
Actor-Critic based Improper Reinforcement Learning

Mohammadi Zaki¹ Avinash Mohan² Aditya Gopalan¹ Shie Mannor³

Abstract

We consider an improper reinforcement learning setting where a learner is given M base controllers for an unknown Markov decision process, and wishes to combine them optimally to produce a potentially new controller that can outperform each of the base ones. This can be useful in tuning across controllers, learnt possibly in mismatched or simulated environments, to obtain a good controller for a given target environment with relatively few trials. Towards this, we propose two algorithms: (1) a Policy Gradient-based approach; and (2) an algorithm that can switch between a simple Actor-Critic (AC) based scheme and a Natural Actor-Critic (NAC) scheme depending on the available information. Both algorithms operate over a class of improper mixtures of the given controllers. For the first case, we derive convergence rate guarantees assuming access to a gradient oracle. For the AC-based approach we provide convergence rate guarantees to a stationary point in the basic AC case and to a global optimum in the NAC case. Numerical results on (i) the standard control theoretic benchmark of stabilizing a cartpole; and (ii) a constrained queueing task show that our improper policy optimization algorithm can stabilize the system even when the base policies at its disposal are unstable.

1. Introduction

A natural approach to design effective controllers for large, complex systems is to first approximate the system using a tried-and-true Markov decision process (MDP) model, such as the Linear Quadratic Regulator (LQR) (Dean et al., 2017) or tabular MDPs (Auer et al., 2009), and then compute (near-) optimal policies for the assumed model. Though this

¹Department of ECE, IISc, Bangalore, India ²Boston University, Massachusetts, USA ³Faculty of Electrical Engineering, Technion, Haifa, Israel and NVIDIA Research, Israel.. Correspondence to: Mohammadi Zaki <mohammadi@iisc.ac.in>.

yields favorable results in principle, it is quite possible that errors in describing or understanding the system – leading to misspecified models – may lead to ‘overfitting’, resulting in subpar controllers in practice. Moreover, in many cases, the stability of the designed controller may be crucial and more desirable than optimizing a fine-grained cost function. From the controller design standpoint, it is often easier, cheaper and more interpretable to specify or hard-code control policies based on domain-specific principles, e.g., anti-lock braking system (ABS) controllers (Radac & Precup, 2018). For these reasons, we investigate in this paper a promising, *general-purpose* reinforcement learning (RL) approach towards designing controllers¹ given pre-designed ensembles of *basic* or *atomic* controllers, which (a) allows for flexibly combining the given controllers to obtain richer policies than the atomic policies, and, at the same time, (b) can preserve the basic structure of the given class of controllers and confer a high degree of interpretability on the resulting hybrid policy.

Overview of the approach. We consider a situation where we are given ‘black-box access’ to M controllers (maps from state to action distributions) $\{K_1, \dots, K_M\}$ for an unknown MDP. By this we mean that we can choose to invoke any of the given controllers at any point during the operation of the system. With the understanding that the given family of controllers is ‘reasonable,’ we frame the problem of learning the best combination of the controllers by trial and error. We first set up an *improper policy class* of all randomized mixtures of the M given controllers – each such mixture is parameterized by a probability distribution over the M base controllers. Applying an improper policy in this class amounts to selecting independently at each time a base controller according to this distribution and implementing the recommended action as a function of the present state of the system. The learner’s goal is to find the best performing mixture policy by iteratively testing from the pool of given controllers and observing the resulting state-action-reward trajectory.

Note that the underlying parameterization in our setting is over a set of given *controllers* which could be potentially abstract and defined for complex MDPs with continuous

¹We use the terms ‘policy’ and ‘controller’ interchangeably in this article.

state/action spaces, instead of the (standard) policy gradient (PG) view where the parameterization directly defines the policy in terms of the state-action map. Our problem, therefore, hews more closely to a *meta RL* framework, in that we operate over a set of controllers that have themselves been designed using some optimization framework to which we are agnostic. This has the advantage of conferring a great deal of generality, since the class of controllers can now be chosen to promote any desirable secondary characteristic such as interpretability, ease of implementation or cost effectiveness.

It is also worth noting that our approach is different from treating each of the base controllers as an ‘expert’ and applying standard mixture-of-experts algorithms, e.g., Hedge or Exponentiated Gradient (Littlestone & Warmuth, 1994; Auer et al., 1995; Kocák et al., 2014; Neu, 2015). Whereas the latter approach is tailored to converge to the best single controller (under the usual gradient approximation framework) and hence qualifies as a ‘proper’ learning algorithm, the former optimization problem is in the improper class of mixture policies which not only contains each atomic controller but also allows for a true mixture (i.e., one which puts positive probability on at least two elements) of many atomic controllers to achieve optimality; we exhibit concrete examples where this is indeed possible.

Our Contributions. We make the following contributions in this context:

- We develop a gradient-based RL algorithm to iteratively tune a softmax parameterization of an improper (mixture) policy defined over the base controllers (Algorithm 1). While this algorithm, *Softmax Policy Gradient* (or Softmax PG), relies on the availability of value function gradients, we later propose a modification that we call *GradEst* (see Alg. 6 in appendix) to Softmax PG to rectify this. GradEst uses a combination of rollouts and Simultaneously Perturbed Stochastic Approximation (SPSA) (Borkar, 2008) to estimate the value gradient at the current mixture distribution.
- We show a convergence rate of $\mathcal{O}(1/t)$ to the optimal value function for finite state-action MDPs. To do this, we employ a novel Non-uniform Łojasiewicz-type inequality (Łojasiewicz, 1963), that lower bounds the 2-norm of the value gradient in terms of the suboptimality of the current mixture policy’s value. Essentially, this helps establish that when the gradient of the value function hits zero, the value function is itself close to the optimum.
- Policy-gradient methods are well-known to suffer from high variance (Peters & Schaal, 2008; Bhatnagar et al., 2009). To circumvent this issue, we develop an algorithm that can switch between a simple Actor-Critic (AC) based scheme and a Natural Actor-Critic (NAC) scheme depending on the available information. The algorithm, ‘ACIL’ (Sec. 5), executes on a single sample path, without requiring any

forced resets, as is common in many RL algorithms. We provide convergence rate guarantees to a stationary point in the basic AC case and to a global optimum in the NAC case, under some additional (but standard) assumptions (of uniform ergodicity). The total complexity of AC is measured to attain an $(\varepsilon + \text{Critic_error})$ -accurate stationary point. The total complexity of NAC is measured to attain an $(\varepsilon + \text{Critic_error} + \text{Actor_error})$ -accurate stationary point. We use linear function approximation to approximate the value function and our convergence analysis show exactly how this approximation affects the final complexity bound.

- We corroborate our theory using extensive simulation studies. For the PG based method we use *GradEst* in two different settings (a) the well-known *CartPole* system and (b) a scheduling task in a constrained queueing system. We discuss both these settings in detail in Sec. 2, where we also demonstrate the power of our improper learning approach in finding control policies with provably good performance. In our experiments (see Sec. 6), we eschew access to exact value gradients and instead rely on a combination of roll outs and SPSA to *estimate* them. For the actor-critic based learner, we demonstrate simulations on various queueing theoretic simulations using the natural-actor-critic based *ACIL*. All the results show that our proposed algorithms quickly converge to the correct mixture of available atomic controllers.

Related Work (brief). We provide a quick survey of relevant literature. A detailed survey is deferred to the appendix.

Policy gradient. The basic policy gradient method has become a cornerstone of modern RL and given birth to an entire class of highly efficient policy search techniques such as CPI (Kakade & Langford, 2002), TRPO (Schulman et al., 2015), PPO (Schulman et al., 2017), and MADDPG (Lowe et al., 2020). A growing body of recent work shows promising results about convergence rates for PG algorithms over finite state-action MDPs (Agarwal et al., 2020a; Shani et al., 2020; Bhandari & Russo, 2019; Mei et al., 2020), where the parameterization is over the entire space of state-action pairs, i.e., $\mathbb{R}^{S \times A}$. These advances, however, are partially offset by negative results such as those in Li et al. (2021), which show that the convergence time is $\Omega\left(|\mathcal{S}|^{2^{1/(1-\gamma)}}\right)$, where \mathcal{S} is the state space of the MDP and γ the discount factor, even with exact gradient knowledge.

Improper learning. The above works concern *proper* learning, where the policy search space is usually taken to be the set of all deterministic policies for an MDP. *Improper* learning, on the other hand, has been studied in statistical learning theory for the IID setting (Daniely et al., 2014; 2013). In this *representation independent* learning framework, the learning algorithm is not restricted to output a hypothesis from a given set of hypotheses.

Boosting. Agarwal et al. (2020b) attempts to frame and solve policy optimization over an improper class by boosting

a given class of controllers. This work, however, is situated in the context of non-stochastic control and assumes perfect knowledge of (i) the memory-boundedness of the MDP, and (ii) the state noise vector in every round, which amounts to essentially knowing the MDP transition dynamics. We work in the stochastic MDP setting and assume no access to the MDP’s transition kernel. Further, it is assumed in (Agarwal et al., 2020b) that all the atomic controllers available are *stabilizing* which, when working with an unknown MDP, is a very strong assumption to make. While making no such assumptions on our atomic controller class; we show our algorithms can begin with provably unstable controllers and yet succeed in stabilizing the system (Sec. 2.2 and 6).

Options framework. Our work differs from the options framework (Barreto et al., 2017; Sutton et al., 1999) for hierarchical RL *in spirit*, in that we allow for each controller to be applied in each round rather than waiting for a sub-task to complete. The current work deals with finding an optimal mixture of basic controllers to solve a particular task. However, if we allow for a state-dependent choice of controllers, then the methods proposed can be generalized for solving hierarchical RL tasks.

Ensemble policy-based RL. Our current work deals with accessing *given* (possibly separately trained) controllers as *black-boxes* and *learning to combine* them optimally. In contrast, in ensemble RL approaches (Maclin & Opitz, 2011; Xiliang et al., 2018; Wiering & van Hasselt, 2008) the base policies are learnt on the fly (e.g., Q-learning, SARSA) by the agent whereas the *combining rule is fixed upfront* (e.g., majority voting, rank voting, Boltzmann multiplication, etc.). Moreover, the base policies *have access* to the new system in Ensemble RL, which gives them a distinct advantage. Our method can serve as a meta-RL adaptation framework with theoretical guarantees which can use such pre-trained models to combine them optimally. To the best of our knowledge, ensemble RL works like (Xiliang et al., 2018; Wiering & van Hasselt, 2008) do not provide theoretical guarantees on the learnt combined policy. Our work on the other hand provides a firm theoretical as well as empirical basis for the methods we propose.

Improper learning with given base controllers. Probably the closest resemblance with our work is that of Banijamali et al. (2019) which aims at finding the best convex combination of a given set of base controllers for a given MDP. They however frame it as a *planning* problem where the transition kernel P is known to the agent. Furthermore, we treat the base controllers as black-box entities, whereas they exploit their structure to compute the state-occupancy measures.

Actor-critic methods. Actor-critic (AC) methods were first introduced in Konda & Tsitsiklis (2000). Natural actor-critic methods were first introduced in (Peters & Schaal, 2008; Bhatnagar et al., 2009). While many studies are available for the asymptotic convergence of AC and NAC, we use the new techniques proposed by Xu et al. (2020) and Barakat

et al. (2021) for showing convergence results.

2. Motivating Examples

We begin with two examples that help illustrate the need for improper learning over a given set of atomic controllers. These examples concretely demonstrate the power of this approach to find (improper) control policies that go well beyond what the atomic set can accomplish, while retaining some of their desirable properties (such as interpretability and simplicity of implementation).

2.1. Ergodic Control of the Cartpole System

Consider the Cartpole system which has, over the years, become a benchmark for testing control strategies (Khalil, 2015). The system’s dynamics, evolving in \mathbb{R}^4 , can be approximated via a Linear Quadratic Regulator around an (unstable) equilibrium state vector that we designate the origin ($\mathbf{x} = \mathbf{0}$). The objective now reduces to finding a (potentially randomized) control policy $u \equiv \{u(t), t \geq 0\}$ that solves $\inf_u J (\mathbb{E}_u \sum_{t=0}^{\infty} \mathbf{x}^\top(t) Q \mathbf{x}(t) + Ru^2(t))$ subject to $\mathbf{x}(t+1) = A_{open} \mathbf{x}(t) + \mathbf{b}u(t)$ at all times $t \geq 0$.

Under standard assumptions of controllability and observability, this optimization has a stationary, linear solution $u^*(t) = -\mathbf{K}^\top \mathbf{x}(t)$ (Bertsekas, 2011). Moreover, setting $A := A_{open} - \mathbf{b}\mathbf{K}^\top$, it is well known that the dynamics $\mathbf{x}(t+1) = A\mathbf{x}(t)$, $t \geq 0$, are stable. The usual design strategy for a given Cartpole involves a combination of system identification, followed by linearization and computing the controller gain \mathbf{K} . This would typically produce a controller with tolerable performance fairly quickly, but would also suffer from nonidealities of parameter estimation.

To alleviate this problem, first consider a generic (ergodic) control policy that builds on this strategy by switching across a menu of controllers $\{K_1, \dots, K_N\}$ produced as above. That is, at any time t , this policy chooses K_i , $i \in [N]$, w.p. p_i , so that the control input at time t is $u(t) = -\mathbf{K}_i^\top \mathbf{x}(t)$ w.p. p_i . Let $A(i) := A_{open} - \mathbf{b}\mathbf{K}_i^\top$. The resulting *controlled* dynamics are given by $\mathbf{x}(t+1) = A(r(t))\mathbf{x}(t)$, $t \geq 0$, where $r(t) = i$ w.p. p_i , IID across t .

This is an example of an *ergodic parameter linear system* (EPLS) (Bolzern et al., 2008), which is said to be *Exponentially Almost Surely Stable* (EAS) if the state norm decays at least exponentially fast with time: $\mathbb{P} \left\{ \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(t)\| \leq -\rho \right\} = 1$ for some $\rho > 0$. Let the random variable $\lambda(\omega) := \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(t, \omega)\|$. For our dynamics $\mathbf{x}(t+1) = A(r(t))\mathbf{x}(t)$, $t \geq 0$, it is seen that the *Lyapunov exponent* $\frac{1}{t} \log \|\mathbf{x}(t)\|$ is at most the quantity $\sum_{i=1}^N p_i \log \|A(i)\|$ a.s. (see appendix for details).

A good mixture controller can now be designed by choosing $\{p_1, \dots, p_N\}$ such that $\lambda(\omega) < -\rho$ for some $\rho > 0$,

ensuring exponentially almost sure stability (subject to $\log \|A(i)\| < 0$ for some i). As we show in the sequel, our policy gradient algorithm (SoftMax PG) learns an improper mixture $\{p_1, \dots, p_N\}$ that (i) can stabilize the system even when a majority of the constituent *atomic* controllers $\{K_1, \dots, K_N\}$ are *unstable*, i.e., converges to a mixture that ensures that the average exponent $\lambda(\omega) < 0$, and (ii) shows better performance than that each of the atomic controllers.

2.2. Scheduling in Constrained Queuing Networks

We consider a system that comprises two queues fed by independent, stochastic arrival processes $A_i(t), i \in \{1, 2\}, t \in \mathbb{N}$. The length of queue i , measured at the beginning of time slot t , is denoted by $Q_i(t) \in \mathbb{Z}_+$.

A common server serves both queues and can drain at most one packet from the system in a time slot. The server, therefore, needs to decide which of the two queues it intends to serve in a given slot (we assume that once the server chooses to serve a packet, service succeeds with probability 1). The server's decision is denoted by the vector $\mathbf{D}(t) \in \mathcal{A} := \{[0, 0], [1, 0], [0, 1]\}$, where a "1" denotes service and a "0" denotes lack thereof. Let $\mathbb{E}A_i(t) = \lambda_i$, and note that the arrival rate $\lambda = [\lambda_1, \lambda_2]$ is **unknown** to the learner. We aim to find a (potentially randomized) policy π to minimize the discounted system backlog given by $J_\pi(\mathbf{Q}(0)) := \mathbb{E}_{\mathbf{Q}(0)}^\pi \sum_{t=0}^{\infty} \gamma^t (Q_1(t) + Q_2(t))$.

Any policy with $J_\pi(\cdot) < \infty$, is said to be *stabilizing* (or, equivalently, a *stable* policy). It is well known that there exist stabilizing policies iff $\lambda_1 + \lambda_2 < 1$ (Tassiulas & Ephremides, 1992). A policy π_{μ_1, μ_2} that chooses Queue i w.p. μ_i in every slot, can provably stabilize a system iff $\mu_i > \lambda_i, \forall i \in \{1, 2\}$. Now, assume our control set consists of two stationary policies K_1, K_2 with $K_1 \equiv \pi_{\varepsilon, 1-\varepsilon}$, $K_2 \equiv \pi_{1-\varepsilon, \varepsilon}$ and sufficiently small $\varepsilon > 0$. That is, we have $M = 2$ controllers K_1, K_2 . Clearly, neither of these can, by itself, stabilize a network with $\lambda = [0.49, 0.49]$.

However, an *improper* mixture of the two that selects K_1 and K_2 each with probability 1/2 can. In fact, as Fig. 1 shows, our improper learning algorithm can stabilize *all* arrival rates in $\mathcal{C}_1 \cup \mathcal{C}_2 \cup \Delta ABC$, without prior knowledge of $[\lambda_1, \lambda_2]$. In other words, our algorithm enlarges the stability region by the triangle ΔABC , over and above $\mathcal{C}_1 \cup \mathcal{C}_2$. We

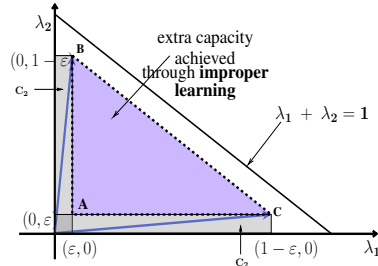


Figure 1: K_1 and K_2 by themselves can only stabilize $\mathcal{C}_1 \cup \mathcal{C}_2$ (gray rectangles). With improper learning, we enlarge the set of stabilizable arrival rates by the triangle ΔABC shown in purple, above.

will return to these examples in Sec. 6, and show, using experiments, (1) how our improper learner converges to the stabilizing mixture of the available policies and (2) if the optimal policy is among the available controllers, how our algorithm can find and converge to it.

3. Problem Statement and Notation

A (finite) Markov Decision Process $(\mathcal{S}, \mathcal{A}, \mathbf{P}, r, \rho, \gamma)$ is specified by a finite state space \mathcal{S} , a finite action space \mathcal{A} , a transition probability matrix \mathbf{P} , where $\mathbf{P}(\tilde{s}|s, a)$ is the probability of transitioning into state \tilde{s} upon taking action $a \in \mathcal{A}$ in state s , a single stage reward function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, a starting state distribution ρ over \mathcal{S} and a discount factor $\gamma \in (0, 1)$. A (stationary) *policy* or *controller* $\pi : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{A})$ specifies a decision-making strategy in which the learner chooses actions (a_t) adaptively based on the current state (s_t) , i.e., $a_t \sim \pi(s_t)$. π and ρ , together with \mathbf{P} , induce a probability measure \mathbb{P}_ρ^π on the space of all sample paths of the underlying Markov process and we denote by \mathbb{E}_ρ^π the associated expectation operator. The **value function** of policy π (also called the value of policy π), denoted by V^π is the total discounted reward obtained by following π , i.e., $V^\pi(\rho) := \mathbb{E}_\rho^\pi \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$.

Improper Learning. We assume that the learner is provided with a finite number of (stationary) controllers $\mathcal{C} := \{K_1, \dots, K_M\}$ and, as described below, set up a parameterized improper policy class $\mathcal{I}_{soft}(\mathcal{C})$ that depends on \mathcal{C} . The aim therefore, is to identify the best policy for the given MDP within this class, i.e.,

$$\pi^* = \operatorname{argmax}_{\pi \in \mathcal{I}_{soft}(\mathcal{C})} V^\pi(\rho). \quad (1)$$

We now describe the construction of the class $\mathcal{I}_{soft}(\mathcal{C})$.

The Softmax Policy Class. We assign weights $\theta_m \in \mathbb{R}$, to each controller $K_m \in \mathcal{C}$ and define $\theta := [\theta_1, \dots, \theta_M]$. The improper class \mathcal{I}_{soft} is parameterized by θ as follows. In each round, the policy $\pi_\theta \in \mathcal{I}_{soft}(\mathcal{C})$ chooses a controller drawn from $\operatorname{softmax}(\theta)$, i.e., the probability of choosing Controller K_m is given by, $\pi_\theta(m) := e^{\theta_m} / \left(\sum_{m'=1}^M e^{\theta_{m'}} \right)$. Note, therefore, that in every round, our algorithm interacts with the MDP only *through* the controller sampled in that round. In the rest of the paper, we will deal exclusively with a fixed and given \mathcal{C} and the resultant \mathcal{I}_{soft} . therefore, we overload the notation $\pi_{\theta_t}(a|s)$ for any $a \in \mathcal{A}$ and $s \in \mathcal{S}$ to denote the probability with which the algorithm chooses action a in state s at time t . For ease of notation, whenever the context is clear, we will also drop the subscript θ i.e., $\pi_{\theta_t} \equiv \pi_t$. Hence, we have at any time $t \geq 0 : \pi_{\theta_t}(a|s) = \sum_{m=1}^M \pi_{\theta_t}(m) K_m(s, a)$. Since we deal with gradient-based methods in the sequel, we define the **value gradient** of policy $\pi_\theta \in \mathcal{I}_{soft}$, by $\nabla_\theta V^{\pi_\theta} \equiv \frac{dV^{\pi_\theta}}{d\theta}$. We say that V^{π_θ} is β -smooth if $\nabla_\theta V^{\pi_\theta}$ is β -Lipschitz (Agarwal et al., 2020a). Finally, let for any two integers a and b ,

Algorithm 1 SoftMax PG

Input: learning rate $\eta > 0$, initial state distribution μ
Initialize: each $\theta_m^1 = 1$, for all $m \in [M]$, $s_1 \sim \mu$.
for $t = 1$ **to** T **do**
 Choose controller $m_t \sim \pi_t$
 Play action $a_t \sim K_{m_t}(s_t, \cdot)$
 Observe $s_{t+1} \sim P(\cdot | s_t, a_t)$
 Update: $\theta_{t+1} = \theta_t + \eta \nabla_{\theta_t} V^{\pi_{\theta_t}}$
end for

$\mathbb{1}_{ab}$ denote the indicator that $a = b$.

Comparison to the standard PG setting. This problem we define is different from the usual policy gradient setting where the parameterization completely defines the policy in terms of the state-action mapping. One can use the methodology followed in (Mei et al., 2020), by assigning a parameter $\theta_{s,m}$ for every $s \in \mathcal{S}, m \in [M]$. With some calculation, it can be shown that this is equivalent to the tabular setting with S states and M actions, with the new ‘reward’ defined by $r(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) r(s, a)$ where $r(s, a)$ is the usual expected reward obtained at state s and playing action $a \in \mathcal{A}$. By following the approach in (Mei et al., 2020) on this modified setting, it can be shown that the policy converges for each $s \in \mathcal{S}$, $\pi_{\theta}(m^*(s) | s) \rightarrow 1$, for every $s \in \mathcal{S}$, which is the optimum policy. However, the problem that we address, is to select a *single* controller (from within \mathcal{I}_{soft} , the convex hull of the given M controllers), which would guarantee maximum return if one plays that single mixture for all time, from among the given set of controllers.

4. Improper Learning using Gradients

In this and the following sections, we propose and analyze a policy gradient-based algorithm that provably finds the best, potentially improper, mixture of controllers for the given MDP. While we employ gradient ascent to optimize the mixture weights, the fact that this procedure works at all is far from obvious. We begin by noting that $V^{\pi_{\theta}}$, as described in Section 3, is *nonconcave* in θ for both direct and softmax parameterizations, which renders analysis with standard tools of convex optimization inapplicable.

Lemma 4.1. (Non-concavity of Value function) *There is an MDP and a set of controllers, for which the maximization problem of the value function (i.e. (1)) is non-concave for both the SoftMax and direct parameterizations, i.e., $\theta \mapsto V^{\pi_{\theta}}$ is non-concave.*

The proof follows from a counterexample whose construction we show in the appendix. Our PG algorithm, **SoftMax PG**, is shown in Algorithm 1. The parameters $\theta \in \mathbb{R}^M$ which define the policy are updated by following the gradient of the value function at the current policy parameters.

Convergence Guarantees. The following result shows that

with SoftMax PG, the value function converges to that of the *best in-class* policy at a rate $\mathcal{O}(1/t)$. Furthermore, the theorem shows an explicit dependence on the number of controllers M , in place of the usual $|\mathcal{S}|$. Note that with perfect gradient knowledge the algorithm becomes deterministic. This is a standard assumption in the analysis of PG algorithms (Fazel et al., 2018; Agarwal et al., 2020a; Mei et al., 2020).

Theorem 4.2 (Convergence of Policy Gradient). *With $\{\theta_t\}_{t \geq 1}$ generated as in Algorithm 1 and using a learning rate $\eta = \frac{(1-\gamma)^2}{7\gamma^2 + 4\gamma + 5}$, for all $t \geq 1$, $V^*(\rho) - V^{\pi_{\theta_t}}(\rho) = \mathcal{O}\left(\frac{1}{t} \frac{M\gamma^2}{c_t^2(1-\gamma)^3}\right)$, where $c_t := \min_{1 \leq s \leq t} \min_{m: \pi^*(m) > 0} \pi_{\theta_s}(m)$.*

Remark 4.3. The quantity c_t in the statement is the minimum probability that SoftMax PG puts on the controllers for which the best mixture π^* has positive probability mass. Empirical evidence (Sec. 6) makes us conjecture that $\lim_{t \rightarrow \infty} c_t$ is *positive*, which shows a convergence rate of $\mathcal{O}(1/t)$.

Remark 4.4. The proof of the above theorem uses the β -smoothness property of the value function under the softmax parameterization along with a new non-uniform Łojasevicz-type inequality (NUŁI) for our probabilistic mixture class, which lower bounds the magnitude of the gradient of the value function, which we mention below.

Lemma 4.5 (NUŁI). $\left\| \frac{\partial}{\partial \theta} V^{\pi_{\theta}}(\mu) \right\|_2 \geq \frac{1}{\sqrt{M}} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \times \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-1} \times [V^*(\rho) - V^{\pi_{\theta}}(\rho)].$

The proof of Theorem 4.2, then follows by an induction argument over $t \geq 1$.

Technical Challenges. We note here that while the basic recipe for the analysis of Theorem 4.2 is similar to (Mei et al., 2020), our setting does not directly inherit the intuition of standard PG (sPG) analysis. **(1)** With $|\mathcal{S} \times \mathcal{A}| < \infty$, the sPG analysis critically depends on the fact that a deterministic optimal policy exists and shows convergence to it. In contrast, in our setting, π^* could be a strictly randomized mixture of the base controllers (see Sec. 2). **(2)** A crucial step in sPG analysis is establishing that the value function $V^{\pi}(s), \forall s \in \mathcal{S}$ increases monotonically with time such that parameter of the optimal action $\theta_{s,a^*} \uparrow \infty$. In the appendix, we supply a simple counterexample showing that monotonicity of the V function is not guaranteed in our setting for every $s \in \mathcal{S}$. **(3)** The value function gradient in sPG has no ‘cross contamination’ from other states, in the sense that modifying the parameter at one state does not affect the values of the others. This plays a crucial part in simplifying the proof of global convergence to the optimal policy in sPG analysis. Our setting cannot leverage this property since the value function gradient at a given *controller* possesses contributions from *all* states.

For the special case of $S = 1$, which is the Multiarmed

Bandits, each controller is a probability distribution over the A arms of the bandit. We call this special case *Bandit-over-Bandits*. We obtain a convergence rate of $\mathcal{O}(M^2/t)$ to the optimum and recover $M^2 \log T$ regret bound when our softmax PG algorithm is applied to this special case. We refer to the appendix for details.

Discussion on c_t . Convergence in Theorem 4.2 depends inversely on c_t^2 . It follows that in order for SoftMax PG to converge, c_t must either (a) converge to a positive constant, or (b) decay (to 0) slower than $\mathcal{O}(1/\sqrt{t})$. The technical challenges discussed above, render proving this extremely hard analytically. Hence, while we currently do not show this theoretically, our experiments in Sec. 6 repeatedly confirm that its empirical analog, i.e., \bar{c}_t (defined formally in Sec. 6) approaches a *positive* value. Hence, we conjecture that the rate of convergence in Thm 4.2 is $\mathcal{O}(1/t)$.

5. Actor-Critic based Improper Learning

Softmax PG follows a gradient ascent scheme to solve the optimization problem (1), but is limited by the requirement of the true gradient in every round. To address situations where this might be unavailable, we resort to a Monte-carlo sampling based procedure (see appendix: Alg 6), which may lead to high variance. In this section, we take an alternative approach and provide a new algorithm based on an actor-critic framework for solving our problem. Actor-Critic methods are well-known to have low variance than their Monte-carlo counterparts (Konda & Tsitsiklis, 2000).

We begin by proposing modifications to the standard Q -function and advantage function definitions. Recall that we wish to solve for the following optimization problem: $\max_{\pi \in \mathcal{I}_{\text{soft}}} \mathbb{E}_{s \sim \rho} [V^\pi(s)]$, where π is some distribution over the M base controllers. Let $\tilde{Q}^\pi(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) Q^\pi(s, a)$. Let $\tilde{A}^\pi(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) A^\pi(s, a) = \sum_{a \in \mathcal{A}} K_m(s, a) Q^\pi(s, a) - V^\pi(s)$, where Q^π and A^π are the usual action-value functions and advantage functions respectively. We also define the new reward function $\tilde{r}(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) r(s, a)$ and a new transition kernel $P(s'|s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) P(s'|s, a)$. Then, following the distribution π over the controllers induces a Markov Chain on the state space \mathcal{S} . Define $\nu_\pi(s, m)$ as the state-controller visitation measure induced by the *policy* π : $\nu_\pi(s, m) := (1 - \gamma) \sum_{t > 0} \gamma^t \mathbb{P}^\pi(s_t = s, m_t = m) = d_\rho^\pi(s) \pi(m)$. With these definitions, we have the following variant of the policy-gradient theorem.

Lemma 5.1 (Modified Policy Gradient Theorem). $\nabla_\theta V^{\pi_\theta}(\rho) = \mathbb{E}_{(s, m) \sim \nu_{\pi_\theta}} [\tilde{Q}^{\pi_\theta}(s, m) \psi_\theta(m)] = \mathbb{E}_{(s, m) \sim \nu_{\pi_\theta}} [\tilde{A}^{\pi_\theta}(s, m) \psi_\theta(m)]$, where $\psi_\theta(m) := \nabla_\theta \log(\pi_\theta(m))$.

Note the independence of the score function ψ from the state

Algorithm 2 Actor-Critic based Improper RL (ACIL)

Input: φ , actor stepsize α , critic stepsize β , regularization parameter λ , 'AC' or 'NAC'
Initialize: $\theta_0 = (1, 1, \dots, 1)_{M \times 1}$, $s_0 \sim \rho$
flag = $\mathbb{1}\{\text{NAC}\}$ {Selects AC or NAC}
for $t \leftarrow 0$ **to** $T - 1$ **do**
 $s_{init} = s_{t-1, B}$ (when $t = 0$, $s_{init} = s_0$)
 $w_t, s_{t, 0} \leftarrow \text{Critic-TD}(s_{init}, \pi_{\theta_t}, \varphi, \beta, T_c, H)$
 $F_t(\theta_t) \leftarrow 0$.
 for $i \leftarrow 0$ **to** $B - 1$ **do**
 $m_{t, i} \sim \pi_{\theta_t}$, $a_{t, i} \sim K_{m_{t, i}}(s_{t, i}, \cdot)$
 $s_{t, i+1} \sim \tilde{P}(\cdot | s_{t, i}, m_{t, i})$
 $\mathcal{E}_{w_t}(s_{t, i}, m_{t, i}, s_{t, i+1}) = \tilde{r}(s_{t, i}, m_{t, i}) + (\gamma \varphi(s_{t, i+1}) - \varphi(s_{t, i}))^\top w_t$
 $F_t(\theta_t) \leftarrow F_t(\theta_t) + \frac{1}{B} \psi_{\theta_t}(m_{t, i}) \psi_{\theta_t}(m_{t, i})^\top$
 end for
 if {flag} **then**
 $G_t := [F_t(\theta_t) + \lambda I]$
 $\theta_{t+1} = \theta_t + G_t^{-1} \frac{\alpha}{B} \sum_{i=0}^{B-1} \mathcal{E}_{w_t}(s_{t, i}, m_{t, i}, s_{t, i+1}) \psi_{\theta_t}(m_{t, i})$
 else
 $\theta_{t+1} = \theta_t + \frac{\alpha}{B} \sum_{i=0}^{B-1} \mathcal{E}_{w_t}(s_{t, i}, m_{t, i}, s_{t, i+1}) \psi_{\theta_t}(m_{t, i})$
 end if
 $\pi_{\theta_{t+1}} = \text{softmax}(\theta_{t+1})$
 end for
Output: $\theta_{\hat{T}}$ with \hat{T} chosen uniformly at random from $\{1, \dots, T\}$

s . For the gradient ascent update of the parameters θ we need to estimate $\tilde{A}^{\pi_\theta}(s, m)$ where (s, m) are drawn according to $\nu_{\pi_\theta}(\cdot, \cdot)$. We recall how to sample from ν_π . Following Konda & Tsitsiklis (2000) and the recent works like Xu et al. (2020); Barakat et al. (2021) and casting into our setting, observe that ν_π is a stationary distribution of a Markov chain over the pair (s, m) with state-to-state transition kernel defined by $\tilde{P}(s'|s, m) := \gamma \tilde{P}(s'|s, m) + (1 - \gamma) \rho(s')$ and $m \sim \pi(\cdot)$.

Algorithm Description. We present the algorithm in detail in Algorithm 2 along with a subroutine Alg 3 which updates the critic's parameters. ACIL is a single-trajectory based algorithm, in the sense that it does not require a forced reset along the run. We begin with the critic's updates. The **critic** uses linear function approximation $V_w(s) := \varphi(s)^\top w$, and uses TD learning to update its parameters $w \in \mathbb{R}^d$. We assume that $\varphi(\cdot) : \mathcal{S} \rightarrow \mathbb{R}^d$ is a known feature mapping. Let Φ be the corresponding $|\mathcal{S}| \times d$ matrix. We assume that the columns of Φ are linearly independent. Next, based on the critic's parameters, the **actor** approximates the $\tilde{A}(s, m)$ function using the TD error: $\mathcal{E}_w(s, m, s') = \tilde{r}(s, m) + (\gamma \varphi(s') - \varphi(s))^\top w$.

In order to provide guarantees of the convergence rates

Algorithm 3 Critic-TD Subroutine

Input: $s_{init}, \pi, \varphi, \beta, T_c, H$
Initialize: w_0
for $k \leftarrow 0$ **to** $T_c - 1$ **do**
 $s_{k,0} = s_{k-1,H}$ (when $k = 0, s_{k,0} = s_{init}$)
 for $j \leftarrow 0$ **to** $H - 1$ **do**
 $m_{k,j} \sim \pi(\cdot), a_{k,j} \sim K_{m_{k,j}}(s_{k,j}, \cdot)$
 $s_{k,j+1} \sim \tilde{P}(\cdot | s_{k,j}, m_{k,j})$
 $\mathcal{E}_{w_k}(s_{k,j}, m_{k,j}, s_{k,j+1}) = \tilde{r}(s_{k,j}, m_{k,j}) +$
 $(\gamma\varphi(s_{k,j+1}) - \varphi(s_{k,j}))^\top w_k$
 end for
 $w_{k+1} = w_k + \frac{\beta}{H} \sum_{i=0}^{H-1} \mathcal{E}_{w_k}(s_{k,i}, m_{k,i}, s_{k,i+1})\varphi(s_{k,i})$
end for
Output: $w_{T_c}, s_{T_c-1,H}$

of Algorithm ACIL, we make the following assumptions, which are standard in RL literature (Konda & Tsitsiklis, 2000; Bhandari et al., 2018; Xu et al., 2020).

Assumption 5.2 (Uniform Ergodicity). For any $\theta \in \mathbb{R}^M$, consider the Markov Chain induced by the policy π_θ , and following the transition kernel $\tilde{P}(\cdot | s, m)$. Let ξ_{π_θ} be the stationary distribution of this Markov Chain. We assume that there exists constants $\kappa > 0$ and $\xi \in (0, 1)$ such that

$$\sup_{s \in \mathcal{S}} \|\mathbb{P}(s_t \in \cdot | s_0 = s, \pi_\theta) - \xi_{\pi_\theta}(\cdot)\|_{TV} \leq \kappa \xi^t.$$

Further, let $L_\pi := \mathbb{E}_{\nu_\pi}[\varphi(s)(\gamma\varphi(s') - \varphi(s))^\top]$ and $v_\pi := \mathbb{E}_{\nu_\pi}[r(s, m, s')\varphi(s)]$. The optimal solution to the critic's TD learning is now $w^* := -L_\pi^{-1}v_\pi$.

Assumption 5.3. There exists a positive constant Γ_L such that for all $w \in \mathbb{R}^d$, we have $\langle w - w^*, L_\pi(w - w^*) \rangle \leq -\Gamma_L \|w - w^*\|_2^2$.

Based on the above two assumptions, let $L_V := \frac{2\sqrt{2}C_{\kappa\xi} + 1}{1-\gamma}$, where $C_{\kappa\xi} = \left(1 + \lceil \log_\xi \frac{1}{\kappa} \rceil + \frac{1}{1-\xi}\right)$.

Theorem 5.4. Consider the Actor-Critic improper learning algorithm ACIL (Alg 2). Assume $\sup_{s \in \mathcal{S}} \|\varphi(s)\|_2 \leq 1$. Under Assumptions 5.2 and 5.3 with step-sizes chosen as $\alpha = \left(\frac{1}{4L_V\sqrt{M}}\right)$, $\beta = \min\{\mathcal{O}(\Gamma_L), \mathcal{O}(1/\Gamma_L)\}$, batch-sizes $H = \mathcal{O}\left(\frac{1}{\varepsilon}\right)$, $B = \mathcal{O}(1/\varepsilon)$, $T_c = \mathcal{O}\left(\frac{\sqrt{M}}{\Gamma_L} \log(1/\varepsilon)\right)$, $T = \mathcal{O}\left(\frac{\sqrt{M}}{(1-\gamma)^2\varepsilon}\right)$, we have $\mathbb{E}[\|\nabla_\theta V(\theta_{\hat{T}})\|_2^2] \leq \varepsilon + \mathcal{O}(\Delta_{critic})$. Hence, the total sample complexity is $\mathcal{O}(M(1-\gamma)^{-2}\varepsilon^{-2} \log(1/\varepsilon))$.

Here, $\Delta_{critic} := \max_{\theta \in \mathbb{R}^M} \mathbb{E}_{\nu_{\pi_\theta}} \left[\left| V^{\pi_\theta}(s) - V^{w_{\pi_\theta}^*} \right|^2 \right]$, which equals zero, if the value function lies in the linear space spanned by the features.

Next we provide the [global optimality](#) guarantee for the Natural-Actor-Critic version of ACIL.

Theorem 5.5. Assume $\sup_{s \in \mathcal{S}} \|\varphi(s)\|_2 \leq 1$. Under Assumptions 5.2 and 5.3 with step-sizes chosen as $\alpha = \left(\frac{\lambda^2}{2\sqrt{M}L_V(1+\lambda)}\right)$, $\beta = \min\{\mathcal{O}(\Gamma_L), \mathcal{O}(1/\Gamma_L)\}$, batch-sizes $H = \mathcal{O}\left(\frac{1}{\Gamma_L\varepsilon^2}\right)$, $B = \mathcal{O}\left(\frac{1}{(1-\gamma)^2\varepsilon^2}\right)$, $T_c = \mathcal{O}\left(\frac{\sqrt{M}}{\Gamma_L} \log(1/\varepsilon)\right)$, $T = \mathcal{O}\left(\frac{\sqrt{M}}{(1-\gamma)^2\varepsilon}\right)$ and $\lambda = \mathcal{O}(\Delta_{critic})$ we have $V(\pi^*) - \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[V(\pi_{\theta_t})] \leq \varepsilon + \mathcal{O}\left(\sqrt{\frac{\Delta_{actor}}{(1-\gamma)^3}}\right) + \mathcal{O}(\Delta_{critic})$. Hence, the total sample complexity is $\mathcal{O}\left(\frac{M}{(1-\gamma)^4\varepsilon^3} \log \frac{1}{\varepsilon}\right)$.

where $\Delta_{actor} := \max_{\theta \in \mathbb{R}^M} \min_{w \in \mathbb{R}^d} \mathbb{E}_{\nu_{\pi_\theta}} [\|\psi_\theta^\top w - A_{\pi_\theta}(s, m)\|_2^2]$ and Δ_{critic} is same as before.

6. Numerical Results

6.1. Simulations with Softmax PG

We now discuss the results of implementing Softmax PG (Alg 1) on the cartpole system and on the constrained queuing examples described in Sec. 2. Since neither value functions nor value gradients for these problems are available in closed-form, we modify SoftMax PG (Algorithm 1) to make it generally implementable using a combination of (1) *rollouts* to estimate the value function of the current (improper) policy and (2) *simultaneous perturbation stochastic approximation* (SPSA) to estimate its value gradient. Specifically, we use the approach in (Flaxman et al., 2005), noting that for a function $V : \mathbb{R}^M \rightarrow \mathbb{R}$, the gradient, $\nabla V(\theta) \approx \mathbb{E}[(V(\theta + \alpha.u) - V(\theta))u] \cdot \frac{M}{\alpha}$, where the perturbation parameter $\alpha \in (0, 1)$ and u is sampled uniformly randomly from the unit sphere. This expression requires evaluation of the value function at the point $(\theta + \alpha.u)$. Since the value function may not be explicitly computable, we employ rollouts, for its evaluation. The full algorithm, *GradEst*, can be found in the appendix (Alg. 6).

Note that all of the simulations shown have been averaged over $\#L = 20$ trials, and the mean and standard deviations plotted. We also show empirically that c_t in Theorem 4.2 is indeed strictly positive. In the sequel, for every trial $l \in [\#L]$, let $\bar{c}_t^l := \inf_{1 \leq s \leq t} \min_{m \in \{m' \in [M] : \pi^*(m') > 0\}} \pi_{\theta_s}(m)$, and $\bar{c}_t := \frac{1}{\#L} \sum_{l=1}^{\#L} \bar{c}_t^l$. Also let $\bar{c}^T := \min_{l \in [\#L]} \min_{1 \leq t \leq T} \bar{c}_t^l$. That is the sequences $\{\bar{c}_t^l\}_{t=1, l=1}^{T, \#L}$ define the minimum probabilities that the algorithm puts, over rounds $1 : t$ in trial l , on controllers with $\pi^*(\cdot) > 0$. $\{\bar{c}_t^l\}_{t=1}^T$ represents its average across the different trials, and \bar{c}^T is the minimum such probability that the algorithm learns across all rounds $1 \leq t \leq T$ and across trials.

Simulations for the Cartpole. We study two different settings for the Cartpole example. Let K_{opt} be the optimal controller for the given system, computed via standard procedures (details can be found in (Bertsekas, 2011)). We

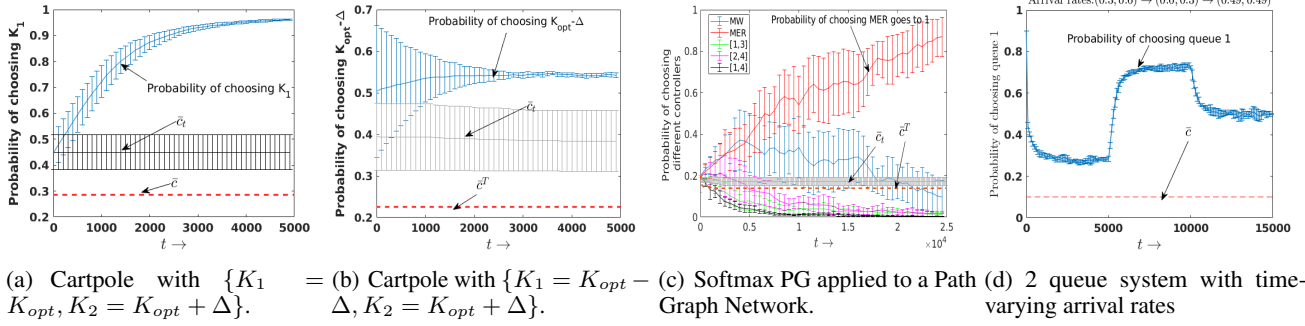


Figure 2: Softmax PG algorithm applied to the cartpole control and path graph scheduling tasks. Each plot shows (a) the learnt probabilities of various base controllers over time, and (b) the minimum probability \bar{c}_t and \bar{c}^T as described in text.

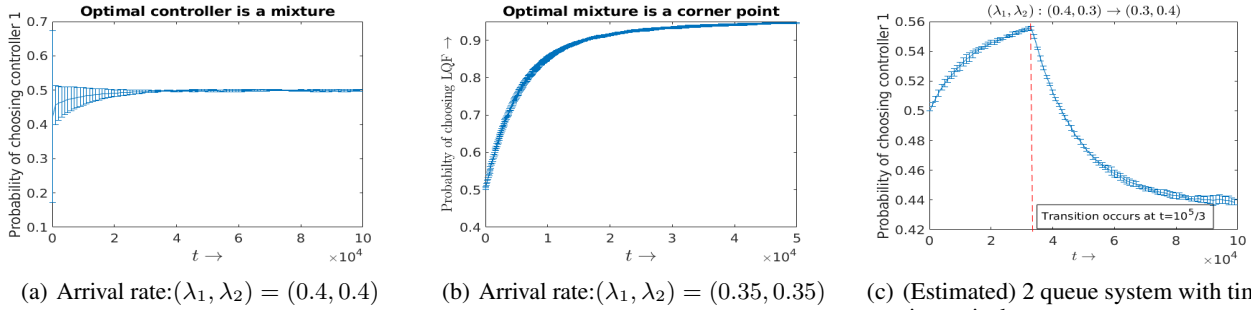


Figure 3: Natural-actor-critic based improper learning algorithm applied to various queuing networks show convergence to the best mixture policy.

set $M = 2$ and consider two scenarios: (i) the two base controllers are $\mathcal{C} \equiv \{K_{opt}, K_{opt} + \Delta\}$, where Δ is a random matrix, each entry of which is drawn IID $\mathcal{N}(0, 0.1)$, (ii) $\mathcal{C} \equiv \{K_{opt} - \Delta, K_{opt} + \Delta\}$. In the first case a corner point of the simplex is optimal. In the second case a strict improper mixture of the available controllers is optimum. As we can see in Fig. 2(a) and 2(b) our policy gradient algorithm converges to the best controller/mixture in both the cases. The details of all the hyperparameters for this setting are provided in the appendix. We note here that in the second setting even though none of the controllers, applied individually, stabilizes the system, our Softmax PG algorithm finds and follows a improper mixture of the controllers which stabilizes the given Cartpole.

Constrained Queuing Networks. We present simulation results for the following networks.

(i) Path Graph Networks. The scheduling constraints in the first network we study dictate that Queues i and $i+1$ cannot be served simultaneously for $i \in [N - 1]$ in any round $t \geq 0$. Such queuing systems are called *path graph* networks (Mohan et al., 2020). We work with $N = 4$. Therefore, sets of queues which can be served simultaneously are $\mathcal{A} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 3\}, \{2, 4\}, \{1, 4\}\}$. The constituents of \mathcal{A} are called *independent sets* in the literature. In each round t , the scheduler selects an independent set to serve the queues therein. Let $Q_j(t)$ be the backlog of Queue j at time t . We use the follow-

ing base controllers: (i) K_1 : Max Weight (MW) controller (Tassioulas & Ephremides, 1992) chooses a set $s_t := \arg\max_{S \in \mathcal{A}} \sum_{j \in S} Q_j(t)$, i.e, the set with the largest backlog, (ii) K_2 : Maximum Egress Rate (MER) controller chooses a set $s_t := \arg\max_{S \in \mathcal{A}} |\{Q_j(t) > 0\}|$, i.e, the set which has the maximum number of non-empty queues. We also choose K_3, K_4 and K_5 which serve the sets $\{1, 3\}, \{2, 4\}, \{1, 4\}$ respectively with probability 1. We fix the arrival rates to the queues $(0.495, 0.495, 0.495, 0.495)$. It is well known that the MER rule is mean-delay optimal in this case (Mohan et al., 2020). In Fig. 2(c), we plot the probability of choosing $K_i, i \in [5]$, learnt by our algorithm. The probability of choosing MER indeed converges to 1.

(ii) Non-stationary arrival rates. Recall the example discussed in Sec. 2.2 of two queues. The scheduler there is now given two base/atomic controllers $\mathcal{C} := \{K_1, K_2\}$, i.e. $M = 2$. Controller K_i serves Queue i with probability 1, $i = 1, 2$. As can be seen in Fig. 2(d), the arrival rates λ to the two queues vary over time (adversarially) during the learning. In particular, λ varies from $(0.3, 0.6) \rightarrow (0.6, 0.3) \rightarrow (0.49, 0.49)$. Our PG algorithm successfully *tracks* this change and *adapts* to the optimal improper stationary policies in each case.

In all the simulations shown above we note that the empirical trajectories of \bar{c}_t and \bar{c}^T become flat after some initial rounds and are bounded away from zero. This supports our conjecture that $\lim_{t \rightarrow \infty} c_t$ in Theorem 4.2 is bounded away

from zero, rendering the theorem statement non-vacuous. Note that Alg. 1 performs well in challenging scenarios, even with estimates of the value function and its gradient.

6.2. Simulations with ACIL

We perform some queueing theoretic simulations on the natural actor critic version of ACIL, which we will call NACIL in this section. Unlike Softmax PG, ACIL estimates gradients using temporal difference instead of SPSA. We study three different settings (1) where in the first case the optimal policy is a strict improper combination of the available controllers and (2) where it is at a corner point, i.e., one of the available controllers itself is optimal (3) arrival rates are time-varying as in the previous section. Our simulations show that in all the cases, ACIL converges to the correct controller mixture.

Recall the example that we discussed in Sec. 2.2. We consider the case with Bernoulli arrivals with rates $\lambda = [\lambda_1, \lambda_2]$ and are given two base/atomic controllers $\{K_1, K_2\}$, where controller K_i serves Queue i with probability 1, $i = 1, 2$. As can be seen in Fig. 3(a) when $\lambda = [0.4, 0.4]$ (equal arrival rates), NACIL converges to an improper mixture policy that serves each queue with probability $[0.5, 0.5]$. Next in Fig 3(b) shows a situation where one of the base controllers, i.e., the “Longest-Queue-First” (LQF) is the optimal controller. NACIL converges correctly to the corner point.

Lastly, Fig. 3(c) shows a setting similar to (ii) Sec. 6.1 above. Here there is a single transition of (λ_1, λ_2) from $(0.4, 0.3) \rightarrow (0.3, 0.4)$ which occurs at $t = \lceil 10^5/3 \rceil$, which is unknown to the learner. We show the probability of choosing controller 1. NACIL tracks the changing arrival rates over time. We supply some more simulations with NACIL in the appendix due to space limitations.

Acknowledgment

This work was partially supported by the Israel Science Foundation under contract 2199/20. Mohammadi Zaki was supported by the Aerospace Network Research Consortium (ANRC) Grant on Airplane IOT Data Management.

References

- Abbasi-Yadkori, Y. and Szepesvári, C. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pp. 1–26, Budapest, Hungary, 09–11 Jun 2011. JMLR Workshop and Conference Proceedings.
- Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. Optimality and approximation with policy gradient methods in markov decision processes. In *Proceedings of Thirty Third Conference on Learning Theory*, pp. 64–66. PMLR, 2020a.
- Agarwal, N., Bruckhim, N., Hazan, E., and Lu, Z. Boosting for control of dynamical systems. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 96–103. PMLR, 13–18 Jul 2020b.
- Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pp. 322–331, 1995.
- Auer, P., Jaksch, T., and Ortner, R. Near-optimal regret bounds for reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 21, pp. 89–96. Curran Associates, Inc., 2009.
- Banijamali, E., Abbasi-Yadkori, Y., Ghavamzadeh, M., and Vlassis, N. Optimizing over a restricted policy class in mdps. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 3042–3050. PMLR, 16–18 Apr 2019.
- Barakat, A., Bianchi, P., and Lehmann, J. Analysis of a target-based actor-critic algorithm with linear function approximation. *CoRR*, abs/2106.07472, 2021.
- Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. Successor features for transfer in reinforcement learning. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- Bertsekas, D. P. Dynamic programming and optimal control 3rd edition, volume ii. *Belmont, MA: Athena Scientific*, 2011.
- Bhandari, J. and Russo, D. Global optimality guarantees for policy gradient methods. *ArXiv*, abs/1906.01786, 2019.

- Bhandari, J., Russo, D., and Singal, R. A finite time analysis of temporal difference learning with linear function approximation. *Oper. Res.*, 69:950–973, 2018.
- Bhatnagar, S., Sutton, R. S., Ghavamzadeh, M., and Lee, M. Natural actor–critic algorithms. *Automatica*, 45(11): 2471–2482, 2009. ISSN 0005-1098.
- Bolzern, P., Colaneri, P., and De Nicolao, G. Almost sure stability of stochastic linear systems with ergodic parameters. *European Journal of Control*, 14(2):114–123, 2008.
- Borkar, V. S. *Stochastic Approximation*. Cambridge Books. Cambridge University Press, December 2008.
- Cassel, A., Cohen, A., and Koren, T. Logarithmic regret for learning linear quadratic regulators efficiently. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1328–1337. PMLR, 13–18 Jul 2020.
- Chen, X. and Hazan, E. Black-box control for linear dynamical systems. *arXiv preprint arXiv:2007.06650*, 2020.
- Daniely, A., Linial, N., and Shalev-Shwartz, S. More data speeds up training time in learning halfspaces over sparse vectors. In *Advances in Neural Information Processing Systems*, volume 26, pp. 145–153. Curran Associates, Inc., 2013.
- Daniely, A., Linial, N., and Shalev-Shwartz, S. From average case complexity to improper learning complexity. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC ’14, pp. 441–448, New York, NY, USA, 2014. Association for Computing Machinery.
- Dean, S., Mania, H., Matni, N., Recht, B., and Tu, S. On the Sample Complexity of the Linear Quadratic Regulator. *arXiv e-prints*, art. arXiv:1710.01688, October 2017.
- Denisov, D. and Walton, N. Regret analysis of a markov policy gradient algorithm for multi-arm bandits. *ArXiv*, abs/2007.10229, 2020.
- Durrett, R. *Probability: Theory and examples*, 2011.
- Fazel, M., Ge, R., Kakade, S. M., and Mesbahi, M. Global convergence of policy gradient methods for the linear quadratic regulator, 2018.
- Flaxman, A. D., Kalai, A. T., and McMahan, H. B. Online convex optimization in the bandit setting: Gradient descent without a gradient. SODA ’05, pp. 385–394, USA, 2005. Society for Industrial and Applied Mathematics.
- Gao, B. and Pavel, L. On the properties of the softmax function with application in game theory and reinforcement learning. *ArXiv*, abs/1704.00805, 2017.
- Gopalan, A. and Mannor, S. Thompson Sampling for Learning Parameterized Markov Decision Processes. In *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pp. 861–898, Paris, France, 03–06 Jul 2015. PMLR.
- Ibrahimi, M., Javanmard, A., and Roy, B. Efficient reinforcement learning for high dimensional linear quadratic systems. In *Advances in Neural Information Processing Systems*, volume 25, pp. 2636–2644. Curran Associates, Inc., 2012.
- Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning*, 2002.
- Khalil, H. K. *Nonlinear Control*. Pearson, 2015.
- Kocák, T., Neu, G., Valko, M., and Munos, R. Efficient learning by implicit exploration in bandit problems with side observations. In *Advances in Neural Information Processing Systems*, volume 27, pp. 613–621. Curran Associates, Inc., 2014.
- Konda, V. and Tsitsiklis, J. Actor-critic algorithms. In Solla, S., Leen, T., and Müller, K. (eds.), *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 2000.
- Lai, T. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1): 4 – 22, 1985. ISSN 0196-8858.
- Li, G., Wei, Y., Chi, Y., Gu, Y., and Chen, Y. Softmax policy gradient methods can take exponential time to converge. *arXiv preprint arXiv:2102.11270*, 2021.
- Littlestone, N. and Warmuth, M. K. The weighted majority algorithm. *Inform. Comput.*, 108(2):212–261, 1994.
- Łojasiewicz, S. *Les équations aux dérivées partielles (paris, 1962)*, 1963.
- Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments, 2020.
- Maclin, R. and Opitz, D. W. Popular ensemble methods: An empirical study. *CoRR*, abs/1106.0257, 2011.
- Mania, H., Tu, S., and Recht, B. Certainty equivalence is efficient for linear quadratic control. In *Advances in Neural Information Processing Systems*, volume 32, pp. 10154–10164. Curran Associates, Inc., 2019.

- Mei, J., Xiao, C., Szepesvari, C., and Schuurmans, D. On the global convergence rates of softmax policy gradient methods. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 6820–6829. PMLR, 2020.
- Mohan, A., Chattopadhyay, A., and Kumar, A. Hybrid mac protocols for low-delay scheduling. In *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 47–55, Los Alamitos, CA, USA, oct 2016. IEEE Computer Society.
- Mohan, A., Gopalan, A., and Kumar, A. Throughput optimal decentralized scheduling with single-bit state feedback for a class of queueing systems. *ArXiv*, abs/2002.08141, 2020.
- Neu, G. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 28, pp. 3168–3176. Curran Associates, Inc., 2015.
- Osband, I., Russo, D., and Van Roy, B. (more) efficient reinforcement learning via posterior sampling. In *Advances in Neural Information Processing Systems*, volume 26, pp. 3003–3011. Curran Associates, Inc., 2013.
- Ouyang, Y., Gagrani, M., Nayyar, A., and Jain, R. Learning unknown markov decision processes: A thompson sampling approach. In *NIPS*, 2017.
- Peters, J. and Schaal, S. Natural actor-critic. *Neurocomputing*, 71(7):1180–1190, 2008. ISSN 0925-2312. Progress in Modeling, Theory, and Application of Computational Intelligenc.
- Radac, M.-B. and Precup, R.-E. Data-driven model-free slip control of anti-lock braking systems using reinforcement q-learning. *Neurocomput.*, 275(C):317–329, January 2018.
- Rummery, G. A. and Niranjan, M. On-line q-learning using connectionist systems. Technical report, 1994.
- Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897, Lille, France, 07–09 Jul 2015. PMLR.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017.
- Shani, L., Efroni, Y., and Mannor, S. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. *ArXiv*, abs/1909.02769, 2020.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–503, 2016.
- Singh, S., Okun, A., and Jackson, A. Artificial intelligence: Learning to play Go from scratch. 550(7676):336–337, October 2017. doi: 10.1038/550336a.
- Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.
- Sutton, R. S., Precup, D., and Singh, S. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1): 181–211, 1999. ISSN 0004-3702.
- Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, volume 12, pp. 1057–1063. MIT Press, 2000.
- Tassioulas, L. and Ephremides, A. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992. doi: 10.1109/9.182479.
- Wiering, M. A. and van Hasselt, H. Ensemble algorithms in reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(4):930–936, 2008.
- Xiliang, C., Cao, L., Li, C.-x., Xu, Z.-x., and Lai, J. Ensemble network architecture for deep reinforcement learning. *Mathematical Problems in Engineering*, 2018:1–6, 04 2018.
- Xu, T., Wang, Z., and Liang, Y. Improving sample complexity bounds for (natural) actor-critic algorithms. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 4358–4369. Curran Associates, Inc., 2020.

A. Glossary of Symbols

1. \mathcal{S} : State space
2. \mathcal{A} : Action space
3. S : Cardinality of \mathcal{S}
4. A : Cardinality of \mathcal{A}
5. M : Number of controllers
6. K_i Controller $i, i = 1, \dots, M$. For finite SA space MDP, K_i is a matrix of size $S \times A$, where each row is a probability distribution over the actions.
7. \mathcal{C} : Given collection of M controllers.
8. $\mathcal{I}_{soft}(\mathcal{C})$: Improper policy class setup by the learner.
9. $\theta \in \mathbb{R}^M$: Parameter assigned to the controllers to controllers, representing weights, updated each round by the learner.
10. $\pi(\cdot)$: Probability of choosing controllers
11. $\pi(\cdot | s)$ Probability of choosing action given state s . Note that in our setting, given $\pi(\cdot)$ over controllers (see previous item) and the set of controllers, $\pi(\cdot | s)$ is completely defined, i.e., $\pi(a | s) = \sum_{m=1}^M \pi(m)K_m(s, a)$. Hence we use simply π to denote the policy followed, whenever the context is clear.
12. $r(s, a)$: Immediate (one-step) reward obtained if action a is played in state s .
13. $\mathbb{P}(s' | s, a)$ Probability of transitioning to state s' from state s having taken action a .
14. $V^\pi(\rho) := \mathbb{E}_{s_0 \sim \rho} [V^\pi(s_0)] = \mathbb{E}_\rho \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)$ Value function starting with initial distribution ρ over states, and following policy π .
15. $Q^\pi(s, a) := \mathbb{E} \left[r(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathbb{P}(s' | s, a) V^\pi(s') \right]$.
16. $\tilde{Q}^\pi(s, m) := \mathbb{E} \left[\sum_{a \in \mathcal{A}} K_m(s, a) r(s, a) + \gamma \sum_{s' \in \mathcal{S}} \mathbb{P}(s' | s, a) V^\pi(s') \right]$.
17. $A^\pi(s, a) := Q^\pi(s, a) - V^\pi(s)$
18. $\tilde{A}(s, m) := \tilde{Q}^\pi(s, m) - V^\pi(s)$.
19. $d_\nu^\pi := \mathbb{E}_{s_0 \sim \nu} \left[(1 - \gamma) \sum_{t=0}^{\infty} \mathbb{P}[s_t = s | s_0, \pi, \mathbb{P}] \right]$. Denotes a distribution over the states, is called the ‘‘discounted state visitation measure’’
20. $c : \inf_{t > 1} \min_{m \in \{m' \in [M] : \pi^*(m') > 0\}} \pi_{\theta_t}(m)$.
21. $\left\| \frac{d_\mu^{\pi^*}}{\mu} \right\|_\infty = \max_s \frac{d_\mu^{\pi^*}(s)}{\mu(s)}$.
22. $\left\| \frac{1}{\mu} \right\|_\infty = \max_s \frac{1}{\mu(s)}$.

B. Expanded Survey of Related Work

In this section, we provide a detailed survey of related works. It is vital to distinguish the approach investigated in the present paper from the plethora of existing algorithms based on ‘proper learning’. Essentially, these algorithms try to find an (approximately) optimal policy for the MDP under investigation. These approaches can broadly be classified in two groups: *model-based* and *model-free*.

The former is based on first learning the dynamics of the unknown MDP followed by planning for this learnt model. Algorithms in this class include Thompson Sampling-based approaches (Osband et al., 2013; Ouyang et al., 2017; Gopalan & Mannor, 2015), Optimism-based approaches such as the UCRL algorithm (Auer et al., 2009), both achieving order-wise optimal $\mathcal{O}(\sqrt{T})$ regret bound.

A particular class of MDPs which has been studied extensively is the Linear Quadratic Regulator (LQR) which is a continuous state-action MDP with linear state dynamics and quadratic cost (Dean et al., 2017). Let $x_t \in \mathbb{R}^m$ be the current state and let $u_t \in \mathbb{R}^n$ be the action applied at time t . The infinite horizon average cost minimization problem for LQR is to find a policy to choose actions $\{u_t\}_{t>1}$ so as to minimize

$$\lim_{T \rightarrow \infty} \mathbb{E} \left[\frac{1}{T} \sum_{t=1}^T x_t^\top Q x_t + u_t^\top R u_t \right]$$

such that $x_{t+1} = Ax_t + Bu_t + n(t)$, $n(t)$ is iid zero-mean noise. Here the matrices A and B are unknown to the learner. Earlier works like (Abbasi-Yadkori & Szepesvári, 2011; Ibrahim et al., 2012) proposed algorithms based on the well-known optimism principle (with confidence ellipsoids around estimates of A and B). These show regret bounds of $\mathcal{O}(\sqrt{T})$.

However, these approaches do not focus on the stability of the closed-loop system. (Dean et al., 2017) describes a robust controller design which seeks to minimize the worst-case performance of the system given the error in the estimation process. They show a sample complexity analysis guaranteeing convergence rate of $\mathcal{O}(1/\sqrt{N})$ to the optimal policy for the given LQR, N being the number of rollouts. More recently, certainty equivalence (Mania et al., 2019) was shown to achieve $\mathcal{O}(\sqrt{T})$ regret for LQRs. Further, (Cassel et al., 2020) show that it is possible to achieve $\mathcal{O}(\log T)$ regret if either one of the matrices A or B are known to the learner, and also provided a lower bound showing that $\Omega(\sqrt{T})$ regret is unavoidable when both are unknown.

The *model-free* approach on the other hand, bypasses model estimation and directly learns the value function of the unknown MDP. While the most popular among these have historically been Q-learning, TD-learning (Sutton & Barto, 2018) and SARSA (Rummery & Niranjan, 1994), algorithms based on gradient-based policy optimization have been gaining considerable attention of late, following their stunning success with playing the game of Go which has long been viewed as the most challenging of classic games for artificial intelligence owing to its enormous search space and the difficulty of evaluating board positions and moves. (Silver et al., 2016) and more recently (Singh et al., 2017) use policy gradient method combined with a neural network representation to beat human experts. Indeed, the Policy Gradient method has become the cornerstone of modern RL and given birth to an entire class of highly efficient policy search algorithms such as TRPO (Schulman et al., 2015), PPO (Schulman et al., 2017), and MADDPG (Lowe et al., 2020).

Despite its excellent empirical performance, not much was known about theoretical guarantees for this approach until recently. There is now a growing body of promising results showing convergence rates for PG algorithms over finite state-action MDPs (Agarwal et al., 2020a; Shani et al., 2020; Bhandari & Russo, 2019; Mei et al., 2020), where the parameterization is over the entire space of state-action pairs, i.e., $\mathbb{R}^{S \times A}$. In particular, (Bhandari & Russo, 2019) show that projected gradient descent does not suffer from spurious local optima on the simplex, (Agarwal et al., 2020a) show that the with softmax parameterization PG converges to the global optima asymptotically. (Shani et al., 2020) show a $\mathcal{O}(1/\sqrt{t})$ convergence rate for mirror descent. (Mei et al., 2020) show that with softmax policy gradient convergence to the global optima occurs at a rate $\mathcal{O}(1/t)$ and at $\mathcal{O}(e^{-t})$ with entropy regularization.

We end this section noting once again that all of the above works concern *proper* learning. Improper learning, on the other hand, has been separately studied in statistical learning theory in the IID setting (Daniely et al., 2014; 2013). In this framework, which is also called *Representation Independent* learning, the learning algorithm is not restricted to output a hypothesis from a given set of hypotheses. We note that improper learning has not been studied in RL literature to the best of our knowledge.

To our knowledge, (Agarwal et al., 2020b) is the only existing work that attempts to frame and solve policy optimization

over an improper class via boosting a given class of controllers. However, the paper is situated in the rather different context of non-stochastic control and assumes perfect knowledge of (i) the memory-boundedness of the MDP, and (ii) the state noise vector in every round, which amounts to essentially knowing the MDP transition dynamics. We work in the stochastic MDP setting and moreover assume no access to the MDP's transition kernel. Further, (Agarwal et al., 2020b) also assumes that all the atomic controllers available to them are *stabilizing* which, when working with an unknown MDP, is a very strong assumption to make. We make no such assumptions on our atomic controller class and, as we show in Sec. 2 and Sec. 6, our algorithms even begin with provably unstable controllers and yet succeed in stabilizing the system.

In summary, the problem that we address concerns finding the best among a *given* class of controllers. None of these need be optimal for the MDP at hand. Moreover, our PG algorithm could very well converge to an improper mixture of these controllers meaning that the output of our algorithms need not be any of the atomic controllers we are provided with. This setting, to the best of our knowledge has not been investigated in the RL literature hitherto.

C. Details of Setup and Modelling of the Cartpole

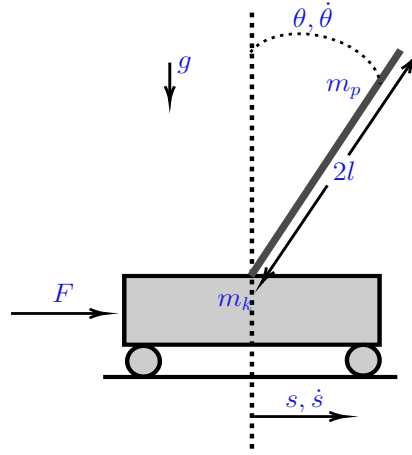


Figure 4: The Cartpole system. The mass of the pendulum is denoted by m_p , that of the cart by m_k , the force used to drive the cart by F , and the distance of the center of mass of the cart from its starting position by s . θ denotes the angle the pendulum makes with the normal and its length is denoted by $2l$. Gravity is denoted by g .

As shown in Fig. 4, it comprises a pendulum whose pivot is mounted on a cart which can be moved in the horizontal direction by applying a force. The objective is to modulate the direction and magnitude of this force F to keep the pendulum from keeling over under the influence of gravity. The state of the system at time t , is given by the 4-tuple $\mathbf{x}(t) := [s, \dot{s}, \theta, \dot{\theta}]$, with $\mathbf{x}(\cdot) = \mathbf{0}$ corresponding to the pendulum being upright and stationary. One of the strategies used to design control policies for this system is by first approximating the dynamics around $\mathbf{x}(\cdot) = \mathbf{0}$ with a linear, quadratic cost model and designing a linear controller for these approximate dynamics. This, after time discretization, The objective now reduces to finding a (potentially randomized) control policy $u \equiv \{u(t), t \geq 0\}$ that solves:

$$\begin{aligned} \inf_u J(\mathbf{x}(0)) &= E_u \sum_{t=0}^{\infty} \mathbf{x}^T(t) Q \mathbf{x}(t) + R u^2(t), \\ \text{s.t. } \mathbf{x}(t+1) &= \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{g}{l\left(\frac{4}{3} - \frac{m_p}{m_p+m_k}\right)} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{g}{l\left(\frac{4}{3} - \frac{m_p}{m_p+m_k}\right)} & 0 \end{pmatrix}}_{A_{open}} \mathbf{x}(t) + \underbrace{\begin{pmatrix} 0 \\ \frac{1}{m_p+m_k} \\ 0 \\ \frac{1}{l\left(\frac{4}{3} - \frac{m_p}{m_p+m_k}\right)} \end{pmatrix}}_{\mathbf{b}} u(t). \end{aligned} \quad (2)$$

Under standard assumptions of controllability and observability, this optimization has a stationary, linear solution $u^*(t) = -\mathbf{K}^\top \mathbf{x}(t)$ (details are available in (??)Chap. 3]bertsekas11dynamic). Moreover, setting $A := A_{open} - \mathbf{b}\mathbf{K}^\top$, it is well know that the dynamics $\mathbf{x}(t+1) = A\mathbf{x}(t)$, $t \geq 0$, are stable.

C.1. Details of simulations settings for the cartpole system

In this section we supply the adjustments we made for specifically for the cartpole experiments. We first mention that we scale down the estimated gradient of the value function returned by the GradEst subroutine (Algorithm 6) (in the cartpole simulation only). The scaling that worked for us is $\frac{10}{\|\nabla V^\pi(\mu)\|}$.

Next, we provide the values of the constants that were described in Sec. C in Table 1.

Parameter	Value
Gravity g	9.8
Mass of pole m_p	0.1
Length of pole l	1
Mass of cart m_k	1
Total mass m_t	1.1

Table 1: Values of the hyperparameters used for the cartpole simulation

D. Stability for Ergodic Parameter Linear Systems (EPLS)

For simplicity and ease of understanding, we connect our current discussion to the cartpole example discussed in Sec. 2.1. Consider a generic (ergodic) control policy that switches across a menu of controllers $\{K_1, \dots, K_N\}$. That is, at any time t , it chooses controller K_i , $i \in [N]$, w.p. p_i , so that the control input at time t is $u(t) = -\mathbf{K}_i^\top \mathbf{x}(t)$ w.p. p_i . Let $A(i) := A_{open} - \mathbf{b}\mathbf{K}_i^\top$. The resulting controlled dynamics are given by

$$\begin{aligned} \mathbf{x}(t+1) &= A(r(t))\mathbf{x}(t) \\ \mathbf{x}(0) &= \mathbf{0}, \end{aligned} \quad (3)$$

where $r(t) = i$ w.p. p_i , IID across time. In the literature, this belongs to a class of systems known as *Ergodic Parameter Linear Systems* (EPLS) (Bolzern et al., 2008), which are said to be *Exponentially Almost Surely Stable* (EAS) if there exists $\rho > 0$ such that for any $\mathbf{x}(0)$,

$$\mathbb{P} \left\{ \omega \in \Omega \left| \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(t, \omega)\| \leq -\rho \right. \right\} = 1. \quad (4)$$

In other words, w.p. 1, the trajectories of the system decay to the origin exponentially fast. The random variable $\lambda(\omega) := \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(t, \omega)\|$ in (4) is called the *Lyapunov Exponent* of the system. For our EPLS,

$$\begin{aligned} \lambda(\omega) &= \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(t, \omega)\| = \limsup_{t \rightarrow \infty} \frac{1}{t} \log \left\| \prod_{s=1}^t A(r(s, \omega)) \mathbf{x}(0) \right\| \\ &\leq \limsup_{t \rightarrow \infty} \frac{1}{t} \log \|\mathbf{x}(0)\| + \limsup_{t \rightarrow \infty} \frac{1}{t} \log \left\| \prod_{s=1}^t A(r(s, \omega)) \right\| \\ &\leq \limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{s=1}^t \log \|A(r(s, \omega))\| \stackrel{(*)}{=} \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{s=1}^t \log \|A(r(s, \omega))\| \\ &\stackrel{(\dagger)}{=} \mathbb{E} \log \|A(r)\| = \sum_{i=1}^N p_i \log \|A(i)\|, \end{aligned} \quad (5)$$

where the equalities (*) and (†) are due to the ergodic law of large numbers. The control policy can now be designed by choosing $\{p_1, \dots, p_N\}$ such that $\lambda(\omega) < -\rho$ for some $\rho > 0$, ensuring exponentially almost sure stability.

E. The Constrained Queuing Example

The system, shown in Fig. 5, comprises two queues fed by independent, stochastic arrival processes $A_i(t), i \in \{1, 2\}, t \in \mathbb{N}$. The length of Queue i , measured at the beginning of time slot t , is denoted by $Q_i(t) \in \mathbb{Z}_+$. A common server serves both queues and can drain at most one packet from the system in a time slot². The server, therefore, needs to decide which of the two queues it intends to serve in a given slot (we assume that once the server chooses to serve a packet, service succeeds with probability 1). The server’s decision is denoted by the vector $\mathbf{D}(t) \in \mathcal{A} := \{[0, 0], [1, 0], [0, 1]\}$, where a “1” denotes service and a “0” denotes lack thereof.

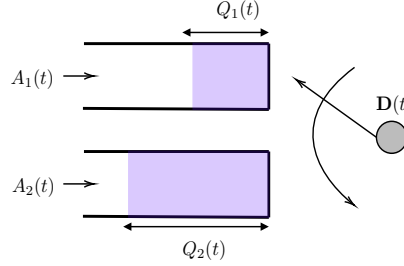


Figure 5: $Q_i(t)$ is the length of Queue i ($i \in \{1, 2\}$) at the beginning of time slot t , $A_i(t)$ is its packet arrival process and $\mathbf{D}(t) \in \{[0, 0], [1, 0], [0, 1]\}$.

For simplicity, we assume that the processes $(A_i(t))_{t=0}^{\infty}$ are both IID Bernoulli, with $\mathbb{E}A_i(t) = \lambda_i$. Note that the arrival rate $\boldsymbol{\lambda} = [\lambda_1, \lambda_2]$ is **unknown** to the learner. Defining $(x)^+ := \max\{0, x\}, \forall x \in \mathbb{R}$, queue length evolution is given by the equations

$$Q_i(t+1) = (Q_i(t) - D_i(t))^+ + A_i(t+1), i \in \{1, 2\}. \quad (6)$$

F. Non-concavity of the Value function

We show here that the value function $V^\pi(\rho)$ is in general non-concave, and hence standard convex optimization techniques for maximization may get stuck in local optima. We note once again that this is *different* from the non-concavity of V^π when the parameterization is over the entire state-action space, i.e., $\mathbb{R}^{S \times A}$.

We show here that for both SoftMax and direct parameterization, the value function is non-concave where, by “direct” parameterization we mean that the controllers K_m are parameterized by weights $\theta_m \in \mathbb{R}$, where $\theta_i \geq 0, \forall i \in [M]$ and $\sum_{i=1}^M \theta_i = 1$. A similar argument holds for softmax parameterization, which we outline in Note F.2.

Lemma F.1. (Non-concavity of Value function) *There is an MDP and a set of controllers, for which the maximization problem of the value function (i.e. (1)) is non-concave for SoftMax parameterization, i.e., $\theta \mapsto V^{\pi_\theta}$ is non-concave.*

²Hence, a *constrained* queuing system.

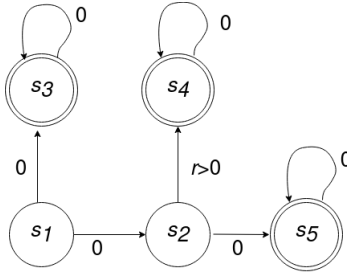


Figure 6: An example of an MDP with controllers as defined in (7) having a non-concave value function. The MDP has $S = 5$ states and $A = 2$ actions. States s_3, s_4 and s_5 are terminal states. The only transition with nonzero reward is $s_2 \rightarrow s_4$.

Proof. Consider the MDP shown in Figure 6 with 5 states, s_1, \dots, s_5 . States s_3, s_4 and s_5 are terminal states. In the figure we also show the allowed transitions and the rewards obtained by those transitions. Let the action set \mathcal{A} consists of only three actions $\{a_1, a_2, a_3\} \equiv \{\text{right}, \text{up}, \text{null}\}$, where 'null' is a dummy action included to accommodate the three terminal states. Let us consider the case when $M = 2$. The two controllers $K_i \in \mathbb{R}^{S \times A}$, $i = 1, 2$ (where each row is probability distribution over \mathcal{A}) are shown below.

$$K_1 = \begin{bmatrix} 1/4 & 3/4 & 0 \\ 3/4 & 1/4 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, K_2 = \begin{bmatrix} 3/4 & 1/4 & 0 \\ 1/4 & 3/4 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}. \quad (7)$$

Let $\theta^{(1)} = (1, 0)^T$ and $\theta^{(2)} = (0, 1)^T$. Let us fix the initial state to be s_1 . Since a nonzero reward is only earned during a $s_2 \rightarrow s_4$ transition, we note for any policy $\pi : \mathcal{A} \rightarrow \mathcal{S}$ that $V^\pi(s_1) = \pi(a_1|s_1)\pi(a_2|s_2)r$. We also have,

$$(K_1 + K_2)/2 = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

We will show that $\frac{1}{2}V^{\pi_{\theta^{(1)}}} + \frac{1}{2}V^{\pi_{\theta^{(2)}}} > V^{\pi_{(\theta^{(1)} + \theta^{(2)})/2}}$.

We observe the following.

$$\begin{aligned} V^{\pi_{\theta^{(1)}}}(s_1) &= V^{K_1}(s_1) = (1/4) \cdot (1/4) \cdot r = r/16. \\ V^{\pi_{\theta^{(2)}}}(s_1) &= V^{K_2}(s_1) = (3/4) \cdot (3/4) \cdot r = 9r/16. \end{aligned}$$

where $V^K(s)$ denotes the value obtained by starting from state s and following a controller matrix K for all time.

Also, on the other hand we have,

$$V^{\pi_{(\theta^{(1)} + \theta^{(2)})/2}} = V^{(K_1 + K_2)/2}(s_1) = (1/2) \cdot (1/2) \cdot r = r/4.$$

Hence we see that,

$$\frac{1}{2}V^{\pi_{\theta^{(1)}}} + \frac{1}{2}V^{\pi_{\theta^{(2)}}} = r/32 + 9r/32 = 10r/32 = 1.25r/4 > r/4 = V^{\pi_{(\theta^{(1)} + \theta^{(2)})/2}}.$$

This shows that $\theta \mapsto V^{\pi_\theta}$ is non-concave, which concludes the proof for direct parameterization.

Remark F.2. For softmax parametrization, we choose the same 2 controllers K_1, K_2 as above. Fix some $\varepsilon \in (0, 1)$ and set $\theta^{(1)} = (\log(1 - \varepsilon), \log \varepsilon)^T$ and $\theta^{(2)} = (\log \varepsilon, \log(1 - \varepsilon))^T$. A similar calculation using softmax projection, and using the fact that $\pi_\theta(a|s) = \sum_{m=1}^M \pi_\theta(m)K_m(s, a)$, shows that under $\theta^{(1)}$ we follow matrix $(1 - \varepsilon)K_1 + \varepsilon K_2$, which yields a Value of $(1/4 + \varepsilon/2)^2 r$. Under $\theta^{(2)}$ we follow matrix $\varepsilon K_1 + (1 - \varepsilon)K_2$, which yields a Value of $(3/4 - \varepsilon/2)^2 r$. On the other hand, $(\theta^{(1)} + \theta^{(2)})/2$ amounts to playing the matrix $(K_1 + K_2)/2$, yielding the a value of $r/4$, as above. One can verify easily that $(1/4 + \varepsilon/2)^2 r + (3/4 - \varepsilon/2)^2 r > 2r/4$. This shows the non-concavity of $\theta \mapsto V^{\pi_\theta}$ under softmax parameterization.

□

G. Example showing that the value function need not be pointwise (over states) monotone over the improper class

Consider the same MDP as in Sec F, however with different base controllers. Let the initial state be s_1 .

The two base controllers $K_i \in \mathbb{R}^{S \times A}$, $i = 1, 2$ (where each row is probability distribution over \mathcal{A}) are shown below.

$$K_1 = \begin{bmatrix} 1/4 & 3/4 & 0 \\ 1/4 & 3/4 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, K_2 = \begin{bmatrix} 3/4 & 1/4 & 0 \\ 3/4 & 1/4 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

Let $\theta^{(1)} = (1, 0)^\top$ and $\theta^{(2)} = (0, 1)^\top$. Let us fix the initial state to be s_1 . Since a nonzero reward is only earned during a $s_2 \rightarrow s_4$ transition, we note for any policy π , that $V^\pi(s_1) = \pi(a_1|s_1)\pi(a_2|s_2)r$ and $V^\pi(s_2) = \pi(a_2|s_2)r$. Note here that the optimal policy of this MDP is *deterministic* with $\pi^*(a_1|s_1) = 1$ and $\pi^*(a_2|s_2) = 1$. The transitions are all deterministic.

However, notice that the optimal policy (with initial state s_1) given K_1 and K_2 is *strict mixture*, because, given any $\theta = [\theta, 1 - \theta]$, $\theta \in [0, 1]$, the value of the policy π_θ is

$$v^{\pi_\theta} = \frac{1}{4}(3 - 2\theta)(1 + 2\theta)r, \quad (9)$$

which is maximized at $\theta = 1/2$. This means that the optimal *non deterministic* policy chooses K_1 and K_2 with probabilities $(1/2, 1/2)$, i.e.,

$$K^* = (K_1 + K_2)/2 = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

We observe the following.

$$\begin{aligned} V^{\pi_{\theta^{(1)}}}(s_1) &= V^{K_1}(s_1) = (1/4) \cdot (3/4) \cdot r = 3r/16. \\ V^{\pi_{\theta^{(2)}}}(s_1) &= V^{K_2}(s_1) = (3/4) \cdot (1/4) \cdot r = 3r/16. \\ V^{\pi_{\theta^{(1)}}}(s_2) &= V^{K_1}(s_2) = (3/4) \cdot r = 3r/4. \end{aligned}$$

On the other hand we have,

$$\begin{aligned} V^{\pi_{(\theta^{(1)} + \theta^{(2)})/2}}(s_1) &= V^{K^*}(s_1) = (1/2) \cdot (1/2) \cdot r = r/4. \\ V^{\pi_{(\theta^{(1)} + \theta^{(2)})/2}}(s_2) &= V^{K^*}(s_2) = (1/2) \cdot r = r/2. \end{aligned}$$

We see that $V^{K^*}(s_1) > \max\{V^{K_1}(s_1), V^{K_2}(s_1)\}$. However, $V^{K^*}(s_2) < V^{K_1}(s_2)$. This implies that playing according to an improved mixture policy (here the optimal given the initial state is s_1) does not necessarily improve the value across *all* states.

H. Proof details for Bandit-over-bandits

In this section we consider the instructive sub-case when $S = 1$, which is also called the Multiarmed Bandit. We provide regret bounds for two cases (1) when the value gradient $\frac{dV^{\pi_{\theta_t}}(\mu)}{d\theta^t}$ (in the gradient update) is available in each round, and (2) when it needs to be estimated.

Note that each controller in this case, is a probability distribution over the A arms of the bandit. We consider the scenario where the agent at each time $t \geq 1$, has to choose a probability distribution K_{m_t} from a set of M probability distributions over actions \mathcal{A} . She then plays an action $a_t \sim K_{m_t}$. This is different from the standard MABs because the learner cannot choose the actions directly, instead chooses from a *given* set of controllers, to play actions. Note the V function has

no argument as $S = 1$. Let $\mu \in [0, 1]^A$ be the mean vector of the arms \mathcal{A} . The value function for any given mixture $\pi \in \mathcal{P}([M])$,

$$\begin{aligned} V^\pi &:= \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t r_t \mid \pi \right] = \sum_{t=0}^{\infty} \gamma^t \mathbb{E} [r_t \mid \pi] \\ &= \sum_{t=0}^{\infty} \gamma^t \sum_{a \in \mathcal{A}} \sum_{m=1}^M \pi(m) K_m(a) \mu_a. \\ &= \frac{1}{1-\gamma} \sum_{m=1}^M \pi_m \mu^\top K_m = \frac{1}{1-\gamma} \sum_{m=1}^M \pi_m \mathbf{v}_m^\mu. \end{aligned} \quad (10)$$

where the interpretation of \mathbf{v}_m^μ is that it is the mean reward one obtains if the controller m is chosen at any round t . Since V^π is linear in π , the maximum is attained at one of the base controllers π^* puts mass 1 on m^* where $m^* := \operatorname{argmax}_{m \in [M]} V^{K_m}$, and V^{K_m} is the value obtained using K_m for all time. In the sequel, we assume $\Delta_i := \mathbf{v}_{m^*}^\mu - \mathbf{v}_i^\mu > 0$.

H.1. Proofs for MABs with perfect gradient knowledge

With access to the exact value gradient at each step, we have the following result, when Softmax PG (Algorithm 1) is applied for the bandits-over-bandits case.

Theorem H.1. *With $\eta = \frac{2(1-\gamma)}{5}$ and with $\theta_m^{(1)} = 1/M$ for all $m \in [M]$, with the availability for true gradient, we have $\forall t \geq 1$,*

$$V^{\pi^*} - V^{\pi_{\theta_t}} \leq \frac{5}{1-\gamma} \frac{M^2}{t}.$$

Also, defining regret for a time horizon of T rounds as

$$\mathcal{R}(T) := \sum_{t=1}^T V^{\pi^*} - V^{\pi_{\theta_t}}, \quad (11)$$

we show as a corollary to Thm. H.4 that,

Corollary H.2.

$$\mathcal{R}(T) \leq \min \left\{ \frac{5M^2}{1-\gamma} \log T, \sqrt{\frac{5}{1-\gamma}} M \sqrt{T} \right\}.$$

Proof. Recall from eq (10), that the value function for any given policy $\pi \in \mathcal{P}([M])$, that is a distribution over the given M controllers (which are itself distributions over actions \mathcal{A}) can be simplified as:

$$V^\pi = \frac{1}{1-\gamma} \sum_{m=1}^M \pi_m \mu^\top K_m = \frac{1}{1-\gamma} \sum_{m=1}^M \pi_m \mathbf{v}_m^\mu$$

where μ here is the (unknown) vector of mean rewards of the arms \mathcal{A} . Here, $\mathbf{v}_m^\mu := \mu^\top K_m$, $i = 1, \dots, M$, represents the mean reward obtained by choosing to play controller K_m , $m \in M$. For ease of notation, we will drop the superscript μ in the proofs of this section. We first show a simplification of the gradient of the value function w.r.t. the parameter θ . Fix a $m \in [M]$,

$$\frac{\partial}{\partial \theta_{m'}} V^{\pi_\theta} = \frac{1}{1-\gamma} \sum_{m=1}^M \frac{\partial}{\partial \theta_m} \pi_\theta(m) \mathbf{v}_m = \frac{1}{1-\gamma} \sum_{m=1}^M \pi_\theta(m') \{1_{mm'} - \pi_\theta(m)\} \mathbf{v}_m. \quad (12)$$

Next we show that V^π is β -smooth. A function $f: \mathbb{R}^M \rightarrow \mathbb{R}$ is β -smooth, if $\forall \theta', \theta \in \mathbb{R}^M$

$$\left| f(\theta') - f(\theta) - \left\langle \frac{d}{d\theta} f(\theta), \theta' - \theta \right\rangle \right| \leq \frac{\beta}{2} \|\theta' - \theta\|_2^2.$$

Let $S := \frac{d^2}{d\theta^2} V^{\pi_\theta}$. This is a matrix of size $M \times M$. Let $1 \leq i, j \leq M$.

$$S_{i,j} = \left(\frac{d}{d\theta} \left(\frac{d}{d\theta} V^{\pi_\theta} \right) \right)_{i,j} \quad (13)$$

$$= \frac{1}{1-\gamma} \frac{d(\pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}))}{d\theta_j} \quad (14)$$

$$= \frac{1}{1-\gamma} \left(\frac{d\pi_\theta(i)}{d\theta_j} (\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) + \pi_\theta(i) \frac{d(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r})}{d\theta_j} \right) \quad (15)$$

$$= \frac{1}{1-\gamma} (\pi_\theta(j)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) - \pi_\theta(i)\pi_\theta(j)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) - \pi_\theta(i)\pi_\theta(j)(\mathbf{r}(j) - \pi_\theta^\top \mathbf{r})). \quad (16)$$

Next, let $y \in \mathbb{R}^M$,

$$\begin{aligned} |y^\top S y| &= \left| \sum_{i=1}^M \sum_{j=1}^M S_{i,j} y(i) y(j) \right| \\ &= \frac{1}{1-\gamma} \left| \sum_{i=1}^M \sum_{j=1}^M (\pi_\theta(j)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) - \pi_\theta(i)\pi_\theta(j)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) - \pi_\theta(i)\pi_\theta(j)(\mathbf{r}(j) - \pi_\theta^\top \mathbf{r})) y(i) y(j) \right| \\ &= \frac{1}{1-\gamma} \left| \sum_{i=1}^M \pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i)^2 - 2 \sum_{i=1}^M \sum_{j=1}^M \pi_\theta(i)\pi_\theta(j)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i) y(j) \right| \\ &= \frac{1}{1-\gamma} \left| \sum_{i=1}^M \pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i)^2 - 2 \sum_{i=1}^M \pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i) \sum_{j=1}^M \pi_\theta(j) y(j) \right| \\ &\leq \frac{1}{1-\gamma} \left| \sum_{i=1}^M \pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i)^2 \right| + \frac{2}{1-\gamma} \left| \sum_{i=1}^M \pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}) y(i) \sum_{j=1}^M \pi_\theta(j) y(j) \right| \\ &\leq \frac{1}{1-\gamma} \|\pi_\theta \odot (\mathbf{r} - \pi_\theta^\top \mathbf{r})\|_\infty \|y \odot y\|_1 + \frac{2}{1-\gamma} \|\pi_\theta \odot (\mathbf{r} - \pi_\theta^\top \mathbf{r})\|_1 \cdot \|y\|_\infty \cdot \|\pi_\theta\|_1 \|y\|_\infty. \end{aligned}$$

The last equality is by the assumption that reward are bounded in $[0,1]$. We observe that,

$$\begin{aligned} \|\pi_\theta \odot (\mathbf{r} - \pi_\theta^\top \mathbf{r})\|_1 &= \sum_{m=1}^M |\pi_\theta(m)(\mathbf{r}(m) - \pi_\theta^\top \mathbf{r})| \\ &= \sum_{m=1}^M \pi_\theta(m) |\mathbf{r}(m) - \pi_\theta^\top \mathbf{r}| \\ &= \max_{i=1, \dots, M} |\mathbf{r}(i) - \pi_\theta^\top \mathbf{r}| \leq 1. \end{aligned}$$

Next, for any $i \in [M]$,

$$\begin{aligned} |\pi_\theta(i)(\mathbf{r}(i) - \pi_\theta^\top \mathbf{r})| &= \left| \pi_\theta(i)\mathbf{r}(i) - \pi_\theta(i)^2 r(i) - \sum_{j \neq i} \pi_\theta(i)\pi_\theta(j)\mathbf{r}(j) \right| \\ &= \pi_\theta(i)(1 - \pi_\theta(i)) + \pi_\theta(i)(1 - \pi_\theta(i)) \leq 2.1/4 = 1/2. \end{aligned}$$

Combining the above two inequalities with the fact that $\|\pi_\theta\|_1 = 1$ and $\|y\|_\infty \leq \|y\|_2$, we get,

$$|y^\top S y| \leq \frac{1}{1-\gamma} \|\pi_\theta \odot (\mathbf{r} - \pi_\theta^\top \mathbf{r})\|_\infty \|y \odot y\|_1 + \frac{2}{1-\gamma} \|\pi_\theta \odot (\mathbf{r} - \pi_\theta^\top \mathbf{r})\|_1 \cdot \|y\|_\infty \cdot \|\pi_\theta\|_1 \|y\|_\infty \leq \frac{1}{1-\gamma} (1/2+2) \|y\|_2^2.$$

Hence V^{π_θ} is β -smooth with $\beta = \frac{5}{2(1-\gamma)}$.

We establish a lower bound on the norm of the gradient of the value function at every step t as below (these type of inequalities are called Łojasiewicz inequalities (Łojasiewicz, 1963))

Lemma H.3. [Lower bound on norm of gradient]

$$\left\| \frac{\partial V^{\pi_\theta}}{\partial \theta} \right\|_2 \geq \pi_{\theta_{m^*}} \left(V^{\pi^*} - V^{\pi_\theta} \right).$$

Proof of Lemma H.3.

Proof. Recall from the simplification of gradient of V^π , i.e., eq (12):

$$\begin{aligned} \frac{\partial}{\partial \theta_m} V^{\pi_\theta} &= \frac{1}{1-\gamma} \sum_{m'=1}^M \pi_\theta(m) \{l_{mm'} - \pi_\theta(m')\} \mathbf{r}'_m \\ &= \frac{1}{1-\gamma} \pi(m) (\mathbf{r}(m) - \pi^\top \mathbf{r}). \end{aligned}$$

Taking norm both sides,

$$\begin{aligned} \left\| \frac{\partial}{\partial \theta} V^{\pi_\theta} \right\| &= \frac{1}{1-\gamma} \sqrt{\sum_{m=1}^M (\pi(m))^2 (\mathbf{r}(m) - \pi^\top \mathbf{r})^2} \\ &\geq \frac{1}{1-\gamma} \sqrt{(\pi(m^*))^2 (\mathbf{r}(m^*) - \pi^\top \mathbf{r})^2} \\ &= \frac{1}{1-\gamma} (\pi(m^*)) (\mathbf{r}(m^*) - \pi^\top \mathbf{r}) \\ &= \frac{1}{1-\gamma} (\pi(m^*)) (\pi^* - \pi)^\top \mathbf{r} \\ &= (\pi(m^*)) [V^{\pi^*} - V^{\pi_\theta}]. \end{aligned}$$

where $\pi^* = e_{m^*}$. □

We will now prove Theorem H.4 and corollary H.2. We restate the result here.

Theorem H.4. With $\eta = \frac{2(1-\gamma)}{5}$ and with $\theta_m^{(1)} = 1/M$ for all $m \in [M]$, with the availability for true gradient, we have $\forall t \geq 1$,

$$V^{\pi^*} - V^{\pi_{\theta_t}} \leq \frac{5}{1-\gamma} \frac{M^2}{t}.$$

Proof. First, note that since V^π is smooth we have:

$$\begin{aligned} V^{\pi_{\theta_t}} - V^{\pi_{\theta_{t+1}}} &\leq - \left\langle \frac{d}{d\theta_t} V^{\pi_{\theta_t}}, \theta_{t+1} - \theta_t \right\rangle + \frac{5}{2(1-\gamma)} \|\theta_{t+1} - \theta_t\|_2^2 \\ &= -\eta \left\| \frac{d}{d\theta_t} V^{\pi_{\theta_t}} \right\|_2^2 + \frac{5}{4(1-\gamma)} \eta^2 \left\| \frac{d}{d\theta_t} V^{\pi_{\theta_t}} \right\|_2^2 \\ &= \left\| \frac{d}{d\theta_t} V^{\pi_{\theta_t}} \right\|_2^2 \left(\frac{5\eta^2}{4(1-\gamma)} - \eta \right) \\ &= - \left(\frac{1-\gamma}{5} \right) \left\| \frac{d}{d\theta_t} V^{\pi_{\theta_t}} \right\|_2^2. \\ &\leq - \left(\frac{1-\gamma}{5} \right) (\pi_{\theta_t}(m^*))^2 [V^{\pi^*} - V^{\pi_{\theta_t}}]^2 \quad \text{Lemma H.3} \\ &\leq - \left(\frac{1-\gamma}{5} \right) \underbrace{\left(\inf_{1 \leq s \leq t} \pi_{\theta_s}(m^*) \right)^2}_{=: c_t} [V^{\pi^*} - V^{\pi_{\theta_t}}]^2. \end{aligned}$$

The first equality is by smoothness, second inequality is by the update equation in algorithm 1.

Next, let $\delta_t := V^{\pi^*} - V^{\pi_{\theta_t}}$. We have,

$$\delta_{t+1} - \delta_t \leq -\frac{(1-\gamma)}{5} c_t^2 \delta_t^2. \quad (17)$$

Claim: $\forall t \geq 1, \delta_t \leq \frac{5}{c_t^2(1-\gamma)} \frac{1}{t}$.

We prove the claim by using induction on $t \geq 1$.

Base case. Since $\delta_t \leq \frac{1}{1-\gamma}$, the claim is true for all $t \leq 5$.

Induction step: Let $\varphi_t := \frac{5}{c_t^2(1-\gamma)}$. Fix a $t \geq 2$, assume $\delta_t \leq \frac{\varphi_t}{t}$.

Let $g : \mathbb{R} \rightarrow \mathbb{R}$ be a function defined as $g(x) = x - \frac{1}{\varphi_t} x^2$. One can verify easily that g is monotonically increasing in $[0, \frac{\varphi_t}{2}]$. Next with equation 19, we have

$$\begin{aligned} \delta_{t+1} &\leq \delta_t - \frac{1}{\varphi_t} \delta_t^2 \\ &= g(\delta_t) \\ &\leq g\left(\frac{\varphi_t}{t}\right) \\ &\leq \frac{\varphi_t}{t} - \frac{\varphi_t}{t^2} \\ &= \varphi_t \left(\frac{1}{t} - \frac{1}{t^2} \right) \\ &\leq \varphi_t \left(\frac{1}{t+1} \right). \end{aligned}$$

This completes the proof of the claim. We will show that $c_t \geq 1/M$ in the next lemma. We first complete the proof of the corollary assuming this.

We fix a $T \geq 1$. Observe that, $\delta_t \leq \frac{5}{(1-\gamma)c_t^2} \frac{1}{t} \leq \frac{5}{(1-\gamma)c_T^2} \frac{1}{t}$.

$$\sum_{t=1}^T V^{\pi^*} - V^{\pi_{\theta_t}} = \frac{1}{1-\gamma} \sum_{t=1}^T (\pi^* - \pi_{\theta_t})^\top \mathbf{r} \leq \frac{5 \log T}{(1-\gamma)c_T^2} + 1.$$

Also we have that,

$$\sum_{t=1}^T V^{\pi^*} - V^{\pi_{\theta_t}} = \sum_{t=1}^T \delta_t \leq \sqrt{T} \sqrt{\sum_{t=1}^T \delta_t^2} \leq \sqrt{T} \sqrt{\sum_{t=1}^T \frac{5}{(1-\gamma)c_T^2} (\delta_t - \delta_{t+1})} \leq \frac{1}{c_T} \sqrt{\frac{5T}{(1-\gamma)}}.$$

We next show that with $\theta_m^{(1)} = 1/M, \forall m$, i.e., uniform initialization, $\inf_{t>1} c_t = 1/M$, which will then complete the proof of Theorem H.4 and of corollary H.2.

Lemma H.5. We have $\inf_{t>1} \pi_{\theta_t}(m^*) > 0$. Furthermore, with uniform initialization of the parameters $\theta_m^{(1)}$, i.e., $1/M, \forall m \in [M]$, we have $\inf_{t>1} \pi_{\theta_t}(m^*) = \frac{1}{M}$.

Proof. We will show that there exists t_0 such that $\inf_{t>1} \pi_{\theta_t}(m^*) = \min_{1 \leq t \leq t_0} \pi_{\theta_t}(m^*)$, where $t_0 = \min \{t : \pi_{\theta_t}(m^*) \geq C\}$.

We define the following sets.

$$\begin{aligned} \mathcal{S}_1 &= \left\{ \theta : \frac{dV^{\pi_{\theta}}}{d\theta_{m^*}} \geq \frac{dV^{\pi_{\theta}}}{d\theta_m}, \forall m \neq m^* \right\} \\ \mathcal{S}_2 &= \{ \theta : \pi_{\theta}(m^*) \geq \pi_{\theta}(m), \forall m \neq m^* \} \\ \mathcal{S}_3 &= \{ \theta : \pi_{\theta}(m^*) \geq C \} \end{aligned}$$

Note that \mathcal{S}_3 depends on the choice of C . Let $C := \frac{M-\Delta}{M+\Delta}$. We claim the following:

Claim 2. (i) $\theta_t \in \mathcal{S}_1 \implies \theta_{t+1} \in \mathcal{S}_1$ and (ii) $\theta_t \in \mathcal{S}_1 \implies \pi_{\theta_{t+1}}(m^*) \geq \pi_{\theta_t}(m^*)$.

Proof of Claim 2. (i) Fix a $m \neq m^*$. We will show that if $\frac{dV^{\pi_\theta}}{d\theta_t(m^*)} \geq \frac{dV^{\pi_\theta}}{d\theta_t(m)}$, then $\frac{dV^{\pi_\theta}}{d\theta_{t+1}(m^*)} \geq \frac{dV^{\pi_\theta}}{d\theta_{t+1}(m)}$. This will prove the first part.

Case (a): $\pi_{\theta_t}(m^*) \geq \pi_{\theta_t}(m)$. This implies, by the softmax property, that $\theta_t(m^*) \geq \theta_t(m)$. After gradient ascent update step we have:

$$\begin{aligned}\theta_{t+1}(m^*) &= \theta_t(m^*) + \eta \frac{dV^{\pi_{\theta_t}}}{d\theta_t(m^*)} \\ &\geq \theta_t(m) + \eta \frac{dV^{\pi_{\theta_t}}}{d\theta_t(m)} \\ &= \theta_{t+1}(m).\end{aligned}$$

This again implies that $\theta_{t+1}(m^*) \geq \theta_{t+1}(m)$. By the definition of derivative of V^{π_θ} w.r.t θ_t (see eq (12)),

$$\begin{aligned}\frac{dV^{\pi_\theta}}{d\theta_{t+1}(m^*)} &= \frac{1}{1-\gamma} \pi_{\theta_{t+1}(m^*)} (\mathbf{r}(m^*) - \pi_{\theta_{t+1}}^T \mathbf{r}) \\ &= \frac{1}{1-\gamma} \pi_{\theta_{t+1}(m)} (\mathbf{r}(m) - \pi_{\theta_{t+1}}^T \mathbf{r}) \\ &= \frac{dV^{\pi_\theta}}{d\theta_{t+1}(m)}.\end{aligned}$$

This implies $\theta_{t+1} \in \mathcal{S}_1$.

Case (b): $\pi_{\theta_t}(m^*) < \pi_{\theta_t}(m)$. We first note the following equivalence:

$$\frac{dV^{\pi_\theta}}{d\theta(m^*)} \geq \frac{dV^{\pi_\theta}}{d\theta(m)} \iff (\mathbf{r}(m^*) - \mathbf{r}(m)) \left(1 - \frac{\pi_\theta(m^*)}{\pi_\theta(m)}\right) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}).$$

which can be simplified as:

$$(\mathbf{r}(m^*) - \mathbf{r}(m)) \left(1 - \frac{\pi_\theta(m^*)}{\pi_\theta(m)}\right) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) = (\mathbf{r}(m^*) - \mathbf{r}(m)) (1 - \exp(\theta_t(m^*) - \theta_t(m))) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}).$$

The above condition can be rearranged as:

$$\mathbf{r}(m^*) - \mathbf{r}(m) \geq (1 - \exp(\theta_t(m^*) - \theta_t(m))) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}).$$

By lemma I.10, we have that $V^{\pi_{\theta_{t+1}}} \geq V^{\pi_{\theta_t}} \implies \pi_{\theta_{t+1}}^T \mathbf{r} \geq \pi_{\theta_t}^T \mathbf{r}$. Hence,

$$0 < \mathbf{r}(m^*) - \pi_{\theta_{t+1}}^T \mathbf{r} \leq \pi_{\theta_t}^T \mathbf{r}.$$

Also, we note:

$$\theta_{t+1}(m^*) - \theta_{t+1}(m) = \theta_t(m^*) + \eta \frac{dV^{\pi_{\theta_t}}}{d\theta_t(m^*)} - \theta_{t+1}(m) - \eta \frac{dV^{\pi_{\theta_t}}}{d\theta_t(m)} \geq \theta_t(m^*) - \theta_t(m).$$

This implies, $1 - \exp(\theta_{t+1}(m^*) - \theta_{t+1}(m)) \leq 1 - \exp(\theta_t(m^*) - \theta_t(m))$.

Next, we observe that by the assumption $\pi_t(m^*) < \pi_t(m)$, we have

$$1 - \exp(\theta_t(m^*) - \theta_t(m)) = 1 - \frac{\pi_t(m^*)}{\pi_t(m)} > 0.$$

Hence we have,

$$\begin{aligned}(1 - \exp(\theta_{t+1}(m^*) - \theta_{t+1}(m))) (\mathbf{r}(m^*) - \pi_{\theta_{t+1}}^T \mathbf{r}) &\leq (1 - \exp(\theta_t(m^*) - \theta_t(m))) (\mathbf{r}(m^*) - \pi_{\theta_t}^T \mathbf{r}) \\ &\leq \mathbf{r}(m^*) - \mathbf{r}(m).\end{aligned}$$

Equivalently,

$$\left(1 - \frac{\pi_{t+1}(m^*)}{\pi_{t+1}(m)}\right) (\mathbf{r}(m^*) - \pi_{t+1}^T \mathbf{r}) \leq \mathbf{r}(m^*) - \mathbf{r}(m).$$

Finishing the proof of the claim 2(i).

(ii) Let $\theta_t \in \mathcal{S}_1$. We observe that:

$$\begin{aligned} \pi_{t+1}(m^*) &= \frac{\exp(\theta_{t+1}(m^*))}{\sum_{m=1}^M \exp(\theta_{t+1}(m))} \\ &= \frac{\exp(\theta_t(m^*) + \eta \frac{dV^{\pi_t}}{d\theta_t(m^*)})}{\sum_{m=1}^M \exp(\theta_t(m) + \eta \frac{dV^{\pi_t}}{d\theta_t(m)})} \\ &\geq \frac{\exp(\theta_t(m^*) + \eta \frac{dV^{\pi_t}}{d\theta_t(m^*)})}{\sum_{m=1}^M \exp(\theta_t(m) + \eta \frac{dV^{\pi_t}}{d\theta_t(m^*)})} \\ &= \frac{\exp(\theta_t(m^*))}{\sum_{m=1}^M \exp(\theta_t(m))} = \pi_t(m^*) \end{aligned}$$

This completes the proof of Claim 2(ii). □

Claim 3. $\mathcal{S}_2 \subset \mathcal{S}_1$ and $\mathcal{S}_3 \subset \mathcal{S}_1$.

Proof. To show that $\mathcal{S}_2 \subset \mathcal{S}_1$, let $\theta \in c\mathcal{S}_2$. We have $\pi_\theta(m^*) \geq \pi_\theta(m), \forall m \neq m^*$.

$$\begin{aligned} \frac{dV^{\pi_\theta}}{d\theta(m^*)} &= \frac{1}{1-\gamma} \pi_\theta(m^*) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) \\ &> \frac{1}{1-\gamma} \pi_\theta(m) (\mathbf{r}(m) - \pi_\theta^T \mathbf{r}) \\ &= \frac{dV^{\pi_\theta}}{d\theta(m)}. \end{aligned}$$

This shows that $\theta \in \mathcal{S}_1$. For showing the second part of the claim, we assume $\theta \in \mathcal{S}_3 \cap \mathcal{S}_2^c$, because if $\theta \in \mathcal{S}_2$, we are done. Let $m \neq m^*$. We have,

$$\begin{aligned} \frac{dV^{\pi_\theta}}{d\theta(m^*)} - \frac{dV^{\pi_\theta}}{d\theta(m)} &= \frac{1}{1-\gamma} (\pi_\theta(m^*) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) - \pi_\theta(m) (\mathbf{r}(m) - \pi_\theta^T \mathbf{r})) \\ &= \frac{1}{1-\gamma} \left(2\pi_\theta(m^*) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) + \sum_{i \neq m^*, m}^M \pi_\theta(i) (\mathbf{r}(i) - \pi_\theta^T \mathbf{r}) \right) \\ &= \frac{1}{1-\gamma} \left(\left(2\pi_\theta(m^*) + \sum_{i \neq m^*, m}^M \pi_\theta(i) \right) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) - \sum_{i \neq m^*, m}^M \pi_\theta(i) (\mathbf{r}(m^*) - \mathbf{r}(i)) \right) \\ &\geq \frac{1}{1-\gamma} \left(\left(2\pi_\theta(m^*) + \sum_{i \neq m^*, m}^M \pi_\theta(i) \right) (\mathbf{r}(m^*) - \pi_\theta^T \mathbf{r}) - \sum_{i \neq m^*, m}^M \pi_\theta(i) \right) \\ &\geq \frac{1}{1-\gamma} \left(\left(2\pi_\theta(m^*) + \sum_{i \neq m^*, m}^M \pi_\theta(i) \right) \frac{\Delta}{M} - \sum_{i \neq m^*, m}^M \pi_\theta(i) \right). \end{aligned}$$

Observe that, $\sum_{i \neq m^*, m}^M \pi_\theta(i) = 1 - \pi(m^*) - \pi(m)$. Using this and rearranging we get,

$$\frac{dV^{\pi_\theta}}{d\theta(m^*)} - \frac{dV^{\pi_\theta}}{d\theta(m)} \geq \frac{1}{1-\gamma} \left(\pi(m^*) \left(1 + \frac{\Delta}{M} \right) - \left(1 - \frac{\Delta}{M} \right) + \pi(m) \left(1 - \frac{\Delta}{M} \right) \right) \geq \frac{1}{1-\gamma} \pi(m) \left(1 - \frac{\Delta}{M} \right) \geq 0.$$

The last inequality follows because $\theta \in \mathcal{S}_3$ and the choice of C . This completes the proof of Claim 3. \square

Claim 4. There exists a finite t_0 , such that $\theta_{t_0} \in \mathcal{S}_3$.

Proof. The proof of this claim relies on the asymptotic convergence result of (Agarwal et al., 2020a). We note that their convergence result hold for our choice of $\eta = \frac{2(1-\gamma)}{5}$. As noted in (Mei et al., 2020), the choice of η is used to justify the gradient ascent lemma I.10. Hence we have $\pi_{\theta_t} \rightarrow 1$ as $t \rightarrow \infty$. Therefore, there exists a finite t_0 such that $\pi_{\theta_{t_0}}(m^*) \geq C$ and hence $\theta_{t_0} \in \mathcal{S}_3$. \square

This completes the proof that there exists a t_0 such that $\inf_{t>1} \pi_{\theta_t}(m^*) = \inf_{1 \leq t \leq t_0} \pi_{\theta_t}(m^*)$, since once the $\theta_t \in \mathcal{S}_3$, by Claim 3, $\theta_t \in \mathcal{S}_1$. Further, by Claim 2, $\forall t \geq t_0$, $\theta_t \in \mathcal{S}_1$ and $\pi_{\theta_t}(m^*)$ is non-decreasing after t_0 . \square

With uniform initialization $\theta_1(m^*) = \frac{1}{M} \geq \theta_1(m)$, for all $m \neq m^*$. Hence, $\pi_{\theta_1}(m^*) \geq \pi_{\theta_1}(m)$ for all $m \neq m^*$. This implies $\theta_1 \in \mathcal{S}_2$, which implies $\theta_1 \in \mathcal{S}_1$. As established in Claim 2, \mathcal{S}_1 remains invariant under gradient ascent updates, implying $t_0 = 1$. Hence we have that $\inf_{t>1} \pi_{\theta_t}(m^*) = \pi_{\theta_1}(m^*) = 1/M$, completing the proof of Theorem H.4 and corollary H.2. \square

H.2. Proofs for MABs with noisy gradients

When value gradients are unavailable, we follow a direct policy gradient algorithm instead of softmax projection. The full pseudo-code is provided here in Algorithm 4. At each round $t \geq 1$, the learning rate for η is chosen asynchronously for each controller m , to be $\alpha \pi_t(m)^2$, to ensure that we remain inside the simplex, for some $\alpha \in (0, 1)$. To justify its name as a policy gradient algorithm, observe that in order to minimize regret, we need to solve the following optimization problem:

$$\min_{\pi \in \mathcal{P}(\{M\})} \sum_{m=1}^M \pi(m) (\tau_\mu(m^*) - \tau_\mu(m)).$$

A direct gradient with respect to the parameters $\pi(m)$ gives us a rule for the policy gradient algorithm. The other changes in the update step (eq 18), stem from the fact that true means of the arms are unavailable and importance sampling.

We have the following result.

Theorem H.6. *With value of α chosen to be less than $\frac{\Delta_{min}}{\tau_{m^*}^\mu - \Delta_{min}}$, (π_t) is a Markov process, with $\pi_t(m^*) \rightarrow 1$ as $t \rightarrow \infty$, a.s. Further the regret till any time T is bounded as*

$$\mathcal{R}(T) \leq \frac{1}{1-\gamma} \sum_{m \neq m^*} \frac{\Delta_m}{\alpha \Delta_{min}^2} \log T + C,$$

where $C := \frac{1}{1-\gamma} \sum_{t>1} \mathbb{P} \{ \pi_t(m^*(t)) \leq \frac{1}{2} \} < \infty$.

We make couple of remarks before providing the full proof of Theorem H.6.

Remark H.7. The ‘‘cost’’ of not knowing the true gradient seems to cause the dependence on Δ_{min} in the regret, as is not the case when true gradient is available (see Theorem H.4 and Corollary H.2). The dependence on Δ_{min} as is well known from the work of (Lai & Robbins, 1985), is unavoidable.

Remark H.8. The dependence of α on Δ_{min} can be removed by a more sophisticated choice of learning rate, at the cost of an extra $\log T$ dependence on regret (Denisov & Walton, 2020).

Algorithm 4 Projection-free Policy Gradient (for MABs)

Input: learning rate $\eta \in (0, 1)$

Initialize each $\pi_1(m) = \frac{1}{M}$, for all $m \in [M]$.

for $t = 1$ **to** T **do**

$m_*(t) \leftarrow \operatorname{argmax}_{m \in [M]} \pi_t(m)$

Choose controller $m_t \sim \pi_t$.

Play action $a_t \sim K_{m_t}$.

Receive reward R_{m_t} by pulling arm a_t .

Update $\forall m \in [M], m \neq m_*(t)$:

$$\pi_{t+1}(m) = \pi_t(m) + \eta \left(\frac{R_m \mathbb{1}_m}{\pi_t(m)} - \frac{R_{m_*(t)} \mathbb{1}_{m_*(t)}}{\pi_t(m_*(t))} \right) \quad (18)$$

Set $\pi_{t+1}(m_*(t)) = 1 - \sum_{m \neq m_*(t)} \pi_{t+1}(m)$.

end for

Proof. The proof is an extension of that of Theorem 1 of (Denisov & Walton, 2020) for the setting that we have. The proof is divided into three main parts. In the first part we show that the recurrence time of the process $\{\pi_t(m^*)\}_{t>1}$ is almost surely finite. Next we bound the expected value of the time taken by the process $\pi_t(m^*)$ to reach 1. Finally we show that almost surely, $\lim_{t \rightarrow \infty} \pi_t(m^*) \rightarrow 1$, in other words the process $\{\pi_t(m^*)\}_{t>1}$ is transient. We use all these facts to show a regret bound.

Recall $m_*(t) := \operatorname{argmax}_{m \in [M]} \pi_t(m)$. We start by defining the following quantity which will be useful for the analysis of algorithm 4.

Let $\tau := \min \{t \geq 1 : \pi_t(m^*) > \frac{1}{2}\}$.

Next, let $\mathcal{S} := \{\pi \in \mathcal{P}([M]) : \frac{1-\alpha}{2} \leq \pi(m^*) < \frac{1}{2}\}$.

In addition, we define for any $a \in \mathbb{R}$, $\mathcal{S}_a := \{\pi \in \mathcal{P}([M]) : \frac{1-\alpha}{a} \leq \pi(m^*) < \frac{1}{x}\}$. Observe that if $\pi_1(m^*) \geq 1/a$ and $\pi_2(m^*) < 1/a$ then $\pi_1 \in \mathcal{S}_a$. This fact follows just by the update step of the algorithm 4, and choosing $\eta = \alpha \pi_t(m)$ for every $m \neq m^*$.

Lemma H.9. For $\alpha > 0$ such that $\alpha < \frac{\Delta_{min}}{\mathfrak{r}(m^*) - \Delta_{min}}$, we have that

$$\sup_{\pi \in \mathcal{S}} \mathbb{E} [\tau \mid \pi_1 = \pi] < \infty.$$

Proof. The proof here is for completeness. We first make note of the following useful result: For a sequence of positive real numbers $\{a_n\}_{n>1}$ such that the following condition is met:

$$a(n+1) \leq a(n) - b \cdot a(n)^2,$$

for some $b > 0$, the following is always true:

$$a_n \leq \frac{a_1}{1 + bt}.$$

This inequality follows by rearranging and observing the a_n is a non-increasing sequence. A complete proof can be found in eg. ((Denisov & Walton, 2020), Appendix A.1). Returning to the proof of lemma, we proceed by showing that the sequence $1/\pi_t(m^*) - ct$ is a supermartingale for some $c > 0$. Let $\Delta_{min} := \Delta$ for ease of notation. Note that if the condition on α holds then there exists an $\varepsilon > 0$, such that $(1 + \varepsilon)(1 + \alpha) < \mathfrak{r}^*/(\mathfrak{r}^* - \Delta)$, where $\mathfrak{r}^* := \mathfrak{r}(m^*)$. We choose c to be

$$c := \alpha \cdot \frac{\mathfrak{r}^*}{1 + \alpha} - \alpha(\mathfrak{r}^* - \Delta)(1 + \varepsilon) > 0.$$

Next, let x to be greater than M and satisfying:

$$\frac{x}{x - \alpha M} \leq 1 + \varepsilon.$$

Let $\xi_x := \min\{t \geq 1 : \pi_t(m^*) > 1/x\}$. Since for $t = 1, \dots, \xi_x - 1$, $m_*(t) \neq m^*$, we have $\pi_{t+1}(m^*) = (1 + \alpha)\pi_t(m^*)$ w.p. $\pi_t(m^*)\mathbf{r}^*$ and $\pi_{t+1}(m^*) = \pi_t(m^*) + \alpha\pi_t(m^*)^2/\pi_t(m_*(t))^2$ w.p. $\pi_t(m_*(t))\mathbf{r}_*(t)$, where $\mathbf{r}_*(t) := \mathbf{r}(m_*(t))$.

Let $y(t) := 1/\pi_t(m^*)$, then we observe by a short calculation that,

$$y(t+1) = \begin{cases} y(t) - \frac{\alpha}{1+\alpha}y(t), & w.p. \frac{\mathbf{r}^*}{y(t)} \\ y(t) + \alpha \frac{y(t)}{\pi_t(m_*(t))y(t) - \alpha}, & w.p. \pi_t(m_*(t))\mathbf{r}_*(t) \\ y(t) & otherwise. \end{cases}$$

We see that,

$$\begin{aligned} & \mathbb{E}[y(t+1) \mid H(t)] - y(t) \\ &= \frac{\mathbf{r}^*}{y(t)} \cdot \left(y(t) - \frac{\alpha}{1+\alpha}y(t) \right) + \pi_t(m_*(t))\mathbf{r}_*(t) \cdot \left(y(t) + \alpha \frac{y(t)}{\pi_t(m_*(t))y(t) - \alpha} \right) - y(t) \left(\frac{\mathbf{r}^*}{y(t)} + \pi_t(m_*(t))\mathbf{r}_*(t) \right) \\ &\leq \alpha(\mathbf{r}^* - \Delta)(1 + \varepsilon) - \frac{\alpha\mathbf{r}^*}{1 + \alpha} = -c. \end{aligned}$$

The inequality holds because $\mathbf{r}_*(t) \leq \mathbf{r}^*\Delta$ and that $\pi_t(m_*) > 1/M$. By the Optional Stopping Theorem (Durrett, 2011),

$$-c\mathbb{E}[\xi_x \wedge t] \geq \mathbb{E}[y(\xi_x \wedge t) - \mathbb{E}[y(1)]] \geq -\frac{x}{1 - \alpha}.$$

The final inequality holds because $\pi_1(m^*) \geq \frac{1-\alpha}{x}$.

Next, applying the monotone convergence theorem gives $\mathbb{E}[\xi_x] \leq \frac{x}{c(1-\alpha)}$. Finally to show the result of lemma H.9, we refer the reader to (Appendix A.2, (Denisov & Walton, 2020)), which follow from standard Markov chain arguments. \square

Next we define an embedded Markov Chain $\{p(s), s \in \mathbb{Z}_+\}$ as follows. First let $\sigma(k) := \min\{t \geq \tau(k) : \pi_t(m^*) < \frac{1}{2}\}$ and $\tau(k) := \min\{t \geq \sigma(k-1) : \pi_t(m^*) \geq \frac{1}{2}\}$. Note that within the region $[\tau(k), \sigma(k))$, $\pi_t(m^*) \geq 1/2$ and in $[\sigma(k), \tau(k+1))$, $\pi_t(m^*) < 1/2$. We next analyze the rate at which $\pi_t(m^*)$ approaches 1. Define

$$\begin{aligned} p(s) &:= \pi_{t_s}(m^*) \text{ where} & t_s &= s + \sum_{i=0}^k (\tau(i+1) - \sigma(i)) \\ &\text{for} & s &\in \left[\sum_{i=0}^k (\sigma(i) - \tau(i)), \sum_{i=0}^{k+1} (\sigma(i) - \tau(i)) \right) \end{aligned}$$

Also let,

$$\sigma_s := \min\{t > 0 : \pi_{t+t_s}(m^*) > 1/2\}$$

and,

$$\tau_s := \min\{t > \sigma_s : \pi_{t+t_s}(m^*) \leq 1/2\}$$

Lemma H.10. *The process $\{p(s)\}_{s>1}$, is a submartingale. Further, $p(s) \rightarrow 1$, as $s \rightarrow \infty$. Finally,*

$$\mathbb{E}[p(s)] \geq 1 - \frac{1}{1 + \alpha \frac{\Delta^2}{\left(\sum_{m' \neq m^*} \Delta_{m'}\right) s}}.$$

Proof. We first observe that,

$$p(s+1) = \begin{cases} \pi_{t_s+1}(m^*) & \text{if } \pi_{t_s+1}(m^*) \geq 1/2 \\ \pi_{t_s+\tau_s}(m^*) & \text{if } \pi_{t_s+1}(m^*) < 1/2 \end{cases}$$

Since $\pi_{t_s+\tau_s}(m^*) \geq 1/2$, we have that,

$$p(s+1) \geq \pi_{t_s+1}(m^*) \text{ and } p(s) = \pi_{t_s}(m^*).$$

Since at times t_s , $\pi_{t_s}(m^*) > 1/2$, we know that m^* is the leading arm. Thus by the update step, for all $m \neq m^*$,

$$\pi_{t_s+1}(m) = \pi_{t_s}(m) + \alpha \pi_{t_s}(m)^2 \left[\frac{I_m R_m(t_s)}{\pi_{t_s}(m)} - \frac{I_{m^*} R_{m^*}(t_s)}{\pi_{t_s}(m^*)} \right].$$

Taking expectations both sides,

$$\mathbb{E} [\pi_{t_s+1}(m) \mid H(t_s)] - \pi_{t_s}(m) = \alpha \pi_{t_s}(m)^2 (\mathbf{r}_m - \mathbf{r}_{m^*}) = -\alpha \Delta_m \pi_{t_s}(m)^2.$$

Summing over all $m \neq m^*$:

$$-\mathbb{E} [\pi_{t_s+1}(m^*) \mid H(t_s)] + \pi_{t_s}(m^*) = -\alpha \sum_{m \neq m^*} \Delta_m \pi_{t_s}(m)^2.$$

By Jensen's inequality,

$$\begin{aligned} \sum_{m \neq m^*} \Delta_m \pi_{t_s}(m)^2 &= \left(\sum_{m' \neq m^*} \Delta_{m'} \right) \sum_{m \neq m^*} \frac{\Delta_m}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)} \pi_{t_s}(m)^2 \\ &\geq \left(\sum_{m' \neq m^*} \Delta_{m'} \right) \left(\sum_{m \neq m^*} \frac{\Delta_m \pi_{t_s}(m)}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)} \right)^2 \\ &\geq \left(\sum_{m' \neq m^*} \Delta_{m'} \right) \frac{\Delta^2 \left(\sum_{m \neq m^*} \pi_{t_s}(m) \right)^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)^2} \\ &= \frac{\Delta^2 (1 - \pi_{t_s}(m^*))^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}. \end{aligned}$$

Hence we get,

$$p(s) - \mathbb{E} [p(s+1) \mid H(t_s)] \leq -\alpha \frac{\Delta^2 (1 - p(s))^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)} \implies \mathbb{E} [p(s+1) \mid H(t_s)] \geq p(s) + \alpha \frac{\Delta^2 (1 - p(s))^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}.$$

This implies immediately that $\{p(s)\}_{s>1}$ is a submartingale.

Since, $\{p(s)\}$ is non-negative and bounded by 1, by Martingale Convergence Theorem, $\lim_{s \rightarrow \infty} p(s)$ exists. We will now show that the limit is 1. Clearly, it is sufficient to show that $\limsup_{s \rightarrow \infty} p(s) = 1$. For $a > 2$, let

$$\varphi_a := \min \left\{ s \geq 1 : p(s) \geq \frac{a-1}{a} \right\}.$$

As is shown in (Denisov & Walton, 2020), it is sufficient to show $\varphi_a < \infty$, with probability 1, because then one can define a sequence of stopping times for increasing a , each finite w.p. 1. which implies that $p(s) \rightarrow 1$. By the previous display, we have

$$\mathbb{E} [p(s+1) \mid H(t_s)] - p(s) \geq \alpha \frac{\Delta^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right) a^2}$$

as long as $p(s) \leq \frac{\alpha-1}{\alpha}$. Hence by applying Optional Stopping Theorem and rearranging we get,

$$\mathbb{E} [\varphi_a] \leq \lim_{s \rightarrow \infty} \mathbb{E} [\varphi_a \wedge s] \leq \frac{\left(\sum_{m' \neq m^*} \Delta_{m'} \right) \alpha^2}{\alpha \Delta} (1 - \mathbb{E} [p(1)]) < \infty.$$

Since φ_a is a non-negative random variable with finite expectation, $\varphi_a < \infty a.s.$ Let $q(s) = 1 - p(s)$. We have :

$$\mathbb{E} [q(s+1)] - \mathbb{E} [q(s)] \leq -\alpha \frac{\Delta^2 (q(s))^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}.$$

By the useful result H.2, we get,

$$\mathbb{E} [q(s)] \leq \frac{\mathbb{E} [q(1)]}{1 + \alpha \frac{\Delta^2 \mathbb{E} [q(1)]}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right) s}} \leq \frac{1}{1 + \alpha \frac{\Delta^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right) s}}.$$

This completes the proof of the lemma. \square

Finally we provide a lemma to tie the results above. We refer (Appendix A.5 (Denisov & Walton, 2020)) for the proof of this lemma.

Lemma H.11.

$$\sum_{t>1} \mathbb{P} [\pi_t(m^*) < 1/2] < \infty.$$

Also, with probability 1, $\pi_t(m^*) \rightarrow 1$, as $t \rightarrow \infty$.

Proof of regret bound: Since $\mathbf{r}^* - \mathbf{r}(m) \leq 1$, we have by the definition of regret (see eq 11)

$$\mathcal{R}(T) = \mathbb{E} \left[\frac{1}{1-\gamma} \sum_{t=1}^T \left(\sum_{m=1}^M \pi^*(m) \mathbf{r}_m - \pi_t(m) \mathbf{r}_m \right) \right].$$

Here we recall that $\pi^* = e_{m^*}$, we have:

$$\begin{aligned} \mathcal{R}(T) &= \frac{1}{1-\gamma} \mathbb{E} \left[\sum_{t=1}^T \left(\sum_{m=1}^M (\pi^*(m) \mathbf{r}_m - \pi_t(m) \mathbf{r}_m) \right) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\sum_{m=1}^M \left(\sum_{t=1}^T (\pi^*(m) \mathbf{r}_m - \pi_t(m) \mathbf{r}_m) \right) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\sum_{t=1}^T \left(\mathbf{r}^* - \sum_{m=1}^M \pi_t(m) \mathbf{r}_m \right) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\left(\sum_{t=1}^T \mathbf{r}^* - \sum_{t=1}^T \sum_{m=1}^M \pi_t(m) \mathbf{r}_m \right) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\left(\sum_{t=1}^T \mathbf{r}^* (1 - \pi_t(m^*)) - \sum_{t=1}^T \sum_{m \neq m^*} \pi_t(m) \mathbf{r}_m \right) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\left(\sum_{t=1}^T \sum_{m \neq m^*} \mathbf{r}^* \pi_t(m) - \sum_{t=1}^T \sum_{m \neq m^*} \pi_t(m) \mathbf{r}_m \right) \right] \\ &= \frac{1}{1-\gamma} \sum_{m \neq m^*} (\mathbf{r}^* - \mathbf{r}_m) \mathbb{E} \left[\sum_{t=1}^T \pi_t(m) \right]. \end{aligned}$$

Hence we have,

$$\begin{aligned}\mathcal{R}(T) &= \frac{1}{1-\gamma} \sum_{m \neq m^*} (\mathbf{r}^* - \mathbf{r}_m) \mathbb{E} \left[\sum_{t=1}^T \pi_t(m) \right] \\ &\leq \frac{1}{1-\gamma} \sum_{m \neq m^*} \mathbb{E} \left[\sum_{t=1}^T \pi_t(m) \right] \\ &= \frac{1}{1-\gamma} \mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \right]\end{aligned}$$

We analyze the following term:

$$\begin{aligned}\mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \right] &= \mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \mathbb{1}_{\{\pi_t(m^*) \geq 1/2\}} \right] + \mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \mathbb{1}_{\{\pi_t(m^*) < 1/2\}} \right] \\ &= \mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \mathbb{1}_{\{\pi_t(m^*) \geq 1/2\}} \right] + C_1.\end{aligned}$$

where, $C_1 := \sum_{t=1}^{\infty} \mathbb{P}[\pi_t(m^*) < 1/2] < \infty$ by Lemma H.11. Next we observe that,

$$\begin{aligned}\mathbb{E} \left[\sum_{t=1}^T (1 - \pi_t(m^*)) \mathbb{1}_{\{\pi_t(m^*) \geq 1/2\}} \right] &= \mathbb{E} \left[\sum_{s=1}^T q(s) \mathbb{1}_{\{\pi_t(m^*) \geq 1/2\}} \right] \leq \mathbb{E} \left[\sum_{s=1}^T q(s) \right] \\ &= \sum_{t=1}^T \frac{1}{1 + \alpha \frac{\Delta^2}{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)^s}} \leq \sum_{t=1}^T \frac{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}{\alpha \Delta^2 s} \\ &\leq \frac{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}{\alpha \Delta^2} \log T.\end{aligned}$$

Putting things together, we get,

$$\begin{aligned}\mathcal{R}(T) &\leq \frac{1}{1-\gamma} \left(\frac{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}{\alpha \Delta^2} \log T + C_1 \right) \\ &= \frac{1}{1-\gamma} \left(\frac{\left(\sum_{m' \neq m^*} \Delta_{m'} \right)}{\alpha \Delta^2} \log T \right) + C.\end{aligned}$$

This completes the proof of Theorem H.6.

□

I. Proofs for MDPs

First we recall the policy gradient theorem.

Theorem I.1 (Policy Gradient Theorem (Sutton et al., 2000)).

$$\frac{\partial}{\partial \theta} V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \sum_{a \in \mathcal{A}} \frac{\partial \pi_\theta(a|s)}{\partial \theta} Q^{\pi_\theta}(s, a).$$

Let $s \in \mathcal{S}$ and $m \in [M]$. Let $\tilde{Q}^{\pi_\theta}(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) Q^{\pi_\theta}(s, a)$. Also let $\tilde{A}(s, m) := \tilde{Q}(s, m) - V(s)$.

Lemma I.2 (Gradient Simplification). *The softmax policy gradient with respect to the parameter $\theta \in \mathbb{R}^M$ is $\frac{\partial}{\partial \theta} V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_\theta(m) \tilde{A}(s, m)$, where $\tilde{A}(s, m) := \tilde{Q}(s, m) - V(s)$ and $\tilde{Q}(s, m) := \sum_{a \in \mathcal{A}} K_m(s, a) Q^{\pi_\theta}(s, a)$, and $d_\mu^{\pi_\theta}(\cdot)$ is the discounted state visitation measure starting with an initial distribution μ and following policy π_θ .*

The interpretation of $\tilde{A}(s, m)$ is the advantage of following controller m at state s and then following the policy π_θ for all time versus following π_θ always. As mentioned in section 4, we proceed by proving smoothness of the V^π function over the space \mathbb{R}^M .

Proof. From the policy gradient theorem I.1, we have:

$$\begin{aligned} \frac{\partial}{\partial \theta_{m'}} V^{\pi_\theta}(\mu) &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \sum_{a \in \mathcal{A}} \frac{\partial \pi_{\theta_{m'}}(a|s)}{\partial \theta} Q^{\pi_\theta}(s, a) \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \sum_{a \in \mathcal{A}} \frac{\partial}{\partial \theta_{m'}} \left(\sum_{m=1}^M \pi_\theta(m) K_m(s, a) \right) Q^{\pi_\theta}(s, a) \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \sum_{m=1}^M \sum_{a \in \mathcal{A}} \left(\frac{\partial}{\partial \theta_{m'}} \pi_\theta(m) \right) K_m(s, a) Q(s, a) \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \sum_{a \in \mathcal{A}} \pi_{m'} \left(K_{m'}(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right) Q(s, a) \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_{m'} \sum_{a \in \mathcal{A}} \left(K_{m'}(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right) Q(s, a) \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_{m'} \left[\sum_{a \in \mathcal{A}} K_{m'}(s, a) Q(s, a) - \sum_{a \in \mathcal{A}} \sum_{m=1}^M \pi_m K_m(s, a) Q(s, a) \right] \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_{m'} \left[\tilde{Q}(s, m') - V(s) \right] \\ &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_{m'} \tilde{A}^{\pi_\theta}(s, m'). \end{aligned}$$

□

Lemma I.3. $V^{\pi_\theta}(\mu)$ is $\frac{7\gamma^2+4\gamma+5}{2(1-\gamma)^2}$ -smooth.

Proof. The proof uses ideas from (Agarwal et al., 2020a) and (Mei et al., 2020). Let $\theta_\alpha = \theta + \alpha u$, where $u \in \mathbb{R}^M$, $\alpha \in \mathbb{R}$. For any $s \in \mathcal{S}$,

$$\sum_a \left| \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \alpha} \Big|_{\alpha=0} \right| = \sum_a \left| \left\langle \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \theta_\alpha} \Big|_{\alpha=0}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| = \sum_a \left| \left\langle \frac{\partial \pi_{\theta_\alpha}(a|s)}{\partial \theta_\alpha} \Big|_{\alpha=0}, u \right\rangle \right|$$

$$\begin{aligned}
 &= \sum_a \left| \sum_{m''=1}^M \sum_{m=1}^M \pi_{\theta_{m''}} (l_{mm''} - \pi_{\theta_m}) K_m(s, a) u(m'') \right| \\
 &= \sum_a \left| \sum_{m''=1}^M \pi_{\theta_{m''}} \left(K_{m''}(s, a) u(m'') - \sum_{m=1}^M K_m(s, a) u(m'') \right) \right| \\
 &\leq \sum_a \sum_{m''=1}^M \pi_{\theta_{m''}} K_{m''}(s, a) |u(m'')| + \sum_a \sum_{m''=1}^M \sum_{m=1}^M \pi_{\theta_{m''}} \pi_{\theta_m} K_m(s, a) |u(m'')| \\
 &= \sum_{m''=1}^M \pi_{\theta_{m''}} |u(m'')| \underbrace{\sum_a K_{m''}(s, a)}_{=1} + \sum_{m''=1}^M \sum_{m=1}^M \pi_{\theta_{m''}} \pi_{\theta_m} |u(m'')| \underbrace{\sum_a K_m(s, a)}_{=1} \\
 &= \sum_{m''=1}^M \pi_{\theta_{m''}} |u(m'')| + \sum_{m''=1}^M \sum_{m=1}^M \pi_{\theta_{m''}} \pi_{\theta_m} |u(m'')| \\
 &= 2 \sum_{m''=1}^M \pi_{\theta_{m''}} |u(m'')| \leq 2 \|u\|_2.
 \end{aligned}$$

Next we bound the second derivative.

$$\sum_a \left| \frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \alpha^2} \Big|_{\alpha=0} \right| = \sum_a \left| \left\langle \frac{\partial}{\partial \theta_\alpha} \frac{\partial \pi_{\theta_\alpha}(a | s)}{\partial \alpha} \Big|_{\alpha=0}, u \right\rangle \right| = \sum_a \left| \left\langle \frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \alpha^2} \Big|_{\alpha=0}, u, u \right\rangle \right|.$$

Let $H^{a,\theta} := \frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \theta^2} \in \mathbb{R}^{M \times M}$. We have,

$$\begin{aligned}
 H_{i,j}^{a,\theta} &= \frac{\partial}{\partial \theta_j} \left(\sum_{m=1}^M \pi_{\theta_i} (l_{mi} - \pi_{\theta_m}) K_m(s, a) \right) \\
 &= \frac{\partial}{\partial \theta_j} \left(\pi_{\theta_i} K_i(s, a) - \sum_{m=1}^M \pi_{\theta_i} \pi_{\theta_m} K_m(s, a) \right) \\
 &= \pi_{\theta_j} (l_{ij} - \pi_{\theta_i}) K_i(s, a) - \sum_{m=1}^M K_m(s, a) \frac{\partial \pi_{\theta_i} \pi_{\theta_m}}{\partial \theta_j} \\
 &= \pi_j (l_{ij} - \pi_i) K_i(s, a) - \sum_{m=1}^M K_m(s, a) (\pi_j (l_{ij} - \pi_i) \pi_m + \pi_i \pi_j (l_{mj} - \pi_m)) \\
 &= \pi_j \left((l_{ij} - \pi_i) K_i(s, a) - \sum_{m=1}^M \pi_m (l_{ij} - \pi_i) K_m(s, a) - \sum_{m=1}^M \pi_i (l_{mj} - \pi_m) K_m(s, a) \right).
 \end{aligned}$$

Plugging this into the second derivative, we get,

$$\begin{aligned}
 & \left| \left\langle \frac{\partial^2}{\partial \theta^2} \pi_\theta(a|s)u, u \right\rangle \right| \\
 &= \left| \sum_{j=1}^M \sum_{i=1}^M H_{i,j}^{a,\theta} u_i u_j \right| \\
 &= \left| \sum_{j=1}^M \sum_{i=1}^M \pi_j \left((l_{ij} - \pi_i) K_i(s, a) - \sum_{m=1}^M \pi_m (l_{ij} - \pi_i) K_m(s, a) - \sum_{m=1}^M \pi_i (l_{mj} - \pi_m) K_m(s, a) \right) u_i u_j \right| \\
 &= \left| \sum_{i=1}^M \pi_i K_i(s, a) u_i^2 - \sum_{i=1}^M \sum_{j=1}^M \pi_i \pi_j K_i(s, a) u_i u_j - \sum_{i=1}^M \sum_{m=1}^M \pi_i \pi_m K_m(s, a) u_i^2 \right. \\
 &\quad \left. + \sum_{i=1}^M \sum_{j=1}^M \sum_{m=1}^M \pi_i \pi_j \pi_m K_m(s, a) u_i u_j - \sum_{i=1}^M \sum_{j=1}^M \pi_i \pi_j K_j(s, a) u_i u_j \right. \\
 &\quad \left. + \sum_{i=1}^M \sum_{j=1}^M \sum_{m=1}^M \pi_i \pi_j \pi_m K_m(s, a) u_i u_j \right| \\
 &= \left| \sum_{i=1}^M \pi_i K_i(s, a) u_i^2 - 2 \sum_{i=1}^M \sum_{j=1}^M \pi_i \pi_j K_i(s, a) u_i u_j \right. \\
 &\quad \left. - \sum_{i=1}^M \sum_{m=1}^M \pi_i \pi_m K_m(s, a) u_i^2 + 2 \sum_{i=1}^M \sum_{j=1}^M \sum_{m=1}^M \pi_i \pi_j \pi_m K_m(s, a) u_i u_j \right| \\
 &= \left| \sum_{i=1}^M \pi_i u_i^2 \left(K_i(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right) - 2 \sum_{i=1}^M \pi_i u_i \sum_{j=1}^M \pi_j u_j \left(K_i(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right) \right| \\
 &\leq \sum_{i=1}^M \pi_i u_i^2 \underbrace{\left| K_i(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right|}_{\leq 1} + 2 \sum_{i=1}^M \pi_i |u_i| \sum_{j=1}^M \pi_j |u_j| \underbrace{\left| K_i(s, a) - \sum_{m=1}^M \pi_m K_m(s, a) \right|}_{\leq 1} \\
 &\leq \|u\|_2^2 + 2 \sum_{i=1}^M \pi_i |u_i| \sum_{j=1}^M \pi_j |u_j| \leq 3 \|u\|_2^2.
 \end{aligned}$$

The rest of the proof is similar to (Mei et al., 2020) and we include this for completeness. Define $P(\alpha) \in \mathbb{R}^{S \times S}$, where $\forall (s, s')$,

$$[P(\alpha)]_{(s,s')} = \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(a | s) \cdot \mathbb{P}(s' | s, a).$$

The derivative w.r.t. α is,

$$\left[\frac{\partial}{\partial \alpha} P(\alpha) \Big|_{\alpha=0} \right]_{(s,s')} = \sum_{a \in \mathcal{A}} \left[\frac{\partial}{\partial \alpha} \pi_{\theta_\alpha}(a | s) \Big|_{\alpha=0} \right] \cdot \mathbb{P}(s' | s, a).$$

For any vector $x \in \mathbb{R}^S$,

$$\left[\frac{\partial}{\partial \alpha} P(\alpha) \Big|_{\alpha=0} x \right]_{(s)} = \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left[\frac{\partial}{\partial \alpha} \pi_{\theta_\alpha}(a | s) \Big|_{\alpha=0} \right] \cdot \mathbb{P}(s' | s, a) \cdot x(s').$$

The l_∞ norm can be upper-bounded as,

$$\left\| \frac{\partial}{\partial \alpha} P(\alpha) \Big|_{\alpha=0} x \right\|_\infty = \max_{s \in \mathcal{S}} \left| \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left[\frac{\partial}{\partial \alpha} \pi_{\theta_\alpha}(a | s) \Big|_{\alpha=0} \right] \cdot \mathbb{P}(s' | s, a) \cdot x(s') \right|$$

$$\begin{aligned} &\leq \max_{s \in \mathcal{S}} \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left| \frac{\partial}{\partial \alpha} \pi_{\theta_\alpha}(a | s) \Big|_{\alpha=0} \right| \cdot \mathbb{P}(s' | s, a) \cdot \|x\|_\infty \\ &\leq 2 \|u\|_2 \|x\|_\infty. \end{aligned}$$

Now we find the second derivative,

$$\left[\frac{\partial^2 P(\alpha)}{\partial \alpha^2} \Big|_{\alpha=0} \right]_{(s, s')} = \sum_{a \in \mathcal{A}} \left[\frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \alpha^2} \Big|_{\alpha=0} \right] \mathbb{P}(s' | s, a)$$

taking the l_∞ norm,

$$\begin{aligned} \left\| \left[\frac{\partial^2 P(\alpha)}{\partial \alpha^2} \Big|_{\alpha=0} \right] x \right\|_\infty &= \max_s \left| \sum_{s' \in \mathcal{S}} \sum_{a \in \mathcal{A}} \left[\frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \alpha^2} \Big|_{\alpha=0} \right] \mathbb{P}(s' | s, a) x(s') \right| \\ &\leq \max_s \sum_{s' \in \mathcal{S}} \left[\left| \frac{\partial^2 \pi_{\theta_\alpha}(a | s)}{\partial \alpha^2} \Big|_{\alpha=0} \right| \right] \mathbb{P}(s' | s, a) \|x\|_\infty \leq 3 \|u\|_2 \|x\|_\infty. \end{aligned}$$

Next we observe that the value function of π_{θ_α} :

$$V^{\pi_{\theta_\alpha}}(s) = \underbrace{\sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(a | s) r(s, a)}_{r_{\theta_\alpha}} + \gamma \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(a | s) \sum_{s' \in \mathcal{S}} \mathbb{P}(s' | s, a) V^{\pi_{\theta_\alpha}}(s').$$

In matrix form,

$$\begin{aligned} V^{\pi_{\theta_\alpha}} &= r_{\theta_\alpha} + \gamma P(\alpha) V^{\pi_{\theta_\alpha}} \\ \implies (Id - \gamma P(\alpha)) V^{\pi_{\theta_\alpha}} &= r_{\theta_\alpha} \\ V^{\pi_{\theta_\alpha}} &= (Id - \gamma P(\alpha))^{-1} r_{\theta_\alpha}. \end{aligned}$$

Let $M(\alpha) := (Id - \gamma P(\alpha))^{-1} = \sum_{t=0}^{\infty} \gamma^t [P(\alpha)]^t$. Also, observe that

$$\begin{aligned} \mathbf{1} &= \frac{1}{1-\gamma} (Id - \gamma P(\alpha)) \mathbf{1} \implies M(\alpha) \mathbf{1} = \frac{1}{1-\gamma} \mathbf{1}. \\ \implies \forall i \|[M(\alpha)]_{i,:}\|_1 &= \frac{1}{1-\gamma} \end{aligned}$$

where $[M(\alpha)]_{i,:}$ is the i^{th} row of $M(\alpha)$. Hence for any vector $x \in \mathbb{R}^S$, $\|M(\alpha)x\|_\infty \leq \frac{1}{1-\gamma} \|x\|_\infty$.

By assumption I.6, we have $\|r_{\theta_\alpha}\|_\infty = \max_s |r_{\theta_\alpha}(s)| \leq 1$. Next we find the derivative of r_{θ_α} w.r.t α .

$$\begin{aligned} \left| \frac{\partial r_{\theta_\alpha}(s)}{\partial \alpha} \right| &= \left| \left(\frac{\partial r_{\theta_\alpha}(s)}{\partial \theta_\alpha} \right)^T \frac{\partial \theta_\alpha}{\partial \alpha} \right| \\ &\leq \left| \sum_{m''=1}^M \sum_{m=1}^M \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(m'') (l_{mm''} - \pi_{\theta_\alpha}(m)) K_m(s, a) r(s, a) u(m'') \right| \\ &= \left| \sum_{m''=1}^M \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(m'') K_{m''}(s, a) r(s, a) u(m'') - \sum_{m''=1}^M \sum_{m=1}^M \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(m'') \pi_{\theta_\alpha}(m) K_m(s, a) r(s, a) u(m'') \right| \\ &\leq \left| \sum_{m''=1}^M \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(m'') K_{m''}(s, a) r(s, a) - \sum_{m''=1}^M \sum_{m=1}^M \sum_{a \in \mathcal{A}} \pi_{\theta_\alpha}(m'') \pi_{\theta_\alpha}(m) K_m(s, a) r(s, a) \right| \|u\|_\infty \leq \|u\|_2. \end{aligned}$$

Similarly, we can calculate the upper-bound on second derivative,

$$\begin{aligned}
 \left\| \frac{\partial r_{\theta_\alpha}}{\partial \alpha^2} \right\|_\infty &= \max_s \left| \frac{\partial r_{\theta_\alpha}(s)}{\partial \alpha^2} \right| \\
 &= \max_s \left| \left(\frac{\partial}{\partial \alpha} \left\{ \frac{\partial r_{\theta_\alpha}(s)}{\partial \alpha} \right\} \right)^\top \frac{\partial \theta_\alpha}{\partial \alpha} \right| \\
 &= \max_s \left| \left(\frac{\partial^2 r_{\theta_\alpha}(s)}{\partial \alpha^2} \frac{\partial \theta_\alpha}{\partial \alpha} \right)^\top \frac{\partial \theta_\alpha}{\partial \alpha} \right| \leq 5/2 \|u\|_2^2.
 \end{aligned}$$

Next, the derivative of the value function w.r.t α is given by,

$$\frac{\partial V^{\pi_{\theta_\alpha}}(s)}{\partial \alpha} = \gamma e_s^\top M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) r_{\theta_\alpha} + e_s^\top M(\alpha) \frac{\partial r_{\theta_\alpha}}{\partial \alpha}.$$

And the second derivative,

$$\begin{aligned}
 \frac{\partial^2 V^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2} &= \underbrace{2\gamma^2 e_s^\top M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) r_{\theta_\alpha}}_{T1} + \underbrace{\gamma e_s^\top M(\alpha) \frac{\partial^2 P(\alpha)}{\partial \alpha^2} M(\alpha) r_{\theta_\alpha}}_{T2} \\
 &\quad + \underbrace{2\gamma e_s^\top M(\alpha) \frac{\partial P(\alpha)}{\partial \alpha} M(\alpha) \frac{\partial r_{\theta_\alpha}}{\partial \alpha}}_{T3} + \underbrace{e_s^\top M(\alpha) \frac{\partial^2 r_{\theta_\alpha}}{\partial \alpha^2}}_{T4}.
 \end{aligned}$$

We use the above derived bounds to bound each of the term in the above display. The calculations here are same as shown for Lemma 7 in (Mei et al., 2020), except for the particular values of the bounds. Hence we directly, mention the final bounds that we obtain and refer to (Mei et al., 2020) for the detailed but elementary calculations.

$$\begin{aligned}
 |T1| &\leq \frac{4}{(1-\gamma)^3} \|u\|_2^2 \\
 |T2| &\leq \frac{3}{(1-\gamma)^2} \|u\|_2^2 \\
 |T3| &\leq \frac{2}{(1-\gamma)^2} \|u\|_2^2 \\
 |T4| &\leq \frac{5/2}{(1-\gamma)} \|u\|_2^2.
 \end{aligned}$$

Combining the above bounds we get,

$$\begin{aligned}
 \left| \frac{\partial^2 V^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2} \Big|_{\alpha=0} \right| &\leq \left(\frac{8\gamma^2}{(1-\gamma)^3} + \frac{3\gamma}{(1-\gamma)^2} + \frac{4\gamma}{(1-\gamma)^2} + \frac{5/2}{(1-\gamma)} \right) \|u\|_2^2 \\
 &= \frac{7\gamma^2 + 4\gamma + 5}{2(1-\gamma)^3} \|u\|_2^2.
 \end{aligned}$$

Finally, let $y \in \mathbb{R}^M$ and fix a $\theta \in \mathbb{R}^M$:

$$\begin{aligned}
 \left| y^\top \frac{\partial^2 V^{\pi_\theta}(s)}{\partial \theta^2} y \right| &= \left| \frac{y}{\|y\|_2}^\top \frac{\partial^2 V^{\pi_\theta}(s)}{\partial \theta^2} \frac{y}{\|y\|_2} \right| \cdot \|y\|_2^2 \\
 &\leq \max_{\|u\|_2=1} \left| \left\langle \frac{\partial^2 V^{\pi_\theta}(s)}{\partial \theta^2} u, u \right\rangle \right| \cdot \|y\|_2^2 \\
 &= \max_{\|u\|_2=1} \left| \left\langle \frac{\partial^2 V^{\pi_{\theta_\alpha}}(s)}{\partial \theta_\alpha^2} \Big|_{\alpha=0}, \frac{\partial \theta_\alpha}{\partial \alpha}, \frac{\partial \theta_\alpha}{\partial \alpha} \right\rangle \right| \cdot \|y\|_2^2 \\
 &= \max_{\|u\|_2=1} \left| \frac{\partial^2 V^{\pi_{\theta_\alpha}}(s)}{\partial \alpha^2} \Big|_{\alpha=0} \right| \cdot \|y\|_2^2
 \end{aligned}$$

$$\leq \frac{7\gamma^2 + 4\gamma + 5}{2(1-\gamma)^3} \|y\|_2^2.$$

Let $\theta_\xi := \theta + \xi(\theta' - \theta)$ where $\xi \in [0, 1]$. By Taylor's theorem $\forall s, \theta, \theta'$,

$$\begin{aligned} \left| V^{\pi_{\theta'}}(s) - V^{\pi_\theta}(s) - \left\langle \frac{\partial V^{\pi_\theta}(s)}{\partial \theta} \right\rangle (\theta' - \theta) \right| &= \frac{1}{2} \left| (\theta' - \theta)^\top \frac{\partial^2 V^{\pi_{\theta_\xi}}(s)}{\partial \theta_\xi^2} (\theta' - \theta) \right| \\ &\leq \frac{7\gamma^2 + 4\gamma + 5}{4(1-\gamma)^3} \|\theta' - \theta\|_2^2. \end{aligned}$$

Since $V^{\pi_\theta}(s)$ is $\frac{7\gamma^2 + 4\gamma + 5}{2(1-\gamma)^3}$ smooth for every s , $V^{\pi_\theta}(\mu)$ is also $\frac{7\gamma^2 + 4\gamma + 5}{2(1-\gamma)^3}$ - smooth. \square

Lemma I.4 (Value Difference Lemma-1). *For any two policies π and π' , and for any state $s \in \mathcal{S}$, the following is true.*

$$V^{\pi'}(s) - V^\pi(s) = \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^{\pi'}(s') \sum_{m=1}^M \pi'_m \tilde{A}(s', m).$$

Proof.

$$\begin{aligned} V^{\pi'}(s) - V^\pi(s) &= \sum_{m=1}^M \pi'_m \tilde{Q}'(s, m) - \sum_{m=1}^M \pi_m \tilde{Q}(s, m) \\ &= \sum_{m=1}^M \pi'_m (\tilde{Q}'(s, m) - \tilde{Q}(s, m)) + \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}(s, m) \\ &= \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}(s, m) + \underbrace{\sum_{m=1}^M \pi'_m \sum_{a \in \mathcal{A}} K_m(s, a)}_{= \sum_{a \in \mathcal{A}} \pi_\theta(a|s)} \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s, a) [V^{\pi'}(s') - V^\pi(s')] \\ &= \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^{\pi'}(s') \sum_{m'=1}^M (\pi'_{m'} - \pi_{m'}) \tilde{Q}(s', m') \\ &= \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^{\pi'}(s') \sum_{m'=1}^M \pi'_{m'} (\tilde{Q}(s', m') - V(s')) \\ &= \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^{\pi'}(s') \sum_{m'=1}^M \pi'_{m'} \tilde{A}(s', m'). \end{aligned}$$

\square

Lemma I.5. (Value Difference Lemma-2) *For any two policies π and π' and state $s \in \mathcal{S}$, the following is true.*

$$V^{\pi'}(s) - V^\pi(s) = \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^\pi(s') \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}^{\pi'}(s', m).$$

Proof. We will use \tilde{Q} for \tilde{Q}^π and \tilde{Q}' for $\tilde{Q}^{\pi'}$ as a shorthand.

$$\begin{aligned}
 V^{\pi'}(s) - V^\pi(s) &= \sum_{m=1}^M \pi'_m \tilde{Q}'(s, m) - \sum_{m=1}^M \pi_m \tilde{Q}(s, m) \\
 &= \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}'(s, m) + \sum_{m=1}^M \pi_m (\tilde{Q}'(s, m) - \tilde{Q}(s, m)) \\
 &= \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}'(s, m) + \\
 &\quad \gamma \sum_{m=1}^M \pi_m \left(\sum_{a \in \mathcal{A}} K_m(s, a) \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s, a) V'(s') - \sum_{a \in \mathcal{A}} K_m(s, a) \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s, a) V(s') \right) \\
 &= \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}'(s, m) + \gamma \sum_{a \in \mathcal{A}} \pi_\theta(a|s) \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s, a) [V'(s') - V(s')] \\
 &= \frac{1}{1-\gamma} \sum_{s' \in \mathcal{S}} d_s^\pi(s') \sum_{m=1}^M (\pi'_m - \pi_m) \tilde{Q}'(s', m).
 \end{aligned}$$

□

Assumption I.6. The reward $r(s, a) \in [0, 1]$, for all pairs $(s, a) \in \mathcal{S} \times \mathcal{A}$.

Assumption I.7. Let $\pi^* := \operatorname{argmax}_{\pi \in \mathcal{P}_M} V^\pi(s_0)$. We make the following assumption.

$$E_{m \sim \pi^*} [Q^{\pi_\theta}(s, m)] - V^{\pi_\theta}(s) \geq 0, \forall s \in \mathcal{S}, \forall \pi_\theta \in \Pi.$$

Let the best controller be a point in the M -simplex, i.e., $K^* := \sum_{m=1}^M \pi_m^* K_m$.

Lemma I.8 (NUED). $\left\| \frac{\partial}{\partial \theta} V^{\pi_\theta}(\mu) \right\|_2 \geq \frac{1}{\sqrt{M}} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \times \left\| \frac{d_\mu^{\pi^*}}{d_\mu^{\pi_\theta}} \right\|_\infty^{-1} \times [V^*(\rho) - V^{\pi_\theta}(\rho)].$

Proof.

$$\begin{aligned}
 \left\| \frac{\partial}{\partial \theta} V^{\pi_\theta}(\mu) \right\|_2 &= \left(\sum_{m=1}^M \left(\frac{\partial V^{\pi_\theta}(\mu)}{\partial \theta_m} \right)^2 \right)^{1/2} \\
 &\geq \frac{1}{\sqrt{M}} \sum_{m=1}^M \left| \frac{\partial V^{\pi_\theta}(\mu)}{\partial \theta_m} \right| \quad (\text{Cauchy-Schwarz}) \\
 &= \frac{1}{\sqrt{M}} \sum_{m=1}^M \frac{1}{1-\gamma} \left| \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \pi_m \tilde{A}(s, m) \right| \quad \text{Lemma I.2} \\
 &\geq \frac{1}{\sqrt{M}} \sum_{m=1}^M \frac{\pi_m^* \pi_m}{1-\gamma} \left| \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \tilde{A}(s, m) \right| \\
 &\geq \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \frac{1}{\sqrt{M}} \sum_{m=1}^M \frac{\pi_m^*}{1-\gamma} \left| \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \tilde{A}(s, m) \right| \\
 &\geq \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \frac{1}{\sqrt{M}} \left| \sum_{m=1}^M \frac{\pi_m^*}{1-\gamma} \sum_{s \in \mathcal{S}} d_\mu^{\pi_\theta}(s) \tilde{A}(s, m) \right|
 \end{aligned}$$

$$\begin{aligned}
 &= \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \left| \frac{1}{\sqrt{M}} \sum_{s \in \mathcal{S}} d_{\mu}^{\pi_{\theta}}(s) \sum_{m=1}^M \frac{\pi_m^*}{1-\gamma} \tilde{A}(s, m) \right| \\
 &= \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \frac{1}{\sqrt{M}} \sum_{s \in \mathcal{S}} d_{\mu}^{\pi_{\theta}}(s) \sum_{m=1}^M \frac{\pi_m^*}{1-\gamma} \tilde{A}(s, m) \quad \text{Assumption I.7} \\
 &\geq \frac{1}{\sqrt{M}} \frac{1}{1-\gamma} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-1} \sum_{s \in \mathcal{S}} d_{\rho}^*(s) \sum_{m=1}^M \pi_m^* \tilde{A}(s, m) \\
 &= \frac{1}{\sqrt{M}} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right) \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-1} [V^*(\rho) - V^{\pi_{\theta}}(\rho)] \quad \text{Lemma I.4.}
 \end{aligned}$$

□

I.1. Proof of the Theorem 4.2

Lemma I.9 (Modified Policy Gradient Theorem). $\nabla_{\theta} V^{\pi_{\theta}}(\rho) = \mathbb{E}_{(s,m) \sim \nu_{\pi_{\theta}}} [\tilde{Q}^{\pi_{\theta}}(s, m) \psi_{\theta}(m)] = \mathbb{E}_{(s,m) \sim \nu_{\pi_{\theta}}} [\tilde{A}^{\pi_{\theta}}(s, m) \psi_{\theta}(m)]$, where $\psi_{\theta}(m) := \nabla_{\theta} \log(\pi_{\theta}(m))$.

Let $\beta := \frac{7\gamma^2 + 4\gamma + 5}{(1-\gamma)^2}$. We have that,

$$\begin{aligned}
 V^*(\rho) - V^{\pi_{\theta}}(\rho) &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d_{\rho}^{\pi_{\theta}}(s) \sum_{m=1}^M (\pi_m^* - \pi_m) \tilde{Q}^{\pi^*}(s, m) \quad \text{(Lemma I.5)} \\
 &= \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} \frac{d_{\rho}^{\pi_{\theta}}(s)}{d_{\mu}^{\pi_{\theta}}(s)} d_{\mu}^{\pi_{\theta}}(s) \sum_{m=1}^M (\pi_m^* - \pi_m) \tilde{Q}^{\pi^*}(s, m) \\
 &\leq \frac{1}{1-\gamma} \left\| \frac{1}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty} \sum_{s \in \mathcal{S}} \sum_{m=1}^M (\pi_m^* - \pi_m) \tilde{Q}^{\pi^*}(s, m) \\
 &\leq \frac{1}{(1-\gamma)^2} \left\| \frac{1}{\mu} \right\|_{\infty} \sum_{s \in \mathcal{S}} \sum_{m=1}^M (\pi_m^* - \pi_m) \tilde{Q}^{\pi^*}(s, m) \\
 &= \frac{1}{(1-\gamma)} \left\| \frac{1}{\mu} \right\|_{\infty} [V^*(\mu) - V^{\pi_{\theta}}(\mu)] \quad \text{(Lemma I.5).}
 \end{aligned}$$

Let $\delta_t := V^*(\mu) - V^{\pi_{\theta_t}}(\mu)$.

$$\begin{aligned}
 \delta_{t+1} - \delta_t &= V^{\pi_{\theta_t}}(\mu) - V^{\pi_{\theta_{t+1}}}(\mu) \quad \text{(Lemma I.3)} \\
 &\leq -\frac{1}{2\beta} \left\| \frac{\partial}{\partial \theta} V^{\pi_{\theta_t}}(\mu) \right\|_2^2 \quad \text{(Lemma I.10)} \\
 &\leq -\frac{1}{2\beta} \frac{1}{M} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right)^2 \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-2} \delta_t^2 \quad \text{(Lemma 4.5)} \\
 &\leq -\frac{1}{2\beta} (1-\gamma)^2 \frac{1}{M} \left(\min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right)^2 \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-2} \delta_t^2 \\
 &\leq -\frac{1}{2\beta} (1-\gamma)^2 \frac{1}{M} \left(\min_{1 \leq s \leq t} \min_{m: \pi_{\theta_m}^* > 0} \pi_{\theta_m} \right)^2 \left\| \frac{d_{\rho}^{\pi^*}}{d_{\mu}^{\pi_{\theta}}} \right\|_{\infty}^{-2} \delta_t^2 \\
 &= -\frac{1}{2\beta} \frac{1}{M} (1-\gamma)^2 \left\| \frac{d_{\mu}^{\pi^*}}{\mu} \right\|_{\infty}^{-2} c_t^2 \delta_t^2,
 \end{aligned}$$

where $c_t := \min_{1 \leq s \leq t} \min_{m: \pi_m^* > 0} \pi_{\theta_s}(m)$. Hence we have that,

$$\delta_{t+1} \leq \delta_t - \frac{1}{2\beta} \frac{(1-\gamma)^2}{M} \left\| \frac{d_{\mu}^{\pi^*}}{\mu} \right\|_{\infty}^{-2} c_t^2 \delta_t^2. \quad (19)$$

The rest of the proof follows from a induction argument over $t \geq 1$.

Base case: Since $\delta_t \leq \frac{1}{1-\gamma}$, and $c_t \in (0, 1)$, the result holds for all $t \leq \frac{2\beta M}{(1-\gamma)} \left\| \frac{d_{\mu}^{\pi^*}}{\mu} \right\|_{\infty}^2$.

For ease of notation, let $\varphi_t := \frac{2\beta M}{c_t^2(1-\gamma)^2} \left\| \frac{d_{\mu}^{\pi^*}}{\mu} \right\|_{\infty}^2$. We need to show that $\delta_t \leq \frac{\varphi_t}{t}$, for all $t \geq 1$.

Induction step: Fix a $t \geq 2$, assume $\delta_t \leq \frac{\varphi_t}{t}$.

Let $g : \mathbb{R} \rightarrow \mathbb{R}$ be a function defined as $g(x) = x - \frac{1}{\varphi_t} x^2$. One can verify easily that g is monotonically increasing in $[0, \frac{\varphi_t}{2}]$. Next with equation 19, we have

$$\begin{aligned} \delta_{t+1} &\leq \delta_t - \frac{1}{\varphi_t} \delta_t^2 \\ &= g(\delta_t) \\ &\leq g\left(\frac{\varphi_t}{t}\right) \\ &\leq \frac{\varphi_t}{t} - \frac{\varphi_t}{t^2} \\ &= \varphi_t \left(\frac{1}{t} - \frac{1}{t^2} \right) \\ &\leq \varphi_t \left(\frac{1}{t+1} \right) \\ &\leq \varphi_{t+1} \left(\frac{1}{t+1} \right). \end{aligned}$$

where the last step follows from the fact that $c_{t+1} \leq c_t$ (infimum over a larger set does not increase the value). This completes the proof.

Lemma I.10. Let $f : \mathbb{R}^M \rightarrow \mathbb{R}$ be β -smooth. Then gradient ascent with learning rate $\frac{1}{\beta}$ guarantees, for all $x, x' \in \mathbb{R}^M$:

$$f(x) - f(x') \leq -\frac{1}{2\beta} \left\| \frac{df(x)}{dx} \right\|_2^2.$$

Proof.

$$\begin{aligned} f(x) - f(x') &\leq -\left\langle \frac{\partial f(x)}{\partial x}, x' - x \right\rangle + \frac{\beta}{2} \|x' - x\|_2^2 \\ &= \frac{1}{\beta} \left\| \frac{df(x)}{dx} \right\|_2^2 + \frac{\beta}{2} \frac{1}{\beta^2} \left\| \frac{df(x)}{dx} \right\|_2^2 \\ &= -\frac{1}{2\beta} \left\| \frac{df(x)}{dx} \right\|_2^2. \end{aligned}$$

□

J. Proofs for (Natural) Actor-critic based improper learning

We will begin with some useful lemmas.

Lemma J.1. For any $\theta, \theta' \in \mathbb{R}^M$, we have $\|\psi_\theta(m) - \psi_{\theta'}(m)\|_2 \leq \|\theta - \theta'\|_2$.

Proof. Recall, $\psi_\theta(m) := \nabla_\theta \log \pi_\theta(m)$. Fix $m' \in [M]$,

$$\begin{aligned} \frac{\partial \log \pi_\theta(m)}{\partial \theta_{m'}} &= \frac{\partial \log \left(\frac{e^{\theta_m}}{\sum_{j=1}^M e^{\theta_j}} \right)}{\partial \theta_{m'}} \\ &= \frac{\partial}{\partial \theta_{m'}} \left(\theta_m - \log \left(\sum_{j=1}^M e^{\theta_j} \right) \right) \\ &= \mathbb{1}\{m' = m\} - \frac{e^{\theta_{m'}}}{\sum_{j=1}^M e^{\theta_j}} \\ &= \mathbb{1}\{m' = m\} - \pi_\theta(m'). \end{aligned}$$

$$\begin{aligned} \|\psi_\theta(m) - \psi_{\theta'}(m)\|_2 &\leq \|\theta - \theta'\|_2 = \|\nabla_\theta \log \pi_\theta(m) - \nabla_{\theta'} \log \pi_{\theta'}(m)\|_2 \\ &= \|\pi_\theta(\cdot) - \pi_{\theta'}(\cdot)\|_2 \\ &\leq^{(*)} \|\theta - \theta'\|_2. \end{aligned}$$

Here (*) follows from the fact that the softmax function is 1-Lipschitz (Gao & Pavel, 2017). \square

Lemma J.2. For all $m \in [M]$ and $\theta \in \mathbb{R}^M$, $\|\psi_\theta(m)\|_2 \leq \sqrt{2}$.

Proof. Proof follows by noticing that $\|\psi_\theta(m)\|_2 = \|\nabla_\theta \log \pi_\theta(m)\|_2 \leq \sqrt{2}$, where the last inequality follows because the 2-norm of a probability vector is bounded by 1. \square

Lemma J.3. For all $\theta, \theta' \in \mathbb{R}^M$, $\|\pi_\theta(\cdot) - \pi_{\theta'}(\cdot)\|_{TV} \leq \frac{\sqrt{M}}{2} \|\theta - \theta'\|_2$.

Proof.

$$\begin{aligned} \|\pi_\theta(\cdot) - \pi_{\theta'}(\cdot)\|_{TV} &= \frac{1}{2} \|\pi_\theta(\cdot) - \pi_{\theta'}(\cdot)\|_1 \\ &\leq \frac{\sqrt{M}}{2} \|\pi_\theta(\cdot) - \pi_{\theta'}(\cdot)\|_2. \end{aligned}$$

The inequality follows from relation between 1-norm and 2-norm. \square

Proposition J.4. For any $\theta, \theta' \in \mathbb{R}^M$,

$$\nabla V(\theta) - \nabla V(\theta') \leq \sqrt{M} L_V \|\theta - \theta'\|_2$$

where $L_V = \frac{2\sqrt{2}C_{\kappa\xi} + 1}{1-\gamma}$, and $C_{\kappa\xi} = \left(1 + \lceil \log_\xi \frac{1}{\kappa} \rceil + \frac{1}{1-\xi}\right)$.

Proof. We follow the same steps as in Proposition 1 in (Xu et al., 2020) along with Lemmas J.3, J.2, J.1 and that the maximum reward is bounded by 1. \square

We will now restate a useful result from (Xu et al., 2020), about the convergence of the critic parameter w_t to the equilibrium point w^* of the underlying ODE, applied to our setting.

Proposition J.5. *Suppose assumptions 5.3 and 5.2 hold. Then is $\beta \leq \min \left\{ \frac{\Gamma_L}{16}, \frac{8}{\Gamma_L} \right\}$ and $H \geq \left(\frac{4}{\Gamma_L} + 2\alpha \right) \left[\frac{1536[1+(\kappa-1)\xi]}{(1-\xi)\Gamma_L} \right]$. We have*

$$\mathbb{E} \left[\|w_{T_c} - w^*\|_2^2 \right] \leq \left(1 - \frac{\Gamma_L}{16} \alpha \right)^{T_c} \|w_0 - w^*\|_2^2 + \left(\frac{4}{\Gamma_L} + 2\alpha \right) \frac{1536(1 + R_w^2)[1 + (\kappa - 1)\xi]}{(1 - \xi)H}.$$

If we further let $T_c \geq \frac{16}{\Gamma_L \alpha} \log \frac{2\|w_0 - w^\|_2^2}{\varepsilon}$ and $H \geq \left(\frac{4}{\Gamma_L} + 2\alpha \right) \frac{3072(R_w^2 + 1)[1 + (\kappa - 1)\xi]}{(1 - \xi)\Gamma_L \varepsilon}$, then we have $\mathbb{E} \left[\|w_{T_c} - w^*\|_2^2 \right] \leq \varepsilon$ with total sample complexity given by $T_c H = \mathcal{O} \left(\frac{1}{\alpha \varepsilon} \log \frac{1}{\varepsilon} \right)$.*

Proof. Proof follows along the similar lines as in Thm. 4 in Xu et al. (2020) and by using $\|\varphi(s)(\gamma\varphi(s') - \varphi(s))^\top\|_F \leq (1 + \gamma) \leq 2$ and assuming $\|\varphi(s)\|_2 \leq 1$ for all $s, s' \in \mathcal{S}$. \square

J.1. Actor-critic based improper learning

Proof of Theorem 5.4. Let $v_t(w) := \frac{1}{B} \sum_{i=0}^{B-1} \mathcal{E}(s_{t,i}, m_{t,i}, s_{t,i+1}) \psi_{\theta_t}(m_{t,i})$ and $A_w(s, m) := \mathbb{E}_{\bar{P}} [\mathcal{E}(s, m, s') | (s, m)]$ and $g(w, \theta) := \mathbb{E}_{\nu_\theta} [A_w(s, m) \psi_\theta(m)]$ for all $\theta \in \mathbb{R}^M, w \in \mathbb{R}^d, s \in \mathcal{S}, m \in [M]$. Using Prop J.4 we get,

$$\begin{aligned} V(\theta_{t+1}) &\geq V(\theta_t) + \langle \nabla_\theta V(\theta_t), \theta_{t+1} - \theta_t \rangle - \frac{\sqrt{M}L_V}{2} \|\theta_{t+1} - \theta_t\|_2^2 \\ &= V(\theta_t) + \alpha \langle \nabla_\theta V(\theta_t), v_t(w_t) - \nabla_\theta V(\theta_t) + \nabla_\theta V(\theta_t) \rangle - \frac{\sqrt{M}L_V \alpha^2}{2} \|v_t(w_t)\|_2^2 \\ &= V(\theta_t) + \alpha \|\nabla_\theta V(\theta_t)\|_2^2 \\ &\quad + \alpha \langle \nabla_\theta V(\theta_t), v_t(w_t) - \nabla_\theta V(\theta_t) \rangle - \frac{\sqrt{M}L_V \alpha^2}{2} \|v_t(w_t)\|_2^2 \\ &\geq V(\theta_t) + \left(\frac{1}{2} \alpha - \sqrt{M}L_V \alpha^2 \right) \|\nabla_\theta V(\theta_t)\|_2^2 - \left(\frac{1}{2} \alpha + \sqrt{M}L_V \alpha^2 \right) \|v_t(w_t) - \nabla_\theta V(\theta_t)\|_2^2 \end{aligned}$$

Taking expectations and rearranging, we have

$$\begin{aligned} &\left(\frac{1}{2} \alpha - \sqrt{M}L_V \alpha^2 \right) \mathbb{E} \left[\|\nabla_\theta V(\theta_t)\|_2^2 | \mathcal{F}_t \right] \\ &\leq \mathbb{E} [V(\theta_{t+1}) | \mathcal{F}_t] - V(\theta_t) + \left(\frac{1}{2} \alpha + \sqrt{M}L_V \alpha^2 \right) \mathbb{E} \left[\|v_t(w_t) - \nabla_\theta V(\theta_t)\|_2^2 | \mathcal{F}_t \right]. \end{aligned}$$

Next we will upperbound $\mathbb{E} \left[\|v_t(w_t) - \nabla_\theta V(\theta_t)\|_2^2 | \mathcal{F}_t \right]$.

$$\begin{aligned} &\|v_t(w_t) - \nabla_\theta V(\theta_t)\|_2^2 \\ &\leq 3 \|v_t(w_t) - v_t(w_{\theta_t}^*)\|_2^2 + 3 \|v_t(w_{\theta_t}^*) - g(w_{\theta_t}^*)\|_2^2 + 3 \|g(w_{\theta_t}^*) - \nabla_\theta V(\theta_t)\|_2^2. \end{aligned}$$

$$\begin{aligned} \|v_t(w_t) - v_t(w_{\theta_t}^*)\|_2^2 &= \left\| \frac{1}{B} \sum_{i=0}^{B-1} [\mathcal{E}_{w_t}(s_{t,i}, m_{t,i}, s_{t,i+1}) - \mathcal{E}_{w_{\theta_t}^*}(s_{t,i}, m_{t,i}, s_{t,i+1})] \psi(m_{t,i}) \right\|_2^2 \\ &\leq \frac{1}{B} \sum_{i=0}^{B-1} \left\| [\mathcal{E}_{w_t}(s_{t,i}, m_{t,i}, s_{t,i+1}) - \mathcal{E}_{w_{\theta_t}^*}(s_{t,i}, m_{t,i}, s_{t,i+1})] \psi(m_{t,i}) \right\|_2^2 \\ &\leq \frac{2}{B} \sum_{i=0}^{B-1} \left\| [\mathcal{E}_{w_t}(s_{t,i}, m_{t,i}, s_{t,i+1}) - \mathcal{E}_{w_{\theta_t}^*}(s_{t,i}, m_{t,i}, s_{t,i+1})] \right\|_2^2 \\ &= \frac{2}{B} \sum_{i=0}^{B-1} \|(\gamma\varphi(s_{t,i+1}) - \varphi(s_{t,i}))^\top (w_t - w_{\theta_t}^*)\|_2^2 \end{aligned}$$

$$\leq \frac{8}{B} \sum_{i=0}^{B-1} \|(w_t - w_{\theta_t}^*)\|_2^2 = 8 \|(w_t - w_{\theta_t}^*)\|_2^2.$$

Next we have,

$$\begin{aligned} \|g(w_{\theta_t}^*) - \nabla_{\theta} V(\theta_t)\|_2^2 &= \left\| \mathbb{E}_{\nu_{\theta_t}} [A_{w_{\theta_t}^*}(s, m) \psi_{\theta_t}(m)] - \mathbb{E}_{\nu_{\theta_t}} [A_{\pi_{\theta_t}}(s, m) \psi_{\theta_t}(m)] \right\|_2^2 \\ &\leq 2 \mathbb{E}_{\nu_{\theta_t}} \left\| A_{w_{\theta_t}^*}(s, m) - A_{\pi_{\theta_t}}(s, m) \right\|_2^2 \\ &= 2 \mathbb{E}_{\nu_{\theta_t}} \left[|\gamma \mathbb{E} [V_{w_{\theta_t}^*}(s') - V_{\pi_{\theta_t}}(s') | s, m] + V_{\pi_{\theta_t}}(s) - V_{w_{\theta_t}^*}(s)|^2 \right] \\ &\leq 8 \Delta_{critic}. \end{aligned}$$

Finally we bound the last term $\|v_t(w_{\theta_t}^*) - g(w_{\theta_t}^*)\|_2^2$ by using Assumption 5.2 we have,

$$\|v_t(w_{\theta_t}^*) - g(w_{\theta_t}^*)\|_2^2 \leq \mathbb{E} \left[\left\| \frac{1}{B} \sum_{i=0}^{B-1} \mathcal{E}_{w_{\theta_t}^*}(s_{t,i}, m_{t,i}, s_{t,i+1}) \psi_{\theta_t}(m_{t,i}) - \mathbb{E}_{\nu_{\theta_t}} [A_{w_{\theta_t}^*}(s, m) \psi_{\theta_t}(m)] \right\|_2^2 \middle| \mathcal{F}_t \right].$$

We will now proceed in the similar manner as in (Xu et al., 2020) (eq 24 to eq 26), and using Lemma J.2, we have

$$\mathbb{E} \left[\|v_t(w_{\theta_t}^*) - g(w_{\theta_t}^*)\|_2^2 \middle| \mathcal{F}_t \right] \leq \frac{32(1 + R_w)^2 [1 + (\kappa - 1)\xi]}{B(1 - \xi)}.$$

Putting things back we have,

$$\mathbb{E} \left[\|v_t(w_t) - \nabla_{\theta} V(\theta_t)\|_2^2 \middle| \mathcal{F}_t \right] \leq \frac{96(1 + R_w)^2 [1 + (\kappa - 1)\xi]}{B(1 - \xi)} + 24 \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 24 \Delta_{critic}.$$

Hence we get,

$$\begin{aligned} &\left(\frac{1}{2} \alpha - \sqrt{M} L_V \alpha^2 \right) \mathbb{E} \left[\|\nabla_{\theta} V(\theta_t)\|_2^2 \right] \\ &\leq \mathbb{E} [V(\theta_{t+1})] - \mathbb{E} [V(\theta_t)] \\ &+ \left(\frac{1}{2} \alpha + \sqrt{M} L_V \alpha^2 \right) \left(\frac{96(1 + R_w)^2 [1 + (\kappa - 1)\xi]}{B(1 - \xi)} + 24 \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 24 \Delta_{critic} \right). \end{aligned}$$

We put $\alpha = \frac{1}{4L_V\sqrt{M}}$ above to get,

$$\begin{aligned} \left(\frac{1}{16L_V\sqrt{M}} \right) \mathbb{E} \left[\|\nabla_{\theta} V(\theta_t)\|_2^2 \right] &\leq \mathbb{E} [V(\theta_{t+1})] - \mathbb{E} [V(\theta_t)] \\ &+ \left(\frac{1}{4L_V\sqrt{M}} \right) \left(\frac{96(1 + R_w)^2 [1 + (\kappa - 1)\xi]}{B(1 - \xi)} + 24 \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 24 \Delta_{critic} \right). \end{aligned}$$

which simplifies as

$$\begin{aligned} &\mathbb{E} \left[\|\nabla_{\theta} V(\theta_t)\|_2^2 \right] \\ &\leq 16L_V\sqrt{M} (\mathbb{E} [V(\theta_{t+1})] - \mathbb{E} [V(\theta_t)]) + \frac{384(1 + R_w)^2 [1 + (\kappa - 1)\xi]}{B(1 - \xi)} + 96 \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 96 \Delta_{critic}. \end{aligned}$$

Taking summation over $t = 0, 1, 2, \dots, T-1$ and dividing by T ,

$$\begin{aligned}
 & \mathbb{E} \left[\|\nabla_{\theta} V(\theta_{\hat{T}})\|_2^2 \right] \\
 &= \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left[\|\nabla_{\theta} V(\theta_t)\|_2^2 \right] \\
 &\leq \frac{16L_V\sqrt{M} (\mathbb{E}[V(\theta_T)] - \mathbb{E}[V(\theta_0)])}{T} + \frac{384(1+R_w)^2[1+(\kappa-1)\xi]}{B(1-\xi)} + 96\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 96\Delta_{critic} \\
 &\leq \frac{16L_V\sqrt{M}}{(1-\gamma)T} + \frac{384(1+R_w)^2[1+(\kappa-1)\xi]}{B(1-\xi)} + 96\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] + 96\Delta_{critic}
 \end{aligned}$$

We now let $B \geq \frac{1152}{(1+R_w)^2[1+(\kappa-1)\xi]}(1-\xi)\varepsilon$, $\mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right] \leq \frac{\varepsilon}{288}$ and $T \geq \frac{48L_V\sqrt{M}}{(1-\gamma)\varepsilon}$, then we have

$$\mathbb{E} \left[\|\nabla_{\theta} V(\theta_{\hat{T}})\|_2^2 \right] \leq \varepsilon + \mathcal{O}(\Delta_{critic}).$$

This leads to the final sample complexity of $(B + HT_c)T = \left(\frac{1}{\varepsilon} + \frac{\sqrt{M}}{\varepsilon} \log \frac{1}{\varepsilon}\right) \left(\frac{\sqrt{M}}{(1-\gamma)^2\varepsilon}\right) = \mathcal{O}\left(\frac{M}{(1-\gamma)^2\varepsilon^2} \log \frac{1}{\varepsilon}\right)$ \square

J.2. Natural-actor-critic based improper learning

J.2.1. PROOF OF THEOREM 5.5

Proof. We first show that the natural actor-critic improper learner converges to a stationary point. We will then show convergence to the global optima which is what is different from that of (Xu et al., 2020).

Let $v_t(w) := \frac{1}{B} \sum_{i=0}^{B-1} \mathcal{E}_w(s_{t,i}, m_{t,i}, s_{t,i+1}) \psi_{\theta_t}(m_{t,i})$, $A_w(s, m) := \mathbb{E}_{\tilde{P}}[\mathcal{E}(s, m, s') | s, m]$ and $g(w, \theta) := \mathbb{E}_{\nu_{\theta}}[A_w(s, m) \psi_{\theta}(m)]$ for $w \in \mathbb{R}^d$ and $\theta \in \mathbb{R}^M$. Also let $u_t(w) := [F_t(\theta_t) + \lambda I]^{-1} \left[\frac{1}{B} \sum_{i=0}^{B-1} \mathcal{E}_w(s_{t,i}, m_{t,i}, s_{t,i+1}) \psi_{\theta_t}(m_{t,i}) \right] = [F_t(\theta_t) + \lambda I]^{-1} v_t(w)$.

Recall Prop J.4. We have

Lemma J.6. Assume $\sup_{s \in \mathcal{S}} \|\varphi(s)\|_2 \leq 1$. Under Assumptions 5.2 and 5.3 with step-sizes chosen as $\alpha = \left(\frac{\lambda^2}{2\sqrt{M}L_V(1+\lambda)}\right)$, we have

$$\begin{aligned}
 \mathbb{E} \left[\|\nabla_{\theta} V(\theta_{\hat{T}})\|_2^2 \right] &= \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \left[\|\nabla_{\theta} V(\theta_t)\|_2^2 \right] \\
 &\leq \frac{16\sqrt{M}L_V(1+\lambda)^2}{\lambda^2} \frac{\mathbb{E}[V(\theta_T)] - V(\theta_0)}{T} + \frac{108}{\lambda^2} [2(1+\lambda)^2 + \lambda^2] \frac{\sum_{t=0}^{T-1} \mathbb{E} \left[\|(w_t - w_{\theta_t}^*)\|_2^2 \right]}{T} \\
 &\quad + [2(1+\lambda)^2 + \lambda^2] \left(\frac{32}{\lambda^4(1-\gamma)^2} + \frac{432(1+2R_w)^2}{\lambda^2} \right) \frac{1+(\kappa-1)\xi}{(1-\xi)B} + \frac{216}{\lambda^2} [2(1+\lambda)^2 + \lambda^2] \Delta_{critic}.
 \end{aligned}$$

Proof. Proof is similar to first part of proof of Thm 6 in (Xu et al., 2020) and similar to Thm 5.4, along with using Prop J.4 and Lemmas J.3, J.2 and J.1. \square

We now move to proving the global optimality of natural actor critic based improper learner. Let $KL(\cdot, \cdot)$ be the KL-divergence between two distributions. We denote $D(\theta) := KL(\pi^*, \pi_{\theta})$, $u_{\theta_t}^{\lambda} := (F(\theta_t) + \lambda I)^{-1} \nabla_{\theta} V(\theta_t)$ and $u_{\theta_t}^{\dagger} :=$

$F(\theta_t)^\dagger \nabla_{\theta} V(\theta_t)$. We see that

$$\begin{aligned}
 & \mathbb{D}(\theta_t) - \mathbb{D}(\theta_{t+1}) \\
 &= \sum_{m=1}^M \pi^*(m) \log \pi_{\theta_{t+1}}(m) - \log \pi_{\theta_t}(m) \\
 &\stackrel{(i)}{=} \sum_{s \in \mathcal{S}} d_{\rho}^{\pi^*}(s) \sum_{m=1}^M \pi^*(m) \log \pi_{\theta_{t+1}}(m) - \log \pi_{\theta_t}(m) \\
 &= \mathbb{E}_{\nu_{\pi^*}} [\log \pi_{\theta_{t+1}}(m) - \log \pi_{\theta_t}(m)] \\
 &\stackrel{(ii)}{\geq} \mathbb{E}_{\nu_{\pi^*}} [\nabla_{\theta} \log(\pi_{\theta_t}(m))]^\top (\theta_{t+1} - \theta_t) - \frac{\|\theta_{t+1} - \theta_t\|_2^2}{2} \\
 &= \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (\theta_{t+1} - \theta_t) - \frac{\|\theta_{t+1} - \theta_t\|_2^2}{2} \\
 &= \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top u_t(w_t) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &= \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top u_{\theta_t}^\lambda + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &= \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &= \alpha \mathbb{E}_{\nu_{\pi^*}} [A_{\pi_{\theta_t}}(s, m)] + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m) + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) \\
 &+ \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &\stackrel{(iii)}{=} (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m) + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) \\
 &+ \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &\geq (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \alpha \sqrt{\mathbb{E}_{\nu_{\pi^*}} [(\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m))^2} + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) \\
 &+ \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &\stackrel{(iv)}{\geq} (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \sqrt{\left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_t}}} \right\|_{\infty}} \alpha \sqrt{\mathbb{E}_{\nu_{\pi^*}} [(\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m))^2} + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) \\
 &+ \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &\stackrel{(v)}{\geq} (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \sqrt{\frac{1}{1 - \gamma} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_{\infty}} \alpha \sqrt{\mathbb{E}_{\nu_{\pi^*}} [(\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m))^2} \\
 &+ \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_{\theta_t}^\lambda - u_{\theta_t}^\dagger) + \alpha \mathbb{E}_{\nu_{\pi^*}} [\psi_{\theta_t}(m)]^\top (u_t(w_t) - u_{\theta_t}^\lambda) - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 &\stackrel{(vi)}{\geq} (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \sqrt{\frac{1}{1 - \gamma} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_{\infty}} \alpha \sqrt{\mathbb{E}_{\nu_{\pi^*}} [(\psi_{\theta_t}(m)]^\top u_{\theta_t}^\dagger - A_{\pi_{\theta_t}}(s, m))^2} - \alpha C_{soft} \lambda \\
 &- 2\alpha \|u_t(w_t) - u_{\theta_t}^\lambda\|_2 - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2}.
 \end{aligned}$$

where (i) is by taking an extra expectation without changing the inner summand, (ii) follows by Lemma J.1 and Lemma 5 in (Xu et al., 2020), (iii) follows by the value difference lemma (Lemma I.4), (iv) follows by defining $\left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_t}}} \right\|_{\infty} := \max_{s, m} \frac{\nu_{\pi^*}(s, m)}{\nu_{\pi_{\theta_t}}(s, m)}$, (v) follows because $\nu_{\pi_{\theta_t}}(s, m) \geq (1 - \gamma)\nu_{\pi_{\theta_0}}(s, m)$, (vi) follows by Lemma 6 in (Xu et al., 2020) and

Lemma J.2.

Next, we denote $\Delta_{actor} := \max_{\theta \in \mathbb{R}^M} \min_{w \in \mathbb{R}^d} \mathbb{E}_{\nu_{\pi_\theta}} [\|\psi_\theta^\top w - A_{\pi_\theta}(s, m)\|^2]$ as the actor error.

$$\begin{aligned}
 & \mathbb{D}(\theta_t) - \mathbb{D}(\theta_{t+1}) \\
 & \geq (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \sqrt{\frac{1}{1 - \gamma} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_\infty} \alpha \sqrt{\Delta_{actor}} - \alpha C_{soft} \lambda \\
 & \quad - 2\alpha \|u_t(w_t) - u_{\theta_t}^\lambda\|_2 - \frac{\alpha^2 \|u_t(w_t)\|_2^2}{2} \\
 & \geq (1 - \gamma)(V(\pi^*) - V(\pi_{\theta_t})) - \sqrt{\frac{1}{1 - \gamma} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_\infty} \alpha \sqrt{\Delta_{actor}} - \alpha C_{soft} \lambda \\
 & \quad - 2\alpha \|u_t(w_t) - u_{\theta_t}^\lambda\|_2 - \frac{\alpha^2 \|u_t(w_t) - u^\lambda(\theta_t)\|_2^2}{2} - \frac{\alpha^2 \|\nabla_\theta V(\theta_t)\|_2^2}{\lambda^2}.
 \end{aligned}$$

Rearranging and dividing by $(1 - \gamma)\alpha$, and taking expectation both sides we get

$$\begin{aligned}
 & V(\pi^*) - \mathbb{E}[V(\pi_{\theta_t})] \\
 & \leq \frac{\mathbb{E}[\mathbb{D}(\theta_t)] - \mathbb{E}[\mathbb{D}(\theta_{t+1})]}{(1 - \gamma)\alpha} + \frac{2\sqrt{\mathbb{E}[\|u_t(w_t) - u_{\theta_t}^\lambda\|_2^2]}}{1 - \gamma} + \frac{\alpha \mathbb{E}[\|u_t(w_t) - u^\lambda(\theta_t)\|_2^2]}{2(1 - \gamma)} \\
 & \quad + \frac{\alpha}{\lambda^2(1 - \gamma)} \mathbb{E}[\|\nabla_\theta V(\theta_t)\|_2^2] + \sqrt{\frac{1}{(1 - \gamma)^3} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_\infty} \sqrt{\Delta_{actor}} + \frac{C_{soft} \lambda}{1 - \gamma}.
 \end{aligned}$$

Next we use the same argument as in eq (33) and Lemma 2 in Xu et al. (2020) to bound the second term.

$$\mathbb{E}[\|u_t(w_t) - u_{\theta_t}^\lambda\|_2^2] \leq \frac{C}{B} + \frac{108 \mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]}{\lambda^2} + \frac{216 \Delta_{critic}}{\lambda^2}.$$

where $C := \frac{18}{\lambda^2} \frac{24(1+2R_w)^2[1+(\kappa-1)\xi]}{B(1-\xi)} + \frac{4}{\lambda^4(1-\gamma)^2} \cdot \frac{8[1+(\kappa-1)\xi]}{(1-\xi)B}$. Using this in the bound and using $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for positive a, b above, we have,

$$\begin{aligned}
 & V(\pi^*) - \mathbb{E}[V(\pi_{\theta_t})] \\
 & \leq \frac{\mathbb{E}[\mathbb{D}(\theta_t)] - \mathbb{E}[\mathbb{D}(\theta_{t+1})]}{(1 - \gamma)\alpha} + \frac{2}{1 - \gamma} \left(\sqrt{\frac{C}{B}} + 11 \sqrt{\frac{\mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]}{\lambda^2}} + 15 \sqrt{\frac{\Delta_{critic}}{\lambda^2}} \right) \\
 & \quad + \frac{\alpha}{2(1 - \gamma)} \left(\frac{C}{B} + 108 \frac{\mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]}{\lambda^2} + 216 \frac{\Delta_{critic}}{\lambda^2} \right) \\
 & \quad + \frac{\alpha}{\lambda^2(1 - \gamma)} \mathbb{E}[\|\nabla_\theta V(\theta_t)\|_2^2] + \sqrt{\frac{1}{(1 - \gamma)^3} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_\infty} \sqrt{\Delta_{actor}} + \frac{C_{soft} \lambda}{1 - \gamma}.
 \end{aligned}$$

Summing over all $t = 0, 1, \dots, T-1$ and then dividing by T we get,

$$\begin{aligned}
 V(\pi^*) - \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[V(\pi_{\theta_t})] &\leq \frac{\mathbb{D}(\theta_0) - \mathbb{E}[\mathbb{D}(\theta_T)]}{(1-\gamma)\alpha T} + \frac{2}{(1-\gamma)} \left(\sqrt{\frac{C}{B}} + 15\sqrt{\frac{\Delta_{critic}}{\lambda^2}} \right) + \frac{22}{(1-\gamma)T} \sum_{t=0}^{T-1} \sqrt{\frac{\mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]}{\lambda^2}} \\
 &+ \frac{\alpha}{2(1-\gamma)} \left(\frac{C}{B} + 216\frac{\Delta_{critic}}{\lambda^2} \right) + \frac{54\alpha}{(1-\gamma)T} \sum_{t=0}^{T-1} \frac{\mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]}{\lambda^2} \\
 &+ \frac{\alpha}{\lambda^2(1-\gamma)T} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla_{\theta} V(\theta_t)\|_2^2] + \sqrt{\frac{1}{(1-\gamma)^3} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_{\infty}} \sqrt{\Delta_{actor}} + \frac{C_{soft}\lambda}{1-\gamma}.
 \end{aligned}$$

We now put the value of $\alpha \leq \frac{\lambda^2}{2\sqrt{M}L_V(1+\lambda)}$, we get,

$$\begin{aligned}
 V(\pi^*) - \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[V(\pi_{\theta_t})] &\leq C_1 \frac{\sqrt{M}}{T} + \frac{C_2}{\sqrt{B}} + C_3 \sqrt{\Delta_{critic}} + \frac{C_4}{T} \sum_{t=0}^{T-1} \sqrt{\mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2]} \\
 &+ \frac{C_5}{B} + C_6 \sqrt{\Delta_{critic}} + \frac{C_7}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|w_t - w_{\theta_t}^*\|_2^2] + \frac{C_8}{T} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla_{\theta} V(\theta_t)\|_2^2] \\
 &+ \sqrt{\frac{1}{(1-\gamma)^3} \left\| \frac{\nu_{\pi^*}}{\nu_{\pi_{\theta_0}}} \right\|_{\infty}} \sqrt{\Delta_{actor}} + C_9 \lambda.
 \end{aligned}$$

Letting $T = \mathcal{O}\left(\frac{\sqrt{M}}{(1-\gamma)^2 \varepsilon}\right)$, $B = \mathcal{O}\left(\frac{1}{(1-\gamma)^2 \varepsilon^2}\right)$ then $\mathbb{E}[\|\nabla_{\theta} V(\theta_t)\|_2^2] \leq \varepsilon^2$ and

$$V(\pi^*) - \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[V(\pi_{\theta_t})] \leq \varepsilon + \mathcal{O}\left(\sqrt{\frac{\Delta_{actor}}{(1-\gamma)^3}}\right) + \mathcal{O}(\Delta_{critic}) + \mathcal{O}(\lambda).$$

This leads to the total sample complexity as

$$(B + HT_c)T = \mathcal{O}\left(\left(\frac{1}{(1-\gamma)^2 \varepsilon^2} + \frac{\sqrt{M}}{\varepsilon^2} \log \frac{1}{\varepsilon}\right) \frac{\sqrt{M}}{(1-\gamma)^2 \varepsilon}\right) = \mathcal{O}\left(\frac{M}{(1-\gamma)^4 \varepsilon^3} \log \frac{1}{\varepsilon}\right).$$

□

K. Simulation Details

In this section we describe the details of the Sec. 6. Recall that since neither value functions nor value gradients are available in closed-form, we modify SoftMax PG (Algorithm 1) to make it generally implementable using a combination of (1) rollouts to estimate the value function of the current (improper) policy and (2) a stochastic approximation-based approach to estimate its value gradient.

The Softmax PG with Gradient Estimation or **SPGE** (Algorithm 5), and the gradient estimation algorithm 6, *GradEst*, are shown below.

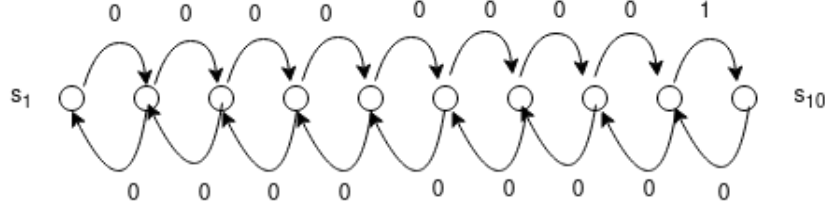


Figure 7: A chain MDP with 10 states.

Algorithm 5 Softmax PG with Gradient Estimation (SPGE)

- 1: **Input:** learning rate $\eta > 0$, perturbation parameter $\alpha > 0$, Initial state distribution μ
 - 2: Initialize each $\theta_m^1 = 1$, for all $m \in [M]$, $s_1 \sim \mu$
 - 3: **for** $t = 1$ **to** T **do**
 - 4: Choose controller $m_t \sim \pi_t$.
 - 5: Play action $a_t \sim K_{m_t}(s_t, \cdot)$.
 - 6: Observe $s_{t+1} \sim P(\cdot | s_t, a_t)$.
 - 7: $\nabla_{\theta^t} V^{\pi_{\theta^t}}(\mu) = \text{GradEst}(\theta_t, \alpha, \mu)$
 - 8: Update:
 $\theta^{t+1} = \theta^t + \eta \cdot \nabla_{\theta^t} V^{\pi_{\theta^t}}(\mu)$.
 - 9: **end for**
-

Algorithm 6 GradEst (subroutine for SPGE)

- 1: **Input:** Policy parameters θ , parameter $\alpha > 0$, Initial state distribution μ .
 - 2: **for** $i = 1$ **to** $\#\text{runs}$ **do**
 - 3: $u^i \sim \text{Unif}(S^{M-1})$.
 - 4: $\theta_\alpha = \theta + \alpha \cdot u^i$
 - 5: $\pi_\alpha = \text{softmax}(\theta_\alpha)$
 - 6: **for** $l = 1$ **to** $\#\text{rollouts}$ **do**
 - 7: Generate trajectory $(s_0, a_0, r_0, s_1, a_1, r_1, \dots, s_{1t}, a_{1t}, r_{1t})$ using the policy π_α : and $s_0 \sim \mu$.
 - 8: $\text{reward}^l = \sum_{j=0}^{1t} \gamma^j r_j$
 - 9: **end for**
 - 10: $\text{mr}(i) = \text{mean}(\text{reward})$
 - 11: **end for**
 - 12: $\text{GradValue} = \frac{1}{\#\text{runs}} \sum_{i=1}^{\#\text{runs}} \text{mr}(i) \cdot u^i \cdot \frac{M}{\alpha}$.
 - 13: **Return:** GradValue.
-

Next we report some extra simulations we performed under different environments.

K.1. State Dependent controllers – Chain MDP

We consider a linear chain MDP as shown in Figure 7. As evident from the figure, $|\mathcal{S}| = 10$ and the learner has only two actions available, which are $\mathcal{A} = \{\text{left}, \text{right}\}$. Hence the name ‘chain’. The numbers on the arrows represent the reward obtained with the transition. The initial state is s_1 . We let s_{100} as the terminal state. Let us define 2 base controllers, K_1 and K_2 , as follows.

$$K_1(\text{left} \mid s_j) = \begin{cases} 1, & j \in [9] \setminus \{5\} \\ 0.1, & j = 5 \\ 0, & j = 10. \end{cases}$$

$$K_2(\text{left} \mid s_j) = \begin{cases} 1, & j \in [9] \setminus \{6\} \\ 0.1, & j = 6 \\ 0, & j = 10. \end{cases}$$

and obviously $K_i(\text{right} \mid s_j) = 1 - K_i(\text{left} \mid s_j)$ for $i = 1, 2$. An improper mixture of the two controllers, i.e., $(K_1 + K_2)/2$ is the optimal in this case. We show that our policy gradient indeed converges to the ‘correct’ combination, see Figure 8. We here provide an elementary calculation of our claim that the mixture $K_{\text{mix}} := (K_1 + K_2)/2$ is indeed better than applying K_1 or K_2 for all time. We first analyze the value function due to K_i , $i = 1, 2$ (which are the same due to *symmetry* of the problem and the probability values described).

$$V^{K_i}(s_1) = \mathbb{E} \left[\sum_{t>0} \gamma^t r_t(a_t, s_t) \right]$$

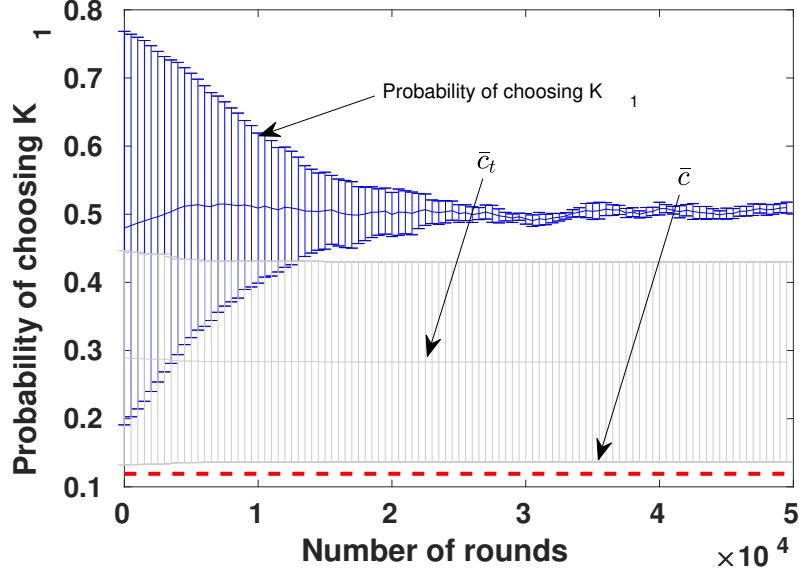


Figure 8: Softmax PG alg applied to the linear Chain MDP with various randomly chosen initial distribution. Plot shows probability of choosing controller K_1 averaged over #trials

$$\begin{aligned}
 &= 0.1 \times \gamma^9 + 0.1 \times 0.9 \times 0.1 \times \gamma^{11} + 0.1 \times 0.9 \times 0.1 \times 0.9 \times 0.1 \times \gamma^{13} \dots \\
 &= 0.1 \times \gamma^9 \left(1 + (0.1 \times 0.9 \gamma^2) + (0.1 \times 0.9 \gamma^2)^2 + \dots \right) = \frac{0.1 \times \gamma^9}{1 - 0.1 \times 0.9 \times \gamma^2}.
 \end{aligned}$$

We will next analyze the value if a true mixture controller i.e., K_{mix} is applied to the above MDP. The analysis is a little more intricate than the above. We make use of the following key observations, which are elementary but crucial.

1. Let Paths be the set of all sequence of states starting from s_1 , which terminate at s_{10} which can be generated under the policy K_{mix} . Observe that

$$V^{K_{\text{mix}}}(s_1) = \sum_{p \in \text{Paths}} \gamma^{\text{length}(p)} P[p] .1. \quad (20)$$

Recall that reward obtained from the transition $s_9 \rightarrow s_{10}$ is 1.

2. Number of distinct paths with exactly n loops: 2^n .
3. Probability of each such distinct path with n cycles:

$$\begin{aligned}
 &= \underbrace{(0.55 \times 0.45) \times (0.55 \times 0.45) \times \dots \times (0.55 \times 0.45)}_{n \text{ times}} \times 0.55 \times 0.55 \times \gamma^{9+2n} \\
 &= (0.55)^2 \times \gamma^9 (0.55 \times 0.45 \times \gamma^2)^n.
 \end{aligned}$$

4. Finally, we put everything together to get:

$$\begin{aligned}
 V^{K_{\text{mix}}}(s_1) &= \sum_{n=0}^{\infty} 2^n \times (0.55)^2 \times \gamma^9 \times (0.55 \times 0.45 \times \gamma^2)^n \\
 &= \frac{(0.55)^2 \times \gamma^9}{1 - 2 \times 0.55 \times 0.45 \times \gamma^2} > V^{K_i}(s_1).
 \end{aligned}$$

This shows that a mixture performs better than the constituent controllers. The plot shown in Fig. 8 shows the Softmax PG algorithm (even with estimated gradients and value functions) converges to a (0.5,0.5) mixture correctly.

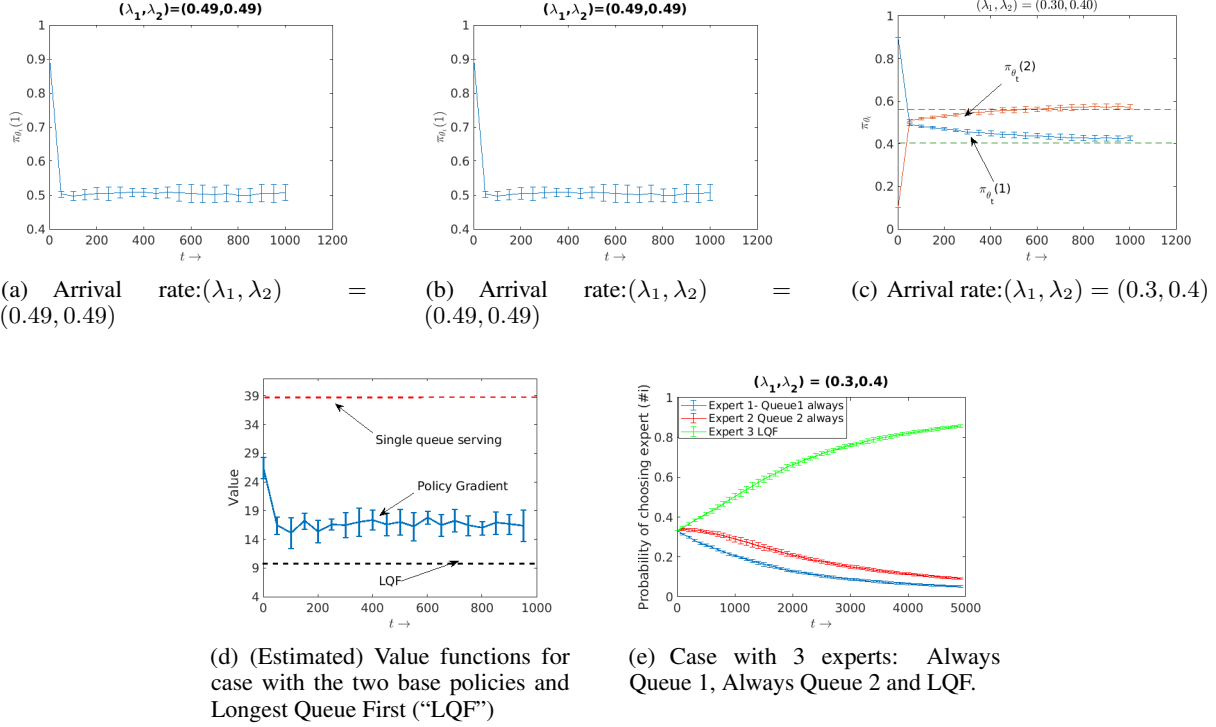


Figure 9: Softmax policy gradient algorithm applies show convergence to the best mixture policy.

K.2. Stationary Bernoulli Queues

We study two different settings (1) where in the first case the optimal policy is a strict improper combination of the available controllers and (2) where it is at a corner point, i.e., one of the available controllers itself is optimal. Our simulations show that in both the cases, PG converges to the correct controller distribution.

Recall the example that we discussed in Sec. 2.2. We consider the case with Bernoulli arrivals with rates $\lambda = [\lambda_1, \lambda_2]$ and are given two base/atomic controllers $\{K_1, K_2\}$, where controller K_i serves Queue i with probability 1, $i = 1, 2$. As can be seen in Fig. 9(b) when $\lambda = [0.49, 0.49]$ (equal arrival rates), GradEst converges to an improper mixture policy that serves each queue with probability $[0.5, 0.5]$. Note that this strategy will also stabilize the system whereas both the base controllers lead to instability (the queue length of the unserved queue would obviously increase without bound). Figure 9(c), shows that with unequal arrival rates too, GradEst quickly converges to the best policy.

Fig. 9(d) shows the evolution of the value function of GradEst (in blue) compared with those of the base controllers (red) and the *Longest Queue First* (LQF) which, as the name suggests, always serves the longest queue in the system (black). LQF, like any policy that always serves a nonempty queue in the system whenever there is one³, is known to be optimal in the sense of delay minimization for this system (Mohan et al., 2016). See Sec. K in the Appendix for more details about this experiment.

Finally, Fig. 9(e) shows the result of the second experimental setting with three base controllers, one of which is delay optimal. The first two are K_1, K_2 as before and the third controller, K_3 , is LQF. Notice that K_1, K_2 are both queue length-agnostic, meaning they could attempt to serve empty queues as well. LQF, on the other hand, always and only serves nonempty queues. Hence, in this case the optimal policy is attained at one of the corner points, i.e., $[0, 0, 1]$. The plot shows the PG algorithm converging to the correct point on the simplex.

Here, we justify the value of the two policies which always follow one fixed queue, that is plotted as straight line in Figure 9(d). Let us find the value of the policy which always serves queue 1. The calculation for the other expert (serving queue 2 only) is similar. Let $q_i(t)$ denote the length of queue i at time t . We note that since the expert (policy) always recommends

³Tie-breaking rule is irrelevant.

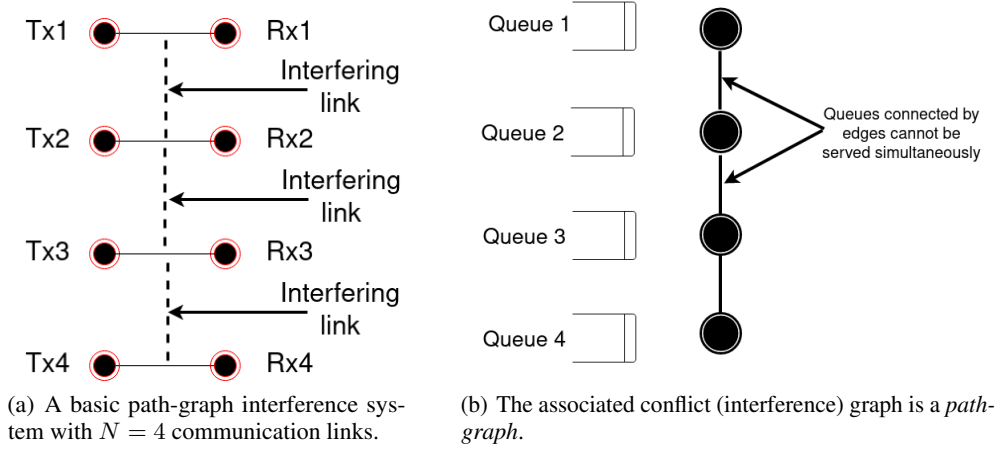


Figure 10: An example of a path graph network. The interference constraints are such that physically adjacent queues cannot be served simultaneously.

to serve one of the queue, the expected *cost* suffered in any round t is $c_t = q_1(t) + q_2(t) = 0 + t \cdot \lambda_2$. Let us start with empty queues at $t = 0$.

$$\begin{aligned}
 V^{Expert1}(\mathbf{0}) &= \mathbb{E} \left[\sum_{t=0}^T \gamma^t c_t \mid Expert1 \right] \\
 &= \sum_{t=0}^T \gamma^t \cdot t \cdot \lambda_2 \\
 &\leq \lambda_2 \cdot \frac{\gamma}{(1-\gamma)^2}.
 \end{aligned}$$

With the values, $\gamma = 0.9$ and $\lambda_2 = 0.49$, we get $V^{Expert1}(\mathbf{0}) \leq 44$, which is in good agreement with the bound shown in the figure.

K.3. Details of Path (Interference) Graph Networks

Consider a system of parallel transmitter-receiver pairs as shown in Figure 10(a). Due to the physical arrangement of the Tx-Rx pairs, no two adjacent systems can be served simultaneously because of interference. This type of communication system is commonly referred to as a *path graph network* (Mohan et al., 2020). Figure 10(b) shows the corresponding *conflict graph*. Each Tx-Rx pair can be thought of as a queue, and the edges between them represent that the two connecting queues, cannot be served simultaneously. On the other hand, the sets of queues which can be served simultaneously are called *independent sets* in the queuing theory literature. In the figure above, the independent sets are $\{\emptyset, \{1\}, \{2\}, \{3\}, \{4\}, \{1, 3\}, \{2, 4\}, \{1, 4\}\}$.

Finally, in Table 2, we report the mean delay values of the 5 base controllers we used in our simulation Fig. 2(c), Sec.6. We see the controller K_2 which was chosen to be MER, indeed has the lowest cost associated, and as shown in Fig. 2(c), our Softmax PG algorithm (with estimated value functions and gradients) converges to it.

Table 2: Mean Packet Delay Values of Path Graph Network Simulation.

Controller	Mean delay (# time slots) over 200 trials	Standard deviation
$K_1(MW)$	22.11	0.63
$K_2(MER)$	20.96	0.65
$K_3(\{1, 3\})$	80.10	0.92
$K_4(\{2, 4\})$	80.22	0.90
$K_5(\{1, 4\})$	80.13	0.91

K.4. Cartpole Experiments

We investigate further the example in our simulation in which the two constituent controllers are $K_{opt} + \Delta$ and $K_{opt} - \Delta$. We use OpenAI gym to simulate this situation. In the Figure 2(b), it was shown our Softmax PG algorithm (with estimated values and gradients) converged to a improper mixture of the two controllers, i.e., $\approx (0.53, 0.47)$. Let K_{conv} be defined as the (randomized) controller which chooses K_1 with probability 0.53, and K_2 with probability 0.47. Recall from Sec. 2.1 that this control law converts the linearized cartpole into an Ergodic Parameter Linear System (EPLS). In Table 3 we report the average number of rounds the pendulum stays upright when different controllers are applied for all time, over trajectories of length 500 rounds. The third column displays an interesting feature of our algorithm. Over 100 trials, the base controllers do not stabilize the pendulum for a relatively large number of trials, however, K_{conv} successfully does so most of the times.

Table 3: A table showing the number of rounds the constituent controllers manage to keep the cartpole upright.

Controller	Mean number of rounds before the pendulum falls \wedge 500	# Trials out of 100 in which the pendulum falls before 500 rounds
$K_1(K_{opt} + \Delta)$	403	38
$K_2(K_{opt} - \Delta)$	355	46
K_{conv}	465	8

We mention here that if one follows K^* , which is the optimum controller matrix one obtains by solving the standard Discrete-time Algebraic Riccati Equation (DARE) (Bertsekas, 2011), the pole does not fall over 100 trials. However, as indicated in Sec.1, constructing the optimum controller for this system from scratch requires exponential, in the number of state dimension, sample complexity (Chen & Hazan, 2020). On the other hand K_{conv} performs very close to the optimum, while being sample efficient.

Choice of hyperparameters. In the simulations, we set learning rate to be 10^{-4} , $\#runs = 10$, $\#rollouts = 10$, $lt = 30$, discount factor $\gamma = 0.9$ and $\alpha = 1/\sqrt{\#runs}$. All the simulations have been run for 20 trials and the results shown are averaged over them. We capped the queue sizes at 1000.

K.5. Some extra simulations for natural-actor-critic based improper learner NACIL

- First we show a queuing theory where we have 2 queues to be served and we have two base controllers similar to as we discussed in the Sec 2. However, here we have two different arrival rates for the two queues $(\lambda_1, \lambda_2) \equiv (0.4, 0.3)$, i.e., the arrival rates are unequal. We plot in Fig. 11 the probability of choosing the two different controllers. We see that ACIL converges to the “correct” mixture of the base controllers.
- Next, we show a simulation on the setting in Sec. K.1, which we called a Chain MDP. We recall that this setting consists of two base controllers K_1 and K_2 , however a $(1/2, 1/2)$ mixture of the two controllers was shown (analytically) to perform better than each individual ones. As the plot in Fig. 12 shows NACIL identifies the correct combination and follows it.

Choice of hyperparameters. For the queuing theoretic simulations of Algorithm 2 ACIL, we choose $\alpha = 10^{-4}$, $\beta = 10^{-3}$. We choose the identity mapping $\varphi(s) \equiv s$, where s is the current state of the system which is a N -length vector, which consists of the i^{th} queue length at the i^{th} position. λ was chosen to be 0.1. The other parameters are chosen as $B = 50$, $H = 30$ and $T_c = 20$. We choose a buffer of size 1000 to keep the states bounded, i.e., if a queue exceeds a size 1000, those arrivals are ignored and queue length is not increased. This is used to normalize the $\|\varphi(s)\|_2$ across time.

L. Additional Comments

- *Comment on the ‘simple’ experimental settings.* The motivating examples may seem “simple” and trainable from scratch with respect to progress in the field of RL. However, our main point is that there are situations where, for example, one may have trained controllers for a range of environments *in simulation*. However, the real life environment may differ from the simulated ones. We demonstrate that exploiting such basic pre-learnt controllers via our approach can help in generating a better (meta) controller for a new, unseen environment, instead of learning a new controller for the new environment from scratch.

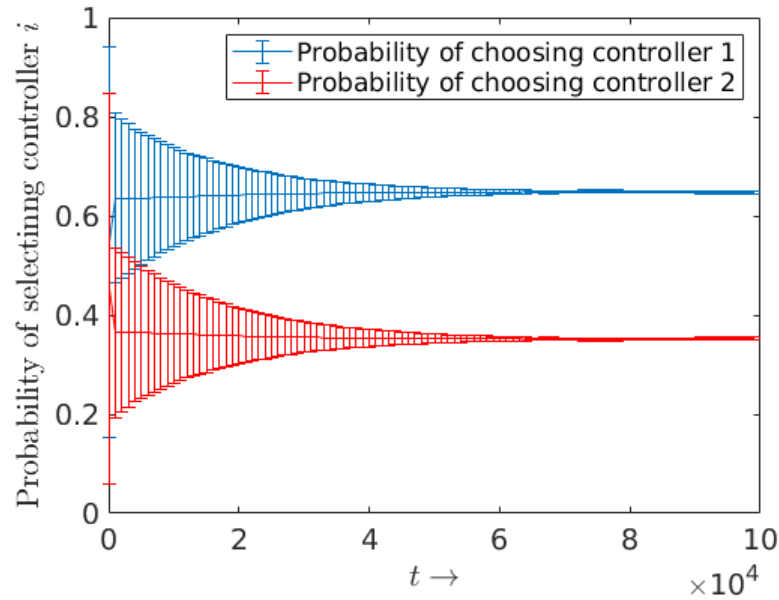


Figure 11: NACIL alg applied to the a queuing system with two queues, having arrival rates $(\lambda_1, \lambda_2) \equiv (0.4, 0.3)$. Plot shows probability of choosing controllers K_1 and K_2 averaged over 20 trials

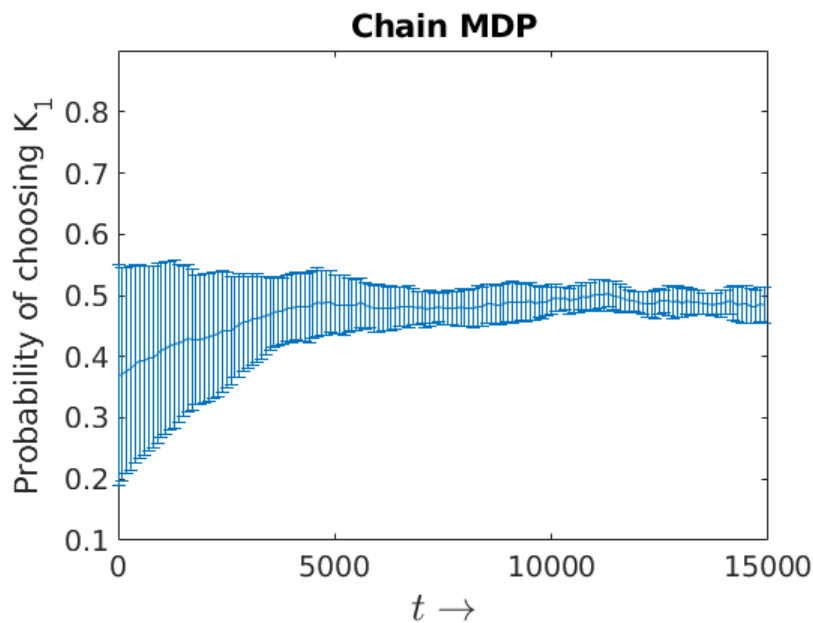


Figure 12: NACIL alg applied to the linear Chain MDP with various randomly chosen initial distribution. Plot shows probability of choosing controller K_1 averaged over 20 trials

- *On characterizing the performance of the optimal mixture policy.* As correctly noticed by the reviewer, the inverted pendulum experiment showed that the optimal mixture policy can vastly outperform the component controllers. Currently, however, we do not provide any theoretical guarantees regarding this, since this depends on the structure of the policy space and the underlying MDP, which is very challenging. We hope to explore this task in our future work.

M. Discussion

We have considered the problem of using a menu of baseline controllers and combining them using improper probabilistic mixtures to form a superior controller. In many relevant MDP learning settings, we saw that this is indeed possible, and the policy gradient and actor-critic based analyses indicate that this approach may be widely applicable. This work opens up a plethora of avenues. One can consider a richer class of mixtures that can look at the current state and mix accordingly. For example, an attention model can be used to choose which controller to use, or other state-dependent models can be relevant. Another example is to artificially force switching across controllers to occur less frequently than in every round. This can help create *momentum* and allow the controlled process to 'mix' better, when using complex controllers.

A few caveats are in order regarding the potential societal impact and consequences of this work. As such, this paper offers a way of combining or 'blending' a given class of decision-making entities in the hope of producing a 'better' one. In this process, the definitions of what constitutes 'optimal' or 'expected' behavior from a policy are likely to be subjective, and may encode biases and attitudes of the system designer(s). More importantly, it is possible that the base policy class (or some elements of it) have undesirable properties to begin with (e.g., bias or insensitivity), which could get amplified in the improper learning process as an unintended outcome. We sound ample caution to practitioners who contemplate adopting this method.

Finally, in the present setting, the base controllers are fixed. It would be interesting to consider adding adaptive, or 'learning' controllers as well as the fixed ones. Including the base controllers can provide baseline performance below which the performance of the learning controllers would not drop.