On the Sample Complexity Bounds of Bilevel Reinforcement Learning

Mudit Gaur
Purdue University

Utsav Singh
IIT Kanpur

Amrit Singh Bedi*
University Of Central Florida

Raghu Pasupathy*
Purdue University

Vaneet Aggarwal*
Purdue University

Abstract

Bilevel reinforcement learning (BRL) has emerged as a powerful framework for aligning generative models, yet its theoretical foundations, especially sample complexity bounds, remain relatively underexplored. In this work, we present the first sample complexity bound for BRL, establishing a rate of $\tilde{\mathcal{O}}(\epsilon^{-3})$ in continuous state-action spaces. Traditional MDP analysis techniques do not extend to BRL due to its nested structure and non-convex lower-level problems. We overcome these challenges by leveraging the Polyak-Łojasiewicz (PL) condition and the MDP structure to obtain closed-form gradients, enabling tight sample complexity analysis. Our analysis also extends to general bi-level optimization settings with non-convex lower levels, where we achieve state-of-the-art sample complexity results of $\tilde{\mathcal{O}}(\epsilon^{-3})$ improving upon existing bounds of $\tilde{\mathcal{O}}(\epsilon^{-6})$. Additionally, we address the computational bottleneck of hypergradient estimation by proposing a fully first-order, Hessian-free algorithm suitable for large-scale problems.

1 Introduction

Bilevel reinforcement learning (BRL) has emerged as a powerful framework for modeling hierarchical decision-making processes, particularly in the context of artificial intelligence (AI) alignment. Recent works, such as those by [1, 8, 30, 35], have demonstrated the potential of bilevel formulations to address challenges in reinforcement learning from human feedback (RLHF) and inverse reinforcement learning. Despite these advancements, the theoretical understanding of BRL remains limited, especially concerning sample complexity in parameterized settings. Most existing theoretical analyses such as [40] are confined to tabular settings due to their analytical tractability, while empirical studies [9] are conducted in parameterized environments, leading to a *disconnect between theory and practice*.

A significant challenge in bridging this gap is the computational complexity associated with the hierarchical structure of BRL. Practical algorithms often circumvent the need for second-order gradient evaluations to solve bilevel problems by employing first-order approximations [1]. However, these simplification raises critical questions about the theoretical performance loss incurred by such approximations in BRL. Therefore, understanding and deriving tight sample complexity bounds for BRL are crucial for guiding the development of more efficient algorithms and for assessing the trade-offs between theoretical rigor and practical applicability. To this end, we present the first sample complexity result for BRL in continuous state action settings, achieving a bound of $\tilde{\mathcal{O}}(\epsilon^{-3})$. This result extends to standard bilevel optimization problems, providing a significant theoretical contribution wixth practical implications.

*Equal contribution.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Key challenges and our approach. The theoretical analysis of BRL is not possible using the existing theoretical frameworks [15, 29, 11, 12, 13] used to analyze MDP algorithms with a known reward function. Existing bi-level algorithms are also ill-suited to the BRL setup since they require unbiased gradients [3, 18], which are not available in the BRL setup. Many bi-level algorithms [4, 19] also require the estimation of second-order terms such as Hessian, which make them computationally infeasible as well in high-dimensional setups. Some works in the field of BRL do employ the approximation of the second-order Hessian [1, 40]. However, these works are limited to tabular state spaces. Other approaches such as [30] use a penalty based reformulation of the BRL problem. This work is still restricted to the tabular setup. From a theoretical standpoint, none of the works develop a method to analyze the sample complexity of the work for continuous state action space, works that have a sample complexity such as [40] only do so for a tabular state space. We overcome this challenge by (i) proposing a first-order BRL algorithm that works for continuous state-action spaces, (ii) providing the first-ever sample complexity results for a BRL algorithm. We use a penalized bi-level framework with non-convex lower level initially proposed in [22] for standard optimization, but it is not straightforward to apply to reinforcement learning settings, which is the main focus of our work.

In order to obtain our sample complexity result, we use the insight that the gradient parameter estimation step in the algorithm laid out in [2] (lines 3-8 of Algorithm 1) are an SGD step on a loss function that satisfies the Polyak-Łojasiewicz (PL) property. We combine this insight with our novel recursive analysis of the optimality gap (lemma 1) for stochastic gradient descent (SGD) with biased gradient estimate to obtain the first ever sample complexity result for BRL. We also demonstrate that our analysis holds for the standard bi-level penalty-based formulation of [22] with unbiased gradient estimates and provides state-of-the-art sample complexity results for the same (Theorem 1).

We obtain this result with our unique insight that the parameter estimation steps of the algorithm provided in [2], are gradient descent steps on a loss function that satisfies the Polyak-Łojasiewicz (PL) property. Additionally, we use the MDP structure to obtain closed form expressions of the gradients. This allows us to obtain the first ever sample complexity of BRL. Our analysis also applies to the general bi-level setup as a special case where we obtain state of the art sample complexity results of ϵ^{-4}.

We summarize our main contributions as follows.

- **Novel sample complexity bounds in BRL:** We derive the first sample complexity bounds for BRL with parameterized settings, achieving a bound of $\tilde{\mathcal{O}}(\epsilon^{-3})$. Our analysis addresses the challenges posed by non-convex lower-level problems and does not rely on computationally expensive second-order derivatives.
- **Generalization to standard bilevel optimization:** Our theoretical results extend beyond reinforcement learning to standard bilevel optimization problems, assuming access to unbiased gradients for the upper and lower level objectives. For setups with non-convex lower-level problems, our method achieves a state-of-the-art sample complexity of $\tilde{\mathcal{O}}(\epsilon^{-3})$.
- We perform proof of concept experiments on Mujoco Tasks to demonstrate that the proposed first order algorithm works in practice.

2 Related Works

We first go over the prevailing literature in the field of bilevel optimization. Once we have established a broad overview of the existing results in the field, we will lay out the existing results in the field of BRL and how they compare to the bilevel optimization results.

Bilevel optimization problems have been studied extensively from the theoretical perspective in recent years. Approaches such as [19] have been shown to achieve convergence, but with expensive evaluations of Hessian / Jacobian matrices and Hessian / Jacobian vector products. Works such as [33, 41] forgo the use of exact Hessian/Jacobian matrices but instead approximate them. Works such as [21] do not require even the approximation of the second-order terms. However, in all of the aforementioned works, the lower level is restricted to be convex. In general, bilevel optimization with non-convex lower-level objectives is not computationally tractable without further assumptions, even for the special case of min-max optimization [7]. Therefore, additional assumptions are necessary for the lower-level problem. The work in [22] established a penalty-based framework for solving bilevel

Table 1: This table shows a comparison of state-of-the-art sample complexity results for bilevel reinforcement learning (BRL). Our result is among the first to establish sample complexity bounds for continuous state-action spaces.

References	Continuous Space	Iteration Complexity	Sample Complexity
[31]	✗	$\tilde{\mathcal{O}}(\epsilon^{-1})$	✗
[1]	✗	$\tilde{\mathcal{O}}(\epsilon^{-1})$	✗
[40]	✗	$\tilde{\mathcal{O}}(\epsilon^{-1.5})$	$\tilde{\mathcal{O}}(\epsilon^{-3.5})$
[30]	✗	$\tilde{\mathcal{O}}(\epsilon^{-1})$	✗
This Work	✓	$\tilde{\mathcal{O}}(\epsilon^{-1})$	$\tilde{\mathcal{O}}(\epsilon^{-3})$

optimizations with a possible non-convex lower levels with the PL assumption on the lower-level function. The work in [2] obtained convergence in the bilevel setup with a non-convex lower level with an improved sample complexity with respect to [22], where it obtained ϵ^{-6} compared to ϵ^{-7}. Note that PL assumption has been used previously in fields such as optimization [20], RL [16, 14] as well as analysis of modern ML methods such as flow matching [17] and diffusion models [17, 34].

Bilevel reinforcement learning has been used in several applications such as RLHF [6, 39], reward shaping [45], Stackelberg Markov game [25, 32], AI-economics with two-level deep RL [43], social environment design [42], incentive design [5], etc. Another recent work [1] studies the policy alignment problem and introduces a corrected reward learning objective for RLHF that leads to strong performance gain. There are a very limited number of theoretical convergence results for such a setup. The PARL algorithm [1] achieves convergence of the BRL setup using the implicit gradient method that requires not only the strong convexity of the lower-level objective but also necessitates the use of second-order derivatives. Note that in general the lower level of BRL is the discounted reward which is not convex. The work of [31] employs a penalty-based framework to achieve convergence for a BRL setup using a first-order algorithm. Similarly, [40] establishes convergence by deriving an expression for the hypergradient without assuming convexity of the lower-level problem. However, it is important to note that all existing convergence results in BRL thus far do not provide sample complexity for continuous state action spaces. Despite the existence of sample complexity results for bilevel optimization with non-convex lower-level objectives in the broader bilevel literature without tabular state space restriction, such results remain absent in the context of BRL.

3 Problem Formulation

Markov Decision Process (MDP). We consider a discounted MDP defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r_\phi, \gamma)$, where $\mathcal{S}$ is a bounded measurable state space and $\mathcal{A}$ is a bounded measurable action space. We remark that in our setup, both the state and action spaces can be infinite, though they remain bounded. In the MDP, $P : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is the probability transition function and $r_\phi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ represents the parameterized reward function, ($\phi \in \Theta$) where Θ is a compact space. In order to encourage exploration, in many cases an additional KL-regularization term is preferred. This can be accounted for by defining the reward function as

$$r_\phi(s, a) = r_\phi(s, a) + \beta h_{\pi,\pi_{\text{ref}}}(s, a), \tag{1}$$

where $h_{\pi,\pi_{\text{ref}}}(s_i, a_i) = \log\left(\frac{\pi(a_i|s_i)}{\pi_{\text{ref}}(a_i|s_i)}\right)$ is the KL regularization term where π_{ref} is the reference policy. This form of the KL penalty is used in RLHF works such as in [44]. Note that our analysis works for any regularization term that is uniformly bounded. Finally, $0 < \gamma < 1$ is the discount factor. A policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ maps each state to a probability distribution over the action space. The state-action value function or Q function is defined as follows:

$$Q^\pi_\phi(s, a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_\phi(s_t, a_t) | s_0 = s, a_0 = a\right]. \tag{2}$$

For a discounted MDP, we define the optimal action value functions as

$$Q^*_\phi(s, a) = \sup_\pi Q^\pi_\phi(s, a), \qquad \forall (s, a) \in \mathcal{S} \times \mathcal{A}. \tag{3}$$

We have the expected average return given by

$$J(\phi, \lambda) = \mathbb{E}_{s\sim\nu, a\sim\pi_\lambda(.|s)}[Q_\phi^{\pi_\lambda}(s,a)], \tag{4}$$

where the policy is parameterized as $\{\pi_\lambda, \lambda \in \Lambda\}$ and Λ is a compact set.

Bilevel reinforcement learning (BRL). With the above notation in place, we can formulate the BRL problem as

$$\begin{aligned} &\min_{\phi} G(\phi,\ \lambda^*(\phi)) \\ &\text{where}\ \ \lambda^*(\phi) = \arg\min_{\lambda} -J(\phi, \lambda), \end{aligned} \tag{5}$$

where the upper-level objective $G(\phi, \lambda^*(\phi))$ is a function of the reward parameter ϕ, while the lower-level objective is a function of the policy parameter λ. We denote the lower level loss function as $-J(\phi, \lambda)$ as opposed to $J(\phi, \lambda)$ to keep our notation in line with the bi-level literature; a similar notation is followed in [31].

Existing approaches and limitations. To solve the problem in (5), one popular approach is to rewrite the problem in (5) in the following manner

$$\begin{aligned} &\min_{\phi} \Phi(\phi) := G(\phi, \lambda^*(\phi)) \\ &\text{where}\ \ \lambda^*(\phi) = \arg\min_{\lambda} -J(\phi, \lambda), , \end{aligned} \tag{6}$$

which is known as the *hyper-objective* approach, where Φ is the hyper-objective. To solve it, we need the calculation of the hyper-gradient given by

$$\nabla_\phi \Phi(\phi) = \nabla_\phi G(\phi, \lambda^*(\phi)) + v.\nabla_\lambda G(\phi, \lambda^*(\phi)), \tag{7}$$

where the term v apart from the gradient of Φ is given as

$$v = -[\nabla^2_{\lambda,} J(\phi, \lambda^*(\phi))]^{-1} \nabla^2_{\phi,\lambda} J(\phi, \lambda^*(\phi)) \tag{8}$$

This approach has been used in the existing literature [41, 33, 1]. Apart from having to calculate the Hessian and its inverse, this technique requires that the lower-level objective J be convex. One solution, which is employed in [41, 33], is to estimate first-order approximations of the Hessian. This is because the calculation of second-order terms, which in many cases can get prohibitively expensive from a computational perspective.

4 Proposed Approach

To avoid computationally expensive Hessians and for situations where the lower levels are not necessarily convex, penalty-based methods such as those developed in [22] have been proposed. Based on that, in this paper, we consider the proxy objective

$$\Phi_\sigma(\phi) = \min_{\lambda} \left(G(\phi, \lambda) + \frac{J(\phi, \lambda^*(\phi)) - J(\phi, \lambda)}{\sigma} \right), \tag{9}$$

where σ is a positive constant. The gradient of $\Phi_\sigma(\phi)$ is given by

$$\nabla_\phi \Phi_\sigma(\phi) = \nabla_\phi G(\phi, \lambda^*_\sigma(\phi)) + \frac{\nabla_\phi J(\phi, \lambda^*(\phi)) - \nabla_\phi J(\phi, \lambda^*_\sigma(\phi))}{\sigma}, \tag{10}$$

where $\lambda^*(\phi) = \arg\min_\lambda -J(\phi, \lambda)$ and $\lambda^*_\sigma(\phi) = \arg\min_\lambda -(J(\phi, \lambda) - \sigma G(\phi, \lambda))$. For future notational convenience, we define the penalty function $h_\sigma(\phi, \lambda) = J(\phi, \lambda) - \sigma G(\phi, \lambda)$. A key advantage of this formulation is the fact that, unlike the method involving the hyper-gradient, it does not require the calculation of costly second-order terms. It is also applicable to setups where the lower level is non-convex. Despite these advantages, the theoretical analysis of this setup (even for the standard bi-level framework) is not well explored.

Remark (differences with [22, 2]). Existing analyses in standard bilevel optimization settings have achieved sample complexities of $\tilde{\mathcal{O}}(\epsilon^{-7})$ and $\tilde{\mathcal{O}}(\epsilon^{-6})$ in [22] and [2], respectively. These results apply to bilevel problems without an MDP structure, where the lower-level objective is non-convex but it is reasonable to assume access to unbiased gradient estimates with bounded variance for both upper- and lower-level objectives. However, such assumptions do not hold in bilevel reinforcement learning (BRL), where gradient estimates are inherently biased due to the underlying MDP dynamics. In this work, we develop a sample complexity analysis tailored to the BRL setting. We also specialize our analysis to the standard bilevel optimization setup and demonstrate that our approach yields improved sample complexity bounds compared to prior work (see Table 2).

Algorithm development. We will describe the algorithm to solve the problem described in Equation (9). We achieve this by implementing a gradient descent step in which the gradient is given by the expression in Equation (10). In order to estimate this gradient, we have to estimate the three terms $\nabla_\phi G(\phi, \lambda^*_\sigma(\phi))$, $\nabla_\phi J(\phi, \lambda^*(\phi))$ and $\nabla_\phi J(\phi, \lambda^*_\sigma(\phi))$. In turn, these terms require the estimation of the terms $\lambda^*(\phi)$ and $\lambda^*_\sigma(\phi)$.

For the gradient of $J(\phi, \lambda)$ with respect to the upper level variable and reward parameter ϕ, note that there was no existing closed-form expression. We show in Lemma 6 in the Appendix A that a closed form of $\nabla_\phi J(\phi, \lambda)$ is given by

$$\nabla_\phi J(\phi, \lambda) = \sum_{i=1}^{\infty} \gamma^{i-1} \mathbb{E} \nabla_\phi r_\phi(s_i, a_i), \tag{11}$$

Here, the expectation is over the state action distribution induced by the policy λ. This expression is obtained by following an argument similar to the proof of the policy gradient theorem in [36]. Note that we can only obtain a truncated estimate for $\nabla_\phi J(\phi, \lambda)$, which will also lead to bias. In Algorithm 1, we take an average of this truncated estimate over B batches for a more stable estimate. We define the sample-based average here as

$$\nabla_\phi J(\phi, \lambda, B) = \frac{1}{B} \sum_{j=1}^{B} \nabla_\phi \hat{J}_j(\phi, \lambda). \tag{12}$$

where $\nabla_\phi \hat{J}_j(\phi, \lambda) = \sum_{j=1}^{H} \nabla_\phi r_\phi(s_{j,i}, a_{j,i})$. Here, $(s_{j,i}, a_{j,i})$ are the i^{th} state-action pair of the j^{th} trajectory sampled from the policy π_λ.

For the gradient for the lower-level loss function gradient $J(\phi, \lambda)$ with respect to the lower-level variable λ we use the policy gradient function to obtain

$$\begin{aligned}\nabla_\lambda J(\phi, \lambda) &= \mathbb{E}_{(s,a) \sim d_\nu^{\pi_\lambda}} [\nabla_\lambda \log \pi_\lambda(a|s) Q_\phi^\lambda(s, a)] \\ &\quad + \mathbb{E}_{(s_i, a_i \sim \pi_\lambda)} \beta \sum_{i=1}^{\infty} \gamma^{i-1} \nabla_\lambda h_{\pi_\lambda, \pi_{ref}}(s_i, a_i)\end{aligned} \tag{13}$$

Here $d_\nu^{\pi_\lambda}$ denotes the stationary distribution of the state action space induced by the policy π_λ. The second term on the right-hand side is due to the presence of the KL regularization term in the reward $r(\phi)$. Note that in real-world applications of RL algorithms, such as actor-critic, the estimate of Q_ϕ^λ is not an unbiased estimate, but instead a parametrized function, such as a neural network, is used to approximate it, leading to bias. Additionally we cannot sample the infinite sum $\mathbb{E}_{(s_i, a_i \sim \pi_\lambda)} \beta \sum_{i=1}^{\infty} \nabla_\lambda h_{\pi_\lambda, \pi_{ref}}(s_i, a_i)$ but have to get a finite truncated estimate, which also leads to bias.We denote by $\nabla_\lambda \hat{J}(\phi, \lambda, n, B)$ the estimate of $\nabla_\lambda J(\phi, \lambda)$ as

$$\begin{aligned}\nabla_\lambda \hat{J}(\phi, \lambda, n, B) &= \frac{1}{n} \sum_{i=1}^{n} [\nabla_\lambda \log \pi_\lambda(a_i|s_i) \hat{Q}_\phi^\lambda(s_i, a_i)] \\ &\quad + \frac{\beta}{B} \sum_{j=1}^{B} \sum_{i=1}^{H} \gamma^{i-1} \nabla_\lambda h_{\pi_\lambda, \pi_{ref}}(s_{j,i}, a_{j,i})\end{aligned} \tag{14}$$

Note that the estimate of $Q_\phi^\lambda(s, a)$ denoted by $\hat{Q}_\phi^\lambda(s, a)$ is estimated using n samples. For upper-level loss functions, unbiased gradient estimates can be calculated, as demonstrated in [1]. For notational convenience, we define

$$\nabla G(\phi, \lambda, B) = \frac{1}{B} \sum_{i=1}^{B} \nabla \hat{G}_i(\phi, \lambda), \tag{15}$$

where B is the size of the gradient sample dataset and $\nabla \hat{G}_i(\phi, \lambda)$ is the gradient estimate sample i^{th}. Note here that the batch size B and horizon length H can vary across the different gradients. We keep this notation the same across gradients with respect to ϕ and λ for notational convenience.

Now that we have expressions for the gradients of the upper and lower level function, we now move onto the estimation of $\nabla_\phi J(\phi, \lambda^*(\phi))$ and $\nabla_\phi J(\phi, \lambda_\sigma^*(\phi))$. Consider the term $\lambda_\sigma^*(\phi)$ which is a minimizer of the function given by $h_\sigma(\phi, \lambda)$. Thus, it is obtained by performing a gradient descent on $h_\sigma(\phi, \lambda)$ with respect to λ. Similarly, $\lambda^*(\phi)$ is the minimizer of the function given by $J(\phi, \lambda)$ and can be obtained by gradient descent. Note that these steps are performed on lines 4-7 of Algorithm 1. The gradient descent step for the proxy loss function $\Phi_\sigma(\phi)$ is performed on line 11. We estimate the gradients of $G(\phi, \lambda)$ and $J(\phi, \lambda)$ with respect to ϕ using the expression in Equations (11) and (15).

Algorithm 1 A first-order approach to bilevel RL

1: **Input:** $\mathcal{S}$, $\mathcal{A}$, Time Horizon $T \in \mathcal{Z}$, Number of gradient estimation updates for lower level $K \in \mathcal{Z}$, sample batch size $n \in \mathcal{Z}$, gradient batch size $B \in \mathcal{Z}$, Horizon length $H \in \mathcal{Z}$, starting policy parameters $\lambda_0^0, {\lambda'}_0^0$, starting reward parameter ϕ_0
2: **for** $t \in \{0, \cdots, T-1\}$ **do**
3:     **for** $k \in \{0, \cdots, K-1\}$ **do**
4:         $d_k = \nabla_\lambda \hat{J}(\lambda_t^k, \phi_t, n, B)$
5:         $d_k^{'} = \nabla_\lambda \hat{J}({\lambda'}_t^k, \phi_t, n, B) - \sigma.\nabla_\lambda \hat{G}(\phi_t, {\lambda'}_t^k, B)$
6:         $\lambda_t^{k+1} = \lambda_t^k + \tau \cdot \frac{d_k}{||d_k||}$
7:         ${\lambda'}_t^{k+1} = {\lambda'}_t^k + \tau^{'} \cdot \frac{d_k^{'}}{||d_k^{'}||}$
8:     **end for**
9:     $d_t = \nabla_\phi \hat{G}(\phi_t, {\lambda'}_t^K, B) - \frac{1}{\sigma} \left(\nabla_\phi \hat{J}(\phi_t, \lambda_t^K, B) - \nabla_\phi \hat{J}(\phi_t, {\lambda'}_t^K, B) \right)$
10:     $\phi_{t+1} = \phi_t - \eta \cdot d_t$
11: **end for**

5 Theoretical Analysis

We begin by outlining the assumptions required for our analysis, followed by the presentation of our convergence results. We then provide a detailed theoretical analysis, explaining the derivation of these results.

Assumption 1. *For any $\phi \in \Theta$, $\lambda \in \Lambda$ and $\sigma \in \mathbb{R}^+$, we have the following assumptions*

1. *For all $0 \leq \sigma \leq \sigma_0$, the function $h_\sigma(\phi, \lambda)$ satisfies the inequality*

$$||\nabla h_\sigma(\phi, \lambda)||^2 \leq \mu(h_\sigma(\phi, \lambda) - h_\sigma(\phi, \lambda_\sigma^*)) \tag{16}$$

where $\lambda_\sigma^* = \arg\min_{\lambda \in \Lambda}(h_\sigma(\phi, \lambda))$ and σ_0 is a positive constant.

2. *The functions $h_\sigma(\phi, \lambda)$ and $J(\phi, \lambda)$ are Lipschitz and smooth in variables ϕ and λ.*

3. *The functions $h_\sigma(\phi, \lambda)$ and $J(\phi, \lambda)$ have Lipschitz and smooth Hessians in both ϕ and λ.*

In [22], the first Assumption in Equation (16) was shown to ensure that the proxy objective $\phi_\lambda(\phi)$ is differentiable. This assumption also exists in the literature [2] to ensure the existence of the gradient given in Equation (10). It is thus key for the setup given in Equation (9) to be solvable using gradient descent. The Assumption 1.2 is a standard assumption in bi-level literature used for convergence analyses [18, 2]. The Assumption 1.3 ensures that solving for the optimal point of the proxy objective Φ_σ brings us close the optimal point of the true objective Φ.

Assumption 2. *For any fixed $\lambda \in \Lambda, \phi \in \Phi$ and $\theta \in \Theta$ be the parameters of the neural network class used to parametrize the Q, where Θ is a compact set, and μ is a distribution over $\mathcal{S} \times \mathcal{A}$. Then it holds that*

$$\min_{\theta \in \Theta} \mathbb{E}_{s,a \sim \mu} \left(Q_\theta(s,a) - Q_\phi^{\pi_\lambda}(s,a) \right)^2 \le \epsilon_{approx}.$$

Assumption 2 ensures that a class of neural networks is able to approximate the function obtained by applying the Bellman operator to a neural network of the same class. Similar assumptions are also considered in [10, 38, 15]. This assumption ensures that we are able to find an accurate estimate of the Q function. This assumption accounts for the bias in gradient estimation, something not present in the standard bi-level setup. In works such as [30] a similar constant denoted by ϵ_{oracle} is used

Assumption 3 (For upper level)**.** *For any fixed $\lambda, \lambda_1, \lambda_2 \in \Lambda$, $\phi, \phi_1, \phi_2 \in \Theta$ and $(s,a) \in \mathcal{S} \times \mathcal{A}$, we have the following properties*

1. $||\nabla r_\phi(s,a)|| \le C_1$
2. $||\nabla \log \pi_\lambda(s,a)|| \le C_2$
3. $||\nabla r_{\phi_1}(s,a) - \nabla r_{\phi_2}(s,a)|| \le C_3 ||\phi_1 - \phi_2||$
4. $||\nabla \log \pi_{\lambda_1}(s,a) - \nabla \log \pi_{\lambda_2}(s,a)|| \le C_4 ||\lambda_1 - \lambda_1||$

where $C_1 - C_5$ and $C_2 \ge 1$ are positive constants. Additionally, there exist $\varepsilon, \bar{\varepsilon} \in (0,1]$ such that $\pi_\lambda(a \mid s) \ge \varepsilon$ for all $a \in \mathcal{A}$ and $\lambda \in \Lambda$, and $\pi_{ref}(a \mid s) \ge \bar{\varepsilon}$ for all $a \in \mathcal{A}$

Similar assumptions have been utilized in prior policy gradient-based works [26, 29], as well as actor critic algorithms, such as [10, 15, 11].

Assumption 4 (For upper level)**.** *For any fixed $\lambda \in \Lambda$ and $\phi \in \Theta$ we have access to unbiased gradients*

$$\mathbb{E}[\nabla \hat{G}(\phi, \lambda)] = \nabla G(\phi, \lambda) \tag{17}$$

and the gradient estimates have bounded variance

$$\mathbb{E}\|\nabla \hat{G}(\phi, \lambda) - \mathbb{E}[\nabla(G)(\phi, \lambda)]\|^2 \le \sigma_G^2 \tag{18}$$

The assumption for an unbiased gradient with bounded variance is present both in bilevel literature [22, 2] as well as BRL literature [1]. Works such as [31] simply assume access to exact gradients of the upper loss function.

Main Result: With all the assumptions in place, we are now ready to present the main theoretical results of this work. First, we will state the convergence result for Algorithm 1. This result establishes the sample complexity bounds for BRL which are the first such results of it's kind. Then, we will go into detail about how these results are obtained, by providing a brief overview of the techniques and lemmas used in establishing the convergence result.

Theorem 1. *Suppose Assumptions 1-4 hold and we have $0 < \eta \le \frac{1}{2L}$, $0 \le \tau \le \frac{1}{L_J}$, $0 \le \tau' \le \frac{1}{L_h}$ where L, L_J, L_σ are the smoothness constants of Φ_σ,J and h_σ respectively. Then from Algorithm 1, we obtain*

$$\frac{1}{T}\sum_{t=1}^{T} \|\nabla \Phi(\phi_t)\|^2 \le \tilde{\mathcal{O}}\left(\frac{1}{T}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-k}}{\sigma^2}\right) + \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 n}\right) + \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{\sigma^2 B}\right) + \tilde{\mathcal{O}}(\sigma^2) \tag{19}$$

$$+ \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{20}$$

If we set $\sigma^2 = \tilde{\mathcal{O}}(\epsilon)$, $B = \tilde{\mathcal{O}}(\epsilon^{-2})$, $n = \tilde{\mathcal{O}}(\epsilon^{-2})$, $T = \tilde{\mathcal{O}}(\epsilon^{-1})$, $K = \tilde{\mathcal{O}}(\log\left(\frac{1}{\epsilon}\right))$ and $H = \tilde{\mathcal{O}}(\log\left(\frac{1}{\epsilon}\right))$ then we obtain

$$\frac{1}{T}\sum_{t=1}^{T} \|\nabla \Phi(\phi_t)\|^2 \le \tilde{\mathcal{O}}(\epsilon) + \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{21}$$

This gives us a sample complexity of $n.K.T + B.K.H.T + B.H.T = \tilde{\mathcal{O}}(\epsilon^{-3})$.

Thus we have obtained the first ever sample complexity result for BRL setup. Notably, this result improves on works such as [1, 31] in that our result does not require the state or action space to be finite, while also providing sample complexity and not just iteration complexity results.

5.1 Proof sketch of Theorem 1:

The proof is divided into two main parts. The first part is where we establish the local convergence bound of the upper loss function in terms of the error in estimating the gradient of Φ_σ as given in Equation (13). This is done using the smoothness assumption on Φ. The next step is to upper bound the error incurred in estimating the gradient of Φ_σ. The gradient estimation error is shown to be composed of estimating the three terms on the right-hand side of Equation (9). The error in estimating each term is shown to be composed in estimating $\lambda^*_\sigma(\phi)$ (or $\lambda^*(\phi)$) and the error due to having access to an empirical estimate of the gradient. In the estimation of $\lambda^*_\sigma(\phi)$ (or $\lambda^*(\phi)$). A key insight here is to recognize that in the inner loop of Algorithm 1 we are performing a gradient descent with respect to the parameter λ on the functions $J(\phi, \lambda)$ and $h_\sigma(\phi, \lambda)$. We use this insight in combination with the PL property from Assumption 1 to upper bound the error in estimating $\lambda^*_\sigma(\phi)$ (or $\lambda^*(\phi)$).

Establishing local convergence bound for Φ: Under Assumption 1, from the smoothness of Φ, we have

$$\Phi(\phi_{t+1}) \leq \Phi(\phi_t) + \langle \nabla_\phi \Phi(\phi_t), \phi_{t+1} - \phi_t \rangle + L\|\phi_{t+1} - \phi_t\|^2, \tag{22}$$

Now, with a step size $\eta \leq \frac{1}{2L}$, where α_1 is the smoothness parameter of Φ, we get

$$\begin{aligned}\frac{1}{T}\sum_{i=1}^{T} \|\nabla \Phi(\phi_t)\|^2 \leq & \frac{1}{T}\sum_{t=0}^{t=T} \mathbb{E}\|\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t)\|^2 + \tilde{\mathcal{O}}\left(\frac{1}{T}\right) \\ & + \tilde{\mathcal{O}}(\sigma^2). \end{aligned} \tag{23}$$

Note that $\nabla\hat{\Phi}_\sigma$ denotes the empirical estimate of the gradient of the proxy loss function Φ_σ. Note that we get the term $\tilde{\mathcal{O}}(\sigma^2)$ using Lemma 4.3 from [2].

Gradient estimation error: The error in the estimation of the gradient at each iteration k of Algorithm 1 given by $\|\nabla_\phi \Phi(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))\|$, which is the error between the gradient of the upper objective $\nabla_\phi \Phi(\phi_t)$ and our estimate of the gradient of the pseudo-objective $\nabla_\phi \hat{\Phi}_\sigma(\phi_t))$. This error is decomposed as follows.

$$\begin{aligned}\underbrace{\mathbb{E}\|\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))\|}_{A'_k} \leq & \mathbb{E}\|\nabla_\phi G(\phi_t, \lambda^*_\sigma(\phi)) - \nabla_\phi G(\phi_t, {\lambda'}^K_t, B)\| \\ & + \frac{1}{\sigma}\mathbb{E}\|\nabla_\phi J(\phi_t, \lambda^*(\phi)) - \nabla_\phi J(\phi_t, \lambda^K_t, B)\| \\ & + \frac{1}{\sigma}\mathbb{E}\|\nabla_\phi J(\phi_t, \lambda^*_\sigma(\phi)) - \nabla_\phi J(\phi_t, {\lambda'}^K_t, B)\|. \end{aligned} \tag{24}$$

Thus, the error incurred in the estimation of the gradient terms can be broken into the error in estimation of the three terms, $\nabla G(\phi_t, \lambda^*_\sigma(\phi_t))$, $\nabla J(\phi_t, \lambda^*(\phi_t))$ and $\nabla J(\phi_t, \lambda^*_\sigma(\phi_t))$. We first focus on the estimation error for the term $\nabla_\phi J(\phi, \lambda^*_\sigma(\phi))$ where the error in estimation can be decomposed as

$$\begin{aligned}\mathbb{E}\|\nabla_{\phi_t} J(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}^K_t, B)\| \leq & \mathbb{E}\|\nabla_\phi J(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}^K_t)\| \\ & + \mathbb{E}\|\nabla_\phi J(\phi_t, {\lambda'}^K_t) - \nabla_\phi J(\phi_t, {\lambda'}^K_t, B)\|. \end{aligned} \tag{25}$$

The second term on the right-hand side of Equation (25) is the error incurred due to the difference between the gradient of J and its empirical estimate. This error is upper bounded using the defintion of the gradient given in Equation (11).

The first term on the right-hand side is the error incurred due to the error in estimating $\lambda^*_\sigma(\phi)$. In order to show this, we write the following

$$\mathbb{E}\|\nabla_\phi J(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}^K_t)\|^2 \leq L_J \mathbb{E}\|\lambda^*_\sigma(\phi_t) - {\lambda'}^K_t\|^2 \tag{26}$$

$$\leq L_\sigma \cdot \mu \mathbb{E}|h_\sigma(\phi_t, \lambda^*_\sigma(\phi_t)) - h_\sigma(\phi_t, {\lambda'}^K_t)|. \tag{27}$$

We get Equation (26) from the smoothness of $J(\phi, \lambda)$ assumed in Assumption 1. We get Equation (27) from Equation (26) by using the quadratic growth property of PL functions applied to $h_\sigma(\phi, \lambda)$ also assumed in Assumption 1.

In order to bound the right hand side of Equation (27), we establish the following result.

Lemma 1. *Consider an L-smooth differentiable function denoted by $f(\lambda)$ satisfying the PL property with PL constant μ. If we apply the stochastic gradient descent with step size $0 \leq \eta \leq \frac{1}{L}$, then we obtain the following*

$$(f(\lambda_k) - f(\lambda^*)) = \tilde{\mathcal{O}}\left(e^{-k}\right) + \mathcal{O}(\beta(n, B, H)) \tag{28}$$

where $\forall \lambda \in \Lambda$, $\beta(n)$ satisfies

$$\mathbb{E}\|\nabla_\lambda f(\lambda_k) - \nabla_\lambda \hat{f}(\lambda_k)\|^2 \leq \beta(n, B, H) \tag{29}$$

and $\nabla_\lambda \hat{f}(\lambda)$ denotes the estimate of $\nabla_\lambda f(\lambda)$ and $\lambda^* = argmin_{\lambda \in \Lambda} f(\lambda)$.

This result is obtained using a recursive analysis of the optimality gap when performing an SGD in the presence of biased gradient estimates. Using this lemma, we can bound the right-hand side of Equation (27) in terms of error in estimating the gradient of h_σ with respect to λ. Thus, we obtain

$$\mathbb{E}|h_\sigma(\phi_t, \lambda^*_\sigma(\phi_t)) - h_\sigma(\phi_t, {\lambda'}^K_t)| \leq \tilde{\mathcal{O}}\left(e^{-K}\right) + \mathcal{O}(\beta(n, B, H)) \tag{30}$$

where $\forall \lambda \in \Lambda$, $\beta(n, B, H)$ satisfies $\mathbb{E}\|\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)\|^2 \leq \beta(n, B, H)$. Using the expression for gradients of $J(\phi, \lambda)$ and $G(\phi, \lambda)$ we are able obtain the following result

$$\mathbb{E}\|\nabla_\phi J(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}^K_t)\|^2 \leq \tilde{\mathcal{O}}\left(e^{-k}\right) + \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{31}$$

where n is the number of samples used to estimate the Q function. The details of this are given in Lemma 5 of the Appendix. For upper bounding the other two terms on the right-hand side of Equation (24), we use a similar decomposition and analysis. These are described in detail in Lemma 3 and Lemma 4 of the Appendix. Finally, plugging the obtained expressions back into the right-hand side of Equation (24) and the resulting expression into the right-hand side of Equation (23) gives us Theorem 1. We provide an evaluation of Algorithm 1 in Appendix F.

6 Standard Bilevel Optimization: A Special Case

In this section, we show how the techniques used to establish Theorem 1 can also yield a state-of-the-art sample complexity result for standard bilevel optimization with a non-convex lower level (where the lower level is not an RL problem). The key distinction between our BRL setup and standard bilevel optimization is that it is assumed that we have access to unbiased gradients with bounded variance [22, 2]. This is not the case in the BRL setup as discussed in Section 4. We show that assuming access to unbiased gradients with bounded variance enables achieving a state-of-the-art sample complexity result for bilevel optimization.

The bilevel optimization problem is similar to (6), and is given as

$$\min_\phi \Phi(\phi) := G(\phi, \lambda \in \Lambda^*(\phi)),$$
$$\text{where } \Lambda^* \in \arg\min_\lambda -J(\phi, \lambda). \tag{32}$$

As before, we solve the proxy problem in Equation (9) using gradient descent with the gradient expression from Equation (10). The key difference here is the availability of unbiased gradients for both the upper- and lower-level loss functions, as captured in the following assumption.

Assumption 5. *For any fixed $\lambda \in \Lambda$ and $\phi \in \Theta$ we have access to unbiased gradients*

$$\mathbb{E}[\nabla \hat{G}(\phi, \lambda)] = \nabla G(\phi, \lambda), \tag{33}$$
$$\mathbb{E}[\nabla \hat{J}(\phi, \lambda,)] = \nabla G(\phi, \lambda) \tag{34}$$

and the gradient estimates have bounded variance

$$\mathbb{E}\|\nabla \hat{G}(\phi, \lambda) - \mathbb{E}\nabla(G)(\phi, \lambda)\|^2 \leq \sigma_G^2, \tag{35}$$
$$\mathbb{E}\|\nabla \hat{J}(\phi, \lambda) - \mathbb{E}\nabla(G)(\phi, \lambda)\|^2 \leq \sigma_J^2 \tag{36}$$

Table 2: If we assume access to unbiased gradients, we obtain a state of the art sample complexity of ϵ^{-3} for bilevel optimization without lower level convexity restriction.

References	Non-convex LL	Without second order	Iteration complexity	Sample complexity
[19]	✗	✗	$\tilde{\mathcal{O}}(\epsilon^{-1})$	$\tilde{\mathcal{O}}(\epsilon^{-2})$
[33]	✗	✓	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-4})$
[21]	✗	✓	$\tilde{\mathcal{O}}(\epsilon^{-\frac{5}{2}})$	$\tilde{\mathcal{O}}(\epsilon^{-\frac{5}{2}})$
[41]	✗	✓	$\tilde{\mathcal{O}}(\epsilon^{-\frac{3}{2}})$	$\tilde{\mathcal{O}}(\epsilon^{-\frac{3}{2}})$
[22]	✓	✓	$\tilde{\mathcal{O}}(\epsilon^{-5})$	$\tilde{\mathcal{O}}(\epsilon^{-7})$
[2]	✓	✓	$\tilde{\mathcal{O}}(\epsilon^{-2})$	$\tilde{\mathcal{O}}(\epsilon^{-6})$
This Work	✓	✓	$\tilde{\mathcal{O}}(\epsilon^{-1})$	$\tilde{\mathcal{O}}(\epsilon^{-3})$

This provides the gradient estimate for the lower-level loss function, and Equation (15) is the gradient estimate for the upper-level loss function. Here, $\nabla \hat{J}_i(\phi, \lambda)$ are independent sampled unbiased estimates of $\nabla J(\phi, \lambda)$, and B represents the batch size. We assume that these samples of the estimate can be independently sampled. Additionally, we assume that this can be done for the gradient with respect to both λ and ϕ. This is in line with other BRL works such as [1, 31]. We also define the following term

$$\nabla J(\phi, \lambda, B) = \frac{1}{B}\sum_{j=1}^{B} \nabla \hat{J}_j(\phi, \lambda). \tag{37}$$

which is what we use instead of $\nabla J(\phi, \lambda, n, B)$ in Algorithm 1 for the standard bi-level setup. For a bi-level optimization with a non-convex lower level, we obtain

Theorem 2. *Suppose Assumptions 1 and 5 hold and we have* $0 < \eta \leq \frac{1}{2L}$, $0 \leq \tau_k \leq \frac{1}{L_J}$, $0 \leq \tau_k \leq \frac{1}{L_h}$ *where* L, L_J, L_σ *are the smoothness constants of* Φ_σ, J *and* h_σ *respectively. We further replace* $\nabla_\lambda J(\phi, \lambda, n, B)$ *with* $\nabla J(\phi, \lambda, B)$ *as defined in* (37)*. Then, from Algorithm 1 we obtain*

$$\frac{1}{T}\sum_{t=1}^{T} \|\nabla \Phi(\phi_t)\|^2 \leq \tilde{\mathcal{O}}\left(\frac{1}{T}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-k}}{\sigma^2}\right) + \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 B}\right) + \tilde{\mathcal{O}}(\sigma^2) \tag{38}$$

If we set $\sigma^2 = \tilde{\mathcal{O}}(\epsilon)$, $B = \tilde{\mathcal{O}}(\epsilon^{-2})$, $T = \tilde{\mathcal{O}}(\epsilon^{-1})$, $K = \tilde{\mathcal{O}}(\log\left(\frac{1}{\epsilon}\right))$.

$$\frac{1}{T}\sum_{t=1}^{T} \|\nabla \Phi(\phi_t)\|^2 \leq \tilde{\mathcal{O}}(\epsilon) \tag{39}$$

This gives us a sample complexity of $B.K.T + B.T = \tilde{\mathcal{O}}(\epsilon^{-3})$.

Note the absence of the term $\mathcal{O}(\epsilon_{\text{approx}})$ as we have assumed access to unbiased gradient estimates for both upper and lower loss functions. As noted earlier, our result advances previous analyses of bi-level optimization with non-convex lower levels. [22] established a sample complexity of $\tilde{\mathcal{O}}(\epsilon^{-7})$, later improved to $\mathcal{O}(\epsilon^{-6})$ by [2]. Table 2 highlights how our approach enhances existing results in bi-level optimization and brings convergence results from non-convex lower level setups to those of convex lower level setups such as [18, 41].

7 Conclusion

This paper established the first sample complexity bounds for bilevel reinforcement learning (BRL) in parameterized settings, achieving $O(\epsilon^{-3})$. Our approach, leveraging penalty-based formulations and first-order methods, improves scalability without requiring costly Hessian computations. These results extend to standard bilevel optimization, setting a new state-of-the-art for non-convex lower-level problems. Our work provides a foundation for more efficient BRL algorithms with applications in AI alignment and RLHF. Future direction include improving the theoretical bounds in this paper, and evaluating the proposed algorithm in different applications.

8 Acknowledgment

The work was supported in part by the National Science Foundation under grant CCF-2149588 and Cisco Systems, Inc.

References

[1] Souradip Chakraborty, Amrit Bedi, Alec Koppel, Huazheng Wang, Dinesh Manocha, Mengdi Wang, and Furong Huang. Parl: A unified framework for policy alignment in reinforcement learning. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[2] Lesi Chen, Jing Xu, and Jingzhao Zhang. On finding small hyper-gradients in bilevel optimization: Hardness results and improved analysis. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 947–980, 2024.

[3] Tianyi Chen, Yuejiao Sun, Quan Xiao, and Wotao Yin. A single-timescale method for stochastic bilevel optimization. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 2466–2488, 2022.

[4] Xuxing Chen, Minhui Huang, Shiqian Ma, and Krishna Balasubramanian. Decentralized stochastic bilevel optimization with improved per-iteration complexity. In *International Conference on Machine Learning*, pages 4641–4671, 2023.

[5] Zhuoqun Chen, Yangyang Liu, Bo Zhou, and Meixia Tao. Caching incentive design in wireless d2d networks: A stackelberg game approach. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6, 2016.

[6] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[7] Constantinos Daskalakis, Stratis Skoulakis, and Manolis Zampetakis. The complexity of constrained min-max optimization. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1466–1478, 2021.

[8] Mucong Ding, Souradip Chakraborty, Vibhu Agrawal, Zora Che, Alec Koppel, Mengdi Wang, Amrit Bedi, and Furong Huang. Sail: Self-improving efficient online alignment of large language models. *arXiv preprint arXiv:2406.15567*, 2024.

[9] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adverserial inverse reinforcement learning. In *International Conference on Learning Representations*, 2018.

[10] Zuyue Fu, Zhuoran Yang, and Zhaoran Wang. Single-timescale actor-critic provably finds globally optimal policy. In *International Conference on Learning Representations*, 2021.

[11] Swetha Ganesh, Jiayu Chen, Washim Uddin Mondal, and Vaneet Aggarwal. Order-optimal global convergence for actor-critic with general policy and neural critic parametrization. In *The 41st Conference on Uncertainty in Artificial Intelligence*, 2025.

[12] Swetha Ganesh, Washim Uddin Mondal, and Vaneet Aggarwal. Order-optimal regret with novel policy gradient approaches in infinite-horizon average reward mdps. In *International Conference on Artificial Intelligence and Statistics*, pages 3421–3429. PMLR, 2025.

[13] Swetha Ganesh, Washim Uddin Mondal, and Vaneet Aggarwal. A sharper global convergence analysis for average reward reinforcement learning via an actor-critic approach. In *Forty-second International Conference on Machine Learning*, 2025.

[14] Mudit Gaur, Vaneet Aggarwal, and Mridul Agarwal. On the global convergence of fitted q-iteration with two-layer neural network parametrization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 11013–11049. PMLR, 23–29 Jul 2023.

[15] Mudit Gaur, Amrit Bedi, Di Wang, and Vaneet Aggarwal. Closing the gap: Achieving global convergence (Last iterate) of actor-critic under Markovian sampling with neural network parametrization. In *Proceedings of the 41st International Conference on Machine Learning*, pages 15153–15179, 2024.

[16] Mudit Gaur, Amrit Singh Bedi, Di Wang, and Vaneet Aggarwal. On the global convergence of natural actor-critic with two-layer neural network parametrization. *arXiv preprint arXiv:2306.10486*, 2023.

[17] Mudit Gaur, Prashant Trivedi, Shuchin Aeron, Amrit Singh Bedi, George K Atia, and Vaneet Aggarwal. Generative modeling with continuous flows: Sample complexity of flow matching. *arXiv preprint arXiv:2512.01286*, 2025.

[18] Riccardo Grazzi, Massimiliano Pontil, and Saverio Salzo. Bilevel optimization with a lower-level contraction: Optimal sample complexity without warm-start. *Journal of Machine Learning Research*, 24(167):1–37, 2023.

[19] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*, pages 4882–4892, 2021.

[20] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 795–811. Springer, 2016.

[21] Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method for stochastic bilevel optimization. In *International Conference on Machine Learning*, pages 18083–18113. PMLR, 2023.

[22] Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. On penalty methods for nonconvex bilevel optimization and first-order stochastic approximation. In *The Twelfth International Conference on Learning Representations*, 2024.

[23] Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[24] Kimin Lee, Laura M Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *Proceedings of the 38th International Conference on Machine Learning*, pages 6152–6163, 2021.

[25] Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model-based reinforcement learning with self-play. In *International Conference on Machine Learning*, pages 7001–7010, 2021.

[26] Saeed Masiha, Saber Salehkaleybar, Niao He, Negar Kiyavash, and Patrick Thiran. Stochastic second-order methods improve best-known sample complexity of SGD for gradient-dominated functions. In *Advances in Neural Information Processing Systems*, 2022.

[27] Reginald McLean, Evangelos Chatzaroulas, Luc McCutcheon, Frank Röder, Tianhe Yu, Zhanpeng He, K. R. Zentner, Ryan Julian, J K Terry, Isaac Woungang, Nariman Farsad, and Pablo Samuel Castro. Meta-world+: An improved, standardized, rl benchmark. *arXiv preprint arXiv:2505.11289*, 2025.

[28] Katherine Metcalf, Miguel Sarabia, Natalie Mackraz, and Barry-John Theobald. Sample-efficient preference-based reinforcement learning with dynamics aware rewards. In *Proceedings of The 7th Conference on Robot Learning*, pages 1484–1532, 2023.

[29] Washim U Mondal and Vaneet Aggarwal. Improved sample complexity analysis of natural policy gradient algorithm with general parameterization for infinite horizon discounted reward markov decision processes. In *International Conference on Artificial Intelligence and Statistics*, pages 3097–3105, 2024.

[30] Han Shen and Tianyi Chen. On penalty-based bilevel gradient descent method. In *International Conference on Machine Learning*, pages 30992–31015, 2023.

[31] Han Shen, Zhuoran Yang, and Tianyi Chen. Principled penalty-based methods for bilevel reinforcement learning and rlhf. *Journal of Machine Learning Research*, 26(114):1–49, 2025.

[32] Zhuoqing Song, Jason D. Lee, and Zhuoran Yang. Can we find nash equilibria at a linear rate in markov games? In *The Eleventh International Conference on Learning Representations*, 2023.

[33] Daouda Sow, Kaiyi Ji, and Yingbin Liang. On the convergence theory for hessian-free bilevel algorithms. *Advances in Neural Information Processing Systems*, 35:4136–4149, 2022.

[34] Aadithya Srikanth, Mudit Gaur, and Vaneet Aggarwal. Discrete state diffusion models: A sample complexity perspective. *arXiv preprint arXiv:2510.10854*, 2025.

[35] Saksham Sahai Srivastava and Vaneet Aggarwal. A technical survey of reinforcement learning techniques for large language models. *arXiv preprint arXiv:2507.04136*, 2025.

[36] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

[37] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

[38] Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural policy gradient methods: Global optimality and rates of convergence. *International Conference on Learning Representations*, 2020.

[39] Yichong Xu, Ruosong Wang, Lin Yang, Aarti Singh, and Artur Dubrawski. Preference-based reinforcement learning with finite-time guarantees. *Advances in Neural Information Processing Systems*, 33:18784–18794, 2020.

[40] Yan Yang, Bin Gao, and Ya-xiang Yuan. Bilevel reinforcement learning via the development of hyper-gradient without lower-level convexity. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, pages 4780–4788, 2025.

[41] Yifan Yang, Peiyao Xiao, and Kaiyi Ji. Achieving $\mathcal{O}(\epsilon^{-1.5})$ complexity in hessian/jacobian-free stochastic bilevel optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[42] Edwin Zhang, Sadie Zhao, Tonghan Wang, Safwan Hossain, Henry Gasztowtt, Stephan Zheng, David C Parkes, Milind Tambe, and Yiling Chen. Social environment design. *arXiv preprint arXiv:2402.14090*, 2024.

[43] Stephan Zheng, Alexander Trott, Sunil Srinivasa, David C Parkes, and Richard Socher. The ai economist: Taxation policy design via two-level deep multiagent reinforcement learning. *Science advances*, 8(18):eabk2607, 2022.

[44] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

[45] Haosheng Zou, Tongzheng Ren, Dong Yan, Hang Su, and Jun Zhu. Learning task-distribution reward shaping with meta-learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):11210–11218, May 2021.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims are demonstrated in the key results in Lemmas and Theorems, with explanations next to them.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The assumptions given in the paper give the limitations of this work. Further, future work direction in the conclusions describe another limitation of this work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We have provided the assumptions used in the work at one place, which are used in all the results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Details provided in Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Details provided in Appendix F.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Details provided in Appendix F.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Details provided in Appendix F.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details provided in Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: All the points mentioned in the NeurIPS Code of Ethics are taken into consideration.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Since the work is primarily theoretical in nature, no potential negative societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing works used are properly credited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Details provided in Appendix F

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does neither involve crowd-sourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

A Proof of Lemma 1

Proof.

$$\lambda_{t+1} \;=\; \lambda_t \;-\; \eta\, \nabla \hat{f}(\lambda_t),$$

where the biased stochastic gradient

$$b_t \;\coloneqq\; \mathbb{E}\Big[\nabla \hat{f}(\lambda_t)\Big] - \nabla f(\lambda_t), \qquad \mathbb{E}\Big[\big\|\nabla \hat{f}(\lambda_t) - \mathbb{E}[\nabla \hat{f}(\lambda_t)]\big\|^2\Big] = \sigma_t^2.$$

By L-smoothness, for $y = \lambda_{t+1} = \lambda_t - \eta\, \nabla \hat{f}(\lambda_t)$,

$$\begin{aligned} f(\lambda_{t+1}) &\le f(\lambda_t) + \big\langle \nabla f(\lambda_t),\, \lambda_{t+1} - \lambda_t \big\rangle + \frac{L}{2}\, \|\lambda_{t+1} - \lambda_t\|^2 \\ &= f(\lambda_t) - \eta \,\big\langle \nabla f(\lambda_t),\, \nabla \hat{f}(\lambda_t) \big\rangle + \frac{L}{2}\, \eta^2 \,\big\|\nabla \hat{f}(\lambda_t)\big\|^2. \end{aligned}$$

Taking expectation and using the bias notation,

$$\begin{aligned} \mathbb{E}[f(\lambda_{t+1})] &\le f(\lambda_t) - \eta \,\big\langle \nabla f(\lambda_t),\, \nabla f(\lambda_t) + b_t \big\rangle + \frac{L}{2}\, \eta^2 \,\mathbb{E}\Big[\big\|\nabla \hat{f}(\lambda_t)\big\|^2\Big] \\ &= f(\lambda_t) - \eta \,\|\nabla f(\lambda_t)\|^2 - \eta \,\big\langle \nabla f(\lambda_t),\, b_t \big\rangle + \frac{L}{2}\, \eta^2 \,\mathbb{E}\Big[\big\|\nabla \hat{f}(\lambda_t)\big\|^2\Big]. \end{aligned}$$

Apply Cauchy–Schwarz and Young's inequality to the bias inner product:

$$-\eta \,\langle \nabla f(\lambda_t), b_t\rangle \;\le\; \eta \,\|\nabla f(\lambda_t)\|\,\|b_t\| \;\le\; \frac{\eta}{2}\,\|\nabla f(\lambda_t)\|^2 \;+\; \frac{\eta}{2}\,\|b_t\|^2$$

For the second moment, decompose around the (biased) mean:

Combining the bounds yields

$$\mathbb{E}[f(\lambda_{t+1})] \tag{40}$$

$$\le f(\lambda_t) + \Big(-\tfrac{\eta}{2} + L\eta^2 \Big)\, \|\nabla f(\lambda_t)\|^2 + \frac{\eta}{2}\, ||b_t||^2 + \frac{L}{2}\, \eta^2\, (2b_t^2 + \sigma^2). \tag{41}$$

$$\le f(\lambda_t) + \Big(-\tfrac{\eta}{2} + L\eta^2 \Big)\, \|\nabla f(\lambda_t)\|^2 + \frac{\eta}{2}\, \big(||b_t||^2 + \sigma^2\big) + \frac{L}{2}\, \eta^2\, (2b_t^2 + 2\sigma_t^2). \tag{42}$$

$$\le f(\lambda_t) + \Big(-\tfrac{\eta}{2} + L\eta^2 \Big)\, \|\nabla f(\lambda_t)\|^2 + \frac{\eta}{2}\big(\mathbb{E}||\nabla \hat{f}(\lambda_t) - \nabla f(\lambda_t)||^2\big) \tag{43}$$

$$+ \frac{L}{2}\, \eta^2\, 2(\mathbb{E}||\nabla \hat{f}(\lambda_t) - \nabla f(\lambda_t)||^2). \tag{44}$$

$$\le f(\lambda_t) + \Big(-\tfrac{\eta}{2} + L\eta^2 \Big)\, \|\nabla f(\lambda_t)\|^2 + \frac{\eta}{2}\, \big(\beta(n, B, H)\big) + \frac{L}{2}\, \eta^2\, (2\beta(n, B, H)) \tag{45}$$

We obtain equation (43) from equation 42 by using the identity that the square of the bias plus variance is equal to the mean square error. We obtain (45) from equation 43 from the definition of $\beta(n, B, H)$.

$$f(\lambda_{t+1}) \le f(\lambda_t) - \left(\eta - \frac{L.\eta^2}{2}\right) ||\nabla f(\lambda_t)||^2 + \frac{(L.\eta^2 + \eta)\beta(n, B, H)}{2} \tag{46}$$

Now applying the PL inequality (Assumption 1), $\|\nabla f(\lambda_t)\|^2 \ge 2\mu\,(f(\lambda_t) - f^*)$, we substitute in the above inequality to get

$$f(\lambda_{t+1}) - f^* \le \left(1 - 2\mu\left(\eta - \frac{L\eta^2}{2}\right)\right)(f(\lambda_t) - f^*) + \frac{(L.\eta^2 + \eta)\beta(n, B, H)}{2}. \tag{47}$$

Define the contraction factor

$$\rho \coloneqq 1 - 2\mu\left(\eta - \frac{L.\eta^2}{2}\right). \tag{48}$$

we get the recursion:

$$\delta_{t+1} \leq \rho \cdot \delta_t + \frac{(L.\eta^2 + \eta).\beta(n, B, H)}{2}. \tag{49}$$

When $\eta \leq \frac{1}{L}$, we have

$$\eta - \frac{L\eta^2}{2} \geq \frac{\eta}{2} \Rightarrow \rho \leq 1 - \mu\eta. \tag{50}$$

Unrolling the recursion we have

$$\delta_t \leq (1 - \mu\eta)^t \delta_0 + \frac{(L.\eta^2 + \eta)\beta(n, B, H)}{2} \sum_{j=0}^{t-1} (1 - \mu\eta)^j. \tag{51}$$

Using the geometric series bound:

$$\sum_{j=0}^{t-1} (1 - \mu\eta)^j \leq \frac{1}{\mu\eta}, \tag{52}$$

we conclude that

$$\delta_t \leq (1 - \mu\eta)^t \delta_0 + \frac{(L.\eta^2 + \eta)\beta(n, B, H)}{2\mu\cdot\eta}. \tag{53}$$

Hence, we have the convergence result

$$f(\lambda_t) - f^* \leq (1 - \mu\eta)^t \delta_0 + \frac{(L.\eta + \eta)\cdot\beta(n, B, H)}{2\mu\cdot\eta}. \tag{54}$$

□

Lemma 2 (Uniform bound for a sample-based KL gradient estimator)**.** *Let $\mathcal{A}$ be a action space and, for a fixed state s, let $\pi_\lambda(\cdot \mid s)$ and $\bar{\pi}(\cdot \mid s)$ be two policies on $\mathcal{A}$, with parameter $\lambda \in \Lambda$. Assume:*

*(i) (*Bounded score*) There exists $B < \infty$ such that $\|\nabla_\lambda \log \pi_\theta(a \mid s)\| \leq B$ for all $a \in \mathcal{A}$ and $\theta \in \Theta$.*

*(ii) (*Common support bounded away from 0*) There exist $\varepsilon, \bar{\varepsilon} \in (0, 1]$ such that $\pi_\lambda(a \mid s) \geq \varepsilon$ for all $a \in \mathcal{A}$ and $\lambda \in \Lambda$, and $\pi_{ref}(a \mid s) \geq \bar{\varepsilon}$ for all $a \in \mathcal{A}$.*

Define the per-sample contribution

$$g_\theta(s, a) \;:=\; \nabla_\lambda \log \pi_\theta(a \mid s) \left(1 + \log \pi_\theta(a \mid s) - \log \pi_{ref}(a \mid s)\right),$$

so that $\nabla_\lambda D_{\mathrm{KL}}(\pi_\theta \| \pi_{ref}) = \mathbb{E}_{a \sim \pi_\theta(\cdot|s)}[\, g_\theta(s, a)\,]$. Then, with $C_{\log} := \log(1/\varepsilon) + \log(1/\bar{\varepsilon})$,

$$\|g_\theta(s, a)\| \;\leq\; B\,(1 + C_{\log}) \qquad \text{for all } a \in \mathcal{A} \text{ and } \theta \in \Theta,$$

and consequently, for any $n \geq 1$ and i.i.d. draws $a_1, \ldots, a_n \sim \pi_\theta(\cdot \mid s)$, the Monte-Carlo estimator $\hat{g}_n := \frac{1}{n}\sum_{i=1}^{n} g_\theta(s_i, a_i)$ satisfies $\|\hat{g}_n\| \leq B\,(1 + C_{\log})$.

Proof. (i) and (ii) are satisfied from Assumption 3, for every $a \in \mathcal{A}$ and $\theta \in \Theta$, $\pi_\theta(a \mid s) \in [\varepsilon, 1]$ and $\pi_{ref}(a \mid s) \in [\bar{\varepsilon}, 1]$, hence $\log \pi_\theta(a \mid s) \in [\log \varepsilon, 0]$ and $\log \pi_{ref}(a \mid s) \in [\log \bar{\varepsilon}, 0]$. Therefore

$$\left|\log \pi_\theta(a \mid s) - \log \pi_{ref}(a \mid s)\right| \;\leq\; \log(1/\varepsilon) + \log(1/\bar{\varepsilon}) \;=\; C_{\log},$$

and thus $\left|1 + \log \pi_\theta(a \mid s) - \log \pi_{ref}(a \mid s)\right| \leq 1 + C_{\log}$. By (i),

$$\|g_\theta(s, a)\| = \left\|\nabla_\lambda \log \pi_\theta(a \mid s)\right\| \left|1 + \log \pi_\theta(a \mid s) - \log \pi_{ref}(a \mid s)\right| \leq B\,(1 + C_{\log}).$$

This bound is deterministic (independent of the sample index) and holds for all a, θ, so taking averages over samples preserves it: $\|\hat{g}_n\| \leq B\,(1 + C_{\log})$. □

B Proof of Theorem 1

Proof. Using same steps as for Lemma A we get

$$\Phi(\phi_{t+1}) \leq \Phi(\phi_t) - \left(\frac{\eta}{2} - \frac{L.\eta^2}{2}\right)||\nabla\Phi(\phi_t)||^2 + \frac{\eta + L\eta^2}{2}\mathbb{E}||\nabla_\phi\Phi(\phi_k) - \nabla_\phi\hat{\Phi}_\sigma(\phi_k)||^2 \tag{55}$$

Now rearranging terms, summing Equation (55) over T and dividing by T on both sides we get

$$\frac{1}{T}\sum_{t=1}^{T}||\nabla\Phi(\phi_t)||^2 \leq \frac{1}{T}\sum_{t=0}^{t=T}\underbrace{\mathbb{E}||\nabla_\phi\Phi(\phi_k) - \nabla_\phi\hat{\Phi}_\sigma(\phi_k)||^2}_{A_t} + \tilde{\mathcal{O}}\left(\frac{1}{T}\right). \tag{56}$$

We now bound A_t as follows

$$\begin{aligned}
\mathbb{E}||\nabla_\phi\Phi(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))|| &= \mathbb{E}||\nabla_\phi\Phi(\phi_t) - \nabla_\phi\Phi_\sigma(\phi_t) + \nabla_\phi\Phi_\sigma(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))||, && (57)\\
&\leq \mathbb{E}||\nabla_\phi\Phi(\phi_t) - \nabla_\phi\Phi_\sigma(\phi_t))|| \\
&+ \mathbb{E}||\nabla_\phi\Phi_\sigma(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))||, && (58)\\
&\leq \mathcal{O}(\sigma) + \mathbb{E}\underbrace{||\nabla_\phi\Phi_\sigma(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))||}_{A_t}, && (59)
\end{aligned}$$

The first term on the right-hand side denotes the gap between the gradient of the objective function and the gradient of the pseudo-objective Φ_σ. We get the upper bound on this term from Lemma 4.3 of [2]. The term $A_t^{'}$ denotes the error incurred in estimating the true gradient of the pseudo-objective.

$$\begin{aligned}
\underbrace{\mathbb{E}||\nabla_\phi\Phi_\sigma(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))||^2}_{A_t} &\leq \mathbb{E}\Bigg|\Bigg|\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t)) + \frac{\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t))}{\sigma} \\
&- \nabla_\phi G(\phi_t, \lambda_t^K, B) + \frac{\nabla_{\phi_t}\hat{J}(\phi_t, \lambda_t^K) - \nabla_\phi J(\phi_t, {\lambda'}_t^K(\phi)), B}{\sigma}\Bigg|\Bigg|^2, && (60)\\
&\leq \mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K, B)||^2 \\
&+ \frac{1}{\sigma}\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_t^K, B)||^2 \\
&+ \frac{1}{\sigma}\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}_t^K, B)||^2. && (61)
\end{aligned}$$

As stated in the main text, the error in estimation of the gradient of the pseudo objective is split into the error in estimating $\nabla_\phi G(\phi, \lambda_\sigma^*(\phi))$, $\nabla_\phi J(\phi, \lambda^*(\phi))$ and $\nabla_\phi J(\phi, \lambda_\sigma^*(\phi))$ whose respective sample based estimates are denoted by $\nabla_\phi\hat{G}(\phi, {\lambda'}_t^K)$, $\nabla_\phi\hat{J}(\lambda_t^K, \phi)$ and $\nabla_\phi\hat{J}(\phi, {\lambda'}_t^k)$ respectively. From Lemmas 3, 4, and 5 we have

$$\underbrace{\mathbb{E}||\nabla_\phi\Phi_\sigma(\phi_t) - \nabla_\phi\hat{\Phi}_\sigma(\phi_t))||^2}_{A_t} \leq \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{\sigma^2 B}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-K}}{\sigma^2}\right) + \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{62}$$

Plugging Equation (62) into Equation (61), then plugging the result into Equation (56) and squaring both sides we get.

$$\frac{1}{T}\sum_{i=1}^{T}||\nabla\Phi(\phi_t)||^2 \leq \tilde{\mathcal{O}}\left(\frac{1}{T}\right) + \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{\sigma^2 B}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-K}}{\sigma^2}\right) + \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx})$$

(63)

Here T is the number of iterations of the outer loop of Algorithm 1, K is the number of iterations of the inner loop of Algorithm 1. n is the number of samples required for the gradients of J with respect to λ. B is the number of samples used to evaluate the gradients of G with respect to λ and ϕ respectively and the gradients of J with respect to ϕ.

□

C Supplementary Lemmas For Theorem 1

Lemma 3. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-4 we have*

$$\mathbb{E}||\nabla G(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi G(\phi, {\lambda'}^K_t, B)||^2 \leq \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}).$$

Proof.

$$\begin{aligned}\mathbb{E}||\nabla_\phi G(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}^K_t, B)||^2 &\leq \mathbb{E}||\nabla_\phi G(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}^K_t) \\ &+ \nabla_\phi G(\phi, {\lambda'}^K_t) - \nabla_\phi G(\phi_t, {\lambda'}^K_t, B)||^2, \quad (64)\\ &\leq \underbrace{\mathbb{E}||\nabla_\phi G(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}^K_t)||^2}_{A'_K} \\ &+ \underbrace{\mathbb{E}||\nabla_\phi G(\phi_t, {\lambda'}^K_t) - \nabla_\phi G(\phi_t, {\lambda'}^K_t, B)||^2}_{B'_K} \quad (65)\end{aligned}$$

A'_K represents the error incurred in due to difference between $\lambda^*_\sigma(\phi_t)$ and our estimate ${\lambda'}^K_t$. B'_K represents the difference between the true gradient $\nabla_\phi G(\phi, {\lambda'}^K_t)$ and its sample-based estimate. We first bound A'_K as follows

$$\begin{aligned}\mathbb{E}||\nabla_\phi G(\phi_t, \lambda^*_\sigma(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}^K_t)||^2 &\leq L\mathbb{E}||\lambda^*_\sigma(\phi_t) - {\lambda'}^K_t)||^2 \quad (66)\\ &\leq L_G \cdot \lambda' \mathbb{E}||h_\sigma(\phi_t, \lambda^*_\sigma(\phi_t)) - h_\sigma(\phi_t, {\lambda'}^K_t))||. \quad (67)\end{aligned}$$

Here L_G is the smoothness constant of $G(\lambda, \phi)$. We get Equation (67) from Equation (66) by the quadratic growth property applied to $h_\sigma(\phi, \lambda))$ using Assumption 1. Now, consider the function $h_\sigma(\phi, \lambda)$. We know from Assumption 1 that it satisfies the PL condition, therefore using Lemma 1 we obtain

$$\mathbb{E}||h_\sigma(\phi_t, \lambda^*(\phi_t)) - h_\sigma(\phi_t, \lambda^K_t))|| \leq \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \mathcal{O}(\beta(n, B, H)), \quad (68)$$

Where $\beta(n, B, H)$ is such that $\mathbb{E}||\nabla_\lambda h_\sigma(\phi, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi, \lambda)||^2 \leq \beta(n, B, H)$. Here, the expectation is with respect to the state action pairs sampled to estimate $\nabla_\lambda J(\phi, \lambda)$.

Now we have $\nabla_\lambda \hat{h}_\sigma(\phi, \lambda)$ as

$$\nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda) = \frac{1}{n}\sum_{i=1}^{n} \nabla \log(\pi_\lambda(a_i|s_i))\hat{Q_{\phi_t}}(s_i, a_i) + \frac{\beta}{B}\sum_{j=1}^{n}\sum_{i=1}^{H} \gamma^{i-1}\nabla_\lambda h_{\pi_\lambda, \pi_{ref}}(s_{i,j}, a_{i,j}) + \frac{1}{B}\sum_{i=1}^{B} \nabla_\lambda G(\phi_t, \lambda) \quad (69)$$

Thus, in order to bound $\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)||^2$, we decompose $\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)||^2$ as follows

$$\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)||^2$$

$$
\begin{aligned}
\leq\quad & 2\underbrace{(\mathbb{E}||\nabla_\lambda J(\phi_t,\lambda)-\frac{1}{n}\sum_{i=1}^{n}\nabla\log(\pi_\lambda(a_i|s_i))\hat{Q}_{\phi_t}(s_i,a_i))||)^2,}_{A} \\
+\quad & 4\underbrace{(\mathbb{E}||\beta\sum_{i=1}^{\infty}\mathbb{E}_{(s_i,a_i\sim\pi_\lambda)}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_i^{'},a_i^{'})-\beta\frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{H}\gamma^{i-1}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j},a_{i,j})||^2)}_{B} \\
+\quad & \sigma 4\underbrace{(\mathbb{E}||\nabla_\lambda G(\phi_t,{\lambda'}_t^K)-\nabla_\lambda G(\phi_t,{\lambda'}_t^K,B)||)}_{C} \\
+\quad & \tilde{\mathcal{O}}\left(\exp^{-K}\right).
\end{aligned} \tag{70}
$$

Now consider the terms in A, if we define $H = \mathbb{E}(\nabla\log\pi_\lambda(a|s)\hat{Q}_{\phi_t}(s,a))$ and $d = \frac{1}{n}\sum_{i=1}^{n}\nabla\log(\pi_\lambda(a_i|s_i))\hat{Q}_{\phi_t}(s_i,a_i))$ then we decompose A as follows

$$
\begin{aligned}
&\mathbb{E}||\nabla J(\phi_t,\lambda)-d+H-H|| \\
&\leq \mathbb{E}||\nabla J(\phi_t,\lambda)-H||)+\mathbb{E}||d-H|| \qquad (71)\\
&\leq \mathbb{E}||\nabla J(\phi_t,\lambda)-H|| \\
&+\mathbb{E}\left|\left|\frac{1}{n}\sum_{i=1}^{n}\left(\nabla\log\pi_{\lambda_k}(a_i|s_i)\hat{Q}(s_i,a_i)-(H)\right)\right|\right| \qquad (72)\\
&\leq \mathbb{E}||\nabla J(\phi,\lambda)-H||+ \\
&\mathbb{E}\sqrt{d\cdot\sum_{p=1}^{d}\left(\left(\sum_{i=1}^{n}\frac{1}{n}\nabla\log\pi_\lambda(a_i|s_i)\hat{Q}_\phi(s_i,a_i)\right)_p-(H)_p\right)^2} \qquad (73)\\
&\mathbb{E}(||\nabla J(\phi,\lambda)-d||) \\
&\leq \mathbb{E}||\nabla J(\phi,\lambda)-H||+ \\
&\sqrt{d\cdot\sum_{p=1}^{d}\mathbb{E}\left(\left(\sum_{i=1}^{n}\frac{1}{n}\nabla\log\pi_{\lambda_k}(a_i|s_i)\hat{Q}_\phi(s_i,a_i)\right)_p-(H)_p\right)^2} \qquad (74)\\
&\leq \mathbb{E}||\nabla J(\lambda_k)-H||+\frac{1}{\sqrt{n}}dM_gV_{max} \qquad (75)\\
&\leq M_g\mathbb{E}_{(s,a)}|Q_\phi^{\pi_\lambda}(s,a)-\hat{Q}_\phi(s,a)|+\frac{1}{\sqrt{n}}dM_gV_{max} \qquad (76)
\end{aligned}
$$

From [15] we have that

$$
\mathbb{E}|Q^{\pi_{\lambda_k}}(s,a)-\hat{Q}_\phi(s,a)|\leq\tilde{O}\left(\frac{1}{\sqrt{n}}\right)+\tilde{\mathcal{O}}(\epsilon_{approx}) \tag{77}
$$

Thus, we obtain

$$
A\leq\tilde{O}\left(\frac{1}{n}\right)+\tilde{\mathcal{O}}(\epsilon_{approx}) \tag{78}
$$

We obtain Equation (73) from Equation (72) by noting that l1 norm is upper bounded by the l2 norm multiplied by the square root of the dimensions. Here $(\nabla\log\pi_\lambda(a_i|s_i)\hat{Q}_{\phi_t}(s_i,a_i))_p$ and $(H)_p$ in Equation (73) are the p^{th} co-ordinates of the gradients. We obtain Equation (74) from Equation (73) by applying Jensen's inequality on the final term on the right hand side. We obtain Equation (75) from Equation (74) by noting that the variance of the random variable $\nabla\log\pi_{\lambda_k}(a|s)\hat{Q}(s,a)$ is bounded from Assumption 3 and Assumption 2 which implies that Θ is a compact set. We combine this with the fact that the variance of the mean is the variance divided by the number of samples, which in this case is n. We obtain Equation (76) from Equation (75) by using the policy gradient identity which states that $\nabla J(\phi, \lambda) = \mathbb{E}\nabla log\pi_\lambda(a|s)Q_\phi^{\pi_\lambda}(s,a)$ where M is such that $||\nabla \text{log}\pi_{\lambda_k}(a|s)|| \leq M_g$ for all $\lambda \in \Lambda$. We know that $||\nabla \text{log}\pi_{\lambda_k}(a|s)||$ are upper bounded by Assumption 3

We now bound B as follows

$$\mathbb{E}||\beta \sum_{i=1}^{\infty} \gamma^{i-1}\mathbb{E}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_i,a_i) - \frac{\beta}{B}\sum_{i=1}^{H}\sum_{j=1}^{B}\gamma^{i-1}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j}^{'},a_{i,j}^{'})|| \tag{79}$$

$$\leq \beta\mathbb{E}||\sum_{i=1}^{H}\gamma^{i-1}\mathbb{E}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_i,a_i) - \frac{1}{B}\sum_{i=1}^{H}\sum_{j=1}^{B}\gamma^{i-1}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j}^{'},a_{i,j}^{'})|| + \beta\mathbb{E}||\sum_{i=H}^{\infty}\gamma^{i-1}\mathbb{E}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_i,a_i)||, \tag{80}$$

$$\leq \beta\sum_{i=1}^{H}\gamma^{i-1}\mathbb{E}||\nabla_\lambda\mathbb{E}h_{\pi_\lambda,\pi_{ref}}(s_i,a_i) - \frac{1}{B}\sum_{j=1}^{B}\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j}^{'},a_{i,j}^{'})|| + \beta\sum_{i=H}^{\infty}\gamma^{i-1}\mathbb{E}||\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j}^{'},a_{i,j}^{'})||, , \tag{81}$$

$$\leq \mathcal{O}\left(\frac{\gamma^H}{\sqrt{B}}\right) + \mathcal{O}(\gamma^H). \tag{82}$$

Note that $(s_{i,j}^{'},a_{i,j}^{'})$ are the sample estimates of (s_i,a_i). We obtain Equation (80) from Equation (79) by splitting the first term on the left hand side of Equation (80) at the point $i = H$. We get Equation (82) from Equation (81) by considering the fact that the first term on the right hand side Equation (81) is a variance term bounded by a factor of $\frac{1}{\sqrt{n}}$ since the overall variance of the term B is bounded by Lemma 2. The second term on the right hand side of Equation (81) is bounded since the term $\nabla_\lambda h_{\pi_\lambda,\pi_{ref}}(s_{i,j}^{'},a_{i,j}^{'})$ is bounded from Lemma 2. Thus, we obtain

$$B \leq \mathcal{O}\left(\frac{\gamma^{2H}}{n}\right) + \mathcal{O}(\gamma^H) \tag{83}$$

We now bound C as follows

$$\mathbb{E}||\nabla_\lambda G(\phi_t,\lambda'^K_t) - \nabla_\lambda G(\phi_t,\lambda'^K_t,B)||$$

$$= \mathbb{E}\sqrt{d\cdot\sum_{p=1}^{d}\left(\left(\nabla_\lambda G_i(\phi_t,\lambda'^K_t)\right)_p - \left(\sum_{i=1}^{B}\frac{1}{B}\mathbb{E}\nabla_\lambda\hat{G}_i(\phi,\lambda'^K_t)\right)_p\right)^2}, \tag{84}$$

$$\leq \sqrt{\frac{d}{B^2}\cdot\sum_{p=1}^{d}\mathbb{E}\left(\sum_{i=1}^{B}\left(\nabla_\lambda G_{\tau_i}(\phi_t,\lambda'^K_t)_p - \mathbb{E}_\tau\nabla_\lambda\hat{G}_(\tau_i)(\phi_t,\lambda'^K_t)_p\right)\right)^2}, \tag{85}$$

$$\leq \sqrt{\frac{d^2.B.\sigma_G}{B^2}}, \tag{86}$$

$$\leq \sqrt{d.\frac{\sigma_G}{B}}, \tag{87}$$

$$\leq \tilde{O}\left(\frac{1}{\sqrt{B}}\right). \tag{88}$$

Here, the right-hand side of Equation (84) comes from writing out the definition of the ℓ_1 norm where the subscript of p denotes the p^{th} co-ordinate of the gradient. Equation (86) is obtained from Equation (85) by using Jensen's Inequality, and Equation (88) is obtained from 86 using Assumption 4 which states that the variance of ∇G estimator is bounded.

This gives us

$$C \leq \mathcal{O}\left(\frac{1}{B}\right) \tag{89}$$

Combining Equation (82) and (77) we have that

$$\begin{aligned}\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t,\lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t,\lambda)||^2 &\leq \mathcal{O}\left(\frac{\gamma^{2H}}{B}\right) + \mathcal{O}\left(\frac{1}{B}\right) + \mathcal{O}(\gamma^H) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}) \\ &\leq \mathcal{O}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}(\epsilon_{approx})\end{aligned} \tag{90}$$

Which in turn gives us

$$\mathbb{E}||h_\sigma(\phi_t,\lambda_t^K)) - h_\sigma(\phi_t,\lambda^*(\phi_t))|| \leq \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \mathcal{O}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{91}$$

We can bound $B_K^{'}$ in the exact same manner as C where the gradient is with respect to λ instead of ϕ to get

$$B_K^{'} \leq \mathcal{O}\left(\frac{1}{B}\right) \tag{92}$$

Thus we obtain

$$\mathbb{E}||\nabla_\phi G(\phi_t,\lambda_t^K) - \nabla_\phi G(\phi,\lambda_t^K,B)||^2 \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{93}$$

Substituting Equation (91) into Equation (67). Then put the result from Equation (67) and Equation (93) in Equation (65) to get the required result.

□

Lemma 4. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-4 we have*

$$\begin{aligned}\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*(\phi_t)) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2 \quad &\leq \quad \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) \\ &+ \quad \tilde{\mathcal{O}}(\epsilon_{approx})\end{aligned} \tag{94}$$

Proof.

$$\begin{aligned} & \mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*(\phi_t)) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2 \\ \leq\ & \mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*(\phi_t)) - \nabla_\phi J(\phi_t,\lambda_t^K) + \nabla_\phi J(\phi_t,\lambda_t^K) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2, \quad (95) \\ \leq\ & \mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*(\phi_t)) - \nabla_\phi J(\phi_t,\lambda_t^K)||^2 + \mathbb{E}||\nabla_\phi J(\phi_t,\lambda_t^K)\nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2, \quad (96) \\ \leq\ & L\mathbb{E}||(\lambda^*(\phi_t)) - (\lambda_t^K)||^2 + \mathbb{E}||\nabla_\phi J(\phi_t,\lambda_t^K) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2, \quad (97) \\ \leq\ & \underbrace{\mu \cdot L\mathbb{E}||J(\phi_t,\lambda^*(\phi_t)) - J(\phi_t,\lambda_t^K)||^2}_{A_K^{''}} + \underbrace{\mathbb{E}||\nabla_\phi J(\phi_t,\lambda_t^K) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||^2}_{B_K^{''}}. \quad (98)\end{aligned}$$

We get Equation (97) from Equation (96) by the smoothness of $J(\phi,\lambda)$ using Assumption 1. We get Equation (98) from (97) by the quadratic growth inequality on $J(\phi,\lambda)$. The first term $A_K^{''}$ is upper bounded using the same way as is done for $A_K^{'}$ Lemma 3, with the only difference being the absence of the term C in Equation (70). Thus, we have

$$\mathbb{E}||J(\phi_t,\lambda^*(\phi_t)) - J(\phi_t,\lambda_t^K))|| \quad \leq \quad \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}) \tag{99}$$

We bound $B_K^{''}$ as follows

$$\mathbb{E}||\nabla_\phi J(\phi_t,\lambda_t^K) - \nabla_\phi J(\phi_t,\lambda_t^K(\phi),B)||$$

$$
\begin{aligned}
&= \mathbb{E}\left|\left|\sum_{i=1}^{\infty}\gamma^{i-1}\mathbb{E}[\nabla_\phi r_{\phi_t}(s_i,a_i)] - \frac{1}{B}\sum_{j=1}^{B}\sum_{i=1}^{H}\gamma^{i-1}\nabla_\phi r_{\phi_t}(s^{'}_{i,j},a^{'}_{i,j})\right|\right| \\
&\leq \sum_{i=1}^{H}\gamma^{i-1}\Big(\mathbb{E}||\mathbb{E}[\nabla_\phi r_{\phi_t}(s_i,a_i)] - \frac{1}{B}\sum_{j=1}^{B}\nabla_\phi r_{\phi_t}(s_{i,j},a_{i,j})||\Big) \\
&+ \left|\left|\sum_{i=H}^{\infty}\gamma^{i-1}\mathbb{E}[\nabla_\phi r_{\phi_t}(s_i,a_i)]\right|\right|, \qquad (100) \\
&\leq \tilde{\mathcal{O}}\left(\frac{\gamma^H}{\sqrt{B}}\right) + \tilde{\mathcal{O}}(\gamma^H). \qquad (101)
\end{aligned}
$$

Thus we have

$$
\mathbb{E}||\nabla_\phi J(\phi_t,\lambda_t^K) - \nabla_\phi J(\phi_t,\lambda_t^K,B)||^2 \leq \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}(\gamma^H). \qquad (102)
$$

We get Equation (101) from Equation (100) since the first term on the right hand side of Equation (100) is variance term with a sample size of B. The last term on the right hand side of Equation (100) is upper bounded by γ^H since the term $\nabla_\phi r_\phi(s_i,a_i)$ is upper bounded by Assumption 3.

Plugging the result of Equation (102) and Equation (99) into Equation (98) gives us the required result..

□

Lemma 5. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-4 we have*

$$
\begin{aligned}
\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t,\lambda^{'K}_t(\phi),B)||^2 &\leq \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{n}\right) \\
&+ \tilde{\mathcal{O}}(\epsilon_{approx}) \qquad (103)
\end{aligned}
$$

Proof.

$$
\begin{aligned}
&\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t,\lambda^{'K}_t(\phi),B)|| \\
\leq\ &\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t,\lambda^{'k}_t) + \nabla_\phi J(\phi_t,\lambda^{'k}_t) - \nabla_\phi J(\phi_t,\lambda^{'K}_t(\phi),B)||^2, \qquad (104) \\
\leq\ &\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^*_\sigma(\phi_t)) - \nabla_\phi J(\phi_t,\lambda^{'k}_t)||^2 + ||\nabla_\phi J(\phi_t,\lambda^{'k}_t) - \nabla_\phi J(\phi_t,\lambda^{'K}_t(\phi),B)||^2, \qquad (105) \\
\leq\ &L_J.\mathbb{E}||(\lambda^*_\sigma(\phi_t)) - (\lambda^{'K}_t)||^2 + \mathbb{E}||\nabla_\phi J(\phi_t,\lambda^{'k}_t) - \nabla_\phi J(\phi_t,\lambda^{'K}_t(\phi),B)||^2, \qquad (106) \\
\leq\ &\underbrace{L_J.\mu\mathbb{E}||h_\sigma(\phi_t,\lambda^*_\sigma(\phi_t)) - h_\sigma(\phi,\lambda^{'K}_t)||}_{A^{'''}_k} + \underbrace{\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^{'K}_t) - \nabla_\phi J(\phi_t,\lambda^{'K}_t,B)||^2}_{B^{'''}_k}, \qquad (107)
\end{aligned}
$$

We get Equation (107) from Equation (106) using Assumption 1. Note that $B^{'''}_k$ here is the same as $B^{''}_K$ in Lemma 4. Thus we have

$$
\mathbb{E}||\nabla_\phi J(\phi_t,\lambda^{'K}_t) - \nabla_\phi \hat{J}(\phi_t,\lambda^{'K}_t(\phi))||^2 \leq \tilde{\mathcal{O}}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}(\gamma^H) \qquad (108)
$$

Further, we have

$$
\mathbb{E}||h_\sigma(\phi_t,\lambda^*_\sigma(\phi_t)) - h_\sigma(\phi_t,\lambda^{'K}_t)|| \leq \tilde{\mathcal{O}}\left(\frac{1}{n}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \mathcal{O}\left(\frac{\gamma^{2H}}{B}\right) + \tilde{\mathcal{O}}(\epsilon_{approx}), \qquad (109)
$$

This is the same result as for A'_K in Lemma 3.

Plugging Equations (108) and (109) into Equation (107) given us the required result. □

Lemma 6. *For a given $\lambda \in \Lambda$ and $\phi \in \Theta$ we have*

$$\nabla_\phi J(\phi, \lambda) = \sum_{i=1}^{\infty} \gamma^{i-1} \mathbb{E} \nabla_\phi r_\phi(s_i, a_i) \tag{110}$$

Proof. We start by writing the gradient of $J(\phi, \lambda)$ with respect to ϕ as follows

$$\begin{aligned}
&\nabla_\phi J(\phi, \lambda) \\
=\;& \nabla_\phi \int_{s_1,a_1} Q^\lambda_\phi(s_1, a_1)\pi_\lambda(a_1|s_1)d(s_1) \quad (111)\\
=\;& \int_{s_1,a_1} \nabla_\phi r_\phi(s_1, a_1)\pi_\lambda(a_1|s_1)d(s_1) \\
+\;& \gamma \cdot \nabla_\phi \int_{s_1,a_1} \int_{s_2,a_2} Q^\lambda_\phi(s_2, a_2)d(s_2|a_1)\pi_\lambda(a_2|s_2)d(s_1)\pi_\lambda(a_1|s_1), \quad (112)\\
=\;& \int_{s_1,a_1} \nabla_\phi r_\phi(s_1, a_1)\pi_\lambda(a_1|s_1)d(s_1) \\
+\;& \gamma \cdot \int_{s_2,a_2} \int_{s_1,a_1} \nabla_\phi r_\phi(s_2, a_2)d(s_2|a_1)\pi_\lambda(a_2|s_2)d(s_1)\pi_\lambda(a_1|s_1) \\
+\;& \gamma^2 \cdot \nabla_\phi \int_{s_1,a_1} \int_{s_2,a_2} \int_{s_3,a_3} Q^\lambda_\phi(s_3, a_3)d(a_3|s_3)d(s_3|a_2)d(s_2|a_1)\pi_\lambda(a_2|s_2)d(s_1)\pi_\lambda(a_1|s_1), \quad (113)\\
=\;& \int_{s_1,a_1} \nabla_\phi r_\phi(s_1, a_1)d(s_1, a_1) \\
+\;& \gamma \cdot \int_{s_2,a_2} \nabla_\phi r_\phi(s_2, a_2)d(s_2, a_3) + \gamma^2 \cdot \nabla_\phi \int_{s_3,a_3} Q^\lambda_\phi(s_3, a_3)d(s_3, a_3). \quad (114)
\end{aligned}$$

We get Equation (112) from Equation (111) by noting that $Q^\lambda_\phi(s, a) = r_\phi + \int_{s',a'} Q^\lambda_\phi(s', a')d(s'|a)\pi_\lambda(a'|s')$. We repeat the same process on the second term on the right hand side of Equation (112) to obtain Equation (113). Continuing this sequence, we get

$$\nabla_\phi J^\lambda_\phi = \sum_{i=1}^{\infty} \gamma^{i-1} \mathbb{E} \nabla_\phi r_\phi(s_i, a_i) \tag{115}$$

Here, s_i, a_i belong to the distribution of the i^{th} state action pair induced by following the policy λ. □

D Proof of Theorem 2

Proof. As is done for the proof of Theorem 1 we obtain the following from the smoothness assumption on Φ.

$$\frac{1}{T}\sum_{i=1}^{T} ||\nabla \Phi(\phi_t)||^2 \leq \frac{1}{T}\sum_{k=0}^{t=T} \underbrace{\mathbb{E}||\nabla_\phi \Phi(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t)||^2}_{A_t} + \tilde{\mathcal{O}}\left(\frac{1}{T}\right). \tag{116}$$

We now bound A_t as follows

$$\begin{aligned}
\mathbb{E}||\nabla_\phi \Phi(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))||^2 &= \mathbb{E}||\nabla_\phi \Phi(\phi_t) - \nabla_\phi \Phi_\sigma(\phi_t) + \nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))||^2, \quad (117)\\
&\leq \mathbb{E}||\nabla_\phi \Phi(\phi_t) - \nabla_\phi \Phi_\sigma(\phi_t))||^2 \\
&+ \mathbb{E}||\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))||^2, \quad (118)
\end{aligned}$$

$$\leq \quad \mathcal{O}(\sigma) + \underbrace{\mathbb{E}||\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))||^2}_{A_t'}, \tag{119}$$

The first term on the right hand side denotes the gap between the gradient of the objective function and the gradient of the pseudo-objective Φ_σ. We get the upper bound on this term form [2]. The term $A_t^{'}$ denotes the error incurred in estimating the true gradient of the pseudo-objective.

$$\begin{aligned}
\underbrace{\mathbb{E}||\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t)||^2}_{A_t'} \quad \leq \quad & \mathbb{E}\left|\left|\nabla_\phi G(\phi, \lambda_\sigma^*(\phi)) + \frac{\nabla_\phi J(\lambda^*(\phi), \phi) - \nabla_\phi J(\phi, \lambda_\sigma^*(\phi))}{\sigma}\right.\right. \\
- \quad & \left.\left.\nabla_\phi G(\phi_t, {\lambda'}_t^K, B) + \frac{\nabla_\phi J(\phi_t, \lambda_t^K(\phi_t), B) - \nabla_\phi J(\phi_t, {\lambda'}_t^K(\phi), B)}{\sigma}\right|\right|^2, \qquad (120) \\
\leq \quad & \mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K, B)||^2 \\
+ \quad & \frac{1}{\sigma}\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_t^K, B)||^2 \\
+ \quad & \frac{1}{\sigma}\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi_t, {\lambda'}_t^k, B)||^2. \qquad (121)
\end{aligned}$$

As stated in the main text, the error in estimation of the gradient of the pseudo objective is split into the error in estimating $\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t))$, $\nabla_\phi J(\phi_t, \lambda^*(\phi_t))$ and $\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t))$ whose respective sample based estimates are denoted by $\nabla_\phi \hat{G}(\phi_t, {\lambda'}_t^K)$, $\nabla_\phi \hat{J}(\phi_t \lambda_t^K)$ and $\nabla_\phi \hat{J}(\phi_t, {\lambda'}_t^K)$ respectively. From Lemmas 7, 8, and 9 we have

$$\underbrace{\mathbb{E}||\nabla_\phi \Phi_\sigma(\phi_t) - \nabla_\phi \hat{\Phi}_\sigma(\phi_t))||^2}_{A_t'} \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 B}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-K}}{\sigma^2}\right) \tag{122}$$

Plugging Equation (122) into Equation (121), then plugging the result into Equation (116) we get

$$\frac{1}{T}\sum_{i=1}^{T} ||\nabla \Phi(\phi_t)||^2 \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{T}\right) + \tilde{\mathcal{O}}\left(\frac{\exp^{-K}}{\sigma^2}\right) + \tilde{\mathcal{O}}\left(\frac{1}{\sigma^2 B}\right) + \tilde{\mathcal{O}}(\sigma^2) \tag{123}$$

Here T is the number of iterations of the outer loop of Algorithm 1, K is the number of iterations of the inner loop of Algorithm 1. B is the number of samples required for the all the gradient evaluations. □

E Supplementary Lemmas For Theorem 2

Lemma 7. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-2 and Assumptions 5 we have*

$$\mathbb{E}||\nabla G(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi G(\phi_t, \lambda_t^K, B)||^2 \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) \tag{124}$$

Proof.

$$\begin{aligned}
\mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K, B)||^2 \quad \leq \quad & \mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K) \\
+ \quad & \nabla_\phi G(\phi_t, {\lambda'}_t^K) - \nabla_\phi G(\phi_t, {\lambda'}_t^K, B)||^2, \qquad (125) \\
\leq \quad & 2\,\underbrace{\mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K)||^2}_{A_K'} \\
+ \quad & 2\,\underbrace{\mathbb{E}||\nabla_\phi G(\phi_t, {\lambda'}_t^K) - \nabla_\phi G(\phi_t, {\lambda'}_t^K, B)||^2}_{B_K'} \qquad (126)
\end{aligned}$$

We first bound $A_K^{'}$.

$$\begin{aligned}\mathbb{E}||\nabla_\phi G(\phi_t, \lambda_\sigma^*(\phi)) - \nabla_\phi G(\phi_t, {\lambda'}_t^K)||^2 &\leq L\mathbb{E}||\lambda_\sigma^*(\phi_t) - {\lambda'}_t^K)||^2 &(127)\\ &\leq L_1 \cdot \mu \mathbb{E}||h_\sigma(\phi_t, \lambda_\sigma^*(\phi_t)) - h_\sigma(\phi_t, {\lambda'}_t^k)||. &(128)\end{aligned}$$

Here L_1 is the smoothness constant of $G(\lambda, \phi)$. We get Equation (128) from Equation (127) by Assumption 1. Now, consider the function $J(\phi, \lambda)$. We know from Lemma 1 that it satisfies the weak gradient condition, therefore, applying the same logic for $J(\phi, \lambda)$ that we did for $\Phi(\sigma)$. Using Assumption 1, and Lemma 1 we obtain

$$\mathbb{E}||h_\sigma(\phi_t, \lambda_\sigma^*(\phi_t)) - h_\sigma(\phi_t, {\lambda'}_t^K)|| \leq \beta(B) + \tilde{\mathcal{O}}\left(\exp^{-K}\right), \tag{129}$$

where $\beta(n, B, H)$ satisfies $\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla h_\sigma(\phi_t, \lambda))||^2 \leq \delta(B)$. Note we changed notation from $\beta(n, B, H)$ to $\beta(B)$ since B samples are used to evaluate the gradients. Now in this case, we have an unbiased estimate of $\nabla h_\sigma(\phi_t, \lambda^*(\phi_t))$. Therefore, from assumption 5 we have that.

Now, the term $\mathbb{E}||\nabla h_\sigma(\phi_t, \lambda) - \nabla \hat{h}_\sigma(\phi, \lambda)||^2$, it can be decomposed as follows

$$\begin{aligned}&\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)||^2\\ =\ &\mathbb{E}||\nabla_\lambda J(\phi_t, \lambda) + \sigma \nabla_\lambda G(\phi_t, \lambda) - \nabla_\lambda J(\phi_t, \lambda, B) - \sigma \nabla_\lambda G(\phi_t, \lambda, B)||^2, &(130)\\ \leq\ &\underbrace{\mathbb{E}||\nabla_\lambda J(\phi_t, \lambda) - \nabla_\lambda J(\phi_t, \lambda, B)||^2}_{A'''} + \sigma \underbrace{\mathbb{E}||\nabla_\lambda G(\phi_t, \lambda) - \nabla_\lambda G(\phi_t, \lambda, B)||^2}_{B'''}. &(131)\end{aligned}$$

Note that both $A^{'''}$ and $B^{'''}$ can be bounded same as C in Lemma 3. thus we have

$$A^{'''} \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{132}$$

$$B^{'''} \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{133}$$

Thus we have $\beta(B) = \tilde{\mathcal{O}}\left(\frac{1}{B}\right)$. Which gives us

$$\mathbb{E}||h_\sigma(\phi_t, \lambda_\sigma^*) - h_\sigma(\phi_t, {\lambda'}_t^K)|| \leq \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{134}$$

Similarly $B_k^{'}$ here is bounded the same way as C in Lemma 3 to get

$$\mathbb{E}||\nabla_\phi G(\phi_t, \lambda_t^K) - \nabla_\phi G(\phi_t, \lambda_t^K, B)||^2 \leq \mathcal{O}\left(\frac{1}{B}\right) \tag{135}$$

Plugging Equation (134) and (135) into Equation (126) gives us the required result. □

Lemma 8. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-2 and Assumptions 5 we have*

$$\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi)) - \nabla_\phi J(\phi_t, \lambda_t^K(\phi), B)||^2 \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) \tag{136}$$

Proof.

$$\begin{aligned}&\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_t^K, B)||^2\\ \leq\ &\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_t^K) + \nabla_\phi J(\phi, \lambda_t^K) - J(\phi, \lambda_t^K(\phi), B)||^2, &(137)\\ \leq\ &\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda_t^K)||^2 + ||\nabla_\phi J(\phi_t, \lambda_t^K) - \nabla_\phi J(\phi, \lambda_t^K, B)||^2, &(138)\end{aligned}$$

$$\leq \quad L^{'}\mathbb{E}||(\lambda^*(\phi_t)) - (\lambda_t^K)||^2 + ||\nabla_\phi J(\phi_t, \lambda_t^K) - \nabla_\phi J(\phi_t, \lambda_t^K(\phi), B)||^2, \tag{139}$$

$$\leq \quad \underbrace{L^{'}\cdot\mu\mathbb{E}||J(\phi_t, \lambda^*(\phi_t)) - J(\phi_t, \lambda_t^K)||}_{A^{''}} + \underbrace{\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_t^K) - \nabla_\phi J(\phi_t, \lambda_t^K, B)||^2}_{B^{''}}. \tag{140}$$

We get Equation (139) form Equation (138) by using Assumption 1. The first term $A^{''}$ is upper the same way starting from Equation (129) as in Lemma 7 to give

$$\mathbb{E}||J(\phi_t, \lambda^*(\phi_t)) - J(\phi_t, \lambda_t^K)|| \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) \tag{141}$$

$B^{''}$ is bounded in the same manner as $B_k^{'}$ in Lemma 3 to give

$$\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_t^K) - \nabla_\phi J(\phi, \lambda_t^K(\phi), B)|| \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{142}$$

Plugging Equation (141) and (142) into Equation (140) given us the required result.

□

Lemma 9. *For a fixed $\phi_t \in \Theta$ and iteration t of Algorithm 1 under Assumptions 1-2 and Assumptions 5 we have*

$$\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda^{'K}_t, B)||^2 \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{B}\right) + \tilde{\mathcal{O}}\left(\exp^{-K}\right) \tag{143}$$

Proof.

$$\mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi, \lambda^{'K}_t, B)||^2$$

$$\leq \quad \mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda^{'K}_t) + \nabla_\phi J(\phi_t, \lambda^{'K}_t) - \nabla_\phi J(\phi_t, \lambda^{'K}_t(\phi), B)||^2, \tag{144}$$

$$\leq \quad \mathbb{E}||\nabla_\phi J(\phi_t, \lambda_\sigma^*(\phi_t)) - \nabla_\phi J(\phi_t, \lambda^{'K}_t)||^2 + ||\nabla_\phi J(\phi_t, \lambda^{'K}_t) - \nabla_\phi J(\phi_t, \lambda^{'K}_t(\phi), B)||^2, \tag{145}$$

$$\leq \quad L_J\mathbb{E}||(\lambda_\sigma^*(\phi_t)) - (\lambda^{'K}_t)||^2 + \mathbb{E}||\nabla_\phi J(\phi_t, \lambda^{'K}_t) - \nabla_\phi J(\phi_t, \lambda^{'K}_t(\phi), B)||^2, \tag{146}$$

$$\leq \quad \underbrace{L_J.L_\sigma\mathbb{E}||h_\sigma(\phi, \lambda_\sigma^*(\phi_t)) - h_\sigma(\phi_t, \lambda^{'K}_t)||}_{A^{''}} + \underbrace{\mathbb{E}||\nabla_\phi J(\phi, \lambda^{'K}_t) - \nabla_\phi J(\phi, \lambda^{'K}_t, B)||^2}_{B^{''}}. \tag{147}$$

We get Equation (147) from Equation (146) using Assumption 1. Note that $B^{''}$ can be bounded same as $B_k^{'}$ in Lemma 3. Thus we have

$$\mathbb{E}||\nabla_\phi J(\phi_t, \lambda^{'k}_t) - \nabla_\phi J(\phi_t, \lambda^{'K}_t(\phi), B)||^2 \quad \leq \quad \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{148}$$

For $A^{''}$ note that now the gradient descent is happening on the objective given by $h_\sigma = J(\lambda, \phi) - \sigma G(\phi, \lambda)$. Applying the same logic as we did for $J(\phi, \lambda)$, from Assumption 1 and Lemma 1 we get

$$\mathbb{E}||h_\sigma(\phi_t, \lambda_\sigma^*(\phi_t)) - h_\sigma(\phi_t, \lambda^{'k}_t)|| \quad \leq \quad \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \delta(B) \tag{149}$$

where $\delta(B)$ is such that $\mathbb{E}||\nabla_\lambda h_\sigma(\phi_t, \lambda) - \nabla_\lambda \hat{h}_\sigma(\phi_t, \lambda)||^2 \leq \delta(B)$

Now, consider the term $\mathbb{E}||\nabla h_\sigma(\phi_t, \lambda) - \nabla \hat{h}_\sigma(\phi_t, \lambda)||^2$, it can be decomposed as follows

$$\mathbb{E}||\nabla h_\sigma(\phi_t, \lambda) - \nabla \hat{h}_\sigma(\phi_t, \lambda)||^2$$

$$= \quad \mathbb{E}||\nabla_\lambda J(\phi_t, \lambda) + \sigma\nabla_\lambda G(\phi_t, \lambda) - \nabla_\lambda J(\phi_t, \lambda, B) - \sigma\nabla_\lambda G(\phi_t, \lambda, B)||^2, \tag{150}$$

$$\leq \quad \underbrace{\mathbb{E}||\nabla_\lambda J(\phi_t, \lambda) - \nabla_\lambda J(\phi_t, \lambda, B)||^2}_{A^{'''}} + \sigma\underbrace{\mathbb{E}||\nabla_\lambda G(\phi_t, \lambda) - \nabla_\lambda G(\phi_t, \lambda, n)||^2}_{B^{'''}}. \tag{151}$$

Note that both $A^{'''}$ and $B^{'''}$ can be bounded same as $B_k^{'}$ in Lemma 3. thus we have

$$A^{'''} \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{152}$$

$$B^{'''} \leq \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{153}$$

Thus we have $\delta(B) = \tilde{\mathcal{O}}\left(\frac{1}{B}\right)$. Which gives us

$$\mathbb{E}||h_\sigma(,\phi_t, \lambda^*_\sigma(\phi_t) - h_\sigma(\phi_t, {\lambda'}^k_t)|| \quad \leq \quad \tilde{\mathcal{O}}\left(\exp^{-K}\right) + \tilde{\mathcal{O}}\left(\frac{1}{B}\right) \tag{154}$$

Plugging Equation (154) and (148) into Equation (147) gives us the required result. □

F Experiments

F.1 Setup

The upper objective function to evaluate the reward is defined as follows

$$G(\lambda, \phi) = -\mathbb{E}_{y,\tau_0,\tau_1\sim\rho_H(\lambda)}(y{\cdot}P_\phi(\tau_0 > \tau_1) + (1-y){\cdot}(1 - P_\phi(\tau_0 > \tau_1))) \tag{155}$$

Where $\rho_H(\lambda)$ is the distribution of a trajectory of length H by following the policy λ and y is the preference which is 1 if trajectory 1 is preferred and 0 if Trajectory 0 is preferred which is drawn from some unknown distribution ρ. Also, $P_\phi(\tau_0 > \tau_1)$ is defined as

$$P_\phi(\tau_0 > \tau_1) = \frac{\exp\sum_{h=0}^{H-1} r_\phi(s^0_h, a^0_h)}{\exp\sum_{h=0}^{H-1} r_\phi(s^0_h, a^0_h) + \exp\sum_{h=0}^{H-1} r_\phi(s^1_h, a^1_h)}, \tag{156}$$

The objective to be minimized is given in Equation (9) as follows:

$$\Phi_\sigma(\phi) = \min_\lambda \left[G(\phi,\lambda) + \frac{1}{\sigma}\big(J(\phi,\lambda^*(\phi)) - J(\phi,\lambda)\big)\right],$$

where $\lambda^*(\phi) = \arg\max_\lambda J(\phi,\lambda)$ (noting the sign convention for the lower-level maximization of the return J).

To make this more implementable in an RL context, we reformulate the lower-level optimality using value functions. Let $V(\phi,\lambda)$ denote the value function under policy π_λ (i.e., $J(\phi,\lambda) = \mathbb{E}_{s\sim\nu,\, a\sim\pi_\lambda}[V(\phi,\lambda)]$, where ν is the initial state distribution). The optimal lower-level policy should maximize the value function, and should therefore satisfy

$$V(\phi,\lambda^*(\phi)) = V^*(\phi) = \max_\lambda V(\phi,\lambda).$$

Substituting this into the penalty form yields:

$$G(\phi,\lambda) + \frac{1}{\sigma}\big(V(\phi,\lambda^*(\phi)) - V(\phi,\lambda)\big) = G(\phi,\lambda) + \frac{1}{\sigma}\big(V^*(\phi) - V(\phi,\lambda)\big).$$

Directly minimizing the objective in Equation 9 is difficult in practice. Thus, for implementation, we make a practical approximation by dropping the $V(\phi,\lambda)$ term (which is non-negative under the assumption of non-negative rewards, a common setup in discounted MDPs where $V(\phi,\lambda) \geq 0$). This provides an upper bound on the objective while simplifying computation:

$$G(\phi,\lambda) + \frac{1}{\sigma}V^*(\phi).$$

In the code, this manifests as the regularization term added to the upper-level loss G, effectively encouraging the outer optimization (over ϕ) to maximize the optimal value V^* scaled by $1/\sigma$. This aligns with the bi-level structure by implicitly penalizing deviations from lower-level optimality without explicit inner-loop solving for λ^* at every step. We demonstrate improved performance over the PEBBLE [24] baseline in the two benchmarks using this approximation. We leave the implementation of the full Algorithm 1 as well as obtaining a tighter upper bound on Equation (9) to future work.

F.2 Implementation Details

We evaluate the effectiveness of this method, which solves the simplified objective, on two distinct environments: the Walker locomotion task from the DeepMind Control Suite [37] and the Door Open manipulation task from Meta-world [27]. These environments are chosen as representative benchmarks for robotic locomotion and manipulation, respectively, and both present the challenge of learning from limited, preference-based feedback rather than direct access to ground-truth rewards.

To demonstrate the efficacy of this approach, we compare against PEBBLE [24] baseline, which also uses preference-based feedback for solving complex tasks. Both PEBBLE as well as the proposed method utilize unsupervised exploration as proposed in PEBBLE [24], with disagreement-based sampling for query selection, a standard approach in preference-based reinforcement learning [28]. For the PEBBLE baseline, we employ the publicly released code from B-Pref [23], maintaining identical hyper-parameters and network architectures, such as the number of layers, learning rate, and the frequency of supervised reward learning. Our method builds on the PEBBLE framework, leveraging its core components while introducing our core contributions. We provide each task with a fixed budget of human preference labels: 100 labels for the Walker task and 1,000 labels for the Door Open task. All experiments are conducted on a single machine with an NVIDIA RTX 1080 Ti GPU, and we report results averaged over multiple independent runs with different random seeds.

F.3 Results

The training curves in Figure 1 illustrate the performance improvement of this approach against PEBBLE on both the Walker and Door Open tasks. In the Walker environment, the agent is rewarded for moving forward, and in our setting, the agent receives only preference-based feedback. The proposed method demonstrates improvements over the PEBBLE baseline, achieving higher average velocities and more stable learning trajectories with few preference labels. On the Door Open manipulation task, this approach similarly outperforms the baseline, successfully opening the door more consistently and efficiently.

These results highlight the effectiveness of this method in improving feedback efficiency and task performance, even in settings with limited preference-feedback. It is to be noted that this approach improves over the PEBBLE baseline without the need for second-order terms, unlike [1]. Other bi-level works such as [31] do not demonstrate improvement over state-of-the-art bi-level algorithms. Overall, these experiments validate the advantages of this proposed approach in both locomotion and manipulation scenarios, underscoring its potential for real-world robotic applications. The code can be found at https://github.com/MuditGaur/Neurips_2025_Bilevel_RL.

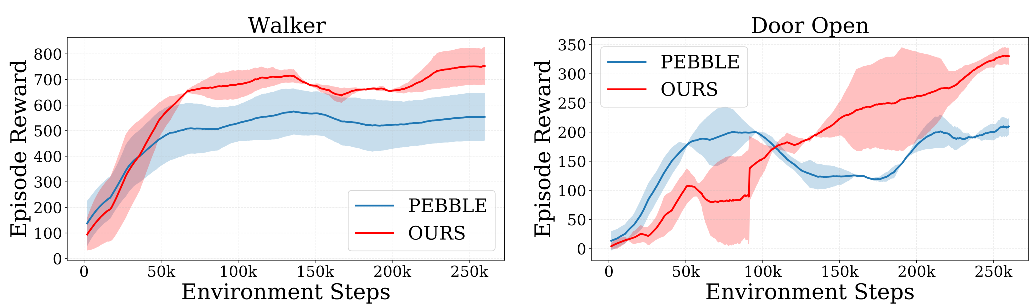

Figure 1: Training curves on Walker locomotion task (left) from the DeepMind Control Suite [37] and the Door Open manipulation task (right) from Meta-world [27]. The solid line and shaded regions respectively, denote mean and standard deviation of the success rate, across multiple seeds. Blue curve: PEBBLE, Red curve: OURS.