode*; 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 $x_{t+1}=\delta$; $t\leftarrow t+1$

## 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 the absence of the absorption state). To ensure the finiteness of the objective in (5), we consider the *discounted* Shannon entropy [43], [44]

$$H_d^\mu(s) = -\mathbb{E}\left[\sum_{t=0}^{\infty}\alpha^t\Big(\log\mu_{u_t|x_t}+\log p_{x_t x_{t+1}}^{u_t}\Big)\Big|x_0=s\right] \tag{16}$$

with a discount factor of $\alpha\in(0,1)$ which we chose to be independent of the discount factor $\gamma$ 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 $H_d^\mu(s)$ in (16) is given by

$$V_{\beta,I}^\mu(s) = \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t\hat c_{x_t x_{t+1}}^{u_t} + \frac{\alpha^t}{\beta}\log\mu(u_t|x_t)\Big|x_0=s\right] \tag{17}$$

where $\hat c_{x_t x_{t+1}}^{u_t} = c_{x_t x_{t+1}}^{u_t} + (\gamma^t/\beta\alpha^t)\log p_{x_t x_{t+1}}^{u_t}$, and the subscript $I$ stands for the "infinite entropy" case. Note that the free-energy functions (7) and (17) differ only with regards to the discount factor $\alpha$ and, thus, our solution methodology in this section is similar to the one in Section III-B.

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

$$V_{\beta,I}^\mu(s) = \sum_{a,s'}\mu_{a|s}p_{ss'}^a\left(\check c_{ss'}^a + \frac{\gamma}{\alpha\beta}\log\mu_{a|s} + \gamma V_{\beta,I}^\mu(s')\right) \tag{18}$$

where $\check c_{ss'}^a = c_{ss'}^a + \gamma/\alpha\beta\log p_{ss'}^a$.

*Proof:* See Appendix C. The above derivation shows and exploits the algebraic structure $\alpha\sum_{s'}p_{ss'}^a H_d^\mu(s') = \sum_{s'}p_{ss'}^a\log p_{ss'}^a + \log\alpha\mu(a|s) + \lambda_s$ (Lemma 4). ∎

The optimal policy satisfies $(\partial V^{\mu}_{\beta,I}(s))/(\partial \mu(a|s)) = 0$, which results into Gibb's distribution

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

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

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 $V^*_{\beta,I}(s)(:= V^{\mu^*_{\beta,I}}_{\beta,I}(s))$:

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

*Remark 5:* The subsequent steps to learn the optimal policy $\mu^*_{\beta,I}$ in (19) are similar to the steps demonstrated in Section III-C. We forego the similar analysis here.

*Remark 6:* When $\alpha = \gamma$ the policy $\mu^*_{\beta,I}$ in (19), state–action value function $\Phi_\beta$ in (20), and the free-energy function $V^*_{\beta,I}$ in (21) corresponds to the similar expressions that are obtained in the ER methods [14]. However, in this article, we do not require that $\alpha = \gamma$. On the other hand, we propose that $\alpha$ should take up large values. In fact, our simulations in Section VI demonstrate better convergence rates that are obtained when $\gamma < \alpha = (1-\epsilon)$ as compared to when $\gamma = \alpha$.

## V. Parameterized MDPs

### A. Problem Formulation

As stated in Section I, many application areas, such as small cell networks (Fig. 1), pose a parmaterized MDP that requires simultaneously determining the 1) optimal policy $\mu^*$ and 2) the unknown state and action parameters $\zeta = \{\zeta_s\}$ and $\eta = \{\eta_a\}$ such that the state value function

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

is minimized $\forall\ s \in \mathcal{S}$, where $x_t(\zeta)$ denotes the state $x_t \in \mathcal{S}$ with the associated parameter $\zeta_{x_t}$ and $u_t(\eta)$ denotes the action $u_t \in \mathcal{A}$ with the associated action parameter value $\eta_{u_t}$. 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 $J^{\mu}_{\zeta\eta}(s)$ and the Shannon entropy $H^{\mu}(s)$ of the MDP for all $\mu \in \Gamma$. We further assume that the state-transition probability $\{p^a_{ss'}\}$ is independent of the state and action parameters $\zeta, \eta$.

### 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 $\zeta$ and $\eta$ 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 $\zeta$ and $\eta$. To determine the optimal (local) parameters $\zeta$ and $\eta$, we set $\sum_{s'\in\mathcal{S}}([\partial V^*_{\beta,\zeta\eta}(s')]/[\partial\zeta_s]) = 0\ \forall\ s$, and $\sum_{s'\in\mathcal{S}}([\partial V^*_{\beta,\zeta\eta}(s')]/[\partial\eta_a]) = 0\ \forall\ a$, which we implement by using the gradient descent steps

$$\zeta_s^+ = \zeta_s - \eta\sum_{s'\in\mathcal{S}} G^{\beta}_{\zeta_s}(s'),\ \ \eta_a^+ = \eta_a - \bar{\eta}\sum_{s'\in\mathcal{S}} G^{\beta}_{\eta_a}(s'). \quad (23)$$

Here, $G^{\beta}_{\zeta_s}(s') := ([\partial V^*_{\beta,\zeta\eta}(s')]/[\partial\zeta_s])$ and $G^{\beta}_{\eta_a}(s') := ([\partial V^*_{\beta,\zeta\eta}(s')]/[\partial\eta_a])$. The derivatives $G^{\beta}_{\zeta_s}$ and $G^{\beta}_{\eta_a}$ are assumed to be bounded (see Proposition 2). We compute these derivatives as $G^{\beta}_{\zeta_s}(s') = \sum_{a'}\mu_{a'|s'}K^{\beta}_{\zeta_s}(s',a')$ and $G^{\beta}_{\eta_a}(s') = \sum_{a'}\mu_{a'|s'}L^{\beta}_{\eta_a}(s',a')\ \forall\ s' \in \mathcal{S}$, where $K^{\beta}_{\zeta_s}(s',a')$ and $L^{\beta}_{\eta_a}(s',a')$ are the fixed points of their corresponding Bellman equations $K^{\beta}_{\zeta_s}(s',a') = [T_1K^{\beta}_{\zeta_s}](s,a)$ and $L^{\beta}_{\eta_a}(s',a') = [T_2L^{\beta}_{\eta_a}](s',a')$ where

$$\left[T_1K^{\beta}_{\zeta_s}\right](s',a') = \sum_{s''}p^{a'}_{s's''}\left[\frac{\partial c^{a'}_{s's''}}{\partial\zeta_s} + \gamma G^{\beta}_{\zeta_s}(s'')\right]$$

$$\left[T_2L^{\beta}_{\eta_a}\right](s',a') = \sum_{s''}p^{a'}_{s's''}\left[\frac{\partial c^{a'}_{s's''}}{\partial\eta_a} + \gamma G^{\beta}_{\eta_a}(s'')\right]. \quad (24)$$

Note that in the above equations we have suppressed the dependence of the instantaneous cost function $c^{a'}_{s's''}$ on the parameters $\zeta$ and $\eta$ to avoid notational clutter.

*Theorem 4:* The operators $[T_1K^{\beta}_{\zeta_s}](s',a')$ and $[T_2L^{\beta}_{\eta_a}](s',a')$ defined in (24) are contraction maps with respect to a weighted maximum norm $\|\cdot\|_\xi$, where $\|X\|_\xi = \max_{s',a'}(X(s',a')/\xi_{s'})$ and $\xi \in \mathbb{R}^{|\mathcal{S}|}$ is a vector of positive components $\xi_s$.

*Proof:* Refer to Appendix D for details. ∎

As previously stated in Section I, the state value function $J^{\mu}_{\zeta\eta}(\cdot)$ in (22) is generally nonconvex function of the parameters $\zeta$ and $\eta$ and riddled with multiple poor local minima with the resulting optimization problem being possibly NP-hard [27]. In our algorithm for parameterized MDPs we anneal $\beta$ from $\beta_{\min}$ to $\beta_{\max}$, similar to our approach for nonparameterized MDPs in Section III-B, where the solution from the current $\beta$ iteration is used to initialize the subsequent $\beta$ iteration. However, in addition to facilitating a steady transition from an exploratory policy to an exploitative policy, annealing facilitates a gradual homotopy from the convex negative Shannon entropy function to the nonconvex state value function $J^{\mu}_{\zeta\eta}$ which prevents our algorithm from getting stuck in a poor local minimum. The underlying idea of our heuristic is to track the optimal as the initial convex function deforms to the actual nonconvex cost. Also, minimizing the Lagrangian $V^*_{\beta}(s)$ at $\beta = \beta_{\min} \approx 0$ determines the global minimum thereby making our algorithm insensitive to initialization. Algorithm 2 illustrates steps to determine policy and parameters for a parameterized MDP.

### C. Parameterized Reinforcement Learning

In many applications, formulated as parameterized MDPs, the explicit knowledge of the cost function $c^a_{ss'}$, its dependence on the parameters $\zeta$ and $\eta$, and the state-transition probabilities $\{p^a_{ss'}\}$ 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 $\zeta$, $\eta$ values which

**Algorithm 2:** Parameterized Markov Decision Process

**Input:** $\beta_{\min}$, $\beta_{\max}$, $\tau$; **Output:** $\mu^*$, $\zeta$ and $\eta$.
**Initialize:** $\beta = \beta_{\min}$, $\mu_{a|s} = \frac{1}{|\mathcal{A}|}$, and $\zeta, \eta$ to 0
**while** $\beta \leq \beta_{\max}$ **do**
  **while** *True* **do**
    **while** *until convergence* **do**
      update $\Lambda_\beta$,$\mu_\beta$,$G^\beta_{\zeta_s}$,$G^\beta_{\eta_a}$ in (10), (9) and (24)
    update $\zeta$, $\eta$ in (23) **if** $\|G_{\zeta_s}\|$, $\|G_{\eta_a}\| < \epsilon$, break
  $\beta \leftarrow \tau\beta$

can subsequently be used to simultaneously learn the policy $\mu^*_{\beta,\zeta\eta}$ and the unknown state and action parameters $\zeta$ and $\eta$ via stochastic iterative updates. At each $\beta$ iteration in our learning algorithm, we employ the stochastic iterative updates in (14) to determine the optimal policy $\mu^*_{\beta,\zeta\eta}$ for given $\zeta$, $\eta$ values and subsequently employ the stochastic iterative updates

$$K^{t+1}_{\zeta_s}(x_t,u_t) = (1-\nu_t(x_t,u_t))K^t_{\zeta_s}(x_t,u_t) + \nu_t(x_t,u_t)\left[\frac{\partial c^{u_t}_{x_t x_{t+1}}}{\partial \zeta_s} + \gamma G^t_{\zeta_s}(x_{t+1})\right] \quad (25)$$

where $G^t_{\zeta_s}(x_{t+1}) = \sum_a \mu_{a|x_{t+1}} K^t_{\zeta_s}(x_{t+1},a)$ to learn the derivative $G^{\beta*}_{\zeta_s}(\cdot)$. Similar updates are used to learn $G^{\beta*}_{\eta_a}(\cdot)$. The parameter values $\zeta$ and $\eta$ are then updated using the gradient descent step in (23). The following proposition formalizes the convergence of the updates in (25) to the fixed point $G^{\beta*}_{\zeta_s}$.

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

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

the updates in (25) converge to the unique fixed point $G^{\beta*}_{\zeta_s}(s')$ of the map $T_1 : G_{\zeta_s} \to G_{\zeta_s}$ in (24).

*Proof:* Refer to Appendix D for details. ∎

*Algorithmic Details:* Refer to 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 $\mu^*$, here the agent interacts with the environment by 1) taking an action $u_t \in \mathcal{A}$ and *also* providing 2) estimated parameter $\zeta$, $\eta$ 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 $\mu^*_\beta$ at a given value of the parameters $\zeta$ and $\eta$ using the iterations (14) and then learn the fixed points $G^{\beta*}_{\zeta_s}$, $G^{\beta*}_{\zeta_a}$ using the iterations in (25) to update the parameters $\zeta$ and $\eta$ using (23). Note that the iterations (25) 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 *$\epsilon$-distinct environments* and finite difference method. Here, the $\epsilon$-distinct environments represent the same underlying MDP but are distinct only in one of the parameter values. However, if two $\epsilon$-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.

**Algorithm 3:** Parameterized Reinforcement Learning

**Input:** $\beta_{\min}$, $\beta_{\max}$, $\tau$, $T$, $\nu_t$; **Output:** $\mu^*$, $\zeta$, $\eta$
**Initialize:** $\beta = \beta_{\min}$, $\mu_t = \frac{1}{|\mathcal{A}|}$, and $\zeta, \eta, G^\beta_\zeta, G^\beta_\eta, K^\beta_\zeta, L^\beta_\eta, \bar{\Lambda}_\beta$ to 0.
**while** $\beta \leq \beta_{\max}$ **do**
  Use Algorithm 1 to obtain $\mu^*_{\beta,\zeta\eta}$ at given $\zeta$, $\eta$, $\beta$.
  Consider *env1*($\zeta$,$\eta$), *env2*($\zeta'$, $\eta'$); set $\zeta' = \zeta$, $\eta' = \eta$
  **while** $\{\zeta_s\}$, $\{\eta_a\}$ *converge* **do**
    **for** $\forall s \in \mathcal{S}$ **do**
      **for** *episode* $= 1$ *to* $T$ **do**
        reset *env1, env2* at state $x_t$,
        **while** *True* **do**
          sample action $u_t \sim \mu^*(\cdot|x_t)$.
          *env1:* obtain $c_t$, $x_{t+1}$.
          *env2:* set $\zeta'_s = \zeta_s + \Delta\zeta_s$, get $c'_t$, $x_{t+1}$.
          find $G^{t+1}_{\zeta_s}(x_t)$ with $\frac{\partial c^{u_t}_{x_t x_{t+1}}}{\partial\zeta_s} \approx \frac{c'_t - c_t}{\Delta\zeta_s}$.
          break if $x_{t+1} = \delta$; $t \leftarrow t+1$.
    Similarly learn $G^\beta_{\eta_a}$. Update $\{\zeta_s\}$, $\{\eta_a\}$ in (23).
  $\beta \leftarrow \tau\beta$

*Remark 7:* Parameterized MDPs with infinite Shannon entropy $H^\mu$ can be analogously addressed using the above methodology.

*Remark 8:* The MDPs addressed in Sections III–V consider different variants of the discounted infinite horizon problems. MDPs in Section III address the class of sequential problems that have a nonzero probability of reaching a cost-free termination state (i.e., a finite Shannon entropy value). MDPs considered in Section IV need not reach a termination state (possibly infinite value of Shannon entropy), and the underlying sequential decision problem continues for the length of horizon determined by the discounting factor $\gamma$. Parameterized MDPs in Section V can have finite or infinite Shannon entropy, but they comprise of states and actions that have an unknown parameter associated to them.

## VI. Simulations

We broadly classify our simulations into two categories. First, in the model-free RL setting, we demonstrate our Algorithm 1 to determine the control policy $\mu^*$ for the finite and infinite Shannon entropy variants of the Gridworld environment in Fig. 2. Each cell in the Gridworld denotes a state. The cells colored black are invalid states. An agent can choose to move vertically, horizontaly, diagonally, or stay at the current cell. Each action is followed by a probability to slip in the neighboring states [probability of 0.05 to slip in each of the vertical and horizontal directions, and probability of 0.025 to slip in each of the diagonal directions—cumulative $p(slip) \approx 0.3$]. For the finite entropy case, each step incurs a unit cost. The process terminates when the agent reaches the terminal state **T**. For the infinite entropy case, each step incurs

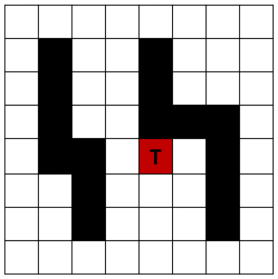


Fig. 2. Gridworld environment.

a unit reward. Second, in the parameterized MDPs and RL setting, we demonstrate our Algorithms 2 and 3 in designing a 5G small cell network. This involves simultaneously determining the locations of the small cells in the network as well as the optimal routing path of the communication packets from the base station to the users.

We compare our MEP-based Algorithm 1 with the benchmark algorithms ER G-learning (also referred to as Soft $Q$-learning) [14], $Q$-learning [5], and Double $Q$-learning [13]. Note that our choice of the benchmark algorithm G-learning (or, ER Soft Q presented in [14]) is based on its commonality to our framework as discussed in 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 [14] already establishes the efficacy of the G-learning algorithm over the following algorithms in literature $Q$-learning, Double $Q$-learning, $\Psi$-learning [45], Speedy $Q$-learning [46], and the consistent Bellman Operator $\mathcal{T}_C$ of [47]. Below, we highlight features and advantages of our MEP-based Algorithm 1.

*Faster Convergence to Optimal $J^*$:* Fig. 3(a1)–(a3) (finite entropy variant of Gridworld) and Fig. 3(b1)–(b3) (infinite entropy variant of Gridworld) illustrate the faster convergence of our MEP-based Algorithm 1 for different discount factor $\gamma$ values. Here, at each episode, the percentage error $\Delta V/J^*$ between the value function $V_\beta^\mu$ corresponding to the *learned* policy $\mu = \mu(ep)$ in the episode $ep$, and the optimal value function $J^*$ is given by

$$\frac{\Delta V(ep)}{J^*} = \frac{1}{N}\sum_{i=1}^{N}\sum_{s\in\mathcal{S}}\frac{\left|V_{\beta,i}^{\mu(ep)}(s) - J^*(s)\right|}{J^*(s)} \tag{26}$$

where $N$ denotes the total experimental runs and $i$ indexes the value function $V_{\beta,i}^\mu$ for each run. As observed in Fig. 3(a1)–(a3), and (b1)–(b3), our Algorithm 1 converges even faster as the discount factor $\gamma$ decreases. We characterize the faster convergence rates also in terms of the convergence time—more precisely the *percentage* $\bar{E}_{pr}$ of total episodes taken for the learning error $\Delta V/J^*$ to reach within 5% of the best [see Fig. 3(a4) and (b4)]. As is observed in the figures, the performance of our (MEP-based) algorithm in comparison to ER G learning is better across all values (0.65–0.95) of discount factor $\gamma$. Note that the performance of Algorithm 1 obtains even better with decreasing $\gamma$ values where the smaller discount factor values occur in instances such as the context of recommendation systems [48], and teaching RL-agents using human-generated rewards [49].

*Robustness to Noise in Data:* Fig. 3(c1)–(c4) demonstrate robustness to noisy environments; here, the instantaneous cost $c(s, a, s')$ in the finite horizon variant of Gridworld is noisy. For the purpose of simulations, we add the Gaussian noise $\mathcal{N}(0, \sigma^2)$ with $\sigma = 1$ for vertical and horizontal actions, and $\sigma = 0.5$ for diagonal movements. Here, at each episode, we compare the percentage error $\Delta V/J^*$ in the *learned* value functions $V_\beta$ [corresponding to the state–action value estimate in (14)] of the respective algorithms. Similar to our observations and conclusions in Fig. 3(a1)–(a3) and (b1)–(b3), we see faster convergence of our MEP-based algorithm over the benchmark algorithms in Fig. 3(c1)–(c3) in the case of the noisy environment. Also, Fig. 3(c4) demonstrates that across all discount factor values (0.65–0.95), Algorithm 1 converges faster than the ER Soft Q learning.

*Simultaneously Determining the Unknown Parameters and Policy in Parameterized MDPs:* We design the 5G small cell network (see Fig. 1) both when the underlying model ($c_{ss'}^a$ and $p_{ss'}^a$) 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 $\delta$ at $z$ in the domain $\Omega \subset \mathbb{R}^2$ as shown in Fig. 4(a). The objective is to determine the locations $\{y_j\}_{j=1}^5$ (parameters) of the small cells $\{f_j\}_{j=1}^5$ and determine the corresponding communication routes (policy). Here, the state space of the underlying MDP is $\mathcal{S} = \{n_1, \ldots, n_{46}, f_1, \ldots, f_5\}$ where the locations $y_1, \ldots, y_5$ of the small cells are the unknown parameters $\{\zeta_s\}$ of the MDP, the action space is $\mathcal{A} = \{f_1, \ldots, f_5\}$, and the cost function $c(s, a, s') = \|\rho(s) - \rho(s')\|_2^2$ where $\rho(\cdot)$ denotes the spatial location of the respective states. The objective is to simultaneously determine the parameters (unknown small cell locations) and the control policy (communication routes in the 5G network). We consider two cases where 1) $p_{ss'}^a$ is deterministic, that is, an action $a$ at the state $s$ results into $s' = a$ with probability 1 and 2) $p_{ss'}^a$ 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. In addition, due to the absence of prior work in the literature on network design problems modeled as parameterized MDPs, we compare our results only with the solution resulting from a straightforward sequential methodology [as shown in Fig. 4(a)] where we first partition the user nodes into five distinct clusters to allocate a small cell in each cluster, and then determine optimal routes in the network.

*Deterministic $p(s, a, s')$:* Fig. 4(b) illustrates the allocation of small cells and the corresponding communication routes (resulting from optimal policy $\mu^*$) as determined by Algorithm 2. Here, the network is designed to minimize the cumulative cost of communication from each user node and small cell. As denoted in the figure, the route $\delta \to y_3 \to y_4 \to y_1 \to n_i$ carries the communication packet from the base station $\delta$ to the respective user nodes $n_i$ as indicated by the gray arrow from $y_1$. The cost incurred here is approximately 180% lesser than that in Fig. 4(a) clearly indicating the advantage obtained from simultaneously determining the parameters and policy over a sequential methodology. In the

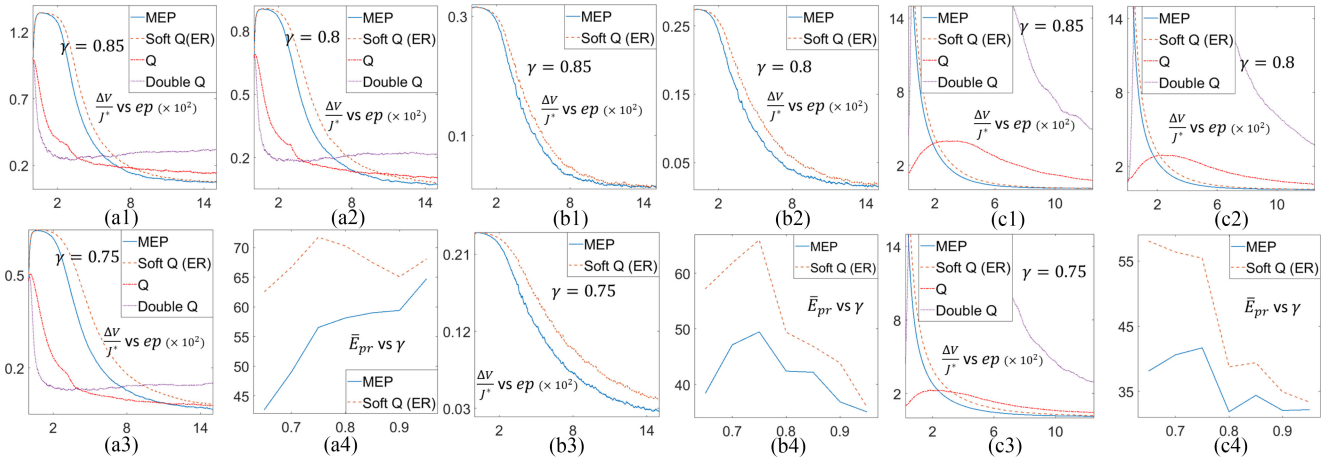


Fig. 3. Performance of the MEP-based algorithm. Illustrations on the Gridworld environment in Fig. 2. (a1)–(a3) *Finite Entropy Variant:* Illustrates faster convergence of Algorithm 1 (MEP) at different $\gamma$ values. (a4) Demonstrates faster rates of convergence of Algorithm 1 (MEP) for $\gamma$ values ranging from 0.65 to 0.95. (b1)–(b3) *Infinite Entropy Variant*: Demonstrates faster convergence of Algorithm 1 (MEP) to $J^*$. (b4) Illustrates the consistent faster convergence rates of MEP with $\gamma$ ranging from 0.65 to 0.95. (c1)–(c4) *Finite Entropy Version (with added Gaussian noise)*: Similar observations as in (a1)–(a4) with significantly higher instability in learning with the Double $Q$ algorithm.

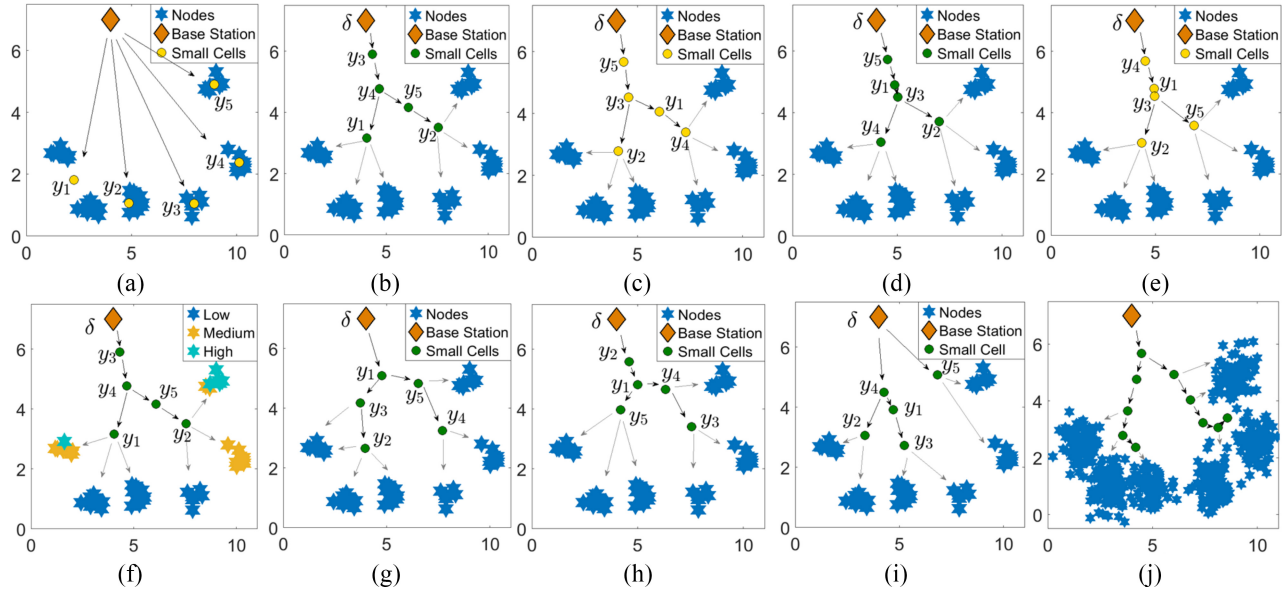


Fig. 4. Parameterized MDPs and RL—design of 5G small cell network. State space $\mathcal{S} = \{\{n_i\}, \{f_j\}, \delta\}$ comprises of the user nodes $\{n_i\}$, small cells $\{f_j\}$, and base station $\delta$. The unknown parameters $\zeta_s$ denote the locations $\{y_j\}$ of the small cells. Action space comprises of the small cells $\mathcal{A} = \{f_j\}$. Based on our modeling of the network, there are no unknown action parameters $\{\eta_a\}$. (a) Small cell locations $\{y_j\}$ and communication routes determined using a straightforward sequential methodology. (b) and (c) 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) and (e) Solutions obtained using Algorithms 2 and 3, respectively, when $p^a_{ss'}$ is probabilistic. (f) Sensitivity analysis of the solutions with respect to user node locations $\{x_i\}$. (g) and (h) Network design obtained when considering the entropy of the distribution over the control actions and paths of the MDP, respectively. (i) Network design obtained without annealing in Algorithm 2. (j) Simulation on a larger dataset (user base increased by more than ten times).

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 $\{y_j\}_{j=1}^5$ as well as the optimal policy $\{\mu^*(a|s)\}$ as demonstrated in Fig. 4(c). It is evident from Fig. 4(b) and (c) that the solutions obtained when the model is completely known and unknown are approximately the 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 $p(s, a, s')$:* Fig. 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. As before, here the network is designed to minimize the cumulative cost of communication from each user node and small cell. The cost associated to the network design is approximately 127% lesser than in Fig. 4(a). Fig. 4(e) illustrates the solution as obtained by 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 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 Fig. 4(b) into the categories of 1) low; 2) medium; and 3) high sensitiveness such that the final solution is least susceptible to the user nodes in 1) and most susceptible to the nodes in 3). 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 (24). 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 $V^{\mu}_{\beta}$ 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 Fig. 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.

*Entropy Over Paths Versus Entropy of the Policy:* We demonstrate the benefit of maximizing the entropy of the distribution $\{p_{\mu}(\omega|s)\}$ over the paths of an MDP as compared to the distribution $\{\mu(a|s)\}$ over the control actions. Fig. 4(g) demonstrates the 5G network obtained by considering the distribution over the control policy, and Fig. 4(h) illustrates the network obtained by considering the distribution over the entire paths. The network cost incurred in Fig. 4(h) is 5% less than the cost incurred in Fig. 4(g). Here, we have considered the above demonstrated probabilistic $p^{a}_{ss'}$ scenario and minimized the cumulative communication cost incurred only from the user nodes.

*Avoiding Poor Local Minima and Large-Scale Setups:* As noted in Section V-B, annealing $\beta$ from a small value $\beta_{\min}(\approx 0)$ to a large value $\beta_{\max}$ prevents the algorithm from getting stuck at a poor local minima. Fig. 4(i) demonstrates the network design obtained where Algorithm 2 does not anneal $\beta$, and iteratively solves the optimization problem at $\beta = \beta_{\max}$. The resulting network incurs a 11% higher cost in comparison to the network obtained in 4(h) where Algorithm 2 anneals $\beta$ from a small to a large value. Fig. 4(j) demonstrates the 5G network design obtained using Algorithm 2 when the user nodes are increased by around 12 times (610), and the allocated small cells are doubled to 10.

## 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 $J^{\mu}$. 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 $\mu^*$) over the paths of the MDP that is close to some given prior distribution $q(\mathcal{P}|s)$ [15], [16]. Here, the objective is to minimize the KL-divergence $D_{KL}(p_{\mu}\|q)$) under the constraint $J = J_0$ [as in (5)].

*2) Nondependence on Choice of $J_0$ in (5):* In our framework, we do not explicitly determine and work with the value of $J_0$. Instead we work with the Lagrange parameter $\beta$ in the Lagrangian $V^{\mu}_{\beta}(s)$ in (7) corresponding to the optimization problem (5). It is known from the sensitivity analysis [6] that the small values of $\beta$ correspond to the large values of $J_0$, and the large values of $\beta$ correspond to the small values of $J_0$. Thus, in our algorithms, we solve the optimization problem (5) beginning at small values of $\beta = \beta_{\min} \approx 0$ (that corresponds to some feasible large $J_0$), and anneal it to a large value $\beta_{\max}$ (that corresponds to a small $J_0$ value) at which the stochastic policy $\mu$ in (9) converges to either 0 or 1. Also at $\beta \approx 0$, the stochastic policy $\mu^*_{\beta}$ in (9) follows a uniform distribution, which implicitly fixes the value of $J_0$. Therefore, the initial value of $J_0$ in the proposed algorithms is fixed and is not required to be prespecified.

*3) Computational Complexity:* Our MEP-based Algorithm 1 performs exactly the same number of computations as the Soft $Q$-learning algorithm [14] 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 $\mu^*$ in (9), exhibits a similar number of computational steps.

*4) Scheduling $\beta$ and Phase Transition:* In our Algorithm 1, we follow a linear schedule $\beta_k = \sigma k$ ($\sigma > 0$) as suggested in the benchmark algorithm [14] to anneal the parameter $\beta$. In the case of parameterized MDPs (Algorithms 2 and 3), we geometrically anneal $\beta$ (i.e., $\beta_{k+1} = \tau\beta_k$, $\tau > 1$) from a small value $\beta_{\min}$ to a large value $\beta_{\max}$ at which the control policy $\mu^*_{\beta}$ converges to either 0 or 1. Several other MEP-based algorithms (that address problems akin to parameterized MDPs) such as deterministic annealing [7], incorporate geometric annealing of $\beta$. The underlying idea in [7] is that the solution undergoes significant changes only at certain *critical* $\beta_{cr}$ (*phase transition*) and shows insignificant changes between two consecutive critical $\beta_{cr}$s. Thus, for all practical purposes, geometric annealing of $\beta$ works well. Similar to [7], our Algorithms 2 and 3 also undergo the phase transition and we are working on its analytical expression.

*5) 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 Fig. 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 $q_{\mu}(f_j) \coloneqq \sum_{a,n_i} \mu(a|n_i)p(f_j|a,n_i) \leq c_j$ where $q_{\mu}(f_j)$ measures the 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 $\dot{x}_i = f(x,t)$. The resulting policy $\mu^*_{\beta}$ and small cells $\{y_j\}$ will also be time varying. We treat the free-energy function $V^{\mu}_{\beta}$ in (11) as a *control-Lyapunov* function and determine time varying $\mu^*_{\beta}$ and $\{y_j\}$ such that $\dot{V}^{\mu}_{\beta} \leq 0$.

*6) 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 Fig. 4 may have an associated uncertainty in its location $x_i$ due 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, the probability $p_\mu(\omega|x_0 = s)$ of taking path $\omega$ 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

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

On the right-hand 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 $\lambda_s$ are some constants. Taking the derivative of the resulting expression, we obtain

$$\frac{\partial H^\mu(s')}{\partial \mu_{a|s}} = \rho(s,a)\delta_{ss'} + \sum_{a',s''} \mu_{a'|s'} p^{a'}_{s's''} \frac{\partial H^\mu(s'')}{\partial \mu_{a|s}} - \lambda_s \quad (27)$$

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 nonzero [i.e., $p^a_{ss'} > 0$] we have that $p_\mu(s') > 0$. Now, we multiply (27) 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

$$V^\mu_\beta(s) = \sum_{a,s'} \mu_{a|s} p^a_{ss'} \left[ \hat{c}^a_{ss'} + \frac{1}{\beta} \log \mu_{a|s} + \gamma V^\mu_{\gamma\beta}(s') \right] \quad (28)$$

where $\hat{c}^a_{ss'} = c^a_{ss'} + 1/\beta \log p^a_{ss'}$ and $V^\mu_{\gamma\beta}(s') = V^\mu_\beta(s') - 1 - \gamma/\gamma\beta H(s')$. Substituting $V_{\gamma\beta}(s')$ and the algebraic expression obtained in Lemma 2 in (28), 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 is 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 $c^a_{ss'} = -1 - 1/\beta \log(|\mathcal{A}||\mathcal{S}|)$ 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 [50] 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, $-\xi_s \le \sum_{s'} p^a_{ss'} \big( c^a_{ss'} + \gamma/\beta \log p^a_{ss'} - \gamma \xi_{s'} \big)$ 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 $[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)$

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

$$\ge \gamma \sum_{s',a'} p^a_{ss'} \hat{\mu}_{a'|s'} \Big( \hat{\Lambda}_\beta(s',a') - \check{\Lambda}_\beta(s',a') \Big) =: \gamma \Delta_{\hat{\mu}} \quad (29)$$

where we use the Log sum inequality to obtain (29), 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 $|[T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta](s,a)|$

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

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

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

$$\left| \left[ T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta \right](s,a) \right| \le \gamma\lambda\xi_s \sum_{a' \in \mathcal{A}} \bar{\mu}_{a'|s'} \left\| \hat{\Lambda}_\beta - \check{\Lambda}_\beta \right\|_\xi \quad (32)$$

$$\Rightarrow \left\| T\hat{\Lambda}_\beta - T\check{\Lambda}_\beta \right\|_\xi \le \gamma\lambda \left\| \hat{\Lambda}_\beta - \check{\Lambda}_\beta \right\|_\xi \text{ with } \gamma\lambda < 1. \quad (33)$$

## 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 except the following lemma that illustrates the algebraic

structure of the discounted Shannon entropy $H_d^\mu(\cdot)$ 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 $H_d^\mu(\cdot)$ corresponding to the MDP in Section IV satisfies the algebraic term $\alpha \sum_{s'} p_{ss'}^a H_d^\mu(s') = \sum_{s'} p_{ss'}^a \log p_{ss'}^a + \log \alpha\mu(a|s) + \lambda_s$.

*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 $\{q_{ss'}^a\}$ and policy $\{\zeta_{a|s}\}$ from original MDP as

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

Next, we define $T^\mu := \alpha H_d^\mu$ that satisfies $T^\mu(s') = \sum_{a's''} \eta_{a'|s'} p_{s's''}^{a'} [-\log p_{s's''}^a - \log \eta_{a'|s'} + T^\mu(s'')]$ 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 [2, Proposition 5.5]. Let $\bar{T}$ be the map in (15). The stochastic iterative updates in (14) can be rewritten as $\bar{\Psi}_{t+1}(x_t, u_t) = (1 - \nu_t(x_t, u_t))\bar{\Psi}_t(x_t, u_t) + \nu_t(x_t, u_t)\big([\bar{T}\bar{\Psi}_t](x_t, u_t) + w_t(x_t, u_t)\big)$ where $w_t(x_t, u_t) = c_{x_t x_{t+1}}^{u_t} - \gamma^2/\beta \log \sum_a \exp(-\beta/\gamma \bar{\Psi}_t(s_{t+1}, a)) - \bar{T}\bar{\Psi}_t(x_t, u_t)$. Let $\mathcal{F}_t$ represent the history of the stochastic updates, that is, $\mathcal{F}_t = \{\bar{\Psi}_0, \ldots, \bar{\Psi}_t, w_0, \ldots, w_{t-1}, \nu_0, \ldots, \nu_t\}$, 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 [2, Proposition 4.4].

*Proof of Theorem 4:* We show that the map $T_1$ in (24) is a contraction map. For any $K_{\zeta_s}^\beta$ and $\bar{K}_{\zeta_s}^\beta$, we obtain that $|[T_1 K_{\zeta_s}^\beta - T_1 \bar{K}_{\zeta_s}^\beta](s')| \leq \gamma \sum_{a,s''} p_{s's''}^a \mu_{a|s'} |K_{\zeta_s}^\beta(s'', a) - \bar{K}_{\zeta_s}^\beta(s'', a)|$. Note that this inequality is similar to the one in (30); thus, we follow the exact same steps from (30)–(33) to show that $\|T_1 K_{\zeta_s}^\beta - T_1 \bar{K}_{\zeta_s}^\beta\|_\xi \leq \gamma\lambda \|K_{\zeta_s}^\beta - \bar{K}_{\zeta_s}^\beta\|_\xi$ 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 $\left|\partial c_{ss'}^a / \partial \zeta_l\right|$ and $\left|\partial c_{ss'}^a / \partial \eta_k\right|$ 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*, vol. 40. New York, NY, USA: Springer, 2012,

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, vol. 5. Belmont, MA, USA: Athena Sci., 1996,

[3] A. Hordijk and L. C. M. Kallenberg, "Linear programming and Markov decision chains," *Manag. Sci.*, 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 *Proc. JMLR Workshop Conf.*, 2014, pp. 496–504.

[5] C. J. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.

[6] E. T. Jaynes, "Information theory and statistical mechanics," *Phys. Rev.*, vol. 106, no. 4, p. 620, 1957.

[7] K. Rose, "Deterministic annealing, clustering, and optimization," Ph.D. dissertation, Dept. Comput. Sci., California Inst. Technol., Pasadena, CA, USA, 1991.

[8] P. Sharma, S. Salapaka, and C. Beck, "A scalable approach to combinatorial library design for drug discovery," *J. Chem. Inf. Model.*, vol. 48, no. 1, pp. 27–41, 2008.

[9] J.-G. Yu, J. Zhao, J. Tian, and Y. Tan, "Maximal entropy random walk for region-based visual saliency," *IEEE Trans. Cybern.*, vol. 44, no. 9, pp. 1661–1672, Sep. 2014.

[10] Y. Xu, S. M. Salapaka, and C. L. Beck, "Aggregation of graph models and Markov chains by deterministic annealing," *IEEE Trans. Autom. Control*, vol. 59, no. 10, pp. 2807–2812, Oct. 2014.

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

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

[13] H. V. Hasselt, "Double Q-learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 2613–2621.

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

[15] J. Grau-Moya, F. Leibfried, and P. Vrancx, "Soft Q-learning with mutual-information regularization," in *Proc. ICLR*, 2019, pp. 1–19.

[16] J. Peters, K. Mulling, and Y. Altun, "Relative entropy policy search," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, pp. 1607–1612.

[17] G. Neu, A. Jonsson, and V. Gómez, "A unified view of entropy-regularized Markov decision processes," 2017. [Online]. Available: arXiv:1705.07798.

[18] K. Asadi and M. L. Littman, "An alternative softmax operator for reinforcement learning," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 243–252.

[19] O. Nachum, M. Norouzi, K. Xu, and D. Schuurmans, "Bridging the gap between value and policy based reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 2775–2785.

[20] B. Dai *et al.* "SBEED: Convergent reinforcement learning with nonlinear function approximation," 2017. [Online]. Available: arXiv:1712.10285.

[21] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1889–1897.

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

[23] A. Aguilar-Garcia *et al.*, "Location-aware self-organizing methods in femtocell networks," *Comput. Netw.*, vol. 93, pp. 125–140, Dec. 2015.

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

[25] 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 Trans. Cybern.*, vol. 46, no. 11, pp. 2643–2655, Nov. 2016.

[26] A. Srivastava and S. M. Salapaka, "Simultaneous facility location and path optimization in static and dynamic networks," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 4, pp. 1700–1711, Dec. 2020.

[27] M. Mahajan, P. Nimbhorkar, and K. Varadarajan, "The planar k-means problem is NP-hard," in *Proc. Int. Workshop Algorithms Comput.*, 2009, pp. 274–285.

[28] A. A. Abin, "Querying beneficial constraints before clustering using facility location analysis," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 312–323, Jan. 2018.

[29] D. Huang, C.-D. Wang, and J.-H. Lai, "Locally weighted ensemble clustering," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1460–1473, May 2018.

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

[31] G. Xiang and J. Su, "Task-oriented deep reinforcement learning for robotic skill acquisition and control," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 1056–1069, Feb. 2021.

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

[33] M. Hausknecht and P. Stone, "Deep reinforcement learning in parameterized action space," 2015. [Online]. Available: arXiv:1511.04143.

[34] W. Masson, P. Ranchod, and G. Konidaris, "Reinforcement learning with parameterized actions," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1934–1940.
[35] E. Wei, D. Wicke, and S. Luke, "Hierarchical approaches for reinforcement learning in parameterized action space," in *Proc. AAAI Spring Symp. Series*, 2018, pp. 1–7.
[36] J. Xiong *et al.*, "Parametrized deep Q-networks learning: Reinforcement learning with discrete-continuous hybrid action space," 2018. [Online]. Available: arXiv:1810.06394.
[37] V. Narayanan and S. Jagannathan, "Event-triggered distributed control of nonlinear interconnected systems using online reinforcement learning with exploration," *IEEE Trans. Cybern.*, vol. 48, no. 9, pp. 2510–2519, Sep. 2018.
[38] E. Çilden and F. Polat, "Toward generalization of automated temporal abstraction to partially observable reinforcement learning," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1414–1425, Aug. 2015.
[39] E. T. Jaynes, *Probability Theory: The Logic of Science*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[40] F. Biondi, A. Legay, B. F. Nielsen, and A. Wkasowski, "Maximizing entropy over Markov processes," *J. Logical Algebraic Methods Program.*, vol. 83, nos. 5–6, pp. 384–399, 2014.
[41] Y. Savas, M. Ornik, M. Cubuktepe, and U. Topcu, "Entropy maximization for constrained Markov decision processes," in *Proc. 56th Annu. Allerton Conf. Commun. Control Comput. (Allerton)*, 2018, pp. 911–918.
[42] S. Ross, J. Pineau, B. Chaib-draa, and P. Kreitmann, "A Bayesian approach for learning and planning in partially observable Markov decision processes," *J. Mach. Learn. Res.*, vol. 12, no. 48, pp. 1729–1770, 2011.
[43] L. P. Hansen, T. J. Sargent, G. Turmuhambetova, and N. Williams, "Robust control and model misspecification," *J. Econ. Theory*, vol. 128, no. 1, pp. 45–90, 2006.
[44] Z. Zhou, M. Bloem, and N. Bambos, "Infinite time horizon maximum causal entropy inverse reinforcement learning," *IEEE Trans. Autom. Control*, vol. 63, no. 9, pp. 2787–2802, Sep. 2018.
[45] K. Rawlik, M. Toussaint, and S. Vijayakumar, "Approximate inference and stochastic optimal control," 2010. [Online]. Available: arXiv:1009.3958.
[46] M. Ghavamzadeh, H. J. Kappen, M. G. Azar, and R. Munos, "Speedy Q-learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2411–2419.
[47] M. G. Bellemare, G. Ostrovski, A. Guez, P. S. Thomas, and R. Munos, "Increasing the action gap: New operators for reinforcement learning," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 1476–1483.
[48] G. Zheng *et al.*, "DRN: A deep reinforcement learning framework for news recommendation," in *Proc. World Wide Web Conf. (WWW)*, 2018, pp. 167–176.
[49] W. B. Knox and P. Stone, "Reinforcement learning from human reward: Discounting in episodic tasks," in *Proc. IEEE RO-MAN 21st IEEE Int. Symp. Robot Human Interact. Commun.*, 2012, pp. 878–885.
[50] K. Kobayashi and T. S. Han, *Mathematics of Information and Coding*, vol. 203. Boston, MA, USA: Amer. Math. Soc., 2007,
[51] P. Sharma, S. M. Salapaka, and C. L. Beck, "Entropy-based framework for dynamic coverage and clustering problems," *IEEE Trans. Autom. Control*, vol. 57, no. 1, pp. 135–150, Jan. 2012.

**Amber Srivastava** received the B.Tech. degree in mechanical engineering from the Indian Institute of Technology Kanpur, Kanpur, India, in 2014, and the master's degree in mathematics from the University of Illinois at Urbana–Champaign, Urbana, IL, USA, in 2020, where he is currently pursuing the Ph.D. degree with the Mechanical Science and Engineering Department.

He was an Assistant Manager with FMCG, Mumbai, India, from 2014 to 2015. His areas of interest are optimization, learning, and controls.

**Srinivasa M. Salapaka** (Senior Member, IEEE) received the B.Tech. degree from the Indian Institute of Technology Madras, Chennai, India, in 1995, and the M.S. and Ph.D. degrees in mechanical engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 1997 and 2002, respectively.

From 2002 to 2004, he was a Postdoctoral Associate with the Massachusetts Institute of Technology, Cambridge, MA, USA. Since 2004, he has been a Faculty Member with the Mechanical Science and Engineering Department, University of Illinois at Urbana–Champaign, Urbana, IL, USA. His areas of current research are controls for nanotechnology, combinatorial optimization, and power electronics.