Journal of Machine Learning Research 26 (2025) 1-49 Submitted 5/24; Published 2/25

# Principled Penalty-based Methods for Bilevel Reinforcement Learning and RLHF

**Han Shen** SHENHANHS@GMAIL.COM
*Department of Electrical, Computer and System Engineering*
*Rensselaer Polytechnic Institute*
*Troy, NY 12180, USA*

**Zhuoran Yang** ZHUORAN.YANG@YALE.EDU
*Department of Statistics and Data Science*
*Yale University*
*New Haven, CT 06520, USA*

**Tianyi Chen** CHENTIANYI19@GMAIL.COM
*Department of Electrical, Computer and System Engineering*
*Rensselaer Polytechnic Institute*
*Troy, NY 12180, USA*

**Editor:** Tor Lattimore

## Abstract

Bilevel optimization has been recently applied to many machine learning tasks. However, their applications have been restricted to the supervised learning setting, where static objective functions with benign structures are considered. But bilevel problems such as incentive design, inverse reinforcement learning (RL), and RL from human feedback (RLHF) are often modeled as dynamic objective functions that go beyond the simple static objective structures, which pose significant challenges of using existing bilevel solutions. To tackle this new class of bilevel problems, we introduce the first principled algorithmic framework for solving bilevel RL problems through the lens of penalty formulation. We provide theoretical studies of the problem landscape and its penalty-based (policy) gradient algorithms. We demonstrate the effectiveness of our algorithms via simulations in the Stackelberg Markov game, RL from human feedback and incentive design.

**Keywords:** bilevel optimization, reinforcement learning, Stackelberg games

## 1 Introduction

Bilevel optimization (BLO) has emerged as an effective framework in machine learning. In a nutshell, BLO involves two coupled optimization problems in the upper and lower levels respectively, where they have different decision variables, denoted by $x$ and $y$ respectively. The lower-level problem is a constraint for the upper-level problem, e.g., in the upper level, we minimize a function $f(x, y)$ with the constraint that $y$ is a solution to the lower-level

Symbol † denotes equal contribution. Preliminary results in this paper were presented in part at the 2024 International Conference on Machine Learning (Shen et al., 2024).

problem determined by $x$, i.e., $y \in \mathcal{Y}^*(x)$. Here $\mathcal{Y}^*(x)$ is the solution set of the lower-level problem determined by $x$.

BLO enjoys a wide range of applications in machine learning, including hyper-parameter optimization (Franceschi et al., 2018), meta-learning (Finn et al., 2017; Rajeswaran et al., 2019; Chen et al., 2023a), continue learning (Borsos et al., 2020), and adversarial learning (Jiang et al., 2021). Existing applications mostly concentrate on supervised learning settings, thus research on BLO has been predominantly confined to the static optimization setting (Franceschi et al., 2017), wherein both the upper and lower-level problems, the objective functions are (strongly-)convex functions. However, this setting is insufficient to model more complex game-theoretic behaviors with sequential decision-making.

Reinforcement learning (RL) (Sutton and Barto, 2018) is a principled framework for sequential decision-making problems and has achieved tremendous empirical success in recent years (Silver et al., 2017; Ouyang et al., 2022). In this work, we study the BLO problem in the context of RL, where the lower-level problem is an RL problem and the upper-level problem can be either smooth optimization or RL. Specifically, in the lower-level problem, the follower solves a Markov decision process (MDP) determined by the leader's decision $x$, and returns an optimal policy of this MDP to the leader, known as the best response policy. The leader aims to maximize its objective function, subject to the constraint that the follower always adopts the best response policy. This formulation of bilevel RL encompasses a range of applications such as Stackelberg Markov games (Stackelberg, 1952), reward learning (Hu et al., 2020), and RL from human feedback (RLHF) (Christiano et al., 2017). As an example, in RL from human feedback, the leader designs a reward $r_x$ for the follower's MDP, with the goal that the resulting optimal policy yields the desired behavior of the leader.

Despite its various applications, the bilevel RL problem is difficult to solve. Broadly speaking, the main technical challenge lies in handling the constraint, i.e., the lower-level problem. The lower-level problem of bilevel RL extends from static smooth optimization to policy optimization in RL, and thus faces significant technical challenges. Such an extension loses a few optimization structures, such as strong convexity and uniform Polyak-Łojasiewicz (PL) condition, which is critical for existing BLO algorithms (Ghadimi and Wang, 2018; Shen and Chen, 2023). Specifically, there are *two mainstream approaches* for BLO: (a) implicit gradient or iterative differentiation methods; and, (b) penalty-based methods. In (a), it is typically assumed that the lower-level objective function is strongly convex (Ji et al., 2021b; Chen et al., 2021), and thus its optimal solution $\mathcal{Y}^*(x)$ is unique. Then methods in (a) are essentially gradient-descent methods for the hyperobjective $f(x, \mathcal{Y}^*(x))$, where the gradient of $\mathcal{Y}^*(x)$ can be computed using the implicit function theorem. However, in our bilevel RL case, the lower-level objective function is the discounted return in MDP, which is known to be non-convex (Agarwal et al., 2020). Thus, the hyperobjective and its gradient are not well-defined. In (b), the bilevel problem is reformulated as a single-level problem by adding a penalty term of the lower-level sub-optimality to the leader's objective function. The penalty reformulation approach has been studied in (Ye, 2012; Shen and Chen, 2023; Ye et al., 2022; Kwon et al., 2023) under the assumption that the lower-level objective function satisfies certain PL conditions. Unfortunately, when it comes to bilevel RL, the lower-level discounted return objective does not satisfy these conditions. To develop the penalty approach for bilevel RL problems, it is unclear (i) what is an appropriate penalty function; (ii) how is the solution to the reformulated problem related to the original bilevel

problem; and, (iii) how to solve the reformulated problem. Therefore, directly extending applying BLO methods to bilevel RL is not straightforward, and new theories and algorithms tailored to the RL lower-level problem are needed, which are the subject of the paper.

### 1.1 Our contributions

To this end, we propose a novel algorithm that extends the idea of penalty-based BLO algorithm (Shen and Chen, 2023) to tackle the specific challenges of bilevel RL. Our approach includes the design of two tailored penalty functions: *value penalty and Bellman penalty*, which are crafted to capture the optimality condition of the lower-level RL problem. In addition, leveraging the geometry of the policy optimization problem, we prove that an approximate solution to our reformulated problem is also an effective solution to the original bilevel problem. Furthermore, we establish the differentiability of the reformulated problem, and propose a first-order policy-gradient-based algorithm. To our knowledge, we establish the first provably convergent first-order algorithm for bilevel RL. Lastly, we conduct experiments on example applications covered by our framework, including the Stackelberg game and RL from human feedback tasks.

Table 1: Summary of main convergence theorems. †: the lower-bound of the penalty constant $\lambda$ to guarantee that certain lower-level optimality gap (value function gap, Bellman gap, and NI function value) is smaller than accuracy $\delta_{value}$, $\delta_{bellman}$ and $\delta_{NI}$ respectively. These optimality gaps will be introduced in their respective sections; ‡: Here $\epsilon$ is the accuracy.

| | Section 3 & Section 4 | Section 5 |
|---|---|---|
| Lower-level problem | single-agent RL | two-player zero-sum |
| Upper-level problem | general objective | |
| Penalty functions | Value or Bellman penalty | Nikaido-Isoda (NI) |
| Penalty constant $\lambda$ | $\Omega(\delta_{value}^{-1})^\dagger$ or $\Omega(\delta_{bellman}^{-0.5})^\dagger$ | $\Omega(\delta_{NI}^{-1})^\dagger$ |
| Inner-loop oracle algorithm | Policy mirror descent (PMD) | |
| Iteration complexity | $\mathcal{O}(\lambda\epsilon^{-1}\log(\lambda^2/\epsilon))^\ddagger$ | |

### 1.2 Related works

**Bilevel optimization.** The bilevel optimization problem can be dated back to (Stackelberg, 1952). The gradient-based bilevel optimization methods have gained growing popularity in the machine learning area; see, e.g., (Sabach and Shtern, 2017; Franceschi et al., 2018; Liu et al., 2020). A prominent branch of gradient-based bilevel optimization is based on the implicit gradient (IG) theorem. The IG based methods have been widely studied under a strongly convex lower-level function, see, e.g., (Pedregosa, 2016; Ghadimi and Wang, 2018; Hong et al., 2023; Ji et al., 2021a; Chen et al., 2021; Khanduri et al., 2021; Shen and Chen, 2022; Li et al., 2022; Xiao et al., 2023b; Giovannelli et al., 2022; Chen et al., 2023b; Yang et al., 2023). The iterative differentiation (ITD) methods, which can be viewed as an iterative relaxation of the IG methods, have been studied in, e.g., (Maclaurin et al., 2015; Franceschi et al., 2017; Nichol et al., 2018; Shaban et al., 2019; Liu et al., 2021b, 2022; Bolte et al., 2022; Grazzi et al., 2020; Ji et al., 2022; Shen and Chen, 2022). However, in our case

Table 2: A holistic comparison between the bilevel RL in this work and the general penalty-based bilevel optimization (OPT) (e.g., (Shen and Chen, 2023; Kwon et al., 2023)), where "GD" stands for the gradient descent and "PMD" stands for the policy mirror descent.

| | Supervised penalty-based bilevel OPT | This work on penalty-based bilevel RL |
|---|---|---|
| Problem application | hyperparameter OPT, adversarial training, continue learning, etc. | Stackelberg Markov game, RL from preference, incentive design, etc |
| Penalty reformulation | Value penalty with assumed property | Value/Bellman/NI penalty with proven property |
| Algorithm | Gradient directly accessible | Need to derive close form gradient and estimate it |
| Iteration complexity | $\tilde{\mathcal{O}}(\lambda\epsilon^{-1})$ with inner-loop GD | $\tilde{\mathcal{O}}(\lambda\epsilon^{-1})$ with inner-loop PMD |

the lower-level objective is the discounted return which is known to be non-convex (Agarwal et al., 2020). Thus it is difficult to apply the aforementioned methods here.

The penalty relaxation of the bilevel optimization problem, early studies of which can be dated back to (Clarke, 1983; Luo et al., 1996), have gained interest from researchers recently (see, e.g., (Shen and Chen, 2023; Ye et al., 2022; Lu and Mei, 2023; Kwon et al., 2023; Xiao et al., 2023a)). Theoretical results for this branch of work are established under certain lower-level error-bound conditions (e.g., uniform Polyak-Łojasiewicz inequalities) weaker than strong convexity. In our case, the lower-level discounted return objective does not satisfy those uniform error bounds. Therefore, the established penalty reformulations may not be directly applied here. See Table 2 for a more detailed comparison between this work and the general penalty-based bilevel optimization.

**Applications of bilevel RL.** The bilevel RL formulation considered in this work covers several applications including reward shaping (Hu et al., 2020; Zou et al., 2019), reinforcement learning from preference (Christiano et al., 2017; Xu et al., 2020; Pacchiano et al., 2021), Stackelberg Markov game (Liu et al., 2021a; Song et al., 2023), incentive design (Chen et al., 2016), etc. A concurrent work (Chakraborty et al., 2024) studies the policy alignment problem, and introduces a corrected reward learning objective for RLHF that leads to strong performance gain. The PARL algorithm in (Chakraborty et al., 2024) is based on the implicit gradient bilevel optimization method that requires the strong convexity of the lower-level objective. On the other hand, PARL uses second-order derivatives of the RL objective, while our proposed algorithm is fully first-order.

## 2 Problem Formulations

In this section, we will first introduce the generic bilevel RL formulation. Then we will show several specific applications of the generic bilevel RL problem.

### 2.1 Bilevel reinforcement learning formulation

Reinforcement learning studies the problem where an agent aims to find a policy that maximizes its accumulated reward under the environment's dynamic. In such a problem, the reward function and the dynamic are fixed given the agent's policy. While in the problem that we are about to study, the reward or the dynamic oftentimes depend on another decision variable, e.g., the reward is parameterized by a neural network in RLHF; or in the Stackelberg game, both the reward and the dynamic are affected by the leader's policy.

Tailoring to this, we first define a so-called parameterized MDP. Given the parameter $x \in \mathbb{R}^{d_x}$, define a parameterized MDP as $\mathcal{M}_\tau(x) \coloneqq \{\mathcal{S}, \mathcal{A}, r_x, \mathcal{P}_x, \tau h\}$ where $\mathcal{S}$ is a finite state space; $\mathcal{A}$ is a finite action space; $r_x(s,a)$ is the parameterized reward given state-action pair $(s,a) \in \mathcal{S}\times\mathcal{A}$; $\mathcal{P}_x$ is a parameterized transition distribution that specifies $\mathcal{P}_x(s'|s,a)$–the probability of transiting to $s'$ given $(s,a)$; a policy $\pi$ specifies $\pi(a|s)$ which is the probability of taking action $a$ given state $s$; and $\tau h$ is the regularization: $\tau \geq 0$ and $h = (h_s)_{s\in\mathcal{S}}$ where each $h_s : \Delta(\mathcal{A}) \mapsto \mathbb{R}_+$ is a strongly-convex regularization function given $s$. When $\tau = 0$, $\mathcal{M}_\tau(x)$ is an unregularized MDP.

Given a policy $\pi$, the value function of $\mathcal{M}_\tau(x)$ under $\pi$ is defined as

$$V^\pi_{\mathcal{M}_\tau(x)}(s) \coloneqq \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \big(r_x(s_t,a_t) - \tau h_{s_t}(\pi(s_t))\big) \big| s_0 = s, \pi\Big] \tag{2.1}$$

where $\gamma \in [0,1)$, $\pi(s) \coloneqq \pi(\cdot|s) \in \Delta(\mathcal{A})$ and the expectation is taken over the trajectory $(s_0, a_0 \sim \pi(s_0), s_1 \sim \mathcal{P}_x(\cdot|s_0,a_0), \dots)$. Given a state distribution $\rho$, we write $V^\pi_{\mathcal{M}_\tau(x)}(\rho) = \mathbb{E}_{s\sim\rho}[V^\pi_{\mathcal{M}_\tau(x)}(s)]$. We also define the $Q$ function as

$$Q^\pi_{\mathcal{M}_\tau(x)}(s,a) \coloneqq r_x(s,a) + \gamma \mathbb{E}_{s'\sim\mathcal{P}_x(\cdot|s,a)}\big[V^\pi_{\mathcal{M}_\tau(x)}(s')\big]. \tag{2.2}$$

and $P^\pi_x(s_t = s|s_0)$ as the probability of reaching state $s$ at time $t$ given initial state $s_0$ under a transition distribution $\mathcal{P}_x$ and a policy $\pi$. The probability $P^\pi_x(s_t = s|s_0,a_0)$ can be defined similarly.

Suppose the policy $\pi$ is parameterized by $y \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$. We define the policy class as $\Pi \coloneqq \{\pi_y : y \in \mathcal{Y}\} \subseteq \Delta(\mathcal{A})^{|\mathcal{S}|}$. For $\mathcal{M}_\tau(x)$, its optimal policy is denoted as $\pi^*_y(x) \in \Pi$ satisfying $V^{\pi^*_y(x)}_{\mathcal{M}_\tau(x)}(s) \geq V^\pi_{\mathcal{M}_\tau(x)}(s)$ for any $\pi \in \Pi$ and $s$. With a function $f : \mathbb{R}^{d_x} \times \mathbb{R}^{d_y} \mapsto \mathbb{R}$, we are interested in the following bilevel RL problem

$$\mathcal{BM}: \min_{x,y} f(x,y), \text{ s.t. } x \in \mathcal{X},\ y \in \mathcal{Y}^*(x) \coloneqq \operatorname*{argmin}_{y\in\mathcal{Y}} -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) \tag{2.3}$$

where $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ and $\mathcal{Y} \subseteq \mathbb{R}^{d_y}$ are compact convex sets; and $\rho$ is a given state distribution with $\rho(s) > 0$ on $\mathcal{S}$. The name 'bilevel' refers to the nested structure in the optimization problem: in the upper level, a function $f(x,y)$ is minimized subject to the lower-level optimality constraint that $\pi_y$ is the optimal policy for $\mathcal{M}_\tau(x)$.

### 2.2 Applications of bilevel reinforcement learning

Next we show several example applications that can be modeled by a bilevel RL problem.

**Stackelberg Markov game.** Consider a Markov game where at each time step, a leader and a follower observe the state and take actions simultaneously. Then according to the current state and actions, the leader and follower receive rewards, and the game transits to the next state. Such a MDP can be defined as $\mathcal{M}_\tau^g := \{\mathcal{S}, \mathcal{A}_l, \mathcal{A}_f, r_l, r_f, \mathcal{P}, \tau h_l, \tau h_f\}$ where $\mathcal{S}$ is the state space; $\mathcal{A}_l$ or $\mathcal{A}_f$ is the leader's or the follower's action space; $r_l(s, a_l, a_f)$ and $r_f(s, a_l, a_f)$ are respectively the leader's and the follower's reward given $(s, a_l, a_f) \in \mathcal{S} \times \mathcal{A}_l \times \mathcal{A}_f$; $\mathcal{P}(s'|s, a_l, a_f)$ is the probability of transiting to state $s'$ given $(s, a_l, a_f)$; the leader's/follower's policy $\pi_x/\pi_y$ defines $\pi_x(a_l|s)/\pi_y(a_f|s)$–the probability of choosing action $a_l/a_f$ given state $s$; and $\tau h_l, \tau h_f$ are the regularization functions respectively for $\pi_x$ and $\pi_y$.

Define the leader's/follower's value function as

$$V_\star^{\pi_x,\pi_y}(s) := \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \big(r_\star(s_t, a_{l,t}, a_{f,t}) - \tau h_{\star,s_t}(\pi_\star(s_t))\big) \big| s_0 = s, \pi_x, \pi_y\Big],\ \star = l \text{ or } f$$

where $\gamma \in [0, 1)$, $\pi_\star(s) := \pi_\star(\cdot|s) \in \Delta(\mathcal{A}_\star)$ and the expectation is taken over the trajectory $(s_0,\ a_{l,0} \sim \pi_x(s_0),\ a_{f,0} \sim \pi_y(s_0),\ s_{l,1} \sim \mathcal{P}(s_0, a_{l,0}, a_{f,0}), \dots)$. The Q function is defined as

$$Q_\star^{\pi_x,\pi_y}(s, a_l, a_f) := r_\star(s, a_l, a_f) + \gamma \mathbb{E}_{s' \sim \mathcal{P}(s,a_l,a_f)}\big[V_\star^{\pi_x,\pi_y}(s')\big].$$

The follower's objective is to find a best-response policy to the leader's policy while the leader aims to find a best-response to the follower's best-response. The Stackelberg Markov game can be formulated as

$$\min_{x,y} -V_l^{\pi_x,\pi_y}(\rho),\ \text{s.t. } x \in \mathcal{X},\ y \in \operatorname*{argmin}_{y \in \mathcal{Y}} -V_f^{\pi_x,\pi_y}(\rho). \tag{2.4}$$

With the proof deferred to Appendix B.1, (2.4) can be viewed as a bilevel RL problem with $f(x, y) = -V_l^{\pi_x,\pi_y}(\rho)$ and a $\mathcal{M}_\tau(x)$ in which $r_x(s, a_f) = \mathbb{E}_{a_l \sim \pi_x(s)}[r_l(s, a_l, a_f)]$ and $\mathcal{P}_x(\cdot|s, a_f) = \mathbb{E}_{a_l \sim \pi_x(s)}[\mathcal{P}(\cdot|s, a_l, a_f)]$.

**Reinforcement learning from human feedback (RLHF).** In the RLHF setting, the agent learns a task without knowing the true reward function. Instead, humans evaluate pairs of state-action segments, and for each pair, they label the segment they prefer. The agent's goal is to learn the task well with a limited amount of labeled pairs.

The original framework of deep RL from human feedback in (Christiano et al., 2017) (we call it DRLHF) consists of two possibly asynchronous learning processes: reward learning from labeled pairs and RL from learned rewards. In short, we maintain a buffer of labeled segment pairs $\{(d_0, l_0, d_1, l_1)_i\}_i$ where each segment $d = (s_t, a_t, \dots, s_{t+T}, a_{t+T})$ is collected with the agent's policy $\pi_y$ and $l_0, l_1$ is the label (e.g., $l_1 = 1, l_0 = 0$ indicates segment $d_1$ is preferred over $d_0$). DRLHF simultaneously learns a reward predictor $r_x$ with the data and trains an RL agent using the learned reward. This process has a hierarchy structure and can be reformulated as a bilevel RL problem:

$$\begin{aligned}&\min_{x,y} -\mathbb{E}_{d_0,d_1 \sim \pi_y}\big[l_0 \log P(d_0 \succ d_1|r_x) + l_1 \log P(d_1 \succ d_0|r_x)\big],\\ &\text{s.t. } y \in \operatorname*{argmin}_y -V_{\mathcal{M}_\tau(x)}^{\pi_y}(\rho).\end{aligned}$$

where $P(d_0 \succ d_1|r_x)$ is the probability of preferring $d_0$ over $d_1$ under reward $r_x$, given by the Bradley-Terry model:

$$P(d_0 \succ d_1|r_x) = \frac{\exp(\sum_{s_t,a_t\in d_0} r_x(s_t,a_t))}{\exp(\sum_{s_t,a_t\in d_0} r_x(s_t,a_t)) + \exp(\sum_{s_t,a_t\in d_1} r_x(s_t,a_t))}. \tag{2.5}$$

**Reward shaping.** In the RL tasks where the reward is difficult to learn from (e.g., the reward signal is sparse where most states give zero reward), we can reshape the reward to enable efficient policy learning while staying true to the original task. Given a task specified by $\mathcal{M}_\tau = \{\mathcal{S}, \mathcal{A}, r, \mathcal{P}, \tau h\}$, the reward shaping problem (Hu et al., 2020) seeks to find a reshaped reward $r_x$ parameterized by $x \in \mathcal{X}$ such that the new MDP with $r_x$ enables more efficient policy learning for the original task. We can define the new MDP as $\mathcal{M}_\tau(x) = \{\mathcal{S}, \mathcal{A}, r_x, \mathcal{P}, \tau h\}$ and formulate the problem as:

$$\min_{x,y} -V^{\pi_y}_{\mathcal{M}_\tau}(\rho),\ \text{s.t. } x\in\mathcal{X},\ y\in\operatorname*{argmin}_{y\in\mathcal{Y}} -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) \tag{2.6}$$

which is a special case of bilevel RL.

## 3 Penalty Reformulation of Bilevel RL

A natural way to solve the bilevel RL problem $\mathcal{BM}$ is through reduction to a single-level problem, that is, to find a single-level problem that shares its local/global solutions with the original problem. Then by solving the single-level problem, we can recover the original solutions. In this section, we will perform single-level reformulation of $\mathcal{BM}$ by penalizing the upper-level objective with carefully chosen functions.

Specifically, we aim to find penalty functions $p(x, y)$ such that the solutions of the following problem recover the solutions of $\mathcal{BM}$:

$$\mathcal{BM}_{\lambda p}:\ \min_{x,y} F_\lambda(x,y) \coloneqq f(x,y) + \lambda p(x,y),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y} \tag{3.1}$$

where $\lambda$ is the penalty constant.

### 3.1 Value penalty and its landscape property

In $\mathcal{BM}$, the lower-level problem of finding the optimal policy $\pi_y$ can be rewritten as its optimality condition: $-V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) = 0$. Thus, $\mathcal{BM}$ can be rewritten as

$$\min_{x,y} f(x,y),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y},\ -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) = 0.$$

A natural penalty function that we call *value penalty* measures the lower-level optimality:

$$p(x,y) = -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho). \tag{3.2}$$

The value penalty specifies the following penalized problem

$$\begin{aligned}\mathcal{BM}_{\lambda p}:\ \min_{x,y} F_\lambda(x,y) &= f(x,y) + \lambda\big(-V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}\big),\\ &\text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y}.\end{aligned} \tag{3.3}$$

To capture the relation between solutions of $\mathcal{BM}_{\lambda p}$ and $\mathcal{BM}$, we have the following lemma.

**Lemma 1 (Relation on solutions)** *Consider choosing $p$ as the value penalty in (3.2). Assume there exists constant $C$ such that $\max_{x\in\mathcal{X},y\in\mathcal{Y}}|f(x,y)|=\frac{C}{2}$. Given accuracy $\delta>0$, choose $\lambda\geq C\delta^{-1}$. If $(x_\lambda,y_\lambda)$ achieves $\epsilon$-minimum of $\mathcal{BM}_{\lambda p}$, it achieves $\epsilon$-minimum of the relaxed $\mathcal{BM}$:*

$$\min_{x,y} f(x,y),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y},\ -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)+\max_{y\in\mathcal{Y}}V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)\leq\epsilon_\lambda \tag{3.4}$$

*where $\epsilon_\lambda\leq\delta+\lambda^{-1}\epsilon$.*

The proof is deferred to Appendix B.2. Perhaps one restriction of the above lemma is that it requires the boundedness of $f$ on $\mathcal{X}\times\mathcal{Y}$. This assumption is usually mild in RL problems, e.g., it is guaranteed in the Stackelberg game provided the reward functions are bounded.

Since $\mathcal{BM}_{\lambda p}$ is in general a non-convex problem, it is also of interest to connect the local solutions between $\mathcal{BM}_{\lambda p}$ and $\mathcal{BM}$. To achieve this, some structural condition is required. Suppose we use direct policy parameterization: $y$ is a vector with its $(s,a)$ element $y_{s,a}=\pi_y(a|s)$, and thus $y=\pi_y$ directly. Then we can prove the following structural condition.

**Lemma 2 (Gradient dominance)** *Given convex policy class $\Pi$ and any $\tau\geq 0$, it holds for any $\pi\in\Pi$ that*

$$\max_{\pi'\in\Pi}\langle\nabla_\pi V^\pi_{\mathcal{M}_\tau(x)}(\rho),\,\pi'-\pi\rangle\geq\frac{1}{(1-\gamma)\min_s\rho(s)}\Big(\max_{\tilde\pi\in\Pi}V^{\tilde\pi}_{\mathcal{M}_\tau(x)}(\rho)-V^\pi_{\mathcal{M}_\tau(x)}(\rho)\Big).$$

See Appendix B.3 for a proof. A similar gradient dominance property was first proven in (Agarwal et al., 2020, Lemma 4.1) for the unregularized MDPs. The above lemma is a generalization of the result in (Agarwal et al., 2020) to a regularized case. Under such a structure of the lower-level problem, we arrive at the following lemma capturing the relation on local solutions.

**Lemma 3 (Relation on local solutions)** *Consider using direct policy parameterization and choosing $p$ as the value penalty in (3.2). Assume $f(x,\cdot)$ is $L$-Lipschitz-continuous on $\mathcal{Y}$. Given accuracy $\delta>0$, choose $\lambda\geq LC_u\delta^{-1}$ where $C_u$ is a constant specified in the proof. If $(x_\lambda,y_\lambda)$ is a local solution of $\mathcal{BM}_{\lambda p}$, it is a local solution of the relaxed $\mathcal{BM}$ in (3.4) with an $\epsilon_\lambda\leq\delta$.*

The proof can be found in Appendix B.4. Lemmas 1 and 3 suggest we can recover the local/global solutions of the bilevel RL problem $\mathcal{BM}$ by locally/globally solving its penalty reformulation $\mathcal{BM}_{\lambda p}$ with the value penalty.

### 3.2 Bellman penalty and its landscape property

Next we introduce the Bellman penalty that can be used as an alternative. To introduce this penalty function, we consider a tabular policy (direct parameterization) $\pi_y$, i.e. $\pi_y(\cdot|s)=y_s$ for all $s$ and $y=(y_s)_{s\in\mathcal{S}}\in\mathcal{Y}=\Pi$. Then we can define the *Bellman penalty* as

$$p(x,y)=g(x,y)-v(x)\ \text{where}\ v(x)\coloneqq\min_{y\in\mathcal{Y}}g(x,y). \tag{3.5}$$

Here $g(x,y)$ is defined as

$$g(x,y) \coloneqq \mathbb{E}_{s\sim\rho}[\langle y_s,\, q_s(x)\rangle + \tau h_s(y_s)], \tag{3.6}$$

where $q_s(x) \in \mathbb{R}^{|\mathcal{A}|}$ is the vector of optimal Q functions, which is defined as

$$q_s(x) = (q_{s,a}(x))_{a\in\mathcal{A}} \text{ where } q_{s,a}(x) \coloneqq -\max_{\pi\in\Pi} Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a). \tag{3.7}$$

It is immediate that $p(x,\cdot)$ is $\tau$-strongly-convex uniformly for any $x\in\mathcal{X}$ by the 1-strong-convexity of $h_s$, and $p(x,y)\geq 0$ by definition. Moreover, we can show that $p(x,y) = g(x,y)-v(x)$ is a suitable optimality metric of the lower-level RL problem in $\mathcal{BM}$. Specifically, we prove that the lower-level RL problem is solved whenever $g(x,y)-v(x)$ is minimized in the following lemma.

**Lemma 4** *Assume $\tau>0$, then we have the following holds.*

- *Given any $x\in\mathcal{X}$, MDP $\mathcal{M}_\tau(x)$ has a unique optimal policy $\pi^*_y(x)$. And we have $\arg\min_{y\in\mathcal{Y}} g(x,y) = \mathcal{Y}^*(x) = \{\pi^*_y(x)\}$. Therefore, $\mathcal{BM}$ can be rewritten as the following problem with $\epsilon=0$:*

$$\mathcal{BM}_\epsilon:\ \min_{x,y} f(x,y),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y},\\ g(x,y)-v(x)\leq\epsilon. \tag{3.8}$$

- *Assume $f(x,\cdot)$ is L-Lipschitz-continuous on $\mathcal{Y}$. More generally for $\epsilon\geq 0$, $\mathcal{BM}_\epsilon$ is an $\epsilon$-approximate problem of $\mathcal{BM}$ in a sense that: given any $x\in\mathcal{X}$, any feasible policy $y_\epsilon$ of $\mathcal{BM}_\epsilon$ is $\epsilon$-feasible for $\mathcal{BM}$:*

$$\|y_\epsilon - \pi^*_y(x)\|^2 \leq \tau^{-1}\epsilon.$$

*Moreover, let $f^*$,$f^*_\epsilon$ respectively be the optimal objective value of $\mathcal{BM}$ and $\mathcal{BM}_\epsilon$, then we have $|f^*-f^*_\epsilon|\leq L\sqrt{\tau^{-1}\epsilon}$.*

The proof is deferred to Appendix B.5. Based on Lemma 4, $g(x,y)-v(x)$ is a suitable optimality metric for the lower-level problem. It is then natural to consider whether we can use it as a penalty function for the lower-level sub-optimality. The Bellman penalty specifies the following penalized problem:

$$\mathcal{BM}_{\lambda p}:\ \min_{x,y} F_\lambda(x,y) = f(x,y) + \lambda\big(g(x,y)-v(x)\big),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y}. \tag{3.9}$$

We have the following result that captures the relation between the solution of $\mathcal{BM}_\epsilon$ and $\mathcal{BM}_{\lambda p}$, which proves the Bellman penalty is indeed a suitable penalty function.

**Lemma 5 (Relation on solutions)** *Consider choosing $p$ as the Bellman penalty in* (3.5)*. Assume $f(x,\cdot)$ is L-Lipschitz-continuous on $\mathcal{Y}$. Given some accuracy $\delta>0$, choose $\lambda\geq L\sqrt{\tau^{-1}\delta^{-1}}$. If $(x_\lambda,y_\lambda)$ is a local/global solution of $\mathcal{BM}_{\lambda p}$, then it is a local/global solution of $\mathcal{BM}_{\epsilon_\lambda}$ with $\epsilon_\lambda\leq\delta$.*

This lemma follows from Proposition 3 in (Shen and Chen, 2023) under the $\tau$-strong-convexity of $g(x,\cdot)$ along with the assumptions in this lemma.

## 4 A Penalty-based Bilevel RL Algorithm

In the previous sections, we have introduced two penalty functions $p(x, y)$ such that the original problem $\mathcal{BM}$ can be approximately solved via solving $\mathcal{BM}_{\lambda p}$. However, it is still unclear how $\mathcal{BM}_{\lambda p}$ can be solved. One challenge is the differentiability of the penalty function $p(x, y)$ in (3.1). In this section, we will first study when $F_\lambda(x, y)$ admits gradients in the generic case, and we will show the specific gradient forms in each application. Based on these results, we propose a penalty-based algorithm and further establish its convergence.

### 4.1 Differentiability of the value penalty

We first consider the value penalty

$$p(x, y) = -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho).$$

For the differentiability in $y$, it follows $\nabla_y p(x, y) = -\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$ which can be conveniently evaluated with the policy gradient theorem. The issue lies in the differentiability of $p(x, y)$ with respect to $x$, where $p(x, y)$ may not be differentiable in $x$ due to the optimality function $\max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$. Fortunately, we will show that in the setting of RL, $p(\cdot, y)$ admits closed-form gradient under relatively mild assumptions below.

**Assumption 1** *Assume*

*(a)* $\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$ *is continuous in* $(x, y)$*; and,*

*(b) given any* $x \in \mathcal{X}$ *and* $y, y' \in \mathcal{Y}^*(x)$*, we have* $\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) = \nabla_x V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\rho)$.

Assumption 1 (a) is mild in the applications, and can often be guaranteed by a continuously differentiable reward function $r_x$. A sufficient condition of Assumption 1 (b) is the optimal policy of $\mathcal{M}_\tau(x)$ on $\Pi$ is unique, e.g., when $\pi_y = \pi_{y'}$ for $y, y' \in \mathcal{Y}^*(x)$. As indicated by Lemma 4, the uniqueness is guaranteed when $\tau > 0$.

**Lemma 6 (Generic gradient form)** *Consider the value penalty* $p$ *in* (3.2)*. Suppose Assumption 1 holds. Then* $p(x, y)$ *is differentiable in* $x$ *with the gradient*

$$\nabla_x p(x, y) = -\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \nabla_x V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)|_{\pi = \pi^*_y(x)} \tag{4.1}$$

*where recall* $\pi^*_y(x)$ *is an optimal policy on policy class* $\Pi = \{\pi_y : y \in \mathcal{Y}\}$ *of* $\mathcal{M}_\tau(x)$.

The proof can be found in Appendix C.1. Next, we can apply the generic result from Lemma 6 to specify the exact gradient formula in different bilevel RL applications discussed in Section 2.2.

**Lemma 7 (Gradient form in the applications)** *Consider the value penalty* $p$ *in* (3.2)*. The gradient of the penalty function in specific applications is listed below.*

*(a) RLHF/reward shaping: Assume* $r_x$ *is continuously differentiable and Assumption 1 (b) holds. Then Lemma 6 holds and we have*

$$\nabla_x p(x, y) = -\mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \nabla r_x(s_t, a_t) \big| \rho, \pi_y\Big] + \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \nabla r_x(s_t, a_t) \big| \rho, \pi^*_y(x)\Big].$$

*(b) Stackelberg game: Assume $\pi_x$ is continuously differentiable and Assumption 1 (b) holds. Then Lemma 6 holds and we have*

$$\nabla_x p(x,y) = -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t \bar{Q}_{f,t}^{\pi_x,\pi_y}\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big] + \mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t \bar{Q}_{f,t}^{\pi_x,\pi_y^*(x)}\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y^*(x)\Big]$$

*where $\bar{Q}_{f,t}^{\pi_x,\pi_y} := Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t}) - \tau h_{f,s_t}(\pi_y(s_t))$. Recall in the Stackelberg setting, $\pi_y^*(x)$ is the optimal follower policy given $\pi_x$; and the expectation is taken over the trajectory generated by $\pi_x,\pi_y$(or $\pi_y^*(x)$), $\mathcal{P}$.*

We defer the proof to Appendix C.2.

### 4.2 Differentiability of the Bellman penalty

For the Bellman penalty in (3.5), though it is straightforward to evaluate $\nabla_y p(x,y) = \nabla_y g(x,y)$, the differentiability of $p(x,y)$ in $x$ is unclear. We next identify some sufficient conditions that allow convenient evaluation of $\nabla_x p(x,y)$.

**Assumption 2** *Assume $\tau > 0$ and the following hold:*

*(a) Given any $(s,a)$, $\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)$ exists and is continuous in $(x,\pi)$; and,*

*(b) Either the discount factor $\gamma = 0$ or: Given $x \in \mathcal{X}$, for the MDP $\mathcal{M}_\tau(x)$, the Markov chain induced by any policy $\pi \in \Pi$ is irreducible*[1]*.*

Assumption 2 (a) is mild and can be satisfied in the applications in Section 2.2. Assumption 2 (b) is a regularity assumption on the MDP (Mitrophanov, 2005), and is often assumed in recent studies on policy gradient algorithms (see e.g., (Wu et al., 2020; Qiu et al., 2021; **?**)).

**Lemma 8 (Generic gradient form)** *Consider the Bellman penalty $p$ in (3.5). Assume Assumption 2 holds. Then $p(x,y)$ is differentiable with the gradient $\nabla_x p(x,y) = \nabla_x g(x,y) - \nabla v(x)$ where*

$$\nabla_x g(x,y) = -\mathbb{E}_{s\sim\rho,a\sim\pi_y(s)}\big[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\big]\big|_{\pi=\pi_y^*(x)} \tag{4.2}$$

$$\nabla v(x) = -\mathbb{E}_{s\sim\rho,a\sim\pi_y^*(x)(s)}\big[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\big]\big|_{\pi=\pi_y^*(x)} \tag{4.3}$$

The proof can be found in Appendix C.3. The above lemma provides the form of gradients for the $\mathcal{BM}$ problem. Next, we show that Lemma 8 holds for the example applications in Section 2.2 and then compute the closed-form of the gradients.

**Lemma 9 (Gradient form in the applications)** *Consider the Bellman penalty $p(x,y)$ in (3.5). The gradient form of the bilevel RL applications is listed below.*

1. In $\mathcal{M}_\tau(x)$, the Markov chain induced by policy $\pi$ is irreducible if for any state $s$ and initial state-action pair $s_0,a_0$, there exists time step $t$ such that $P_x^\pi(s_t = s|s_0,a_0) > 0$, where $P_x^\pi(s_t = s|s_0,a_0)$ is the probability of reaching $s$ at time step $t$ in MDP $\mathcal{M}_\tau(x)$ with policy $\pi$.

(a) *RLHF/reward shaping: Assume $r_x$ is continuously differentiable and Assumption 2 (b) holds. Then Lemma 8 holds and we have*

$$\nabla_x g(x,y) = -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t \nabla r_x(s_t,a_t)\big|s_0\sim\rho, a_0\sim\pi_y(s),\pi_y^*(x)\Big],$$
$$\nabla v(x) = -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t \nabla r_x(s_t,a_t)\big|s_0\sim\rho,\pi_y^*(x)\Big]$$

*where the expectation is taken over the trajectory generated by $\pi_y^*(x)$ and $\mathcal{P}$.*

(b) *Stackelberg game: Assume $\pi_x$ is continuously differentiable and Assumption 2 (b) holds. Then Lemma 8 holds and we have*

$$\begin{aligned}\nabla_x g(x,y) =& -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y^*(x)}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0\sim\rho, a_{f,0}\sim\pi_y(s_0),\pi_x,\pi_y^*(x)\Big]\\ &+\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t\tau h_{f,s_t}(\pi_y^*(x)(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0\sim\rho, a_{f,0}\sim\pi_y(s_0),\pi_x,\pi_y^*(x)\Big]\\ \nabla v(x) =& -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y^*(x)}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0\sim\rho,\pi_x,\pi_y^*(x)\Big]\\ &+\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t\tau h_{f,s_t}(\pi_y^*(x)(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0\sim\rho,\pi_x,\pi_y^*(x)\Big]\end{aligned}$$

*Recall in the Stackelberg setting, $\pi_y^*(x)$ is the optimal follower policy given $\pi_x$; and the expectation is taken over the trajectory generated by $\pi_x,\pi_y^*(x)$ and $\mathcal{P}$.*

The proof is deferred to Appendix C.4 due to space limitation.

### 4.3 A gradient-based algorithm and its convergence

In the previous subsections, we have addressed the challenges of evaluating $\nabla p(x,y)$, enabling the gradient-based methods to optimize $F_\lambda(x,y)$ in (3.1). However, computing $\nabla p(x_k,y_k)$ possibly requires an optimal policy $\pi_y^*(x_k)$ of the lower-level RL problem $\mathcal{M}_\tau(x_k)$. Given $x_k$, the lower-level RL problem can be solved with a wide range of algorithms, and we can use an approximately optimal policy parameter $\hat{\pi}_k \approx \pi_y^*(x_k)$ to compute the approximate penalty gradient $\hat{\nabla}p(x_k,y_k;\hat{\pi}_k) \approx \nabla p(x_k,y_k)$. The explicit formula of $\hat{\nabla}p(x_k,y_k;\hat{\pi}_k)$ can be straightforwardly obtained by replacing the optimal policy with its approximate $\hat{\pi}_k$ in the formula of $\nabla p(x_k,y_k)$ presented in Lemmas 7 and 9. Therefore, we will defer the explicit formula to Appendix C.7 for ease of reading.

Given $\hat{\nabla}p(x_k,y_k;\hat{\pi}_k)$, we approximate the gradient $F_\lambda(x_k,y_k)$ as $\hat{\nabla}F_\lambda(x_k,y_k;\hat{\pi}_k) \coloneqq \nabla f(x_k,y_k) + \lambda\hat{\nabla}p(x_k,y_k;\hat{\pi}_k)$ and update

$$(x_{k+1},y_{k+1}) = \mathrm{Proj}_{\mathcal{Z}}\Big[(x_k,y_k) - \alpha\hat{\nabla}F_\lambda(x_k,y_k;\hat{\pi}_k)\Big] \tag{4.4}$$

where $\mathcal{Z} = \mathcal{X}\times\mathcal{Y}$, and this optimization process is summarized in Algorithm 1.

We next study the convergence of PBRL. To bound the error of the update in Algorithm 1, we make the following assumption on the sub-optimality of the policy $\hat{\pi}_k$.

**Algorithm 1** PBRL: Penalty-based Bilevel RL Gradient-descent

1: Select either the value or the Bellman penalty. Select $(x_1, y_1) \in \mathcal{Z} \coloneqq \mathcal{X} \times \mathcal{Y}$. Select step size $\alpha$, penalty constant $\gamma$ and iteration number $K$.
2: **for** $k = 1$ **to** $K$ **do**
3: Given RL problem $\mathcal{M}_\tau(x_k)$, solve for an approximately optimal policy $\hat{\pi}_k \in \Pi$.
4: Compute the penalty's approximate gradient $\hat{\nabla} p(x_k, y_k; \hat{\pi}_k) \approx \nabla p(x_k, y_k)$
5: Compute the inexact gradient of $F_\lambda$ as $\hat{\nabla} F_\lambda(x_k, y_k; \hat{\pi}_k) = \nabla f(x_k, y_k) + \lambda \hat{\nabla} p(x_k, y_k; \hat{\pi}_k)$
6: $(x_{k+1}, y_{k+1}) = \mathrm{Proj}_{\mathcal{Z}} \left[ (x_k, y_k) - \alpha \hat{\nabla} F_\lambda(x_k, y_k; \hat{\pi}_k) \right]$
7: **end for**

**Assumption 3 (Oracle accuracy)** *Given some accuracy $\epsilon_{orac}$ and step size $\alpha$, assume the following inequality holds*

$$\frac{1}{K}\sum_{k=1}^{K} 20\lambda^2 \|\hat{\nabla} p(x_k, y_k; \hat{\pi}_k) - \nabla p(x_k, y_k)\|^2 \leq \epsilon_{orac} + \frac{1}{K}\sum_{k=1}^{K} \frac{1}{\alpha^2} \|(x_{k+1}, y_{k+1}) - (x_k, y_k)\|^2.$$

This assumption only requires the running average error to be upper-bounded, which is milder than requiring the error to be upper-bounded for each iteration. A sufficient condition of the above assumption is $\|\hat{\pi}_k - \pi_y^*(x_k)\|^2 \leq \epsilon_{orac}$ with some constant $c$, which can be achieved by the policy mirror descent algorithm (see e.g., (Lan, 2023; Zhan et al., 2023)) with iteration complexity $\mathcal{O}(\log(\lambda^2/\epsilon_{orac}))$ (see a justification in Appendix C.7).

Furthermore, to guarantee worst-case convergence, the regularity condition that $f$ and $p$ are Lipschitz-smooth is required. We thereby identify a set of sufficient conditions for the value penalty or Bellman penalty to be smooth.

**Assumption 4 (Smoothness assumption)** *Assume given any $(s, a)$, $h_s(\pi_y(s))$ is $L_h$-Lipschitz smooth on $\mathcal{Y}$; and $Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s, a)$, $V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$ are $L_v$-Lipschitz-smooth on $\mathcal{X} \times \mathcal{Y}$.*

Assumption 4 is satisfied under a smooth $r_x$ and a smooth policy (e.g., softmax policy (Mei et al., 2020)), or a direct policy parameterization paired with smooth regularization function $h_s$. See a detailed justification of this in Appendix C.5.

**Lemma 10 (Lipschitz smoothness of penalty functions)** *Under Assumptions 2 and 4, the value or Bellman penalty function $p(x, y)$ is $L_p$-Lipschitz-smooth on $\mathcal{X} \times \mathcal{Y}$ with constant $L_p$ specified in the proof.*

We refer the reader to Appendix C.6 for proof. Given the smoothness of the penalty terms, we make the final regularity assumption on $f$.

**Assumption 5** *There exists constant $L_f$ such that $f(x, y)$ is $L_f$-Lipschitz smooth in $(x, y)$.*

The projected gradient is a commonly used metric in the convergence analysis of projected gradient type algorithms (Ghadimi et al., 2016). Define the projected gradient of $F_\lambda(x, y)$ as

$$G_\lambda(x_k, y_k) \coloneqq \frac{1}{\alpha}\big((x_k, y_k) - (\bar{x}_{k+1}, \bar{y}_{k+1})\big), \tag{4.5}$$

where $(\bar{x}_{k+1}, \bar{y}_{k+1}) \coloneqq \mathrm{Proj}_{\mathcal{Z}}((x_k, y_k) - \alpha \nabla F_\lambda(x_k, y_k))$. Now we are ready to present the convergence theorem of PBRL.

**Theorem 11 (Convergence of PBRL)** *Consider running the PBRL algorithm. Suppose Assumptions 2–5 hold. Choose step size* $\alpha \leq \frac{1}{L_f+\lambda L_p}$*, then we have*

$$\frac{1}{K}\sum_{k=1}^{K}\|G_\lambda(x_k,y_k)\|^2 \leq \frac{16\big(F_\lambda(x_1,y_1)-\inf_{(x,y)\in\mathcal{Z}} f(x,y)\big)}{\alpha K}+\epsilon_{orac}$$

See Appendix C.8 for the proof of the above theorem. At each outer iteration $k$, let $\mathrm{com}(\epsilon_{orac})$ be the oracle's iteration complexity. Then the above theorem suggests Algorithm 1 has an iteration complexity of $\mathcal{O}(\lambda\epsilon^{-1}\mathrm{com}(\epsilon_{orac}))$. When choosing the oracle as policy mirror descent so that $\mathrm{com}(\epsilon_{orac})=\mathcal{O}(\log(\lambda^2/\epsilon_{\mathrm{orac}}))$ (Lan, 2023; Zhan et al., 2023), we have Algorithm 1 has an iteration complexity of $\tilde{\mathcal{O}}(\lambda\epsilon^{-1})$.

## 5 Bilevel RL with Lower-level Zero-sum Games

In the previous sections, we have introduced a penalty method to solve the bilevel RL problem with a single-agent lower-level MDP. In this section, we seek to extend the previous idea to the case where the lower-level problem is a zero-sum Markov game (Shapley, 1953; Littman, 2001). We will first introduce the formulation of bilevel RL with a zero-sum Markov game as the lower-level problem, and then propose its penalty reformulation with a suitable penalty function. Finally, we establish the finite-time convergence for a projected policy gradient-type bilevel RL algorithm.

### 5.1 Formulation

Given a parameter $x\in\mathbb{R}^{d_x}$, consider a parameterized two-player zero-sum Markov game $\mathcal{M}_\tau(x)=\{\mathcal{S},\mathcal{A},r_x,\mathcal{P}_x,\tau h\}$ where $\mathcal{S}$ is a finite state space; $\mathcal{A}=\mathcal{A}_1\times\mathcal{A}_2$ is a finite joint action space, and $\mathcal{A}_1,\mathcal{A}_2$ are the action spaces of player 1 and 2 respectively; $r_x(s,a)$ ($a=(a_1,a_2)$ is the joint action) is player 1's parameterized reward, and player 2's reward is $-r_x$; the parameterized transition distribution $\mathcal{P}_x$ specifies $\mathcal{P}_x(s'|s,a)$, which is the probability of the next state being $s'$ given when the current state is $s$ and the players take joint action $a$. Furthermore, we let $\pi_i\in\Pi_i$ denote player $i$'s policy, where $\pi_i(a_i|s)$ is the probability of player $i$ taking action $a_i$ given state $s$. Here $\Pi_i$ is the policy class of player $i$ and we assume it is a convex set. We let $\pi\in\Pi=\Pi_1\times\Pi_2$ denote the joint policy.

Let $\tau h$ be a regularization parameter and $h_s$ be a regularization function at state $s\in\mathcal{S}$. Given the joint policy $\pi=(\pi_1,\pi_2)$, the (regularized) value function under $\pi$ is defined as

$$V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s)=V^{\pi}_{\mathcal{M}_\tau(x)}(s)\coloneqq\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t\big(r_x(s_t,a_t)-\tau h_{s_t}(\pi_1(s_t))+\tau h_{s_t}(\pi_2(s_t))\big)\big|s_0=s,\pi\Big] \tag{5.1}$$

where the expectation is taken over the trajectory generated by $a_t\sim(\pi_1(s_t),\pi_2(s_t))$ and $s_{t+1}\sim\mathcal{P}(s_t,a_t)$. Given some state distribution $\rho$, we write $V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)=\mathbb{E}_{s\sim\rho}[V^{\pi}_{\mathcal{M}_\tau(x)}(s)]$. We can also define the Q function as

$$Q^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s,a_1,a_2)=Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\coloneqq r(s,a)+\gamma\mathbb{E}_{s'\sim\mathcal{P}(s,a)}\big[V^{\pi}_{\mathcal{M}_\tau(x)}(s')\big]. \tag{5.2}$$

With a state distribution $\rho$ that satisfies $\min_s \rho(s) > 0$, the $\epsilon$-Nash-Equilibrium (NE) (Ding et al., 2022; Zhang et al., 2023; Ma et al., 2023) is a joint policy $\pi = (\pi_1, \pi_2)$ satisfying

$$NE_\epsilon(x) \coloneqq \Big\{(\pi_1, \pi_2) \in \Pi : V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) \geq V^{\pi_1',\pi_2}_{\mathcal{M}_\tau(x)}(\rho) - \epsilon,\ \forall \pi_1' \in \Pi_1 \text{ and } V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) \leq V^{\pi_1,\pi_2'}_{\mathcal{M}_\tau(x)}(\rho) + \epsilon,\ \forall \pi_2' \in \Pi_2\Big\}. \tag{5.3}$$

Then Nash equilibrium is defined as $\epsilon$-NE with $\epsilon = 0$ and $NE(x) = NE_0(x)$.

**Bilevel RL.** In the bilevel RL problem, we are interested in finding the optimal parameter $x$ such that the Nash equilibrium induced by such a parameter, maximizes an objective function $f$. The mathematical formulation is given as follows:

$$\min_{x,\pi} f(x, \pi),\ \text{s.t. } x \in \mathcal{X},\ \pi \in NE(x). \tag{5.4}$$

In the above problem, we aim to find a parameter $x$ and select among all the Nash Equilibria under $x$ such that a certain loss function $f$ is minimized. We next present a motivating example for this general problem.

**Motivating example: Incentive design.** Adaptive incentive design (Ratliff et al., 2019) involves an incentive designer who tries to manipulate self-interested agents by modifying their payoffs with carefully designed incentive functions. In the case where the agents are playing a zero-sum game, the incentive designer's problem (Yang et al., 2021) can be formulated as (5.4), given by

$$\min_{x,\pi} f(\pi) = -\mathbb{E}_{\pi,\mathcal{P}_{id}}\Big[\sum_{t=0}^{\infty} \gamma^t r_{\text{id}}(s_t, a_t) - c(s_t)\Big] \quad \text{s.t. } x \in \mathcal{X},\ \pi \in NE(x) \tag{5.5}$$

where $\mathcal{P}_{id}(\cdot|s, a)$ is the transition distribution of the designer; $r_{\text{id}}$ is the designer's reward, e.g., the social welfare reward (Yang et al., 2021); the function $c(s)$ is the designer's cost; the expectation is taken over the trajectory generated by agents' joint policy $\pi$ and transition $\mathcal{P}_{id}$; and in the lower level, the MDP $\mathcal{M}_\tau(x)$ is parameterized by $x$ via the incentive reward $r_x$. Note $r_x$ is the agents' reward, which is designed by the designer to control the behavior of the agents such that the designer's reward given by $r_{id}$ and $c$ is maximized.

### 5.2 The Nikaido-Isoda function as a penalty

Different from a static bilevel optimization problem, the problem in (5.4) does not have an optimization problem in the lower level; instead, it has a more abstract constraint set $\pi \in NE(x)$. Our first step is to formulate the problem in (5.4) to a bilevel optimization problem with an optimization reformulation of the Nash equilibrium seeking problem. In doing so, we will use the Nikaido-Isoda (NI) function first introduced in (Nikaidô and Isoda, 1955). It takes a special form in two-player zero-sum games:

$$\psi(x, \pi) \coloneqq \max_{\pi_1 \in \Pi_1} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) - \min_{\pi_2 \in \Pi_2} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho). \tag{5.6}$$

We have the following basic property of this function.

**Lemma 12 (Bilevel formulation)** *Given any $x$ and $\pi \in \Pi$, $\psi(x,\pi) \geq 0$, $\psi(x,\pi) \leq 2\epsilon$ if $\pi \in NE_\epsilon(x)$ and $\pi \in NE_\epsilon(x)$ if $\psi(x,\pi) \leq \epsilon$. Therefore, (5.4) is equivalent to the following bilevel optimization problem*

$$\mathcal{BZ}: \min_{x,\pi} f(x,\pi) \quad \text{s.t. } x \in \mathcal{X},\ \pi \in \arg\min_{\pi\in\Pi} \psi(x,\pi). \tag{5.7}$$

**Proof** From the definition (5.6), we have

$$\psi(x,\pi) = \big(-V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) + \max_{\pi_1\in\Pi_1} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)\big) + \big(V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) - \min_{\pi_2\in\Pi_2} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)\big). \tag{5.8}$$

The result follows since both terms in the RHS of (5.8) are nonnegative on $\Pi$. ■

By the above lemma, $\psi(x,\pi)$ is an optimality metric of the lower-level NE-seeking problem. Therefore, it is natural to consider when $\psi$ is a suitable penalty. Define the penalized problem as

$$\mathcal{BZ}_{\lambda p}: \min_{x,\pi} f(x,\pi) + \lambda\psi(x,\pi),\ \text{s.t. } x \in \mathcal{X},\ \pi \in \Pi. \tag{5.9}$$

To relate $\mathcal{BZ}_{\lambda p}$ with the original problem $\mathcal{BZ}$, certain structures of $\psi(x,\cdot)$ is required. A special structure of $\psi$ has been studied in previous works where each player's payoff is non-Markovian (see e.g., (Von Heusinger and Kanzow, 2009)). The result relies on certain monotonicity conditions on the payoff functions that do not hold in our Markovian setting. Instead, inspired by the previously discussed single-agent case, we prove a gradient dominance condition under the following assumption.

**Assumption 6** *Assume $\tau > 0$ and $V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)$ is continuously differentiable in $(x,\pi_1,\pi_2)$.*

For a justification of the stronger version of this assumption, please see Appendix D.3. Under this assumption, we can prove the following key lemma.

**Lemma 13 (Gradient dominance of $\psi$)** *If Assumption 6 holds, we have the following.*

*(a) Function $\psi$ is differentiable with*

$$\nabla_\pi \psi(x,\pi) = \big(-\nabla_{\pi_1} V^{\pi_1,\pi_2^*}_{\mathcal{M}_\tau(x)}(\rho), \nabla_{\pi_2} V^{\pi_1^*,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)\big)$$

*where $\pi_1^* \coloneqq \operatorname{argmax}_{\pi_1\in\Pi_1} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)$ and $\pi_2^*$ defined similarly; and*

$$\nabla_x \psi(x,\pi) = \nabla_x V^{\pi_1^*,\pi_2}_{\mathcal{M}_\tau(x)}(\rho) - \nabla_x V^{\pi_1,\pi_2^*}_{\mathcal{M}_\tau(x)}(\rho).$$

*(b) There exists a constant $\mu = (1-\gamma)\min_s \rho(s)$ such that given any $x$ and $\tau > 0$, $\psi(x,\pi)$ is $\mu$-gradient dominated in $\pi$:*

$$\max_{\pi'\in\Pi} \langle \nabla_\pi \psi(x,\pi),\ \pi - \pi' \rangle \geq \mu\psi(x,\pi),\ \forall \pi \in \Pi. \tag{5.10}$$

Please see Appendix D.1 for the proof. The proof is based on the gradient dominance condition of the single-agent setting in Lemma 2, along with the max-min special form of the NI function.

With Lemma 13, we are ready to relate $\mathcal{BZ}_{\lambda p}$ with $\mathcal{BZ}$.

**Lemma 14 (Relation on solutions)** *Assume Assumption 6 holds and $f(x,\pi)$ is L-Lipschitz-continuous in $\pi$. Given accuracy $\delta>0$, choose $\lambda \geq \delta^{-1}$. If $(x_\lambda, \pi_\lambda)$ is a local/global solution of $\mathcal{BZ}_{\lambda p}$, it is a local/global solution of the relaxed $\mathcal{BZ}$ with some $\epsilon_\lambda \leq \delta$:*

$$\mathcal{BZ}_\epsilon : \min_{x,\pi} f(x,\pi), \text{ s.t. } x \in \mathcal{X}, \ \pi \in \Pi, \ \psi(x,\pi) \leq \epsilon_\lambda. \tag{5.11}$$

The proof is deferred to Appendix D.2. The above lemma shows one can recover the local/global solution of the approximate problem of $\mathcal{BZ}$ by solving $\mathcal{BZ}_{\lambda p}$ instead. To solve for $\mathcal{BZ}_{\lambda p}$, we propose a projected gradient type update next and establish its finite-time convergence.

### 5.3 A policy gradient-based algorithm and its convergence analysis

To solve for $\mathcal{BZ}$, we consider a projected gradient update to solve for its penalized problem $\mathcal{BZ}_{\lambda p}$. To evaluate the objective function in $\mathcal{BZ}_{\lambda p}$, one will need to evaluate $\nabla\psi(x,\pi)$. Note that evaluating $\nabla_\pi \psi(x,\pi)$ requires the point $\pi_1^*(\pi_2, x)$ and $\pi_2^*(\pi_1, x)$ (defined in Lemma 13), which are optimal policies of a fixed MDP given parameters $(\pi_2, x)$ and $(\pi_1, x)$ respectively.

There are various efficient algorithms to find the optimal policy of a regularized MDP. Thus we assume that at each iteration $k$, we have access to some approximate optimal polices $\hat{\pi}_1^k \approx \pi_1^*(\pi_2^k, x^k)$ and $\hat{\pi}_2 \approx \pi_2^*(\pi_1^k, x^k)$ obtained by certain RL algorithms. With $\hat{\pi}^k = (\hat{\pi}_1^k, \hat{\pi}_2^k)$, we may denote the estimator of $\nabla\psi(x^k,\pi^k)$ as $\hat{\nabla}\psi(x^k,\pi^k;\hat{\pi}^k)$, the definition of which follows from Lemma 13 (a) with $\hat{\pi}_1^k$ and $\hat{\pi}_2^k$ in place of $\pi_1^*$ and $\pi_2^*$ respectively:

$$\begin{aligned}\hat{\nabla}\psi(x^k,\pi^k;\hat{\pi}^k) :=& \Big(\nabla_x V^{\hat{\pi}_1^k,\pi_2^k}_{\mathcal{M}_\tau(x^k)}(\rho) - \nabla_x V^{\pi_1^k,\hat{\pi}_2^k}_{\mathcal{M}_\tau(x^k)}(\rho), \\ & \big(-\nabla_{\pi_1} V^{\pi_1^k,\hat{\pi}_2^k}_{\mathcal{M}_\tau(x^k)}(\rho), \nabla_{\pi_2} V^{\hat{\pi}_1^k,\pi_2^k}_{\mathcal{M}_\tau(x^k)}(\rho)\big)\Big).\end{aligned}$$

We then perform a projected gradient-type update with this estimator:

$$(x^{k+1},\pi^{k+1}) = \mathrm{Proj}_{\mathcal{Z}}\Big[(x^k,\pi^k) - \alpha\big(\nabla f(x^k,\pi^k) + \lambda\hat{\nabla}\psi(x^k,\pi^k;\hat{\pi}^k)\big)\Big] \tag{5.12}$$

where $\mathcal{Z} = \mathcal{X}\times\Pi$. We make the following assumption on the sub-optimality of $\hat{\pi}_1^k$ and $\hat{\pi}_2^k$.

**Assumption 7 (Oracle accuracy)** *Given some pre-defined accuracy $\epsilon_{\mathrm{orac}} > 0$ and the step size $\alpha$, assume the approximate policies $\hat{\pi}_1^k$ and $\hat{\pi}_2^k$ satisfy the following inequality*

$$\frac{1}{K}\sum_{k=1}^K 20\lambda^2\|\hat{\nabla}\psi(x^k,\pi^k;\hat{\pi}^k) - \nabla\psi(x^k,\pi^k)\|^2 \leq \epsilon_{\mathrm{orac}} + \frac{1}{K}\sum_{k=1}^K \frac{1}{\alpha^2}\|(x^{k+1},\pi^{k+1}) - (x^k,\pi^k)\|^2.$$

The left-hand side of the above inequality can be upper bounded by the optimality gaps of the approximate optimal policies $\{\hat{\pi}_1^k, \hat{\pi}_2^k\}$. Note that here the policies $\{\hat{\pi}_1^k, \hat{\pi}_2^k\}$ are not approximate NE. Instead, $\hat{\pi}_1^k$ is the player 1's approximately optimal policy on the Markov model with parameter $x_k$, where player 2 adopts $\pi_2^k$. Thus, to obtain $\hat{\pi}_1^k$, one may use efficient single-agent policy optimization algorithms. For example, when using the policy mirror descent algorithm (Zhan et al., 2023), it will take an iteration complexity of $\mathcal{O}(\log(\lambda^2/\epsilon_{\mathrm{orac}}))$ to solve for accurate enough approximate policies. Similarly, $\hat{\pi}_2^k$ is an approximately optimal

policy of player 2 on the Markov game with parameter $x_k$, where player 1 adopts $\pi_1^k$. Thus $\hat{\pi}_2^k$ can similarly be efficiently obtained using a standard single-agent policy optimization algorithm. Furthermore, a more detailed justification of this assumption is provided in D.5.

We next identify sufficient conditions for the finite-time convergence in (5.12) as follows.

**Assumption 8 (Smoothness assumption of $\psi$)** *Suppose Assumption 6 holds. Additionally, assume the following arguments hold.*

*(a) Given any $s$, $V^{\pi}_{\mathcal{M}_\tau(x)}(s)$ is $L_v$-Lipschitz-smooth on $\mathcal{X} \times \Pi$;*

*(b) If the discount factor $\gamma > 0$ then assume given $x \in \mathcal{X}$, for any state $s$ and initial state-action $s_0, a_0$, there exists $t$ such that $P_x^{\pi}(s_t = s | s_0, a_0) > 0$, where $P_x^{\pi}(s_t = s|s_0, a_0)$ is the probability of reaching $s$ at time $t$ in the MDP $\mathcal{M}_\tau(x)$ under joint policy $\pi$.*

Assumption 8 (a) can be satisfied under a smooth regularization function, and smooth parameterized functions $r_x$ and $\mathcal{P}_x$; see the justification in Appendix D.3. Assumption 8 (b) is in the same spirit as Assumption 2 (b) in the single-agent case. Under Assumption 8, we can prove that the NI function is Lipschitz-smooth.

**Lemma 15 (Smoothness of $\psi$)** *Under Assumption 8, there exists a constant $L_\psi$ such that $\psi(x, \pi)$ is $L_\psi$-Lipschitz-smooth on $\mathcal{X} \times \Pi$.*

The proof of the above lemma can be found in Appendix D.4. With the above smoothness condition, we are ready to establish the convergence result. Define the projected gradient of the objective function in $\mathcal{BZ}_{\lambda p}$ (5.9) as

$$\mathcal{G}_\lambda(x^k, \pi^k) \coloneqq \frac{1}{\alpha}\big((x^k, \pi^k) - (\bar{x}^{k+1}, \bar{\pi}^{k+1})\big), \tag{5.13}$$

where $(\bar{x}^{k+1}, \bar{\pi}^{k+1}) \coloneqq \mathrm{Proj}_{\mathcal{Z}}\big((x^k, \pi^k) - \alpha\big(\nabla f(x^k, \pi^k) + \lambda \nabla \psi(x^k, \pi^k)\big)\big)$. Now we are ready to present the convergence theorem of update (5.12).

**Theorem 16 (Convergence of PBRL with zero-sum lower-level)** *Consider running update (5.12). Suppose Assumptions 5, 6, 7 and 8 hold. Choose step size $\alpha \le \frac{1}{L_f + \lambda L_\psi}$, then we have*

$$\frac{1}{K}\sum_{k=1}^{K} \|\mathcal{G}_\lambda(x^k, \pi^k)\|^2 \le \frac{16\big(f(x^1, \pi^1) + \lambda\psi(x^1, \pi^1) - \inf_{(x,y)\in\mathcal{Z}} f(x, y)\big)}{\alpha K} + \epsilon_{\mathrm{orac}}. \tag{5.14}$$

The proof is deferred to Appendix D.6. At each outer iteration $k$, let $\mathrm{com}(\epsilon_{\mathrm{orac}})$ be the oracle's iteration complexity. Then the above theorem suggests update (5.12) has an iteration complexity of $\mathcal{O}(\lambda\epsilon^{-1}\mathrm{com}(\epsilon_{\mathrm{orac}}))$. As discussed under Assumption 7, one could use policy mirror descent to solve for $\hat{\pi}_1^k, \hat{\pi}_2^k$ when estimating $\nabla\psi(x^k, \pi^k)$, then we have $\mathrm{com}(\epsilon_{\mathrm{orac}}) = \mathcal{O}(\log(\lambda^2/\epsilon_{\mathrm{orac}}))$. In such cases, update (5.12) has an iteration complexity of $\tilde{\mathcal{O}}(\lambda\epsilon^{-1})$.

## 6 Simulation

In this section, we test the empirical performance of PBRL in different tasks.

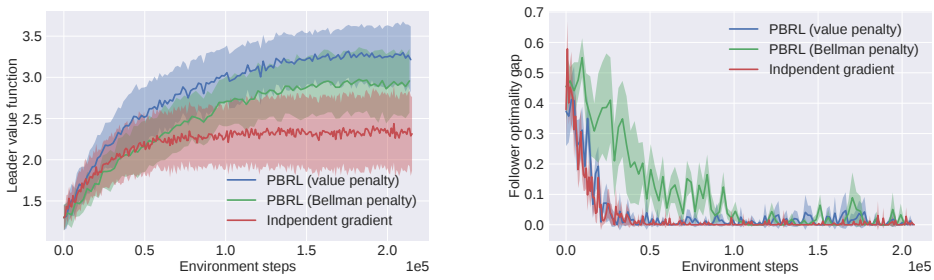


Figure 1: Stackelberg Markov games. The result is generated by running the algorithms in 10 random Stackelberg MDPs. The environment step is the total number of steps taken in the MDP, and is therefore also proportional to the total samples used in training. The leader's value function is $V_l^{\pi_{x_k},\pi_{y_k}}(\rho)$, and the follower's optimality gap is given by $V_f^{\pi_{x_k},\pi_y^*(x_k)}(\rho) - V_f^{\pi_{x_k},\pi_{y_k}}(\rho)$. A zero optimality gap means the follower has found the best response to the leader.

### 6.1 Stackelberg Markov game

We first seek to solve the Stackelberg Markov game formulated as

$$\min_x -V_l^{\pi_x,\pi_y^*(x)}(\rho), \text{ s.t. } x \in \mathbb{R}^{d_x}, \ \pi_y^*(x) = \operatorname*{argmin}_{\pi_y} -V_f^{\pi_x,\pi_y}(\rho), \tag{6.1}$$

where $\pi_x$ and $\pi_y$ is parameterized via the softmax function. Here the transition distribution and rewards are randomly generated. It has a state space of size $|\mathcal{S}| = 100$, and the leader, and follower's action space are of size $|\mathcal{A}_l| = 5$, $|\mathcal{A}_f| = 5$ respectively. Each entry of the rewards $R_l, R_f \in \mathbb{R}^{100\times5\times5}$ is uniformly sampled between $[0, 1]$ and values smaller than 0.7 are set to 0 to promote sparsity. Each entry of the transition matrix is sampled between $[0, 1]$ and then is normalized to be a distribution.

**Baseline.** We implement PBRL with both value and Bellman penalty, and compare them with the independent policy gradient method (Daskalakis et al., 2020; Ding et al., 2022). In the independent gradient method, each player myopically maximizes its own value function, i.e., the leader maximizes $V_l^{\pi_x,\pi_y}(\rho)$ while the follower maximizes $V_f^{\pi_x,\pi_y}(\rho)$. At each step $k$, leader updates $\pi_{x_k}$ with one-step gradient of $V_l^{\pi_x,\pi_{y_k}}(\rho)$ while the follower updates $\pi_{y_k}$ with one-step gradient of $V_f^{\pi_{x_k},\pi_y}(\rho)$. We test all algorithms across 10 randomly generated MDPs.

We report the results in Figure 1. In the right figure, we can see the follower's optimality gap diminishes to zero, that is, the followers have found their optimal policies. In the meantime, the left figure reports the leaders' total rewards for the three methods. Overall, we find that both PBRL with value penalty and Bellman penalty outperform the independent gradient: it can be observed from Figure 1 (left) that PBRL achieves a higher leader's return than the independent gradient, and the PBRL with value penalty reaches the highest value.

### 6.2 Deep reinforcement learning from human feedback

We test our algorithm in RLHF, following the experiment setting in (Christiano et al., 2017); see a description of the general RLHF setting in Section 2.2.

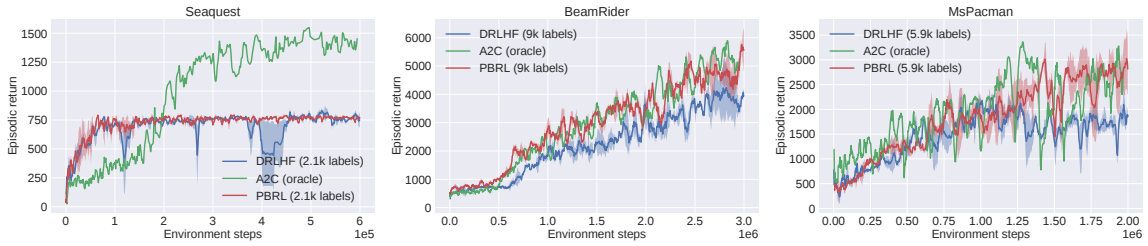


Figure 2: Performance on Atari games measured by true reward. The 'episode return' is the sum of true rewards in an episode. We average the episode returns in 5 consecutive episodes. The 'environment steps are the number of steps taken per worker in policy optimization. We compare the performance of PBRL (ours) and DRLHF both with few labeled pairs, and A2C with true reward.

**Environment and preference collection.** We conduct our experiments in the Arcade Learning Environment (ALE) (Bellemare et al., 2013) through OpenAI gym. The ALE provides the game designer's reward that can be treated as the ground truth reward. For each pair of segments we collect, we assign preference to whichever has the highest ground truth reward. This preference generation process allows us to benchmark our algorithm with DRLHF which also uses this process.

**Baseline.** We compare PBRL with DRLHF (Christiano et al., 2017) and A2C (A3C (Mnih et al., 2016) but synchronous). We use the ground truth reward to train A2C agent, and treat A2C as an oracle algorithm. The oracle algorithm estimates a performance upper bound for other algorithms.

The results are reported in Figure 2. The first two games (Seaquest and BeamRider) are also reported in (Christiano et al., 2017). For Seaquest, the asymptotic performance of DRLHF and PBRL are similar, while DRLHF is more unstable in training. Similar observations can also be made in the original paper of DRLHF. For BeamRider and MsPacman, we find out that PBRL has an advantage over DRLHF on the episode return. It can be observed that PBRL is able to achieve higher best-episode-return than DRLHF, and become comparable to the oracle algorithm.

### 6.3 Incentive design

Here we test our algorithm in the following incentive design problem:

$$\min_{x,\pi} f(\pi) = -\mathbb{E}_{\pi,\mathcal{P}_{id}}\left[\sum_{t=0}^{\infty}\gamma^t r_{\mathrm{id}}(s_t,a_t)\right] \quad \text{s.t. } \pi \in NE(x). \tag{6.2}$$

See a detailed description of this task in the motivating example of Section 5.1. We have $|\mathcal{S}| = 10$, $|\mathcal{A}_1| = |\mathcal{A}_2| = 5$. The designer's transition $\mathcal{P}_{id}(\cdot|s,a)$ and the lower-level transition $\mathcal{P}(\cdot|s,a)$ are randomly generated. Then players' original reward $r(s,a)$ and the designer's reward $r_{id}(s,a)$ are randomly generated between $[0,1]$. The players' reward $r_x(s,a) = r(s,a) + 0.2\sigma(x(s,a))$, where $r$ is the original reward given by the environment, and $\sigma(x(s,a))$ is the incentive reward from the designer. Here $\sigma$ is the sigmoid function and $x \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}_1|\times|\mathcal{A}_2|}$ is the incentive reward parameter. The $\pi_1, \pi_2$ are softmax policies.

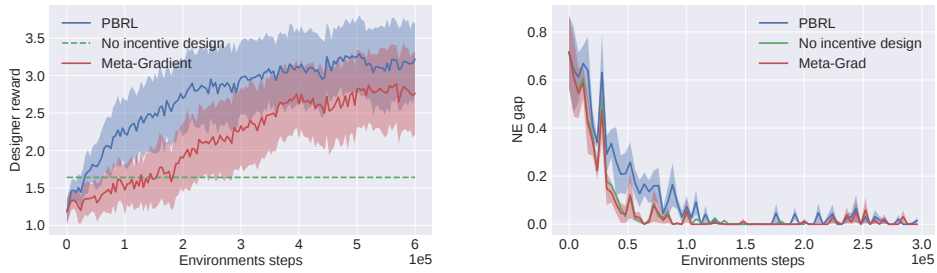


Figure 3: Incentive design. The result is generated by running the algorithms in 5 random environments. The environment step is the total number of steps taken, and is proportional to the total samples used. The designer's reward is $f(\pi)$, which is the expected cumulative designer reward. The NE gap is the estimated value of NI function $\psi(x, \pi)$. A zero NE gap indicates the players have achieved an approximate Nash equilibrium under the $r_x(s, a)$.

**Baseline.** We implement the PBRL update for the zero-sum lower level introduced in Section 5, and compare it with the Meta-Gradient method (Yang et al., 2021). To exclude the case where the original zero-sum game (with no incentive reward) already has a high reward, we also provide the performance when there is no incentive design, i.e., when $\sigma(x(s, a))$ is kept a constant. This will only return an approximate NE of the lower-level zero-sum problem without incentive reward, and therefore provide a baseline. Then if an algorithm's output incentive reward is more effective, the more it improves over the baseline.

It can be observed from Figure 3 (right) that both PBRL and Meta-Gradient have found the approximate NE under their respective incentive reward $r_x$. It can be observed from Figure 3 (left) that the incentive reward $r_x$ of both methods is effective since the designer's reward $f(\pi)$ of both methods exceeds the start line (green). In addition, PBRL is able to outperform Meta-Gradient since the incentive reward $r_x$ of PBRL can guide a $\pi \in NE(x)$ with a higher designer reward.

## 7 Conclusion

In this paper, we propose a penalty-based first-order algorithm for the bilevel reinforcement learning problems. In developing the algorithm, we provide results in three aspects: 1) we find penalty function with proper landscape properties such that the induced penalty reformulation admits solutions for the original bilevel RL problem; 2) to develop a gradient-based method, we study the differentiability of the penalty functions and find out their close form gradients; 3) based on the previous findings, we propose the convergent PBRL algorithm and evaluate on the Stackelberg Markov game, RLHF and incentive design.

## Acknowledgments and Disclosure of Funding

The work was supported by the National Science Foundation (NSF) MoDL-SCALE project 2401297, NSF CAREER 2532349, NSF DMS project 2413243, and Cisco Research Award.

# Table of Contents



## Appendix A. Preliminary results

**Lemma 17 (Lipschitz continuous optimal policy)** *Given $x \in \mathcal{X}$, consider the optimal policies in a convex policy class $\Pi$ of a parameterized MDP $\mathcal{M}_\tau(x)$. Suppose Assumption 2*

*holds, $\tau > 0$ and $\mathcal{X}$ is compact. Then the optimal policy $\pi_y^*(x)$ is unique and the following inequality hold:*

$$\|\pi_y^*(x) - \pi_y^*(x')\| \leq \tau^{-1} C_J \|x - x'\|, \ \forall x, x' \in \mathcal{X} \tag{A.1}$$

*where $C_J$ is a constant specified in the proof.*

**Proof** By Lemma 4, the optimal policy of $\mathcal{M}_\tau(x)$ on a convex policy class $\Pi$ is unique, given by

$$\pi_y^*(q(x)) = \operatorname*{argmin}_{\pi \in \Pi} J(q(x), \pi) \coloneqq \mathbb{E}_{s\sim\rho}[\langle \pi(s), \, q_s(x)\rangle + \tau h_s(\pi(s))] \tag{A.2}$$

where recall $q(x) = (q_s(x))_{s\in\mathcal{S}}$ with

$$q_s(x) = (-\max_{\pi\in\Pi} Q^\pi_{\mathcal{M}_\tau(x)}(s,a))_{a\in\mathcal{A}}. \tag{A.3}$$

We overload the notation $\pi^*$ here with $\pi_y^*(q(x))$ which equals $\pi_y^*(x)$. In (A.2), since $\tau\mathbb{E}_{s\sim\rho}[h_s(\pi(s))]$ is $\tau$-strongly convex at $\pi$ on $\Pi$, $\pi_y^*(q(x))$ satisfies (A.2) if and only if it is a solution of the following parameterized variational inequality (VI)

$$\langle \nabla_\pi J(q(x), \pi), \, \pi - \pi' \rangle \leq 0, \ \forall \pi' \in \Pi \tag{A.4}$$

where

$$\nabla_\pi J(q(x), \pi) = \Big(\rho(s) q_s(x) + \tau\rho(s)\nabla h_s(\pi(s))\Big)_{s\in\mathcal{S}}. \tag{A.5}$$

First, it can be checked that $\nabla_\pi J(q(x), \pi)$ is continuously differentiable at any $(q(x), \pi)$. Secondly, by the uniform strong convexity of $J(q(x), \cdot)$, given any $q(x)$, it holds that

$$(\pi - \pi')^\top \nabla_\pi^2 J\big(q(x), \pi_y^*(q(x))\big)(\pi - \pi') \geq \tau^{-1}\|\pi - \pi'\|^2. \tag{A.6}$$

Given these two properties of the VI, it then follows from (Dontchev and Rockafellar, 2009, Theorem 2F.7) that the solution mapping $\pi_y^*(q(x))$ is $\tau^{-1}$-Lipschitz-continuous locally at any point $q(x)$. Thus $\pi_y^*(q(x))$ is $\tau^{-1}$-Lipschitz-continuous in $q(x)$ globally, yielding

$$\begin{aligned} \|\pi_y^*(q(x)) - \pi_y^*(q(x'))\| &\leq \tau^{-1}\|q(x) - q(x')\| \\ &\leq \tau^{-1} \max_{x\in\mathcal{X}} \|\nabla q(x)\| \|x - x'\| \\ &= \tau^{-1} C_J \|x - x'\| \end{aligned} \tag{A.7}$$

where the second inequality follows from $q(x)$ is continuously differentiable, which can be checked by Lemma 8 under Assumption 2 and the continuity of $\pi_y^*(x)$ we proved earlier; and, $C_J = \max_{x\in\mathcal{X}} \|\nabla q(x)\|$ is well-defined by compactness of $\mathcal{X}$. ■

## Appendix B. Proof in Section 2 and 3

### B.1 Proof that Stackelberg Markov game is a bilevel RL problem

**Lemma 18 (Stackelberg game cast as $\mathcal{BM}$)** *The Stackelberg MDP from the follower's viewpoint can be defined as a parametric MDP:*

$$\begin{aligned}\mathcal{M}_\tau(x) =&\{\mathcal{S}, \mathcal{A}_f, r_x(s,a_f) = \mathbb{E}_{a_l\sim\pi_x(s)}[r_l(s,a_l,a_f)],\\ &\mathcal{P}_x(\cdot|s,a_f) = \mathbb{E}_{a_l\sim\pi_x(s)}[\mathcal{P}(\cdot|s,a_l,a_f)], \tau h_f\}.\end{aligned}$$

*Then we have $V_f^{\pi_x,\pi_y}(s) = V_{\mathcal{M}_\tau(x)}^{\pi_y}(s), \forall s$, and thus the original formulation of Stackelberg game in* (2.4) *can be rewritten as $\mathcal{BM}$:*

$$\mathcal{SG}: \min_{x,y} -V_l^{\pi_x,\pi_y}(\rho),\ \text{s.t. } x\in\mathcal{X},\ y\in\mathcal{Y}^*(x) = \operatorname*{argmin}_{y\in\mathcal{Y}} -V_{\mathcal{M}_\tau(x)}^{\pi_y}(\rho). \tag{B.1}$$

**Proof** Recall that the follower's value function $V_f^{\pi_x,\pi_y}(s)$ under the leader's policy $\pi_x$ and the follower's policy $\pi_y$ is defined as

$$V_f^{\pi_x,\pi_y}(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t\big(r_f(s_t,a_{l,t},a_{f,t}) - \tau h_{f,s_t}(\pi_y(s_t))\big)\big|s_0=s,\pi_x,\pi_y\Big] \tag{B.2}$$

where the leader's action $a_{l,t}\sim\pi_l(s_t)$, the follower's action $a_{f,t}\sim\pi_f(s_t)$, and the state transition follows $s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_{l,t},a_{f,t})$.
It then follows from an expansion of the expectation in (B.2) that

$$\begin{aligned}V_f^{\pi_x,\pi_y}(s) =& \mathbb{E}_{a_{l,0}\sim\pi_x(s_0),a_{f,0}\sim\pi_y(s_0)}\Big[r_f(s_0,a_{l,0},a_{f,0}) - \tau h_{f,s_0}(\pi_y(s_0))\big|s_0=s,\pi_x,\pi_y\Big]\\ &+\gamma\mathbb{E}_{\substack{a_{l,0}\sim\pi_x(s_0),a_{f,0}\sim\pi_y(s_0)\\ s_1\sim\mathcal{P}(s_0,a_{l,0},a_{f,0})\\ a_{l,1}\sim\pi_x(s_1),a_{f,1}\sim\pi_y(s_1)}}\Big[r_f(s_1,a_{l,1},a_{f,1}) - \tau h_{f,s_1}(\pi_y(s_1))\big|s_0=s,\pi_x,\pi_y\Big]\\ &+\dots\\ =&\mathbb{E}_{a_{f,0}\sim\pi_y(s_0)}\Big[r_x(s_0,a_{f,0}) - \tau h_{f,s_0}(\pi_y(s_0))\big|s_0=s,\pi_y\Big]\\ &+\gamma\mathbb{E}_{\substack{a_{f,0}\sim\pi_y(s_0)\\ s_1\sim\mathcal{P}_x(s_0,a_{f,0})\\ a_{f,1}\sim\pi_y(s_1)}}\Big[r_x(s_1,a_{f,1}) - \tau h_{f,s_1}(\pi_y(s_1))\big|s_0=s,\pi_y\Big]+\dots\\ =&V_{\mathcal{M}_\tau(x)}^{\pi_y}(s)\end{aligned}$$

where recall $\mathcal{P}_x(s,a_f) = \mathbb{E}_{a_l\sim\pi_x(s)}[\mathcal{P}(\cdot|s,a_l,a_f)]$ and $r_x(s,a_f) = \mathbb{E}_{a_l\sim\pi_x(s)}[r_l(s,a_l,a_f)]$. Thus we have $V_f^{\pi_x,\pi_y}(s) = V_{\mathcal{M}_\tau(x)}^{\pi_y}(s), \forall s$. Therefore, the Stackelberg Markov game can be written as $\mathcal{BM}$. ■

### B.2 Proof of Lemma 1

**Proof** Since $(x_\lambda, y_\lambda)$ is an $\epsilon$-minima of $\mathcal{BM}_{\lambda p}$, it holds for any $x\in\mathcal{X}$ and $y\in\mathcal{Y}$ that

$$\begin{aligned}&f(x_\lambda,y_\lambda) + \lambda\big(-V_{\mathcal{M}_\tau(x_\lambda)}^{\pi_{y_\lambda}}(\rho) + \max_{y\in\mathcal{Y}} V_{\mathcal{M}_\tau(x_\lambda)}^{\pi_y}\big)\\ &\le f(x,y) + \lambda\big(-V_{\mathcal{M}_\tau(x)}^{\pi_y}(\rho) + \max_{y\in\mathcal{Y}} V_{\mathcal{M}_\tau(x)}^{\pi_y}\big) + \epsilon.\end{aligned} \tag{B.3}$$

Choosing $x = x_\lambda$ and $y \in \mathcal{Y}(x_\lambda)$ in the above inequality and rearranging yields

$$\begin{aligned}\max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)}(\rho) - V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho) &\le \frac{1}{\lambda}\big(f(x_\lambda, y_\lambda) - f(x_\lambda, y) + \epsilon\big) \\ &\le \frac{1}{\lambda}\big(C + \epsilon\big) \le \delta + \lambda^{-1}\epsilon. \end{aligned} \tag{B.4}$$

Define $\epsilon_\lambda \coloneqq \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)}(\rho) - V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho)$ then $\epsilon_\lambda \le \delta + \lambda^{-1}\epsilon$. It follows from (B.3) that for any $x, y$ feasible for (3.4) that

$$\begin{aligned} f(x_\lambda, y_\lambda) &\le f(x, y) + \lambda\big(-V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)} - \epsilon_\lambda\big) + \epsilon \\ &\le f(x, y) + \epsilon. \end{aligned} \tag{B.5}$$

This completes the proof. ■

### B.3 Proof of Lemma 2

**Proof** The following proof holds for any $x$ and thus we omit $x$ in the notations $\mathcal{M}_\tau(x)$, $\pi^*_y(x)$ and $\mathcal{P}_x$ in this proof. We first prove a policy gradient theorem for the regularized MDP. From the Bellman equation, we have

$$V^\pi_{\mathcal{M}_\tau}(s) = \sum_a \pi(a|s) Q^\pi_{\mathcal{M}_\tau}(s, a) - \tau h_s(\pi(s)) \tag{B.6}$$

Differentiating two sides of the equation with respect to $\pi$ gives

$$\nabla V^\pi_{\mathcal{M}_\tau}(s) = \sum_a \nabla\pi(a|s) Q^\pi_{\mathcal{M}_\tau}(s, a) + \sum_a \pi(a|s)\nabla Q^\pi_{\mathcal{M}_\tau}(s, a) - \tau\nabla_\pi h_s(\pi(s)). \tag{B.7}$$

By the definition of $Q$ function, we have $\nabla Q^\pi_{\mathcal{M}_\tau}(s, a) = \sum_{s'} \mathcal{P}(s'|s, a)\nabla V^\pi_{\mathcal{M}_\tau}(s')$. Substituting this inequality into (B.7) yields

$$\begin{aligned} &\nabla V^\pi_{\mathcal{M}_\tau}(s) \\ &= \sum_a \nabla\pi(a|s) Q^\pi_{\mathcal{M}_\tau}(s, a) + \sum_{s'} P_\pi(s_1 = s'|s_0 = s)\nabla V^\pi_{\mathcal{M}_\tau}(s, a) - \tau\nabla_\pi h_s(\pi(s)) \end{aligned} \tag{B.8}$$

where $P^\pi(s_1 = s'|s_0 = s)$ is the probability of $s_1 = s'$ given $s_0 = s$ under policy $\pi$. Note that the above inequality has a recursive structure, thus we can repeatedly applying it to itself and obtain

$$\nabla V^\pi_{\mathcal{M}_\tau}(s) = \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^\pi_s}[\sum_a Q^\pi_{\mathcal{M}_\tau}(\bar{s}, a)\nabla\pi(a|\bar{s})] + \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^\pi_s}[-\nabla_\pi h_{\bar{s}}(\pi(\bar{s}))] \tag{B.9}$$

where $d^\pi_s(\bar{s}) \coloneqq (1-\gamma)\sum_t \gamma^t P^\pi(s_t = \bar{s}|s_0 = s)$ is the discounted visitation distribution. Define $d^\pi_{\mathcal{M}_\tau}(\bar{s}) \coloneqq \mathbb{E}_{s\sim\rho}[d^\pi_s(\bar{s})]$. Since $\nabla\pi(a|\bar{s}) = 1_{\bar{s},a}$ where $1_{\bar{s},a}$ is the indicator vector, we have the regularized policy gradient given by

$$\nabla_\pi V^\pi_{\mathcal{M}_\tau}(\rho) = \frac{1}{1-\gamma}\big[d^\pi_{\mathcal{M}_\tau}(s)\big(Q^\pi_{\mathcal{M}_\tau}(s, \cdot) - \tau\nabla h_s(\pi(s))\big)\big]_{s\in\mathcal{S}}. \tag{B.10}$$

Now we begin the prove the lemma. By the performance difference lemma (see e.g., (Lan, 2023, Lemma 2) and (Zhan et al., 2023, Lemma 5)), for any $\pi \in \Pi$, we have

$$\begin{aligned}
&\max_{\tilde{\pi}\in\Pi} V^{\tilde{\pi}}_{\mathcal{M}_\tau}(\rho) - V^{\pi}_{\mathcal{M}_\tau}(\rho) \\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi^*}_{\mathcal{M}_\tau}}\Big[\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi^*_y(s) - \pi(s)\rangle - \tau h_s(\pi^*_y(s)) + \tau h_s(\pi(s))\Big] \\
&\le \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi^*}_{\mathcal{M}_\tau}}\Big[\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi^*_y(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi^*_y(s) - \pi(s)\rangle\Big]
\end{aligned}$$

where the inequality follows from the convexity of $h_s$. Continuing from the inequality, it follows

$$\begin{aligned}
&(1-\gamma)\big(\max_{\tilde{\pi}\in\Pi} V^{\tilde{\pi}}_{\mathcal{M}_\tau}(\rho) - V^{\pi}_{\mathcal{M}_\tau}(\rho)\big) \\
&\le \mathbb{E}_{s\sim d^{\pi^*}_{\mathcal{M}_\tau}}\Big[\max_{\pi'\in\Pi}\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi'(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi'(s) - \pi(s)\rangle\Big] \\
&= \mathbb{E}_{s\sim d^{\pi}_{\mathcal{M}_\tau}}\left[\frac{d^{\pi^*}_{\mathcal{M}_\tau}(s)}{d^{\pi}_{\mathcal{M}_\tau}(s)}\max_{\pi'\in\Pi}\Big(\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi'(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi'(s) - \pi(s)\rangle\Big)\right] \\
&\le \mathbb{E}_{s\sim d^{\pi}_{\mathcal{M}_\tau}}\left[\Big\|\frac{d^{\pi^*}_{\mathcal{M}_\tau}}{d^{\pi}_{\mathcal{M}_\tau}}\Big\|_\infty \max_{\pi'\in\Pi}\Big(\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi'(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi'(s) - \pi(s)\rangle\Big)\right]
\end{aligned} \tag{B.11}$$

where the last inequality follows from $\frac{d^{\pi^*}_{\mathcal{M}_\tau}(s)}{d^{\pi}_{\mathcal{M}_\tau}(s)} \le \Big\|\frac{d^{\pi^*}_{\mathcal{M}_\tau}}{d^{\pi}_{\mathcal{M}_\tau}}\Big\|_\infty$ and

$$\begin{aligned}
&\max_{\pi'\in\Pi}\Big(\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi'(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi'(s) - \pi(s)\rangle\Big) \\
&\ge \langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot),\, \pi(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)),\, \pi(s) - \pi(s)\rangle = 0.
\end{aligned} \tag{B.12}$$

Continuing from (B.11), we have

$$\begin{aligned}
&(1-\gamma)\big(\max_{\tilde{\pi}\in\Pi} V^{\tilde{\pi}}_{\mathcal{M}_\tau}(\rho) - V^{\pi}_{\mathcal{M}_\tau}(\rho)\big) \\
&\le \frac{1}{(1-\gamma)\min_s \rho(s)}\max_{\pi'\in\Pi}\mathbb{E}_{d^{\pi}_{\mathcal{M}_\tau}}\Big[\Big(\langle Q^{\pi}_{\mathcal{M}_\tau}(s,\cdot), \pi'(s) - \pi(s)\rangle - \tau\langle \nabla h_s(\pi(s)), \pi'(s) - \pi(s)\rangle\Big)\Big] \\
&= \frac{1}{\min_s \rho(s)}\max_{\pi'\in\Pi}\langle \nabla_\pi V^{\pi}_{\mathcal{M}_\tau}(\rho),\, \pi' - \pi\rangle
\end{aligned} \tag{B.13}$$

where the inequality follows from $(1-\gamma)\rho(s) \le d^{\pi}_{\mathcal{M}_\tau}(s) \le 1$ for any $s$ and $\pi$, and the equality follows from (B.10). This proves the result. ■

### B.4 Proof of Lemma 3

**Proof** Given $x_\lambda$, point $y_\lambda$ satisfies the first-order stationary condition:

$$\langle \nabla_y f(x_\lambda, y_\lambda) + \lambda \nabla_y p(x_\lambda, y_\lambda),\, y_\lambda - y'\rangle \le 0,\ \forall y' \in \mathcal{Y} \tag{B.14}$$

which leads to

$$\begin{aligned}\langle \nabla_y p(x_\lambda, y_\lambda),\, y_\lambda - y' \rangle &\le -\frac{1}{\lambda}\langle \nabla_y f(x_\lambda, y_\lambda),\, y_\lambda - y' \rangle \\ &\le \frac{L\|y_\lambda - y'\|}{\lambda} \le \frac{LC_u}{\lambda},\ \forall y' \in \mathcal{Y} \end{aligned} \tag{B.15}$$

where $C_u \coloneqq \max_{y,y' \in \mathcal{Y}} \|y - y'\|$ which is well defined by compactness of $\mathcal{Y}$. For the LHS of the above inequality, we have the following inequality hold

$$\begin{aligned}\min_{y' \in \mathcal{Y}} \langle \nabla_y p(x_\lambda, y_\lambda),\, y_\lambda - y' \rangle &= \max_{y' \in \mathcal{Y}} \langle \nabla_y V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho),\, y' - y_\lambda \rangle \\ &\ge \frac{1}{(1-\gamma)\min_s \rho(s)} \big( \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)}(\rho) - V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho) \big) \end{aligned} \tag{B.16}$$

where the last inequality follows from we are using direct policy parameterization $y = \pi$ and Lemma 2.
Substituting (B.16) into (B.15) yields

$$\max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)}(\rho) - V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho) \le \frac{LC_u}{\lambda}. \tag{B.17}$$

Define $\epsilon_\lambda \coloneqq -V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho) + \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)}(\rho)$ then $\epsilon_\lambda \le \delta$ by choice of $\lambda$.
By local optimality of $(x_\lambda, y_\lambda)$, it holds for any $x \in \mathcal{X}, y \in \mathcal{Y}$ and in the neighborhood of $(x_\lambda, y_\lambda)$ that

$$\begin{aligned} & f(x_\lambda, y_\lambda) + \lambda\big( - V^{\pi_{y_\lambda}}_{\mathcal{M}_\tau(x_\lambda)}(\rho) + \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x_\lambda)} \big) \\ & \le f(x, y) + \lambda\big( - V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)} \big). \end{aligned} \tag{B.18}$$

From the above inequality, it holds for any $(x, y)$ feasible for the relaxed $\mathcal{BM}$ in (3.4) and in neighborhood of $(x_\lambda, y_\lambda)$ that

$$\begin{aligned} f(x_\lambda, y_\lambda) &\le f(x, y) + \lambda\big( - V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y \in \mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)} - \epsilon_\lambda \big) \\ &\le f(x, y) \end{aligned} \tag{B.19}$$

which proves the result. ■

### B.5 Proof of Lemma 4

**Proof** We start with the first bullet. Define

$$V^*_{\mathcal{M}_\tau(x)}(s) \coloneqq \max_{\pi \in \Pi} V^\pi_{\mathcal{M}_\tau(x)}(s),\ Q^*_{\mathcal{M}_\tau(x)}(s,a) \coloneqq r(s,a) + \gamma \mathbb{E}_{s' \sim \mathcal{P}_x(s,a)}[V^*_{\mathcal{M}_\tau(x)}(s')].$$

Then it follows from the definition of the value function that for any $s_0$,

$$\begin{aligned}
&V^*_{\mathcal{M}_\tau(x)}(s_0)\\
&= \max_{\pi\in\Pi} \mathbb{E}\Big[r_x(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \sum_{t=1}^{\infty}\gamma^t\big(r_x(s_t,a_t) - \tau h_{s_t}(\pi(s_t))\big)\big|s_0,\pi\Big]\\
&= \max_{\pi\in\Pi} \mathbb{E}\Big[r_x(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \mathbb{E}\big[\sum_{t=1}^{\infty}\gamma^t\big(r_x(s_t,a_t) - \tau h_{s_t}(\pi(s_t))\big)\big|s_0,a_0,\pi\big]\big|s_0,\pi\Big]
\end{aligned}$$

where the last equality follows from the law of total expectation. Continuing from above, we have

$$\begin{aligned}
V^*_{\mathcal{M}_\tau(x)}(s_0) &= \max_{\pi\in\Pi} \mathbb{E}_{a_0\sim\pi(s_0)}\Big[r_x(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0)}\big[V^{\pi}_{\mathcal{M}_\tau(x)}(s_1)\big]\Big]\\
&\le \max_{\pi\in\Pi} \mathbb{E}_{a_0\sim\pi(s_0)}\Big[r(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0)}\big[V^{*}_{\mathcal{M}_\tau(x)}(s_1)\big]\Big] \quad \text{(B.20)}
\end{aligned}$$

Given $x$, define a policy $\pi^*_y = (\pi^*_y(s))_{s\in\mathcal{S}} \in \Pi$ via

$$\pi^*_y(s_0) \coloneqq \operatorname*{argmax}_{\pi(s_0)} \mathbb{E}_{a_0\sim\pi(s_0)}\Big[r(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0)}\big[V^{*}_{\mathcal{M}_\tau(x)}(s_1)\big]\Big]$$

for any $s_0 \in \mathcal{S}$, where the argmax is a singleton following from the $\tau$-strong convexity of $\tau h$, and we sometimes treat the singleton set as its element for convenience. Given the definition of $\pi^*_y$, it then follows from (B.20) that

$$\begin{aligned}
&V^*_{\mathcal{M}_\tau(x)}(s_0)\\
&\le \mathbb{E}_{a_0\sim\pi^*_y(s_0)}\Big[r(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0)}\big[V^{*}_{\mathcal{M}_\tau(x)}(s_1)\big]\Big]\\
&\le \mathbb{E}_{a_0\sim\pi^*_y(s_0)}\Big[r(s_0,a_0) - \tau h_{s_0}(\pi(s_0))\\
&+ \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0),a_1\sim\pi^*_y(s_1)}\big[r(s_1,a_1) - \tau h_{s_1}(\pi(s_1)) + \gamma\mathbb{E}_{s_2\sim\mathcal{P}_x(s_1,a_1)}[V^{*}_{\mathcal{M}_\tau(x)}(s_2)]\big]\Big] \quad \text{(B.21)}
\end{aligned}$$

where the last inequality is a result of applying (B.20) twice. Continuing to recursively apply (B.20) and then using the definition of $V^{\pi}_{\mathcal{M}_\tau(x)}$ in (2.1) yield

$$V^*_{\mathcal{M}_\tau(x)}(s_0) \le V^{\pi^*_y}_{\mathcal{M}_\tau(x)}(s_0),\ \forall s_0 \in \mathcal{S} \quad \text{(B.22)}$$

which proves $\pi^*_y$ is the optimal policy for $\mathcal{M}_\tau(x)$. In addition, we have

$$\begin{aligned}
\pi^*_y(s_0) &= \operatorname*{argmax}_{\pi(s_0)} \mathbb{E}_{a_0\sim\pi(s_0)}\Big[r(s_0,a_0) - \tau h_{s_0}(\pi(s_0)) + \gamma\mathbb{E}_{s_1\sim\mathcal{P}_x(s_0,a_0)}\big[V^{\pi^*_y}_{\mathcal{M}_\tau(x)}(s_1)\big]\Big]\\
&= \operatorname*{argmax}_{\pi(s_0)} \mathbb{E}_{a_0\sim\pi(s_0)}\Big[Q^{\pi^*_y}_{\mathcal{M}_\tau(x)}(s_0,a_0) - \tau h_{s_0}(\pi(s_0))\Big],\ \forall s_0. \quad \text{(B.23)}
\end{aligned}$$

Then we have $\pi^*_y = \arg\min_{y\in\Pi} g(x,y)$ and thus $\arg\min_{y\in\Pi} g(x,y) \in \mathcal{Y}^*(x)$. To further prove $\arg\min_{y\in\Delta(\mathcal{A})^{|\mathcal{S}|}} g(x,y) = \mathcal{Y}^*(x)$, it then suffices to prove any other policy $\pi \in \Pi$ different

from $\pi_y^*$ is not optimal. Let $s_0'$ be the state such that $\pi_y^*(s_0') \neq \pi(s_0')$. We have

$$\begin{aligned} V^{\pi}_{\mathcal{M}_\tau(x)}(s_0') &\leq \mathbb{E}_{a_0\sim\pi(s_0')}\Big[r(s_0', a_0) - \tau h_{s_0'}(\pi(s_0')) + \gamma \mathbb{E}_{s_1\sim\mathcal{P}_x(s_0', a_0)}\big[V^*_{\mathcal{M}_\tau(x)}(s_1)\big]\Big] \\ &< \mathbb{E}_{a_0\sim\pi_y^*(s_0')}\Big[r(s_0', a_0) - \tau h_{s_0'}(\pi_y^*(s_0')) + \gamma \mathbb{E}_{s_1\sim\mathcal{P}_x(s_0', a_0)}\big[V^*_{\mathcal{M}_\tau(x)}(s_1)\big]\Big] \\ &= V^*_{\mathcal{M}_\tau(x)}(s_0') \end{aligned} \tag{B.24}$$

where the last inequality follows from the strong convexity of $h$ and the definition of $\pi_y^*$; and the last equality follows from $\pi_y^*$ is the optimal policy. This proves the result.
Next we prove the second bullet. We have

$$\|y_\epsilon - \pi_y^*\|^2 \leq \tau^{-1}\big(g(x, y_\epsilon) - v(x)\big) \leq \tau^{-1}\epsilon \tag{B.25}$$

where the first inequality follows from $\tau$-strong-convexity of $g(x, \cdot)$. Next we prove $|f^* - f_\epsilon^*| \leq L\tau^{-1}\epsilon$. Let $f_\epsilon^* = f(x_\epsilon^*, y_\epsilon^*)$. We have

$$f(x_\epsilon^*, \mathcal{Y}(x_\epsilon^*)) - f(x_\epsilon^*, y_\epsilon^*) \leq L\|y_\epsilon^* - \mathcal{Y}(x_\epsilon^*)\| \leq L\sqrt{\tau^{-1}\epsilon} \tag{B.26}$$

where the last inequality follows from (B.25). The result follows from the fact that $f(x_\epsilon^*, \mathcal{Y}(x_\epsilon^*)) \geq f^*$ and $f(x_\epsilon^*, y_\epsilon^*) \leq f^*$. ■

## Appendix C. Proof in Section 4

### C.1 Proof of Lemma 6

We first introduce a generalized Danskin's theorem as follows.

**Lemma 19 (Generalized Danskin's Theorem (Clarke, 1975))** *Let $\mathcal{F}$ be a compact set and let a continuous function $\ell : \mathbb{R}^d \times \mathcal{F} \mapsto \mathbb{R}$ satisfy: 1) $\nabla_x \ell(x, y)$ is continuous in $(x, y)$; and 2) given any $x$, for any $y, y' \in \operatorname{argmax}_{y\in\mathcal{F}} \ell(x, y)$, $\nabla_x \ell(x, y) = \nabla_x \ell(x, y')$. Then let $h(x) := \max_{y\in\mathcal{F}} \ell(x, y)$, we have $\nabla h(x) = \nabla_x \ell(x, y^*)$ for any $y^* \in \operatorname{argmax}_{y\in\mathcal{F}} \ell(x, y)$.*

Lemma 19 first follows (Clarke, 1975, Theorem 2.1) where conditions (a)–(d) are guaranteed by Lemma 19's condition 1). Then by (Clarke, 1975, Theorem 2.1 (4)) that we have the Clarke's generalized gradient set of $h(x) = \max_{y\in\mathcal{F}} \ell(x, y)$ is the convex hull of $\{\nabla_x \ell(x, y), y \in \operatorname{argmax}_{y\in\mathcal{F}} \ell(x, y)\}$. It then follows from Lemma 19's condition 2) that this generalized gradient set is a singleton $\{\nabla_x \ell(x, y^*)\}$ with any $y^* \in \operatorname{argmax}_{y\in\mathcal{F}} \ell(x, y)$. Finally it follows from (Clarke, 1975, Proposition 1.13) that $h(x)$ is differentiable with gradient $\nabla_x \ell(x, y^*)$.
Now to prove Lemma 6, it suffices to prove

$$\nabla \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) = \nabla_x V^{\pi_{y^*}}_{\mathcal{M}_\tau(x)}(\rho)|_{y^*\in\mathcal{Y}^*(x)}$$

. This argument is true following from Assumption 1 and the generalized Danskin's theorem above, with $\ell(x, y) = V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$.

### C.2 Proof of Lemma 7

**Proof (a).** Under the assumptions in (a), Lemma 6 holds. It then follows from

$$\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) = \mathbb{E}\big[\sum_{t=0}^{\infty}\gamma^t \nabla r_x(s_t,a_t)|s_0\sim\rho,\pi_y\big] \tag{C.1}$$

that the result holds.

**(b).** Given the follower's policy $\pi_y$, define the Stackelberg MDP from the leader's view as

$$\begin{aligned}\mathcal{M}(\pi_y) =&\{\mathcal{S},\mathcal{A}_l, r_{\pi_y}(s,a_l)=\mathbb{E}_{a_f\sim\pi_y(s)}[r_f(s,a_f,a_l)]-\tau h_{f,s}(\pi_y(s)),\\ &\mathcal{P}_{\pi_y}(\cdot|s,a_l)=\mathbb{E}_{a_f\sim\pi_y(s)}[\mathcal{P}(\cdot|s,a_l,a_f)]\}\end{aligned} \tag{C.2}$$

Note $\mathcal{M}(\pi_y)$ does not include a regularization for its policy $\pi_x$. By Lemma 18, we have the follower's value function $V_f^{\pi_x,\pi_y}(s)$ can be rewritten from the viewpoint that $\pi_y$ is the main policy and $\pi_x$ is part of the follower's MDP, that is, $V_f^{\pi_x,\pi_y}(s) = V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$. It can be proven similarly that $V_f^{\pi_x,\pi_y}(s) = V^{\pi_x}_{\mathcal{M}(\pi_y)}(s)$. Therefore, we have $V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) = V^{\pi_x}_{\mathcal{M}(\pi_y)}(s)$ and

$$\begin{aligned}\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) &= \nabla_x V^{\pi_x}_{\mathcal{M}(\pi_y)}(s)\\ &= \mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q^{\pi_x}_{\mathcal{M}(\pi_y)}(s_t,a_{l,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x\Big]\end{aligned} \tag{C.3}$$

where the last equality follows from the policy gradient theorem (Sutton et al., 2000). We have

$$\begin{aligned}&Q^{\pi_x}_{\mathcal{M}(\pi_y)}(s,a_l)\\ &= r_{\pi_y}(s,a_l)+\gamma\mathbb{E}_{s'\sim\mathcal{P}_{\pi_y}(s,a_l)}[V^{\pi_x}_{\mathcal{M}(\pi_y)}(s')]\\ &= \mathbb{E}_{a_f\sim\pi_y(s)}[r_f(s,a_f,a_l)]-\tau h_{f,s}(\pi_y(s))+\gamma\mathbb{E}_{s'\sim\mathcal{P}(s,a_l,a_f),a_f\sim\pi_y(s)}[V_f^{\pi_x,\pi_y}(s')]\\ &= \mathbb{E}_{a_f\sim\pi_y(s)}[Q_f^{\pi_x,\pi_y}(s,a_l,a_f)]-\tau h_{f,s}(\pi_y(s))\end{aligned} \tag{C.4}$$

where the last equality follows from the definition of $Q_f^{\pi_x,\pi_y}(s,a_l,a_f)$ in Section 2.2. Substituting the above equality into (C.3) yields

$$\begin{aligned}\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) = &\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big]\\ &-\tau\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big]\end{aligned} \tag{C.5}$$

It then follows from Lemma 6 that

$$\begin{aligned}&\nabla_x p(x,y)\\ &= -\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)+\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)|_{\pi_y=\pi_y^*(x)}\\ &= -\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t\big(Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})-\tau h_{f,s_t}(\pi_y(s_t))\big)\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big]\\ &+\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t\big(Q_f^{\pi_x,\pi_y^*(x)}(s_t,a_{l,t},a_{f,t})-\tau h_{f,s_t}(\pi_y^*(x)(s_t))\big)\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y^*(x)\Big]\end{aligned}$$

where $\pi_y^*(x)$ is the follower's optimal policy given leader's policy $\pi_x$. ■

### C.3 Proof of Lemma 8

**Proof** We first consider $\nabla_x g(x,y)$. To prove $\nabla_x g(x,y)$ exist, it suffices to show $\nabla q_{s,a}(x)$ exist for any $(s,a)$. By Lemma 19, to show $q_{s,a}(x) = -\max_{\pi\in\Pi} Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)$ is differentiable, it remains to show that $\operatorname{argmax}_{\pi\in\Pi} Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)$ is a singleton. By Lemma 4, the optimal policy of $\mathcal{M}_\tau(x)$ is unique. Since the unique optimal policy $\pi_y^*(x) \in \operatorname{argmax}_{\pi\in\Pi} Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)$, it suffices to show any policy $\pi$ different from $\pi_y^*(x)$ leads to $Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a) < Q^{\pi_y^*(x)}_{\mathcal{M}_\tau(x)}(s,a)$. Next, we prove this result.

By the uniqueness of the optimal policy, the policies different from $\pi_y^*(x)$ are non-optimal, that is, for any non-optimal $\pi$, there exists state $\bar{s}$ such that $V^{\pi}_{\mathcal{M}_\tau(x)}(\bar{s}) < V^{\pi_y^*(x)}_{\mathcal{M}_\tau(x)}(\bar{s})$. By the Bellman equation, we have for any $T$,

$$\begin{aligned} Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a) = \mathbb{E}\Big[\sum_{t=0}^{T-1}\gamma^t r_x(s_t,a_t)|\pi, s_0=s, a_0=a\Big] \\ + \gamma^T \mathbb{E}_{s_T\sim P_x^{\pi}(\cdot|s_0=s,a_0=a)}[V^{\pi}_{\mathcal{M}_\tau(x)}(s_T)] \end{aligned}$$

By the irreducible Markov chain assumption, there exists $i$ such that $P_x^{\pi}(s_i=\bar{s}|s_0=s,a_0=a)>0$. Choosing $T=i$ in the above equality yields

$$\begin{aligned} &Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a) \\ &= \mathbb{E}\Big[\sum_{t=0}^{i-1}\gamma^t r_x(s_t,a_t)|\pi, s_0=s, a_0=a\Big] + \gamma^i \mathbb{E}_{s_i\sim P_x^{\pi}(\cdot|s_0=s,a_0=a)}[V^{\pi}_{\mathcal{M}_\tau(x)}(s_i)] \\ &< \mathbb{E}\Big[\sum_{t=0}^{i-1}\gamma^t r_x(s_t,a_t)|\pi, s_0=s, a_0=a\Big] + \gamma^i \mathbb{E}_{s_i\sim P_x^{\pi}(\cdot|s_0=s,a_0=a)}[V^{\pi_y^*(x)}_{\mathcal{M}_\tau(x)}(s_i)] \\ &\le Q^{\pi_y^*(x)}_{\mathcal{M}_\tau(x)}(s,a) \end{aligned} \tag{C.6}$$

where the first inequality follows from $V^{\pi}_{\mathcal{M}_\tau(x)}(\bar{s}) < V^{\pi_y^*(x)}_{\mathcal{M}_\tau(x)}(\bar{s})$ and $P_x^{\pi}(s_i=\bar{s}|s_0=s,a_0=a)>0$; and the last inequality follows from the optimality of $\pi_y^*(x)$.

Given (C.6), we can conclude that $q_{s,a}(x)$ is differentiable with the gradient

$$\nabla q_{s,a}(x) = -\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)|_{\pi=\pi_y^*(x)}. \tag{C.7}$$

Then $\nabla_x g(x,y)$ can be computed as

$$\nabla_x g(x,y) = -\mathbb{E}_{s\sim\rho,a\sim\pi_y(s)}\big[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\big]\big|_{\pi=\pi_y^*(x)}. \tag{C.8}$$

Since $g(x,\cdot)$ is smooth and strongly convex, we can use Danskins' theorem to obtain

$$\nabla v(x) = \nabla_x g(x,y)|_{y=\operatorname{argmin}_{y\in\mathcal{Y}} g(x,y)} = \nabla_x g(x,y)|_{y=\pi_y^*(x)} \tag{C.9}$$

where the last equality follows from Lemma 4. This completes the proof. ■

### C.4 Proof of Lemma 9

**Proof** We first prove the first bullet. We have

$$\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a) = \mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t \nabla_x r_x(s_t,a_t)|\pi, s_0=s, a_0=a\Big]. \tag{C.10}$$

It can be checked that Assumption 2 holds and then $\nabla_x g(x,y)$, $\nabla v(x)$ follow from Lemma 8 with (C.10).

We next prove the second bullet. By (C.5), we have

$$\begin{aligned}\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big]\\ -\tau\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,\pi_x,\pi_y\Big]\end{aligned} \tag{C.11}$$

Then

$$\begin{aligned}\nabla_x Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a_f) &= \nabla_x\big(r_x(s,a_f)+\gamma\mathbb{E}_{s'\sim\mathcal{P}_x(s,a_f)}[V^{\pi_y}_{\mathcal{M}_\tau(x)}(s')\big)\\ &= \nabla_x\big(\mathbb{E}_{a_l\sim\pi_x(s)}[r_l(s,a_l,a_f)]+\gamma\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}[V^{\pi_y}_{\mathcal{M}_\tau(x)}(s')]\big)\end{aligned} \tag{C.12}$$

where the last eqaulity follows from the definition of $\mathcal{M}_\tau(x)$ in Lemma 18. Using the log trick, we can write

$$\begin{aligned}&\nabla_x Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a_f)\\ &=\mathbb{E}_{a_l\sim\pi_x(s)}\Big[\big(r(s,a_l,a_f)+\gamma\mathbb{E}_{s'\sim\mathcal{P}(s,a_l,a_f)}[V_f^{\pi_x,\pi_y}(s')]\big)\nabla\log\pi_x(a_l|s)\Big]\\ &+\gamma\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}[\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(s')]\end{aligned} \tag{C.13}$$

Substituting (C.5) into the above equality yields

$$\begin{aligned}&\nabla_x Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a_f)\\ &=\mathbb{E}_{a_l\sim\pi_x(s)}\Big[\big(r(s,a_l,a_f)+\gamma\mathbb{E}_{s'\sim\mathcal{P}(s,a_l,a_f)}[V_f^{\pi_x,\pi_y}(s')]\big)\nabla\log\pi_x(a_l|s)\Big]\\ &+\gamma\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s',\pi_x,\pi_y\Big]\\ &-\tau\gamma\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s',\pi_x,\pi_y\Big]\end{aligned} \tag{C.14}$$

Using the definition of $Q_f^{\pi_x,\pi_y}$ in the first term, and taking $\gamma$ of the second and third term inside the expectation gives

$$\begin{aligned}
&\nabla_x Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a_f)\\
&=\mathbb{E}_{a_l\sim\pi_x(s)}\big[Q_f^{\pi_x,\pi_y}(s,a_l,a_f)\nabla\log\pi_x(a_l|s)\big]\\
&+\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_1=s',\pi_x,\pi_y\Big]\\
&-\tau\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_1=s',\pi_x,\pi_y\Big]
\end{aligned}$$

Continuing from above, combining the first and second-term yields

$$\begin{aligned}
&\nabla_x Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a_f)\\
&=\mathbb{E}_{a_l\sim\pi_x(s)}\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,a_{l,0}=a_l,a_{f,0}=a_f,\pi_x,\pi_y\Big]\\
&\quad-\tau\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_1=s',\pi_x,\pi_y\Big]\\
&=\mathbb{E}\Big[\sum_{t=0}^{\infty}\gamma^t Q_f^{\pi_x,\pi_y}(s_t,a_{l,t},a_{f,t})\nabla\log\pi_x(a_{l,t}|s_t)\big|s_0=s,a_{f,0}=a_f,\pi_x,\pi_y\Big]\\
&\quad-\tau\mathbb{E}_{a_l\sim\pi_x(s),s'\sim\mathcal{P}(s,a_l,a_f)}\mathbb{E}\Big[\sum_{t=1}^{\infty}\gamma^t h_{f,s_t}(\pi_y(s_t))\nabla\log\pi_x(a_{l,t}|s_t)\big|s_1=s',\pi_x,\pi_y\Big]
\end{aligned}$$

It can then be checked that Assumption 2 holds and the result follows from Lemma 8 and (C.14). ■

### C.5 Sufficient conditions of the smoothness assumption

**Lemma 20** *Suppose the following conditions hold.*

*(a) For any $(s,a)$, the policy parameterization $\pi_y$ satisfies 1) $\sum_a\|\nabla\pi_y(a|s)\|\le B_\pi$; and, 2) $\pi_y(a|s)$ is $L_y$-Lipschitz-smooth.*

*(b) If $\tau>0$ then: 1) for any $s$, assume $|h_s(\pi_y(s))|\le B_h$ and $\|\nabla_y h_s(\pi_y(s))\|\le B_h'$ on $\mathcal{Y}$; and, 2) $h_s(\pi_y(s))$ is $L_h$-Lipschitz-smooth on $\mathcal{Y}$.*

*(c) For any $(s,a,s')$, we have for any $x\in\mathcal{X}$ that 1) $|r_x(s,a)|\le B_r$; and, 2) $V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$ is $L_{vx}$-Lipschitz-smooth on $\mathcal{X}$ uniformly for $y\in\mathcal{Y}$.*

*Then it holds for any $s$ that $V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$ is Lipschitz-smooth on $\mathcal{X}\times\mathcal{Y}$:*

$$\|\nabla V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)-\nabla V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\|\le\max\{L_{vx},L_{vy}\}\|(x,y)-(x',y')\|$$

*where* $L_{vy}=\mathcal{O}\Big(\frac{B_\pi^2(B_r+\tau B_h)}{(1-\gamma)^3}+\frac{\tau B_h'B_\pi+|\mathcal{A}|L_y(B_r+\tau B_h)}{(1-\gamma)^2}+\frac{\tau(B_h'+L_h)}{1-\gamma}\Big)$.

Condition (a) holds for direct parameterization, where $\sum_a \|\nabla \pi_y(a|s)\| \leq |\mathcal{A}|$ and $L_y = 0$; and it also holds for softmax parameterization where $\sum_a \|\nabla \pi_y(a|s)\| = \sum_a \pi_y(a|s)\|\nabla \log \pi_y(a|s)\| \leq 1$ and $L_y = 2$. Condition (b) holds for a smooth composite of regularization function and policy, e.g., softmax and entropy (Mei et al., 2020, Lemma 14), or direct policy with a smooth regularization. function. Condition (c) 1) is guaranteed since $\mathcal{X}$ is compact and $r_x$ is continuous, and 2) needs to be checked for specific applications. For example, in RLHF/Reward shaping, it can be checked from the formula of $\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$ in Lemma 7 that there exists $L_{vx} = \frac{L_r}{1-\gamma}$ if $r_x$ is $L_r$-Lipschitz-smooth.

**Proof** We start the proof by showing $V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$ is Lipschitz-smooth in $y$ on uniformly for any $x$, that is

$$\|\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla_y V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\| \leq L_{vy}\|y - y'\| \tag{C.15}$$

where $L_{vy}$ is a constant independent of $x$. By the regularized policy gradient in (B.9), we have

$$\begin{aligned}\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) = \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_y}_{s,x}}\Big[\sum_a Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] \\ + \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi}_{s,x}}[-\nabla_y h_{\bar{s}}(\pi_y(\bar{s}))]\end{aligned} \tag{C.16}$$

where $d^{\pi_y}_{s,x}(\bar{s}) \coloneqq (1-\gamma)\sum_{t=0}^\infty \gamma^t P^{\pi_y}_x(s_t = \bar{s}|s_0 = s)$ is the discounted visitation distribution, and recall $P^{\pi_y}_x(s_t = \bar{s}|s_0 = s)$ is the probability of reaching state $\bar{s}$ at time step $t$ under $\mathcal{P}_x$ and $\pi_y$. Towards proving (C.15), we prove the following results:

**(1)** We have $Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a)$ is uniformly bounded, and $V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)$ and $Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a)$ are Lipschitz continuous in $y$ uniformly for any $x$.

By the definition of $Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a)$, we have

$$|Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a)| \leq \sum_{t=0}^\infty \gamma^t |r_x(s_t,a_t)| + \tau|h_{s_t}(\pi_y(s_t))| \leq \frac{B_r + \tau B_h}{1-\gamma}, \tag{C.17}$$

therefore it follows from (C.16) that

$$\|\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s)\| \leq B_\pi \frac{B_r + \tau B_h}{(1-\gamma)^2} + \frac{\tau B'_h}{1-\gamma} \tag{C.18}$$

Then by the definition of the Q function

$$Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a) = r_x(s,a) + \gamma\mathbb{E}_{s'\sim\mathcal{P}_x(s,a)}\big[V^{\pi_y}_{\mathcal{M}_\tau(x)}(s')\big]$$

we have

$$\|\nabla_y Q^{\pi_y}_{\mathcal{M}_\tau(x)}(s,a)\| \leq \gamma\big(B_\pi \frac{B_r + \tau B_h}{(1-\gamma)^2} + \frac{\tau B'_h}{1-\gamma}\big) \tag{C.19}$$

**(2)** We have $d^{\pi_y}_{s_0,x}(s)$ is Lipschitz-continuous in $y$ uniformly for any $x$. Define $\mathcal{M}$ as a MDP with $\tau = 0$, $r(s,a) = \mathbf{1}_s$ which is an indicator function of $s$, and transition $\mathcal{P}_x$. Then we can

write $d^{\pi_y}_{s_0,x}(s)$ as

$$\begin{aligned} d^{\pi_y}_{s_0,x}(s) &= \sum_{s'\in\mathcal{S}}\sum_{a'\in\mathcal{A}} d^{\pi_y}_{s_0,x}(s')\pi_y(a'|s')\mathbf{1}_s \\ &= \mathbb{E}_{s\sim d^{\pi_y}_{s_0,x}, a\sim\pi_y(s)}[r(s,a)] \\ &= (1-\gamma)V^{\pi_y}_{\mathcal{M}}(s_0) \end{aligned}$$

where the last equality follows from substituting in $d^{\pi_y}_{s_0,x}(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t P^{\pi_y}_x(s_t = s|s_0)$. It then follows from (C.18) with $\tau = 0$ (since $V^{\pi_y}_{\mathcal{M}}(s_0)$ has $\tau = 0$) that $d^{\pi_y}_{s_0,x}(s)$ is also uniformly Lipschitz continuous with constant $B_\pi$:

$$\sup_{s\in\mathcal{S}} \|d^{\pi_y}_{s_0,x}(s) - d^{\pi_{y'}}_{s_0,x}(s)\| \le B_\pi\|y - y'\|. \tag{C.20}$$

To this end, we can decompose the difference as

$$\begin{aligned} &\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla_y V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s) \\ &= \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_y}_{s,x}}\Big[\sum_a Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] - \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sum_a Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] \\ &+ \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sum_a Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] - \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sum_a Q^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] \\ &+ \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sum_a Q^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big] - \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sum_a Q^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_{y'}(a|\bar{s})\Big] \\ &+ \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_y}_{s,x}}[-\nabla_y h_{\bar{s}}(\pi_y(\bar{s}))] - \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}[-\nabla_y h_{\bar{s}}(\pi_y(\bar{s}))] \\ &+ \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}[-\nabla_y h_{\bar{s}}(\pi_y(\bar{s}))] - \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}[-\nabla_y h_{\bar{s}}(\pi_{y'}(\bar{s}))] \end{aligned}$$

Continuing from the above inequality, we have

$$\begin{aligned} &\|\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla_y V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\| \\ &\le \frac{1}{1-\gamma} 2\sup_s \|d^{\pi_y}_{s,x}(s) - d^{\pi_{y'}}_{s,x}(s)\| \sup\Big\|\sum_a Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a)\nabla\pi_y(a|\bar{s})\Big\| \\ &+ \frac{1}{1-\gamma}\sup_a \big|Q^{\pi_y}_{\mathcal{M}_\tau(x)}(\bar{s},a) - Q^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\bar{s},a)\big| \sum_a \big\|\nabla\pi_y(a|\bar{s})\big\| \\ &+ \frac{1}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\Big[\sup_a \big|Q^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(\bar{s},a)\big| \sum_a \big\|\nabla\pi_y(a|\bar{s}) - \nabla\pi_{y'}(a|\bar{s})\big\|\Big] \\ &+ \frac{\tau}{1-\gamma} 2\sup_s \|d^{\pi_y}_{s,x}(s) - d^{\pi_{y'}}_{s,x}(s)\| \sup\|\nabla_y h_{\bar{s}}(\pi_y(\bar{s}))\| \\ &+ \frac{\tau}{1-\gamma}\mathbb{E}_{\bar{s}\sim d^{\pi_{y'}}_{s,x}}\big[\|\nabla_y h_{\bar{s}}(\pi_y(\bar{s})) - \nabla_y h_{\bar{s}}(\pi_{y'}(\bar{s}))\|\big]. \end{aligned} \tag{C.21}$$

Then given the assumptions (a), (b) in this lemma, along with the (C.17)–(C.20), we can get

$$\|\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla_y V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\| \le L_{vy}\|y - y'\| \tag{C.22}$$

where $L_{vy} = \mathcal{O}\Big(\frac{B_\pi^2(B_r+\tau B_h)}{(1-\gamma)^3} + \frac{\tau B_h' B_\pi + |\mathcal{A}| L_y(B_r+\tau B_h)}{(1-\gamma)^2} + \frac{\tau(B_h'+L_h)}{1-\gamma}\Big)$. Thus we conclude

$$\begin{aligned}
&\|\nabla V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\|^2 \\
&= \|\nabla_y V^{\pi_y}_{\mathcal{M}_\tau(x)}(s) - \nabla_y V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s)\|^2 + \|\nabla_x V^{\pi_{y'}}_{\mathcal{M}_\tau(x)}(s) - \nabla_x V^{\pi_{y'}}_{\mathcal{M}_\tau(x')}(s)\|^2 \\
&\le L_{vy}^2\|y-y'\|^2 + L_{vx}^2\|x-x'\|^2 \le \max\{L_{vy}^2, L_{vx}^2\}\|(x,y)-(x',y')\|^2
\end{aligned} \tag{C.23}$$

which proves the result. ■

## C.6 Proof of Lemma 10

### C.6.1 Smoothness of the value penalty

**Proof** Under the two assumptions, Lemma 17 holds and thus $\pi_y^*(x)$ is unique and is $\tau^{-1}C_J$-Lipschitz continuous on $\mathcal{X}$. Thus for any $y, y' \in \mathcal{Y}^*(x)$, we have $\pi_y = \pi_{y'} = \pi_y^*(x)$. With Lemma 6, we have

$$\begin{aligned}
&\|\nabla \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) - \nabla \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x')}(\rho)\| \\
&= \|\nabla V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)|_{\pi=\pi_y^*(x)} - \nabla V^{\pi}_{\mathcal{M}_\tau(x')}(\rho)|_{\pi=\pi_y^*(x')}\| \\
&\le L_v(\|x-x'\| + \|\pi_y^*(x) - \pi_y^*(x')\|) \\
&\le L_v(1+\tau^{-1}C_J)\|x-x'\|.
\end{aligned} \tag{C.24}$$

It then follows from $V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$ is $L_v$-Lipschitz smooth that the value penalty is $L_v(2+\tau^{-1}C_J)$-Lipschitz smooth. ■

### C.6.2 Smoothness of the Bellman penalty

**Proof** First note that Lemma 17 holds and thus $\pi_y^*(x)$ is $\tau^{-1}C_J$-Lipschitz continuous on $\mathcal{X}$. We have $p(x,y) = g(x,y) - v(x)$ where

$$g(x,y) \coloneqq \mathbb{E}_{s\sim\rho}[\langle y_s,\, q_s(x)\rangle + \tau h_s(y_s)]. \tag{C.25}$$

By Lemma 8,

$$\nabla_x g(x,y) = -\mathbb{E}_{s\sim\rho, a\sim y_s}\left[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\right]\big|_{\pi=\pi_y^*(x)}. \tag{C.26}$$

Since $\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)$ is $L_v$-Lipschitz continuous by the assumption, and $\pi_y^*(x)$ is $\tau^{-1}C_J$-Lipschitz continuous, we have $\nabla_x g(x,y)$ is $L_v(1+\tau^{-1}C_J)$-Lipschitz continuous at $x \in \mathcal{X}$ uniformly for any $y$. We also have $\nabla_x g(x,y)$ is $C_J$-Lipschitz continuous at $y \in \Pi$ uniformly for any $x \in \mathcal{X}$. Therefore, we conclude $\nabla_x g(x,y)$ is $(C_J + L_v(1+\tau^{-1}C_J))$-Lipschitz continuous at $(x,y)$ on $\mathcal{X}\times\Pi$.
Next we have

$$\nabla_y g(x,y) = \Big(\rho(s)q_s(x) + \tau\rho(s)\nabla h_s(y_s)\Big)_{s\in\mathcal{S}}. \tag{C.27}$$

Since $q_s$ is $C_J$-Lipschitz continuous, and $h_s$ is $L_h$-Lipschitz smooth, we have $\nabla_y g(x,y)$ is $(C_J+L_h)$-Lipschitz continuous at $(x,y)$ on $\mathcal{X}\times\Pi$.
Collecting the Lipschitz continuity of $\nabla_x g(x,y)$ and $\nabla_y g(x,y)$ yields $g(x,y)$ is Lipschitz smooth with modulus $L_g = 2C_J + L_v(1+\tau^{-1}C_J) + L_h$. Then we have

$$\|v(x)-v(x')\| = \|g(x,\pi_y^*(x)) - g(x',\pi_y^*(x'))\| \le L_g(\|x-x'\| + \tau^{-1}C_J\|x-x'\|). \quad \text{(C.28)}$$

Then we have $p(x,y) = g(x,y) - v(x)$ is Lipschitz smooth with modulus $L_g(2+\tau^{-1}C_J)$. Together with the assumption that $f$ is $L_f$-Lipschitz smooth gives $F_\lambda$ is $L_v$-Lipschitz smooth with $L_v = L_f + \lambda L_g(2+\tau^{-1}C_J)$. ■

### C.7 Example gradient estimators of the penalty functions

In this section, we give examples of $\hat{\nabla}p(x,y;\hat{\pi})$ that is an estimator of $\nabla p(x,y)$.
**Value penalty.** Consider choosing the value penalty $p(x,y) = -V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \max_{y\in\mathcal{Y}} V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho)$. Then by Lemma 6, we have

$$\nabla_x p(x,y) = -\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \nabla_x V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)|_{\pi=\pi_y^*(x)}$$

where recall $\pi_y^*(x)$ is the optimal policy of MDP $\mathcal{M}_\tau(x)$ on the policy class $\Pi = \{\pi_y : y\in\mathcal{Y}\}$. A natural choice of $\hat{\nabla}p(x,y;\hat{\pi})$ is then

$$\hat{\nabla}p(x,y;\hat{\pi}) \coloneqq \Big( -\nabla_x V^{\pi_y}_{\mathcal{M}_\tau(x)}(\rho) + \nabla_x V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)|_{\pi=\hat{\pi}}, \nabla_y p(x,y)\Big) \quad \text{(C.29)}$$

By (Agarwal et al., 2020, Lemma D.3.), there exists constant $L_v = 2\gamma|\mathcal{A}|/(1-\gamma)^3$ that $V^{\pi}_{\mathcal{M}_\tau(x)}(\rho)$ is $L_v$-Lipschitz-smooth in $\pi$ for any $x$. Then the estimation error can be quantified by

$$\|\hat{\nabla}p(x,y;\hat{\pi}) - \nabla p(x,y)\| \le L_v\|\pi_y^*(x) - \hat{\pi}\|. \quad \text{(C.30)}$$

Therefore, the estimation error is upper bounded by the policy optimality gap $\|\pi_y^*(x)-\hat{\pi}\|$. One may use efficient algorithms (e.g., policy mirror descent (Zhan et al., 2023)) to solve for $\hat{\pi}$, which has an iteration complexity of $\mathcal{O}(\log(1/\epsilon))$ to achieve $\|\pi_y^*(x)-\hat{\pi}\|\le\epsilon$. Then Assumption 3 is guaranteed with complexity $\mathcal{O}(\log(\lambda^2/\epsilon_{orac}))$.
**Bellman penalty.** Consider choosing the Bellman penalty $p(x,y) = g(x,y) - v(x)$ where recall $g(x,y) = \mathbb{E}_{s\sim\rho}[\langle y_s, q_s(x)\rangle + \tau h_s(y_s)]$ and $v(x) = \min_{y\in\mathcal{Y}} g(x,y)$. Then by Lemma 8, we have

$$\begin{aligned}\nabla_x p(x,y) = &-\mathbb{E}_{s\sim\rho,a\sim\pi_y(s)}\big[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\big]\big|_{\pi=\pi_y^*(x)} \\ &+ \mathbb{E}_{s\sim\rho,a\sim\pi(s)}\big[\nabla_x Q^{\pi}_{\mathcal{M}_\tau(x)}(s,a)\big]\big|_{\pi=\pi_y^*(x)}\end{aligned} \quad \text{(C.31)}$$

Therefore, a natural choice of $\hat{\nabla}p(x,y;\hat{\pi})$ is then

$$\begin{aligned}&\hat{\nabla}p(x,y;\hat{\pi}) \\ &\coloneqq \Big( -\mathbb{E}_{s\sim\rho,a\sim\pi_y(s)}\big[\nabla_x Q^{\hat{\pi}}_{\mathcal{M}_\tau(x)}(s,a)\big] + \mathbb{E}_{s\sim\rho,a\sim\hat{\pi}(s)}\big[\nabla_x Q^{\hat{\pi}}_{\mathcal{M}_\tau(x)}(s,a)\big], \nabla_y p(x,y)\Big)\end{aligned}$$

It then follows similarly to (C.30) that Assumption 3 is guaranteed with complexity $\mathcal{O}(-\log(\epsilon_{orac}/\lambda^2))$.

**Example algorithms to get $\hat{\pi}$.** Finally, we also explicitly write down the update to obtain $\hat{\pi}$ to be self-contained. If we are using policy mirror descent, then at each outer-iteration $k$, for $i = 1, ...T$ where $T$ is the inner iteration number, we run

$$\pi_k^{i+1}(\cdot|s) = \operatorname*{argmin}_{p\in\Pi} \Big\{ -\langle p,\, Q^{\pi_k^i}_{\mathcal{M}_\tau(x)}(s,\cdot)\rangle + \tau h_s(p) + \frac{1}{\eta} D_h(p, \pi_k^i; \xi_k^i) \Big\},\ \text{for any } s \in \mathcal{S}$$

where $\eta$ is a learning rate, $D_h$ is the Bregman divergence, and $\xi_k^i$ is given by

$$\xi_k^{i+1}(s,a) = \frac{1}{1+\eta\tau}\xi_k^i(s,a) + \frac{\eta}{1+\eta\tau} Q^{\pi_k^i}_{\mathcal{M}_\tau(x)}(s,a). \tag{C.32}$$

Finally, we set the last iterate $\pi_k^{T+1}(\cdot|s)$ as the approximate optimal policy $\hat{\pi}_k$. For theoretical reasons, we use this update in the analysis to gain a fast rate. While practically our update scheme is not limited to policy mirror descent. As a simple example, the policy gradient-based algorithms can also be used:

$$\hat{y}_k^{i+1} = \mathrm{Proj}_{\mathcal{Y}} \Big[ \hat{y}_k^i + \eta \nabla_{\hat{y}} V^{\pi_{\hat{y}_k^i}}_{\mathcal{M}_\tau(x)}(\rho) \Big],\ \text{for } i = 1, 2, \ldots, T. \tag{C.33}$$

We use the last iterate as the approximate optimal policy parameter: $\hat{\pi}_k = \pi_{\hat{y}_k^{T+1}}$. In the above update, the policy gradient $\nabla_{\hat{y}} V^{\pi_{\hat{y}_k^i}}_{\mathcal{M}_\tau(x)}(\rho)$ can be estimated by a wide range of algorithms including the basic Reinforce (Baxter and Bartlett, 2001), and the advantage actor-critic (Mnih et al., 2016).

### C.8 Proof of Theorem 11

**Proof** In this proof, we write $z = (x, y)$. Consider choosing either the value penalty or the Bellman penalty, then Lemma 10 holds under the assumptions of this theorem. Therefore, $F_\lambda$ is $L_\lambda$-Lipschitz-smooth with $L_\lambda = L_f + \lambda L_p$. Then by Lipschitz-smoothness of $F_\lambda$, it holds that

$$\begin{aligned} F_\lambda(z_{k+1}) &\le F_\lambda(z_k) + \langle \nabla F_\lambda(z_k),\, z_{k+1} - z_k\rangle + \frac{L_\lambda}{2}\|z_{k+1} - z_k\|^2 \\ &\overset{\alpha \le \frac{1}{L_\lambda}}{\le} F_\lambda(z_k) + \langle \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k),\, z_{k+1} - z_k\rangle + \frac{1}{2\alpha}\|z_{k+1} - z_k\|^2 \\ &\quad + \langle \nabla F_\lambda(z_k) - \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k),\, z_{k+1} - z_k\rangle. \end{aligned} \tag{C.34}$$

Consider the second term in the RHS of (C.34). It is known that $z_{k+1}$ can be written as

$$z_{k+1} = \arg\min_{z\in\mathcal{Z}} \langle \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k),\, z\rangle + \frac{1}{2\alpha}\|z - z_k\|^2.$$

By the first-order optimality condition of the above problem, it holds that

$$\langle \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k) + \frac{1}{\alpha}(z_{k+1} - z_k),\, z_{k+1} - z\rangle \le 0,\ \forall z \in \mathcal{Z}.$$

Since $z_k \in \mathcal{Z}$, we can choose $z = z_k$ in the above inequality and obtain

$$\langle \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k),\, z_{k+1} - z_k \rangle \leq -\frac{1}{\alpha}\|z_{k+1} - z_k\|^2. \tag{C.35}$$

Consider the last term in the RHS of (C.34). By Young's inequality, we first have

$$\begin{aligned}
&\langle \nabla F_\lambda(z_k) - \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k),\, z_{k+1} - z_k \rangle \\
&\leq \alpha \|\nabla F_\lambda(z_k) - \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k)\|^2 + \frac{1}{4\alpha}\|z_{k+1} - z_k\|^2 \\
&\leq \alpha\lambda^2 \|\nabla p(z_k) - \hat{\nabla} p(z_k; \hat{\pi}_k)\|^2 + \frac{1}{4\alpha}\|z_{k+1} - z_k\|^2
\end{aligned} \tag{C.36}$$

Substituting (C.36) and (C.35) into (C.34) and rearranging the resulting inequality yield

$$\frac{1}{4\alpha}\|z_{k+1} - z_k\|^2 \leq F_\lambda(z_k) - F_\lambda(z_{k+1}) + \alpha\lambda^2 \|\nabla p(z_k) - \hat{\nabla} p(z_k; \hat{\pi}_k)\|^2. \tag{C.37}$$

With $\bar{z}_{k+1}$ defined in (4.5), we have

$$\begin{aligned}
\|\bar{z}_{k+1} - z_k\|^2 &\leq 2\|\bar{z}_{k+1} - z_{k+1}\|^2 + 2\|z_{k+1} - z_k\|^2 \\
&\leq 2\alpha^2 \|\nabla F_\lambda(z_k) - \hat{\nabla} F_\lambda(z_k; \hat{\pi}_k)\|^2 + 2\|z_{k+1} - z_k\|^2 \\
&\leq 2\alpha^2\lambda^2 \|\nabla p(z_k) - \hat{\nabla} p(z_k; \hat{\pi}_k)\|^2 + 2\|z_{k+1} - z_k\|^2
\end{aligned} \tag{C.38}$$

where the second inequality uses non-expansiveness of $\mathrm{Proj}_{\mathcal{Z}}$.
Together (C.37) and (C.38) imply

$$\|\bar{z}_{k+1} - z_k\|^2 \leq 10\alpha^2\lambda^2 \|\nabla p(z_k) - \hat{\nabla} p(z_k; \hat{\pi}_k)\|^2 + 8\alpha(F_\lambda(z_k) - F_\lambda(z_{k+1})).$$

Since $p(x, y) \geq 0$, $F_\lambda(z) \geq \inf_{z \in \mathcal{Z}} f(z)$ for any $z \in \mathcal{Z}$. Taking a telescope sum of the above inequality and using $G_\lambda(z_k) = \frac{1}{\alpha}(z_k - \bar{z}_{k+1})$ yield

$$\begin{aligned}
\sum_{k=1}^{K} \|G_\lambda(z_k)\|^2 &\leq \frac{8\big(F_\lambda(z_1) - \inf_{z \in \mathcal{Z}} f(z)\big)}{\alpha} + \sum_{k=1}^{K} 10\lambda^2 \|\nabla p(z_k) - \hat{\nabla} p(z_k; \hat{\pi}_k)\|^2 \\
&\leq \frac{8\big(F_\lambda(z_1) - \inf_{z \in \mathcal{Z}} f(z)\big)}{\alpha} + \sum_{k=1}^{K} \frac{1}{2}\|G_\lambda(z_k)\|^2 + \frac{K}{2}\epsilon_{orac}
\end{aligned} \tag{C.39}$$

where the last inequality follows from Assumption 3. Rearranging gives

$$\sum_{k=1}^{K} \|G_\lambda(z_k)\|^2 \leq \frac{16\big(F_\lambda(z_1) - \inf_{z \in \mathcal{Z}} f(z)\big)}{\alpha} + K\epsilon_{orac}. \tag{C.40}$$

This proves the first inequality in this theorem. The result for $\mathcal{OS}$ follows similarly with $F_\lambda(y)$ being $L_v$-Lipschitz-smooth and $\epsilon_{orac} = 0$ since no oracle is needed. ■

## Appendix D. Proof in Section 5

### D.1 Proof of Lemma 13

**(a).** Treating $(\pi_2, x)$ (or $(\pi_1, x)$) as the parameter, it follows from Lemma 4 that $\pi_1^*$ (or $\pi_2^*$) is unique. Under Assumption 6, it then follows from Lemma 19 that (a) holds.
**(b).** We have

$$\begin{aligned}
&\max_{\pi' \in \Pi} \langle \nabla_\pi \psi(x, \pi),\, \pi - \pi' \rangle \\
&= \max_{\pi' \in \Pi} \langle \nabla_{\pi_2} V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho),\, \pi_2 - \pi_2' \rangle + \langle \nabla_{\pi_1} - V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho),\, \pi_1 - \pi_1' \rangle \\
&= \max_{\pi_2' \in \Pi_2} \langle \nabla_{\pi_2} - V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho),\, \pi_2' - \pi_2 \rangle + \max_{\pi_1' \in \Pi_1} \langle \nabla_{\pi_1} V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho),\, \pi_1' - \pi_1 \rangle \\
&\geq \mu \Big[ \max_{\pi_2 \in \Pi_2} \big( - V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho) \big) + V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho) + \max_{\pi_1 \in \Pi_1} \big( V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho) \big) - V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho) \Big] \\
&= \mu \Big[ V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho) - V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho) - \big( - \max_{\pi_1 \in \Pi_1} V_{\mathcal{M}_\tau(x)}^{\pi_1, \pi_2^*}(\rho) + \min_{\pi_2 \in \Pi_2} V_{\mathcal{M}_\tau(x)}^{\pi_1^*, \pi_2}(\rho) \big) \Big] \\
&= \mu \big( \psi(x, \pi) - \min_{\pi \in \Pi} \psi(x, \pi) \big)
\end{aligned} \tag{D.1}$$

where the inequality follows from Lemma 2.

### D.2 Proof of Lemma 14

**Proof** Given $x_\lambda$, point $\pi_\lambda$ satisfies the first-order stationary condition:

$$\langle \nabla_\pi f(x_\lambda, \pi_\lambda) + \lambda \nabla_\pi \psi(x_\lambda, \pi_\lambda),\, \pi_\lambda - \pi' \rangle \leq 0,\ \forall \pi'$$

which leads to

$$\begin{aligned}
\langle \nabla_\pi \psi(x_\lambda, \pi_\lambda),\, \pi_\lambda - \pi' \rangle &\leq -\frac{1}{\lambda} \langle \nabla_\pi f(x_\lambda, \pi_\lambda),\, \pi_\lambda - \pi' \rangle \\
&\leq \frac{L \|\pi_\lambda - \pi'\|}{\lambda} \leq \frac{L}{\lambda},\ \forall \pi'.
\end{aligned} \tag{D.2}$$

Combining the above inequality with Lemma 13 5.10 yields

$$\psi(x_\lambda, \pi_\lambda) \leq \frac{L}{\lambda}.$$

Define $\epsilon_\lambda \coloneqq \psi(x_\lambda, \pi_\lambda)$ then $\epsilon_\lambda \leq \delta$ by choice of $\lambda$.
Since $(x_\lambda, \pi_\lambda)$ is a local solution of $\mathcal{BZ}_{\lambda p}$, it holds for any feasible $(x, \pi)$ in the region where $(x_\lambda, \pi_\lambda)$ attains its minimum that

$$f(x_\lambda, \pi_\lambda) + \lambda \psi(x_\lambda, \pi_\lambda) \leq f(x, \pi) + \lambda \psi(x, \pi). \tag{D.3}$$

From the above inequality, it holds for any $(x, \pi)$ feasible for $\mathcal{BM}_\epsilon$ and in the region that

$$f(x_\lambda, \pi_\lambda) \leq f(x, \pi) + \lambda \big( \psi(x, \pi) - \epsilon_\lambda \big) \leq f(x, \pi) \tag{D.4}$$

which proves the result. ■

### D.3 Justification of the smoothness assumption in the two-player case

**Lemma 21** *Consider the following conditions.*

*(a) For any $s$, assume $|h_s(\pi_1(s))| \le B_h$ and $\|\nabla h_s(\pi_1(s))\| \le B_h'$ for any $\pi_1 \in \Pi_1$, and; $h_s(\pi_1(s))$ is $L_h$-Lipschitz-smooth on $\Pi_1$. Also, assume this holds for player 2's policy $\pi_2 \in \Pi_2$.*

*(b) For any $(s, a_1, a_2)$, we have for any $x \in \mathcal{X}$ that 1) $|r_x(s, a_1, a_2)| \le B_r$, and; 2) $V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(\rho)$ is $L_{vx}'$-Lipschitz-smooth on $\mathcal{X}$ uniformly for $(\pi_1, \pi_2)$.*

Then there exists a universal constant $L_v = \mathcal{O}\Big(\frac{|\mathcal{A}|(B_r+\tau B_h)}{(1-\gamma)^3} + \frac{\tau B_h'|\mathcal{A}|}{(1-\gamma)^2} + \frac{\tau(B_h'+L_h)}{1-\gamma} + L_{vx}'\Big)$ such that $V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s)$ is $L_v$-Lipschitz-smooth on $\mathcal{X} \times \Pi_1 \times \Pi_2$.

**Proof** Recall the definition of the value function

$$V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) = V^{\pi}_{\mathcal{M}_\tau(x)}(s) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \big(r_x(s_t, a_t) - \tau h_{s_t}(\pi_1(s_t)) + \tau h_{s_t}(\pi_2(s_t))\big) \big| s_0 = s, \pi\Big]$$

where the expectation is taken over the trajectory generated by $a_t \sim (\pi_1(s_t), \pi_2(s_t))$ and $s_{t+1} \sim \mathcal{P}(s_t, a_t)$. When viewing $\pi_1$ as the main policy and player 1 as the main player, we can view $(\pi_2, x)$ as the parameter of player 1's MDP, where the reward function is given by $\mathbb{E}_{a_2\sim\pi_2(s)}[r_x(s, a_1, a_2)]$, and the transition is $\mathbb{E}_{a_2\sim\pi_2(s)}[\mathcal{P}(\cdot|s, a_1, a_2)]$. To prove $V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s)$ is Lipschitz-smooth in $\pi_1$, it is then natural to use the previous results on single-agent parameterized MDP in Lemma 20. Specifically, we hope to use (C.22).

Under the assumptions of this lemma, for (C.22) to hold, we additionally need to check Lemma 20 (a).

$$\sum_a \|\nabla \pi_1(a_1|s)\| = \sum_a \|\mathbf{1}_{a_1,s}\| = |\mathcal{A}|, \ \nabla^2 \pi(a|s) = 0 \text{ thus } L_y = 0. \tag{D.5}$$

Then there exists constant $L_{v,1}$ that

$$\|\nabla_{\pi_1} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla_{\pi_1} V^{\pi_1',\pi_2}_{\mathcal{M}_\tau(x)}(s)\| \le L_{v,1}\|\pi_1 - \pi_1'\| \tag{D.6}$$

where $L_{v,1} = \mathcal{O}\Big(\frac{|\mathcal{A}|(B_r+\tau B_h)}{(1-\gamma)^3} + \frac{\tau B_h'|\mathcal{A}|}{(1-\gamma)^2} + \frac{\tau(B_h'+L_h)}{1-\gamma}\Big)$, which is a uniform constant for any $\pi_2, x$. Similarly, there exists a uniform constant $L_{v,2}$ that

$$\|\nabla_{\pi_2} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla_{\pi_2} V^{\pi_1,\pi_2'}_{\mathcal{M}_\tau(x)}(s)\| \le L_{v,2}\|\pi_2 - \pi_2'\|. \tag{D.7}$$

The above two inequalities along with assumption (b) 2) gives

$$\begin{aligned}
\|\nabla V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla V^{\pi_1',\pi_2'}_{\mathcal{M}_\tau(x')}(s)\| &\le \|\nabla_{\pi_1} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla_{\pi_1} V^{\pi_1',\pi_2}_{\mathcal{M}_\tau(x)}(s)\| \\
&\quad + \|\nabla_{\pi_2} V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla_{\pi_2} V^{\pi_1,\pi_2'}_{\mathcal{M}_\tau(x)}(s)\| \\
&\quad + \|\nabla_x V^{\pi_1,\pi_2}_{\mathcal{M}_\tau(x)}(s) - \nabla_x V^{\pi_1,\pi_2'}_{\mathcal{M}_\tau(x)}(s)\| \\
&\le L_{vx}'\|x - x'\| + L_{v,1}\|\pi_1 - \pi_1'\| + L_{v,2}\|\pi_2 - \pi_2'\|.
\end{aligned} \tag{D.8}$$

Then the result holds with $L_v = \max\{L_{vx}', L_{v,1}, L_{v,2}\}$. ■

### D.4 Proof of Lemma 15

**Proof** Denote $\pi_1^*(x,\pi_2) \coloneqq \arg\max_{\pi_1\in\Pi_1} V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2}(\rho)$. Treating $(x,\pi_2)$ as the parameter of player 1's MDP, then we have Lemma 17 holds under Assumption 8. It then follows that there exits a constant $L_\pi$ such that $\pi_1^*(x,\pi_2)$ is $L_\pi$-Lipschitz-continuous. Similar results also hold for $\pi_2^*(x,\pi_1)$. With Lemma 13 (a), we have

$$\begin{aligned}
&\|\nabla\psi(x,\pi)-\nabla\psi(x',\pi')\|^2\\
&=\|\nabla_{(x,\pi_1)}V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2^*(x,\pi_1)}(\rho)-\nabla_{(x,\pi_1)}V_{\mathcal{M}_\tau(x')}^{\pi_1',\pi_2^*(x',\pi_1')}(\rho)\|^2\\
&\ +\|\nabla_{(x,\pi_2)}V_{\mathcal{M}_\tau(x)}^{\pi_1^*(x,\pi_2),\pi_2}(\rho)-\nabla_{(x,\pi_2)}V_{\mathcal{M}_\tau(x')}^{\pi_1^*(x',\pi_2'),\pi_2'}(\rho)\|^2\\
&\le L_v^2(2\|x-x'\|^2+\|\pi_1^*(x,\pi_2)-\pi_1^*(x',\pi_2')\|^2+\|\pi_2^*(x,\pi_1)-\pi_2^*(x',\pi_1')\|^2)\\
&\le 2L_v^2(1+L_\pi^2)\|x-x'\|+L_v^2L_\pi^2(\|\pi_1-\pi_1'\|^2+\|\pi_2-\pi_2'\|^2) \qquad \text{(D.9)}
\end{aligned}$$

which proves $\nabla\psi(x,\pi)$ is $L_v\sqrt{2(1+L_\pi^2)}$-Lipschitz continuous. ■

### D.5 Gradient Estimator Accuracy

We omit the iteration index $k$ here. The following arguments hold for any iteration $k$. Recall that

$$\hat{\nabla}\psi(x,\pi;\hat{\pi})=\Big(\nabla_x V_{\mathcal{M}_\tau(x)}^{\hat{\pi}_1,\pi_2}(\rho)-\nabla_x V_{\mathcal{M}_\tau(x)}^{\pi_1,\hat{\pi}_2}(\rho),\big(-\nabla_{\pi_1}V_{\mathcal{M}_\tau(x)}^{\pi_1,\hat{\pi}_2}(\rho),\nabla_{\pi_2}V_{\mathcal{M}_\tau(x)}^{\hat{\pi}_1,\pi_2}(\rho)\big)\Big)$$

and the formula of $\nabla\psi$ is (see Lemma 13 (a)):

$$\nabla\psi(x,\pi)=\Big(\nabla_x V_{\mathcal{M}_\tau(x)}^{\pi_1^*,\pi_2}(\rho)-\nabla_x V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2^*}(\rho),\big(-\nabla_{\pi_1}V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2^*}(\rho),\nabla_{\pi_2}V_{\mathcal{M}_\tau(x)}^{\pi_1^*,\pi_2}(\rho)\big)\Big)$$

where $\pi_1^*\coloneqq\operatorname{argmax}_{\pi_1\in\Pi_1}V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2}(\rho)$ and $\pi_2^*$ defined similarly. It then follows that

$$\begin{aligned}
&\|\hat{\nabla}\psi(x,\pi;\hat{\pi})-\nabla\psi(x,\pi)\|\\
&\le 2\|\nabla V_{\mathcal{M}_\tau(x)}^{\pi_1^*,\pi_2}(\rho)-\nabla V_{\mathcal{M}_\tau(x)}^{\hat{\pi}_1,\pi_2}(\rho)\|+2\|\nabla V_{\mathcal{M}_\tau(x)}^{\pi_1,\pi_2^*}(\rho)-\nabla V_{\mathcal{M}_\tau(x)}^{\pi_1,\hat{\pi}_2}(\rho)\|\\
&\le 2L_v(\|\hat{\pi}_1-\pi_1^*\|+\|\hat{\pi}_2-\pi_2^*\|) \qquad \text{(D.10)}
\end{aligned}$$

where the last inequality follows from Assumption 8 (a). Fixing $x,\pi_2$, it takes the policy mirror descent algorithm (Zhan et al., 2023) an iteration complexity of $\mathcal{O}(-\log\epsilon)$ to solve for a $\hat{\pi}_1$ such that $\|\hat{\pi}_1-\pi_1^*\|\le\epsilon$ (and similarly for $\hat{\pi}_2$). Therefore, the iteration complexity to guarantee Assumption 7 is $\mathcal{O}(-\log(\epsilon_{orac}/\lambda^2))$.

### D.6 Proof of Theorem 16

**Proof** In this proof, we write $z=(x,\pi)$. Given $\hat{\pi}$, we also define

$$\hat{\nabla}F_\lambda(z;\hat{\pi})\coloneqq\nabla f(z)+\lambda\hat{\nabla}\psi(z;\hat{\pi}). \qquad \text{(D.11)}$$

Thus the update we are analyzing can be written as

$$z^{k+1} = \text{Proj}_{\mathcal{Z}}\left[z^k - \alpha\hat{\nabla}F_\lambda(z^k;\hat{\pi}^k)\right] \tag{D.12}$$

Now we start proving the result. Under the assumptions, we have $F_\lambda$ is $L_\lambda$-Lipschitz-smooth with $L_\lambda = L_f + \lambda L_\psi$, thus it holds that

$$\begin{aligned} F_\lambda(z^{k+1}) &\le F_\lambda(z^k) + \langle \nabla F_\lambda(z^k),\, z^{k+1} - z^k\rangle + \frac{L_\lambda}{2}\|z^{k+1} - z^k\|^2 \\ &\overset{\alpha\le\frac{1}{L_\lambda}}{\le} F_\lambda(z^k) + \langle \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k),\, z^{k+1} - z^k\rangle + \frac{1}{2\alpha}\|z^{k+1} - z^k\|^2 \\ &\quad + \langle \nabla F_\lambda(z^k) - \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k),\, z^{k+1} - z^k\rangle. \end{aligned} \tag{D.13}$$

Consider the second term in the RHS of (D.13). It is known that $z^{k+1}$ can be written as

$$z^{k+1} = \arg\min_{z\in\mathcal{Z}} \langle \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k),\, z\rangle + \frac{1}{2\alpha}\|z - z^k\|^2.$$

By the first-order optimality condition of the above problem, it holds that

$$\langle \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k) + \frac{1}{\alpha}(z^{k+1} - z^k),\, z^{k+1} - z\rangle \le 0,\ \forall z \in \mathcal{Z}.$$

Since $z^k \in \mathcal{Z}$, we can choose $z = z^k$ in the above inequality and obtain

$$\langle \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k),\, z^{k+1} - z^k\rangle \le -\frac{1}{\alpha}\|z^{k+1} - z^k\|^2. \tag{D.14}$$

Consider the last term in the RHS of (D.13). By Young's inequality, we first have

$$\begin{aligned} &\langle \nabla F_\lambda(z^k) - \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k),\, z^{k+1} - z^k\rangle \\ &\le \alpha\|\nabla F_\lambda(z^k) - \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k)\|^2 + \frac{1}{4\alpha}\|z^{k+1} - z^k\|^2 \\ &\le \alpha\lambda^2\|\nabla\psi(z^k) - \hat{\nabla}\psi(z^k;\hat{\pi}^k)\|^2 + \frac{1}{4\alpha}\|z^{k+1} - z^k\|^2 \end{aligned} \tag{D.15}$$

Substituting (D.15) and (D.14) into (D.13) and rearranging the resulting inequality yield

$$\frac{1}{4\alpha}\|z^{k+1} - z^k\|^2 \le F_\lambda(z^k) - F_\lambda(z^{k+1}) + \alpha\lambda^2\|\nabla\psi(z^k) - \hat{\nabla}\psi(z^k;\hat{\pi}^k)\|^2. \tag{D.16}$$

With $\bar{z}^{k+1}$ defined in (5.13), we have

$$\begin{aligned} \|\bar{z}^{k+1} - z^k\|^2 &\le 2\|\bar{z}^{k+1} - z^{k+1}\|^2 + 2\|z^{k+1} - z^k\|^2 \\ &\le 2\alpha^2\|\nabla F_\lambda(z^k) - \hat{\nabla}F_\lambda(z^k;\hat{\pi}^k)\|^2 + 2\|z^{k+1} - z^k\|^2 \\ &\le 2\alpha^2\lambda^2\|\nabla\psi(z^k) - \hat{\nabla}\psi(z^k;\hat{\pi}^k)\|^2 + 2\|z^{k+1} - z^k\|^2 \end{aligned} \tag{D.17}$$

where the second inequality uses non-expansiveness of $\text{Proj}_{\mathcal{Z}}$.
Together (D.16) and (D.17) imply

$$\|\bar{z}^{k+1} - z^k\|^2 \le 10\alpha^2\lambda^2\|\nabla\psi(z^k) - \hat{\nabla}\psi(z^k;\hat{\pi}^k)\|^2 + 8\alpha(F_\lambda(z^k) - F_\lambda(z^{k+1})).$$

Since $p(x,y) \geq 0$, $F_\lambda(z) \geq \inf_{z\in\mathcal{Z}} f(z)$ for any $z \in \mathcal{Z}$. Taking a telescope sum of the above inequality and using $\mathcal{G}_\lambda(z^k) = \frac{1}{\alpha}(z^k - \bar{z}^{k+1})$ yield

$$\begin{aligned}\sum_{k=1}^K \|\mathcal{G}_\lambda(z^k)\|^2 &\leq \frac{8\big(F_\lambda(z_1) - \inf_{z\in\mathcal{Z}} f(z)\big)}{\alpha} + \sum_{k=1}^K 10\lambda^2 \|\nabla\psi(z^k) - \hat{\nabla}\psi(z^k;\hat{\pi}^k)\|^2 \\ &\leq \frac{8\big(F_\lambda(z_1) - \inf_{z\in\mathcal{Z}} f(z)\big)}{\alpha} + \sum_{k=1}^K \frac{1}{2}\|\mathcal{G}_\lambda(z^k)\|^2 + \frac{K}{2}\epsilon_{orac} \qquad \text{(D.18)}\end{aligned}$$

where the last inequality follows from Assumption 7. Rearranging gives

$$\sum_{k=1}^K \|\mathcal{G}_\lambda(z^k)\|^2 \leq \frac{16\big(F_\lambda(z_1) - \inf_{z\in\mathcal{Z}} f(z)\big)}{\alpha} + K\epsilon_{orac}. \qquad \text{(D.19)}$$

This proves the theorem. ■

## Appendix E. Additional Experiment Details

### E.1 Stackelberg Markov game

For the independent policy gradient method (Daskalakis et al., 2020; Ding et al., 2022), we set the learning rate as 0.1, and both the follower and the leader use Monte Carlo sampling with trajectory length 5 and batch size 16 to estimate the policy gradient. For the PBRL algorithms, to estimate a near-optimal policy $\hat{\pi}$ at each outer iteration, we run the policy gradient algorithm for $T$ steps at every outer iteration. For PBRL with value penalty, we set learning rate 0.1, penalty constant $\lambda = 2$, inner iteration number $T = 1$, and we use Monte Carlo sampling with trajectory length 5 and batch size 16 to estimate the policy gradient. For PBRL with the Bellman penalty, we use $\lambda = 7$ and inner iteration number $T = 10$ instead.

### E.2 Deep reinforcement learning from human feedback

We conduct our experiments in the Arcade Learning Environment (ALE) (Bellemare et al., 2013) by OpenAI gymnasium similar to (Mnih et al., 2016) and (Christiano et al., 2017). For the Atari games, we use A2C, which is a synchronous version of (Mnih et al., 2016), as the policy gradient estimator in both DRLHF and PBRL. The policy and the critic share a common base model: The input is fed through 4 convolutional layers of size $8 \times 8$, $5 \times 5$, $4 \times 4$, $4 \times 4$, strides $4, 2, 1, 1$ and number of filters $16, 32, 32, 32$, with ReLU activation. This is followed by a fully connected layer of output size 256 and a ReLU non-linearity. The output of the base model is fed to a fully connected layer with scalar output as a critic, and another fully connected layer of action space size as policy. The reward predictor has the same input ($84 \times 84 \times 4$ stacked image) as the actor-critic. The input is fed through 4 convolutional layers of size $7 \times 7$, $5 \times 5$, $3 \times 3$, $3 \times 3$, strides $3, 2, 1, 1$ with 16 filters each and ReLU activation. It is followed by a fully connected layer of size 64, ReLU activation, and another fully connected layer of action space size that gives the reward function. We

use random dropout (probability 0.5) between fully connected layers to prevent over-fitting (only in reward predictor). The reward predictor and the policy are trained synchronously. The reward predictor is updated for one epoch every 300 A2C update.

We compare trajectories of 25 time steps. At the start of training, we collect 576 pairs of trajectories and warm up the reward predictor for 500 epochs. After training starts, we collect 16 new pairs per reward learning epoch. We only keep the last 3000 pairs in a buffer.

For policy learning, we set the actor-critic learning rate 0.0003, the entropy coefficient 0.01, the actor-critic batch size 16, initial upper-level loss coefficient 0.001 which decays every 3000 actor-critic gradient steps. We find out that learning is very sensitive to this coefficient, so we generally select this coefficient so that the upper-level loss converges stably; for reward learning, we set reward predictor learning rate 0.0003, reward predictor batch size 64, and the reward predictor is trained for one epoch every 500 actor-critic gradient steps. For Beamrider, we change the actor-critic learning rate to $7 \times 10^{-5}$.

### E.3 Incentive design

For the PBRL algorithms, we set the learning rate as 0.1 and a penalty constant $\lambda = 4$. The policy gradients are given by Monte Carlo sampling with trajectory length 5 and batch size 24. To obtain $\hat{\pi}_1^k, \hat{\pi}_2^k$ at each outer iteration $k$, we run the policy gradient algorithm for a single iteration with a learning rate 0.1 at every outer iteration. For the meta-gradient method, we use the same learning rate, trajectory length and batch size as PBRL. The inner iteration number is 1.

## References

A. Agarwal, S. M. Kakade, J. D. Lee, and G. Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. In *Proc. of Conference on Learning Theory*, 2020.

J. Baxter and P. L Bartlett. Infinite-horizon policy-gradient estimation. *journal of artificial intelligence research*, 15:319–350, 2001.

M. Bellemare, Y. Naddaf, J. Veness, and M. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

J. Bolte, E. Pauwels, and S. Vaiter. Automatic differentiation of nonsmooth iterative algorithms. In *Proc. of Advances in Neural Information Processing Systems*, 2022.

Z. Borsos, M. Mutny, and A. Krause. Coresets via bilevel optimization for continual learning and streaming. In *Proc. of Advances in Neural Information Processing Systems*, 2020.

S. Chakraborty, A. Bedi, A. Koppel, D. Manocha, H. Wang, M. Wang, and F. Huang. PARL: A unified framework for policy alignment in reinforcement learning. In *Proc. of International Conference on Learning Representations*, 2024.

L. Chen, S. T. Jose, I. Nikoloska, S. Park, T. Chen, and O. Simeone. Learning with limited samples: Meta-learning and applications to communication systems. *Foundations and Trends® in Signal Processing*, 17(2):79–208, 2023a.

T. Chen, Y. Sun, and W. Yin. Tighter analysis of alternating stochastic gradient method for stochastic nested problems. In *Proc. of Advances in Neural Information Processing Systems*, 2021.

X. Chen, T. Xiao, and K. Balasubramanian. Optimal algorithms for stochastic bilevel optimization under relaxed smoothness conditions. *arXiv preprint arXiv:2306.12067*, 2023b.

Z. Chen, Y. Liu, B. Zhou, and M. Tao. Caching incentive design in wireless d2d networks: A stackelberg game approach. In *IEEE International Conference on Communications*, pages 1–6, 2016.

P. F Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. *Proc. of Advances in Neural Information Processing Systems*, 2017.

F. Clarke. *Optimization and non-smooth analysis*. Wiley-Interscience, 1983.

F. H. Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

C. Daskalakis, D. J Foster, and N. Golowich. Independent policy gradient methods for competitive reinforcement learning. In *Proc. of Advances in Neural Information Processing Systems*, 2020.

D. Ding, C. Wei, K. Zhang, and M. Jovanovic. Independent policy gradient for large-scale markov potential games: Sharper rates, function approximation, and game-agnostic convergence. In *Proc. of International Conference on Machine Learning*, 2022.

A. L Dontchev and R T. Rockafellar. *Implicit functions and solution mappings: A view from variational analysis*, volume 616. Springer, 2009.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. of International Conference on Machine Learning*, 2017.

L. Franceschi, M. Donini, P. Frasconi, and M. Pontil. Forward and reverse gradient-based hyperparameter optimization. In *Proc. of International Conference on Machine Learning*, 2017.

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

S. Ghadimi and M. Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

S. Ghadimi, G. Lan, and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1):267–305, 2016.

T. Giovannelli, G. Kent, and L. Vicente. Inexact bilevel stochastic gradient methods for constrained and unconstrained lower-level problems. *arXiv preprint arXiv:2110.00604*, 2022.

R. Grazzi, L. Franceschi, M. Pontil, and S. Salzo. On the iteration complexity of hypergradient computation. In *Proc. of International Conference on Machine Learning*, pages 3748–3758, 2020.

M. Hong, H.-T. Wai, Z. Wang, and Z. Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1), 2023.

Y. Hu, W. Wang, H. Jia, Y. Wang, Y. Chen, J. Hao, F. Wu, and C. Fan. Learning to utilize shaping rewards: A new approach of reward shaping. In *Proc. of Advances in Neural Information Processing Systems*, 2020.

K. Ji, J. Yang, and Y. Liang. Provably faster algorithms for bilevel optimization and applications to meta-learning. In *Proc. of International Conference on Machine Learning*, 2021a.

K. Ji, J. Yang, and Y. Liang. Bilevel optimization: Convergence analysis and enhanced design. In *Proc. of International Conference on Machine Learning*, 2021b.

K. Ji, M. Liu, Y. Liang, and L. Ying. Will bilevel optimizers benefit from loops. In *Proc. of Advances in Neural Information Processing Systems*, 2022.

H. Jiang, Z. Chen, Y. Shi, B. Dai, and T. Zhao. Learning to defend by learning to attack. In *Proc. of International Conference on Artificial Intelligence and Statistics*, 2021.

P. Khanduri, S. Zeng, M. Hong, H.-T. Wai, Z. Wang, and Z. Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. In *Proc. of Advances in Neural Information Processing Systems*, 2021.

J. Kwon, D. Kwon, S. Wright, and R. Nowak. On penalty methods for nonconvex bilevel optimization and first-order stochastic approximation. *arXiv preprint arXiv:2309.01753*, 2023.

G. Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, 198(1), 2023.

J. Li, B. Gu, and H. Huang. A fully single loop algorithm for bilevel optimization without hessian inverse. In *Proc. of AAAI Conference on Artificial Intelligence*, 2022.

M. Littman. Friend-or-foe q-learning in general-sum games. In *Proc. of International Conference on Machine Learning*, 2001.

Q. Liu, T. Yu, Y. Bai, and C. Jin. A sharp analysis of model-based reinforcement learning with self-play. In *Proc. of International Conference on Machine Learning*, 2021a.

R. Liu, P. Mu, X. Yuan, S. Zeng, and J. Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *Proc. of International Conference on Machine Learning*, 2020.

R. Liu, Y. Liu, S. Zeng, and J. Zhang. Towards gradient-based bilevel optimization with non-convex followers and beyond. In *Proc. of Advances in Neural Information Processing Systems*, 2021b.

R. Liu, P. Mu, X. Yuan, S. Zeng, and J. Zhang. A general descent aggregation framework for gradient-based bi-level optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):38–57, 2022.

Z. Lu and S. Mei. First-order penalty methods for bilevel optimization. *arXiv preprint arXiv:2301.01716*, 2023.

Z. Luo, J. Pang, and D. Ralph. *Mathematical programs with equilibrium constraints*. Cambridge University Press, 1996.

S. Ma, Z. Chen, S. Zou, and Y. Zhou. Decentralized robust v-learning for solving markov games with model uncertainty. *Journal of Machine Learning Research*, 24(371):1–40, 2023.

D. Maclaurin, D. Duvenaud, and R. Adams. Gradient-based hyperparameter optimization through reversible learning. In *Proc. of International Conference on Machine Learning*, 2015.

J. Mei, C. Xiao, C. Szepesvari, and D. Schuurmans. On the global convergence rates of softmax policy gradient methods. In *Proc. of International Conference on Machine Learning*, 2020.

A Y. Mitrophanov. Sensitivity and convergence of uniformly ergodic markov chains. *Journal of Applied Probability*, 42(4):1003–1014, 2005.

V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proc. of International Conference on Machine Learning*, 2016.

A. Nichol, J. Achiam, and J. Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

H. Nikaidô and K. Isoda. Note on non-cooperative convex games. 1955.

L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al. Training language models to follow instructions with human feedback. In *Proc. of Advances in Neural Information Processing Systems*, 2022.

A. Pacchiano, A. Saha, and J. Lee. Dueling rl: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.

F. Pedregosa. Hyperparameter optimization with approximate gradient. In *Proc. of International Conference on Machine Learning*, 2016.

S. Qiu, Z. Yang, J. Ye, and Z. Wang. On finite-time convergence of actor-critic algorithm. *IEEE Journal on Selected Areas in Information Theory*, 2(2):652–664, 2021.

A. Rajeswaran, C. Finn, S. Kakade, and S. Levine. Meta-learning with implicit gradients. In *Proc. of Advances in Neural Information Processing Systems*, 2019.

L. J Ratliff, R. Dong, S. Sekar, and T. Fiez. A perspective on incentive design: Challenges and opportunities. *Annual Review of Control, Robotics, and Autonomous Systems*, 2: 305–338, 2019.

S. Sabach and S. Shtern. A first order method for solving convex bilevel optimization problems. *SIAM Journal on Optimization*, 27(2):640–660, 2017.

A. Shaban, C. Cheng, N. Hatch, and By. Boots. Truncated back-propagation for bilevel optimization. In *Proc. of International Conference on Artificial Intelligence and Statistics*, 2019.

L. Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10): 1095–1100, 1953.

H. Shen and T. Chen. A single-timescale analysis for stochastic approximation with multiple coupled sequences. In *Proc. of Advances in Neural Information Processing Systems*, 2022.

H. Shen and T. Chen. On penalty-based bilevel gradient descent method. *arXiv preprint arXiv:2302.05185*, 2023.

H. Shen, Z. Yang, and T. Chen. Principled penalty-based methods for bilevel reinforcement learning and rlhf. In *Proc. of International Conference on Machine Learning*, 2024.

D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

Z. Song, J. Lee, and Z. Yang. Can we find nash equilibria at a linear rate in markov games? *arXiv preprint arXiv:2303.03095*, 2023.

H. Stackelberg. *The Theory of Market Economy*. Oxford University Press, 1952.

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

R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proc. of Advances in Neural Information Processing Systems*, 2000.

A. Von Heusinger and C. Kanzow. Optimization reformulations of the generalized nash equilibrium problem using nikaido-isoda-type functions. *Computational Optimization and Applications*, 43:353–377, 2009.