Adaptive Model Design for Markov Decision Process

Siyu Chen [*1] **Donglin Yang** [*1] **Jiayang Li** [2] **Senmiao Wang** [2] **Zhuoran Yang** [3] **Zhaoran Wang** [2]

Abstract

In a Markov decision process (MDP), an agent interacts with the environment via perceptions and actions. During this process, the agent aims to maximize its *own* gain. Hence, appropriate regulations are often required, if we hope to take the external costs/benefits of its actions into consideration. In this paper, we study how to regulate such an agent by redesigning model parameters that can affect the rewards and/or the transition kernels. We formulate this problem as a bilevel program, in which the lower-level MDP is regulated by the upper-level model designer. To solve the resulting problem, we develop a scheme that allows the designer to iteratively predict the agent's reaction by solving the MDP and then adaptively update model parameters based on the predicted reaction. The algorithm is first theoretically analyzed and then empirically tested on several MDP models arising in economics and robotics.

1. Introduction

Markov decision process (MDP) is a powerful tool for modeling various dynamic planning problems arising in economic, social, and engineering systems. It has found applications in such diverse fields as financial investment (Derman et al., 1975), repair and maintenance (Golabi et al., 1982; Ouyang, 2007), resource management (Little, 1955; Russell, 1972), inventory and production (Onstad & Rabbinge, 1985; Symonds, 1971), as well as robotic control (Koenig et al., 1998). In an MDP, an agent interacts with the environment via perceptions and actions, seeking to find a policy that maximizes its total reward. However, during this process, it usually does not bear the external costs (Maskin, 1994) of its actions. Although these externalities are not received by the agent, they may be detrimental to other individuals in the system or the system's overall performance.

[*]Equal contribution [1]Tsinghua University, Beijing, China [2]Northwestern University, Evanston, IL, USA [3]Yale University, New Haven, CT, USA. Correspondence to: Zhaoran Wang <zhaoranwang@gmail.com>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

Take a small economy with one manufacturer as an example. At each decision epoch, the manufacturer determines the number of raw materials to purchase as well as the number of products to produce based on system states including the inventory levels, the prices of the materials, and its cash balance (the amount of money on hand), etc. In such an economy, the social welfare is affected by not only the profit of the manufacturer but also the pollution caused by the production. Unfortunately, the primary goal of the manufacturer is only to maximize its own profit. The environmental impact, however, is not under its consideration. Therefore, when the manufacturer's profit is maximized, the environment may have already been damaged by the pollution.

To mitigate the potential environmental impact, appropriate regulation is often necessary to guide the self-interests of the manufacturer towards a systemically optimal state. For example, the government can impose pollution taxes on high-emission products, which would not only reshape the manufacturer's *reward function* and affect the *transition of system states* (e.g., the cash balance). To maximize the profit under the existence of the taxes, the manufacturer then needs to change its production plan. With such power to influence the manufacturer's decision, the government then can adaptively design the pollution taxes such that the resulting production plan maximizes the social welfare.

The above discussion has led us to the following question that motivates our study: *how can we adaptively design the reward function/transition kernel in an MDP to induce a desirable outcome that fulfills the designer's objective?*

The resulting problem may be cast the resulting problem as a Stackelberg game (Stackelberg, 1952) in which the leader designs parameters in the MDP while the follower solves the parameterized MDP accordingly. To solve this Stackelberg game, we adaptively improve the designer's decision based on the MDP agent's best response. To this end, we first predict that response by solving the parameterized MDP and then search for a direction to improve the designer's decision by examining the sensitivity of that response with respect to the parameters, which in turn requires differentiating through the parameterized MDP.

Over the past decades, numerous algorithms have been proposed for *solving* MDPs. Particularly, the rapid development of reinforcement learning (RL) algorithms (Sutton & Barto, 2018) has become one of the keys to the recent success of modern machine learning enterprises. However, how to efficiently *differentiate* through an MDP, i.e., calculate the gradient of the optimal policy with respect to system parameters, is still an open question. This task is intrinsically difficult, because (i) the optimal policy of an MDP is not always unique; (ii) even if the optimal policy is unique, it may still be discontinuous or too sensitive with respect to parameters in the environment (Ahmed et al., 2019). To overcome these difficulties, we propose to add an entropy regularizer to the MDP agent's policy. It results in a regularized MDP model (Geist et al., 2019), which assumes bounded rationality on the agent's behavior. As we shall see, it promises to kill two birds—passing the difficulty posed by multiple lower-level solutions and smoothing the functional geometry at the upper level—with one stone.

However, even though the optimal policy of a regularized MDP is differentiable, calculating the gradient of that optimal policy is still a computational burden, as it requires repeatedly solving the regularized MDP exactly first. To resolve this difficulty, a single-looped algorithm is developed in our work, which updates the MDP agent's policy and the parameters in the MDP simultaneously. We prove that it converges to the optimal solution to the MDP design problem and establish sufficient convergence conditions.

1.1. Related work

Extensive research effort has been devoted to study how to design a non-cooperative game (Requate, 1993; Ehtamo et al., 2002; Lawphongpanich & Hearn, 2004; Li et al., 2020; Liu et al., 2021a). In the optimization literature, the resulting problem is often formulated as a bi-level program. We refer the reader's to Colson et al. (2007) for a comprehensive overview on conventional bi-level programming algorithms. In the machine learning literature, bi-level programming has also found applications in many other fields, e.g., hyperparameter tuning (Franceschi et al., 2018), model-agnostic meta-learning (Finn et al., 2017), actor-critic method (Hong et al., 2020) and ML-based optimal auctions (Dütting et al., 2019). These bi-level programs are typically solved by the gradient method that proposes differentiating through the lower-level optimization problem (Liu et al., 2021b; Rajeswaran et al., 2019; Maclaurin et al., 2015), which is also a base for our work.

More recently, a few recent works have studied how to design an MDP. For example, Li et al. (2019) show that imposing incentives on the reward function can be utilized by social planners to achieve auxiliary objectives in an MDP congestion game. Another example is an "AI economist" introduced to regulate the economical systems with misaligned or unethical incentives at the agent level (Hill et al., 2021). Metelli et al. (2018) focuses on simultaneous shaping of the transition model and the agent's policy to improve the total reward. However, the upper-level designer and lower-level agent in Metelli et al. (2018) share the same objective, which means they are fully cooperative without an externality effect.

1.2. Notation

We define $\mathscr{P}(\mathcal{X})$ as the set of probability measures over the measurable space $\mathcal{X}$. For a differential function f, we denote by $\dot{f}$ the derivative of f. For a finite set $\mathcal{X}$, we denote by $|\mathcal{X}|$ the cardinality of set $\mathcal{X}$. For function $f : \mathcal{X} \to \mathbb{R}$ and $g : \mathcal{X} \to \mathbb{R}$, we denote by $\langle f, g\rangle_{\mathcal{X}}$ the inner product of f and g on $\mathcal{X}$. We denote by $\|M(x,y)\|_{x\sim p_x, y\sim p_y}$ a hybrid norm of order p_x on $\mathcal{X}$ and of order p_y on $\mathcal{Y}$, which is defined by $\|M(x,y)\|_{x\sim p_x, y\sim p_y} = \|\|M(x,y)\|_{x\sim p_x}\|_{y\sim p_y}$.

2. Background

2.1. Markov Decision Process

In reinforcement learning, a sequential decision making problem is usually formulated as a Markov decision process. An MDP can be characterized by a tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where S denotes the state space and A denotes the finite action space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathscr{P}(\mathcal{S})$ is the transition kernel, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $\gamma \in [0, 1)$ is the discount factor. A policy $\pi(\cdot|s) : \mathcal{A} \to \mathscr{P}(\mathcal{A})$ is a distribution over action space $\mathcal{A}$ at any state $s \in \mathcal{S}$. Given a policy π, we can define the corresponding value function and state-action value function as

$$V^\pi(s) = \mathbb{E}_\pi\left[\sum_{m\geq 0} \gamma^m \cdot r(s_m, a_m)\middle| s_0 = s\right], \quad (1)$$

$$Q^\pi(s,a) = r(s,a) + \gamma\big\langle P(\cdot|s,a), V^\pi(\cdot)\big\rangle_{\mathcal{S}}, \quad (2)$$

where $(s_{m+1}, a_{m+1}) \sim P^\pi(\cdot, \cdot \,|\, s_m, a_m)$. Moreover, Q^π and V^π satisfies the following equilibrium

$$V^\pi(s) = \big\langle \pi(\cdot|s), Q^\pi(\cdot, s)\big\rangle_{\mathcal{A}}. \quad (3)$$

Accordingly, the advantage function $A^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as

$$A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s). \quad (4)$$

The visitation measurement in an MDP is defined as

$$\begin{aligned}&\tilde{\mathcal{E}}^\pi_{\mathcal{D}_0}(\mathrm{d}s, \mathrm{d}a)\\ &\quad=(1-\gamma)\sum_{m\geq 0}\gamma^m \cdot \mathbb{P}((s_m, a_m) \in (\mathrm{d}s, \mathrm{d}a)|\pi, \mathcal{D}_0),\\ &\mathcal{E}^\pi_{\mathcal{D}_0}(\mathrm{d}s) = (1-\gamma)\sum_{m\geq 0}\gamma^m \cdot \mathbb{P}(s_m \in \mathrm{d}s|\pi, \mathcal{D}_0),\end{aligned} \quad (5)$$

where $\mathcal{D}_0$ is the initial distribution of s_0 over the state space. It's well known that the optimal state-action value function Q^{π^*} satisfies the Bellman optimality equation, which could refer to

$$Q^{\pi^*}(s,a) = r(s,a) + \gamma \cdot \mathbb{E}_{P(\cdot|s,a)}\left[V^{\pi^*}(\cdot)\right], \quad (6)$$

where the optimal state value function and the optimal policy are defined as

$$V^{\pi^*}(s) = \max_{\pi} \langle \pi(\cdot|s), Q^{\pi^*}(s,\cdot) \rangle_{\mathcal{A}}, \tag{7}$$

$$\pi^* = \arg\max_{\pi} \langle \pi(\cdot|s), Q^{\pi^*}(s,\cdot) \rangle_{\mathcal{A}} \in \Pi^*, \tag{8}$$

where the set of optimal policies is defined as

$$\Pi^*(\mathcal{S},\mathcal{A},\mathcal{P},\mathcal{R},\gamma) = \left\{ \pi | V^{\pi}(s) \geq V^{\pi'}(s), \forall \pi', \forall s \right\}. \tag{9}$$

2.2. Regularized Markov Decision Process

Let $\Omega : \mathbb{R} \to \mathbb{R}$ be a strictly convex and doubly differentiable function and $\epsilon > 0$ be the regularization parameter. The value function Q^*_ϵ of optimal policy π_ϵ with policy entropy regularization satisfies the following equilibrium

$$Q^*_\epsilon(s,a) = r(s,a) + \gamma \cdot \mathbb{E}_{P(\cdot \mid s,a)} \left[V^*_\epsilon(\cdot) \right], \tag{10}$$

where

$$V^*_\epsilon(s) = \max_{\pi} \langle \pi(\cdot|s), Q^*_\epsilon(s,\cdot) \rangle_{\mathcal{A}} - \epsilon^{-1} \sum_a \Omega(\pi(a|s)). \tag{11}$$

The difference between definition 11 with definition 7 is the entropy regularization item Ω/ϵ. Note that V_ϵ is the convex conjugate of $\epsilon^{-1}\sum \Omega$. The maximizing argument π^*_ϵ is unique because the regularization entropy Ω is strictly convex. The optimal policy π^*_ϵ can be derived through KKT condition of definition 11 as follows

$$\begin{aligned} &\pi^*_\epsilon(a|s) = \varphi(\epsilon \cdot (Q^*_\epsilon(s,a) + v)) \\ &\textbf{s.t.} \quad \sum_a \pi^*_\epsilon(a|s) = 1, \end{aligned} \tag{12}$$

where v is the dual variable for the equilibrium $\sum_a \pi^*_\epsilon(a|s) = 1$ and $\varphi(x) = \max\{\dot{\Omega}^{-1}(x), 0\}$. A classical example is the KL divergence $\sum_a \Omega(\pi) = \langle \pi, \log(\pi) \rangle_{\mathcal{A}}$. Its convex conjugate is the smoothing maximum $V^*_\epsilon(s) = \ln \sum_a \exp Q^*_\epsilon(s,a)$, and the optimal policy $\pi^*_\epsilon(a|s) = \exp Q^*_\epsilon(s,a) / \sum_a \exp Q^*_\epsilon(s,a)$.

3. Problem Formulation

In our model, the system environment is formulated as a Markov Decision Process (MDP) in which the MDP agent pursues its interest. The designer seeks to stimulate the desired policy from the MDP agent and achieve the system's overall well-being by tuning some design parameter $\theta \in \mathcal{X}$ that sculpts both the reward and the transition of the MDP. Such a process is modeled as the original MDP design (OMD) problem,

$$\begin{aligned} \text{OMD}: \quad & \max_{\theta \in \mathcal{X}} F(\theta, \pi^*), \\ \text{s.t.} \quad & \pi^* \in \Pi^*(\mathcal{S},\mathcal{A},\gamma,P(\theta),r(\theta)), \end{aligned} \tag{13}$$

where the MDP is given by $(\mathcal{S},\mathcal{A},\gamma,P(\theta),r(\theta))$, Π^* is the set of agent's optimal policies in response to the MDP dynamics given by θ and F corresponds to the objective function the designer aims to maximize. Note that the negative externality is inherent in the inconsistency between F and the agent's reward $r(\theta)$ in such a bilevel problem. Here, the optimal responses of the MDP agent form a set Π^* for the sake that the optimal policy of the MDP agent might not be unique. Specifically, when Π^* has more than one element, the OMD is non-singleton (Liu et al., 2021b) and thereby ill-defined since different $\pi \in \Pi^*$ yields different F in the presence of externality. Even though the optimal policy is unique, the optimal policy can be discontinuous concerning θ, rendering it hard to differentiate through the optimal policy. For example, Ahmed et al. (2019) studied the landscape of objective functions during the policy optimization and suggested that even without stochastically high variance, the objective function can still fluctuate too significantly for policy optimization.

To address the non-singleton and discontinuity problems discussed before, we assume bounded rationality in the MDP agent and introduce policy regularization in the agent's policy. Specifically, we formulate the problem of regularized Markov design (RMD) as follows,

$$\begin{aligned} \text{RMD}: \quad & \max_{\theta \in \mathcal{X}} F(\theta, \pi), \\ \text{s.t.} \quad & \pi = \pi^*_\epsilon(\mathcal{S},\mathcal{A},\gamma,P(\theta),r(\theta)), \end{aligned} \tag{14}$$

where π^*_ϵ is the optimal policy for the ϵ-regularized MDP given by (12). For example, π^*_ϵ corresponds to a softmax policy if the KL divergence is used as the entropy regularization. We thus see that the MDP agent with bounded rationality follows a unique regularized policy π^*_ϵ, making the RMD problem well-defined. We remark that π^*_ϵ also enjoys good properties that gradient methods need, e.g., π^*_ϵ is continuous concerning Q^*_ϵ following (12). Naturally, we will ask how the RMD is related to the OMD. To answer the question, we have the following theorem showing that the optimal objective function of the RMD is upper/lower bounded by the optimistic/pessimistic objective function of the OMD.

Theorem 3.1 (Sub-optimality of the RMD)**.** *Assume that the designer's objective function $F(\theta,\pi)$ is $L_{F,\pi,0}$-Lipschitz continuous with respect to π under the norm $\|\cdot\|_{a\sim 1, s\sim\infty}$. For any positive $\Delta_\pi \in \mathbb{R}^+$ and $\Delta_r \in \mathbb{R}^+$ such that $\Delta_r \geq \epsilon^{-1}(\gamma U_\Omega + (1+\gamma)\cdot \log(2|\mathcal{A}|/\Delta_\pi))$ where $U_\Omega = \max_\pi \sum_a \Omega(\pi(a))$, it holds that*

$$\begin{aligned} &\max_{\theta} F(\theta, \pi^*_\epsilon(r_\theta)) \\ &\quad \leq \max_{\theta} \max_{\substack{\pi \in \Pi^*(P(\theta),\hat{r}(\theta)), \\ \hat{r}(\cdot) \in \hat{R}(\Delta_r)}} F(\theta,\pi) + \Delta_\pi L_{F,\pi,0}, \end{aligned} \tag{15}$$

and that

$$\max_{\theta} F(\theta, \pi_\epsilon^*(r_\theta)) \geq \max_{\theta} \min_{\substack{\pi\in\Pi^*(P(\theta),\hat{r}(\theta)),\\ \hat{r}(\cdot)\in\hat{\mathcal{R}}(\Delta_r)}} F(\theta, \pi) - \Delta_\pi L_{F,\pi,0}. \quad (16)$$

Here, $\hat{\mathcal{R}}$ is the set of reward functions such that $\hat{\mathcal{R}}(\Delta_r) = \left\{\hat{r} : \mathcal{S}\times\mathcal{A}\times\Theta \to \mathbb{R} \mid \|\hat{r} - r\|_{(\theta,s,a)\sim\infty} < \Delta_r\right\}$ where r is the exact reward function and $\Pi^*(P(\theta), \hat{r}(\theta))$ is a simplified denotion for $\Pi^*(\mathcal{S}, \mathcal{A}, \gamma, P(\theta), \hat{r}(\theta))$.

Proof. See §A.1 for more details. □

We remark that the first terms on the right-hand side of (15) and (16) correspond to the optimistic and pessimistic objective function of the OMD, respectively. Here, the optimism/pessimism is taken with respect to $\pi \in \Pi^*(P, \hat{r})$ where $\hat{r} \in \hat{\mathcal{R}}(\Delta_r)$. Therefore, Theorem 3.1 shows that the RMD can be solved to a place amid the pessimistic and the optimistic versions of the OMD up to an error term $\Delta_\pi L_{F,\pi,0}$. Note that the optimistic and the pessimistic solutions are intrinsic to the OMD problem. Particularly, when the optimistic and pessimistic objective functions of the OMD are consistent as $\Delta_r \to 0$, Theorem 3.1 implies the convergence of the optimal objective function of the RMD as Δ_r and Δ_π both diminish by letting $\epsilon \to \infty$. Such a result coincides with the intuition that the RMD approaches the OMD when less regularization is involved. In the remaining part, we will focus on solving the RMD problem as an alternative to the ill-posed OMD problem.

Benefits of regularization. By introducing regularization in the MDP, the RMD problem defined in (14) has a smoother landscape that facilitates adaptive design with gradient methods (Ahmed et al., 2019). Besides, regularization is introduced in many reinforcement learning algorithms, e.g., Trust Region Policy Optimization (TRPO), with the motivation to improve exploration and robustness (Schulman et al., 2015). Moreover, regularization can improve the stability of the proposed algorithm, as is demonstrated in (Chaudhari et al., 2019) that penalty induces objectives with higher β-smoothness and improves stability. Theorem 5.4 in the following section also shows that convergence is guaranteed with enough regularization. Therefore, we remark that by transforming the OMD into RMD, the problem becomes well-defined and easy to solve at the price of introducing some sub-optimality characterized by Theorem 3.1. In §4.1, it is further shown that using Kullback-Leibler (KL) Divergence for regularization enables the gradient of the optimal policy to be updated via a Bellman operator. Making use of such a fact, we propose an easy-to-implement algorithm.

4. Algorithm

In this section, We first propose a general framework for solving the RMD and then study a special case where the design objective function is the total reward on the MDP.

4.1. General Framework for Solving RMD

For simplicity, we define an operator as follows

$$\mathcal{T}_{r,\gamma}^{\theta}(V)(s,a) = r(s,a) + \gamma\mathbb{E}_{P(\cdot\,|\,s,a;\theta)}\left[V(\cdot)\right]. \quad (17)$$

In the regularized MDP, the optimal policy $\pi_\epsilon{}^*$ is uniquely determined by the Q function by (12). Hence, we have

$$\nabla_\theta \pi_\epsilon{}^*(a\,|\,s) = \left\langle \frac{\partial \pi_\epsilon{}^*(a\,|\,s)}{\partial Q_\epsilon^*(a'\,|\,s)}, \nabla_\theta Q_\epsilon^*(a'\,|\,s) \right\rangle_{a'\in\mathcal{A}}. \quad (18)$$

Following (18), we show that the gradient of the regularized policy π_ϵ^* with respect to the design parameter θ is given by

$$\nabla_\theta \pi_\epsilon^*(a|s) = \epsilon\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a)+v)) \sum_{a'} \Big(\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a')+v)) (\nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta Q_\epsilon^*(s,a'))\Big) \Big(\sum_{a''} \dot{\varphi}(\epsilon(Q_\epsilon^*(s,a'')+v))\Big)^{-1}. \quad (19)$$

See §A.2 for more details. Although (19) shows that it is possible to take the gradient of the optimal policy in the regularized MDP, it is still too complicated as we have to calculate the dual variables v. To further simplify (19), we propose using KL divergence as the entropy regularization in the following discussions, i.e. $\Omega(x) = x\ln x$. It is observed that $\varphi(x) = \dot{\Omega}^{-1}(x) = \exp(x-1)$ and we have the expression simplified to

$$\nabla_\theta \pi_\epsilon^*(a|s) = \epsilon \cdot \pi_\epsilon^*(s,a) \cdot \nabla_\theta A_\epsilon(s,a), \quad (20)$$

where we have

$$\nabla_\theta V_\epsilon^*(s) = \mathbb{E}_{\pi_\epsilon{}^*(\cdot\,|\,s)}\left[\nabla_\theta Q_\epsilon^*(s,\cdot)\right], \quad (21)$$

$$\nabla_\theta Q_\epsilon^* = \mathcal{T}_{\nabla_\theta r,\gamma}^{\theta}\left(\nabla_\theta V_\epsilon^* + V_\epsilon^*\nabla_\theta \ln P\right), \quad (22)$$

$$\nabla_\theta A_\epsilon^*(s,a) = \nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta V_\epsilon^*(s). \quad (23)$$

See §A.2 for more details. Here we remark that if r, $\nabla_\theta r$, and $\nabla_\theta \ln P$ are globally bounded, it follows that $\nabla_\theta Q_\epsilon^*$, $\nabla_\theta V_\epsilon^*$, $\nabla_\theta A_\epsilon^*$, and $\nabla_\theta \pi_\epsilon^*$ are also bounded, which is one of the benefits stemmed from the entropy regularization method. Now we are ready to present the gradient of the designer's objective function as follows

$$\nabla_\theta F = \frac{\partial F}{\partial \theta} + \epsilon\mathbb{E}_{\rho^{\pi_\epsilon^*}}\left[\rho^{-1}\cdot\frac{\partial F}{\partial \pi}\cdot\nabla_\theta A_\epsilon\right], \quad (24)$$

where $\rho : \mathcal{S} \to \Delta(\mathcal{S})$ is a reference distribution for sampling across the state space. Now we are ready to present the following general framework for solving the RMD (14) with $\Omega(x) = x \ln x$.

Algorithm 1 General framework for solving the RMD (14) with $\Omega(x) = x \ln x$

Input: outer iterations T, inner iterations K, learning rate η, the gradient of pre-learned transition model $\nabla_\theta \ln P$ and the gradient of the reward function $\nabla_\theta r$ with respect to θ.
Initialize parameter θ_0, value function Q_ϵ^0 and its corresponding gradient $\nabla_{\theta_0} Q_\epsilon^0$
for $t = 0$ **to** $T-1$ **do**
  for $k = 0$ **to** $K-1$ **do**
    $\pi_\epsilon^k(\cdot|s) \propto \exp\left(\epsilon Q_\epsilon^k(s,\cdot)\right)$
    $V_\epsilon^k(s) = \epsilon^{-1} \ln\left(\sum_a \exp\left(\epsilon Q_\epsilon^k(s,a)\right)\right)$
    $\nabla_{\theta_t} {V_\epsilon}^k(s) = \mathbb{E}_{\pi_\epsilon^K}\left[\nabla_{\theta_t} Q_\epsilon^K(s,a)\right]$
    $Q_\epsilon^{k+1} = \mathcal{T}_{r,\gamma}^{\theta}({V_\epsilon}^k)$
    $\nabla_{\theta_t} Q_\epsilon^{k+1} = \mathcal{T}_{\nabla_{\theta_t} r,\gamma}^{\theta}(\nabla_{\theta_t} V_\epsilon^k + V_\epsilon^k \nabla_{\theta_t} \ln P)$
  end for
  $\nabla_{\theta_t} A_\epsilon^K(s,a) = \nabla_{\theta_t} Q_\epsilon^K(s,a) - \nabla_{\theta_t} {V_\epsilon}^K(s)$
  $\nabla_{\theta_t} F = \frac{\partial F}{\partial \theta_t} + \epsilon \mathbb{E}_{\rho^{\pi_\epsilon^K}}\left[\rho^{-1} \cdot \frac{\partial F}{\partial \pi_\epsilon^K} \cdot \nabla_{\theta_t} A_\epsilon^K\right]$
  $\theta_{t+1} = \theta_t + \eta \nabla_{\theta_t} F$
  Reintialize $Q_\epsilon^0 = Q_\epsilon^K$ and $\nabla_{\theta_{t+1}} Q_\epsilon^0 = \nabla_{\theta_t} Q_\epsilon^K$
end for
Output: Optimized parameter θ_T and its corresponding upper-level objective $F(\theta_T, \pi_\epsilon^K)$

Algorithm Details. Algorithm 1 is a two-timescale, model-based algorithm. Roughly, it consists of three steps.

1. Pre-learn how the designing parameters sculpt the MDP environment. Specifically, we take the gradient of the transition model $\nabla_\theta \ln P$ and the gradient of the reward function $\nabla_\theta r$ as the input. We remark that learning the environment is actually quite a difficult task. Here, we only give some hints on learning the environmental model. It is possible to learn the transition kernel P or the reward r in advance using an off-line training set (Lim & Autef, 2019). We also remark that the exact derivatives $\nabla_{\theta_t} \ln P$ and $\nabla_{\theta_t} r$ can be substituted by the stochastic gradients learned via zeroth-order gradient estimators (Nesterov & Spokoiny, 2017).

2. In the inner loop, we simultaneously update the regularized policy π_ϵ and the state-action function Q_ϵ. Besides, to calculate the gradient, we also update $\nabla_\theta Q_\epsilon$. If the inner loop is done in a sampling style, we remark that the gradient $\nabla_{\theta_t} Q_\epsilon$ can be updated using the same samples collected for learning the Q function Q_ϵ. This is because the operator $\mathcal{T}$ for updating $\nabla_\theta Q_\epsilon$ shares the same transition kernel with Q_ϵ. See §C for a detailed sample-based algorithm.

3. In the outer loop, the gradient of upper-level objective F can be obtained by (24). Here, we propose using a reference distribution $\rho \in \Delta(\mathcal{S})$ for sampling over the state space. Afterward, we update the parameter θ_t while maintaining Q_ϵ and $\nabla_\theta Q_\epsilon$ for the next inner loop.

4.2. A Special Case: Total Reward As Design Objective

We study the case where the designer's objective corresponds to maximizing the discounted total reward

$$F(\theta, \pi) = \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma_u^i r_u(s_i, a_i; \theta) \,|\, s_0 \sim \mathcal{D}_0, P^\pi(\theta)\right], \tag{25}$$

where $(s_{i+1}, a_{i+1}) \sim P^\pi(\cdot, \cdot \,|\, s_i, a_i; \theta)$. Plugging (25) into (14), we can see that the designer's objective enjoys the same transition kernel P and policy π as the MDP agent but with a different reward function r_u and discounted factor γ_u. We remark that such a setting is common in many applications where the designer tempts to optimize a long-term objective for the MDP environment design, e.g., long-term economical performance in the taxation design example. We first test whether Theorem 3.1 applies in such a case. By the performance difference lemma (Kakade & Langford, 2002), it holds that

$$\begin{aligned}
&|F(\theta, \pi_1) - F(\theta, \pi_2)| \\
&= (1-\gamma_u)^{-1} \left|\mathbb{E}_{s \sim \mathcal{E}_{\mathcal{D}_0}^{\pi_1}} \langle \pi_1 - \pi_2, Q_u \rangle_{\mathcal{A}}\right| \\
&\le (1-\gamma_u)^{-1} \|\pi_1 - \pi_2\|_{a \sim 1, s \sim \infty} \|Q_u\|_\infty,
\end{aligned} \tag{26}$$

which suggests that F is Lipschitz continuous with respect to π as long as r_u is globally bounded. Thus, Theorem 3.1 holds consequently which shows the sub-optimality of such an RMD. Next, we study the gradient of the designer's objective function. For the objective defined in (25), we have the gradient given by the following lemma.

Lemma 4.1. *With the policy regularized by $\Omega(x) = x \ln x$, the gradient of objective (25) with respect to θ in a regularized MDP is given by*

$$\begin{aligned}
\nabla_\theta F(\theta, \pi_\epsilon^*(\theta)) = (1-\gamma_u)^{-1} \mathbb{E}_{\tilde{\mathcal{E}}_{\mathcal{D}_0}^{\pi_\epsilon^*}} \Big[&\nabla_\theta r_u + \epsilon A_u \cdot \nabla_\theta A_\epsilon \\
&+ \gamma_u \mathbb{E}_{P(\theta)}\left[\nabla_\theta \ln P \cdot V_u\right], \Big],
\end{aligned} \tag{27}$$

where

$$V_u(s) = \mathbb{E}_{\pi_\epsilon^*}\left[Q_u(s,a)\right], \tag{28}$$

$$Q_u = \mathcal{T}_{r_u, \gamma_u}^{\theta}(V_u), \tag{29}$$

$$A_u(s,a) = Q_u(s,a) - V_u(s). \tag{30}$$

Proof. See §A.3 for detailed proof. □

By (27), it follows that the gradient of F can also be viewed as a total reward following transition kernel P and policy ${\pi_\epsilon}^*$, where the reward function is given by $\nabla_\theta r_u + \epsilon A_u \cdot \nabla_\theta A_\epsilon + \gamma_u \mathbb{E}_{P(\theta)}[\nabla_\theta \ln P \cdot V_u]$. Hence, to calculate the gradient of F, we can update $\tilde{V}$ and $\tilde{Q}$ defined as follows,

$$\tilde{V}(s) = \mathbb{E}_{\pi_\epsilon^*}\left[\tilde{Q}(s,\cdot)\right], \tag{31}$$

$$\tilde{Q} = \mathcal{T}^{\theta}_{\nabla_\theta r_u + \epsilon A_u \nabla_\theta A_\epsilon, \gamma_u}\left(\tilde{V} + V_u \nabla_\theta \ln P\right) \tag{32}$$

We summarize the algorithm for solving the RMD with total reward (25) as the design objective as follows. Here in the inner loop, the calculation of ${V_\epsilon}^k, \nabla_{\theta_t} {V_\epsilon}^k, V_u^k, \nabla_{\theta_t} A_\epsilon^k, A_u^k, \tilde{V}^k$ follows (11), (21), (28), (23), (30), and (31) respectively, where we just need to substitute the optimal state value function and policy in the equations with the corresponding variables calculated in the kth inner loop. Note that the calculation of the gradient of F is already inherent in the update of $\nabla_\theta Q_\epsilon$ and $\tilde{Q}$ in the inner loop, which is different from Algorithm 1. Combining the discussions in this section, we give some discussions on the regularization hyper-parameter ϵ.

Algorithm 2 Framework for the RMD with the total reward design objective

Input: outer iterations T, inner iterations K, learning rate η, the gradient of pre-learned transition model $\nabla_\theta \ln P$ and the gradient of the reward function $\nabla_\theta r$ with respect to θ, The initial state distribution $\mathcal{D}_0$.
Initialize θ_0, Q_ϵ^0, $\nabla_{\theta_0} Q_\epsilon^0$, and $\tilde{Q}^0$.
for $t = 0$ **to** $T-1$ **do**
  for $k = 0$ **to** $K-1$ **do**
    $\pi_\epsilon^k(\cdot|s) \propto \exp\left(\epsilon Q_\epsilon^k(s,\cdot)\right)$
    Calculate ${V_\epsilon}^k, \nabla_{\theta_t} {V_\epsilon}^k, V_u^k, \nabla_{\theta_t} A_\epsilon^k, A_u^k, \tilde{V}^k$
    $Q_\epsilon^{k+1} = \mathcal{T}_{r,\gamma}({V_\epsilon}^k)$
    $\nabla_{\theta_t} Q_\epsilon^{k+1} = \mathcal{T}_{\nabla_{\theta_t} r, \gamma}(\nabla_{\theta_t} V_\epsilon^k + V_\epsilon^k \nabla_{\theta_t} \ln P)$
    $Q_u^{k+1} = \mathcal{T}_{r_u,\gamma_u}(V_u^k)$
    $\tilde{Q}^{k+1} = \mathcal{T}_{\nabla_{\theta_t} r_u + \epsilon A_u \nabla_{\theta_t} A_\epsilon, \gamma_u}(\tilde{V}^k + V_u^k \nabla_{\theta_t} \ln P)$
  end for
  $\nabla_{\theta_t} F = \mathbb{E}_{\mathcal{D}_0}[\tilde{V}^K]$
  $\theta_{t+1} = \theta_t + \eta \nabla_{\theta_t} F$
  Reintialize $Q_\epsilon^0 = Q_\epsilon^K$, $\nabla_{\theta_{t+1}} Q_\epsilon^0 = \nabla_{\theta_t} Q_\epsilon^K$, and $\nabla_{\theta_{t+1}} Q_\epsilon^0 = \nabla_{\theta_t} Q_\epsilon^K$.
end for
Output: Optimized parameter θ_T and its corresponding upper-level objective $F(\theta_T, \pi_\epsilon^K)$

How to determine ϵ? Note that ϵ decides how much regularization is involved in the policy of the lower-level agent. Specifically, by setting a larger ϵ, less regularization is introduced in the policy according to (11). On the one hand, a larger ϵ produces a smaller gap in the designer's objective function according to Theorem 3.1. On the other hand, a larger ϵ might result in a larger gradient by (24), indicating that the policy becomes more sensitive to the change in the environment, which might cause the algorithm to become less stable. Besides, with less regularization, a larger ϵ can make the landscape of F more complicated, which might cause the adaptive design to fall into some local optimum. Hence, we see that ϵ introduces a trade-off between the accuracy of the objective function value and the convergence performance of the algorithm. To utilize such a trade-off for improved accuracy, stability, and convergence rate, we propose an ϵ adaptive strategy in §6. Experiments comparing the performance of different ϵ and the ϵ-adaptive strategy can be found in §7.

5. Convergence Analysis

In this section, we show that with suitable choices of the learning rate η and maximal inner iteration number K, the general framework (Algorithm 1) is guaranteed to converge to the optimality. Here, we only study the convergence results for Algorithm 1 which is more representative. We remark that a similar result is obtainable for Algorithm 2 with some more careful analysis.

5.1. Convergence of the Inner Loop

We have the following Lemma showing the convergence result of the inner loop under maximal iteration number K for Algorithm 1.

Lemma 5.1 (Convergence of the gradient of Q_ϵ^*)**.** *For every policy iteration step, it holds that*

$$\left\|Q_\epsilon^{k+1} - Q_\epsilon^*\right\|_\infty \leq \gamma \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty. \tag{33}$$

After K inner iterations, it holds that

$$\begin{aligned} &\left\|\nabla_\theta Q_\epsilon^K - \nabla_\theta Q_\epsilon^*\right\|_{\theta\sim 2,(s,a)\sim\infty} \\ &\leq \gamma^K K \left\|Q_\epsilon^0 - Q_\epsilon^*\right\|_\infty \\ &\quad \cdot \left(4\epsilon \left\|\nabla_\theta Q_\epsilon^*\right\|_{\theta\sim 2,(s,a)\sim\infty} + \left\|\nabla_\theta P\right\|_{\theta\sim 2,s'\sim 1,(s,a)\sim\infty}\right) \\ &\quad + \gamma^K \left\|\nabla_\theta Q_\epsilon^0 - \nabla_\theta Q_\epsilon^*\right\|_{\theta\sim 2,(s,a)\sim\infty} \end{aligned} \tag{34}$$

Proof. See §A.4 for detailed proof. □

From the above lemma, we see that $\|{Q_\epsilon}^{k+1} - Q_\epsilon^*\|_\infty \sim O(\gamma^K)$ and that $\left\|\nabla_\theta Q_\epsilon^K - \nabla_\theta Q_\epsilon^*\right\|_{\theta\sim 2,(s,a)\sim\infty} \sim O(\epsilon\gamma^K K)$. Such a result holds by noting that the error in ${V_\epsilon}^k$ is coupled in the update of $\nabla_\theta {Q_\epsilon}^k$. With such a convergence result for the inner loop, we are now ready to establish the convergence result for the outer loop.

5.2. Convergence of the Outer Loop

To show the convergence result of Algorithm 1, we propose the following assumptions on the continuity and the convexity of the objective function F.

Assumption 5.2 (Continuity)**.** We assume that F is $L_{F,\theta,0}$-continuous, $L_{F,\theta,1}$-smooth with respect to θ, and is $L_{F,\pi,0}$-continuous, $L_{F,\pi,1}$-smooth with respect to π. We also assume that the transition kernel P is $L_{P,\theta,0}$-continuous, $L_{P,\theta,1}$-smooth with respect to θ. The reward function r is B_r-bounded, $L_{r,\theta,0}$-continuous, and $L_{r,\theta,1}$-smooth with respect to θ.

A formal statement of Assumption 5.2 is stated in §A.1 including the norm we consider and the definition of Lipschitz continuity/smooth.

Assumption 5.3 (Convexity)**.** For given ϵ and $l_\epsilon(\theta) = -F(\theta, \pi^*_\epsilon(r(\theta)))$, we assume $l_\epsilon(\theta)$ to be convex and θ^* to be the minimizer of $l_\epsilon(\theta)$. Moreover, for any $\mathcal{L} > l_\epsilon(\theta^*)$, by letting $\mathcal{C}_\mathcal{L} = \{\theta \,|\, l_\epsilon(\theta) - l_\epsilon(\theta^*) < \mathcal{L}\}$ be the sublevel set with respect to $\mathcal{L}$, we assume that $\mathcal{C}_\mathcal{L}$ is compact and bounded such that $\|\theta - \theta^*\|_2 < D_\mathcal{L}$ for any $\theta \in C_\mathcal{L}$.

Here, Assumption 5.2 ensures that (1) the environment including the reward and the transition kernel evolves smoothly with the design parameter θ; (2) the objective function $F(\theta, \pi)$ is partially Lipschitz-smooth with respect to θ and π. In the OMD (13), note that π^* can still be sensitive to the change in the MDP environment. That's why we introduce entropy regularization for the adaptive design. Now, we are ready to present the following theorem on the convergence rate of the algorithm.

Theorem 5.4 (Convergence of Algorithm 1)**.** *Let η be the learning rate and ϵ be the regularization parameter. Suppose that Assumptions 5.2 and 5.3 hold. Suppose it holds for the maximal inner iteration number K and the learning rate η that*

$$\beta^\top A_K \left(\hat{\beta} + 4\eta\alpha\right) \leq \left(1 - \frac{4}{3}\eta L_{l,\theta,1}\right), \qquad (35)$$

where

$$A_K = \gamma^K \begin{bmatrix} 1 & 0 \\ C_0 K & 1 \end{bmatrix}, \qquad (36)$$

$\beta, \alpha, \hat{\beta}$ are positive two-element vectors, and $C_0, L_{l,\theta,1}$ are positive coefficients. Moreover, $C_0, L_{l,\theta,1}, \alpha, \beta, \hat{\beta}$ only depends polynomially on ϵ. It then holds for algorithm 1 that

$$l_\epsilon(\theta_T) - l_\epsilon(\theta^*) \leq O(T^{-1/2}) \qquad (37)$$

Proof. See §A.6 for detailed proof. □

Here, by condition (35), for an admitted learning rate η, we allow $\left(K\gamma^K\right)^{-1} \geq \text{poly}(\epsilon)$, which means that K has a logarithmic growth rate with respect to ϵ. Therefore, we are able to do just a few inner updates before updating the parameter θ, even with a large ϵ where the agent's policy becomes sensitive to the changes in the environment. So, there is another trade-off, i.e., a large ϵ improves accuracy but requires more inner iterations to guarantee convergence. Moreover, by (37) we show that the algorithm has a sublinear convergence rate. Specifically, as $T \to \infty$, the objective function will converge to a sub-optimal solution at a rate of $O(T^{-1/2})$.

6. Extensions

6.1. ϵ-Adaptive Strategy

Note that ϵ introduces trade-offs between stability, landscape complexity, required inner iteration number, and accuracy. Specifically, a smaller ϵ introduces more regularization and smoothens the optimization landscape to improve stability while requiring fewer inner iterations. On the other hand, a larger ϵ improves accuracy in the design objective function. To make better use of entropy regularization, we propose an ϵ-adaptive strategy that controls the amount of regularization by tuning ϵ during the algorithm. Specifically, at the beginning of the algorithm, we suggest using a smaller ϵ that simplifies the optimization landscape, avoids some local optimum, and helps push the design parameter θ in the target direction. Then during the update, we adjust ϵ to a larger value step by step. Eventually, the algorithm ends with a large ϵ and results in a smaller gap in the objective function. The strategy is further tested in our experiments to verify our idea. For §7 for detailed examples.

6.2. Sample-based Version

Note that the updates in Algorithm 1 take the form of Bellman update. Hence, we can conduct the algorithm in a sample-based style if we only have access to the changes in the environment, i.e., $\nabla_\theta \ln P$ and $\nabla_\theta r$, without direct knowledge of P and r. Moreover, to deal with a continuous state space, we use a function approximator for estimating Q_ϵ and $\nabla_\theta Q_\epsilon$. The sample-based algorithm is given in the Appendix. See C for more details.

7. Experiments

7.1. Tax Design for Macroeconomic Model

We test our method on a bi-level macroeconomic model based on (Hill et al., 2021) which seeks to explain the impact of tax rates on the social welfare and market behaviours including hours worked and consumption of goods. We assume there is a representative household employed agent in the lower level. At each time step t, the household agent chooses an action with n_t hours' work and $c_{i,t}$ consumption, where $i \in \{1, \cdots, M\}$ denotes the category of goods, each with a price before tax p_i. Let x denote the income tax rate and y_i denote the consumption tax rate for good i, respectively. The utility for the household agent at times step t is given by $u_t = \sigma(s_t) - \theta n_t^2 + \prod_{i=1}^M \left(c_{i,t}/\left(p_i(1+y_i)\right)\right)^{\alpha_i}$, where the product-of-consumption term corresponds to the

Cobb-Douglas function (Roth et al., 2016) and $\sigma(s_t)$ is the reward for accumulative asset s_t updated at each time step by $s_{t+1} = s_t + (1-x)wn_t - \sum_{i=1}^{M} c_{i,t}$. The social welfare at time step t is given by $v_t = \xi(s_t) + \sum_{i=1}^{M} c_{i,t}/(1+y_i) + \phi \ln\left(\sum_{i=1}^{M} c_{i,t}y_i/(1+y_i) + wxn_t\right)$, where $\xi(\cdot)$ is the reward for the accumulative asset, ϕ is a positive constant. While the household agent follows a policy that maximizes its discounted accumulative reward $U = \sum_{t=0}^{\infty} \gamma_1^t u_t$, the social planner aims to maximize the discounted total social welfare $V = \sum_{t=0}^{\infty} \gamma_2^t v_t$ by tuning the tax rates.

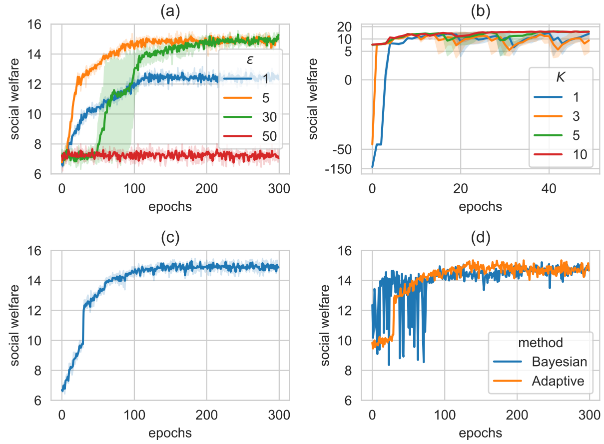

Figure 1. The design objective (discounted total social welfare) with respect to epochs. (a) are tested with ϵ set to 1, 5, 30 and 50. (b) are tested with K set to 1, 3, 5 and 10. (c) adopts the ϵ-adaptive strategy, i.e., ϵ is set to 1, 5, 10, 20, 35, 50 at epoch 0, 30, 60, 90, 150, 200, respectively. (d) compares the adaptive strategy with the lower-level RL and upper-level Bayesian Optimization method used in (Mguni et al., 2019). The adaptation of ϵ follows (c) and we only update the consumption tax rate y_i while income tax rate x remains unchanged.

7.2. Workbench Position Design for Two-Ankle Robotic Arm

We also test our method on a 2D robotic arm environment. We assume that there is a workbench to process several components where the position of each component is fixed. There is a robotic arm with two ankles to fetch these components and put them on the workbench for processing. The designer aims to find the optimal workbench position $p = (x, y)$ that takes the least energy consumption for the robotic arm to finish the component transportation task. We adopt discretized angles θ and angular velocities ω of these two ankles as the joint state space. The action space corresponds to the representative angular acceleration $a \in \{-1, 0, 1\}^2$ at each time step for these two ankles. The transition kernel is given by $(\theta_{t+1}, \omega_{t+1}) = (\theta_t + \omega_t, \omega_t + a_t)$. The agent or the robotic arm is programmed to take the squared distance from its end to the workbench and squared angular velocities of its two ankles as the reward. The designer's objective corresponds to minimizing the discounted total energy consumption. For simplicity, we assume the energy consumption for each movement to be $c_t = |a_t \cdot \omega_t|$.

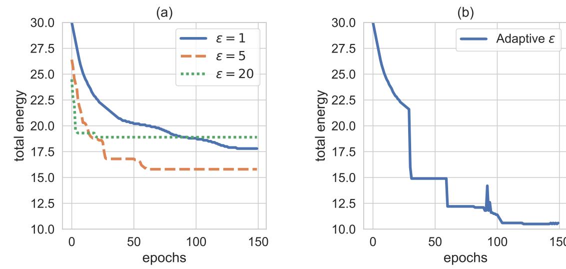

Figure 2. The total energy consumption with respect to epochs. (a) corresponds to setting different ϵ and (b) corresponds to using the ϵ-adaptive strategy, i.e., ϵ is set to 1, 5, 10, 15, 20 at epoch 0, 30, 60, 90, 120, respectively.

7.3. Result Analysis

Selection of ϵ. We see from both (a) in Figure 1 and (a) in Figure 2 that a median ϵ is generally better than an extreme ϵ, in the sense of both convergence rate and accuracy in the design objective function. For example, in the first experiment on taxation design, we observe that a small ϵ yields a large gap in the optimal design objective ($\epsilon = 1$ in (a), Figure 1). On the other hand, Setting a large ϵ might cause the algorithm to be trapped by some local optimum ($\epsilon = 50$ in (a), Figure 1; $\epsilon = 20$ in (a), Figure 2). Even as the algorithm eventually reaches the optimum ($\epsilon = 30$ in (a), Figure 1), it produces large variation during the update and has a slower convergence rate.

ϵ-adaptive strategy. From both (c) in Figure 1 and (b) in Figure 2, we can observe that the ϵ-adaptive strategy converges fast to the optimal in both experiments. Hence, if the ϵ-adaptive strategy is properly designed, hopefully, we can escape some local optimums and reach the global optimum with a small gap in the design objective.

The influence of inner iterations K (35) in Theorem 5.4 indicates that the convergence of Algorithm 1 is guaranteed with enough inner iterations. Intuitively, without enough inner iterations, especially when the initial value function is far from the optimal one, we may encounter a misleading in the optimization phase. To further explore the influence of inner iterations K, we conduct experiments with K set to 1, 3, 5, and 10, respectively. (b) in Figure 1 shows that a small K can mislead the designing parameter at the starting point and yield an unstable learning curve. Such an effect is illustrated by the fact that we do not accurately estimate the agent's optimal policy under a small number of internal iterations.

Comparison with Bayesian Optimization We conduct an experiment comparing the algorithm in Mguni et al. (2019) and ours. Mguni et al. (2019) uses the Bayesian Optimization to determine the optimal modifications of the agents' rewards that result in optimal system performance. To make the comparison fair, in (d) of Figure 1, we only tune the consumption tax rates for both Bayesian Optimization and the adaptive strategy. Results show that our method performs competitively with the Bayesian Optimization method when designing the reward only.

Acknowledgements

Zhaoran Wang acknowledges National Science Foundation (Awards 2048075, 2008827, 2015568, 1934931), Simons Institute (Theory of Reinforcement Learning), Amazon, J.P. Morgan, and Two Sigma for their supports.

References

Ahmed, Z., Le Roux, N., Norouzi, M., and Schuurmans, D. Understanding the impact of entropy on policy optimization. In *International conference on machine learning*, pp. 151–160. PMLR, 2019.

Asadi, K. and Littman, M. L. An alternative softmax operator for reinforcement learning. In *International Conference on Machine Learning*, pp. 243–252. PMLR, 2017.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019.

Colson, B., Marcotte, P., and Savard, G. An overview of bilevel optimization. *Annals of operations research*, 153 (1):235–256, 2007.

Dai, B., Shaw, A., Li, L., Xiao, L., He, N., Liu, Z., Chen, J., and Song, L. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning*, pp. 1125–1134. PMLR, 2018.

Derman, C., Lieberman, G. J., and Ross, S. M. A stochastic sequential allocation model. *Operations Research*, 23(6): 1120–1130, 1975.

Dütting, P., Feng, Z., Narasimhan, H., Parkes, D., and Ravindranath, S. S. Optimal auctions through deep learning. In *International Conference on Machine Learning*, pp. 1706–1715. PMLR, 2019.

Ehtamo, H., Kitti, M., and Hämäläinen, R. P. Recent studies on incentive design problems in game theory and management science. In *Optimal Control and Differential Games*, pp. 121–134. Springer, 2002.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic metalearning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., and Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, pp. 1568–1577. PMLR, 2018.

Geist, M., Scherrer, B., and Pietquin, O. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pp. 2160–2169. PMLR, 2019.

Golabi, K., Kulkarni, R. B., and Way, G. B. A statewide pavement management system. *Interfaces*, 12(6):5–21, 1982.

Hill, E., Bardoscia, M., and Turrell, A. Solving heterogeneous general equilibrium economic models with deep reinforcement learning. *arXiv preprint arXiv:2103.16977*, 2021.

Hong, M., Wai, H.-T., Wang, Z., and Yang, Z. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*, 2020.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning*. Citeseer, 2002.

Koenig, S., Simmons, R., et al. Xavier: A robot navigation architecture based on partially observable markov decision process models. *Artificial Intelligence Based Mobile Robotics: Case Studies of Successful Robot Systems*, (partially):91–122, 1998.

Lawphongpanich, S. and Hearn, D. W. An mpec approach to second-best toll pricing. *Mathematical programming*, 101(1):33–55, 2004.

Li, J., Yu, J., Nie, Y., and Wang, Z. End-to-end learning and intervention in games. *Advances in Neural Information Processing Systems*, 33:16653–16665, 2020.

Li, S. H., Yu, Y., Calderone, D., Ratliff, L., and Açrkmeşe, B. Tolling for constraint satisfaction in markov decision process congestion games. In *2019 American Control Conference (ACC)*, pp. 1238–1243. IEEE, 2019.

Lim, S. H. and Autef, A. Kernel-based reinforcement learning in robust markov decision processes. In *International Conference on Machine Learning*, pp. 3973–3981. PMLR, 2019.

Little, J. D. The use of storage water in a hydroelectric system. *Journal of the Operations Research Society of America*, 3(2):187–197, 1955.

Liu, B., Li, J., Yang, Z., Wai, H.-T., Hong, M., Nie, Y. M., and Wang, Z. Inducing equilibria via incentives: Simultaneous design-and-play finds global optima. *arXiv preprint arXiv:2110.01212*, 2021a.

Liu, R., Gao, J., Zhang, J., Meng, D., and Lin, Z. Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond. *arXiv preprint arXiv:2101.11517*, 2021b.

Maclaurin, D., Duvenaud, D., and Adams, R. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pp. 2113–2122. PMLR, 2015.

Maskin, E. S. The invisible hand and externalities. *The American Economic Review*, 84(2):333–337, 1994.

Metelli, A. M., Mutti, M., and Restelli, M. Configurable markov decision processes. In *International Conference on Machine Learning*, pp. 3491–3500. PMLR, 2018.

Mguni, D., Jennings, J., Macua, S. V., Sison, E., Ceppi, S., and De Cote, E. M. Coordinating the crowd: Inducing desirable equilibria in non-cooperative systems. *arXiv preprint arXiv:1901.10923*, 2019.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing atari with deep reinforcement learning, 2013.

Nesterov, Y. and Spokoiny, V. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Onstad, D. W. and Rabbinge, R. Dynamic programming and the computation of economic injury levels for crop disease control. *Agricultural Systems*, 18(4):207–226, 1985.

Ouyang, Y. Pavement resurfacing planning for highway networks: parametric policy iteration approach. *Journal of infrastructure systems*, 13(1):65–71, 2007.

Rajeswaran, A., Finn, C., Kakade, S., and Levine, S. Meta-learning with implicit gradients. 2019.

Requate, T. Pollution control in a cournot duopoly via taxes or permits. *Journal of Economics*, 58(3):255–291, 1993.

Roth, A., Ullman, J., and Wu, Z. S. Watch and learn: Optimizing from revealed preferences feedback. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 949–962, 2016.

Russell, C. B. An optimal policy for operating a multipurpose reservoir. *Operations Research*, 20(6):1181–1189, 1972.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. PMLR, 2015.

Stackelberg, H. v. Theory of the market economy. 1952.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Symonds, G. H. Solution method for a class of stochastic scheduling problems for the production of a single commodity. *Operations Research*, 19(6):1459–1466, 1971.

A. Proofs of Main Results

A.1. Proof of Theorem 3.1

Proof. Let $\epsilon = (\Delta_r)^{-1} \cdot (\gamma U_\Omega + (1+\gamma)\cdot \log(2|\mathcal{A}|/\Delta_\pi))$, there exists a optimal parameter θ and a unique optimal policy $\pi_\epsilon{}^*(r_\theta)$ that maximize the objective function F. Let $\hat{\mathcal{R}} = \{r' : \mathcal{S}\times\mathcal{A}\times\mathcal{X} \to \mathbb{R} \mid \|r' - r_\theta\|_\infty < \Delta_r\}$. According to Lemma B.1, there exists $\hat{r} \in \hat{\mathcal{R}}$ and $\pi^*(\hat{r})$ satisfying the optimal Bellman equation and $\|\pi^*(\hat{r}) - \pi_\epsilon{}^*(r_\theta)\|_{a\sim 1, s\sim\infty} < \Delta_\pi$. Thus, for any θ, it follows that

$$\begin{aligned} F(\theta, \pi_\epsilon^*(r_\theta)) &\le F(\theta, \pi^*(\hat{r})) + \Delta_\pi L_{F,\pi,0} \\ &\le \sup_{\pi\in\Pi^*(P(\theta),\hat{r}(\theta)),\hat{r}(\theta)\in\hat{R}} F(\theta,\pi) + \Delta_\pi L_{F,\pi,0}. \end{aligned} \tag{38}$$

Here, the first inequality holds since $\|F(\theta, \pi^*(r_\theta)) - F(\theta, \pi^*(\hat{r}))\| \le \Delta_\pi L_{F,\pi,0}$. Similarly, we have

$$\begin{aligned} F(\theta, \pi_\epsilon^*(r_\theta)) &\ge F(\theta, \pi^*(\hat{r})) - \Delta_\pi L_{F,\pi,0} \\ &\ge \inf_{\pi\in\Pi^*(P(\theta),\hat{r}(\theta)),\hat{r}(\theta)\in\hat{R}} F(\theta,\pi) - \Delta_\pi L_{F,\pi,0}. \end{aligned} \tag{39}$$

Taking an supreme over the parameter θ gives the result. Thus we complete our proof. ☐

A.2. The formulation of gradients with respect to the design parameter θ

Plugging (12) into $\sum_a \pi_\epsilon^*(a|s) = 1$, it holds that

$$\sum_a \varphi(\epsilon(Q_\epsilon^*(s,a) + v)) = 1. \tag{40}$$

Take the derivative on both sides of the above equation, we have

$$\sum_a \dot{\varphi}(\epsilon(Q_\epsilon^*(s,a) + v))(\nabla_\theta Q_\epsilon^*(s,a) + \nabla_\theta v) = 0. \tag{41}$$

Following the above equilibrium, it holds that

$$\nabla_\theta v = -\frac{\sum_a \dot{\varphi}(\epsilon(Q_\epsilon^*(s,a)+v))\nabla_\theta Q_\epsilon^*(s,a)}{\sum_a \dot{\varphi}(\epsilon(Q_\epsilon^*(s,a)+v))}. \tag{42}$$

Thus, the gradient of the regularized policy π_ϵ^* with respect to the design parameter θ is given by

$$\begin{aligned} \nabla_\theta \pi_\epsilon^*(a|s) &= \epsilon\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a)+v))\Big(\nabla_\theta Q_\epsilon^*(s,a) + \nabla_\theta v\Big) \\ &= \frac{\epsilon\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a)+v))\sum_{a'}\Big(\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a')+v))(\nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta Q_\epsilon^*(s,a'))\Big)}{\sum_{a''}\dot{\varphi}(\epsilon(Q_\epsilon^*(s,a'')+v))}, \end{aligned} \tag{43}$$

where the second equality holds by plugging in (42).

Consider a special case where $\Omega(x) = x\ln x$. It is observed that $\varphi(x) = \dot{\Omega}^{-1}(x) = \exp(x-1)$. The above expression can be simplified as follows

$$\begin{aligned} \nabla_\theta \pi_\epsilon^*(a|s) &= \epsilon\cdot\pi_\epsilon^*(s,a)\sum_{a'}\pi_\epsilon^*(s,a')(\nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta Q_\epsilon^*(s,a')) \\ &= \epsilon\cdot\pi_\epsilon^*(s,a)(\nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta V_\epsilon^*(s,a')) \\ &= \epsilon\cdot\pi_\epsilon^*(s,a)\nabla_\theta A_\epsilon^*(s,a). \end{aligned} \tag{44}$$

Here, the second equality holds by the fact that $\nabla_\theta V_\epsilon^*(s) = \mathbb{E}_{\pi_\epsilon{}^*(\cdot\,|\,s)}\left[\nabla_\theta Q_\epsilon^*(s,\cdot)\right]$, which is derived from the property of Legendre transformation $\nabla_{Q_\epsilon^*(s,a)} V_\epsilon^*(s,a) = \pi_\epsilon^*(a\mid s)$. The last equality follows from that $\nabla_\theta A_\epsilon^*(s,a) = \nabla_\theta Q_\epsilon^*(s,a) - \nabla_\theta V_\epsilon^*(s)$.

Taking the derivative on both sides of (10), we obtain

$$\begin{aligned} \nabla_\theta Q_\epsilon^*(s,a) &= \nabla_\theta r(s,a) + \gamma\mathbb{E}_{P(\cdot\,|\,s,a;\theta)}\left[\nabla_\theta V_\epsilon^*(\cdot)\right] + \gamma\mathbb{E}_{P(\cdot\,|\,s,a;\theta)}\left[V_\epsilon^*(\cdot)\nabla_\theta \ln P(\cdot\mid s,a)\right] \\ &= \mathcal{T}^\theta_{\nabla_\theta r,\gamma}\left(\nabla_\theta V_\epsilon^* + V_\epsilon^*\nabla_\theta \ln P\right). \end{aligned} \tag{45}$$

A.3. Proof of Lemma 4.1

Proof. If we fix r_u and P, by the performance difference lemma, it holds that

$$\mathrm{d}F\mid_{r_u,P}= (1-\gamma_u)^{-1}\mathbb{E}_{s\sim\mathcal{E}_{\mathcal{D}_0}^{\pi_\epsilon^*+\mathrm{d}\pi_\epsilon^*}}\left[\langle \mathrm{d}\pi_\epsilon^*(\cdot\mid s), Q_u^*(s,\cdot)\rangle_{\mathcal{A}}\right]. \tag{46}$$

If we fix $\pi_\epsilon{}^*$, it holds that

$$\mathrm{d}F\mid_{\pi_\epsilon{}^*} = (1-\gamma_u)^{-1}\mathbb{E}_{s\sim\mathcal{E}_{\mathcal{D}_0}^{\pi_\epsilon{}^*}}\left[\langle \pi_\epsilon{}^*, \mathrm{d}r_u+\gamma_u\langle \mathrm{d}P, V_u^*\rangle_{\mathcal{S}'}\rangle_{\mathcal{A}}\right]. \tag{47}$$

Note that

$$\mathrm{d}\pi_\epsilon^* = \epsilon\pi_\epsilon^*\mathrm{d}A_\epsilon. \tag{48}$$

Therefore, we have

$$\begin{aligned}\nabla_\theta F &= (1-\gamma_u)^{-1}\mathbb{E}_{(s,a)\sim\tilde{\mathcal{E}}_{\mathcal{D}_0}^{\pi_\epsilon^*}}\left[\epsilon\nabla_\theta A_\epsilon\cdot Q_u^* + \mathrm{d}r_u+\gamma_u\langle \mathrm{d}P, V_u^*\rangle_{\mathcal{S}'}\right]\\ &= (1-\gamma_u)^{-1}\mathbb{E}_{(s,a)\sim\tilde{\mathcal{E}}_{\mathcal{D}_0}^{\pi_\epsilon^*}}\left[\epsilon\nabla_\theta A_\epsilon\cdot A_u^* + \mathrm{d}r_u+\gamma_u\langle \mathrm{d}P, V_u^*\rangle_{\mathcal{S}'}\right]\end{aligned} \tag{49}$$

□

A.4. Proof of Lemma 5.1

Proof. The contraction of Q_ϵ in (33) for each policy iteration step was shown in various forms, e.g.(Asadi & Littman, 2017; Dai et al., 2018; Geist et al., 2019). Following their results, we have

$$\left\|Q_\epsilon^{k+1}-Q_\epsilon^*\right\|_\infty \le \gamma\left\|Q_\epsilon^k-Q_\epsilon^*\right\|_\infty. \tag{50}$$

Since $\pi_\epsilon = \mathrm{softmax}(\epsilon Q_\epsilon)$, it holds that

$$\begin{aligned}\left\|\pi_\epsilon^k-\pi_\epsilon^*\right\|_{a\sim1,s\sim\infty} &= \left\|\frac{\sum_{a\in\mathcal{A}}\left|e^{\epsilon Q_\epsilon^k(\cdot,a)}\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*(\cdot,a)} - e^{\epsilon Q_\epsilon^*(\cdot,a)}\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon{}^k(\cdot,a)}\right|}{\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k(\cdot,a)}\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*(\cdot,a)}}\right\|_{s\sim\infty}\\ &\le \left\|\frac{\sum_{a\in\mathcal{A}}\left|e^{\epsilon Q_\epsilon^k}-e^{\epsilon Q_\epsilon^*}\right|\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}+\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}\left|\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}-\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}\right|}{\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}}\right\|_{s\sim\infty}\\ &= \left\|\frac{\sum_{a\in\mathcal{A}}\left|e^{\epsilon Q_\epsilon^k}-e^{\epsilon Q_\epsilon^*}\right|+\left|\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}-\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}\right|}{\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}}\right\|_{s\sim\infty}.\end{aligned} \tag{51}$$

Here, the second inequality can be derived from the Cauchy–Schwartz inequality. Similarly, it also holds that

$$\left\|\pi_\epsilon^k-\pi_\epsilon{}^*\right\|_{a\sim1,s\sim\infty} \le \left\|\frac{\sum_{a\in\mathcal{A}}\left|e^{\epsilon Q_\epsilon^k}-e^{\epsilon Q_\epsilon^*}\right|+\left|\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}-\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}\right|}{\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}}\right\|_{s\sim\infty} \tag{52}$$

Combining two inequality above, (51) is further bounded by

$$\begin{aligned}\left\|\pi_\epsilon^k-\pi_\epsilon^*\right\|_{a\sim1,s\sim\infty} &\le 2\left\|\frac{\sum_{a\in\mathcal{A}}\left|e^{\epsilon Q_\epsilon^k}-e^{\epsilon Q_\epsilon^*}\right|+\left|\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}-\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}\right|}{\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^k}+\sum_{a\in\mathcal{A}}e^{\epsilon Q_\epsilon^*}}\right\|_{s\sim\infty}\\ &\le 4\left\|\frac{\sum_{a\in\mathcal{A}}e^{\epsilon\max\left\{Q_\epsilon^k,Q_\epsilon^*\right\}}\left(1-e^{-\epsilon\left|Q_\epsilon^k-Q_\epsilon^*\right|}\right)}{\sum_{a\in\mathcal{A}}e^{\epsilon\max\left\{Q_\epsilon^k,Q_\epsilon^*\right\}}}\right\|_{s\sim\infty}\\ &= 4\left\|1-e^{-\epsilon\left|Q_\epsilon^k-Q_\epsilon{}^*\right|}\right\|_\infty\\ &\le 4\epsilon\left\|Q_\epsilon^k-Q_\epsilon^*\right\|_\infty.\end{aligned} \tag{53}$$

For simplicity, we denote $\|\cdot\|_{\theta\sim 2,(s,a)\sim\infty}$ by $\|\cdot\|_{2,\infty}$. Now for the iteration of the gradient of the state-action value function, it holds that

$$\begin{aligned}
\left\|\nabla_\theta Q_\epsilon^{k+1} - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} &\le \gamma \cdot \left(\left\| \langle P^{\pi_\epsilon^k}, \nabla_\theta Q_\epsilon^k \rangle_{\mathcal{S}\times\mathcal{A}} - \langle P^{\pi_\epsilon^*}, \nabla_\theta Q_\epsilon^* \rangle_{\mathcal{S}\times\mathcal{A}} \right\|_{2,\infty} + \left\| \langle \nabla_\theta P, V_\epsilon^k - V_\epsilon^* \rangle_{\mathcal{S}} \right\|_{2,\infty} \right) \\
&\le \gamma \cdot \left(\left\| \langle P^{\pi_\epsilon^k} - P^{\pi_\epsilon^*}, \nabla_\theta Q_\epsilon^* \rangle_{\mathcal{S}\times\mathcal{A}} \right\|_{2,\infty} + \left\| \langle P^{\pi_\epsilon^k}, \nabla_\theta {Q_\epsilon}^k - \nabla_\theta Q_\epsilon^* \rangle_{\mathcal{S}\times\mathcal{A}} \right\|_{2,\infty} \right) \\
&\quad + \gamma \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty \\
&\le \gamma \cdot \left(\left\| \langle P^{\pi_\epsilon^k} - P^{\pi_\epsilon^*}, \mathbf{1} \rangle_{\mathcal{S}\times\mathcal{A}} \right\|_{\infty} \cdot \left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} + \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&\quad + \gamma \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty .
\end{aligned} \tag{54}$$

Here, the first inequality can be derived from the Bellman Equation and the second inequality follows the triangle inequality. The last inequality can be derived from the Cauchy–Schwartz inequality and the fact that $\left\|V_\epsilon^k - V_\epsilon^*\right\|_\infty \le \|\langle \pi_\epsilon^k, Q_\epsilon^k(s,\cdot) - Q_\epsilon^*(s,\cdot)\rangle_{\mathcal{A}}\|_\infty \le \|Q_\epsilon^k - Q_\epsilon^*\|_\infty$.

Considering $\left\| \langle P^{\pi_\epsilon^k} - P^{\pi_\epsilon^*}, \mathbf{1} \rangle_{\mathcal{S}\times\mathcal{A}} \right\|_{\infty} = \left\| \sum_{s',a'} P(s'|\cdot,\cdot)\pi_\epsilon^k(a' \mid s') - P(s'|\cdot,\cdot)\pi_\epsilon^*(a' \mid s') \right\|_\infty$, which is bounded by $\left\| \sum_{s'} P(s'|\cdot,\cdot) \left\|\pi_\epsilon^k(\cdot \mid s') - \pi_\epsilon^*(\cdot \mid s')\right\|_1 \right\|_\infty$, we obtain

$$\begin{aligned}
\left\|\nabla_\theta Q_\epsilon^{k+1} - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} &\le \gamma \cdot \left(\left\| \sum_{s'} P(s'|\cdot,\cdot) \left\|\pi_\epsilon^k(\cdot \mid s') - \pi_\epsilon^*(\cdot \mid s')\right\|_1 \right\|_\infty \left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} + \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&\quad + \gamma \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty \\
&\le \gamma \cdot \left(\left\|\pi_\epsilon^k - \pi_\epsilon^*\right\|_{a\sim 1, s\sim\infty} \cdot \left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} + \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&\quad + \gamma \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty .
\end{aligned} \tag{55}$$

Plugging (53) into (55), we obtain

$$\begin{aligned}
\left\|\nabla_\theta Q_\epsilon^{k+1} - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} &\le \gamma \cdot \left(\left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \cdot \left(4\epsilon \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty\right) + \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&\quad + \gamma \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^k - Q_\epsilon^*\right\|_\infty \\
&\le \gamma \cdot \left(\left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \cdot \left(4\epsilon\gamma^k \left\|Q_\epsilon^0 - Q_\epsilon^*\right\|_\infty\right) + \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&\quad + \gamma^{k+1} \left\|\nabla_\theta P\right\|_{\theta_2, s'\sim 1, s, a\sim\infty} \left\|Q_\epsilon^0 - Q_\epsilon^*\right\|_\infty \\
&= \gamma^{k+1} \cdot C_1 \cdot C_2 + \gamma \left\|\nabla_\theta Q_\epsilon^k - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty},
\end{aligned} \tag{56}$$

where $C_1 = 4\epsilon \cdot \left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} + \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty}$ and $C_2 = \left\|Q_\epsilon^0 - Q_\epsilon^*\right\|_\infty$. Taking this inequality for $k = 0, 1, \cdots, K-1$ and summing them up give

$$\begin{aligned}
\left\|\nabla_\theta {Q_\epsilon}^K - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} &\le \gamma^K \left(K C_1 C_2 + \left\|\nabla_\theta Q_\epsilon^0 - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty} \right) \\
&= \gamma^K \left(4\epsilon K \left\|\nabla_\theta Q_\epsilon^*\right\|_{2,\infty} + K \left\|\nabla_\theta P\right\|_{\theta\sim 2, s'\sim 1, s, a\sim\infty} \right) \cdot \left\|Q_\epsilon^0 - Q_\epsilon^*\right\|_\infty \\
&\quad + \gamma^K \left\|\nabla_\theta Q_\epsilon^0 - \nabla_\theta Q_\epsilon^*\right\|_{2,\infty}
\end{aligned} \tag{57}$$

□

A.5. Restatement of Assumption 5.2

Assumption A.1 (Continuity, restatement of Assumption 5.2)**.** We assume $F(\theta, \pi)$ is twice differentiable and that

$$\frac{|F(\theta_1,\cdot) - F(\theta_2,\cdot)|}{\|\theta_1 - \theta_2\|_2} \le L_{F,\theta,0}, \quad \frac{\|\nabla_\theta F(\theta_1,\cdot) - \nabla_\theta F(\theta_2,\cdot)\|_2}{\|\theta_1 - \theta_2\|_2} \le L_{F,\theta,1}, \tag{58}$$

$$\frac{|F(\cdot,\pi_1) - F(\cdot,\pi_2)|}{\|\pi_1 - \pi_2\|_{a\sim 1, s\sim\infty}} \le L_{F,\pi,0}, \quad \frac{\|\nabla_\pi F(\cdot,\pi_1) - \nabla_\pi F(\cdot,\pi_2)\|_{a\sim\infty, s\sim 1}}{\|\pi_1 - \pi_2\|_{a\sim 1, s\sim\infty}} \le L_{F,\pi,1}. \tag{59}$$

We also assume that $r(\cdot,\cdot;\theta)$ and $P(\cdot\,|\,\cdot,\cdot;\theta)$ are twice differentiable and that

$$\frac{\|r(s,a;\theta_1)-r(s,a;\theta_2)\|_{\theta\sim 2,(s,a)\sim\infty}}{\|\theta_1-\theta_2\|_2}\le L_{r,\theta,0},\quad \frac{\|\nabla_\theta r(s,a;\theta_1)-\nabla_\theta r(s,a;\theta_2)\|_{\theta\sim 2,(s,a)\sim\infty}}{\|\theta_1-\theta_2\|_2}< L_{r,\theta,1}, \tag{60}$$

$$\frac{\|\nabla_\theta P(s'\,|\,s,a;\theta_1)-\nabla_\theta P(s'\,|\,s,a;\theta_2)\|_{\theta\sim 2,s'\sim 1,(s,a)\sim\infty}}{\|\theta_1-\theta_2\|_2}\le L_{P,\theta,1}, \tag{61}$$

$$\frac{\|P(s'\,|\,s,a;\theta_1)-P(s'\,|\,s,a;\theta_2)\|_{s'\sim 1,(s,a)\sim\infty}}{\|\theta_1-\theta_2\|_2}\le L_{P,\theta,0}. \tag{62}$$

Moreover, $|r(s,a;\theta)|<B_r$ for $\forall s\in\mathcal{S},a\in\mathcal{A},\theta\in\mathcal{X}$.

A.6. Proof of Theorem 5.4

Assumption A.2. Any α-sublevel set is compact and bounded, i.e., $\|S_\alpha-\theta^*\|\le D_\alpha$.

Conditions. We require the following two conditions.

$$\text{Condition 1:}\quad \sigma_K\triangleq\beta^\top A_K\left(\hat\beta+(1+\lambda^{-1})\eta\alpha\right)\le\left(1-\eta L_{l,\theta,1}(1+\lambda)\right). \tag{63}$$

$$\text{Condition 2:}\quad w\triangleq\frac{1-2\lambda}{1-\lambda}-\frac{\eta L_{l,\theta,1}}{2}>0. \tag{64}$$

We remark that by taking $\lambda=1/3$, Condition 2 can be guaranteed by Condition 1 and these two conditions are summarized by (35).

Proof. Suppose we have $Q_\epsilon^{*1},Q_\epsilon^{*2},\nabla_\theta Q_\epsilon^{*1},\nabla_\theta Q_\epsilon^{*2}$. Let $\pi_\epsilon^*(\cdot\,|\,s)\propto\exp\{\epsilon Q(\cdot\,|\,s)\}$. Following the result in A.4, it then holds that

$$\|\pi_\epsilon^{*1}-\pi_\epsilon^{*2}\|_{a\sim 1,s\sim\infty}\le 4\epsilon\|Q_\epsilon^{*1}-Q_\epsilon^{*2}\|_{(s,a)\sim\infty}. \tag{65}$$

Suppose $V_\epsilon^*(s)=\langle\pi,Q_\epsilon^*(s,\cdot)\rangle_{\mathcal{A}}-\epsilon^{-1}\sum_{a\in\mathcal{A}}\Omega(\pi(a\,|\,s))$, it then holds that

$$\|V_\epsilon^{*1}-V_\epsilon^{*2}\|_{s\sim\infty}\le\|\langle\pi_\epsilon^{*1},Q_\epsilon^{*1}(s,\cdot)-Q_\epsilon^{*2}(s,\cdot)\rangle_{\mathcal{A}}\|_{s\sim\infty}\le\|Q_\epsilon^{*1}-Q_\epsilon^{*2}\|_{(s,a)\sim\infty}. \tag{66}$$

For simplicity, we denote both $\|\cdot\|_{\theta\sim 2,(s,a)\sim\infty}$ and $\|\cdot\|_{\theta\sim 2,s\sim\infty}$ by $\|\cdot\|_{2,\infty}$. Suppose $\nabla_\theta V_\epsilon^*(s)=\langle\pi_\epsilon^*(\cdot\,|\,s),\nabla_\theta Q_\epsilon^*(s,\cdot)\rangle_{\mathcal{A}}$, it then holds that

$$\begin{aligned}\|\nabla_\theta V_\epsilon^{*1}-\nabla_\theta V_\epsilon^{*2}\|_{2,\infty}&\le\|\langle\pi_\epsilon^{*1}-\pi_\epsilon^{*2},\nabla_\theta Q_\epsilon^{*1}\rangle_{\mathcal{A}}\|_{2,\infty}+\|\langle\pi_\epsilon^{*2},\nabla_\theta Q_\epsilon^{*1}-\nabla_\theta Q_\epsilon^{*2}\rangle_{\mathcal{A}}\|_{2,\infty}\\&\le 4\epsilon\|\nabla_\theta Q_\epsilon^{*1}\|_{2,\infty}\|Q_\epsilon^{*1}-Q_\epsilon^{*2}\|_\infty+\|\nabla_\theta Q_\epsilon^{*1}-\nabla_\theta Q_\epsilon^{*2}\|_{2,\infty}.\end{aligned} \tag{67}$$

Suppose $\nabla_\theta A_\epsilon(\cdot,s)=\nabla_\theta Q_\epsilon^*(\cdot,s)-\nabla_\theta V_\epsilon^*(s)$, it then holds that

$$\begin{aligned}\|\nabla_\theta A_\epsilon{}^1-\nabla_\theta A_\epsilon{}^2\|_{2,\infty}&\le\|\nabla_\theta Q_\epsilon^{*1}-\nabla_\theta Q_\epsilon^{*2}\|_{2,\infty}+\|\nabla_\theta V_\epsilon^{*1}-\nabla_\theta V_\epsilon^{*2}\|_{2,\infty}\\&\le 4\epsilon\|\nabla_\theta Q_\epsilon^{*1}\|_{2,\infty}\|Q_\epsilon^{*1}-Q_\epsilon^{*2}\|_\infty+2\|\nabla_\theta Q_\epsilon^{*1}-\nabla_\theta Q_\epsilon^{*2}\|_{2,\infty}.\end{aligned} \tag{68}$$

Suppose $-\nabla_\theta l_\epsilon=\partial F/\partial\theta+\epsilon\langle\partial F/\partial\pi_\epsilon^*,\pi_\epsilon^*\nabla_\theta A_\epsilon\rangle_{\mathcal{S}\times\mathcal{A}}$, it then holds that

$$\begin{aligned}\|\nabla_\theta l_\epsilon^1-\nabla_\theta l_\epsilon^2\|_2&\le\Big\|\frac{\partial F^1}{\partial\theta}-\frac{\partial F^2}{\partial\theta}\Big\|_2+\epsilon\Big\|\Big\langle\frac{\partial F^1}{\partial\pi_\epsilon^{*1}}-\frac{\partial F^2}{\partial\pi_\epsilon^{*2}},\pi_\epsilon^{*1}\nabla_\theta A_\epsilon{}^1\Big\rangle_{\mathcal{S}\times\mathcal{A}}\Big\|_2\\&\quad+\epsilon\Big\|\Big\langle\frac{\partial F_2}{\partial\pi_\epsilon^{*2}},\pi_\epsilon^{*1}\nabla_\theta A_\epsilon{}^1-\pi_\epsilon^{*2}\,\nabla_\theta A_\epsilon{}^2\Big\rangle_{\mathcal{S}\times\mathcal{A}}\Big\|_2\\&\le L_{F,\theta,1}\|\theta_1-\theta_2\|_2+\epsilon\Big\|\frac{\partial F^1}{\partial\pi_\epsilon^{*1}}-\frac{\partial F^2}{\partial\pi_\epsilon^{*2}}\Big\|_{a\sim\infty,s\sim 1}\|\nabla_\theta A_\epsilon{}^1\|_\infty\\&\quad+\epsilon\|\frac{\partial F^2}{\partial\pi_\epsilon^{*2}}\|_{a\sim\infty,s\sim 1}\left(\|\left(\pi_\epsilon^{*1}-\pi_\epsilon^{*2}\right)\nabla_\theta A_\epsilon{}^1\|_{a\sim 1,s\sim\infty}+\|\pi_\epsilon^{*2}\left(\nabla_\theta A_\epsilon{}^1-\nabla_\theta A_\epsilon{}^2\right)\|_{a\sim 1,s\sim\infty}\right).\end{aligned} \tag{69}$$

Plugging (65) and assumption (A.1) into (69), it then holds that

$$\begin{aligned}
\|\nabla_\theta l_\epsilon^1 - \nabla_\theta l_\epsilon^2\|_2 &\leq L_{F,\theta,1}\|\theta_1-\theta_2\| + 4\epsilon^2 L_{F,\pi,1}\|\nabla_\theta {A_\epsilon}^1\|_{2,\infty}\|{Q_\epsilon^*}^1 - {Q_\epsilon^*}^2\|_\infty \\
&\quad + \epsilon L_{F,\pi,0}\left(4\epsilon\|\nabla_\theta {A_\epsilon}^1\|_{2,\infty}\|{Q_\epsilon^*}^1 - {Q_\epsilon^*}^2\|_\infty + \|\nabla_\theta {A_\epsilon}^1 - \nabla_\theta {A_\epsilon}^2\|_{2,\infty}\right) \\
&\leq L_{F,\theta,1}\|\theta_1-\theta_2\| + 4\epsilon^2 (L_{F,\pi,1}+L_{F,\pi,0})\|\nabla_\theta {A_\epsilon}^1\|_{2,\infty}\|{Q_\epsilon^*}^1 - {Q_\epsilon^*}^2\|_\infty \\
&\quad + \epsilon L_{F,\pi,0}\left(4\epsilon\|\nabla_\theta {Q_\epsilon^*}^1\|_{2,\infty}\|{Q_\epsilon^*}^1 - {Q_\epsilon^*}^2\|_\infty + 2\|\nabla_\theta {Q_\epsilon^*}^1 - \nabla_\theta {Q_\epsilon^*}^2\|_{2,\infty}\right) \\
&\leq L_{F,\theta,1}\|\theta_1-\theta_2\|_2 + 4\epsilon^2 (2L_{F,\pi,1}+3L_{F,\pi,0})\|\nabla_\theta {Q_\epsilon^*}^1\|_{2,\infty}\|{Q_\epsilon^*}^1 - {Q_\epsilon^*}^2\|_\infty \\
&\quad + 2\epsilon L_{F,\pi,0}\|\nabla_\theta {Q_\epsilon^*}^1 - \nabla_\theta {Q_\epsilon^*}^2\|_{2,\infty}.
\end{aligned} \tag{70}$$

Here, the second inequality holds by plugging (68), and the third inequality follows from the fact that $\|\nabla_\theta {A_\epsilon}^1\|_{2,\infty} \leq 2\|\nabla_\theta {Q_\epsilon^*}^1\|_{2,\infty}$.

For θ during the update, we have the following inequality

$$\begin{aligned}
\|V_\epsilon^*\|_\infty &= \left\|\langle \pi_\epsilon^*(\cdot\,|\,s), Q_\epsilon^*(s,\cdot)\rangle_{\mathcal{A}} - \epsilon^{-1}\sum_{a\in\mathcal{A}}\Omega(\pi_\epsilon^*(a\,|\,s))\right\|_{s\sim\infty} \\
&\leq \|Q_\epsilon^*\|_\infty + \epsilon^{-1}U_\Omega.
\end{aligned} \tag{71}$$

Using the above results, we have

$$\begin{aligned}
\|Q_\epsilon^*(s,a)\|_\infty &= \|r(s,a) + \gamma\langle P(\cdot\,|\,s,a), V_\epsilon^*(\cdot)\rangle_{\mathcal{S}}\|_\infty \\
&\leq \|r\|_\infty + \gamma\,\|V_\epsilon^*\|_\infty \\
&\leq \|r\|_\infty + \gamma\cdot\left(\|Q_\epsilon^*\|_\infty + \epsilon^{-1}U_\Omega\right).
\end{aligned} \tag{72}$$

We can further derive that

$$\|Q_\epsilon^*\|_\infty \leq (1-\gamma)^{-1}\left(B_r + \gamma\epsilon^{-1}U_\Omega\right), \tag{73}$$

$$\|V_\epsilon^*\|_\infty \leq (1-\gamma)^{-1}\left(B_r + \epsilon^{-1}U_\Omega\right), \tag{74}$$

$$\begin{aligned}
\|\nabla_\theta Q_\epsilon^*\|_{2,\infty} &\leq (1-\gamma)^{-1}\,\|\nabla_\theta r(s,a) + \gamma\langle\nabla_\theta P(\cdot\,|\,s,a), V_\epsilon^*(\cdot)\rangle_{\mathcal{S}}\|_{2,\infty} \\
&\leq (1-\gamma)^{-1}\left(L_{r,\theta,0} + \gamma L_{P,\theta,0}(1-\gamma)^{-1}\left(B_r+\epsilon^{-1}U_\Omega\right)\right),
\end{aligned} \tag{75}$$

$$\|\nabla_\theta V_\epsilon^*\|_{2,\infty} \leq \|\nabla_\theta Q_\epsilon^*\|_{2,\infty}. \tag{76}$$

For θ_1, θ_2 during the update, it holds that

$$\|Q_\epsilon^*(\theta_1) - Q_\epsilon^*(\theta_2)\|_\infty \leq (1-\gamma)^{-1}\,\|r(\theta_1)-r(\theta_2)\|_\infty \leq (1-\gamma)^{-1}L_{r,\theta,0}\,\|\theta_1-\theta_2\|_2 = \alpha_1\,\|\theta_1-\theta_2\|_2, \tag{77}$$

and that

$$\begin{aligned}
\|\nabla_\theta Q_\epsilon^*(\theta_1) - \nabla_\theta Q_\epsilon^*(\theta_2)\|_{2,\infty} &\leq \|\nabla_\theta r(\theta_1) - \nabla_\theta r(\theta_2)\|_{2,\infty} + \gamma\|\nabla_\theta P(\theta_1) - \nabla_\theta P(\theta_2)\|_{\theta\sim 2, s'\sim 1,(s,a)\sim\infty}\|V_\epsilon^*(\theta_1)\|_\infty \\
&\quad + \gamma\,\|\nabla_\theta P(\theta_2)\|_{\theta\sim 2, s'\sim 1,(s,a)\sim\infty}\,\|Q_\epsilon^*(\theta_1) - Q_\epsilon^*(\theta_2)\|_\infty \\
&\quad + \gamma\,\|P(\theta_1) - P(\theta_2)\|_{s'\sim 1,(s,a)\sim\infty}\,\|\nabla_\theta V_\epsilon^*(\theta_1)\|_{2,\infty} \\
&\quad + \gamma\left(4\epsilon\|\nabla_\theta Q_\epsilon^*(\theta_1)\|_\infty\|Q_\epsilon^*(\theta_1)-Q_\epsilon^*(\theta_2)\|_\infty + \|\nabla_\theta Q_\epsilon^*(\theta_1) - \nabla_\theta Q_\epsilon^*(\theta_2)\|_{2,\infty}\right) \\
&\leq (1-\gamma)\alpha_2\,\|\theta_1-\theta_2\|_2 + \gamma\,\|\nabla_\theta Q_\epsilon^*(\theta_1) - \nabla_\theta Q_\epsilon^*(\theta_2)\|_{2,\infty},
\end{aligned} \tag{78}$$

which implies that

$$\|\nabla_\theta Q_\epsilon^*(\theta_1) - \nabla_\theta Q_\epsilon^*(\theta_2)\|_{2,\infty} \leq \alpha_2\,\|\theta_1-\theta_2\|_2. \tag{79}$$

Here, we remark that $\alpha_1 \sim O(1)$ and that $\alpha_2 \sim O(\epsilon)$. As θ updates from θ_t to θ_{t+1}, it holds that

$$\begin{aligned}
\left\|Q_\epsilon^*(\theta_{t+1}) - Q_\epsilon^0(\theta_{t+1})\right\|_\infty &\leq \left\|Q_\epsilon^*(\theta_t) - Q_\epsilon^K(\theta_t)\right\|_\infty + \|Q_\epsilon^*(\theta_t) - Q_\epsilon^*(\theta_{t+1})\|_\infty \\
&\leq \left\|Q_\epsilon^*(\theta_t) - Q_\epsilon^K(\theta_t)\right\|_\infty + \alpha_1\,\|\theta_{t+1}-\theta_t\|_2,
\end{aligned} \tag{80}$$

and that

$$\begin{aligned}\left\|\nabla_\theta Q^*_\epsilon(\theta_{t+1}) - \nabla_\theta Q^0_\epsilon(\theta_{t+1})\right\|_{2,\infty} &\le \left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^K_\epsilon(\theta_t)\right\|_{2,\infty} + \left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^*_\epsilon(\theta_{t+1})\right\|_{2,\infty} \\ &\le \left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^K_\epsilon(\theta_t)\right\|_{2,\infty} + \alpha_2 \left\|\theta_{t+1} - \theta_t\right\|_2 . \end{aligned} \tag{81}$$

Here, we note that

$$\eta^{-1}\left\|\theta_{t+1} - \theta_t\right\|_2 = \left\|\nabla_\theta l^K_\epsilon(\theta_t)\right\|_2 \le \left\|\nabla_\theta l^K_\epsilon(\theta_t) - \nabla_\theta l_\epsilon(\theta_t)\right\|_2 + \left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2 . \tag{82}$$

By (70), we have

$$\begin{aligned}\left\|\nabla_\theta l_\epsilon(\theta_t) - \nabla_\theta l^K_\epsilon(\theta_t)\right\|_2 &\le 4\epsilon^2(2L_{F,\pi,1} + 3L_{F,\pi,0})\|\nabla_\theta Q^*_\epsilon(\theta_t)\|_{\theta\sim 2,(s,a)\sim\infty}\left\|Q^*_\epsilon(\theta_t) - Q^K_\epsilon(\theta_t)\right\|_\infty \\ &\quad + 2\epsilon L_{F,\pi,0}\left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^K_\epsilon(\theta_t)\right\|_{2,\infty} \\ &= \beta_1\left\|Q^*_\epsilon(\theta_t) - Q^K_\epsilon(\theta_t)\right\|_\infty + \beta_2\left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^K_\epsilon(\theta_t)\right\|_{2,\infty} , \end{aligned} \tag{83}$$

where $\beta_1 \sim O(\epsilon^2)$ and $\beta_2 \sim O(\epsilon)$. We also have

$$\left\|\nabla_\theta l_\epsilon(\theta_1) - \nabla_\theta l_\epsilon(\theta_2)\right\|_2 \le (L_{F,\theta,1} + \beta_1\alpha_1 + \beta_2\alpha_2)\left\|\theta_1 - \theta_2\right\|_2 = L_{l,\theta,1}\left\|\theta_1 - \theta_2\right\|_2 , \tag{84}$$

where $L_{l,\theta,1} \sim O(\epsilon^2)$. We define

$$d_{k,t} = \begin{bmatrix}\left\|Q^*_\epsilon(\theta_t) - Q^k_\epsilon(\theta_t)\right\|_\infty \\ \left\|\nabla_\theta Q^*_\epsilon(\theta_t) - \nabla_\theta Q^k_\epsilon(\theta_t)\right\|_{2,\infty}\end{bmatrix}, \quad A_K = \gamma^K\begin{bmatrix}1 & 0 \\ C_0 K & 1\end{bmatrix}, \tag{85}$$

$$\alpha = \begin{bmatrix}\alpha_1 \\ \alpha_2\end{bmatrix}, \quad \beta = \begin{bmatrix}\beta_1 \\ \beta_2\end{bmatrix}, \quad \hat\beta = \begin{bmatrix}\beta_1^{-1} \\ \beta_2^{-1}\end{bmatrix}. \tag{86}$$

where $C_0 \sim O(\epsilon)$. Combining (80), (81), (82), and (83), we have

$$d_{0,t+1} \le d_{K,t} + \eta\alpha\left(\beta^\top d_{K,t} + \left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2\right) = \left(I + \eta\alpha\beta^\top\right)d_{K,t} + \eta\alpha\left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2 . \tag{87}$$

Moreover, we have

$$d_{K,t} \le A_K d_{0,t}. \tag{88}$$

Let $0 < \lambda < 1$. Let's consider the following two cases.

Case 1. $\beta^\top d_{K,t} > \lambda\left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2$. We have the following inequality

$$\left\|\nabla_\theta l^K_\epsilon(\theta_t)\right\|_2 \le \left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2 + \beta^\top d_{K,t} \le (1+\lambda^{-1})\beta^\top d_{K,t}. \tag{89}$$

Moreover, it holds that

$$\begin{aligned}\beta^\top d_{K,t+1} &\le \beta^\top A_K\left(\left(I + \eta\alpha\beta^\top\right)d_{K,t} + \eta\alpha\left\|\nabla_\theta l_\epsilon(\theta_t)\right\|_2\right) \\ &\le \beta^\top A_K\left(\left(I + \eta\alpha\beta^\top\right) + \eta\alpha\lambda^{-1}\beta^\top\right)d_{K,t} \\ &\le \beta^\top A_K\left(\hat\beta + (1+\lambda^{-1})\eta\alpha\right)\beta^\top d_{K,t} \\ &= \sigma_K\beta^\top d_{K,t}. \end{aligned} \tag{90}$$

Suppose that case 1 holds for $t = T_1, \cdots, T_2 - 1$. By (90), we have

$$\left\|\nabla_\theta l_\epsilon(\theta_{T_2})\right\|_2 \le \lambda^{-1}\beta^\top d_{K,T_2} \le \sigma_K^{T_2-T_1}\beta^\top d_{K,T_1}. \tag{91}$$

Moreover, it follows that

$$\begin{aligned}\left\|\sum_{t=T_1}^{T_2-1}\eta\nabla_\theta l^K_\epsilon(\theta_t)\right\|_2 &\le \eta\sum_{t=T_1}^{T_2-1}\left\|\nabla_\theta l^K_\epsilon(\theta_t)\right\|_2 \le \eta(1+\lambda^{-1})\beta^\top\sum_{t=T_1}^{T_2-1} d_{K,t} \\ &\le \eta(1+\lambda^{-1})\beta^\top d_{K,T_1}\frac{1-\sigma_K^{T_2-T_1}}{1-\sigma_K} \\ &\le \eta(1+\lambda^{-1})\beta^\top A_K d_{0,T_1}\cdot\frac{1}{1-\sigma_K}, \end{aligned} \tag{92}$$

which indicates that

$$\|\theta_{T_2}-\theta^*\|_2 \leq \|\theta_{T_1}-\theta^*\|_2 + \frac{\eta(1+\lambda^{-1})}{1-\sigma_K}\beta^\top A_K d_{0,T_1} = \Theta_{T_2}. \tag{93}$$

Combining (91) and (93), we have

$$l_\epsilon(\theta_{T_2}) - l_\epsilon(\theta^*) \leq \|\nabla_\theta l_\epsilon(\theta_{T_2})\|_2 \|\theta_{T_2}-\theta^*\|_2 \leq \sigma_K^{T_2-T_1}\beta^\top A_K d_{0,T_1}\Theta_{T_2}. \tag{94}$$

Such a result shows that the function value error descends exponentially in case 1.

Case 2. $\beta^\top d_{K,t} \leq \lambda \|\nabla_\theta l_\epsilon(\theta_t)\|_2$. We have the following inequalities

$$\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2 \leq \|\nabla_\theta l_\epsilon(\theta_t)\|_2 + \beta^\top d_{K,t} \leq (1+\lambda)\|\nabla_\theta l_\epsilon(\theta_t)\|_2, \tag{95}$$
$$\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2 \geq \|\nabla_\theta l_\epsilon(\theta_t)\|_2 - \beta^\top d_{K,t} \geq (1-\lambda)\|\nabla_\theta l_\epsilon(\theta_t)\|_2. \tag{96}$$

Moreover, it holds that

$$\begin{aligned}
\beta^\top d_{K,t+1} &\leq \beta^\top A_K \left(\left(I+\eta\alpha\beta^\top\right) d_{K,t} + \eta\alpha \|\nabla_\theta l_\epsilon(\theta_t)\|_2\right) \\
&\leq \beta^\top A_K \left(\left(\hat\beta+\eta\alpha\right)\beta^\top d_{K,t} + \eta\alpha\|\nabla_\theta l_\epsilon(\theta_t)\|_2\right) \\
&\leq \lambda\beta^\top A_K\left(\hat\beta + (1+\lambda^{-1})\eta\alpha\right)\|\nabla_\theta l_\epsilon(\theta_t)\|_2 \\
&= \lambda\sigma_K \|\nabla_\theta l_\epsilon(\theta_t)\|_2
\end{aligned} \tag{97}$$

Note that

$$\begin{aligned}
\|\nabla_\theta l_\epsilon(\theta_t)\|_2 &= \left\|\nabla_\theta l_\epsilon(\theta_{t+1}+\eta\nabla_\theta l_\epsilon^K(\theta_t))\right\|_2 \\
&\leq \|\nabla_\theta l_\epsilon(\theta_{t+1})\|_2 + \eta L_{l,\theta,1}\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2 \\
&\leq \|\nabla_\theta l_\epsilon(\theta_{t+1})\|_2 + \eta L_{l,\theta,1}(1+\lambda)\|\nabla_\theta l_\epsilon(\theta_t)\|_2.
\end{aligned} \tag{98}$$

Hence, it follows that

$$\begin{aligned}
\lambda\|\nabla_\theta l_\epsilon(\theta_{t+1})\|_2 &\geq \lambda\left(1-\eta L_{l,\theta,1}(1+\lambda)\right)\|\nabla_\theta l_\epsilon(\theta_t)\|_2 \\
&\geq \left(1-\eta L_{l,\theta,1}(1+\lambda)\right)\sigma_K^{-1}\beta^\top d_{K,t+1} \\
&\geq \beta^\top d_{K,t+1},
\end{aligned} \tag{99}$$

where the last inequality holds by condition 1 (see (63)). Here (99) indicates that case 1 will automatically hold for $t+1, t+2, \cdots$. Moreover, we have

$$\begin{aligned}
l_\epsilon(\theta_{t+1}) &\leq l_\epsilon(\theta_t) - \eta\langle\nabla_\theta l_\epsilon(\theta_t), \nabla_\theta l_\epsilon^K(\theta_t)\rangle + \frac{\eta^2 L_{l,\theta,1}}{2}\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2^2 \\
&\leq l_\epsilon(\theta_t) - \eta\left(1-\frac{\eta L_{l,\theta,1}}{2}\right)\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2^2 + \eta\langle\nabla_\theta l_\epsilon^K(\theta_t)-\nabla_\theta l_\epsilon(\theta_t), \nabla_\theta l_\epsilon^K(\theta_t)\rangle \\
&\leq l_\epsilon(\theta_t) - \eta\left(\frac{1-2\lambda}{1-\lambda} - \frac{\eta L_{l,\theta,1}}{2}\right)\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2^2 \\
&= l_\epsilon(\theta_t) - \eta w\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2^2,
\end{aligned} \tag{100}$$

where the last second inequality holds by noting that

$$\left\|\nabla_\theta l_\epsilon^K(\theta_t) - \nabla_\theta l_\epsilon(\theta_t)\right\|_2 \leq \beta^\top d_{K,t} \leq \lambda\|\nabla_\theta l_\epsilon(\theta_t)\|_2 \leq \frac{\lambda}{1-\lambda}\left\|\nabla_\theta l_\epsilon^K(\theta_t)\right\|_2. \tag{101}$$

By Condition 2, $l_\epsilon(\theta_t)$ will decent thereafter for $t+1, t+2, \cdots$.

Convergence Result. By the above discussion, it is easy to show that if case 1 happens, it only holds for $t = 0, 1, \cdots, \tau-1$, where τ is a positive integer. After case 1 is ended, case 2 will follow thereafter for $t = \tau, \tau+1, \cdots$. Hence, we can split the updates into two stage, i.e., the gradient convergence stage which holds for $t = 0, 1, \cdots, \tau-1$ where case 1 holds and the value convergence stage which holds for $t = \tau, \tau+1, \cdots$ where case 2 holds.

Gradient Convergence Stage. Following (94), we have for $t = \tau$ that

$$l_\epsilon(\theta_\tau) - l_\epsilon(\theta^*) \le \sigma_K^\tau \beta^\top A_K d_{0,0} \Theta = \mathcal{L}_\tau, \tag{102}$$

where

$$\Theta = \left(\|\theta_0 - \theta^*\|_2 + \frac{\eta(1+\lambda^{-1})}{1-\sigma_K} \beta^\top A_K d_{0,0} \right), \tag{103}$$

and it also holds that

$$\|\theta_\tau - \theta^*\|_2 \le \Theta. \tag{104}$$

Value Convergence Stage. By (100), we can verify that $l_\epsilon(\theta_\tau) \ge l_\epsilon(\theta_{\tau+1}) \ge \cdots$. Hence, by Assumption A.2, it holds that $\|\theta_t - \theta^*\|_2 \le D_{\mathcal{L}_\tau}$ for $t = \tau, \tau+1, \cdots$. For $t = \tau, \tau+1, \cdots$, note that

$$\begin{aligned}
& l_\epsilon(\theta_{t+1}) - (1-2w) l_\epsilon(\theta_t) \\
& \le 2w l_\epsilon(\theta_t) - \eta w \left\| \nabla_\theta l_\epsilon^K(\theta_t) \right\|_2^2 \\
& \le 2w \left(l_\epsilon(\theta^*) + \langle \nabla_\theta l_\epsilon(\theta_t), \theta_t - \theta^* \rangle \right) - \eta w \left\| \nabla_\theta l_\epsilon^K(\theta_t) \right\|_2^2 \\
& = 2w l_\epsilon(\theta^*) - w\eta^{-1} \left(\eta^2 \left\| \nabla_\theta l_\epsilon^K(\theta_t) \right\|_2^2 - 2\eta \langle \nabla_\theta l_\epsilon^K(\theta_t), \theta_t - \theta^* \rangle + \|\theta_t - \theta^*\|_2^2 \right) \\
& \quad + w\eta^{-1} \|\theta_t - \theta^*\|_2^2 + 2w \left\langle \nabla_\theta l_\epsilon(\theta_t) - \nabla_\theta l_\epsilon^K(\theta_t), \theta_t - \theta^* \right\rangle \\
& \le 2w l_\epsilon(\theta^*) + w\eta^{-1} \|\theta_t - \theta^*\|_2^2 - w\eta^{-1} \|\theta_{t+1} - \theta^*\|_2^2 + \frac{2w\lambda D_{\mathcal{L}_\tau}}{1-\lambda} \left\| \nabla_\theta l_\epsilon^K(\theta_t) \right\|_2 .
\end{aligned} \tag{105}$$

Taking the inequality for $t = \tau, \tau+1, \cdots, T-1$ and summing them up give

$$\begin{aligned}
& 2w \sum_{t=\tau+1}^{T} l_\epsilon(\theta_t) + (1-2w) \left(l_\epsilon(\theta_T) - l_\epsilon(\theta_\tau) \right) \\
& \le 2w(T-\tau) l_\epsilon(\theta^*) - w\eta^{-1} \left(\|\theta_T - \theta^*\|_2^2 - \|\theta_\tau - \theta^*\|_2^2 \right) + \frac{2w\lambda D_{\mathcal{L}_\tau}}{1-\lambda} \sum_{t=\tau}^{T-1} \left\| \nabla_\theta l_\epsilon^K(\theta_t) \right\|_2 \\
& \le 2w(T-\tau) l_\epsilon(\theta^*) + w\eta^{-1} \|\theta_\tau - \theta^*\|_2^2 + \frac{2w\lambda D_{\mathcal{L}_\tau}}{1-\lambda} \sqrt{(T-\tau) \sum_{t=\tau}^{T} \|\nabla_\theta l_\epsilon^K(\theta_t)\|_2^2} \\
& \le 2w(T-\tau) l_\epsilon(\theta^*) + w\eta^{-1} \|\theta_\tau - \theta^*\|_2^2 + \frac{2w\lambda D_{\mathcal{L}_\tau}}{1-\lambda} \sqrt{(T-\tau) \cdot \frac{l_\epsilon(\theta_\tau) - l_\epsilon(\theta^*)}{\eta w}}.
\end{aligned} \tag{106}$$

Rearranging the inequality gives

$$\begin{aligned}
\frac{1}{T-\tau} \sum_{t=\tau+1}^{T} \left(l_\epsilon(\theta_t) - l_\epsilon(\theta^*) \right) \le \frac{1}{T-\tau} \cdot \left(\frac{\eta^{-1}}{2} \|\theta_\tau - \theta^*\|_2^2 + \frac{1-2\omega}{2\omega} \left(l_\epsilon(\theta_\tau) - l_\epsilon(\theta^*) \right) \right) \\
+ \frac{\lambda D_{\mathcal{L}_\tau}}{1-\lambda} \cdot \sqrt{\frac{l_\epsilon(\theta_\tau) - l_\epsilon(\theta^*)}{\eta\omega(T-\tau)}}.
\end{aligned} \tag{107}$$

Since $l_\epsilon(\theta_t)$ decreases at this stage, it follows that

$$\begin{aligned}
&l_\epsilon(\theta_T) - l_\epsilon(\theta^*) \\
&\leq \min\left\{\frac{1}{T-\tau}\left(\frac{\eta^{-1}}{2}\Theta^2 + \frac{1-2\omega}{2\omega}\mathcal{L}_\tau\right) + \frac{\lambda D_{\mathcal{L}_\tau}}{1-\lambda}\sqrt{\frac{\mathcal{L}_\tau}{\eta\omega(T-\tau)}}, \mathcal{L}_\tau\right\} \\
&\leq \max_{0\leq\tau\leq T}\min\left\{\frac{1}{T-\tau}\left(\frac{\eta^{-1}}{2}\Theta^2 + \frac{1-2\omega}{2\omega}\mathcal{L}_\tau\right) + \frac{\lambda D_{\mathcal{L}_\tau}}{1-\lambda}\sqrt{\frac{\mathcal{L}_\tau}{\eta\omega(T-\tau)}}, \mathcal{L}_\tau\right\}. \qquad (108)
\end{aligned}$$

For a given τ, the first term in the right hand side of (108) diminishes at a rate of $O(T^{-1/2})$. However, $\mathcal{L}_\tau$ diminishes at a rate of $O(\sigma_K^\tau)$ where $\sigma_K < 1$ by our condition. Hence, for sufficiently large T, the maximum is reached when $\tau << T$, which means that the first term is dominant as $T \to \infty$. Therefore, we have that the convergence rate of the design objective function is at least of $O(T^{-1/2})$, which completes the proof of Theorem 5.4.

B. Technical Results

Lemma B.1 (Projection)**.** *For any $\Delta_r > 0, \Delta_\pi > 0$, if it holds that*

$$\varepsilon > \Delta_r^{-1}\left((1+\gamma)\left(\dot\Omega(1) - \dot\Omega\left(\frac{\Delta_\pi}{2|\mathcal{A}|}\right)\right) + \gamma U_\Omega\right), \qquad (109)$$

then for any $r_\varepsilon : \mathcal{S}\times\mathcal{A}\times\mathcal{X} \to \mathbb{R}$, there exists $r : \mathcal{S}\times\mathcal{A}\times\mathcal{X} \to \mathbb{R}$ and $\pi^* \in \Pi^*(\mathcal{S}, \mathcal{A}, \gamma, P, r)$ satisfying

$$\|r - r_\varepsilon\|_{(s,a,\theta)\sim\infty} < \Delta_r. \qquad (110)$$

and

$$\|\pi^*(a \mid s) - \pi_\varepsilon^*(a \mid s)\|_{a\sim 1, s\sim\infty} < \Delta_\pi. \qquad (111)$$

Proof. Lemma B.1 states that with sufficiently large ϵ, for any reward and regularized optimal policy pair $(r_\varepsilon, \pi_\epsilon^*)$, there exists another reward r and corresponding exact optimal policy $\pi^* \in \Pi^*(\mathcal{S}, \mathcal{A}, \gamma, P, r)$ that are close enough to $(r_\varepsilon, \pi_\epsilon^*)$. Such a property can be viewed as the projection of regularized reward-policy pair onto the exact reward-policy pair. We give a proof by construction. Let $\Delta_Q = (1+\gamma)^{-1}\left(\Delta_r - \gamma\varepsilon^{-1}U_\Omega\right)$ and $\delta = \Delta_\pi/\left(2|\mathcal{A}|\right)$, it holds that $\Delta_Q > \varepsilon^{-1}(\dot\Omega(1) - \dot\Omega(\delta))$. For simplicity, let $Q_\varepsilon = Q_\varepsilon^*(\cdot; r_\varepsilon), \pi_\varepsilon = \pi_\varepsilon^*(\cdot; r_\varepsilon)$. We give a proof by construction. For a given state s, we simplify our denotion by $Q(i) = Q(s, a_i)$ and $\pi_\varepsilon(i) = \pi_\epsilon(a_i \mid s)$.

Let $k = \underset{i, \pi_\varepsilon(i)\geq\delta}{\operatorname{argmin}}\, \pi_\varepsilon(i)$ and $\mathcal{B} = \{i; \pi_\varepsilon(i) < \delta\}$, we first construct Q^* and π^* by

$$Q^*(i) = \begin{cases} Q_\varepsilon(k), & \pi_\varepsilon(i) \geq \delta, \\ Q_\varepsilon(i), & \pi_\varepsilon(i) < \delta. \end{cases}$$

$$\pi^*(i) = \begin{cases} \pi_\varepsilon(i) + \frac{1}{|\mathcal{B}|}\sum_{j\in\mathcal{B}}\pi_\varepsilon(j), & \pi_\varepsilon(i) \geq \delta, \\ 0, & \pi_\varepsilon(i) < \delta. \end{cases}$$

Following the π^* defined above, it holds that

$$\begin{aligned}
\|\pi^* - \pi_\varepsilon\|_{a\sim 1, s\sim\infty} &= \sup_s\left(\frac{|\mathcal{A}| - |\mathcal{B}|}{|\mathcal{B}|}\sum_{j\in\mathcal{B}}\pi_\varepsilon(j) + \sum_{i\in\mathcal{B}}\pi_\varepsilon(i)\right) \\
&= \sup_s\left(\frac{|\mathcal{A}|}{|\mathcal{B}|}\sum_{j\in\mathcal{B}}\pi_\varepsilon(j)\right) \leq |\mathcal{A}|\delta < \Delta_\pi.
\end{aligned}$$

For any i such that $\pi_\varepsilon(i) \geq \delta$, we have

$$\begin{aligned}
|Q_\varepsilon(i) - Q_\varepsilon(k)| &= \varepsilon^{-1}\left(\dot\Omega\left(\pi_\varepsilon(i)\right) - \dot\Omega\left(\pi_\varepsilon(k)\right)\right) \\
&\leq \varepsilon^{-1}(\dot\Omega(1) - \dot\Omega(\delta)) \\
&\leq \Delta_Q.
\end{aligned}$$

The first equality holds by the property of Legendre transformation $Q_\epsilon(i) = \dot{\Omega}(\pi_\epsilon(i))$. Hence, we conclude that $\|Q_\varepsilon - Q^*\|_{(s,a)\sim\infty} \leq \Delta_Q$. By now, it remains to see whether $\|r_\varepsilon - r^*\|_{(s,a)\sim\infty} < \Delta_r$. Here, by the Bellman equation, we have

$$\begin{aligned}
\|r(s,a) - r_\varepsilon(s,a)\|_{(s,a)\sim\infty} \leq & \|Q^*(s,a;r) - Q_\varepsilon\left(s,a;r_\varepsilon\right)\|_{(s,a)\sim\infty} \\
& + \gamma \cdot \| \left\langle P(\cdot \mid s,a), |V^*(\cdot;r) - V_\varepsilon\left(\cdot;r_\varepsilon\right)|\right\rangle_{\mathcal{S}'} \|_{(s,a)\sim\infty} \\
\leq & \|Q^*(s,a;r) - Q_\varepsilon\left(s,a;r_\varepsilon\right)\|_{(s,a)\sim\infty} + \gamma \cdot \|V^*(s;r) - V_\varepsilon(s;r)\|_{s\sim\infty} \\
\leq & (1+\gamma)\|Q^*(s,a;r) - Q_\varepsilon\left(s,a;r_\varepsilon\right)\|_{(s,a)\sim\infty} + \gamma\varepsilon^{-1}U_\Omega \\
< & \Delta r.
\end{aligned}$$

Here, the first inequality can be derived from the Cauchy–Schwartz inequality. The second inequality holds since $\langle P(\cdot \mid s,a), |V^*(\cdot;r) - V_\varepsilon\left(\cdot;r_\varepsilon\right)|\rangle_{\mathcal{S}'}$ is bounded by $\|V^*(s;r) - V_\varepsilon(s;r)\|_{s\sim\infty}$. The third inequality follows from the fact that

$$\begin{aligned}
\|V^*(s;r) - V_\varepsilon(s;r)\|_{s\sim\infty} &= \left\| \max_\pi \langle \pi, Q^* \rangle_{\mathcal{A}} - \max_\pi \left(\langle \pi, Q_\epsilon \rangle_{\mathcal{A}} - \epsilon^{-1} \sum_{a\in\mathcal{A}} \Omega\big(\pi(a \,|\, s)\big) \right) \right\|_{s\sim\infty} \\
&\leq \left\| \langle \pi', Q^* - Q_\epsilon \rangle + \epsilon^{-1} \sum_{a\in\mathcal{A}} \Omega(\pi(a \,|\, s)) \right\|_{s\sim\infty} \\
&\leq \|Q^*(s,a;r) - Q_\varepsilon\left(s,a;r_\varepsilon\right)\|_{(s,a)\sim\infty} + \varepsilon^{-1}U_\Omega,
\end{aligned} \tag{112}$$

where $\pi'(\cdot \,|\, s)$ is the optimizer for the smaller one between $V^*(s)$ and $V_\epsilon(s)$ for any $s \in \mathcal{S}$ and $U_\Omega = \max_\pi \sum_a \Omega(\pi(a))$. Since we have proved that $\|Q_\varepsilon - Q^*\|_\infty < \Delta_Q$, by the definition of Δ_Q, it follows that $\|r_\varepsilon - r^*\|_\infty < \Delta r$. Thus we complete the proof of Lemma B.1. $\square$

C. Sample-based algorithm

Algorithm 3 Sample-based Algorithm for the RMD (14) with $\Omega(x) = x \ln x$

Input: outer iterations T, inner iterations K, learning rate η, the gradient of pre-learned transition model $\nabla_\theta \ln P$ and the gradient of the reward function $\nabla_\theta r$ with respect to θ.
Initialize θ_0, Q_ϵ^0, and $\nabla_{\theta_0} Q_\epsilon^0$
for $t = 0$ **to** $T-1$ **do**
  Initialize replay memory $\mathcal{D}$ to capacity N
  Independently sample $(\hat{s}_1, \cdots, \hat{s}_L) \sim \rho$ over $\mathcal{S}$.
  for episode $k = 0$ **to** $K-1$ **do**
    Initialize $s_1 = \{x_i | x_i \in \rho, i = 1, \cdots, M\}$
    for time step $i = 0$ **to** $\lfloor N/M \rfloor$ **do**
      $\pi_\epsilon^k(a|s) \propto \exp\left(\epsilon Q_\epsilon^k(s,a)\right)$
      Select $a_i \sim \pi_\epsilon^k(\cdot \mid s_i)$, obtain r_i and $\nabla_{\theta_t} r_i$, and observe the next state s_{i+1}
      Store transition $(s_i, a_i, r_i, \nabla_{\theta_t} r_i, s_{i+1})$ in $\mathcal{D}$
      Sample random minibatch of transitions $(s_j, a_j, r_j, \nabla_{\theta_t} r_j, s_{j+1})$ from $\mathcal{D}$
      $V_\epsilon^k(s_{j+1}) \leftarrow \epsilon^{-1} \ln\left(\sum_a \exp\left(\epsilon Q_\epsilon^k(s_{j+1}, a)\right)\right)$
      $\nabla_{\theta_t} {V_\epsilon}^k(s_{j+1}) \leftarrow \langle {\pi_\epsilon}^k, \nabla_{\theta_t} Q_\epsilon^k \rangle_{\mathcal{A}}$
      $y_j = r_j + \gamma {V_\epsilon}^k(s_{j+1})$
      $z_j = \nabla_{\theta_t} r_i + \gamma(\nabla_{\theta_t} V_\epsilon^k(s_{j+1}) + V_\epsilon^k(s_{j+1}) \nabla_{\theta_t} \ln P(s_{j+1}|s_j, a_j))$
      Perform a gradient descent step on $(y_j - Q_\epsilon^k(s_j, a_j; \theta_1))^2$ and $(z_j - \nabla_{\theta_t} Q_\epsilon^k(s_j, a_j; \theta_2))^2$, and obtain updated Q_ϵ^{k+1} and $\nabla_{\theta_t} Q_\epsilon^{k+1}$
    end for
  end for
  $\nabla_{\theta_t} A_\epsilon^K(s,a) = \nabla_{\theta_t} Q_\epsilon^K(s,a) - \nabla_{\theta_t} {V_\epsilon}^K(s)$
  $\nabla_{\theta_t} F = \frac{\partial F}{\partial \theta_t} + \epsilon / L \cdot \sum_{i=1}^{L} \sum_a \left(\rho^{-1}(\hat{s}_i) \cdot \partial F / \partial \pi_\epsilon^K(\hat{s}_i, a) \cdot \pi_\epsilon^K(\hat{s}_i, a) \cdot \nabla_{\theta_t} A_\epsilon^K(\hat{s}_i, a)\right)$
  $\theta_{t+1} = \theta_t + \eta \nabla_{\theta_t} F$
  Reinitialize $Q_\epsilon^0 = Q_\epsilon^K$ and $\nabla_{\theta_{t+1}} Q_\epsilon^0 = \nabla_{\theta_{t+1}} Q_\epsilon^K$
end for
Output: Optimized parameter θ_T and its corresponding upper-level objective $F(\theta_T, \pi_\epsilon^K)$

Following the idea of (Mnih et al., 2013), we use the function approximation method to approximate Q_ϵ and ∇Q_ϵ. We utilize the experience replay buffer to store the experience we collect and at each step, we sample a mini-batch from the buffer to train Q_ϵ and $\nabla_{\theta_t} Q_\epsilon$. We only use one replay buffer for the reason that the Bellman Operator for updating Q_ϵ and ∇Q_ϵ shares the same transition kernel and policy.

D. Additional Details of Experiments

D.1. Taxation Design for Macroeconomic Model

State Space & Action Space. The state is the accumulative asset s_t, which is a scalar ranging from $[-100, 100]$. The accumulative asset must be in the range, so there is a truncation operation in the transition kernel. In this experiment, we define 3 categories of goods, so the action space $\mathcal{A}$ is a 4-dimensional discrete space, the shape of which is $10 \times 5 \times 5 \times 5$. A point (i, j, k, l) in this discrete space represents the working hours $n = 8i/9 - 8/9$ and the consumption $c = (1.225j - 1.125, 1.225k - 1.125, 1.225l - 1.125)$ for each kinds of goods respectively.

Other Configurations. The learning rate η is 0.001. The initial asset for the agent follows a Gaussian distribution with mean 0 and variance 2. The initial taxation is set to $(0.4, 0.4, 0.4, 0.4)$. The discounted factor γ_1 and γ_2 are both set to 0.8.

Table 1. Design parameters (income tax rate and tax rates for good $1 \sim 3$) at convergence with different settings of ϵ for the taxation design experiment.

ϵ	INCOME	GOOD 1	GOOD 2	GOOD 3
1	1.2 %	9.17 %	9.08 %	8.92 %
5	2.23 %	8.55 %	8.39 %	8.1 %
30	2.27 %	8.2 %	8.08 %	7.85 %
50	37.51 %	38.64 %	39.11 %	41.2 %
ADAPTIVE	2.43 %	8.12 %	7.99 %	7.75 %

D.2. Workbench Position Design for A Two-ankle Robot Arm

State Space & Action Space. The state space $\mathcal{S}$ is a 4-dimensional discrete space, the shape of which is $100\times100\times9\times9$. A point (i, j, k, l) in this discrete space represents the first ankle's angle $\theta_1 = 2\pi i/100$, the second ankle's angle $\theta_2 = 2\pi j/100$, the first ankle's angular velocity $\omega_1 = k - 1$ and the second ankle's angular velocity $\omega_2 = l - 1$ respectively. The angular velocity must be in the discrete space, so there is a quantification operation in the transition kernel. The action space $\mathcal{A}$ is a 2-dimensional discrete space, the shape of which is 3×3. A point (i, j) in this discrete space represents the first ankle's angular acceleration $a_1 = i - 1$ and the second ankle's angular acceleration $a_2 = j - 1$ respectively.

Reward. At every time step t, the position of the end of the robot arm, whose angles of ankles are $\theta_t = (\theta_{1,t}, \theta_{2,t})$, is defined as follows

$$\begin{aligned} x_{end} &= \cos\theta_{1,t} + \cos\theta_{2,t} \\ y_{end} &= \sin\theta_{1,t} + \sin\theta_{2,t}. \end{aligned} \tag{113}$$

The reward r_t for the control of the robotic arm is defined as follows

$$r_t = -10 \cdot \left((x_{end} - x)^2 + (y_{end} - y)^2\right) - 0.5 \left\|\omega_t\right\|_2 . \tag{114}$$

Here, the reward r_t is coupled with angular velocity l_2-norm, since it effectively reduces the robotic arm's oscillation when it reaches the optimal state. It's obvious that r_t is parameterized by workbench position $p = (x, y)$

The reward r^u_t for the designer is defined as

$$r^u_t = -c_t - 0.1 \left\|\omega_t\right\|_2 , \tag{115}$$

which represents the energy consumption of robotic arm movement for each time step. Thus the upper-level objective for the designer is defined as follows

$$F = \mathbb{E}^{\pi}\left[\sum_t {\gamma_u}^t r^u_t\right] - 0.25 \left\|p - p_0\right\|_2^2 , \tag{116}$$

Where the initial workbench's position $p_0 = (1, -1)$. The first term on the right of the above equation is the discounted cumulative energy consumption given the robotic arm's control policy π, and the second term is an extra cost for setting up a workbench at the position p.

Other Configurations. The learning rate η is 0.01 . The inner iterations K is 100. $\gamma = 0.8$ is the discount factor for robotic arm's control, and $\gamma_u = 0.8$ is the discount factor for calculating the discounted cumulative energy consumption. The initial workbench's position p_0 is at $(1, -1)$. There are two goods, and their respective positions are set to $(0, 0)$ and $(1.872, 0.681)$ respectively.

Table 2. Design parameters (workbench position) at convergence for different settings of ϵ for the workbench position design experiment.

ϵ	x	y
1	2.70	0.76
5	1.88	-0.46
20	1.38	-1.31
ADAPTIVE	1.91	-0.26