Finding Near Optimal Policies via Reducive Regularization in Markov Decision Processes

Wenhao Yang [1] **Xiang Li** [2] **Guangzeng Xie** [1] **Zhihua Zhang** [2]

Abstract

Regularized Markov Decision processes (MDPs) serve as a smooth version of ordinary MDPs to encourage exploration. Given a regularized MDP, however, the optimal policy is often biased when evaluating the value function. Rather than making the coefficient λ of regularized term sufficiently small, we propose a scheme by reducing λ to approximate the optimal policy of the original MDP. We prove that the iteration complexity to obtain an ε-optimal policy could be maintained or even reduced in comparison with setting a sufficiently small λ in both dynamic programming and policy gradient methods. In addition, there exists a strong duality connection between the reduction method and solving the original MDP directly, from which we can derive more adaptive reduction methods for certain reinforcement learning algorithms.

1. Introduction

Reinforcement learning (RL) has achieved great empirical success, especially when policy and value functions are parameterized by neural networks. Many studies have shown the powerful and striking performance of RL compared to human-level performance (Mnih et al., 2015; Schulman et al., 2015; Silver et al., 2016; Haarnoja et al., 2018). In these studies, dynamic programming (Puterman and Shin, 1978; Scherrer et al., 2015; Geist et al., 2019; Azar et al., 2012) and policy gradient methods (Williams, 1992; Sutton et al., 2000; Kakade, 2002) are the most frequently used optimization tools. However, when policy gradient methods are applied, theoretically understanding the success of RL is still limited in the case that policy is searched either on a simplex or a parameterized space. There is a line of recent work (Bhandari and Russo, 2020; Agarwal et al., 2019; Bhandari and Russo, 2019) on convergence performance of policy gradient methods for MDPs without parameterization, while another line of recent work (Mei et al., 2020; Cen et al., 2020; Wang et al., 2019; Dai et al., 2018) focuses on MDPs with parameterization.

In addition, during the process of learning MDPs, it is often observed that the obtained policy becomes deterministic while the environment has not been fully explored yet. Some prior works (Ahmed et al., 2019; Mnih et al., 2016; Fox et al., 2015; Vamplew et al., 2017) propose to impose the Shannon entropy to the reward, making the policy stochastic again. Therefore, the agent can explore the environment all the time rather than trap in a sub-optimal regime and fail. Intuitively and empirically speaking, adding entropy regularization helps soften the learning process and encourage agents to explore more, so it is expected to fasten convergence. Alternatively, Lee et al. (2018) proposed the Tsallis entropy (Tsallis, 1988) as the regularization function while Yang et al. (2019) presented a more general regularized term and studied the asymptotic properties of the optimal regularized policy. Except for the choice of regularization functions, there are some studies focusing on solving regularized MDPs with fast algorithms. Geist et al. (2019); Fox et al. (2015); Schulman et al. (2017) showed that the convergence rate could be linear for dynamic programming methods, while Shani et al. (2019); Mei et al. (2020); Cen et al. (2020) provided convergence rate for policy gradient methods. We will discuss them detailedly in related work.

However, regularized MDPs always incur biased optimal policies because the imposed regularization twists the original optimal policy. The degree of bias is controlled by the regularization coefficient (also called temperature parameter) λ. When λ approaches zero, the regularized optimal policy converges to the unregularized optimal policy. However, existing results (Shani et al., 2019; Mei et al., 2020) of regularized MDPs require the regularization coefficient can not be too small ($\lambda = \Theta(1)$ is required indeed). Otherwise, the convergence rate could be deteriorated.

In the work, we ask the following question: can we design an approach to reduce the temperature such that the output policy is an ε-optimal policy with respect to the original MDP and the convergence speed is maintained or even faster? In this paper we give affirmative answers to the ques-

[1]Academy for Advanced Interdisciplinary Studies, Peking University [2]School of Mathematical Sciences, Peking University. Correspondence to: Wenhao Yang <yangwenhaosms@pku.edu.cn>.

Workshop On Reinforcement Learning Theory at ICML 2021.

tion. We summary our major contributions as follows and leave related work discussion in Appendix A.

- We propose a reduction scheme to tune the temperature parameter λ. Specifically, given the current λ_t, we resort to a sub-solver algorithm to solve the optimal policy in a λ_t-regularized MDP and then half λ_t to $\lambda_{t+1} := \lambda_t/2$. We need only to solve the λ_t-regularized MDP within certain accuracy (Definition 3.1) rather than finding the regularized optimal policy exactly. Our reduction method allows us to use almost any regularized RL algorithms and thus is flexible.
- We show that our reduction scheme would not increase iteration complexity than simply setting λ sufficiently small (it is often the case that $\lambda = O((1-\gamma)\varepsilon)$. For dynamic programming methods, we show that our approach maintains the convergence rate compared with the unregularized MDP and the case of setting λ sufficiently small. For projected gradient ascent, we improve previous analysis (Shani et al., 2019) from $\widetilde{O}(\frac{|\mathcal{S}||\mathcal{A}|\rho^2\rho_\nu^2}{\epsilon^2(1-\gamma)^4})$ to $\widetilde{O}(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2}{\epsilon(1-\gamma)^3})$ when full information of MDP is acquired (i.e., exact policy evaluation is accessed). Furthermore, we also promote our results to an approximation version, where only a ν-restart model can be accessed. With our reduction scheme, the sample complexity (the number of trajectories) can be improved from $\widetilde{O}(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^6\rho_\nu^6}{\varepsilon^6(1-\gamma)^8})$ to $\widetilde{O}(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^5\rho_\nu^6}{\varepsilon^5(1-\gamma)^7})$.
- We reveal that our reduction scheme is a dual approach for solving the unregularized MDP. Thus, we can derive more efficient adaptive reduction schemes from the dual perspective. Although we conjecture that the order of $\frac{1}{\varepsilon}$ is the best we would hope, we can still design efficient learning algorithms to reduce dependence on other terms.

2. Preliminaries and Notation

Markov Decision Processes. An infinite-horizon MDP is defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \mu, \gamma)$, where $\mathcal{S}$ is the state space and $\mathcal{A}$ the action space, both assumed to be finite with respective sizes S and A. Here $r\colon \mathcal{S}\times\mathcal{A}\to[0,R]$ is the bounded reward function. Let $\Delta(\mathcal{X})$ denote the set of probabilities on $\mathcal{X}$, that is, $\Delta(\mathcal{X}) = \{P\colon \sum_{x\in\mathcal{X}} P(x) = 1, P(x)\geq 0\}$. Then $P\colon \mathcal{S}\times\mathcal{A}\to\Delta(\mathcal{S})$ is the unknown transition probability distribution and $\mu\in\Delta(\mathcal{S})$ is the initial state distribution. $\gamma\in[0,1)$ is the discount factor. Let $V^\pi\in\mathbb{R}^S$ be the value of a policy π with its $s\in\mathcal{S}$ entry given by $V^\pi(s) := \mathbb{E}_{\tau\sim\pi}[\sum_{t=0}^\infty \gamma^t r(s_t,a_t)|s_0=s]$, where $\tau\sim\pi$ means the trajectory $\tau=(a_0,s_1,a_1,s_2,a_2,\cdots)$ is generated according to the policy π. It is known that $V^\pi = (I-\gamma P^\pi)^{-1}r^\pi$ where $[P^\pi]_{s,s'} := \mathbb{E}_{a\sim\pi(\cdot|s)}P(s'|s,a)$ and $r^\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}r(s,a)$. For a given initial distribution μ on s_0, we set $V^\pi(\mu) := \mathbb{E}_{s_0\sim\mu}V^\pi(s)$.

Regularized Markov decision processes. Given any convex function $\Omega\colon \Delta(\mathcal{A})\to\mathbb{R}$, for any policy π and state $s\in\mathcal{S}$, we abuse the notation a little bit and denote by $\Omega(\pi,s) := \Omega(\pi(\cdot|s))$ for simplicity. A regularized MDP can be described by a tuple $(\mathcal{S},\mathcal{A},P,r,\mu,\gamma,\lambda,\Omega)$, for simplicity, which we denote by $\mathcal{M}(\lambda)$ (Geist et al., 2019). We similarly denote the value function of π in $\mathcal{M}(\lambda)$ by $V_\lambda^\pi\in\mathbb{R}^S$ with its $s\in\mathcal{S}$ entry given by

$$V_\lambda^\pi(s) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty \gamma^t(r(s_t,a_t)-\lambda\Omega(\pi,s_t))\Big|s_0=s\right]. \tag{1}$$

Sometimes it is convenient to focus on the effect of regularization alone, so we define $\Phi^\pi\in\mathbb{R}^S$ with its $s\in\mathcal{S}$ entry given by

$$\Phi^\pi(s) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty \gamma^t\Omega(\pi,s_t)\Big|s_0=s\right]. \tag{2}$$

Typically, in $\mathcal{M}(\lambda)$, we consider the sum of discounted rewards with regularization as follows:

$$\begin{aligned} J(\pi,\lambda) &= \mathbb{E}_{s_0\sim\mu}\mathbb{E}_{\tau\sim\pi}\sum_{t=0}^\infty \gamma^t\left(r(s_t,a_t)-\lambda\Omega(\pi,s_t)\right) \\ &= V^\pi(\mu) - \lambda\Phi^\pi(\mu), \end{aligned} \tag{3}$$

where λ is the regularization coefficient (or called temperature) and $V^\pi(\mu) = \mathbb{E}_{s_0\sim\mu}V^\pi(s_0)$, $\Phi^\pi(\mu) = \mathbb{E}_{s_0\sim\mu}\Phi^\pi(s_0)$.

For a given $\lambda\geq 0$, the optimal policy is $\pi_\lambda^* := \operatorname{argmax}_\pi J(\pi,\lambda) = \operatorname{argmax}_\pi V_\lambda^\pi(\mu)$ with the optimal value function as $V_\lambda^* := V_\lambda^{\pi_\lambda^*}$. Prior work shows that it is sometimes more efficient to maximize $J(\pi,\lambda)$ and obtain $\pi_\lambda^* = \operatorname{argmax}_\pi J(\pi,\lambda)$ (Shani et al., 2019). By contrast, unregularized $\mathcal{M}$ takes $J(\pi) := J(\pi,0) = V^\pi(\mu)$. We refer to a policy π as an ε-optimal policy if $V_\lambda^*(\mu) - V_\lambda^\pi(\mu)\leq\varepsilon$.

Discounted state visitation distribution of a policy π is defined as $d_{s_0}^\pi(s) := (1-\gamma)\sum_{t=0}^\infty\gamma^t P^\pi(s_t=s|s_0)$, where $P^\pi(s_t=s|s_0)$ is the state visitation probability by executing π with initial state s_0. Again, we overload notation and write: $d_\mu^\pi(s) = \mathbb{E}_{s_0\sim\mu}d_{s_0}^\pi(s)$ and concatenate it as a row vector $d_\mu^\pi\in\mathbb{R}^S$. Interestingly, we have $d_\mu^\pi = (1-\gamma)\mu(I-\lambda P^\pi)^{-1}$. It is often the case that we use another distribution (say, ν) as the initial state distribution used in a RL algorithm, but still use μ to measure the sub-optimality of our policies.

Bregman Divergence. Given any strictly convex and continuously differentiable function $\Omega\colon\Delta(\mathcal{A})\to\mathbb{R}$, for any two policies $\pi,\pi'\in\Delta(\mathcal{A})^S$ and $s\in\mathcal{S}$, the Bregman divergence between π,π' at s is defined as

$D_\Omega(\pi'||\pi)(s) := D_\Omega(\pi'(\cdot|s)\|\pi(\cdot|s)) = \Omega(\pi'(\cdot|s)) - \Omega(\pi(\cdot|s)) - \langle\nabla_{\pi(\cdot|s)}\Omega(\pi(\cdot|s)), \pi' - \pi\rangle$. For simplicity, we let $D_\Omega(\pi'||\pi) \in \mathbb{R}^S$ with its $s \in \mathcal{S}$ entry given by $D_\Omega(\pi'||\pi)(s)$. In this paper, we always argue Assumption 2.1 holds.

Assumption 2.1 (Bounded convex regularization)**.** *Assume regularization function $\Omega\colon \Delta(\mathcal{A}) \to \mathbb{R}^- \cup \{0\}$ is (i) 1-strongly convex w.r.t. norm $\|\cdot\|$, that is, for any two policies π', π,*

$$\Omega(\pi') \geq \Omega(\pi) + \langle\nabla\Omega(\pi), \pi' - \pi\rangle. + \frac{1}{2}\|\pi' - \pi\|^2,$$

and (ii) uniformly bounded, that is, $\max_{\pi\in\Delta(\mathcal{A})} |\Omega(\pi)| \leq C_\Phi$.

Performance of optimal regularized policy. Given a fixed λ, the performance of optimal regularized policy π^*_λ is guaranteed by the following proposition.

Proposition 2.1. *Denote $\pi^*_\lambda \in \operatorname{argmax}_\pi J(\pi, \lambda)$ and $V^*(s) = \max_\pi V^\pi$. Then*

$$\|V^* - V^{\pi^*_\lambda}\|_\infty \leq \frac{\lambda}{1-\gamma} C_\Phi,$$

Controlling the bias. The regularized MDP often has a biased optimal policy as shown in Proposition 2.1. In order to find an ε-optimal policy w.r.t. the original unregularized MDP, many prior works propose to fix a sufficiently small $\lambda = O((1-\gamma)\varepsilon/C_\Phi)$. However, in some cases, the convergence rate could be much slower than that when $\lambda = \Theta(1)$. Thus we propose an adaptive reduction scheme to help control the bias in the next section.

3. Methodology

Algorithm 1 AdaptReduce

Input: T: the number of epochs, λ_0: an initial regularization parameter, an algorithm Alg that tries to produce the optimal policy of $J(\pi, \lambda)$ for a given λ.
for iteration $t = 0$ **to** $T - 1$ **do**
  $\hat{\pi}_{t+1} \leftarrow \mathsf{Alg}(\hat{\pi}_t, \lambda_t)$;
  $\lambda_{t+1} \leftarrow \frac{\lambda_t}{2}$;
end for
Return: $\hat{\pi}_T$

In this section we propose AdaptReduce (Algorithm 1) to control the bias from λ. AdaptReduce works in the following way. At the beginning of AdaptReduce, we set $\hat{\pi}_0$ as any given initial policy. At each iteration $t = 0, 1, \cdots, T-1$, we first focus on finding the optimal policy in the regularized MDP $\mathcal{M}(\lambda_t)$ and then update the value of regularization coefficient λ_t. Specifically, we run Alg with starting policy $\hat{\pi}_t$ in each iteration, and let the output be $\hat{\pi}_{t+1}$. After all T iterations are finished, AdaptReduce simply outputs $\hat{\pi}_T$. Here we do not aim to solve out the optimal policy in $\mathcal{M}(\lambda_t)$ exactly, because our target is to find the original optimal policy π^* and the reason we prefer to optimize in a regularized MDP is the benefit (like faster convergence (Shani et al., 2019) and better exploration) it would offer.

3.1. Convergence Analysis

Definition 3.1. *We say an algorithm $\mathsf{Alg}(\pi_0, \lambda, \varepsilon, \hat{\varepsilon})$ maximizing $J(\pi, \lambda)$ satisfies approximate convergence with given accuracy ε property in time $\mathsf{Time}(\lambda)$ if, for every starting policy π_0, it produces $\pi_1 \leftarrow \mathsf{Alg}(\pi_0, \lambda, \varepsilon, \hat{\varepsilon})$ such that $V^*_\lambda - V^{\pi_1}_\lambda \leq \varepsilon + \hat{\varepsilon}$.*

In this paper, we consider two types of $\mathsf{Alg}(\pi_0, \lambda, \varepsilon, \hat{\varepsilon})$: (i) $\varepsilon = \frac{1}{4}(V^*_\lambda(\mu) - V^{\pi_0}_\lambda(\mu))$; (ii) At timestep t, $\mathsf{Alg}(\pi_t, \lambda_t, \varepsilon_t, \hat{\varepsilon})$ outputs π_{t+1} with $\varepsilon_t = \frac{\lambda_0}{2^t}\frac{C_\Phi}{1-\gamma}$. We denote them by Prop-I($\hat{\varepsilon}$) and Prop-II($\hat{\varepsilon}$), respectively. The requirement of Prop-I($\hat{\varepsilon}$) is mainly twofold: (a) the first part is homogeneous contraction convergence; (b) the second is the relaxation for the closeness, an approximation error no larger than $\hat{\varepsilon}$ allowed. However, Prop-I($\hat{\varepsilon}$) means Alg is able to contract the initial error by a fixed factor in a given time, which, however, is barely met for accurate controlling. In the case, it is better to require Alg obtaining π^*_λ within an absolute error rather than contracting initial errors. Thus, we are motivated to propose another type of stopping criteria (Prop-II($\hat{\varepsilon}$)) to deal with some sub-solvers that cannot satisfy Prop-I($\hat{\varepsilon}$) property.

Remark 3.1. *In Definition 3.1, we give two types of errors ε and $\hat{\varepsilon}$. In this paper, ε denotes the desired accuracy we would like to achieve, while $\hat{\varepsilon}$ denotes the computational and statistical error evolving from sub-solver. Besides, $\mathsf{Time}(\lambda)$ denotes the time Alg needs to produce an output policy satisfying the Prop-I($\hat{\varepsilon}$) or Prop-II($\hat{\varepsilon}$) property.*

When the sub-solver Alg satisfies either Prop-I($\hat{\varepsilon}$) or Prop-II($\hat{\varepsilon}$), it is guaranteed that the convergence rate is linear w.r.t. the number of iterations in Algorithm 1, which is shown in Theorems 3.1 and 3.2.

Theorem 3.1. *Let Assumption 2.1 hold, λ_0 be the initial regularization parameter, and $D_0 = V^*_{\lambda_0}(\mu) - V^{\hat{\pi}_0}_{\lambda_0}(\mu)$. Then for any T, AdaptReduce with any solver algorithm satisfying the Prop-I($\hat{\varepsilon}$) property gives such a policy $\hat{\pi}_T$ that*

$$V^*(\mu) - V^{\hat{\pi}_T}(\mu) \leq \frac{D_0}{4^T} + \frac{4}{3}\hat{\varepsilon} + \frac{6\lambda_0 C_\Phi}{1-\gamma}\frac{1}{2^T}.$$

Theorem 3.2. *Let Assumption 2.1 hold, λ_0 be the initial regularization parameter, and $D_0 = V^*_{\lambda_0}(\mu) - V^{\hat{\pi}_0}_{\lambda_0}(\mu)$. Then for any T, AdaptReduce with any solver algorithm satisfy-*

ing Prop-II($\hat{\varepsilon}$) *yields such a policy* $\hat{\pi}_T$ *that*

$$V^*(\mu) - V^{\hat{\pi}_T}(\mu) \leq \frac{6\lambda_0}{2^T}\frac{C_\Phi}{1-\gamma} + \hat{\varepsilon}.$$

3.2. Case Studies for Alg

Recall that Algorithm 1 needs another RL algorithm Alg as the sub-solver. In this part, we consider two popular choices of Alg and show that the convergence rate would not be increased. We firstly consider the simple case when dynamic programming method is applied, which satisfies Prop-I($\hat{\varepsilon}$), and we show that the convergence rate is maintained. To save the space, we leave this case in Appendix B.2. Then we apply projected policy gradient, which satisifes Prop-II($\hat{\varepsilon}$), and we show that the convergence rate is improved. We also discuss other special algorithms (Shani et al., 2019; Cen et al., 2020) in Appendix B.5.

3.2.1. Projected Gradient Ascent

Similar with Agarwal et al. (2019), we consider Projected Gradient Ascent (PGA) in Algorithm 2 to solve regularized MDPs. Before we provide detailed convergence results, we have to clarify the policy optimization oracle. Even though we would like to obtain a near optimal policy w.r.t. initial distribution μ, we always only own access with ν-restart model (Shani et al., 2019; Kakade and Langford, 2002). Fortunately, for tabular MDPs, the optimal policy π^*_λ is simultaneously optimal for all starting s (Bellman and Dreyfus, 1959; Yang et al., 2019). Thus, we have Theorem 3.3 to control value difference under different initial distributions. To simplify, we always assume $\min_s \nu(s) > 0$ while $\mu \ll \nu$ is already enough.

Algorithm 2 Projected Gradient Ascent

Input: an initial policy π_0, a regularization parameter λ, and T the number of iterations.
for iteration $t = 0$ **to** $T-1$ **do**
$\pi_{t+1} = \operatorname{argmin}_{\pi\in\Delta(\mathcal{A})^{\mathcal{S}}}\langle -\nabla_\pi J_\nu(\pi_t,\lambda), \pi - \pi_t\rangle + \frac{1}{2\eta}\|\pi-\pi_t\|_2^2$
end for
Return: π_T

Theorem 3.3. *For any two initial distributions* μ, ν *such that* $\min_s \nu(s) > 0$*, we have that for any* π*,*

$$J_\mu(\pi^*_\lambda,\lambda) - J_\mu(\pi,\lambda) \leq \left\|\frac{\mu}{\nu}\right\|_\infty \left(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi,\lambda)\right),$$

where $\left\|\frac{\mu}{\nu}\right\|_\infty = \max_s \frac{\mu(s)}{\nu(s)}$.

Assumption 3.1 (Bounded distribution mismatch)**.** *Assume that* $0 < \left\|\frac{\mu}{\nu}\right\| \leq \rho$ *and* $\left\|\frac{d_{\pi^*_\lambda,\nu}}{d_{\pi,\nu}}\right\|_\infty \leq \rho_\nu$ *for all* $0 \leq \lambda \leq \lambda_0$ *and* $\pi \in \Delta(\mathcal{A})^{\mathcal{S}}$.

In Assumption 3.1, ρ measures distribution mismatch between target initial distribution μ and behavior initial distribution ν. Besides, ρ_ν measures uniformity of ν. If ν is a uniform distribution on $\mathcal{S}$, $\rho_\nu = 1$ is obtained. Also, ρ_ν can be upper bounded by $\frac{1}{(1-\gamma)\min_s \nu(s)}$ trivially.

Under these assumptions, we are able to state our results for a fixed λ in Theorem 3.4.

Theorem 3.4 (Convergence of Projected Gradient Ascent)**.** *Assume* $\Omega(\pi)$ *is smooth,* λ *is fixed and Assumption 3.1 holds. Then Algorithm 2 outputs an policy* π_T *satisfying:*

$$J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_T,\lambda) \leq 4\rho_\nu\sqrt{|\mathcal{S}|}\left(\sqrt{\frac{2L_\lambda(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda))}{T}}\right),$$

where L_λ *is the smoothness of* $J_\nu(\pi,\lambda)$.

Corollary 3.1. *Under the same setting with Theorem 3.4 and letting Assumption 3.1 hold, Algorithm 2 outputs an policy* π_T *satisfying:*

$$J_\mu(\pi^*_\lambda,\lambda) - J_\mu(\pi_T,\lambda) \leq 4\rho\rho_\nu\sqrt{|\mathcal{S}|}\left(\sqrt{\frac{2L_\lambda(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda))}{T}}\right).$$

Remark 3.2. *We give an intuition for Theorem 3.4. By Agarwal et al. (2019) and assumption that* $\Omega(\pi)$ *is L-smooth, it is guaranteed that* V^π_λ *is also L-smooth. In this case, we can always find a first-order stationary point with rate* $O(\frac{1}{\varepsilon^2})$*. Agarwal et al. (2019) showed that the stationary point is also a global optimal point, which concludes Theorem 3.4.*

Theorem 3.4 tells us that the value function can be improved with rate $\widetilde{O}\left(\sqrt{\Delta/T}\right)$, where $\Delta = J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda)$. If we let the sub-solver host the Prop-I($\hat{\varepsilon}$) property, then $\mathsf{Time}(\lambda) = \widetilde{O}\left(\frac{1}{\Delta}\right)$. But we note that Δ could be arbitrarily small and it is not possible to lower bound Δ, which means that the sub-solver can not solve it in time independent on Δ. Thus we relax the Prop-I($\hat{\varepsilon}$) property to argue the sub-solver satisfies the Prop-II($\hat{\varepsilon}$) property. By Corollary 3.1, to obtain an ε-optimal policy, the iteration complexity could be $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|\rho^2\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon^2(1-\gamma)^4}\right)$. But, with Algorithm 2 as a sub-solver in Algorithm 1, the iteration complexity could reduce to $O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$ as Theorem 3.5 implies.

Corollary 3.2 (Time of Projected Gradient Ascent)**.** *Under the same settings in Theorem 3.4, at timestep* t *of Algorithm 1 with* Prop-II($\hat{\varepsilon}$)*, the iteration number of Projected Gradient Ascent is at most:*

$$\mathsf{Time}(\lambda_t) = \frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t.$$

Theorem 3.5. *Under the same settings in Theorem 3.4,* Alg *taking Algorithm 2 as the sub-solver gives an ε-optimal policy w.r.t. initial distribution μ in iteration $T = O\left(\log \frac{6\rho\lambda_0 C_\Phi}{\varepsilon(1-\gamma)}\right)$ in total time $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$, where $\widetilde{C}_\Phi$ is dependent on $|\Omega|, \|\nabla\Omega\|_\infty$ and $\|\nabla^2\Omega\|_\infty$, which can be referred to in Appendix B.3.5.*

Besides, we also extend our results to the case when exact information of dynamics is unknown (Theorem B.6). The sampling scheme and proof can be referred in Appendix B.3.6.

Theorem 3.6. *Suppose $\widehat{\pi}_{t+1}$ attains the minimum of $J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_{k+1}, \lambda_t)$ over $k = 0, ..., T_t - 1$ at each time-step t in Algorithm 1, where $T_t =$ Time(λ_t). Under the same setting in Theorem B.6,* Alg *taking Algorithm 2 with sampling as sub-solver leads to an ε-optimal policy w.r.t. initial distribution μ. Moreover, with probability $1-\delta$, the total iteration complexity is $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$ and the number of trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^5\rho_\nu^6(1+\lambda_0\widetilde{C}_\Phi)^3}{\varepsilon^5(1-\gamma)^7}\right)$.*

Remark We also investigate other algorithms such as the exact TRPO (Shani et al., 2019) and policy gradient in softmax parameterization (Mei et al., 2020). We show that the exact TRPO is equivalent to dynamic programming and that policy gradient in softmax parameterization is inefficient, in Appendices B.4 and B.5.

4. Discussions on AdaptReduce and Primal-Dual

In this section we illustrate the motivation of Algorithm 1 from a primal dual perspective. In fact, solving the unregularized MDP is equivalent to solving the regularized MDP by decaying λ as Theorem 4.1 describes.

Theorem 4.1 (Strong Duality)**.** *Under Assumption 2.1, and if $|\max_\pi J(\pi, \lambda)| < \infty$ for any fixed λ, we have that*

$$J(\pi^*) = \max_\pi \min_{\lambda\geq 0} J(\pi, \lambda) = \min_{\lambda\geq 0} \max_\pi J(\pi, \lambda), \qquad (4)$$

where $J(\pi, \lambda)$ is defined in (3)*, $J(\pi) := J(\pi, 0)$, and $\pi^* = \operatorname{argmax}_\pi J(\pi)$.*

The non-positivity of Ω and uniformly bounded property are crucial for Theorem 4.1, which make sure that the optimal λ^* solving the minimax problem will be exactly zero. In this case, the corresponding optimal policy will be our target π^*. Theorem 4.1 bridges the regularized MDP and the optimal policy π^*. It indicates that we could make use of the exchangeability of the maximum and minimum to devise efficient algorithms. It also enriches our tools to analyze the effect of λ decay.

The adaptive reduction scheme given in Section 3 decays λ exponentially fast and requires the sub-solver algorithm satisfies the Prop-I$(\hat{\varepsilon})$ or Prop-II$(\hat{\varepsilon})$ property. Based on the discussion of Section 3.2, intuitively, as long as we run a proper solver algorithm for an enough long time, the Prop-I$(\hat{\varepsilon})$ or Prop-II$(\hat{\varepsilon})$ property can be met, which implies many steps of policy gradient ascent are executed between two consecutive λ decayings. However, it is not saying that decaying λ exponentially is the only one scheme. Compared with other decaying schemes such as $\lambda_t = \frac{1}{t^\alpha}$ and $\lambda_t = \frac{1}{\log(t+1)}$, $\lambda_t = \frac{1}{2^t}$ is the best for sub-solvers we have analyzed. We have shown that the iteration complexity for obtaining the ε-optimal policy is $\widetilde{O}\left(\frac{1}{\varepsilon}\right)$ in terms of first order gradient method without parameterization. It still keeps open whether there exists a better decaying scheme to obtain a better convergence result. But we conjecture that the best we hope is $\widetilde{O}\left(\frac{1}{\varepsilon}\right)$ in terms of ε (Conjecture 4.1). For example, if we let projected gradient ascent satisfy Alg$(\pi_t, \lambda_t, \varepsilon_t, 0)$ with $\varepsilon_t = \lambda_t \frac{C_\Phi}{1-\gamma}$, the total time could be $\sum_{t=0}^{T-1}$ Time$(\lambda_t) = \Theta(\sum_{t=0}^{T-1} \frac{\lambda_{t-1}}{\lambda_t^2})$. If $\lambda_t/\lambda_{t-1} = o(1)$, which is faster than $\lambda_t = 1/2^t$, then the total time could be larger than $1/\varepsilon$ by the fact $\lambda_T = O(\varepsilon)$.

Although Bhandari and Russo (2020) conjectured that projected gradient ascent for the tabular unregularized MDP can achieve linear convergence rate, we argue that linear convergence rate can be met only with more information (e.g., $d_\pi(s)$) than just gradient. For other terms such as $\frac{1}{1-\gamma}$, we argue that it is still loose and we leave it to future work.

Conjecture 4.1 (Lower bound)**.** *There exist MDPs, given that learning rate η_t satisfies $\eta_t \leq \eta$ where η is a constant, and oracle:*

$$\pi_{t+1} \in \operatorname*{argmin}_{\pi\in\Delta(\mathcal{A})^\mathcal{S}} -\langle \nabla J(\pi_t, \lambda_t), \pi - \pi_t\rangle + \frac{1}{2\eta_t}\|\pi - \pi_t\|_2^2.$$

For any $\{\lambda_t\}_{t=1}^T$ such that $\lambda_{t+1} \leq \lambda_t$ and $\lim_{t\to\infty}\lambda_t = 0$, we have $V^{\pi^*} - V^{\pi_T} = \widetilde{\Omega}(\frac{1}{T})$, where we ignore parameters polynominally dependent on MDP.

Remark 4.1. *Here we argue that it is important that η_t is bounded. If assuming that η_t could be arbitrary positive, we can always do exact line search such that the iteration complexity could be $O(\log\frac{1}{\varepsilon})$ (Bhandari and Russo, 2020).*

5. Conclusion

In this paper we have proposed an adaptive reduction scheme to decay the regularization coefficient. We have shown that the iteration complexity for obtaining an ε-optimal policy can be accelerated by our scheme compared with setting λ sufficiently small. Moreover, we have discussed the connection of our approach with the primal dual problem, stating that the reduction scheme can be regarded as solving the dual problem. It still keeps open how to devise a more efficient decaying scheme for some other sub-solvers in regularized RL scenarios. We would like to address this issue in future work.

References

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. *arXiv preprint arXiv:1908.00261*, 2019.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International Conference on Machine Learning*, pages 151–160. PMLR, 2019.

Mohammad Gheshlaghi Azar, Vicenc Gómez, and Hilbert J Kappen. Dynamic policy programming. *The Journal of Machine Learning Research*, 13(1):3207–3245, 2012.

Richard Bellman and Stuart Dreyfus. Functional approximations and dynamic programming. *Mathematical Tables and Other Aids to Computation*, pages 247–251, 1959.

Jalaj Bhandari and Daniel Russo. Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786*, 2019.

Jalaj Bhandari and Daniel Russo. A note on the linear convergence of policy gradient methods. *arXiv preprint arXiv:2007.11120*, 2020.

Jalaj Bhandari and Daniel Russo. On the linear convergence of policy gradient methods for finite mdps. In *International Conference on Artificial Intelligence and Statistics*, pages 2386–2394. PMLR, 2021.

Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *arXiv preprint arXiv:2007.06558*, 2020.

Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, Jianshu Chen, and Le Song. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning*, pages 1133–1142, 2018.

Roy Fox, Ari Pakman, and Naftali Tishby. Taming the noise in reinforcement learning via soft updates. *arXiv preprint arXiv:1512.08562*, 2015.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. *arXiv preprint arXiv:1901.11275*, 2019.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pages 267–274, 2002.

Sham M Kakade. A natural policy gradient. In *Advances in neural information processing systems*, pages 1531–1538, 2002.

Kyungjae Lee, Sungjoon Choi, and Songhwai Oh. Sparse markov decision processes with causal sparse tsallis entropy regularization for reinforcement learning. *IEEE Robotics and Automation Letters*, 3(3):1466–1473, 2018.

Gen Li, Yuting Wei, Yuejie Chi, Yuantao Gu, and Yuxin Chen. Softmax policy gradient methods can take exponential time to converge. *arXiv preprint arXiv:2102.11270*, 2021.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. *arXiv preprint arXiv:2005.06392*, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

Gergely Neu, Anders Jonsson, and Vicenc Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

Martin L Puterman and Moon Chirl Shin. Modified policy iteration algorithms for discounted markov decision problems. *Management Science*, 24(11):1127–1137, 1978.

Bruno Scherrer, Mohammad Ghavamzadeh, Victor Gabillon, Boris Lesner, and Matthieu Geist. Approximate modified policy iteration and its application to the game of tetris. *J. Mach. Learn. Res.*, 16:1629–1676, 2015.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897, 2015.

John Schulman, Xi Chen, and Pieter Abbeel. Equivalence between policy gradients and soft q-learning. *arXiv preprint arXiv:1704.06440*, 2017.

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. *arXiv preprint arXiv:1909.02769*, 2019.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587): 484–489, 2016.

Elena Smirnova and Elvis Dohmatob. On the convergence of approximate and regularized policy iteration schemes. *arXiv preprint arXiv:1909.09621*, 2019.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000.

Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52(1-2): 479–487, 1988.

Peter Vamplew, Richard Dazeley, and Cameron Foale. Softmax exploration strategies for multiobjective reinforcement learning. *Neurocomputing*, 263:74–86, 2017.

Nino Vieillard, Tadashi Kozuno, Bruno Scherrer, Olivier Pietquin, Rémi Munos, and Matthieu Geist. Leverage the average: an analysis of regularization in rl. *arXiv preprint arXiv:2003.14089*, 2020.

Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural policy gradient methods: Global optimality and rates of convergence. *arXiv preprint arXiv:1909.01150*, 2019.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Wenhao Yang, Xiang Li, and Zhihua Zhang. A regularized approach to sparse optimal policy in reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 5938–5948, 2019.

Appendix

A. Related Work

There are a plenty of works solving (regularized) MDPs. For the limited space, we mainly discuss dynamic programming and first order gradient methods in this paper. For the former, in spite of regularization, we can still define a (regularized) Bellman operator that serves as a γ-contraction (Yang et al., 2019; Neu et al., 2017). Applying the regularized Bellman operator iteratively, dynamic programming methods achieve a linear convergence rate as before (Neu et al., 2017; Vieillard et al., 2020; Smirnova and Dohmatob, 2019). Smirnova and Dohmatob (2019) analyzed the convergence of a general form of regularized policy iteration when λ_t vanishes in a asymptotic sense. However, they did not give a specific decay scheme and only analyzed how regularized policy iteration converges with different asymptotic λ decaying rates.

In tabular unregularized MDP scenarios, Agarwal et al. (2019) showed that the vanilla policy gradient method achieves an $\widetilde{O}(\frac{1}{\varepsilon^2})$ iteration complexity, while Bhandari and Russo (2020; 2021) argued that the complexity rate should be $O(\log \frac{1}{\varepsilon})$ in the same setting. In particular, to achieve linear convergence, Bhandari and Russo (2020) used the exact line search to select a good learning rate, which ensures the resulting method behaves no worse than policy iteration and thus converging linearly. Actually, Bhandari and Russo (2020) required to use the optimal learning rate which is likely to be quite large and even infinity. The potential of learning rate explosion together with time-consuming line search makes their method impractical.

Shani et al. (2019) analyzed a variant of TRPO and showed that its iteration complexity is $\widetilde{O}(\frac{1}{\lambda\varepsilon})$ in regularized MDPs which is quite smaller than that in the unregularized case $\widetilde{O}(\frac{1}{\varepsilon^2})$. Therefore, they claimed their method converges faster in regularized MDPs in terms of dependence on ε. However, in order to obtain an ε-optimal policy, the temperature λ is highly related to the desired accuracy ε (which typically is $\Theta(\varepsilon)$), so the complexity remains the same. For vanilla policy gradient methods with softmax-parameterized policies, Mei et al. (2020) showed that for a fixed coefficient λ the iteration complexities for unregularized and regularized MDPs are $\widetilde{O}(\frac{1}{\varepsilon})$ and $\widetilde{O}(\log \frac{1}{\varepsilon})$ respectively. Though it seems much faster than many previous works (including ours), we note that problem-dependent parameters are hidden in the $\widetilde{O}$. This leads to a problematic iteration complexity in regularized MDPs when λ is set sufficiently small. As a consequence, when choosing softmax policy gradient as the sub-solver and using the analysis in (Mei et al., 2020), our adaptive decaying scheme is not even efficient (i.e., it might depend on ε exponentially). We detail the discussion in Appendix B.4. It implies the current analysis on softmax policy gradient is not tight or softmax policy gradient is not efficient as we expect. Indeed, Li et al. (2021) showed that the softmax policy gradient method may suffer an exponential lower on horizon $\frac{1}{1-\gamma}$. Cen et al. (2020) proposed to apply natural gradient to softmax-parameterized policies, obtaining a linear convergence rate. Our adaptive method maintains their convergence rate, and we discuss it detailedly in Appendix B.5.

B. Proof of Section 3

B.1. Proof of Theorem 3.1

Lemma B.1 (Boundness of Φ^π)**.** *For any regularization function Ω satisfying Assumption 2.1, it follows that for any policy π,*

$$\|\Phi^\pi\|_\infty \le \frac{C_\Phi}{1-\gamma} \text{ and } |\Phi^\pi(\mu)| \le \frac{C_\Phi}{1-\gamma}.$$

Proof. It is easy to verify that for any given π and for any $s \in \mathcal{S}$,

$$\Phi^\pi(s) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t\Omega(\pi, s_t)|s_0 = s\right] = \frac{1}{1-\gamma}\sum_{s'\in\mathcal{S}} d_s^\pi(s')\Phi(\pi, s')$$

which directly implies that $|\Phi^\pi(s)| \le \frac{C_\Phi}{1-\gamma}$ uniformly in s. □

With Lemma B.1, we can prove that Algorithm 1 with any Alg satisfying Prop-I($\hat{\varepsilon}$) will find the optimal policy in an exponentially fast speed.

Proof of Theorem 3.1. For any given $s \in \mathcal{S}$, define $D_t(s) = V^*_{\lambda_t}(s) - V^{\hat{\pi}_t}_{\lambda_t}(s)$ to be the initial value difference between $\hat{\pi}_t$ and the optimal policy of $\mathcal{M}(\lambda_t)$ before we call Alg in iteration t. Let $D_t = \mathbb{E}_{s_0\sim\mu} D_t(s_0) = V^*_{\lambda_t}(\mu) - V^{\hat{\pi}_t}_{\lambda_t}(\mu)$ for

simplicity (which is always non-negative). Then,

$$\begin{aligned}
0 \le D_t = V^*_{\lambda_t}(\mu) - V^{\hat{\pi}_t}_{\lambda_t}(\mu) &\overset{(a)}{=} V^{\pi^*_{\lambda_t}}_{\lambda_t}(\mu) - V^{\hat{\pi}_t}_{\lambda_t}(\mu) \\
&\overset{(b)}{=} V^{\pi^*_{\lambda_t}}_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) + (\lambda_{t-1}-\lambda_t)\Phi^{\pi^*_\lambda}(\mu) - (\lambda_{t-1}-\lambda_t)\Phi^{\hat{\pi}_t}(\mu) \\
&\overset{(c)}{\le} V^{\pi^*_{\lambda_t}}_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) + 2(\lambda_{t-1}-\lambda_t)\frac{C_\Phi}{1-\gamma} \\
&\overset{(d)}{\le} V^*_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) + 2(\lambda_{t-1}-\lambda_t)\frac{C_\Phi}{1-\gamma} \\
&\overset{(e)}{=} V^*_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) + \frac{\lambda_0}{2^{t-1}}\frac{C_\Phi}{1-\gamma} \\
&\overset{(f)}{\le} \frac{1}{4}D_{t-1} + \hat{\varepsilon} + \frac{\lambda_0}{2^{t-1}}\frac{C_\Phi}{1-\gamma}
\end{aligned}$$

where (a) follows from the notation of the optimal policy $V^*_{\lambda_t} = V^{\pi^*_{\lambda_t}}_{\lambda_t}$; (b) uses the value function decomposition; (c) uses the boundness of Ω (which is $\|\Phi^\pi\|_\infty \le \frac{C_\Phi}{1-\gamma}$); (d) uses the the optimality of $\pi^*_{\lambda_{t-1}}$, i.e., $V^{\pi^*_{\lambda_t}}_{\lambda_{t-1}}(\mu) \le V^*_{\lambda_{t-1}}(\mu) := V^{\pi^*_{\lambda_{t-1}}}_{\lambda_{t-1}}(\mu)$; (e) uses the fact that $\lambda_t = \frac{\lambda_0}{2^t}$; and (f) uses the Prop-I($\hat{\varepsilon}$) property of Alg, which ensures that $V^*_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) \le \frac{1}{4}(V^*_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_{t-1}}_{\lambda_{t-1}}(\mu)) + \hat{\varepsilon} = \frac{1}{4}D_{t-1} + \hat{\varepsilon}$. Recursively applying the above inequality, we have

$$D_T \le \frac{D_0}{4^T} + \frac{4}{3}\hat{\varepsilon} + \frac{\lambda_0 C_\Phi}{1-\gamma}\sum_{i=0}^{T-1}\frac{1}{2^{T-1-i}4^i} \le \frac{D_0}{4^T} + \frac{4}{3}\hat{\varepsilon} + \frac{4\lambda_0 C_\Phi}{1-\gamma}\frac{1}{2^T}. \tag{5}$$

In sum, we obtain a policy $\hat{\pi}_T$ satisfying

$$\begin{aligned}
0 \le V^*(\mu) - V^{\hat{\pi}_T}(\mu) &\overset{(a)}{=} V^{\pi^*}(\mu) - V^{\hat{\pi}_T}(\mu) \\
&\overset{(b)}{=} V^{\pi^*}_{\lambda_T}(\mu) - V^{\hat{\pi}_T}_{\lambda_T}(\mu) + \lambda_T\Phi^{\hat{\pi}_T}(\mu) - \lambda_T\Phi^{\pi^*}(s) \\
&\overset{(c)}{\le} V^{\pi^*}_{\lambda_T}(\mu) - V^{\hat{\pi}_T}_{\lambda_T}(\mu) + 2\lambda_T\frac{C_\Phi}{1-\gamma} \\
&\overset{(d)}{\le} V^*_{\lambda_T}(\mu) - V^{\hat{\pi}_T}_{\lambda_T}(\mu) + 2\lambda_T\frac{C_\Phi}{1-\gamma}
\end{aligned}$$

where (a) follows from the notation of the optimal policy $V^* = V^{\pi^*}$; (b) uses the value decomposition of $V^*_{\lambda_T}$ and $V^{\hat{\pi}_T}_{\lambda_T}$; (c) uses the boundness of Φ^π shown in Lemma B.1; and (d) uses the the optimality of $\pi^*_{\lambda_T}$, i.e., $V^{\pi^*}_{\lambda_T}(\mu) \le V^*_{\lambda_T}(\mu) := V^{\pi^*_{\lambda_T}}_{\lambda_T}(\mu)$. Since the bound holds uniformly for all $s \in \mathcal{S}$, we have

$$\begin{aligned}
V^*(\mu) - V^{\hat{\pi}_T}(\mu) &\le V^*_{\lambda_T}(\mu) - V^{\hat{\pi}_T}_{\lambda_T}(\mu) + 2\lambda_T\frac{C_\Phi}{1-\gamma} \\
&\overset{(a)}{=} D_T + \frac{2\lambda_0 C_\Phi}{1-\gamma}\frac{1}{2^T} \overset{(b)}{\le} \frac{D_0}{4^T} + \frac{4}{3}\hat{\varepsilon} + \frac{6\lambda_0 C_\Phi}{1-\gamma}\frac{1}{2^T}.
\end{aligned}$$

Above, (a) uses the definition of D_T and λ_T and (b) applies (5). □

B.1.1. Proof of Theorem 3.2

Proof. Similar with Theorem 3.1, we denote $D_t(s) = V^*_{\lambda_t}(s) - V^{\hat{\pi}_t}_{\lambda_t}(s)$ to be the initial value difference and $D_t = V^*_{\lambda_t}(\mu) - V^{\hat{\pi}_t}_{\lambda_t}(\mu)$. By proof of Theorem 3.1,

$$0 \le D_t \le V^*_{\lambda_{t-1}}(\mu) - V^{\hat{\pi}_t}_{\lambda_{t-1}}(\mu) + \frac{\lambda_0}{2^{t-1}}\frac{C_\phi}{1-\gamma}$$

$$\overset{(a)}{\leq} \frac{4\lambda_0}{2^t}\frac{C_\phi}{1-\gamma} + \hat{\varepsilon} \tag{6}$$

where (a) follows from Prop-II($\hat{\varepsilon}$).

Thus, for the output $\hat{\pi}_T$, we have:

$$\begin{aligned} 0 \leq V^*(\mu) - V^{\hat{\pi}_T}(\mu) &\leq D_T + \frac{2\lambda_0}{2^T}\frac{C_\phi}{1-\gamma} \\ &\leq \frac{6\lambda_0}{2^T}\frac{C_\phi}{1-\gamma} + \hat{\varepsilon} \end{aligned} \tag{7}$$

□

B.2. Proof related to RMPI

B.2.1. Warm-up: Regularized Modified Policy Iteration

We define the regularized Bellman operator $\mathcal{T}_\lambda^\pi : \mathbb{R}^S \to \mathbb{R}^S$ with $s \in \mathcal{S}$ entry given by

$$[\mathcal{T}_\lambda^\pi V](s) = \mathbb{E}_{a\sim\pi(\cdot|s)} Q(s,a) - \lambda\Omega(\pi(\cdot|s)), \tag{8}$$

$$Q(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)} V(s'). \tag{9}$$

It can be shown that the regularized Bellman operators have the same properties as the classical ones. So Geist et al. (2019) applied classical dynamic programming to solve the regularized MDP problem. They proposed regularized MPI (RMPI) by modifying a classic dynamic programming method Modified policy iteration (MPI) (Puterman and Shin, 1978).

Algorithm 3 Regularized Modified Policy Iteration (RMPI)

Input: an initial policy π_0, a regularization parameter λ, and K the number of iteration.
for iteration $k = 0$ **to** $K-1$ **do**
  find π_{k+1} such that $\max_{\pi\in\Delta(\mathcal{A})} \mathcal{T}_\lambda^\pi V_k \leq \mathcal{T}_\lambda^{\pi_{k+1}} V_k + \epsilon_0 1_S$ point-wisely;
  $V_{k+1} = \left(\mathcal{T}_\lambda^{\pi_{k+1}}\right)^m V_k + \epsilon_0 1_S$
end for
Return: π_K

Theorem B.1. *Define* $C_\infty^i = \frac{1-\gamma}{\gamma^i}\sum_{j=i}^\infty \gamma^j \max_{\pi_1,\ldots,\pi_j} \left\| \frac{\mu P_{\pi_1} P_{\pi_2} \ldots P_{\pi_j}}{\mu} \right\|_\infty$ *and* $C = \max_{i\geq 0} C_\infty^i$. *Then RMPI (Algorithm 3) satisfies the* Prop-I($\hat{\varepsilon}$) *property with* $\hat{\varepsilon} = \frac{(1+2\gamma)C}{(1-\gamma)^2}\epsilon_0$ *and* Time$(\lambda) = \ln\frac{8C}{1-\gamma} / \ln\frac{1}{\gamma}$.

Theorem B.1 shows that RMPI has the Prop-I($\hat{\varepsilon}$) property; here $\hat{\epsilon}$ resulted from the numerical error ϵ_0. Besides, the time Time$(\lambda) = \ln\frac{8C}{1-\gamma} / \ln\frac{1}{\gamma}$ has nothing to do with λ; indeed, for a sequence that is generated by a contraction mapping, it needs only a constant number of iterations to get a constant factor close to the fix point. The fact that the regularized Bellman operator is a γ-contraction (see Proposition 2 in Geist et al. (2019)) makes Time(λ) unrelated with λ.

Corollary B.1. *Given an accuracy* ϵ, *if* ϵ_0 *is sufficiently small such that* $\epsilon_0 \leq \frac{(1-\gamma)^2}{(1+2\gamma)C}\cdot\epsilon$, *then* Alg *with RMPI serving as the sub-solver produces an* ϵ*-optimal policy in outer iteration* $T = O(\log\frac{C_0}{\epsilon})$ *and in time (total iteration)* $O\left(\frac{1}{1-\gamma}\log\frac{8C}{1-\gamma}\log\frac{C_0}{\epsilon}\right)$ *with* $C_0 = V_{\lambda_0}^*(\mu) - V_{\lambda_0}^{\hat{\pi}_0}(\mu) + \frac{\lambda_0 C_\Phi}{(1-\gamma)}$.

Corollary B.1 is directly derived from Theorem B.1. When $\lambda = 0$, RMPI is reduced to MPI, the latter still satisfying the Prop-I($\hat{\varepsilon}$) property. For sake of simplicity, assume $\epsilon_0 = 0$ without loss of generality[1]. To reach an ϵ-optimal policy, MPI needs $O(\frac{1}{1-\gamma}\log\frac{1}{\epsilon(1-\gamma)})$ iterations. Interestingly, RMPI needs the same number of iterations up to logarithmic factors, which implies there is no harm to use RMPI instead of MPI. We discuss why we cannot expect faster convergence rate of RMPI in Appendix B.2.4.

[1] Otherwise, we can set ϵ_0 sufficiently small such that $\hat{\epsilon} = O(\epsilon)$, then the effect of $\hat{\epsilon}$ is ignorable.

B.2.2. PROOF OF THEOREM B.1

Proof. Here we will make use of Corollary 1 in (Geist et al., 2019) or Theorem 7 in (Scherrer et al., 2015) to prove Theorem B.1. Let ν, μ be two state distributions with μ used for initial state distribution and ν for performance measure. Let p, q and q' such that $\frac{1}{q} + \frac{1}{q'} = 1$. We define a norm by $\|V\|_{p,\mu} := [\mathbb{E}_{s_0\sim\mu}|V(s_0)|^p]^{\frac{1}{p}}$ and the concentrability coefficients as:

$$C_q^i = \frac{1-\gamma}{\gamma^i}\sum_{j=i}^{\infty}\gamma^j \max_{\pi_1,\dots,\pi_j}\left\|\frac{\rho P_{\pi_1}P_{\pi_2}\dots P_{\pi_j}}{\nu}\right\|_{q,\nu}.$$

Then, by Corollary 1 in (Geist et al., 2019), we have

$$\|V_\lambda^* - V_\lambda^{\pi_k}\|_{p,\rho} \le 2\sum_{i=1}^{k-1}\frac{\gamma^i}{1-\gamma}\left(C_q^i\right)^{\frac{1}{p}}\|\epsilon_{k-i}\|_{pq',\nu} + \sum_{i=0}^{k-1}\frac{\gamma^i}{1-\gamma}\left(C_q^i\right)^{\frac{1}{p}}\left\|\epsilon'_{k-i}\right\|_{pq',\nu} + g(k) \tag{10}$$

with

$$g(k) = \frac{2\gamma^k}{1-\gamma}\left(C_q^k\right)^{\frac{1}{p}}\min\left(\|V_\lambda^* - V_\lambda^{\pi_0}\|_{pq',\nu}, \|V_\lambda^{\pi_0} - \mathcal{T}_\lambda^{\pi_1}V_\lambda^{\pi_0}\|_{pq',\nu}\right).$$

Here we set $p = q' = 1$, which implies $q = \infty$, and $\epsilon_k = \epsilon'_k = \epsilon_0 1_S$ for $k \ge 0$. Under the choice of parameters, we can simplify C_q^i by noting that

$$\left\|\frac{\rho P_{\pi_1}P_{\pi_2}\dots P_{\pi_j}}{\nu}\right\|_{\infty,\nu} \le \left\|\frac{\rho P_{\pi_1}P_{\pi_2}\dots P_{\pi_j}}{\nu}\right\|_{\infty} \le \frac{\max_s\rho(s)}{\min_s\nu(s)}, \text{ for any } j \ge 1 \text{ and } \pi_1,\cdots,\pi_j. \tag{11}$$

Therefore, $C_\infty^i \le \frac{\max_s\rho(s)}{\min_s\nu(s)}$ for all $i \ge 0$.

Next we let $\rho = \mu$, $\nu = \mu$, and denote $C = \max_i C_\infty^i$, it follows that

$$\begin{aligned}
V_\lambda^*(\mu) - V_\lambda^{\pi_k}(\mu) &= |\mathbb{E}_{s_0\sim\mu}\left[V_\lambda^*(s_0) - V_\lambda^{\pi_k}(s_0)\right]| \\
&\overset{(a)}{=} \mathbb{E}_{s_0\sim\mu}|V_\lambda^*(s_0) - V_\lambda^{\pi_k}(s_0)| \\
&= \|V_\lambda^* - V_\lambda^{\pi_k}\|_{1,\mu} \\
&\overset{(b)}{\le} g(k) + \frac{2C(\gamma-\gamma^k)\epsilon_0}{(1-\gamma)^2} + \frac{C(1-\gamma^k)\epsilon_0}{(1-\gamma)^2} \\
&\overset{(c)}{\le} \frac{2C\gamma^k}{1-\gamma}\left(V_\lambda^*(\mu) - V_\lambda^{\pi_0}(\mu)\right) + \frac{(1+2\gamma)C}{(1-\gamma)^2}\epsilon_0
\end{aligned}$$

where (a) follows the fact that $V_\lambda^*(s) \ge V_\lambda^\pi(s)$ for any π and $s \in \mathcal{S}$ satisfying $\mu(s) > 0$; (b) follows from (10); (c) uses the same argument for (a) that is $V_\lambda^*(\mu) - V_\lambda^{\pi_k}(\mu) = \|V_\lambda^* - V_\lambda^{\pi_k}\|_{1,\mu}$. Then when the k is large enough such that $\frac{2\gamma^k}{1-\gamma} \le \frac{1}{4}$, which implies $k \ge \ln\frac{8C}{1-\gamma}/\ln\frac{1}{\gamma}$, RMPI satisfy Prop-I($\frac{(1+2\gamma)C}{(1-\gamma)^2}\epsilon_0$) in times $\ln\frac{8C}{1-\gamma}/\ln\frac{1}{\gamma}$. □

B.2.3. RMPI WITH ALTERNATIVE REDUCTION

Note that RMPI alone converges linearly, which means one step of RMPI is already able to improve the current policy considerably. When applying RMPI as a sub-solver for AdaptReduce, we run $\widetilde{\mathcal{O}}(1)$ steps of RMPI between consecutive λ decays. In fact, We find that this can be relaxed into a more simplified form (Algorithm 4), where the policy π and coefficient λ can be updated alternatively. Under this scheme, we show in Theorem B.2 that the convergence rate can be imporved to $O\left(\frac{1}{1-\gamma}\log\frac{C_0}{\varepsilon(1-\gamma)}\right)$, a logarithmic term removed.

Theorem B.2. *Given an accuracy ε and assuming $\frac{1}{2} \le \gamma^2$, if ε_0 satisfies $\varepsilon_0 \le \frac{(1-\gamma)^2}{6}\varepsilon$, the output π_T of Algorithm 4 is an ϵ-optimal policy after $O\left(\frac{1}{1-\gamma}\log\frac{C_0}{\varepsilon(1-\gamma)}\right)$ iterations.*

In this part, we denote $V_{T+1} = \mathcal{T}_{\lambda_T}^{\pi_T}V_T + \varepsilon_0 1_S$.

Lemma B.2. $\|V_{T+1} - V_{\lambda_T}^{\pi_T}\|_\infty \le \frac{\gamma}{1-\gamma}\|V_T - V_{T+1}\|_\infty + \frac{1}{1-\gamma}\varepsilon_0$

Algorithm 4 RMPI with Alternative Reduction

Input: initial value function V_0 and policy π_0, a regularization parameter λ_0, and T the number of iteration.
for iteration $t = 0$ **to** $T-1$ **do**
$V_{t+1} = \mathcal{T}_{\lambda_t}^{\pi_t} V_t + \varepsilon_0 1_{\mathbf{S}}$
$\lambda_{t+1} = \frac{1}{2}\lambda_t$
find π_{t+1} such that $\max_{\pi \in \Delta(\mathcal{A})} \mathcal{T}_{\lambda_{t+1}}^{\pi} V_{t+1} \leq \mathcal{T}_{\lambda_{t+1}}^{\pi_{t+1}} V_{t+1} + \varepsilon_0 1_S$
end for
Return: π_T

Proof. By definition of Bellman equation, we have:

$$V_{T+1} - V_{\lambda_T}^{\pi_T} = \mathcal{T}_{\lambda_T}^{\pi_T} V_T - V_{\lambda_T}^{\pi_T} + \varepsilon_0 1_S \tag{12}$$
$$= \mathcal{T}_{\lambda_T}^{\pi_T} V_T - \mathcal{T}_{\lambda_T}^{\pi_T} (\mathcal{T}_{\lambda_T}^{\pi_T})^{\infty} V_T + \varepsilon_0 1_S \tag{13}$$

Taking infinite norm, we have:

$$\|V_{T+1} - V_{\lambda_T}^{\pi_T}\|_\infty \leq \gamma \|V_T - (\mathcal{T}_{\lambda_T}^{\pi_T})^{\infty} V_T\|_\infty + \varepsilon_0 \tag{14}$$
$$\leq \gamma \sum_{n=0}^{\infty} \|(\mathcal{T}_{\lambda_T}^{\pi_T})^n V_T - (\mathcal{T}_{\lambda_T}^{\pi_T})^{n+1} V_T\|_\infty + \varepsilon_0 \tag{15}$$
$$\leq \gamma \sum_{n=0}^{\infty} \gamma^n \|V_T - \mathcal{T}_{\lambda_T}^{\pi_T} V_T\|_\infty + \varepsilon_0 \tag{16}$$
$$\leq \frac{\gamma}{1-\gamma} \|V_T - V_{T+1}\|_\infty + \frac{1}{1-\gamma} \varepsilon_0 \tag{17}$$

□

Lemma B.3. *For any $t = 0, 1, ..., T$, $\|V^* - V_{t+1}\|_\infty \leq \lambda_t C_\phi + \gamma \|V^* - V_t\|_\infty + \varepsilon_0$*

Proof. By definition, we have:

$$\mathcal{T} V^* - \mathcal{T}_{\lambda_t} V_t - \varepsilon_0 1_S \leq V^* - V_{t+1} = \mathcal{T} V^* - \mathcal{T}_{\lambda_t}^{\pi_t} V_t - \varepsilon_0 1_S \leq \mathcal{T} V^* - \mathcal{T}_{\lambda_t} V_t \tag{18}$$

Taking infinity norm both sides, we have:

$$\|V^* - V_{t+1}\|_\infty \leq \|\mathcal{T} V^* - \mathcal{T}_{\lambda_t} V_t\|_\infty + \varepsilon_0 \tag{19}$$
$$\leq \max_{\pi} \lambda_t \|\langle \phi(\pi), \pi \rangle\|_\infty + \gamma \|P^\pi (V^* - V_t)\|_\infty + \varepsilon_0 \tag{20}$$
$$\leq \lambda_t C_\phi + \gamma \|V^* - V_t\|_\infty + \varepsilon_0 \tag{21}$$

□

Proof of Theorem B.2. Denote $V_{T+1} = \mathcal{T}_{\lambda_T}^{\pi_T} V_T + \varepsilon_0 1_S$, we have:

$$0 \leq V^* - V^{\pi_T} = V^* - V_{\lambda_T}^{\pi_T} - \lambda_T \Phi(\pi_T) \tag{22}$$
$$\leq V^* - V_{\lambda_T}^{\pi_T} + \frac{\lambda_T}{1-\gamma} C_\phi \tag{23}$$
$$= V^* - V_{T+1} + V_{T+1} - V_{\lambda_T}^{\pi_T} + \frac{\lambda_T}{1-\gamma} C_\phi \tag{24}$$

By Lemma B.3 and assuming $\frac{1}{2} \leq \gamma^2$, we obtain the following inequality by recursion and $\lambda_{t+1} = \frac{1}{2}\lambda_t$:

$$\|V^* - V_T\|_\infty \leq \frac{\gamma^{T-1}}{1-\gamma} \lambda_0 C_\phi + \frac{1}{1-\gamma} \varepsilon_0 + \gamma^T \|V^* - V_0\|_\infty \tag{25}$$

By Lemma B.2, we obtain the final bound of $\|V^* - V^{\pi_T}\|_\infty$:

$$\|V^* - V^{\pi_T}\|_\infty \le \|V^* - V_{T+1}\|_\infty + \|V_{T+1} - V_{\lambda_T}^{\pi_T}\|_\infty + \frac{\lambda_T}{1-\gamma} C_\phi \tag{26}$$

$$\le \|V^* - V_{T+1}\|_\infty + \frac{\gamma}{1-\gamma}\|V^* - V_{T+1}\|_\infty + \frac{\gamma}{1-\gamma}\|V^* - V_T\|_\infty + \frac{\varepsilon_0}{1-\gamma} + \frac{\lambda_T}{1-\gamma} C_\phi \tag{27}$$

$$= \frac{1}{1-\gamma}\|V^* - V_{T+1}\|_\infty + \frac{\gamma}{1-\gamma}\|V^* - V_T\|_\infty + \frac{\varepsilon_0}{1-\gamma} + \frac{\lambda_T}{1-\gamma} C_\phi \tag{28}$$

$$\le \frac{2\gamma}{1-\gamma}\|V^* - V_T\|_\infty + \frac{2\varepsilon_0}{1-\gamma} + \frac{2\lambda_T}{1-\gamma} C_\phi \tag{29}$$

$$\le \frac{2\gamma^T}{(1-\gamma)^2}\lambda_0 C_\phi + \frac{2}{(1-\gamma)^2}\varepsilon_0 + \frac{2\gamma^{T+1}}{1-\gamma}\|V^* - V_0\|_\infty + \frac{2\lambda_T}{1-\gamma} C_\phi \tag{30}$$

$$\le \frac{4\gamma^T}{(1-\gamma)^2}\lambda_0 C_\phi + \frac{2}{(1-\gamma)^2}\varepsilon_0 + \frac{2\gamma^{T+1}}{1-\gamma}\|V^* - V_0\|_\infty \tag{31}$$

Let $\varepsilon_0 \le \frac{(1-\gamma)^2}{6}\varepsilon$ and $T \ge \max\{\frac{\log \frac{\varepsilon(1-\gamma)^2}{12\lambda_0 C_\phi}}{\log\gamma}, \frac{\log \frac{\varepsilon(1-\gamma)}{6\|V^* - V_0\|_\infty}}{\log\gamma} - 1\}$, we have $\|V^* - V^{\pi_T}\|_\infty \le \varepsilon$ □

B.2.4. Does regularization help in DP method?

In DP method with exact information of rewards and transition probability, we show that the convergence rate is $\widetilde{O}(\frac{1}{1-\gamma})$ for both unregularized and regularized MDPs, which indicates that regularization performs at least well as the case when no regularization is applied. Even though Geist et al. (2019) expected an acceleration with regularization, we argue that it's not true in the worst case by following examples.

Theorem B.3. *There exists an MDP with $|\mathcal{S}| = |\mathcal{A}| = 2$, the sequence of V_t generated by Value Iteration (VI) satisfies:*

$$|V_t(s) - V^*(s)| = C \cdot \gamma^t$$

for all s and t.

Proof. We denote $\mathcal{S} = \{s_1, s_2\}$ and $\mathcal{A} = \{a_1, a_2\}$. Besides, we also define the reward as $R(s_1, \cdot) = 1$ and $R(s_2, \cdot) = 0$, and the transition probability as $P(s_1|s_1, a_1) = P(s_2|s_1, a_2) = P(s_1|s_2, a_2) = P(s_2|s_2, a_1) = 1$. For any policy π, we have:

$$P^\pi = \begin{bmatrix} \pi(a_1|s_1) & 1 - \pi(a_1|s_1) \\ 1 - \pi(a_1|s_2) & \pi(a_1|s_2) \end{bmatrix}$$

Thus, we can explicitly express the one-step value iteration as follows:

$$V_{t+1} = \max_\pi \begin{pmatrix} 1 + \gamma\pi(a_1|s_1)(V_t(s_1) - V_t(s_2)) + \gamma V_t(s_2) \\ \gamma\pi(a_1|s_2)(V_t(s_2) - V_t(s_1)) + \gamma V_t(s_1) \end{pmatrix}$$

Starting from $V_0(s_1) = V_0(s_2) = 0$, we have:

$$|V_t(s_1) - V^*(s_1)| = |V_t(s_2) - V^*(s_2)| = \frac{\gamma^t}{1-\gamma}$$

□

Theorem B.4. *There exists an MDP with $|\mathcal{S}| = |\mathcal{A}| = 2$, for any fixed $\lambda > 0$, the sequence of $V_{\lambda,t}$ generated by Regularized Value Iteration (RVI) satisfies:*

$$|V_{\lambda,t}(s) - V_\lambda^*(s)| = C \cdot \gamma^t$$

for all s and t.

Proof. The example of Theorem B.3 meets the result. However, the computation is complicated. To simplify, we make a little change of the example. We denote $\mathcal{S} = \{s_1, s_2\}$ and $\mathcal{A} = \{a_1, a_2\}$. Besides, we also define the reward as $R(s_1, \cdot) = 1$ and $R(s_2, \cdot) = 0$, and the transition probability as $P(s_1|s_1, a_1) = P(s_1|s_1, a_2) = P(s_1|s_2, a_2) = P(s_2|s_2, a_1) = 1$. For any policy π, we have:

$$P^\pi = \begin{bmatrix} 1 & 0 \\ 1 - \pi(a_1|s_2) & \pi(a_1|s_2) \end{bmatrix}$$

Thus, we can also explicitly express the one-step regularized value iteration as follows:

$$V_{\lambda,t+1} = \max_\pi \begin{pmatrix} 1 + \lambda H(\pi(\cdot|s_1)) + \gamma V_{\lambda,t}(s_1) \\ \lambda H(\pi(\cdot|s_2)) + \gamma\pi(a_1|s_2)(V_{\lambda,t}(s_2) - V_{\lambda,t}(s_1)) + \gamma V_{\lambda,t}(s_1) \end{pmatrix}$$

By solving the maximum operator, we obtain:

$$V_{\lambda,t+1} = \begin{pmatrix} 1 + \lambda \log 2 + \gamma V_{\lambda,t}(s_1) \\ \lambda \log(1 + e^{\frac{\gamma\Delta_t}{\lambda}}) + \gamma V_{\lambda,t}(s_2) \end{pmatrix}$$

where $\Delta_t = V_{\lambda,t}(s_1) - V_{\lambda,t}(s_2)$. By Bellman equation, we have:

$$V^*_\lambda(s_1) = \frac{1 + \lambda \log 2}{1 - \gamma}$$

and $V^*_\lambda(s_2)$ satisfies the following equation:

$$V^*_\lambda(s_2) = \frac{\lambda}{1 - \gamma} \log(1 + e^{\frac{\gamma\Delta^*}{\lambda}})$$

where $\Delta^* = V^*_\lambda(s_1) - V^*_\lambda(s_2)$. Thus we have:

$$|V_{\lambda,t+1}(s_1) - V^*_\lambda(s_1)| = |1 + \lambda \log 2 + \gamma V_{\lambda,t}(s_1) - V^*_\lambda(s_1)| = \gamma\,|V_{\lambda,t}(s_1) - V^*_\lambda(s_1)|$$

Noting that:

$$(1 - \gamma)\Delta_{t+1} = 1 + \lambda \log 2 - \lambda \log(1 + e^{\frac{\gamma\Delta_t}{\lambda}})$$

we have:

$$\begin{aligned} V^*_\lambda(s_2) - V_{\lambda,t+1}(s_2) &= V^*_\lambda(s_2) - \lambda \log(1 + e^{\frac{\gamma\Delta_t}{\lambda}}) - \gamma V_{\lambda,t}(s_2) \\ &= V^*_\lambda(s_2) + (1 - \gamma)\Delta_{t+1} - (1 + \lambda \log 2) - \gamma V_{\lambda,t}(s_2) \\ &= \gamma(V^*_\lambda(s_2) - V_{\lambda,t}(s_2)) + (1 - \gamma)(\Delta_{t+1} - \Delta^*) \end{aligned}$$

Re-arranging both sides, we obtain:

$$\begin{aligned} \gamma(V^*_\lambda(s_2) - V_{\lambda,t+1}(s_2)) &= \gamma(V^*_\lambda(s_2) - V_{\lambda,t}(s_2)) + (1 - \gamma)(V_{\lambda,t+1}(s_1) - V^*_\lambda(s_1)) \\ &= \gamma(V^*_\lambda(s_2) - V_{\lambda,t}(s_2)) + (1 - \gamma)\gamma^{t+1}(V_{\lambda,0}(s_1) - V^*_\lambda(s_1)) \end{aligned}$$

Thus, we can also say:

$$|V^*_\lambda(s_2) - V_{\lambda,t}(s_2)| = C \cdot \gamma^t$$

□

Remark B.1. *Theorem B.3 and Theorem B.4 show that the best convergence rate of VI and RVI are both γ^t. Thus, the regularization term can be useless in the worst case in dynamic programming with exact knowledge to transition probabilities.*

B.3. Proof related to Projected Gradient Ascent

Proo of Theorem 3.3. Note that π_λ^* maximizes $V_\lambda^\pi(s)$ for all $s \in \mathcal{S}$. Thus, we have:

$$\begin{aligned} J_\mu(\pi_\lambda^*, \lambda) - J_\mu(\pi, \lambda) &= \mathbb{E}_\mu \left[V_\lambda^{\pi_\lambda^*}(s) - V_\lambda^\pi(s) \right] \\ &\leq \mathbb{E}_\nu \frac{\mu(s)}{\nu(s)} \left[V_\lambda^{\pi_\lambda^*}(s) - V_\lambda^\pi(s) \right] \\ &\leq \left\| \frac{\mu}{\nu} \right\|_\infty (J_\nu(\pi_\lambda^*, \lambda) - J_\nu(\pi, \lambda)) \end{aligned}$$

□

B.3.1. Derivative of $J(\pi, \lambda)$ w.r.t. π

Lemma B.4. *For any $(s, a) \in \mathcal{S} \times \mathcal{A}$,*

$$\frac{\partial J(\pi, \lambda)}{\pi(a|s)} = \frac{1}{1-\gamma} d_{\pi,\rho}(s) \left(Q_\lambda^\pi(s, a) - \lambda \nabla_{s,a} \Omega(\pi(\cdot|s)) \right) \tag{32}$$

where ρ is the initial distribution.

Proof. For given s, a, we have:

$$\frac{\partial V_\lambda^\pi(s)}{\partial \pi(a|s)} = \frac{\partial}{\partial \pi(a|s)} \sum_{a'} \pi(a'|s) Q_\lambda^\pi(s, a') - \lambda \Omega(\pi(\cdot|s)) \tag{33}$$

$$= Q_\lambda^\pi(s, a) - \lambda \nabla_{s,a} \Omega(\pi(\cdot|s)) + \sum_{a'} \pi(a'|s) \frac{\partial Q_\lambda^\pi(s, a')}{\partial \pi(a|s)} \tag{34}$$

$$= Q_\lambda^\pi(s, a) - \lambda \nabla_{s,a} \Omega(\pi(\cdot|s)) + \gamma \sum_{a', s'} \pi(a'|s) P(s'|s, a') \frac{\partial V_\lambda^\pi(s')}{\partial \pi(a|s)} \tag{35}$$

$$\frac{\partial V_\lambda^\pi(\tilde{s})}{\partial \pi(a|s)} = \gamma \sum_{a', s'} \pi(a'|\tilde{s}) P(s'|\tilde{s}, a') \frac{\partial V_\lambda^\pi(s')}{\partial \pi(a|\tilde{s})} \tag{36}$$

By recursively expanding $\frac{\partial V_\lambda^\pi(s')}{\partial \pi(a|s)}$, we get:

$$\frac{\partial J(\pi, \lambda)}{\pi(a|s)} = \frac{1}{1-\gamma} d_{\pi,\rho}(s) \left(Q_\lambda^\pi(s, a) - \lambda \nabla_{s,a} \Omega(\pi(\cdot|s)) \right) \tag{37}$$

□

B.3.2. Smoothness of $J(\pi, \lambda)$ w.r.t. π

The proof technique in this section is the same with (Agarwal et al., 2019). The difference is that we additionally consider the smoothness of regularization rather than the value function alone.

Define $\pi_\alpha(\cdot|s) = \pi(\cdot|s) + \alpha u_s$, where u_s is an arbitrary direction with unit ℓ_2 norm. Our aim is to bound

$$\max_{\|u_s\|_2 = 1, \forall s} \left| \frac{d^2 V_\lambda^{\pi_\alpha}(s)}{d\alpha^2} \bigg|_{\alpha=0} \right| \tag{38}$$

To that end, we have:

$$\frac{d^2 V_\lambda^{\pi_\alpha}(s)}{d\alpha^2} = \sum_a \frac{d^2 \pi_\alpha(a|s)}{d\alpha^2} \left(Q_\lambda^{\pi_\alpha}(a|s) - \lambda \Omega(\pi_\alpha(\cdot|s)) \right) + 2 \sum_a \frac{d\pi_\alpha(a|s)}{d\alpha} \frac{d \left(Q_\lambda^{\pi_\alpha}(a|s) - \lambda \Omega(\pi_\alpha(\cdot|s)) \right)}{d\alpha}$$

$$+\sum_a \pi_\alpha(a|s)\frac{d^2\left(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega(\pi_\alpha(\cdot|s))\right)}{d\alpha^2}, \tag{39}$$

and we are going to bound the above three terms separately. By our definition, the first term is equal to zero because $\frac{d^2\pi_\alpha}{d\alpha^2}=0$.

For the rest, note that $Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega(\pi_\alpha(\cdot|s))=1_{(s,a)}^T M(\alpha)(R-\lambda\Omega_\alpha)$, where $M(\alpha)=(I-\gamma\mathbb{P}(\alpha))^{-1}$, $\Omega_\alpha(s,a)=\Omega(\pi_\alpha(\cdot|s))$ and $\mathbb{P}(\alpha)(s',a'|s,a)=\pi_\alpha(a'|s')\mathbb{P}(s'|s,a)$. What's more,

$$\frac{d\left(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega(\pi_\alpha(\cdot|s))\right)}{d\alpha}=\gamma 1_{(s,a)}^T M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)(R-\lambda\Omega_\alpha)-\lambda 1_{(s,a)}^T M(\alpha)\frac{d\Omega_\alpha}{d\alpha} \tag{40}$$

$$\frac{d^2\left(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\phi(\pi_\alpha(\cdot|s))\right)}{d\alpha^2}=2\gamma^2 1_{(s,a)}^T M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)(R-\lambda\Omega_\alpha) \tag{41}$$

$$-2\lambda\gamma 1_{(s,a)}^T M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)\frac{d\Omega_\alpha}{d\alpha}-\lambda 1_{(s,a)}^T M(\alpha)\frac{d^2\Omega_\alpha}{d\alpha^2} \tag{42}$$

By definition, we have:

$$\|M(\alpha)x\|_\infty \le \frac{1}{1-\gamma}\|x\|_\infty \tag{43}$$

$$\max_{\|u_s\|_2=1}\left\|\frac{d\mathbb{P}(\alpha)}{d\alpha}x\right\|_\infty \le \max_{\|u_s\|_2=1}\max_{(s,a)}\left|\sum_{s',a'}u_{s'}(a')\mathbb{P}(s'|s,a)x_{s',a'}\right| \tag{44}$$

$$\le \max_{\|u_s\|_2=1}\max_{(s,a)}\sum_{s'}\mathbb{P}(s'|s,a)\sum_{a'}|u_{s'}(a')|\,\|x\|_\infty \tag{45}$$

$$\le \sqrt{|\mathcal{A}|}\|x\|_\infty, \tag{46}$$

where the third inequality used $\sum_{a'}|u_{s'}(a')|\le\sqrt{|\mathcal{A}|\sum_{a'}u_{s'}^2(a')}=\sqrt{|\mathcal{A}|}$.

By assumption, $r(s,a)\in[0,1]$, $\Omega_\alpha\in[0,C_\Phi]$, $|\Omega'_\alpha|\in[0,C_\Phi^{(1)}]$, $|\Omega''_\alpha|\in[0,C_\Phi^{(2)}]$, then

$$\begin{aligned}
&\max_{\|u_s\|_2=1}\left|\frac{d(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega_\alpha(\pi(a|s)))}{d\alpha}\right| \\
&\le\gamma\max_{\|u_s\|_2=1}\left\|M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)(R-\lambda\Omega_\alpha)\right\|_\infty+\lambda\max_{\|u_s\|_2=1}\left\|M(\alpha)\frac{d\Omega_\alpha}{d\alpha}\right\|_\infty \\
&\le\frac{\gamma\sqrt{|\mathcal{A}|}(1+\lambda C_\Phi)}{(1-\gamma)^2}+\frac{\lambda C_\Phi^{(1)}}{1-\gamma}
\end{aligned}$$

Hence, the second term can be bounded as:

$$2\left|\sum_a u_{a,s}\frac{d\left(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega(\pi_\alpha(a|s))\right)}{d\alpha}\right|\le 2\sqrt{|\mathcal{A}|}\left[\frac{\gamma\sqrt{|\mathcal{A}|}(1+\lambda C_\Phi)}{(1-\gamma)^2}+\frac{\lambda C_\Phi^{(1)}}{1-\gamma}\right] \tag{47}$$

$$=\frac{2\gamma|\mathcal{A}|(1+\lambda C_\Phi)}{(1-\gamma)^2}+\frac{2\lambda\sqrt{|\mathcal{A}|}C_\Phi^{(1)}}{(1-\gamma)} \tag{48}$$

Next, we consider the third term:

$$\max_{\|u_s\|_2=1}\left|\frac{d^2(Q_\lambda^{\pi_\alpha}(a|s)-\lambda\Omega_\alpha(\pi(\cdot|s)))}{d\alpha^2}\right|\le 2\gamma^2\max_{\|u_s\|_2=1}\left\|M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)(R-\lambda\Omega_\alpha)\right\|_\infty \tag{49}$$

$$+2\lambda\gamma\max_{\|u_s\|_2=1}\left\|M(\alpha)\frac{d\mathbb{P}(\alpha)}{d\alpha}M(\alpha)\frac{d\Omega_\alpha}{d\alpha}\right\|_\infty \tag{50}$$

$$+\lambda\max_{\|u_s\|_2=1}\left\|M(\alpha)\frac{d^2\Omega_\alpha}{d\alpha^2}\right\|_\infty \tag{51}$$

$$\leq \frac{2\gamma^2|\mathcal{A}|(1+\lambda C_\Phi)}{(1-\gamma)^3} + \frac{2\lambda\gamma\sqrt{|\mathcal{A}|}C_\Phi^{(1)}}{(1-\gamma)^2} + \frac{\lambda C_\Phi^{(2)}}{1-\gamma} \tag{52}$$

Finally, we have:

$$\max_{\|u_s\|_2=1}\left|\frac{d^2V_\lambda^{\pi_\alpha}(s)}{d\alpha^2}\Big|_{\alpha=0}\right| \leq \frac{2\gamma|\mathcal{A}|(1+\lambda C_\Phi)}{(1-\gamma)^2} + \frac{2\lambda\sqrt{|\mathcal{A}|}C_\Phi^{(1)}}{(1-\gamma)} + \frac{2\gamma^2|\mathcal{A}|(1+\lambda C_\Phi)}{(1-\gamma)^3} + \frac{2\lambda\gamma\sqrt{|\mathcal{A}|}C_\Phi^{(1)}}{(1-\gamma)^2} + \frac{\lambda C_\Phi^{(2)}}{1-\gamma} \tag{53}$$

$$\leq \frac{2\gamma|\mathcal{A}|(C_1+\lambda C_2)}{(1-\gamma)^3} + \frac{2\lambda\gamma C_2^{(1)}}{(1-\gamma)^2} + \frac{2\lambda C_2^{(1)}}{1-\gamma} + \frac{\lambda C_2^{(2)}}{\sqrt{|\mathcal{A}|}(1-\gamma)} \tag{54}$$

$$\leq \frac{4\gamma|\mathcal{A}|}{(1-\gamma)^3} + \lambda\cdot\frac{4\gamma|\mathcal{A}|C_\Phi + 2(1-\gamma)\sqrt{|\mathcal{A}|}C_\Phi^{(1)} + (1-\gamma)^2C_\Phi^{(2)}}{(1-\gamma)^3} \tag{55}$$

which says V_λ^π is L-smooth w.r.t. π.

B.3.3. Convergence of $J_\mu(\pi,\lambda)$ for a fixed λ

Lemma B.5. *For any given π, λ, denote π_λ^* is the optimal policy for regularized MDP with λ, we have the following equation:*

$$J_\mu(\pi_\lambda^*,\lambda) - J_\mu(\pi,\lambda) = \frac{1}{1-\gamma}\mathbf{E}_{d_{\pi_\lambda^*,\mu}\pi_\lambda^*}[A_\lambda^\pi(s,a) - \lambda\Omega(\pi_\lambda^*(\cdot|s))] \tag{56}$$

where $A_\lambda^\pi(s,a) = Q_\lambda^\pi(s,a) - V_\lambda^\pi(s)$

Proof.

$$J_\mu(\pi_\lambda^*,\lambda) - J_\mu(\pi,\lambda) = \mathbf{E}_{\pi_\lambda^*}\sum_{t=0}^\infty \gamma^t(r(s_t,a_t) - \lambda\Omega(\pi_\lambda^*(\cdot|s_t))) - J_\mu(\pi,\lambda) \tag{57}$$

$$= \mathbf{E}_{\pi_\lambda^*}\sum_{t=0}^\infty \gamma^t\left[r(s_t,a_t) + V_\lambda^\pi(s_t) - V_\lambda^\pi(s_t)\right] - J_\mu(\pi,\lambda) - \lambda\Phi_\mu(\pi_\lambda^*) \tag{58}$$

$$= \mathbf{E}_{\pi_\lambda^*}\sum_{t=0}^\infty \gamma^t\left[r(s_t,a_t) + \gamma V_\lambda^\pi(s_{t+1}) - V_\lambda^\pi(s_t)\right] - \lambda\Phi_\mu(\pi_\lambda^*) \tag{59}$$

$$= \mathbf{E}_{\pi_\lambda^*}\sum_{t=0}^\infty \gamma^t A_\lambda^\pi(s_t,a_t) - \lambda\Phi_\mu(\pi_\lambda^*) \tag{60}$$

$$= \frac{1}{1-\gamma}\mathbf{E}_{d_{\pi_\lambda^*,\mu}\pi_\lambda^*}[A_\lambda^\pi(s,a) - \lambda\Omega(\pi_\lambda^*(\cdot|s))] \tag{61}$$

□

Lemma B.6. *Denote $\pi_{t+1} = Proj(\pi_t + \eta_\pi\nabla_\pi J_\nu(\pi_t,\lambda))$, we have the following improvement guarantee:*

$$J_\nu(\pi_{t+1},\lambda) - J_\nu(\pi_t,\lambda) \geq \frac{2-\eta_\pi L}{2\eta_\pi}\|\pi_{t+1}-\pi_t\|_2^2, \tag{62}$$

where L is the smoothness coefficient of $J_\nu(\pi,\lambda)$.

Proof. By smoothness of $J_\nu(\pi,\lambda)$, we have:

$$J_\nu(\pi_{t+1},\lambda) \geq J_\nu(\pi_t,\lambda) + \langle\nabla_\pi J_\nu(\pi_t,\lambda), \pi_{t+1}-\pi_t\rangle - \frac{L}{2}\|\pi_{t+1}-\pi_t\|_2^2 \tag{63}$$

By first order stationary condition, we have:

$$\langle\pi_{t+1} - \pi_t - \eta_\pi\nabla_\pi J_\nu(\pi_t,\lambda), \pi_{t+1}-\pi_t\rangle \leq 0 \tag{64}$$

So we obtain the final result:

$$J_\nu(\pi_{t+1},\lambda) - J_\nu(\pi_t,\lambda) \geq (\frac{1}{\eta_\pi} - \frac{L}{2})\|\pi_{t+1}-\pi_t\|_2^2. \tag{65}$$

□

Lemma B.7. *Denote* $G(\pi_t,\lambda) = \frac{1}{\eta_\pi}\left[\pi_t - Proj(\pi_t + \eta_\pi \nabla_\pi J_\nu(\pi_t,\lambda))\right]$ *and let* $\eta_\pi = \frac{1}{L}$*, we have:*

$$\min_{t=0,1,\ldots,T-1} \|G(\pi_t,\lambda)\|_2 \leq \sqrt{\frac{2L(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda))}{T}}. \tag{66}$$

Proof. By Lemma B.6, we have $J_\nu(\pi_{t+1},\lambda) - J_\nu(\pi_t,\lambda) \geq \frac{1}{2L}\|G(\pi_t,\lambda)\|_2^2$. By summing over t, we have:

$$\min_{t=0,1,\ldots,T-1} \|G(\pi_t,\lambda)\|_2^2 \leq \frac{2L(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda))}{T} \tag{67}$$

□

Lemma B.8. *If* $J_\nu(\pi,\lambda)$ *is L-smooth w.r.t* π *for a fixed* λ*, the following inequality holds for projected gradient descent when* $\|G(\pi_t,\lambda)\|_2 \leq \varepsilon$*:*

$$\max_{\pi+\delta\in\Delta(\mathbf{A})^{|\mathbf{S}|},\|\delta\|_2\leq 1} \langle \delta, \nabla_\pi J_\nu(\pi_{t+1},\lambda)\rangle \leq \varepsilon(\eta_\pi L + 1) \tag{68}$$

Remark: Lemma B.8 can be referred from (Agarwal et al., 2019).

Theorem B.5. *Let* $\eta_\pi = 1/L_\lambda$ *and Assumption 3.1 hold, we have:*

$$J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_T,\lambda) \leq 4\rho_\nu\sqrt{|\mathcal{S}|}\left(\sqrt{\frac{2L_\lambda(J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_0,\lambda))}{T}}\right) \tag{69}$$

Proof. By Lemma B.5, we have:

$$\begin{aligned}
J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi,\lambda) &= \frac{1}{1-\gamma}\mathbf{E}_{d_{\pi^*_\lambda,\nu}}\langle \pi^*_\lambda(\cdot|s), A^\pi_\lambda(s,\cdot) - \lambda\Omega(\pi^*_\lambda(\cdot|s))\rangle \\
&\leq \frac{1}{1-\gamma}\mathbf{E}_{d_{\pi^*_\lambda,\nu}}\max_{\tilde\pi(\cdot|s)}\langle \tilde\pi(\cdot|s), A^\pi_\lambda(s,\cdot) - \lambda\Omega(\tilde\pi(\cdot|s))\rangle \\
&\leq \frac{1}{1-\gamma}\left[\max_s \frac{d_{\pi^*_\lambda,\nu}(s)}{d_{\pi,\nu}(s)}\right]\mathbf{E}_{d_{\pi,\nu}}\max_{\tilde\pi(\cdot|s)}\langle \tilde\pi(\cdot|s), A^\pi_\lambda(s,\cdot) - \lambda\Omega(\tilde\pi(\cdot|s))\rangle \\
&\leq \frac{\rho_\nu}{1-\gamma}\max_{\tilde\pi}\left[\mathbf{E}_{d_{\pi,\nu},\tilde\pi}A^\pi_\lambda(s,a) - \lambda\Omega(\tilde\pi(\cdot|s))\right]
\end{aligned} \tag{70}$$

where the second inequality holds by $\max_{\tilde\pi(\cdot|s)}\langle\tilde\pi(\cdot|s), A^\pi_\lambda(s,\cdot) - \lambda\Omega(\tilde\pi(\cdot|s))\rangle \geq 0$ (let $\widetilde{\pi} = \pi$), and the final step follows $\tilde\pi(\cdot|s)$ are independent with each state. Next we turn to upper bound $\mathbf{E}_{d_{\pi,\nu},\tilde\pi}A^\pi_\lambda(s,a) - \lambda\Omega(\tilde\pi(\cdot|s))$.

$$\begin{aligned}
\mathbf{E}_{d_{\pi,\nu},\tilde\pi}\left[A^\pi_\lambda(s,a) - \lambda\Omega(\tilde\pi(\cdot|s))\right] &\overset{(a)}{=} \mathbf{E}_{d_{\pi,\nu},\tilde\pi}\left[A^\pi_\lambda(s,a) - \lambda\Omega(\tilde\pi(\cdot|s))\right] - \mathbf{E}_{d_{\pi,\nu},\pi}\left[A^\pi_\lambda(s,a) - \lambda\Omega(\pi(\cdot|s))\right] \\
&\overset{(b)}{=} \mathbf{E}_{d_{\pi,\nu},\tilde\pi}\left[Q^\pi_\lambda(s,a) - \lambda\Omega(\tilde\pi(\cdot|s))\right] - \mathbf{E}_{d_{\pi,\nu},\pi}\left[Q^\pi_\lambda(s,a) - \lambda\Omega(\pi(\cdot|s))\right] \\
&= \mathbf{E}_{d_{\pi,\nu}}\left[\langle\tilde\pi(\cdot|s) - \pi(\cdot|s), Q^\pi_\lambda(s,\cdot)\rangle - \lambda(\Omega(\tilde\pi(\cdot|s)) - \Omega(\pi(\cdot|s)))\right] \\
&\overset{(c)}{\leq} \mathbf{E}_{d_{\pi,\nu}}\left[\langle\tilde\pi(\cdot|s) - \pi(\cdot|s), Q^\pi_\lambda(s,\cdot)\rangle - \lambda\langle\tilde\pi(\cdot|s) - \pi(\cdot|s), \nabla\Omega(\pi(\cdot|s))\rangle\right] \\
&\overset{(d)}{=} (1-\gamma)\langle\tilde\pi - \pi, \nabla J_\nu(\pi,\lambda)\rangle
\end{aligned} \tag{71}$$

where (a) follows from $\sum_a \pi(a|s)A^\pi_\lambda(s,a) - \lambda\Omega(\pi(\cdot|s)) = 0$, (b) follows from $A^\pi_\lambda(s,a) = Q^\pi_\lambda(s,a) - V^\pi_\lambda(s)$, (c) follows since $\Omega(\pi)$ is a convex function, and (d) follows from the definition of $\nabla J_\nu(\pi,\lambda)$.Then we obtain an upper bound of $J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi,\lambda)$:

$$J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi,\lambda) \leq \rho_\nu\max_{\tilde\pi}\langle\tilde\pi - \pi, \nabla J_\nu(\pi,\lambda)\rangle$$

$$\leq 2\rho_\nu\sqrt{|\mathcal{S}|}\max_{\pi+\delta\in\Delta(\mathcal{A})^{|\mathcal{S}|},\|\delta\|_2\leq 1}\langle\delta,\nabla_\pi J_\nu(\pi,\lambda)\rangle \tag{72}$$

where the final inequality holds as $\|\tilde{\pi}-\pi\|_2\leq 2\sqrt{|\mathcal{S}|}$ and:

$$\max_{\tilde{\pi}}\langle\tilde{\pi}-\pi,\nabla J(\pi,\lambda)\rangle\leq 2\sqrt{|\mathcal{S}|}\max_{\pi+\delta\in\Delta(\mathcal{A})^{|\mathcal{S}|},\|\delta\|_2\leq 1}\langle\delta,\nabla_\pi J_\nu(\pi,\lambda)\rangle$$

By Lemma B.8 and Lemma B.7, we have:

$$\min_{t=0,1,\ldots,T-1}\max_{\pi_t+\delta\in\Delta(\mathcal{A})^{|\mathcal{S}|},\|\delta\|_2\leq 1}\langle\delta,\nabla_\pi J_\nu(\pi_{t+1},\lambda)\rangle\leq 2\sqrt{\frac{2L_\lambda(J_\nu(\pi^*_\lambda,\lambda)-J_\nu(\pi_0,\lambda))}{T}} \tag{73}$$

Gathering all these results together, we have:

$$\begin{aligned}J_\nu(\pi^*_\lambda,\lambda)-J_\nu(\pi_T,\lambda)&\leq\min_{t=0,1,\ldots,T-1}J_\nu(\pi^*_\lambda,\lambda)-J_\nu(\pi_{t+1},\lambda)\\&\leq 4\rho_\nu\sqrt{|\mathcal{S}|}\left(\sqrt{\frac{2L_\lambda(J_\nu(\pi^*_\lambda,\lambda)-J_\nu(\pi_0,\lambda))}{T}}\right)\end{aligned} \tag{74}$$

where the first inequality holds by Lemma B.6. □

B.3.4. Proof of Corollary 3.2

Proof. By Equation (6) and Theorem 3.4, we have:

$$\begin{aligned}J_\nu(\pi^*_{\lambda_t},\lambda_t)-J_\nu(\hat{\pi}_{t+1},\lambda_t)&\leq 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{2L_{\lambda_t}D_t}{T}}\\&\leq 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{8\lambda_0 L_{\lambda_t}C_\Phi}{2^t(1-\gamma)T}}\end{aligned} \tag{75}$$

Let the RHS of above inequality equals $\frac{\lambda_0}{2^t}\frac{C_\phi}{1-\gamma}$, then the total time satisfies Prop-II$(\hat{\varepsilon})$ at timestep t is at most:

$$\mathsf{Time}(\lambda_t)\leq\frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t \tag{76}$$

□

B.3.5. Proof of Theorem 3.5

Proof. In order to obtain an ε-optimal policy w.r.t. initial distribution μ, we have to get an ε/ρ-optimal policy w.r.t. initial distribution ν at first by Theorem 3.3. By Corollary 3.2, we can obtain an ε-optimal policy in total time:

$$\sum_{t=0}^{T-1}\mathsf{Time}(\lambda_t)\leq\sum_{t=0}^{T-1}\frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t \tag{77}$$

Note that $L_\lambda\leq\frac{4|\mathcal{A}|}{(1-\gamma)^3}+\lambda\cdot\frac{|\mathcal{A}|\widetilde{C}_\Phi}{(1-\gamma)^3}$, where $\widetilde{C}_\Phi=4\gamma C_\Phi+2(1-\gamma)C_\Phi^{(1)}+(1-\gamma)^2C_\Phi^{(2)}$, and $T=O(\log_2\frac{6\rho\lambda_0 C_\Phi}{\varepsilon(1-\gamma)})$, thus the total time is:

$$\begin{aligned}\sum_{t=0}^{T-1}\mathsf{Time}(\lambda_t)&\leq\frac{128|\mathcal{S}|\rho_\nu^2(1-\gamma)}{\lambda_0 C_\Phi}\sum_{t=0}^{T-1}L_{\lambda_t}2^t\\&\leq\frac{128|\mathcal{S}||\mathcal{A}|\rho_\nu^2}{(1-\gamma)^2\lambda_0 C_\Phi}(2^T+\lambda_0 C_2 T)\\&=O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)\end{aligned} \tag{78}$$

□

B.3.6. SAMPLING

In reality, it is unrealistic to obtain the exact value of gradient at each time-step. A more practical way is to estimate gradients with sufficient samples. In this scenario, convergence is guaranteed when we have enough samples. Given a ν-restart model, we always start from an initial state $s_0 \sim \nu$. Thus, by interacting with MDP under a given policy π, we can obtain independent trajectories and approximate the value of $J_\nu(\pi, \lambda)$. However, to obtain an estimation of gradient $\nabla J_\nu(\pi, \lambda)$, we need to obtain an initial state $s_0 \sim d_{\pi,\nu}$ and start interacting with MDP, which is unattainable when transition probability is unknown. Fortunately, Kakade and Langford (2002); Shani et al. (2019) proposed an approximate method. In particular, one draws a start state s from $\nu(s)$, and accepts it as the initial state with probability $1-\gamma$. Otherwise, one transits it to next state with probability γ. The process is repeated until an acceptance is made. In process of one trajectory sampling, one also stops interaction when the length of this trajectory achieves $K = \widetilde{O}\left(\frac{1}{1-\gamma}\right)$, which is known as an effective horizon.

Theorem B.6 (Convergence of Projected Gradient Ascent with Sampling)**.** *Under the same setting in Theorem 3.4 and letting truncated length of trajectories be $K = \log \frac{12(1+\lambda C_\Phi)}{\varepsilon(1-\gamma)^2} / \log \frac{1}{\gamma}$, we have that with probability $1-\delta$ and number of trajectories $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon^2(1-\gamma)^4}\right)$ at each time-step t:*

$$\begin{aligned}&\min_{t=0,\ldots,T-1} J_\nu(\pi^*_\lambda, \lambda) - J_\nu(\pi_{t+1}, \lambda)\\ &\leq 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{L_\lambda(J(\pi^*_\lambda, \lambda) - J(\pi_0, \lambda))}{T} + \varepsilon}.\end{aligned} \tag{79}$$

In order to make the LHS equal to ε'/ρ, which guarantees ε'-optimal policy w.r.t. initial distribution μ, the total iteration complexity could be $T = O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon'^2(1-\gamma)^4}\right)$ and the total number of sampled trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^6\rho_\nu^6(1+\lambda\widetilde{C}_\Phi)^4}{\varepsilon'^6(1-\gamma)^8}\right)$.

In Theorem B.6, the convergence is guaranteed with high probability. Suppose we could obtain $\pi_{\widehat{t}}$ attaining the minimum of RHS of Eq. (79). We can take projected gradient ascent with sampling as a sub-solver in Algorithm 1, which also accelerates iteration and reduces sample complexity by a factor of $\frac{1}{\varepsilon(1-\gamma)}$ as shown in Theorem B.7.

Theorem B.7. *Suppose $\widehat{\pi}_{t+1}$ attains the minimum of $J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_{k+1}, \lambda_t)$ over $k = 0, ..., T_t - 1$ at each time-step t in Algorithm 1, where $T_t = \mathsf{Time}(\lambda_t)$. Under the same setting in Theorem B.6, Alg taking Algorithm 2 with sampling as sub-solver leads to an ε-optimal policy w.r.t. initial distribution μ. Moreover, with probability $1-\delta$, the total iteration complexity is $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$ and the number of trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^5\rho_\nu^6(1+\lambda_0\widetilde{C}_\Phi)^3}{\varepsilon^5(1-\gamma)^7}\right)$.*

Theorem B.7 shows that we can find a $\widehat{\pi}_{t+1}$ that attains the minimum difference in sub-solver. In practice, we can use $\{\pi_{i+1}\}_{i=0}^{T_t-1}$ to simulate another trajectories to evaluate $J(\pi_{i+1}, \lambda_t)$. As long as we have enough samples, it is guaranteed that $\widehat{J}(\pi_{i+1}, \lambda_t) \approx J(\pi_{t+1}, \lambda_t)$ as Theorem B.8 shows. Compared with gradient estimation which requires $\widetilde{O}\left(\frac{1}{\varepsilon^4(1-\gamma)^4}\right)$ trajectories for each policy, value estimation requires samples $\widetilde{O}\left(\frac{1}{\varepsilon^2(1-\gamma)^2}\right)$ to achieve the same statistical error level. Thus, we can select a policy $\widehat{\pi}_{t+1}$ attaining the maximum estimated value $\widehat{J}(\pi_{k+1}, \lambda_t)$ over $k = 0, ..., T_t - 1$ and take it as the initial policy in time-step $t+1$.

Theorem B.8. *At time-step t, with probability $1-2\delta$ and $N = \frac{2(1+\lambda_t C_\Phi)^2}{\varepsilon^2(1-\gamma)^2}\log\frac{T_t}{\delta}$ i.i.d. trajectories for each policy π_i, we have that*

$$\begin{aligned}&J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_{\widehat{i}}, \lambda_t)\\ &\leq \min_{i=0,\ldots,T_t-1} J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_{i+1}, \lambda_t) + 2\varepsilon,\end{aligned} \tag{80}$$

where $\widehat{i} = \operatorname{argmax}_{i=0,\ldots,T_t-1} \widehat{J}_\nu(\pi_{i+1}, \lambda_t)$.

Next, we turn to prove our results of sampling. To simplify, we denote $\widetilde{\nabla} J_\nu(\pi, \lambda)$ as an estimator of $\nabla J_\nu(\pi, \lambda)$, where $\nabla_{s,a} J_\nu(\pi, \lambda) = \frac{1}{1-\gamma} d_{\pi,\nu}(s)(Q^\pi_\lambda(s,a) - \lambda\nabla_{s,a}\Omega(\pi(\cdot|s))$.

Lemma B.9. *Denote $\pi_{t+1} = Proj(\pi_t + \eta_\pi \widetilde{\nabla}_\pi J_\nu(\pi_t, \lambda))$, we have the following improvement guarantee:*

$$J_\nu(\pi_{t+1}, \lambda) - J_\nu(\pi_t, \lambda) \geq \frac{2 - \eta_\pi L_\lambda}{2\eta_\pi}\|\pi_{t+1} - \pi_t\|_2^2 + \varepsilon_t, \tag{81}$$

where L_λ is the smoothness coefficient of $J_\nu(\pi,\lambda)$ and $\varepsilon_t = \langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \pi_{t+1}-\pi_t\rangle$.

Proof. By L-smooth, we have:

$$J_\nu(\pi_{t+1},\lambda) \geq J_\nu(\pi_t,\lambda) + \langle \widetilde{\nabla} J_\nu(\pi_t,\lambda), \pi_{t+1}-\pi_t\rangle - \frac{L}{2}\|\pi_{t+1}-\pi_t\|_2^2 + \varepsilon_t \tag{82}$$

By first-order condition, we have:

$$\langle \pi_{t+1} - \pi_t - \eta\widetilde{\nabla} J_\nu(\pi_t,\lambda), \pi_{t+1}-\pi_t\rangle \leq 0 \tag{83}$$

Combining these two together, we obtain desired result. □

It's not easy to obtain exact knowledge of $d_{\pi,\nu}$. But, we can obtain an another $d_{\pi,\nu,K}$ to approximate $d_{\pi,\nu}$. According to Kakade and Langford (2002); Shani et al. (2019), we draw a start state s from $\nu(s)$, and accept it as initial state for trajectory simulation with probabilty $1-\gamma$. Otherwise, we transits it to next state with probability γ. We repeat the process until an acceptance is made within time K. In mathematical form, $d_{\pi,\nu,K}(s) = (1-\gamma)\nu(s) + (1-\gamma)\sum_{t=1}^{K-1}\gamma^t P(s_t = s|\nu,\pi) + \gamma^K P(s_T = s|\nu,\pi)$.

From a state-action pair $(s_0,a_0) \sim d_{\pi,\nu,K} \times \mathcal{U}(\mathcal{A})$, we can simulate a truncated trajectory $(s_0,a_0,s_1,a_1,...,s_{K-1},a_{K-1})$ with given policy π. Denote $\widehat{Q}_\lambda^\pi(s_0,a_0) = R(s_0,a_0) + \sum_{t=1}^{K-1}\gamma^t(R(s_t,a_t) - \lambda\Omega(\pi(\cdot|s_t)))$, which is an unbiased estimator of

$$\overline{Q}_\lambda^\pi(s_0,a_0) = \mathbb{E}_{\pi,\mathbb{P}}\left[R(s_0,a_0) + \sum_{t=1}^{K-1}\gamma^t(R(s_t,a_t) - \lambda\Omega(\pi(\cdot|s_t)))\right] \tag{84}$$

Note that $\overline{Q}_\lambda^\pi$ could be close enough with Q_λ^π when K is sufficiently large. Thus, we can derive an estimator of $\nabla_{s,a}J_\nu(\pi,\lambda)$, with N independent trajectories, as:

$$\widetilde{\nabla}_{s,a}J_\nu(\pi,\lambda) = \frac{|\mathcal{A}|}{1-\gamma}\frac{1}{N}\sum_{n=1}^{N}\left[\widehat{Q}_\lambda^\pi(s_{0,n},a_{0,n}) - \lambda\nabla_{s,a}\Omega(\pi(\cdot|s_{0,n}))\right] I\{s_{0,n}=s, a_{0,n}=a\} \triangleq \frac{1}{N}\sum_{n=1}^{N}\widehat{X}_n(s,a) \tag{85}$$

Lemma B.10. *For fixed $\widetilde{\pi}$, we have:*

$$P\left(\left|\langle\widetilde{\nabla}J_\nu(\pi,\lambda) - \mathbb{E}\widetilde{\nabla}J_\nu(\pi,\lambda), \widetilde{\pi}-\pi\rangle\right| \geq \varepsilon\right) \leq 2\exp\left(-\frac{N\varepsilon^2(1-\gamma)^4}{2|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}\right) \tag{86}$$

where $\mathbb{E}\widetilde{\nabla}_{s,a}J_\nu(\pi,\lambda) = \frac{1}{1-\gamma}d_{\pi,\nu,K}(s)\left(\overline{Q}_\lambda^\pi(s,a) - \lambda\nabla_{s,a}\Omega(\pi(\cdot|s))\right)$.

Proof. First of all, we proof that $\langle\widehat{X}_n, \widetilde{\pi}-\pi\rangle$ is bounded:

$$\begin{aligned}
|\langle\widehat{X}_n, \widetilde{\pi}-\pi\rangle| &= \left|\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\widehat{X}_n(s,a)(\widetilde{\pi}(a|s)-\pi(a|s))\right| \\
&\leq \frac{|\mathcal{A}|}{1-\gamma}\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\|\widehat{Q}_\lambda^\pi - \lambda\nabla\Omega(\pi)\|_\infty|\widetilde{\pi}(a|s)-\pi(a|s)|I\{s_{0,k}=s, a_{0,k}=a\} \\
&\leq \frac{|\mathcal{A}|}{1-\gamma}\|\widehat{Q}_\lambda^\pi - \lambda\nabla\Omega(\pi)\|_\infty\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}|\widetilde{\pi}(a|s)-\pi(a|s)|I\{s_{0,k}=s\} \\
&\leq \frac{2|\mathcal{A}|(1+\lambda\widetilde{C_\Phi})}{(1-\gamma)^2}
\end{aligned} \tag{87}$$

By Hoeffding's inequality, we have:

$$P\left(\left|\langle\widetilde{\nabla}J_\nu(\pi,\lambda) - \mathbb{E}\widetilde{\nabla}J_\nu(\pi,\lambda), \widetilde{\pi}-\pi\rangle\right| \geq \varepsilon\right) \leq 2\exp\left(-\frac{N\varepsilon^2(1-\gamma)^4}{2|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}\right) \tag{88}$$

□

Lemma B.11. *For any $\pi, \widetilde{\pi}$, we have:*

$$\left|\langle \nabla J_\nu(\pi,\lambda) - \mathbb{E}\widetilde{\nabla} J_\nu(\pi,\lambda), \widetilde{\pi} - \pi\rangle\right| \le \frac{6\gamma^K(1+\lambda C_\Phi)}{(1-\gamma)^2} \tag{89}$$

Proof. By definition, we decompose the error $\left|\langle \nabla J_\nu(\pi,\lambda) - \mathbb{E}\widetilde{\nabla} J_\nu(\pi,\lambda), \widetilde{\pi} - \pi\rangle\right|$ into two parts I_1, I_2:

$$I_1 = \left|\sum_{s\in\mathcal{S}} \frac{1}{1-\gamma} d_{\pi,\nu}(s) \langle Q_\lambda^\pi(s,\cdot) - \overline{Q}_\lambda^\pi(s,\cdot), \widetilde{\pi}(\cdot|s) - \pi(\cdot|s)\rangle\right|$$
$$I_2 = \left|\sum_{s\in\mathcal{S}} \frac{1}{1-\gamma} \left(d_{\pi,\nu}(s) - d_{\pi,\nu,T}(s)\right) \langle \overline{Q}_\lambda^\pi(s,\cdot), \widetilde{\pi}(\cdot|s) - \pi(\cdot|s)\rangle\right|$$

For I_1, we have:

$$\begin{aligned} I_1 &\le \sum_{s\in\mathcal{S}} \frac{1}{1-\gamma} d_{\pi,\nu}(s) \left|\langle \overline{Q}_\lambda^\pi(s,\cdot) - Q_\lambda^\pi(s,\cdot), \widetilde{\pi}(\cdot|s) - \pi(\cdot|s)\rangle\right| \\ &\le \frac{1}{1-\gamma} \sum_{s\in\mathcal{S}} d_{\pi,\nu}(s) \left\|\overline{Q}_\lambda^\pi(s,\cdot) - Q_\lambda^\pi(s,\cdot)\right\|_\infty \left\|\widetilde{\pi}(\cdot|s) - \pi(\cdot|s)\right\|_1 \\ &\le \frac{2}{1-\gamma} \left\|\overline{Q}_\lambda^\pi - Q_\lambda^\pi\right\|_\infty \\ &\le \frac{2\gamma^K(1+\lambda C_\Phi)}{(1-\gamma)^2} \end{aligned} \tag{90}$$

For I_2, we mainly concern the difference between $d_{\pi,\nu}$ and $d_{\pi,\nu,T}$:

$$\begin{aligned} \sum_{s\in\mathcal{S}} |d_{\pi,\nu}(s) - d_{\pi,\nu,K}(s)| &= \sum_{s\in\mathcal{S}} \left|(1-\gamma)\sum_{t=K}^\infty \gamma^t P(s_t = s|\nu,\pi) - \gamma^K P(s_K = s|\nu,\pi)\right| \\ &\le (1-\gamma)\sum_{s\in\mathcal{S}}\sum_{t=K}^\infty \gamma^t P(s_t = s|\nu,\pi) + \gamma^K \sum_{s\in\mathcal{S}} P(s_K = s|\nu,\pi) \\ &= 2\gamma^K \end{aligned} \tag{91}$$

Thus, we have:

$$\begin{aligned} I_2 &\le \sum_{s\in\mathcal{S}} \frac{1}{1-\gamma} |d_{\pi,\nu}(s) - d_{\pi,\nu,T}(s)| \left|\langle \overline{Q}_\lambda^\pi(s,\cdot), \widetilde{\pi}(\cdot|s) - \pi(\cdot|s)\rangle\right| \\ &\le \frac{2}{1-\gamma} \sum_{s\in\mathcal{S}} |d_{\pi,\nu}(s) - d_{\pi,\nu,T}(s)| \, \|\overline{Q}_\lambda^\pi\|_\infty \\ &\le \frac{4\gamma^K(1+\lambda C_\Phi)}{(1-\gamma)^2} \end{aligned} \tag{92}$$

Combining I_1, I_2 together, we have:

$$\left|\langle \nabla J_\nu(\pi,\lambda) - \mathbb{E}\widetilde{\nabla} J_\nu(\pi,\lambda), \widetilde{\pi} - \pi\rangle\right| \le \frac{6\gamma^K(1+\lambda C_\Phi)}{(1-\gamma)^2} \tag{93}$$

□

Lemma B.12. *Fixing randomness before π_t and letting $K = \log \frac{12(1+\lambda C_\Phi)}{\varepsilon(1-\gamma)^2} / \log \frac{1}{\gamma}$, with probability $1-\delta$ and trajectory sample size $N = \frac{8|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon^2(1-\gamma)^4}\left(|\mathcal{S}|\log|\mathcal{A}| + \log\frac{2}{\delta}\right)$, we have:*

$$|\varepsilon_t| \le \varepsilon \tag{94}$$

Proof. By definition of ε_t, we have:

$$\begin{aligned} P\left(|\varepsilon_t| > \varepsilon\right) &= P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \pi_{t+1} - \pi_t\rangle\right| > \varepsilon\right) \\ &\le P\left(\max_{\widetilde{\pi}} \left|\langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\rangle\right| > \varepsilon\right) \\ &\le |\mathcal{A}|^{|\mathcal{S}|} P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\right\rangle| > \varepsilon\right) \end{aligned} \tag{95}$$

where the last inequality holds that $\widetilde{\pi}$ is the maximum point of a linear function, which is deterministic, and the number of deterministic policies are $|\mathcal{A}|^{|\mathcal{S}|}$. Thus, we continue to bound the following term:

$$\begin{aligned} &P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\right\rangle| > \varepsilon\right) \\ =&P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \mathbb{E}\widetilde{\nabla} J_\nu(\pi_t,\lambda) + \mathbb{E}\widetilde{\nabla} J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\rangle\right| > \varepsilon\right) \\ \le&P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \mathbb{E}\widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\rangle\right| + \left|\langle \mathbb{E}\widetilde{\nabla} J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\rangle\right| > \varepsilon\right) \end{aligned} \tag{96}$$

By setting $K = \log \frac{12(1+\lambda C_\Phi)}{\varepsilon(1-\gamma)^2} / \log \frac{1}{\gamma}$ in Lemma B.11, we have:

$$\begin{aligned} P\left(\left|\langle \nabla J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\right\rangle| > \varepsilon\right) &\le P\left(\left|\langle \mathbb{E}\widetilde{\nabla} J_\nu(\pi_t,\lambda) - \widetilde{\nabla} J_\nu(\pi_t,\lambda), \widetilde{\pi} - \pi_t\right\rangle| > \frac{\varepsilon}{2}\right) \\ &\le 2\exp\left(-\frac{N\varepsilon^2(1-\gamma)^4}{8|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}\right) \end{aligned} \tag{97}$$

where the last inequality follows from Lemma B.10. Combining all these together and taking sample size $N = \frac{8|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon^2(1-\gamma)^4}\left(|\mathcal{S}|\log|\mathcal{A}| + \log\frac{2}{\delta}\right)$, we obtain:

$$P\left(|\varepsilon_t| > \varepsilon\right) \le \delta \tag{98}$$

□

Lemma B.13. *Taking $\eta = 1/L_\lambda$, with probability $1-\delta$ and the number of trajectories from $t = 0, ..., T-1$ being $\frac{8|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2 T}{\varepsilon^2(1-\gamma)^4}\left(|\mathcal{S}|\log|\mathcal{A}| + \log\frac{2T}{\delta}\right)$, we have:*

$$\min_{t=0,\dots,T-1} \|G(\pi_t,\lambda)\|_2^2 \le \frac{2L_\lambda(J_\nu(\pi_\lambda^*,\lambda) - J_\nu(\pi_0,\lambda))}{T} + \varepsilon \tag{99}$$

where $G(\pi_t,\lambda) = (\pi_{t+1} - \pi_t)/\eta$.

Proof. By Lemma B.9, we have:

$$\min_{t=0,\dots,T-1} \|G(\pi_t,\lambda)\|_2^2 \le \frac{2L_\lambda(J_\nu(\pi_\lambda^*,\lambda) - J_\nu(\pi_0,\lambda))}{T} - \frac{\sum_{t=0}^{T-1}\varepsilon_t}{T} \tag{100}$$

Besides, denote $\mathcal{F}_{t+1} = \sigma\left(\mathcal{F}_t \cup \sigma(\{X_{t,n}\}_{n=1}^N)\right)$ and $\mathcal{F}_0$ contains all information before π_0, where $X_{n,t}$ are i.i.d random variables drawn from π_t and environment. Thus, we have:

$$\begin{aligned} P\left(\left|\frac{\sum_{t=0}^{T-1}\varepsilon_t}{T}\right| > \varepsilon\right) &\le \sum_{t=0}^{T-1} P\left(|\varepsilon_t| > \varepsilon\right) \\ &= \sum_{t=0}^{T-1} \mathbb{E}\left[P(|\varepsilon_t| > \varepsilon|\mathcal{F}_t)\right] \\ &\le \delta \end{aligned} \tag{101}$$

where the last inequality follows from Lemma B.12 with number of trajectories being $N_t = \frac{8|\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon^2(1-\gamma)^4}\left(|\mathcal{S}|\log|\mathcal{A}| + \log\frac{2T}{\delta}\right)$ at each step $t = 0, ..., T-1$. □

Theorem B.9. *Fixing λ and taking $\eta = 1/L_\lambda$, with probability $1-\delta$ and number of trajectories $\widetilde{O}\left(\frac{|\mathcal{S}||\mathcal{A}|^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon^2(1-\gamma)^4}\right)$ at each time-step t, we have:*

$$\min_{t=0,\dots,T-1} J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_{t+1},\lambda) \le 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{L_\lambda(J(\pi^*_\lambda,\lambda)-J(\pi_0,\lambda))}{T}+\varepsilon} \tag{102}$$

In order to make the LHS equaling with ε', the total iteration complexity could be $T = O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho_\nu^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon'^2(1-\gamma)^4}\right)$ and the total number of sampled trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho_\nu^6(1+\lambda\widetilde{C}_\Phi)^4}{\varepsilon'^6(1-\gamma)^8}\right)$

Proof. By proof of Theorem B.5, with probability $1-\delta$, we have:

$$\begin{aligned}
\min_{t=0,\dots,T-1} J_\nu(\pi^*_\lambda,\lambda) - J_\nu(\pi_{t+1},\lambda) &\le 4\rho_\nu\sqrt{|\mathcal{S}|}\min_{t=0,\dots,T-1}\|G(\pi_t,\lambda)\|_2 \\
&\le 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{2L_\lambda(J(\pi^*_\lambda,\lambda)-J(\pi_0,\lambda))}{T}+\varepsilon} \\
&\le 4\rho_\nu\sqrt{|\mathcal{S}|}\left(\sqrt{\frac{2L_\lambda(J(\pi^*_\lambda,\lambda)-J(\pi_0,\lambda))}{T}}+\sqrt{\varepsilon}\right)
\end{aligned} \tag{103}$$

By taking $4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\varepsilon} \le \frac{\varepsilon'}{2}$, the number of trajectories at each time-step is $\widetilde{O}\left(\frac{|\mathcal{S}|^3|\mathcal{A}|^2\rho_\nu^4(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon'^4(1-\gamma)^4}\right)$. Besides, in order to make the LHS less than ε', then $T = O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho_\nu^2(1+\lambda\widetilde{C}_\Phi)^2}{\varepsilon'^2(1-\gamma)^4}\right)$ is enough. The total number of trajectories could be $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho_\nu^6(1+\lambda\widetilde{C}_\Phi)^4}{\varepsilon'^6(1-\gamma)^8}\right)$. □

Finally, we consider combining Algorithm 1 with Projected Gradient Ascent serving as sub-solver and only samples can be accessed. Besides, we also assume we can find $\arg\min_{t=0,\dots,T-1} J(\pi^*_\lambda,\lambda) - J(\pi_t,\lambda)$ firstly, and later we will relax this assumption.

Lemma B.14. *At each time-step t with λ_t, taking $\eta = 1/L_{\lambda_t}$, $T_t = \frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t$ and $\varepsilon = \frac{\widehat{\varepsilon}^2}{16|\mathcal{S}|\rho_\nu^2}$, with probability $1-\delta_t$ we have:*

$$\min_{k=0,\dots,K-1} J_\nu(\pi^*_{\lambda_t},\lambda) - J_\nu(\pi_{k+1},\lambda) \le \frac{\lambda_0 C_\Phi}{2^t(1-\gamma)} + \widehat{\varepsilon} \tag{104}$$

Proof. By Theorem B.9, we have:

$$\min_{k=0,\dots,T_t-1} J_\nu(\pi^*_{\lambda_t},\lambda_t) - J_\nu(\pi_{k+1},\lambda_t) \le 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{2L_{\lambda_t}D_t}{T_t}+\varepsilon} \le 4\rho_\nu\sqrt{|\mathcal{S}|}\sqrt{\frac{8\lambda_0 L_{\lambda_t}C_\Phi}{2^t(1-\gamma)T_t}+\frac{2L_{\lambda_t}\widehat{\varepsilon}}{T_t}+\varepsilon}$$

By taking $T_t = \frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t$ and $\varepsilon = \frac{\widehat{\varepsilon}^2}{16|\mathcal{S}|\rho_\nu^2}$, we have:

$$\min_{k=0,\dots,T_t-1} J_\nu(\pi^*_{\lambda_t},\lambda) - J_\nu(\pi_{k+1},\lambda) \le \frac{\lambda_0 C_\Phi}{2^t(1-\gamma)} + \widehat{\varepsilon} \tag{105}$$

□

Theorem B.10. *Suppose we can obtain $\widehat{\pi}_{t+1}$ attaining minimum of $J_\nu(\pi^*_{\lambda_t},\lambda_t) - J_\nu(\pi_{k+1},\lambda_t)$ over $k = 0,...,T_t-1$ at each time-step t. To obtain an ε-optimnal policy w.r.t initial distribution μ, with probability $1-\delta$, the total iteration complexity is $O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$ and the number of trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^5\rho_\nu^6(1+\lambda_0\widetilde{C}_\Phi)^6}{\varepsilon^5(1-\gamma)^7}\right)$.*

Proof. By Lemma B.14, the iteration complexity at each timestep t is $\mathsf{Time}(\lambda_t) = \frac{128|\mathcal{S}|\rho_\nu^2 L_{\lambda_t}(1-\gamma)}{\lambda_0 C_\Phi}2^t$, which is the same as Corollary 3.2. Thus, the total iteration complexity could be $O\left(\frac{|\mathcal{S}||\mathcal{A}|\rho\rho_\nu^2\widetilde{C}_\Phi}{\varepsilon(1-\gamma)^3}\right)$ by taking $T = O(\log_2\frac{12\rho\lambda_0 C_\Phi}{\varepsilon(1-\gamma)})$. By Theorem 3.2 and Theorem 3.3, the final output of Algorithm 1 satisfies:

$$V^*(\mu) - V^{\widehat{\pi}_T}(\mu) \le \frac{\varepsilon}{2} + \rho\widehat{\varepsilon} \tag{106}$$

In order to obatian an ε-optimal policy, the number of trajectories we have to sample at each time-step in Projected Gradient Ascent is $\widetilde{O}\left(\frac{|\mathcal{S}|^3|\mathcal{A}|^2\rho^4\rho_\nu^4(1+\lambda_0\widetilde{C}_\Phi)^2}{\varepsilon^4(1-\gamma)^4}\right)$ by Theorem B.9. Thus, the total number of trajectories is $\widetilde{O}\left(\frac{|\mathcal{S}|^4|\mathcal{A}|^3\rho^5\rho_\nu^6(1+\lambda_0\widetilde{C}_\Phi)^3}{\varepsilon^5(1-\gamma)^7}\right)$, where we ignore logarithm terms, especially $\log\left(\frac{T\sum_{t=1}^T \mathsf{Time}(\lambda_t)}{\delta}\right)$. □

However, $\widehat{\pi}_t$ cannot be obtained directly by $J_\nu(\pi,\lambda)$. Thus we evaluate $J_\nu(\pi,\lambda)$ by re-sampling as $\widehat{J}_\nu(\pi,\lambda)$.

Lemma B.15. *Suppose $\{X_{i,t}\}_{i=1}^N$ are i.i.d and bounded by $[0,M]$ for each t. With sample complexity $N=\frac{M^2}{2\varepsilon^2}\log\frac{T}{\delta}$ and probability $1-\delta$,*

$$\max_{t=0,\dots,T-1}\frac{1}{N}\sum_{i=1}^N X_{i,t}\geq \max_{t=0,\dots,T-1}\mu_t-\varepsilon \tag{107}$$

where μ_t is mean of $X_{i,t}$.

Proof. In fact we have:

$$\begin{aligned}P\left(\max_{t=0,\dots,T-1}\frac{1}{N}\sum_{i=1}^N X_{i,t}<\max_{t=0,\dots,T-1}\mu_t-\varepsilon\right)&\leq P\left(\exists \text{t, s.t. } \frac{1}{N}\sum_{i=1}^N X_{i,t}<\mu_t-\varepsilon\right)\\&\leq\sum_{t=0}^{T-1}P\left(\frac{1}{N}\sum_{i=1}^N X_{i,t}<\mu_t-\varepsilon\right)\\&\leq T\exp\left(-\frac{2\varepsilon^2N}{M^2}\right)\end{aligned} \tag{108}$$

Thus, taking $N=\frac{M^2}{2\varepsilon^2}\log\frac{T}{\delta}$ is enough. □

Lemma B.16. *Under the same setting in Lemma B.15, denote $\widehat{t}=\mathrm{argmax}_{t=0,\dots,T-1}\frac{1}{N}\sum_{i=1}^N X_{i,t}$, with probability $1-2\delta$, we have:*

$$\mu_{\widehat{t}}\geq\max_{t=0,\dots,T-1}\mu_t-2\varepsilon \tag{109}$$

Proof. By Lemma B.15 and symmetric property, we have:

$$P\left(\frac{1}{N}\sum_{i=1}^N X_{i,\widehat{t}}>\mu_{\widehat{t}}+\varepsilon\right)\leq\frac{\delta}{T} \tag{110}$$

Thus, we have:

$$\begin{aligned}P\left(\mu_{\widehat{t}}<\max_{t=0,\dots,T-1}\mu_t-2\varepsilon\right)&\leq P\left(\frac{1}{N}\sum_{i=1}^N X_{i,\widehat{t}}>\mu_{\widehat{t}}+\varepsilon\right)+P\left(\max_{t=0,..,T-1}\mu_t-\varepsilon>\max_{t=0,..,T-1}\frac{1}{N}\sum_{i=1}^N X_{i,t}\right)\\&\leq\frac{1+T}{T}\delta\leq 2\delta\end{aligned} \tag{111}$$

□

At time-step t, the output policies of Projected Gradient Ascent are $\pi_0,...,\pi_{T_t-1}$. For each policy π_k, we interact with environment to obtain n trajectories $(s_{n,0},a_{n,0},...s_{n,K-1},a_{n,K-1})$ from initial distribution ν and policy π_k, and evaluate $J_\nu(\pi_k,\lambda)$ by:

$$\widehat{J}(\pi_k,\lambda)=\frac{1}{N}\sum_{n=1}^N\widehat{Q}_\lambda^{\pi_k}(s_{n,0},a_{n,0})-\lambda\Omega(\pi_k(\cdot|s_{n,0})) \tag{112}$$

where $\widehat{Q}_\lambda^{\pi_k}(s_{n,0}, a_{n,0}) = R(s_{n,0}, a_{n,0}) + \sum_{k=1}^{K-1} \gamma^t (R(s_{n,t}, a_{n,t}) - \lambda\Omega(\pi_k(\cdot|s_{n,t})))$. Thus, the expectation of $\widehat{J}_\nu(\pi_k, \lambda)$ is:

$$\mathbb{E}\widehat{J}_\nu(\pi_k, \lambda) = \mathbb{E}_{\pi,\mathbb{P}}\left[\sum_{t=0}^{K-1} \gamma^t (R(s_t, a_t) - \lambda\Omega(\pi(\cdot|s_t)))\right] \tag{113}$$

Theorem B.11. *At time-step t, with probability* $1 - 2\delta$ *and* $N = \frac{2(1+\lambda C_\Phi)^2}{\varepsilon^2(1-\gamma)^2} \log \frac{T_t}{\delta}$ *for each policy* π_i*,*

$$J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_{\widehat{i}}, \lambda_t) \leq \min_{i=1,\ldots,T_t} J_\nu(\pi^*_{\lambda_t}, \lambda_t) - J_\nu(\pi_i, \lambda_t) + 2\varepsilon \tag{114}$$

where $\widehat{i} = \mathrm{argmax}_{i=1,\ldots,T_t} \widehat{J}_\nu(\pi_i, \lambda_t)$.

Proof. Note that $|\widehat{J}| \leq \frac{1+\lambda C_\Phi}{1-\gamma}$, by Lemma B.15 and setting $K = \log \frac{2(1+\lambda C_\Phi)}{\varepsilon(1-\gamma)} / \log \frac{1}{\gamma}$, we have:

$$\begin{aligned}
P\left(\max_{i=1,\ldots,T_t} \widehat{J}_\nu(\pi_i, \lambda_t) < \max_{i=1,\ldots,T_t} J_\nu(\pi_i, \lambda_t) - \varepsilon\right) &\leq \sum_{i=1}^{T_t} P\left(\widehat{J}_\nu(\pi_i, \lambda_t) < J_\nu(\pi_i, \lambda_t) - \varepsilon\right) \\
&\leq \sum_{i=1}^{T_t} P\left(\widehat{J}_\nu(\pi_i, \lambda_t) < \mathbb{E}\widehat{J}_\nu(\pi_i, \lambda_t) - \varepsilon/2\right) \\
&\leq T_t \exp\left(-\frac{\varepsilon^2 N(1-\gamma)^2}{2(1+\lambda C_\Phi)^2}\right)
\end{aligned}$$

where the second inequality holds by $J_\nu(\pi, \lambda) - \mathbb{E}\widehat{J}_\nu(\pi, \lambda) \leq \frac{\gamma^K(1+\lambda C_\Phi)}{1-\gamma}$. Thus, by Lemma B.16, we have:

$$P\left(J_\nu(\pi_{\widehat{i}}, \lambda_t) < \max_{i=1,\ldots,T_t} J_\nu(\pi_i, \lambda_t) - 2\varepsilon\right) \leq 2\delta \tag{115}$$

□

B.4. Policy Gradient Based Alg: Gradient Ascent with Softmax Parameterization

In Remark 3.2, it's mentioned that $\Omega(\pi) = \sum_a \pi(a|s) \log(\pi(a|s))$ breaks the L-smooth assumption in tabular MDP setting. However, if policy is parameterized by softmax, the L-smooth property holds. In this scenario, Mei et al. (2020) showed that the iteration complexity of policy gradient (Algorithm 5) is $O(\log \frac{1}{\varepsilon})$ for a fixed λ (Theorem B.12). However, in this section, we show that the iteration complexity is inefficient (exponentially dependent on $1/\varepsilon$) for general MDPs. In such situation, can AdaptReduce help make the algorithm efficient (polynominally dependent on $1/\varepsilon$)? We have a negative answer.

Algorithm 5 Policy Gradient Ascent

Input: an initial parameter θ_0, a regularization parameter λ and T the number of iteration.
for iteration $t = 0$ **to** $T - 1$ **do**
  $\theta_{t+1} = \theta_t + \eta \nabla_\theta J(\pi_{\theta_t}, \lambda)$
end for
Return: π_T

Definition B.1 (Softmax Parameterization)**.** *Given a vector* $\theta \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$*, the softmax parameterization of policy* π *is defined as:*

$$\pi_\theta(a|s) = \frac{e^{\theta(s,a)}}{\sum_{a' \in \mathcal{A}} e^{\theta(s,a')}}$$

Theorem B.12 (Theorem 6 in (Mei et al., 2020))**.** *Suppose the initial distribution* $\mu(s) > 0$ *for all* $s \in \mathcal{S}$ *and policy* π *is parameterized by Definition B.1, policy gradient with learning rate* $\eta = \frac{(1-\gamma)^3}{8+\lambda(4+8\log|\mathcal{A}|)}$ *outputs a policy* π_t *satisfying:*

$$J(\pi^*_\lambda, \lambda) - J(\pi_t, \lambda) \leq \left\|\frac{1}{\mu}\right\|_\infty \frac{J(\pi^*_\lambda, \lambda) - J(\pi_0, \lambda)}{1-\gamma} e^{-C_\lambda t} \tag{116}$$

where $C_\lambda = \frac{c_\lambda(1-\gamma)^4 \min_s \mu(s)}{(8/\lambda+4+8\log|\mathcal{A}|)|\mathcal{S}|} \left\| \frac{d_{\pi^*_\lambda,\mu}}{\mu} \right\|_\infty^{-1}$.

Remark B.2. *If we consider C_λ as a constant, the iteration could be improved upon $\widetilde{O}(\log \frac{1}{\varepsilon})$. However, it's not the whole story. For a fixed λ, in order to obtain an ε-optimal policy by Theorem B.12 in polynominal time, we need to investigate how large C_λ^{-1} could be, which is equivalent to investigate how large $1/c_\lambda$ could be. Note that $c_\lambda = \inf_{t\geq 1} \min_{s,a} \pi_t(a|s) \approx \min_{s,a} \pi_\infty(a|s) \approx \min_{s,a} \pi^*_\lambda(a|s)$ by (Mei et al., 2020). However, we have the close form of π^*_λ:*

$$\pi^*_\lambda(\cdot|s) = \frac{e^{Q^*_\lambda(s,\cdot)/\lambda}}{\sum_{a\in\mathcal{A}} e^{Q^*_\lambda(s,a)/\lambda}} \tag{117}$$

Thus, for each state s and assuming $Q^*_\lambda(s,a_1) \leq ... \leq Q^*_\lambda(s,a_{|\mathcal{A}|})$, we can lower and upper bound $\min_a \pi^*_\lambda(a|s)$ as follows:

$$\frac{1}{|\mathcal{A}|e^{\left(Q^*_\lambda(s,a_{|\mathcal{A}|})-Q^*_\lambda(s,a_1)\right)/\lambda}} \leq \min_a \pi^*_\lambda(a|s) \leq \frac{1}{e^{\left(Q^*_\lambda(s,a_{|\mathcal{A}|})-Q^*_\lambda(s,a_1)\right)/\lambda}} \tag{118}$$

Note that $\Delta^*_\lambda :\overset{\Delta}{=} Q^*_\lambda(s,a_{|\mathcal{A}|}) - Q^*_\lambda(s,a_1) \leq \frac{1+\lambda\log|\mathcal{A}|}{1-\gamma}$. When $\Delta^*_\lambda \approx \frac{1+\lambda\log|\mathcal{A}|}{1-\gamma}$, we obtain $\min_a \pi^*_\lambda(a|s)$ is of order $\Theta(e^{-\frac{1}{\lambda(1-\gamma)}}|\mathcal{A}|^{-\frac{1}{1-\gamma}})$, which is even exponentially dependent on $\frac{1}{1-\gamma}$. In order to obtain an ε-optimal policy by Theorem B.12 for a fixed λ in polynominal time, we have to assume $\Delta^*_\lambda \approx \lambda\log\left(\text{poly}(\frac{1}{\lambda},\frac{1}{1-\gamma})\right)$. How to reduce the exponential dependence on other term still remains open. We consider it as future work.

By setting $\lambda = O(\varepsilon(1-\gamma)/C_\Phi)$, the iteration complexity for Algorithm 5 is $\widetilde{O}\left(e^{\frac{1}{\varepsilon(1-\gamma)}}\right)$, as $\Delta^*_\lambda \to \Delta^* > 0$ while $\lambda \to 0$. Thus, the upper bound is not efficient. Though, we can also obtain that policy gradient method satisfies Prop-I($\hat{\varepsilon}$) property in Corollary B.2. Negatively, we can't upper bound total time $\sum_{t=0}^{T-1}$ Time(λ_t) polynominally dependent on $\frac{1}{\varepsilon}$ and $\frac{1}{1-\gamma}$ as the sub-solver is inefficient when λ is sufficiently small. In fact, the total time is of order:

$$\sum_{t=0}^{T-1} \frac{1}{\lambda_t c_{\lambda_t}} \approx \sum_{t=0}^{T-1} \frac{1}{\lambda_t} e^{\frac{1}{\lambda_t}} \tag{119}$$

Note that $\lambda_T = O((1-\gamma)\varepsilon/C_\Phi)$, thus no matter how to adjust the decaying rate of λ_t we always obtain $e^{1/\varepsilon(1-\gamma)}$ in total time. In conclusion, we argue that efficient sub-solver algorithm is a sufficient condition for AdaptReduce leveraging an efficient algorithm. Besides, it's not saying that vanilla policy gradient method with softmax parameterization is inefficient at all for regularized MDP. We leave it as future work to derive an efficient upper bound in this situation.

Corollary B.2 (Policy Gradient satisfies Prop-I(0))**.** *Under the same setting in Theorem B.12, Policy Gradient satisfies the Prop-I($\hat{\varepsilon}$) property with $\hat{\varepsilon} = 0$ and:*

$$\textit{Time}(\lambda) = \frac{1}{C_\lambda}\log\left(\frac{4\left\|\frac{1}{\mu}\right\|_\infty}{1-\gamma}\right) \tag{120}$$

B.5. Policy Gradient Based Alg: Other Methods

Shani et al. (2019) proposed another type of policy gradient to solve regularized MDP as follows:

$$\pi_{t+1} = \underset{\pi\in\Delta(\mathcal{A})^{\mathcal{S}}}{\text{argmin}} -\langle\nabla J(\pi_t,\lambda), \pi-\pi_t\rangle + \frac{1}{\eta}\mathbb{E}_{s\sim d^{\pi_t}} D_\Omega(\pi(\cdot|s)||\pi_t(\cdot|s)) \tag{121}$$

When $\Omega(\pi(\cdot|s)) = \sum_a \pi(a|s)\log(\pi(a|s))$, we can obtain a close form of update from equation (121):

$$\pi_{t+1}(\cdot|s) \propto \pi_t^{1-\lambda\eta}(\cdot|s) \odot e^{\eta Q^{\pi_t}_\lambda(s,\cdot)} \tag{122}$$

However, by definition of $\nabla J(\pi,\lambda)$, we can also re-write the update rule (121) into:

$$\pi_{t+1} = \underset{\pi\in\Delta(\mathcal{A})^{\mathcal{S}}}{\text{argmin}} \mathbb{E}_{s\sim d_{\pi_t}}\left(-\langle Q^{\pi_t}_\lambda(s,\cdot) - \lambda\nabla\Omega(\pi_t(\cdot|s)), \pi(\cdot|s)-\pi_t(\cdot|s)\rangle + \frac{1}{\eta}D_\Omega(\pi(\cdot|s)||\pi_t(\cdot|s))\right) \tag{123}$$

As $D_\Omega(\pi||\pi_t) = \Omega(\pi) - \Omega(\pi_t) - \langle \nabla\Omega(\pi_t), \pi - \pi_t\rangle$ and we assume $\lambda\eta = 1$, the update rule can be simplified to as:

$$\pi_{t+1} = \operatorname*{argmin}_{\pi\in\Delta(\mathcal{A})^{\mathcal{S}}} \mathbb{E}_{d_{\pi_t}} - \langle Q_\lambda^{\pi_t}(s,\cdot) - \lambda\Omega(\pi(\cdot|s)), \pi(\cdot|s)\rangle \tag{124}$$

which is exactly RMPI with exact evaluation version ($m = \infty$) by solving above problem for each state $s \in \mathcal{S}$. Cen et al. (2020) also showed near the same update rule as equation (124) while π is parameterized by softmax and optimized by natural gradient. Thus, the analysis of convergence rate with λ reduction can be covered by Section B.2.1.

Remark B.3. *In fact, we can also set $\eta = \Theta(1)$ and controlling λ in (Shani et al., 2019; Cen et al., 2020). Thus the iteration complexity for fixed small $\lambda = O(\varepsilon(1-\gamma)/C_\Phi)$ is $\widetilde{O}\left(\frac{1}{\varepsilon(1-\gamma)}\right)$. Besides, combining with Algorithm 1, the final iteration complexity is also $\widetilde{O}\left(\frac{1}{\varepsilon(1-\gamma)}\right)$ in terms of ε. Either methods is slower than the case $\lambda\eta = 1$.*

C. Proof of Section 4

C.1. Proof of Theorem 4.1

Proof. The first part is rather obvious. Since the reward r and the regularization function Ω are both uniformly bounded as defined, then $|J(\pi,\lambda)| \le |V^\pi(\mu)| + \lambda|\Phi^\pi(\mu)| \le \frac{R+\lambda C_\Phi}{1-\gamma} < \infty$ for any fixed λ. Therefore, we have that $\max_\pi J(\pi,\lambda)$ is finite and thus well-defined for any fixed λ.

For the second part (here we don't assume Ω is bounded), it is an important observation that $J(\pi,\lambda) = V^\pi(\mu) - \lambda\Phi^\pi(\mu)$ increases in λ since Ω is non-positive. Therefore,

$$\min_{\lambda\ge 0} J(\pi,\lambda) = J(\lambda,0) = J(\pi)$$

whose absolute value is bounded. By taking maximum in π on both side of the last equality, we finish the first equality in (4).

Let's focus on the second equality then. For simplicity, denote by $LHS = \max_\pi \min_{\lambda\ge 0} J(\pi,\lambda)$ and $RHS = \min_{\lambda\ge 0}\max_\pi J(\pi,\lambda)$. From previous discussion, we already have $LHS = J(\pi^*)$. For one thing, note that we always have

$$\max_\pi \min_{\lambda\ge 0} J(\pi,\lambda) \le \min_{\lambda\ge 0}\max_\pi J(\pi,\lambda),$$

then $LHS \le RHS$. For the other thing,

$$\max_\pi J(\pi,\lambda) = J(\pi^*_\lambda,\lambda) \le V^{\pi^*}(\mu) + \frac{\lambda C_\phi}{1-\gamma}$$

Minimizing λ both sides, we gain $RHS \le J(\pi^*) = LHS$ Simply putting the two results, we must have equality throughout. □