Bi-Level Reinforcement Learning Pathway for Sim-to-Real Optimality

Akhil S Anand, Shambhuraj Sawant , Paavo Parmas, Jasper Hoffmann, Dirk Peter Reinhardt, Sebastien Gros

Keywords: Sensitivity Analysis, Stochastic Policy Gradient, Reinforcement Learning, Bilevel Optimization, Objective Mismatch, Sim-to-Real RL

Summary

Training reinforcement learning (RL) policies in simulation before deploying them in real-world environments is a common strategy when real-world interaction is costly. However, the policies trained in simulation often perform poorly in the real world due to a mismatch between the simulation model and the real-world environment. This poor performance is caused by an objective mismatch: simulation models are typically designed for predictive accuracy, while policies are trained to maximize task performance. Motivated by the need to overcome this objective mismatch, we analyze the sensitivity of RL policies trained in simulation using stochastic policy gradient (SPG) methods in an actor-critic setting with respect to simulation parameters. Based on this sensitivity analysis, we formulate a bi-level RL approach that adapts simulation parameters based on the performance of the policy in the real world, thereby addressing the objective mismatch in a principled way. We provide a convergence analysis of the proposed bi-level RL approach and illustrate the concept with a proof-of-concept bi-level Proximal Policy Optimization (PPO) algorithm.

Contribution(s)

1. We derive the sensitivity of locally converged policies trained in simulation with SPG methods in an actor–critic setting with respect to simulation dynamics and reward parameters by implicitly differentiating the SPG fixed-point condition.
   Context: Prior work on differentiating RL policies with respect to the simulation model assumes specific policy structures (e.g., softmax over Q-functions). In contrast, our analysis directly differentiates the fixed-point condition of general SPG methods and applies to parametric policies trained with actor–critic methods without imposing such structural assumptions.
2. Based on this sensitivity analysis, we formulate a bi-level RL approach that adapts simulation parameters based on the performance of the policy in the real world, thereby addressing the objective mismatch in a principled way. We illustrate this approach with a bi-level PPO algorithm.
   Context: Current approaches address the objective mismatch problem by embedding proxy objectives into simulation model training to account for policy performance. Complementing these approaches, the proposed bi-level RL approach couples the simulation parameter learning objective and the policy performance objective by adapting simulation parameters using gradients of real-world policy performance.
3. We provide a thorough convergence analysis of the proposed bi-level RL approach.
   Context: The analysis shows that simulation and policy parameters can be updated simultaneously, provided the policy evolves on a faster timescale than the simulator parameters, thereby eliminating the need for full policy convergence at each simulator update step.

Bi-Level Reinforcement Learning Pathway for Sim-to-Real Optimality

Akhil S Anand[1], **Shambhuraj Sawant** [1], **Paavo Parmas**[2], **Jasper Hoffmann**[3], **Dirk Peter Reinhardt**[1], **Sebastien Gros**[1]
akhil.s.anand@ntnu.no

[1]**Norwegian University of Science and Technology (NTNU), Norway**
[2]**University of Tokyo, Japan**
[3]**University of Freiburg, Germany**

Abstract

Training Reinforcement Learning (RL) policies using simulation models before deployment in real-world environments is a common strategy when real-world interaction is expensive. This approach is used in sim-to-real RL and in dyna-style model-based RL. A key limitation of this approach is that the policies trained in simulation often perform poorly in the real world due to discrepancies between the simulation model and the real-world environment, referred to as the sim-to-real gap. This gap reflects the objective mismatch: simulation models are typically constructed for predictive accuracy, whereas policies are trained to maximize task performance. Since the policy learned in simulation is implicitly defined by the simulation parameters, understanding the sensitivity of the learned policy to these parameters enables gradient-based adaptation of the simulation model to improve real-world policy performance. Motivated by this, we derive the sensitivity of locally converged policies trained with Stochastic Policy Gradient (SPG) methods in an actor–critic setting, which is the most widely used approach in RL. Based on this sensitivity analysis, we formulate a bi-level RL approach that can address the objective mismatch problem by learning simulation parameters using gradients of real-world policy performance, thereby directly coupling simulation model adaptation with policy performance. We provide a thorough convergence analysis of the proposed bi-level RL approach and illustrate the concept through a proof-of-concept bi-level Proximal Policy Optimization (PPO) algorithm.

1 Introduction

Training Reinforcement Learning (RL) policies in simulation models of real-world dynamics and reward functions before deployment in real-world environments is a common strategy when real-world interaction is expensive (Zhao et al., 2020). This approach appears in multiple flavors, including Model-based Reinforcement Learning (MBRL), where simulation models are learned online (Moerland et al., 2023), and sim-to-real RL, where high-fidelity simulators are constructed offline with limited online adaptation (Da et al., 2025). This is supported by advances in the capabilities and fidelity of simulators and world models (Makoviychuk et al., 2021; Collins et al., 2021). Despite these advances, the inevitable discrepancies between the simulation model and the real world lead to a significant performance gap upon deploying the policy in the real world, referred to as the sim-to-real gap (Da et al., 2025).

Recognizing that perfectly modeling the stochastic real world is infeasible even under good data coverage, a fundamental cause of the sim-to-real gap is that simulation models are

typically constructed to predict real-world dynamics and rewards accurately, whereas policy training aims to maximize task performance. These objectives are not necessarily aligned, as the conditions that maximize the prediction accuracy of a simulation model differ from those that maximize the real-world performance of policies trained in that model, except in special cases (Anand et al., 2025; 2024b). This is well recognized as the *objective mismatch* problem in MBRL (Wei et al., 2024). This issue can persist even under good data coverage and low epistemic uncertainty, because simulation models trained for prediction accuracy are not informed by which aspects of the true dynamics matter for decision-making. It is particularly important in stochastic settings, especially for problems with economic or non-smooth objectives, where decision performance can depend on features of the real-world distribution that prediction-based model training need not preserve (Anand et al., 2025; 2024b). Despite theoretical and empirical evidence supporting this observation, there remains no consensus on the appropriate objective for training simulation models (Wei et al., 2024).

In theory, for any stochastic real-world environment, it is possible to adjust the simulation model such that the optimal policy in simulation can achieve optimal performance in the real world (Anand et al., 2024b;a; 2025). We refer to this property as *sim-to-real optimality*. Motivated by this observation, we analyze how standard RL policies trained in simulation (in-sim policies) can achieve sim-to-real optimality. Specifically, we consider Stochastic Policy Gradient (SPG) methods in an actor–critic setting, which are the most widely used methods in RL. Since the in-sim policy is implicitly defined by the simulation parameters, understanding the sensitivity of this policy with respect to these parameters enables gradient-based adaptation of the simulation model to improve real-world policy performance. This perspective is related to prior work (Anand et al., 2025; Nikishin et al., 2022; Chen et al., 2022) on differentiating the Markov Decision Process (MDP) underlying in-sim policy training. However, existing approaches are restrictive as they rely on differentiating Bellman optimality equations and therefore assume policies induced by Dynamic Programming (DP), typically represented as softmax distributions over value functions (Nikishin et al., 2022; Chen et al., 2022).

To this end, we analyze the sensitivity of locally converged policies trained with SPG methods in an actor–critic setting using the Implicit Function Theorem (IFT), making the analysis applicable to generic RL policies without requiring assumptions on their structure. Based on this sensitivity analysis, we frame the problem of finding simulation parameters that enable sim-to-real optimality as a policy-gradient RL problem with real-world feedback. This forms a bi-level RL approach, where the inner level trains an RL policy in simulation, and the outer level adapts simulation parameters to maximize the real-world performance of the in-sim policy, thereby aligning the objectives of simulation adaptation and policy performance. We provide a convergence analysis of the proposed bi-level RL approach using a three-timescale separation and illustrate the concept with a bi-level Proximal Policy Optimization (PPO) algorithm (Schulman et al., 2017), validating it through simple simulated examples.

Although motivated by addressing objective mismatch and sim-to-real gap, *the central contribution of this work is the sensitivity analysis of SPG-based policies*, which may be useful more broadly wherever the sensitivity properties of SPG methods are of interest.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 introduces the preliminaries. Section 4 establishes the notion of sim-to-real optimality. Section 5 presents the central results. Section 6 presents illustrative examples. Sections 7 and 8 present the discussion and conclusions.

2 Related Work

Closely related works study the sensitivity of optimal policies with respect to the underlying MDP parameters (Nikishin et al., 2022; Chen et al., 2022). These approaches differentiate Bellman optimality equations and therefore assume policies derived from value functions via

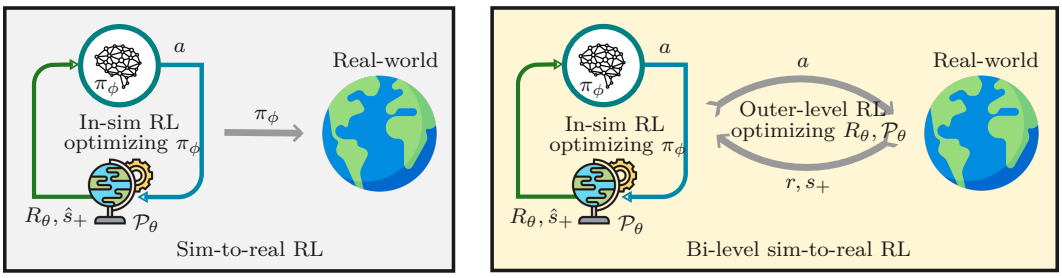

Figure 1: (Left) standard sim-to-real RL, (right) the proposed bi-level RL framework.

DP, typically represented as softmax (Boltzmann) distributions over Q-values (Nikishin et al., 2022; Chen et al., 2022). For example, Nikishin et al. (2022) applies implicit differentiation to the Bellman fixed-point equations of the optimal Q-function, while Chen et al. (2022) derives sensitivities of entropy-regularized optimal policies using the softmax structure induced by KL regularization. Consequently, these formulations assume policies are expressed as softmax transformations of value functions and do not directly cover policies with arbitrary parameterizations, such as Neural Network (NN) policies optimized via direct policy search. In contrast, our work derives sensitivities of policies obtained through SPG learning by implicitly differentiating the stationary condition of the policy gradient objective. This allows the analysis to apply directly to general parametric NN policies.

Objective mismatch is a well-recognized problem in MBRL (Lambert et al., 2020; Eysenbach et al., 2022; Janner et al., 2019), underscored by theoretical and empirical evidence (Hansen et al., 2022; Farahmand et al., 2017; Schrittwieser et al., 2020; Wan et al., 2024; Voelcker et al., 2023). Decision-aware approaches address this by incorporating value- or policy-level objectives into model learning (Wei et al., 2024). *Value-based* methods aim to learn simulation models that capture value functions instead of transition dynamics, following the *Proper Value Equivalence Principle* (Grimm et al., 2020; 2021). *Policy-based* methods instead use policy-gradient updates to adapt the simulation model, for example, by matching real-world and simulated policy gradients or minimizing gradient discrepancies (Jia et al., 2024; D'Oro et al., 2020; Abachi et al., 2021). Similarly, most sim-to-real RL methods are robustness-oriented rather than performance-oriented, optimizing proxy objectives such as simulator accuracy or variability instead of real-world performance directly (Da et al., 2025; Tobin et al., 2017; Duan et al., 2024); see Appendix E. We provide a complementary perspective by explicitly analyzing the sensitivity of the policy to the simulation model. Rather than heuristically modifying the model-learning objective, the sensitivity analysis is used to directly optimize model parameters to improve policy performance. Our approach falls within *Differentiable Planning*, where policy optimization is embedded in a differentiable program to jointly optimize the policy and simulator toward a common objective.

3 Preliminaries

Reinforcement Learning and Optimal Decision-Making

Formally, an RL problem is defined as a MDP, $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}(s'|s,a)$ is the real-world transition model defining the probability of transitioning to state s' given s, a (Sutton and Barto, 2018). $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the scalar reward function provided by the environment, and $\gamma \in [0,1]$ is the discount factor. A policy $\pi(a|s)$ defines the agent's behavior, mapping states to a distribution over actions. The objective of RL is to find the optimal policy $\pi^\star$ that maximizes the expected return $J(\pi) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$. The state-value function and action-value function are defined as $V^\pi(s) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s\right]$ and $Q^\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s, a_0 = a\right]$, respectively. An advantage function is

defined as $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$. The optimal value- $V^\star(s)$ and Q-functions $Q^\star(s,a)$ satisfy the Bellman optimality conditions: $V^\star(s) = \max_a Q^\star(s,a)$, $Q^\star(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim \mathcal{P}}\left[V^\star(s')\right]$. The optimal policy satisfies $\pi^\star(s) = \arg\max_a Q^\star(s,a)$.

In-sim Reinforcement Learning

We consider a simulation model as an abstract function $\mathcal{P}_\theta$, approximating the real-world distribution $\mathcal{P}$, used to generate the next state given a current state-action pair $s' \sim \mathcal{P}_\theta(\cdot \mid s,a)$ and to generate simulation-based trajectories mimicking the real system. The parameters θ may include the parameters of the functions defining the simulation dynamics. We consider a reward function $R_\theta(s,a)$ in the simulation as a model of the real-world reward r. The parameters of R_θ and $\mathcal{P}_\theta$ together constitute what we refer to as simulation parameters θ. The in-sim policy π_ϕ is an implicit function of simulation parameters θ, derived by solving:

$$\phi^\star = \arg\max_\phi \mathbb{E}_{\tau \sim \pi_\phi, \mathcal{P}_\theta}\left[\sum_{t=0}^{\infty} \gamma^t R_\theta(s_t,a_t)\right]. \tag{1}$$

The objects V_θ, Q_θ, and π_ϕ are implicitly defined by $\mathcal{P}_\theta$ and R_θ. The optimal policy $\pi_{\phi^\star}$ can be derived using the Bellman equations: $V^\star_\theta(s) = \max_a Q^\star_\theta(s,a)$, $Q^\star_\theta(s,a) = R_\theta(s,a) + \gamma\, \mathbb{E}_{s' \sim \mathcal{P}_\theta}[V^\star_\theta(s')]$, $\pi_{\phi^\star}(s) = \arg\max_a Q^\star_\theta(s,a)$.

Notation: Throughout the paper, quantities indexed by θ denote in-simulation quantities, whereas quantities without a θ subscript denote the corresponding real-world quantities.

4 Sim-to-Real Optimality

In this section, we establish the notion of *sim-to-real optimality*. Assuming the real-world and in-sim MDPs defined in Section 3 share the same state-space $\mathcal{S}$ and have a unique optimal policy, the in-sim MDP achieves sim-to-real optimality if the optimal policy obtained in simulation is also optimal for the real-world environment, i.e., $\pi_{\phi^\star} = \pi^\star$. This can be expressed as the necessary and sufficient optimality condition:

$$\arg\max_a Q^\star_\theta(s,a) = \arg\max_a Q^\star(s,a), \quad \forall\, s\,. \tag{2}$$

Sim-to-real optimality is critical in applications where real-world performance is the priority. Conventionally, the performance objective is pursued indirectly by constructing accurate simulation models of real-world dynamics via supervised learning and then training policies on these models. However, for *stochastic* systems, predictive accuracy of the simulation model does not necessarily correlate with the real-world performance of the resulting in-sim policy (Anand et al., 2025). This issue can persist even under good data coverage, because prediction-oriented model training is not informed by which aspects of the true stochastic dynamics matter for decision-making. It is particularly important in problems with economic or non-smooth objectives, where optimal decisions may depend on features of the transition distribution that prediction-based identification need not preserve (Anand et al., 2025).

However, it has been shown that even an in-sim policy under an imperfect simulation model $\mathcal{P}_\theta$ can achieve sim-to-real optimality under specific conditions on the model or on the in-sim reward function (Anand et al., 2025; Gros and Zanon, 2019). Following this observation, we propose framing the identification of the optimal simulation parameters $\theta^\star$ satisfying the necessary condition for sim-to-real optimality in (2) as a policy gradient RL problem:

$$\begin{aligned} \theta^\star \in \arg\max_\theta \quad & \mathbb{E}_{\tau \sim \pi_{\phi^\star}, \mathcal{P}}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t)\right] \\ \text{s.t.} \quad \phi^\star \in \arg\max_\phi \quad & \mathbb{E}_{\tau \sim \pi_\phi, \mathcal{P}_\theta}\left[\sum_{t=0}^{\infty} \gamma^t R_\theta(s_t,a_t)\right]. \end{aligned} \tag{3}$$

The RL formulations in (1) and (3) together form a bi-level RL problem. The inner level constitutes the in-sim RL problem to derive $\pi_{\phi^\star}$, which is implicitly defined by $\mathcal{P}_\theta$ and R_θ (1). Upon deployment in the real-world environment, the policy generates real transition data (s, a, s_+, r). The outer-level RL then adapts the simulator dynamics $\mathcal{P}_\theta$ and reward function R_θ to optimize real-world policy performance using the real-world transition data. The outer-level RL does not fit the simulation parameters to mimic real-world dynamics, but instead optimizes them to maximize the performance of the policy.

Additionally, the solution $\theta^\star$ to (3) is in general not unique. This allows selecting, among the simulation parameters that yield the same optimal policy performance, those that are also consistent with the real-world dynamics according to a user-specified loss $\mathcal{L}(\cdot)$, i.e., $\theta^\star = \arg\min_\theta \mathcal{L}(\mathcal{P}_\theta, \mathcal{P}; R_\theta, r)$ s.t. (3) holds for all s, a. Here, predictive accuracy is not introduced as a competing objective, but only as a selection or regularization criterion within the set of optimal models defined by $\theta^\star$. The existence of such optimal models within the support of prediction-oriented models is justified in Proposition 1 of (Anand et al., 2025). Common choices of $\mathcal{L}(\cdot)$ include Maximum Likelihood Estimation (MLE), Bayesian estimation, or expected-value losses such as MSE. In practice, this can be implemented by regularizing the policy-gradient step induced by (3) with $\nabla_\theta \mathcal{L}$.

5 Bi-level RL

In this section, we derive the mathematical tools to solve the bi-level RL problem (3). We begin by stating the regularity conditions required in the following assumption.

Assumption 1. *The policy $\pi_\phi(a|s)$, simulation model $\mathcal{P}_\theta(s'|s,a)$, and reward function $R_\theta(s,a)$ are continuously differentiable with respect to their respective parameters ϕ and θ.*

For generality across continuous and discrete state-action domains and stochastic simulation models, we consider that the in-sim and real-world RL agents are trained using SPG, satisfying:

$$\hat{\varphi}(\phi, \theta) = \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}[\nabla_\phi \log \pi_\phi(a|s) Q_\theta^{\pi_\phi}(s,a)]\,, \tag{4}$$

where $Q_\theta^{\pi_\phi}$ is the critic of policy π_ϕ *as seen* by the simulations, and $\hat{\rho}_\theta^{\pi_\phi}$ is the simulated discounted Markov chain density associated with policy π_ϕ under the model $\mathcal{P}_\theta$.

Assumption 2. *We assume that the in-sim SPG iterates have locally converged to a stationary policy parameter ϕ satisfying $\hat{\varphi}(\phi,\theta) = 0$. We further assume that $\hat{\varphi}(\phi,\theta)$ is continuously differentiable in (ϕ,θ), that the stationary solution ϕ is locally isolated, and that the Jacobian $\nabla_\phi \hat{\varphi}(\phi,\theta)$ is nonsingular.*

Under Assumption 2, SPG yields a locally optimal policy π_ϕ for the simulation model, and the bi-level RL problem can be written as

$$\theta^\star = \arg\max_\theta \mathbb{E}_{\tau\sim\pi_\phi,\mathcal{P}}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \quad \text{s.t.} \quad \hat{\varphi}(\phi,\theta) = 0\,. \tag{5}$$

The local regularity assumption is reasonable in practice, since SPG methods typically converge to locally stable stationary policies with non-degenerate curvature (Agarwal et al., 2021). The constraint $\hat{\varphi}(\phi,\theta) = 0$ therefore locally defines ϕ as a unique differentiable function of θ. Consequently, the sensitivity of the in-sim policy parameters ϕ with respect to simulation parameters θ can be obtained via the Implicit Function Theorem (IFT) (Krantz and Parks, 2002),

$$\nabla_\theta \phi \;=\; -\big(\nabla_\phi \hat{\varphi}(\phi,\theta)\big)^{-1} \nabla_\theta \hat{\varphi}(\phi,\theta)\,. \tag{6}$$

Note that there exist iterative methods and their approximations to invert the Jacobian $\nabla_\phi \hat{\varphi}(\phi,\theta)$ (Liesen and Strakos, 2013; Yang et al., 2024). Using $\nabla_\theta \phi$, we can then formulate a solution to (5) as the outer-level SPG using the real-world transition data:

$$\varphi(\phi,\theta) := \mathbb{E}_{s\sim\rho^{\pi_\phi}}\Big[\nabla_\theta \log \pi_\phi(a \mid s)\, Q^{\pi_\phi}(s,a)\Big], \tag{7a}$$

$$\nabla_\theta \log \pi_\phi(a \mid s) = \nabla_\theta \phi \, \nabla_\phi \log \pi_\phi(a \mid s)\,, \tag{7b}$$

where Q^{π_ϕ} is the critic of the in-sim policy π_ϕ *as seen* by the real world, ρ^{π_ϕ} is the discounted Markov chain density resulting from using in-sim policy π_ϕ in the real world, and (7b) is the result of the chain-rule with the IFT condition. To update the simulation parameters via (7a), we must compute the sensitivity of the in-sim RL solution with respect to simulation parameters, given by the gradient $\nabla_\theta \phi$. Computing $\nabla_\theta \phi$ is the only additional component of (7a) compared to the classic SPG.

5.1 Sensitivity Analysis of Stochastic Policy Gradient

We now introduce two lemmas that will help us compute $\nabla_\theta \phi$ and derive the full expression for the real-world SPG in (7a). Note that the ∂ in Lemma 1 and Lemma 2 denotes a partial derivative with respect to either ϕ or θ.

Lemma 1 (Sensitivity of a Markov Chain)**.** *Let $\pi_\phi(a \mid s)$ be a differentiable stochastic policy parameterized by ϕ, let $\mathcal{P}_\theta(s' \mid s, a)$ be a differentiable transition model parameterized by θ, and let ρ^0 denote the initial state distribution. For convenience, define the closed-loop one-step transition density $\mathcal{P}_\theta^{\pi_\phi}(s_{i+1}, a_i \mid s_i) \triangleq \pi_\phi(a_i \mid s_i)\, \mathcal{P}_\theta(s_{i+1} \mid s_i, a_i)$. The policy and transition model induce a Markov chain with discounted state distribution $\hat{\rho}_{\theta,k}^{\pi_\phi}$ at time step k. Then, for any bounded measurable function $\eta(s_k, a_k)$, the sensitivity of its expected value with respect to a parameter (denoted by ∂) can be written as*

$$\partial\, \mathbb{E}_{s \sim \hat{\rho}_{\theta,k}^{\pi_\phi}} \big[\eta(s_k, a_k)\big] = \mathbb{E}_{\substack{s_0 \sim \rho^0,\, a_i \sim \pi_\phi, s_{i+1} \sim \mathcal{P}_\theta^{\pi_\phi}}} \Big[\eta(s_k, a_k) \Big(\partial \log \pi_\phi(a_k \,|\, s_k) + \sum_{i=0}^{k-1} \partial \log \mathcal{P}_\theta^{\pi_\phi}(s_{i+1} \,|\, s_i, a_i) \Big) \Big]\,. \tag{8}$$

Here, we assume that the policy and transition densities are strictly positive on the support of trajectories induced by the Markov chain, ensuring that log-likelihood derivatives in (8) remain finite. The proof of Lemma 1 is provided in Appendix A.1.

Lemma 2 (Sensitivities of Q^π)**.** *Let $(s_t, a_t)_{t \geq 0}$ be samples from the in-simulation dynamics $\mathcal{P}_\theta(\cdot \mid s_t, a_t)$ and the policy $\pi_\phi(\cdot \mid s_t)$, with $(s_0, a_0) = (s, a)$. Then the sensitivity of the simulation-based critic with respect to the reward and model parameters θ under a fixed policy π_ϕ is given by*

$$\begin{aligned} \nabla_\theta Q_\theta^{\pi_\phi}(s, a) = \mathbb{E}_{\substack{s \sim \hat{\rho}_\theta^{\pi_\phi}, \\ a \sim \pi_\phi}} \Bigg[& \sum_{t=0}^{\infty} \gamma^t \, \nabla_\theta R_\theta(s_t, a_t) \\ & + \sum_{t=0}^{\infty} \gamma^{t+1} \, \nabla_\theta \log \mathcal{P}_\theta(s_{t+1} \mid s_t, a_t)\, Q_\theta^{\pi_\phi}(s_{t+1}, a_{t+1}) \;\Bigg|\; s_0 = s,\ a_0 = a \Bigg]\,, \end{aligned} \tag{9}$$

and with respect to the policy parameters ϕ under fixed reward R_θ and model $\mathcal{P}_\theta$ is given by

$$\nabla_\phi Q_\theta^{\pi_\phi}(s, a) = \mathbb{E}_{\substack{s \sim \hat{\rho}_\theta^{\pi_\phi} \\ a \sim \pi_\phi}} \Bigg[\sum_{t=0}^{\infty} \gamma^{t+1} \, \nabla_\phi \log \pi_\phi(a_{t+1} \mid s_{t+1})\, Q_\theta^{\pi_\phi}(s_{t+1}, a_{t+1}) \;\Bigg|\; s_0 = s,\ a_0 = a \Bigg]\,. \tag{10}$$

The proof of Lemma 2 is provided in Appendix A.2. Note that score-terms in the critic sensitivities (9) and (10) can induce high variance, so variance reduction methods such as a baseline or the reparameterization trick should be used. A reparameterization version of the critic sensitivity terms (9) and (10) is provided in Appendix A.2.

There exists an alternative method to estimate the critic sensitivities, which is provided in Lemma 4 in Appendix A.3. This avoids unrolling entire trajectories as required in Lemma 2 by directly differentiating the Bellman equations. However, this approach is computationally less efficient than the method in Lemma 2, as it requires iteratively estimating sensitivities for both the Q and V functions. For completeness, we provide a proof of equivalence between these two sensitivity estimation approaches in Appendix A.4.

Using these two lemmas, we now present the main result on the sensitivity analysis of SPG in the following theorem.

Theorem 1 (Sensitivity of in-sim SPG)**.** *Consider a simulation model $\mathcal{P}_\theta$, reward function R_θ, and the resulting in-sim policy π_ϕ obtained by solving the in-simulation RL problem using SPG (4) in an actor-critic setting. Assume that the policy parameters ϕ correspond to a locally stationary solution of the SPG update, i.e., $\hat{\varphi}(\phi, \theta) = 0$. Then the sensitivity of the in-sim policy parameters ϕ with respect to the simulation model and reward parameters θ is given by*

$$\nabla_\theta \phi = -\Big(\underbrace{\mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_{\phi\phi}\log\pi_\phi\, Q_\theta^{\pi_\phi} + \nabla_\phi \log\pi_\phi\, \nabla_\phi Q_\theta^{\pi_\phi}\right]}_{\text{Gradient of } \hat{\varphi} \text{ w.r.t. } \phi} + \underbrace{\nabla_\phi \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log\pi_\phi\, Q_\theta^{\pi_\phi}\right]}_{\text{Gradient of dist. } \hat{\rho}_\theta^{\pi_\phi} \text{ w.r.t. } \phi}\Big)^{-1}$$
$$\Big(\underbrace{\mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log\pi_\phi\, \nabla_\theta Q_\theta^{\pi_\phi}\right]}_{\text{Gradient of } \hat{\varphi} \text{ w.r.t. } \theta} + \underbrace{\nabla_\theta \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log\pi_\phi\, Q_\theta^{\pi_\phi}\right]}_{\text{Gradient of dist. } \hat{\rho}_\theta^{\pi_\phi} \text{ w.r.t. } \theta}\Big). \tag{11}$$

Here, $\nabla_\phi \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}[\cdot]$ is the effect of changing ϕ on the simulation-based distribution $\hat{\rho}_\theta^{\pi_\phi}$, and the influence of that change on the expected value *under a fixed cost and model*. $\nabla_\theta \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}[\cdot]$ is the effect of changing θ on the simulation-based distribution $\hat{\rho}_\theta^{\pi_\phi}$, and the influence of that change on the expected value *under a fixed policy*. The terms in the second line of (11) describe how perturbations to the simulation parameters modify both the value estimates and the distribution of simulated trajectories, which alters the policy gradient. The inverse Jacobian term in the first line of (11) maps these changes in the policy gradient to the resulting shift in the locally optimal policy parameters. A detailed and intuitive explanation of each of the terms in (11) is provided in the first part of Appendix A.

In practice, all components in Theorem 1 will be estimated from the in-sim transition data $(\hat{s}, a, \hat{s}_+, R)$ generated from model sampling. Let $(s_0^j, a_0^j, R_0^j, \ldots, s_N^j, a_N^j, R_N^j)$ be the sampled trajectory $j \in \{1, \ldots, n\}$ of length N that should be long enough for Lemma 2 to apply. The expected value approximation of the Markov Chain sensitivity of the in-sim policy gradient with respect to θ and ϕ is given by:

$$\nabla_\phi \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log\pi_\phi\, Q_\theta^{\pi_\phi}\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} \nabla_\phi \log\pi_\phi\left(a_k^j \middle| s_k^j\right)\, Q_\theta^{\pi_\phi}\left(s_k^j, a_k^j\right)$$
$$\otimes \left(W_{\phi,k}^j + \nabla_\phi \log\pi_\phi\left(a_k^j \middle| s_k^j\right)\right) \tag{12}$$

where: $\otimes$ represents the outer product and $W_{\phi,0}^j = 0$; $W_{\phi,k}^j = W_{\phi,k-1}^j + \nabla_\phi \log\pi_\phi\left(a_{k-1}^j \middle| s_{k-1}^j\right)$. Similarly,

$$\nabla_\theta \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log\pi_\phi\, Q_\theta^{\pi_\phi}\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} \nabla_\phi \log\pi_\phi\left(a_k^j \middle| s_k^j\right)\, Q_\theta^{\pi_\phi}\left(s_k^j, a_k^j\right) \otimes W_{\theta,k}^j \tag{13}$$

where $W_{\theta,0}^j = 0$ and $W_{\theta,k}^j = W_{\theta,k-1}^j + \nabla_\theta \log\mathcal{P}_\theta\left(s_k^j \middle| s_{k-1}^j, a_{k-1}^j\right)$.

A high-level algorithmic description of the bi-level RL approach is provided in Algorithm 1.

5.2 Convergence Analysis and Efficient Implementation of Bi-level RL

Convergence Analysis. A detailed convergence analysis for the bi-level RL approach is provided in Appendix B. The analysis establishes almost-sure convergence of the proposed bi-level SPG scheme under a three-timescale stochastic-approximation scheme: the critic and its sensitivities evolve on the fastest timescale, the in-sim policy on an intermediate timescale, and the outer-level update of simulation parameters on the slowest timescale. An important practical implication of this analysis for Algorithm 1 is that the in-sim policy does

Algorithm 1 Bi-Level SPG Framework

Given: initialized $\mathcal{P}_\theta(s_+ \mid s, a)$, $R_\theta(s, a)$, $\pi_\phi(a \mid s)$, $V^{\pi_\phi}(s)$, and $Q^{\pi_\phi}(s, a)$.
for each bi-level RL iteration **do**
  Train in-sim policy π_ϕ via SPG for N steps (Eq. (4)).
  Roll out π_ϕ over $\mathcal{P}_\theta, R_\theta$ to collect in-sim transitions (s, a, s_+, R).
  Compute $\nabla_\theta \phi$ from these in-sim transitions (Eq. (11)).
  Roll out π_ϕ on the real world $\mathcal{P}$ to collect transitions (s, a, s_+, r).
  Update θ according to Eq. (7a).
end for

not need to be fully optimized before each update to the simulation parameters. Instead, it is enough that the in-sim policy is updated on a faster timescale than the outer level, so that it remains sufficiently close to the local optimum corresponding to the current simulation parameters. This provides formal justification for taking outer-level steps with partially updated in-sim policies, provided that the critic and sensitivity errors remain controlled.

Efficient Implementation. The inverse of a Jacobian operator in the policy parameter space ($\nabla_\phi \hat{\varphi}$) in (11) can often be high-dimensional in deep RL settings. Although Theorem 1 yields the full sensitivity matrix $\nabla_\theta \phi \in \mathbb{R}^{d_\phi \times d_\theta}$, a practical implementation of the outer update in (7a) does *not* require forming this matrix explicitly. Since the outer objective (the real-world return) (5) is scalar, the required quantity is the vector-Jacobian product $(\nabla_\theta \phi)^\top \nabla_\phi \mathcal{L}$, where $\mathcal{L}$ denotes the outer objective in (5). Using an adjoint variable, this product can be computed by (i) solving a single linear system involving $(\nabla_\phi \hat{\varphi})^\top$ and (ii) evaluating one additional vector–Jacobian product through the residual $\hat{\varphi}(\phi, \theta)$. This approach avoids constructing any dense $d_\phi \times d_\theta$ objects, and fits naturally with reverse-mode automatic differentiation. The details of this approach and a comparison of the computational efficiency are provided in Appendix C.

5.3 Bi-level PPO

Using the sensitivity analysis in Theorem 1, we now formulate a bi-level version of the PPO algorithm (Schulman et al., 2017) that updates the simulation parameters toward achieving sim-to-real optimality. Consider the in-sim PPO objective given by:

$$\mathcal{L}_\theta(\phi) := \mathbb{E}_{\substack{s \sim \hat{\rho}_\theta^{\pi_\phi}, \\ a \sim \pi_\phi}} \left[\frac{\pi_\phi(a \mid s)}{\pi_{\bar{\phi}}(a \mid s)} A_\theta^{\pi_\phi}(s, a) + \beta \, \mathcal{H}(\pi_\phi(\cdot \mid s)) \right], \tag{14}$$

where $\bar{\phi}$ and ϕ denote the policy parameters before and after the current update step, and $A_\theta^{\pi_\phi}$ denotes the in-sim advantage, and $\mathcal{H}$ is the policy entropy with β as the entropy coefficient. In (14), the practical components in a typical PPO implementation, such as generalized advantage estimation, clipping the importance sampling ratio, and policy divergence, are omitted for clarity. The in-sim policy is updated according to: $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_\theta(\phi)$, which yields the approximate fixed point $\phi(\theta)$ once converged sufficiently. The corresponding outer-level PPO objective for updating θ is given by:

$$\mathcal{L}(\phi, \theta) := \mathbb{E}_{\substack{s \sim \rho_\theta^{\pi_\phi}, \\ a \sim \pi_\phi}} \left[\frac{\pi_\phi(a \mid s)}{\pi_{\bar{\phi}}(a \mid s)} A^{\pi_\phi}(s, a) + \beta \, \mathcal{H}(\pi_\phi(\cdot \mid s)) \right], \tag{15}$$

where A^{π_ϕ} denotes the outer-level advantage function. The outer-level parameter update is $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\phi, \theta)$, with the outer-level gradient $\nabla_\theta \mathcal{L}(\phi, \theta) = (\partial \mathcal{L}(\phi, \theta)/\partial \phi) \, \nabla_\theta \phi(\theta)$, where $\nabla_\theta \phi(\theta)$ is obtained from (11) by replacing $Q_\theta^{\pi_\phi}$ with $A_\theta^{\pi_\phi}$. A pseudo-code of the bi-level PPO algorithm is provided in Algorithm 2.

In practice, PPO performs stochastic updates that approach a neighborhood of a local stationary point of the surrogate objective (14). The derived sensitivity therefore describes how small perturbations in simulator parameters change the PPO policy in this neighborhood. Learning can further be accelerated by performing multiple simulator updates per outer

Algorithm 2 Bi-Level PPO Algorithm

Initialize simulation parameters θ, policy parameters ϕ, inner critic V_θ, outer critic V.
for $k = 1, \ldots, N_{\text{outer}}$ **do** ▷ Outer-level update of simulation parameters
  for $j = 1, \ldots, N_{\text{inner}}$ **do** ▷ In-sim PPO training
    Roll out π_ϕ in the simulation $\mathcal{P}_\theta, R_\theta$ to collect (s, a, s_+, R).
    Estimate returns and in-sim advantages A_θ; update V_θ.
    Update ϕ.
  end for
  Roll out π_ϕ in simulator $\mathcal{P}_\theta, R_\theta$ to collect (s, a, s_+, R).
  Compute $\nabla_\theta \phi(\theta)$ from in-sim (Eq. (11)).
  Roll out π_ϕ on the real system $\mathcal{P}$ to collect (s, a, s_+, r).
  Estimate returns and real-world advantages A; update V.
  Update $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}(\phi, \theta)$.
end for

iteration, similar to standard PPO. This would require additionally updating the policy parameters at the outer level based on real-world data, allowing the outer-level policy updates to serve as a proxy for the fully optimized in-sim policy, while also providing better policy initializations for the inner level. This strategy is effective when small simulator perturbations induce smooth, local changes in policy parameters.

6 Illustrative Examples

We present three illustrative realizations of the bi-level RL approach: (i) stochastic control in a discrete state–action space, (ii) stochastic continuous control with linear dynamics, and (iii) a quadcopter stabilization task. The first two examples validate the sensitivity analysis in (11) by evaluating the outer-level SPG updates, while the third demonstrates the bi-level PPO algorithm. In each case, we initialize five simulation models with parameters randomly sampled around the real-world environment parameters. These experiments are intended solely to illustrate the proposed framework rather than serve as a performance benchmark. Additional experimental details are provided in Appendix D.

Discrete MDP. We consider a stochastic discrete MDP with state and action sets $\mathcal{S} = \{s_0, s_1, s_2\}$ and $\mathcal{A} = \{0, 1\}$, where each action may yield different successor states. The state-transition model includes the parameters: $\theta^f_{s,a} = (\theta^f_{s,a,0}, \theta^f_{s,a,1}, \theta^f_{s,a,2}) \in \mathbb{R}^3$ via $\mathcal{P}_\theta(s' \mid s, a) = \text{softmax}(\theta^f_{s,a})_{s'}$. Rewards use $\theta^R \in \mathbb{R}^{3\times 2}$ via $R_\theta(s, a) = \theta^R_{s,a}$. In this example, we use DP to compute exact solutions of a discrete in-sim policy and its value function. This discrete policy is converted to a stochastic policy that uses action preferences (logits) $h_\phi(s, a) = \phi^\pi_{s,a}$, i.e., the learnable score for action a in state s, with $\theta^\pi \in \mathbb{R}^{3\times 2}$. The policy selects actions by $\pi_\theta(a \mid s) = \text{softmax}\big(h_\theta(s, \cdot)\big)_a$. Then we estimate the critic sensitivity (9) and (10) in a tabular form iteratively until convergence using on-policy in-sim data. We use the same in-sim on-policy data to estimate the Markov chain sensitivities (12) and (13).

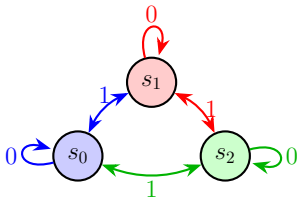

Figure 2: Discrete MDP.

Continuous MDP. We consider a linear system with additive Gaussian noise; $s_+ = \theta_s\, s + \theta_a\, a + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$ and a parameterized exponential reward function $R_\theta(s, a) = \exp\big(-\lambda\big(\theta_q\, s^2 + \theta_r\, a^2\big)\big)$. The real system parameters are: $\theta_s = 1$, $\theta_a = 1$, $\theta_q = 1$, $\theta_r = 1$. The simulation model and reward parameters are chosen randomly between 0 and 1. We solve the in-sim policy and value function exactly using Linear Quadratic Regulator (LQR), and then use NNs to approximate them. By approximating them with NNs, we can validate the robustness of the sensitivity estimates with regard to the noisy NN predictions. Then we estimate the critic sensitivity (9) and (10)

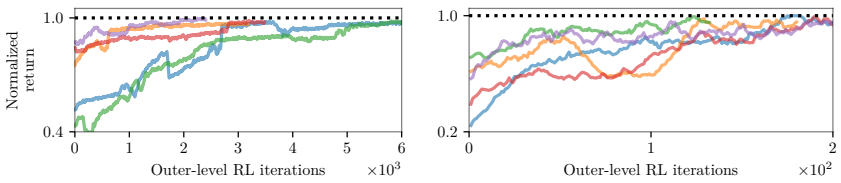

Figure 3: Normalized return of 5 randomized trials for (left) discrete and (right) continuous MDP examples.

for each sample iteratively until convergence using in-sim data under the NN policy, and the same data to estimate the Markov chain sensitivities (12) and (13).

During the bi-level RL training, the simulation parameters are iteratively updated as described in Algorithm 1, until convergence. Since these two problems are solved using exact methods, their resulting policies naturally satisfy the convergence condition for the SPG: $\hat{\varphi}(\phi, \theta) = 0$. The training progress of the outer-level RL is shown in Fig. 3. Its convergence serves as a validation of the sensitivity estimation in Theorem 1.

Quadcopter Control Task. We consider a 2D quadrotor stabilization task implemented in *safe-control-gym*(Yuan et al., 2022), where the goal is to stabilize the quadrotor at an arbitrary reference point from random initial positions. Both the real-world and simulation environments are realized as separate instances of the same simulator. The simulation dynamics are parameterized through the quadrotor mass and length. The ground-truth parameters are $m_0 = 0.033\,\text{kg}$ and $l_0 = 0.039\,\text{m}$. In the simulator, the mass is initialized as $m = \theta_m m_0$ with $\theta_m = 0.5$, while the length is initialized at l_0 since it has little influence on performance. During bi-level PPO training using Algorithm 2, both mass and length parameters are adapted. The policy and critic are implemented using NNs. Further experimental details are provided in Appendix D.3.

Fig. 4 shows that the in-sim policy stabilizes at a different location than the desired one when deployed on the real-world environment, whereas the policy trained with bi-level PPO closes the sim-to-real gap. We did not tune hyperparameters, as this experiment only demonstrates the convergence behavior of bi-level PPO on a simple task (see Appendix D.3). Since the objective is stabilization to a reference, a small adjustment to the mass parameter is sufficient to achieve near-optimal performance. As a result, the estimated mass parameter converges to a value that differs from the ground truth while still yielding optimal performance.

7 Discussions

Although Theorem 1 assumes that the in-sim policy needs to fully converge to a local optimum for applying the IFT, in practice, a partially converged in-sim policy only introduces an additional perturbation to the already noisy real-world gradient driving the outer-level RL. The three-time-scale analysis in Appendix B formally justifies this; bi-level RL convergence does not require the in-sim policy to fully converge for each update to the simulation parameters, but rather requires *timescale separation*: the critic and its sensitivities must stabilize on the fastest timescale, the in-sim policy must evolve on a faster timescale than the simulation parameters, and the outer simulator updates must be sufficiently slow. This is empirically validated in the quadcopter control task. Importantly, the regime in which in-sim convergence matters most is when the outer-level approaches a local optimum and the true gradient norm becomes small; however, in that same regime the outer updates are naturally small, which makes the moving target $\phi^\star(\theta)$ change slowly and allows the in-sim policy to track it with relatively few inner updates (see Appendix B).

The bi-level RL approach is applicable to any in-sim policy optimization scheme, provided one can estimate the sensitivity of the in-sim policy with respect to simulation parameters. Notable examples include bi-level RL schemes for Model Predictive Control (MPC) (Reiter

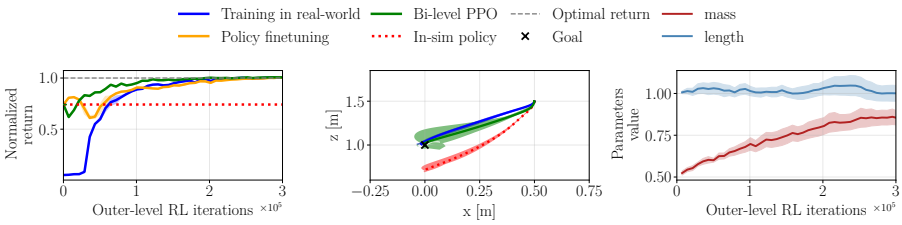

Figure 4: Quadcopter example. (Left) The gap between the optimal return and the performance of the in-sim policy illustrates the sim-to-real gap. Bi-level PPO and direct policy fine-tuning in the real world reduce this gap compared to training the policy purely in simulation. (Middle) Trajectory tracking performance of the different policies. (Right) Evolution of the simulation parameters during bi-level PPO training.

et al., 2025; Kordabad et al., 2023), and Cross Entropy Method (CEM) (Amos and Yarats, 2020; Wan et al., 2024). Consequently, the key contribution of the framework presented in this paper is the sensitivity analysis of the SPG-based RL used to train the in-sim policy. This extends the previous work of (Nikishin et al., 2022) that assumed that the policy is derived from a Q-function via a softmax, allowing the derivation of a model gradient by implicit differentiation of the Bellman equations. The bi-level RL approach is directly applicable to Dyna-style MBRL settings, where the dynamics model and reward functions, which are typically learned for prediction accuracy, can be adapted by an outer-level RL loop to improve the real-world performance.

As demonstrated with PPO, the sensitivity analysis in Theorem 1 could be incorporated into most standard RL algorithms, such as Soft Actor Critic (SAC) (Haarnoja et al., 2018), by differentiating a few extra terms in their policy gradient equations without significant additional complexity. One could also deploy separate RL algorithms in the inner and outer levels. Additionally, both approaches for estimating the critic sensitivities, in Lemma 2 and Lemma 4, are equally applicable, as we show their equivalence in Appendix A.4. However, we favor the unrolled formulation in Lemma 2 due to its lower computational complexity and its better suitability for parallelization, making it the more practical choice in our setting.

In sim-to-real RL, an alternative to adapting the simulator is to fine-tune an in-sim policy directly on real-world data. While this can be effective in some cases, fine-tuning the simulator with bi-level RL offers a more general and flexible approach. By fine-tuning the simulation parameters for sim-to-real optimality, we can exploit a full suite of RL techniques in simulation, including domain randomization. Moreover, learning an optimal simulation model provides valuable insights into which scenarios drive decision-making in practice, thereby enhancing explainability. Additionally, the simulator parameter space can often be kept low-dimensional, as practitioners typically adjust only a small subset of physically meaningful parameters. Fine-tuning this small set of parameters can be computationally more efficient and stable than fine-tuning very large policy networks. Moreover, fine-tuning the in-sim reward parameters provides extra flexibility, considering that sim-to-real optimality can be achieved by adapting only the reward parameters while keeping the simulation model parameters fixed (Gros and Zanon, 2019). Since the simulation model parameters, typically obtained via classical model-fitting approaches (Ljung, 1995), approximate the real-world dynamics with reasonable accuracy, the optimal simulation parameters likely lie within a neighborhood of those estimates. Consequently, in practice, the outer-level RL can be implemented as a local search on the simulation parameter space (Grudic and Ungar, 2000) similar to (Jia et al., 2024).

Limitations

The practical deployment of the bi-level RL framework requires the development of scalable bi-level RL algorithms based on the results of Section 5. Given the challenges inherent in real-world online learning, offline-RL is a suitable framework for the outer-level RL, which

is not addressed in this work. The proposed bi-level RL framework only provides local, not global, sim-to-real optimality, as is typical of gradient-based methods in contrast to global optimization approaches such as Bayesian optimization or evolutionary algorithms. Additionally, sim-to-real optimality is limited by model specification: if the simulator class is not sufficiently expressive, the desired optimality may not be achievable.

Moreover, the proposed bi-level RL approach can be computationally heavy, due to the estimation of the critic sensitivities (9). While the VJP formulation of the outer-level gradient described in Section 5.2 avoids the computational complexity of explicitly forming $\nabla_\theta \phi$, parallelization is still required to make the framework scalable to larger problems.

While bi-level RL accounts for the aleatoric uncertainties in the real-world environment, it does not account for epistemic uncertainties. However, in real-world decision-making problems, epistemic uncertainty can significantly impact the value function estimates, leading to issues such as overestimation. To address this, epistemic uncertainties must be managed using probabilistic models of dynamics, or by penalizing the in-sim selection of out-of-distribution state-action pairs (Kidambi et al., 2020).

One key assumption in our framework is that the simulator and the real system share the same state representation. While this assumption simplifies the analysis, real-world environments often provide only partial or noisy observations (e.g., images or sensor readings). Therefore, extending the bi-level RL to work directly with the observation space is an interesting direction.

The bi-level RL framework requires regularity conditions on the dynamics model and reward functions (Assumption 1). Although this may seem limiting, differentiable simulators are rapidly becoming standard, including domains with discrete state or action spaces and applications such as contact robotics, where dynamics are inherently discontinuous (Newbury et al., 2024). In the case of MBRL, when both dynamics and reward functions are parameterized by NNs, they are inherently differentiable.

A limitation of the proposed sensitivity analysis is the local regularity assumption (Assumption 2) that $\nabla_\phi \hat{\varphi}(\phi, \theta)$ is nonsingular. In deep RL, this Jacobian may become nearly singular due to overparameterization, redundant parameterizations, saturated policies, flat local objectives, or inaccurate critic estimates, even when the in-sim policy has locally converged. In such cases, the sensitivity $\nabla_\theta \phi$ can become poorly conditioned, leading to large and numerically unstable outer-level updates. Practical implementations should therefore regularize or damp the linear solve, use conservative step sizes or trust-region updates, and monitor the conditioning of $\nabla_\phi \hat{\varphi}(\phi, \theta)$.

8 Conclusions

We derived the sensitivity of locally converged RL policies trained with SPG methods in an actor–critic setting with respect to simulation parameters. Based on this sensitivity analysis, we formulated a bi-level RL approach that addresses the objective mismatch problem by adapting simulation parameters using gradients of real-world policy performance, thereby directly coupling the simulation model adaptation objective with the policy performance objective. We further provided a thorough convergence analysis of the proposed bi-level RL approach, along with a proof-of-concept bi-level PPO algorithm. We also discussed practical considerations for implementing this approach, including jointly updating simulation and policy parameters and using VJP for computational efficiency. While this paper establishes the theoretical foundations and mathematical tools for bi-level RL, further algorithmic developments are necessary to scale the approach to large-scale RL problems. Although motivated by achieving sim-to-real optimality in RL, this work may be useful wherever sensitivity properties of SPG methods are of interest.

References

Romina Abachi, Mohammad Ghavamzadeh, and Amir-massoud Farahmand. Policy-Aware Model Learning for Policy Gradient Methods, January 2021.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research*, 22(98):1–76, 2021.

Brandon Amos and Denis Yarats. The Differentiable Cross-Entropy Method. In *Proceedings of the 37th International Conference on Machine Learning*, pages 291–302. PMLR, November 2020.

Akhil S Anand, Arash Bahari Kordabad, Mario Zanon, and Sebastien Gros. Optimality conditions for model predictive control: Rethinking predictive model design. *arXiv preprint arXiv:2412.18268*, 2024a.

Akhil S Anand, Shambhuraj Sawant, Dirk Reinhardt, and Sebastien Gros. Data-driven predictive control and mpc: Do we achieve optimality? *IFAC-PapersOnLine*, 58(15):73–78, 2024b.

Akhil S Anand, Shambhuraj Sawant, Dirk Peter Reinhardt, and Sebastien Gros. Predicting what matters: Training ai models for better decisions. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Vivek S Borkar and Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 100. Springer, 2008.

Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial-learned loss for domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 3521–3528, 2020.

Siyu Chen, Donglin Yang, Jiayang Li, Senmiao Wang, Zhuoran Yang, and Zhaoran Wang. Adaptive model design for markov decision process. In *International Conference on Machine Learning*, pages 3679–3700. PMLR, 2022.

Jack Collins, Shelvin Chand, Anthony Vanderkop, and David Howard. A review of physics simulators for robotic applications. *IEEE Access*, 9:51416–51431, 2021.

Longchao Da, Justin Turnau, Thirulogasankar Pranav Kutralingam, Alvaro Velasquez, Paulo Shakarian, and Hua Wei. A survey of sim-to-real methods in rl: Progress, prospects and challenges with foundation models. *arXiv preprint arXiv:2502.13187*, 2025.

Siddharth Desai, Haresh Karnan, Josiah P Hanna, Garrett Warnell, and Peter Stone. Stochastic grounded action transformation for robot learning in simulation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6106–6111. IEEE, 2020.

Pierluca D'Oro, Alberto Maria Metelli, Andrea Tirinzoni, Matteo Papini, and Marcello Restelli. Gradient-Aware Model-Based Policy Search. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):3801–3808, April 2020. ISSN 2374-3468. doi: 10.1609/aaai.v34i04.5791.

Helei Duan, Bikram Pandit, Mohitvishnu S Gadde, Bart Van Marum, Jeremy Dao, Chanho Kim, and Alan Fern. Learning vision-based bipedal locomotion for challenging terrain. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 56–62. IEEE, 2024.

Marie Duflo. *Random iterative models*, volume 34. Springer Science & Business Media, 2013.

Benjamin Eysenbach, Swapnil Asawa, Shreyas Chaudhari, Sergey Levine, and Ruslan Salakhutdinov. Off-dynamics reinforcement learning: Training for transfer with domain classifiers. *arXiv preprint arXiv:2006.13916*, 2020.

Benjamin Eysenbach, Alexander Khazatsky, Sergey Levine, and Russ R Salakhutdinov. Mismatched no more: Joint model-policy optimization for model-based rl. *Advances in Neural Information Processing Systems*, 35:23230–23243, 2022.

Amir-massoud Farahmand, Andre Barreto, and Daniel Nikovski. Value-aware loss function for model-based reinforcement learning. In *Artificial Intelligence and Statistics*, pages 1486–1494. PMLR, 2017.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

Andrzej Granas, James Dugundji, et al. *Fixed point theory*, volume 14. Springer, 2003.

Christopher Grimm, Andre Barreto, Satinder Singh, and David Silver. The Value Equivalence Principle for Model-Based Reinforcement Learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 5541–5552. Curran Associates, Inc., 2020.

Christopher Grimm, André Barreto, Greg Farquhar, David Silver, and Satinder Singh. Proper value equivalence. *Advances in neural information processing systems*, 34:7773–7786, 2021.

Sébastien Gros and Mario Zanon. Data-driven economic nmpc using reinforcement learning. *IEEE TAC*, 65(2):636–648, 2019.

Gregory Z Grudic and Lyle H Ungar. Localizing search in reinforcement learning. In *AAAI/IAAI*, pages 590–595, 2000.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. Pmlr, 2018.

Josiah Hanna and Peter Stone. Grounded action transformation for robot learning in simulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Nicklas A Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 8387–8406. PMLR, 17–23 Jul 2022.

Sebastian Höfer, Kostas Bekris, Ankur Handa, Juan Camilo Gamboa, Melissa Mozifian, Florian Golemo, Chris Atkeson, Dieter Fox, Ken Goldberg, John Leonard, et al. Sim2real in robotics and automation: Applications and challenges. *IEEE transactions on automation science and engineering*, 18(2):398–400, 2021.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

Chengxing Jia, Fuxiang Zhang, Tian Xu, Jing-Cheng Pang, Zongzhang Zhang, and Yang Yu. Model gradient: Unified model and policy learning in model-based reinforcement learning. *Frontiers of Computer Science*, 18(4):184339, August 2024. ISSN 2095-2228. doi: 10.1007/s11704-023-3150-5.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. *Advances in neural information processing systems*, 33:21810–21823, 2020.

Arash Bahari Kordabad, Dirk Reinhardt, Akhil S Anand, and Sebastien Gros. Reinforcement learning for mpc: Fundamentals and current challenges. *IFAC-PapersOnLine*, 56(2): 5773–5780, 2023.

Steven George Krantz and Harold R Parks. *The implicit function theorem: history, theory, and applications*. Springer Science & Business Media, 2002.

Nathan Lambert, Brandon Amos, Omry Yadan, and Roberto Calandra. Objective mismatch in model-based reinforcement learning. *arXiv preprint arXiv:2002.04523*, 2020.

Jörg Liesen and Zdenek Strakos. *Krylov subspace methods: principles and analysis*. Numerical Mathematics and Scie, 2013.

Zhishuai Liu and Pan Xu. Distributionally robust off-dynamics reinforcement learning: Provable efficiency with linear function approximation. In *International Conference on Artificial Intelligence and Statistics*, pages 2719–2727. PMLR, 2024.

Lennart Ljung. *System identification toolbox: User's guide*. Citeseer, 1995.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

Thomas M Moerland, Joost Broekens, Aske Plaat, Catholijn M Jonker, et al. Model-based reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 16(1): 1–118, 2023.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

Rhys Newbury, Jack Collins, Kerry He, Jiahe Pan, Ingmar Posner, David Howard, and Akansel Cosgun. A review of differentiable simulators. *IEEE Access*, 2024.

Evgenii Nikishin, Romina Abachi, Rishabh Agarwal, and Pierre-Luc Bacon. Control-oriented model-based reinforcement learning with implicit differentiation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7886–7894, 2022.

Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Kanishka Rao, Chris Harris, Alex Irpan, Sergey Levine, Julian Ibarz, and Mohi Khansari. Rl-cyclegan: Reinforcement learning aware simulation-to-real. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11157–11166, 2020.

Rudolf Reiter, Jasper Hoffmann, Dirk Reinhardt, Florian Messerer, Katrin BaumgÃĪrtner, Shamburaj Sawant, Joschka Boedecker, Moritz Diehl, and Sebastien Gros. Synthesis of model predictive control and reinforcement learning: Survey and classification. *arXiv preprint arXiv:2502.02133*, 2025.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609, 2020.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.

Claas Voelcker, Victor Liao, Animesh Garg, and Amir-massoud Farahmand. Value Gradient weighted Model-Based Reinforcement Learning, June 2023.

Weikang Wan, Ziyu Wang, Yufei Wang, Zackory Erickson, and David Held. Difftori: Differentiable trajectory optimization for deep reinforcement and imitation learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 109430–109459. Curran Associates, Inc., 2024.

Ran Wei, Nathan Lambert, Anthony D McDonald, Alfredo Garcia, and Roberto Calandra. A unified view on solving objective mismatch in model-based reinforcement learning. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. Survey Certification.

Yifan Yang, Hao Ban, Minhui Huang, Shiqian Ma, and Kaiyi Ji. Tuning-free bilevel optimization: New algorithms and convergence analysis. *arXiv preprint arXiv:2410.05140*, 2024.

Zhaocong Yuan, Adam W Hall, Siqi Zhou, Lukas Brunke, Melissa Greeff, Jacopo Panerati, and Angela P Schoellig. Safe-control-gym: A unified benchmark suite for safe learning-based control and reinforcement learning in robotics. *IEEE Robotics and Automation Letters*, 7(4):11142–11149, 2022.

Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, pages 737–744. IEEE, 2020.

A Proof of Theorem 1, Lemma 1, and Lemma 2

Here we provide the proof for lemma 1, lemma 2, and theorem 1. We consider the stochastic policy gradient:

$$\hat{\varphi}\left(\phi,\theta\right)=\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}Q_{\theta}^{\pi_{\phi}}\right] \tag{16}$$

where ϕ are the policy parameters, θ are the parameters attached to the model and cost in the simulation-based training environment, and $Q_{\theta}^{\pi_{\phi}}$ is the critic of policy π_{ϕ} *as seen* by the simulations. Distribution $\hat{\rho}_{\theta}^{\pi_{\phi}}$ is the simulated "discounted Markov chain density" associated to policy π_{ϕ}. Assuming that the in-sim policy training sets the policy parameters ϕ such that (16) is zero, ϕ becomes an implicit function of θ.

The real-world (outer-level) policy gradient reads as:

$$\varphi\left(\phi,\theta\right)=\mathbb{E}_{s\sim\rho^{\pi_{\phi}}}\left[\nabla_{\theta}\log\pi_{\phi}Q^{\pi_{\phi}}\right] \tag{17}$$

where $Q^{\pi_{\phi}}$ is the critic of policy π_{ϕ} *as seen* by the real world, and $\rho^{\pi_{\phi}}$ is the "discounted Markov chain density" resulting from using policy π_{ϕ} in the real world. Here, θ implicitly affects $\log\pi_{\phi}$, as any changes in θ would change the local optima for the inner-level resulting in a different policy π_{ϕ}, hence, we have,

$$\varphi\left(\phi,\theta\right)=\mathbb{E}_{s\sim\rho^{\pi_{\phi}}}\left[\nabla_{\theta}\phi\nabla_{\phi}\log\pi_{\phi}Q^{\pi_{\phi}}\right] \tag{18}$$

The key element to compute the world policy gradient is to find the sensitivity $\nabla_{\theta}\phi$ using $\hat{\varphi}\left(\phi,\theta\right)=0$. The IFT readily applies (under some conditions), and

$$\nabla_{\phi}\hat{\varphi}\nabla_{\theta}\phi+\nabla_{\theta}\hat{\varphi}=0 \tag{19}$$

$$\nabla_{\theta}\phi=-\nabla_{\phi}\hat{\varphi}^{-1}\nabla_{\theta}\hat{\varphi} \tag{20}$$

Sensitivities $\nabla_{\theta}\hat{\varphi}$, $\nabla_{\phi}\hat{\varphi}$ can be computed as:

$$\nabla_{\theta}\hat{\varphi}=\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\nabla_{\theta}Q_{\theta}^{\pi_{\phi}}\right]+\nabla_{\theta}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\,Q_{\theta}^{\pi_{\phi}}\right] \tag{21a}$$

$$\nabla_{\phi}\hat{\varphi}=\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi\phi}\log\pi_{\phi}Q_{\theta}^{\pi_{\phi}}+\nabla_{\phi}\log\pi_{\phi}\nabla_{\phi}Q_{\theta}^{\pi_{\phi}}\right]+\nabla_{\phi}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\,Q_{\theta}^{\pi_{\phi}}\right] \tag{21b}$$

where:

- $\nabla_{\phi\phi}\log\pi_{\phi}$ is an explicit expression from the policy function
- $\nabla_{\theta}Q_{\theta}^{\pi_{\phi}}$ is the influence of changing the cost and model on the simulation-based critic *under a fixed policy.*
- $\nabla_{\phi}Q_{\theta}^{\pi_{\phi}}$ is the influence of changing the policy parameters on the simulation-based critic *under a fixed cost and model.*
- $\nabla_{\phi}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\cdot\right]$ is the effect of changing ϕ on the simulation-based distribution $\hat{\rho}_{\theta}^{\pi_{\phi}}$, and the influence of that change on the expected value *under a fixed cost and model.*
- $\nabla_{\theta}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\cdot\right]$ is the effect of changing θ on the simulation-based distribution $\hat{\rho}_{\theta}^{\pi_{\phi}}$, and the influence of that change on the expected value *under a fixed policy.*

From (19):

$$\nabla_{\theta}\phi=-\Big(\underbrace{\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi\phi}\log\pi_{\phi}\,Q_{\theta}^{\pi_{\phi}}+\nabla_{\phi}\log\pi_{\phi}\,\nabla_{\phi}Q_{\theta}^{\pi_{\phi}}\right]}_{\text{Gradient of }\hat{\varphi}\text{ w.r.t. }\phi}+\underbrace{\nabla_{\phi}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\,Q_{\theta}^{\pi_{\phi}}\right]}_{\text{Gradient of dist. }\hat{\rho}_{\theta}^{\pi_{\phi}}\text{ w.r.t. }\phi}\Big)^{-1}$$
$$\Big(\underbrace{\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\,\nabla_{\theta}Q_{\theta}^{\pi_{\phi}}\right]}_{\text{Gradient of }\hat{\varphi}\text{ w.r.t. }\theta}+\underbrace{\nabla_{\theta}\mathbb{E}_{s\sim\hat{\rho}_{\theta}^{\pi_{\phi}}}\left[\nabla_{\phi}\log\pi_{\phi}\,Q_{\theta}^{\pi_{\phi}}\right]}_{\text{Gradient of dist. }\hat{\rho}_{\theta}^{\pi_{\phi}}\text{ w.r.t. }\theta}\Big). \tag{22}$$

Explanation of Theorem 1

The Theorem 1 defines how small perturbations in the simulator parameters θ propagate through the RL process to affect the resulting policy parameters ϕ. The inverse term in the

first line of (11) captures how changes in the policy gradient translate into changes in the stationary policy parameters. This term corresponds to the Jacobian of the policy gradient with respect to the policy parameters. The first expectation inside this term represents the local curvature of the policy objective: the term $\nabla_{\phi\phi} \log \pi_\phi \, Q_\theta^{\pi_\phi}$ captures the direct curvature of the policy log-likelihood weighted by the critic, while the term $\nabla_\phi \log \pi_\phi \, \nabla_\phi Q_\theta^{\pi_\phi}$ accounts for how changes in the policy parameters modify the critic estimates. The second component of this line, $\nabla_\phi \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}[\cdot]$, captures the additional effect that changing the policy parameters alters the state distribution induced by the policy in the simulator, which in turn modifies the policy gradient.

The second line of (11) describes how perturbations in the simulator parameters change the policy gradient itself. The first expectation term captures the effect of simulator parameters on the critic $Q_\theta^{\pi_\phi}$ through the simulator dynamics and reward model. The second term, $\nabla_\theta \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}[\cdot]$, captures how changes in simulator parameters modify the state distribution generated by the simulator, which indirectly alters the policy gradient.

The four components together describe how perturbations in simulator parameters influence the learned policy. Perturbations in the simulator modify both the value estimates and the distribution of simulated trajectories, which alters the policy gradient. The inverse Jacobian then maps these changes in the policy gradient to the resulting shift in the locally optimal policy parameters.

With this, we next unpack the effect of changing ϕ, θ on the simulation-based distribution $\hat{\rho}_\theta^{\pi_\phi}$, and the influence of that change on the expected value. In order to develop that aspect, let us develop the notion of sensitivity over Markov Chains.

A.1 Proof of Lemma 1: Sensitivity over Markov Chains

For compactness, let us consider a generic function $\eta(s,a)$ of the state and action, and its expected value at a specific time stage k:

$$\mathbb{E}_{s\sim\hat{\rho}_{\theta,k}^{\pi_\phi}}[\eta(s_k,a_k)] = \int \eta(s_k,a_k) \int \rho^0(s_0)\ \pi_\phi(a_k|s_k) \prod_{i=0}^{k-1} \mathcal{P}_\theta^{\pi_\phi}(s_{i+1},s_i,a_i)\,\mathrm{d}s_i\mathrm{d}a_i\mathrm{d}s_k \quad (23)$$

where

$$\mathcal{P}_\theta^{\pi_\phi}(s_{i+1},s_i,a_i) = \mathcal{P}_\theta(s_{i+1}|s_i,a_i)\,\pi_\phi(a_i|s_i) \quad (24)$$

where $\mathcal{P}_\theta$ is the simulation model (assumed stochastic here), and ρ^0 the initial condition distribution, assumed fixed here. Differentiating this expected value w.r.t. ϕ, θ (assuming Leibniz rule) then reads as (arguments omitted for simplicity):

$$\begin{aligned}
&\partial \mathbb{E}_{s\sim\hat{\rho}_{\theta,k}^{\pi_\phi}}[\eta(s_k,a_k)] \quad (25)\\
&\quad = \int \eta \int \rho^0 \partial \left(\pi_\phi(a_k|s_k) \prod_{i=0}^{k-1} \mathcal{P}_\theta^{\pi_\phi} \right) \mathrm{d}s_i\mathrm{d}a_i\mathrm{d}s_k \\
&\quad = \int \eta \int \rho^0 \partial \log \left(\pi_\phi(a_k|s_k) \prod_{i=0}^{k-1} \mathcal{P}_\theta^{\pi_\phi} \right) \pi_\phi(a_k|s_k) \prod_{i=0}^{k-1} \mathcal{P}_\theta^{\pi_\phi} \mathrm{d}s_i\mathrm{d}a_i\mathrm{d}s_k \\
&\quad = \int \eta \int \rho^0 \left(\partial \log \pi_\phi(a_k|s_k) + \sum_{i=0}^{k-1} \partial \log \mathcal{P}_\theta^{\pi_\phi} \right) \pi_\phi(a_k|s_k) \prod_{i=0}^{k-1} \mathcal{P}_\theta^{\pi_\phi} \mathrm{d}s_i\mathrm{d}a_i\mathrm{d}s_k
\end{aligned}$$

Which has the expected value form:

$$\partial\, \mathbb{E}_{s\sim\hat{\rho}_{\theta,k}^{\pi_\phi}}\left[\eta(s_k,a_k)\right] = \mathbb{E}_{\substack{s_0\sim\rho^0,\, a_i\sim\pi_\phi,\\ s_{i+1}\sim\mathcal{P}_\theta^{\pi_\phi}}}\left[\eta(s_k,a_k) \left(\partial \log \pi_\phi(a_k\,|\,s_k) + \sum_{i=0}^{k-1} \partial \log \mathcal{P}_\theta^{\pi_\phi} \right) \right]. \quad (26)$$

Computational aspect

In practice, (26) will be evaluated from a model sampling. Let us label

$$s_0^j, a_0^j, \dots, s_N^j, a_N^j \tag{27}$$

the sampled trajectory $j \in \{1, \dots, n\}$ of length N. The expected value approximation would then read:

$$\mathbb{E}_{s\sim\hat{\rho}_{\theta,k}^{\pi_\phi}}\left[\eta\left(s_k, a_k\right)\right] \approx \frac{1}{n}\sum_{j=1}^{n} \eta\left(s_k^j, a_k^j\right) \tag{28}$$

Its sensitivities would then read as:

$$\partial\mathbb{E}_{s\sim\hat{\rho}_{\theta,k}^{\pi_\phi}}\left[\eta\left(s_k, a_k\right)\right] \approx \frac{1}{n}\sum_{j=1}^{n} \eta\left(s_k^j, a_k^j\right) \otimes \left(W_k^j + \partial \log \pi_\phi\left(a_k^j \middle| s_k^j\right)\right) \tag{29}$$

where

$$W_k^j = \sum_{i=0}^{k-1} \partial \log \mathcal{P}_\theta^{\pi_\phi}\left(s_{i+1}^j \middle| s_i^j, a_i^j\right) \tag{30}$$

Note that W_k^j can be carried forward as a dynamic alongside the model trajectory, i.e.

$$W_k^j = W_{k-1}^j + \partial \log \mathcal{P}_\theta^{\pi_\phi}\left(s_k^j \middle| s_{k-1}^j, a_{k-1}^j\right), \quad W_0^j = 0 \tag{31}$$

Expected value over full Markov Chain

For policy gradient methods, we need to compute objects in the form

$$\mathbb{E}\left[\sum_{k=0}^{\infty} \eta\left(s_k, a_k\right)\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} \eta\left(s_k^j, a_k^j\right) \tag{32}$$

and process their sensitivities. There is no significant difference from Sec. A.1, i.e. the sensitivities read as:

$$\partial\mathbb{E}\left[\sum_{k=0}^{\infty} \eta\left(s_k, a_k\right)\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} \eta\left(s_k^j, a_k^j\right) \otimes \left(W_k^j + \partial \log \pi_\phi\left(a_k^j \middle| s_k^j\right)\right) \tag{33}$$

Efforts can be put in computing (33) in the most efficient way in terms of flops and memory requirements, but that may be best left for other people to figure out.

Sensitivity for the policy gradient

We can then provide computational expressions for the policy gradient. We observe that

$$\begin{aligned}\nabla_\phi\mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log \pi_\phi\, Q_\theta^{\pi_\phi}\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} &\nabla_\phi \log \pi_\phi\left(s_k^j\right)\, Q_\theta^{\pi_\phi}\left(s_k^j, a_k^j\right) \\ &\otimes \left(W_{\phi,k}^j + \partial \log \pi_\phi\left(a_k^j \middle| s_k^j\right)\right)\end{aligned} \tag{34}$$

where $W_{\phi,0}^j = 0$ and

$$\begin{aligned} W_{\phi,k}^j &= W_{\phi,k-1}^j + \nabla_\phi \log \mathcal{P}_\theta^{\pi_\phi}\left(s_k^j, s_{k-1}^j, a_{k-1}^j\right) \\ &= W_{\phi,k-1}^j + \nabla_\phi \log \pi_\phi\left(a_{k-1}^j \middle| s_{k-1}^j\right)\end{aligned} \tag{35}$$

Similarly

$$\begin{aligned}\nabla_\theta\mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\left[\nabla_\phi \log \pi_\phi\, Q_\theta^{\pi_\phi}\right] \approx \frac{1}{nN}\sum_{k=0}^{N}\sum_{j=1}^{n} &\nabla_\phi \log \pi_\phi\left(s_k^j\right)\, Q_\theta^{\pi_\phi}\left(s_k^j, a_k^j\right) \\ &\otimes \left(W_{\theta,k}^j + \partial \log \pi_\phi\left(a_k^j \middle| s_k^j\right)\right)\end{aligned} \tag{36}$$

where $W^j_{\theta,0} = 0$ and

$$\begin{aligned} W^j_{\theta,k} &= W^j_{\theta,k-1} + \nabla_\theta \log \mathcal{P}_\theta^{\pi_\phi}\left(s^j_k, s^j_{k-1}, a^j_{k-1}\right) \\ &= W^j_{\theta,k-1} + \nabla_\theta \log \mathcal{P}_\theta\left(s^j_k \middle| s^j_{k-1}, a^j_{k-1}\right) \end{aligned} \tag{37}$$

These terms ought to be added to (21b), (21a) respectively. In implementations, these terms would simply weigh the contributions in forming the policy gradient sensitivities.

A.2 Proof of Lemma 2: Sensitivities of Policy Evaluation

A.2.1 Operator-theoretic preliminaries

We work on the state–action space $\mathcal{S}\times\mathcal{A}$. For the *in-sim* fast layer, write the transition kernel and reward as $\mathcal{P}_\theta(\mathrm{d}s' \mid s,a)$ and $R_\theta(s,a)$. Let $\pi_\phi(\mathrm{d}a \mid s)$ be the policy. The discount factor is given by $\gamma \in (0,1)$. Let $\mathcal{B}_\infty$ be the Banach space of bounded measurable $f : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ with sup-norm $\|f\|_\infty = \sup_{(s,a)} |f(s,a)|$.

To interpret likelihood-ratio / score terms such as $\nabla_\phi \log \pi_\phi(a \mid s)$ and $\nabla_\theta \log \mathcal{P}_\theta(s' \mid s,a)$, we assume there exist fixed reference measures $\lambda_\mathcal{A}$ on $\mathcal{A}$ and $\lambda_\mathcal{S}$ on $\mathcal{S}$ such that $\pi_\phi(\cdot \mid s) \ll \lambda_\mathcal{A}$ for all s and $\mathcal{P}_\theta(\cdot \mid s,a) \ll \lambda_\mathcal{S}$ for all (s,a). We denote the corresponding densities by $\pi_\phi(a \mid s)$ and $\mathcal{P}_\theta(s' \mid s,a)$, so the above log/score expressions are well-defined.

Markov and Bellman operators. Given any bounded function $f : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$, the policy–induced Markov operator $P_\theta^{\pi_\phi} : \mathcal{B}_\infty \to \mathcal{B}_\infty$ (is the "one-step look-ahead" operator for the Markov chain over state–action pairs), maps f to a new function as

$$(P_\theta^{\pi_\phi} f)(s,a) \;:=\; \int_\mathcal{S}\int_\mathcal{A} f(s',a')\,\pi_\phi(\mathrm{d}a' \mid s')\,\mathcal{P}_\theta(\mathrm{d}s' \mid s,a) \;=\; \mathbb{E}\big[f(s',a') \mid s_0 = s,\, a_0 = a\big]. \tag{38}$$

Then $\|P_\theta^{\pi_\phi} f\|_\infty \le \|f\|_\infty$. The (discounted) Bellman operator for action values is

$$(\hat{T}_{\theta,\phi} Q_\theta) \;=\; R_\theta + \gamma\,(P_\theta^{\pi_\phi} Q_\theta).$$

$\hat{T}_{\theta,\phi}$ is a γ-contraction on $(\mathcal{B}_\infty, \|\cdot\|_\infty)$, hence the inverse of $(I - \gamma P_\theta^{\pi_\phi})$ exists and admits a unique fixed point $Q_\theta^{\pi_\phi} \in \mathcal{B}_\infty$ solving $(I - \gamma P_\theta^{\pi_\phi}) Q_\theta^{\pi_\phi} = R_\theta$. (Puterman, 2014).

Poisson equations for sensitivities Under the policy $\pi = \pi_{\phi(\theta)}$, the critic $Q_\theta^{\pi_{\phi(\theta)}}$ solves the Bellman fixed point equation:

$$Q_\theta^{\pi_\phi} \;=\; R_\theta \;+\; \gamma\, P_\theta^{\pi_{\phi(\theta)}}\, Q_\theta^{\pi_\phi}. \tag{39}$$

For a parameter $\eta \in \{\phi, \theta\}$, $\nabla_\eta Q_\theta^{\pi_\phi}$ can be derived by differentiating the Bellman fixed point (39) with respect to η

$$\nabla_\eta Q_\theta^{\pi_\phi} = \nabla_\eta R_\theta + \gamma\,(\nabla_\eta P_\theta^{\pi_\phi})\, Q_\theta^{\pi_\phi} + \gamma\, P_\theta^{\pi_\phi} \nabla_\eta Q_\theta^{\pi_\phi}. \tag{40}$$

Rearranging terms gives the linear system

$$\big(I - \gamma P_\theta^{\pi_\phi}\big) \nabla_\eta Q_\theta^{\pi_\phi} = b_\eta, \tag{41}$$

where

$$b_\eta := \nabla_\eta R_\theta + \gamma\,(\nabla_\eta P_\theta^{\pi_\phi})\, Q_\theta^{\pi_\phi}. \tag{42}$$

With Neumann series $(I - \gamma P_\theta^{\pi_\phi})^{-1} = \sum_{t\ge 0} (\gamma P_\theta^{\pi_\phi})^t$ which converges because $\gamma < 1$, applying it to (41) yields

$$\nabla_\eta Q_\theta^{\pi_\phi} = \big(I - \gamma P_\theta^{\pi_\phi}\big)^{-1} b_\eta = \sum_{t=0}^{\infty} (\gamma P_\theta^{\pi_\phi})^t\, b_\eta. \tag{43}$$

Now, the following Lemma provides the sensitivity of the Markov operator.

Lemma 3. *For any bounded f, the sensitivity of the Markov operator w.r.t. θ and ϕ are given by:*

$$\begin{aligned}\nabla_\phi P_\theta^{\pi_\phi} f(s,a) &= \mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot\,|\,s,a)\ a'\sim\pi_\phi(\cdot\,|\,s')}[\,\nabla_\phi \log \pi_\phi(a'|s')\, f(s',a')\,|\,s,a]\,,\\ \nabla_\theta P_\theta^{\pi_\phi} f(s,a) &= \mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot\,|\,s,a)\ a'\sim\pi_\phi(\cdot\,|\,s')}[\,\nabla_\theta \log \mathcal{P}_\theta(s'|s,a)\, f(s',a')\,|\,s,a]\ ,\end{aligned} \tag{44}$$

respectively.

Proof. For any bounded f and fixed (s,a),

$$\begin{aligned}(P_\theta^{\pi_\phi} f)(s,a) &= \int_{\mathcal{S}}\int_{\mathcal{A}} f(s',a')\,\pi_\phi(a'\mid s')\,\mathcal{P}_\theta(s'\mid s,a)\,da'\,ds'\\ &=: \int_{\mathcal{S}} g_\phi(s')\,\mathcal{P}_\theta(s'\mid s,a)\,ds', \qquad g_\phi(s') := \int_{\mathcal{A}} f(s',a')\,\pi_\phi(a'\mid s')\,da'.\end{aligned} \tag{45}$$

Only $g_\phi(s')$ depends on ϕ in (45). Differentiate under the inner integral (over a'):

$$\begin{aligned}\nabla_\phi\, g_\phi(s') &= \int_{\mathcal{A}} f(s',a')\,\nabla_\phi\,\pi_\phi(a'\mid s')\,\mathrm{d}a'\\ &= \int_{\mathcal{A}} f(s',a')\,\pi_\phi(a'\mid s')\,\nabla_\phi \log \pi_\phi(a'\mid s')\,\mathrm{d}a'\\ &= \mathbb{E}_{a'\sim\pi_\phi(\cdot|s')}\big[f(s',a')\,\nabla_\phi \log \pi_\phi(a'\mid s')\big].\end{aligned} \tag{46}$$

Now differentiate (45) and use that $\mathcal{P}_\theta$ does not depend on ϕ:

$$\begin{aligned}\nabla_\phi\,(P_\theta^{\pi_\phi} f)(s,a) &= \int_{\mathcal{S}} \nabla_\phi\, g_\phi(s')\,\mathcal{P}_\theta(s'\mid s,a)\,\mathrm{d}s'\\ &= \int_{\mathcal{S}} \mathbb{E}_{a'\sim\pi_\phi(\cdot|s')}\big[f(s',a')\,\nabla_\phi \log \pi_\phi(a'\mid s')\big]\,\mathcal{P}_\theta(s'\mid s,a)\,\mathrm{d}s'\\ &= \mathbb{E}[\nabla_\phi \log \pi_\phi(a'\mid s')\, f(s',a')\mid s,a]\,.\end{aligned} \tag{47}$$

In (45), only $\mathcal{P}_\theta(\cdot\mid s,a)$ depends on θ as we keep the policy fixed:

$$\nabla_\theta\,(P_\theta^{\pi_\phi} f)(s,a) = \int_{\mathcal{S}} g_\phi(s')\,\nabla_\theta\,\mathcal{P}_\theta(s'\mid s,a)\,\mathrm{d}s'. \tag{48}$$

Assuming $\mathcal{P}_\theta(s'\mid s,a) > 0$ (a.e.) and differentiable in θ,

$$\nabla_\theta\,\mathcal{P}_\theta(s'\mid s,a) = \mathcal{P}_\theta(s'\mid s,a)\,\nabla_\theta \log \mathcal{P}_\theta(s'\mid s,a).$$

Therefore

$$\begin{aligned}\nabla_\theta\,(P_\theta^{\pi_\phi} f)(s,a) &= \int_{\mathcal{S}} g_\phi(s')\,\mathcal{P}_\theta(s'\mid s,a)\,\nabla_\theta \log \mathcal{P}_\theta(s'\mid s,a)\,\mathrm{d}s'\\ &= \mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot|s,a)}\big[\,g_\phi(s')\,\nabla_\theta \log \mathcal{P}_\theta(s'\mid s,a)\,\big]\\ &= \mathbb{E}[\nabla_\theta \log \mathcal{P}_\theta(s'|s,a)\, f(s',a')\mid s,a]\,,\end{aligned} \tag{49}$$

□

A.2.2 Critic Sensitivities based on fixed point

(i) Derivative of Q w.r.t. ϕ: Combining (41) and (47) yields,

$$(I-\gamma P_\theta^{\pi_\phi})\,\nabla_\phi Q_\theta^{\pi_\phi} \;=\; \nabla_\phi R_\theta \;+\; \gamma\,\mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot\,|\,s,a)\ a'\sim\pi_\phi(\cdot\,|\,s')}\big[\nabla_\phi \log \pi_\phi(a'|s')\, Q_\theta^{\pi_\phi}(s',a')\mid s,a\big]\,, \tag{50}$$

$$\nabla_\phi Q_\theta^{\pi_\phi} \;=\; \sum_{t=0}^{\infty}(\gamma P_\theta^{\pi_\phi})^t\Big(\nabla_\phi R_\theta + \gamma\,\nabla_\phi P_\theta^{\pi_\phi}\, Q_\theta^{\pi_\phi}\Big), \tag{51}$$

$$\nabla_\phi Q_\theta^{\pi_\phi}(s,a) \;=\; \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^{t+1}\,\nabla_\phi \log \pi_\phi(a_{t+1}\mid s_{t+1})\, Q_\theta^{\pi_\phi}(s_{t+1},a_{t+1})\;\middle|\; s_0{=}s, a_0{=}a\right]. \tag{52}$$

(ii) Derivative w.r.t. θ: Combining (41) and (49)

$$(I - \gamma P_\theta^{\pi_\phi})\,\nabla_\theta Q_\theta^{\pi_\phi} \;=\; \nabla_\theta R_\theta \;+\; \gamma\,\mathbb{E}\big[\nabla_\theta \log \mathcal{P}_\theta(s'|S,A)\,V_\theta^{\pi_\phi}(s') \mid s_0{=}s,\, a_0{=}a\big]\,, \tag{53}$$

$$\nabla_\theta Q_\theta^{\pi_\phi} \;=\; \sum_{t=0}^{\infty}(\gamma P_\theta^{\pi_\phi})^t\Big(\nabla_\theta R_\theta + \gamma\,\nabla_\theta P_\theta^{\pi_\phi}\,Q_\theta^{\pi_\phi}\Big), \tag{54}$$

$$\nabla_\theta Q_\theta^{\pi_\phi}(s,a) = \mathbb{E}\Bigg[\sum_{t=0}^{\infty}\gamma^t\,\nabla_\theta R_\theta(s_t,a_t) \;+ \tag{55}$$

$$\sum_{t=0}^{\infty}\gamma^{t+1}\,\nabla_\theta \log \mathcal{P}_\theta(s_{t+1}\mid s_t,a_t)\,Q_\theta^{\pi_\phi}(s_{t+1},a_{t+1})\;\Bigg|\; s_0=s, a_0=a\Bigg]. \tag{56}$$

If the simulator is differentiable, we can use the corresponding Reparameterization Gradient (RP) term instead of the Likelihood-ratio Gradient (LR) terms. Assume independent noise sources $\epsilon_t \sim p(\epsilon)$ and $\xi_{t+1} \sim p(\xi)$, which do not depend on (θ,ϕ). Then the simulator dynamics $\mathcal{P}_\theta$ and the policy π_ϕ admit the following reparameterized representations:

$$s_{t+1} = \mathcal{P}_\theta(s_t,a_t,\epsilon_t), \tag{57}$$

$$a_{t+1} = \pi_\phi(s_{t+1},\xi_{t+1}), \tag{58}$$

where $\mathcal{P}_\theta$ and g_ϕ are deterministic, differentiable mappings. In this form, sampling from $\mathcal{P}_\theta$ or π_ϕ is expressed as pushing forward a parameter-free noise distribution through a parameterized transformation.

$$\nabla_\theta Q_\theta^{\pi_\phi}(s,a) = \mathbb{E}\Bigg[\sum_{t=0}^{\infty}\gamma^t\,\nabla_\theta R_\theta(s_t,a_t) + \sum_{t=0}^{\infty}\gamma^{t+1}\,\nabla_{s_{t+1}}Q_\theta^{\pi_\phi}\big(s_{t+1},a_{t+1}\big)\,\nabla_\theta\mathcal{P}_\theta(s_t,a_t,\epsilon_t)\;\Big|\; s_0=s,\; a_0=a\Bigg], \tag{59}$$

Where,

$$\nabla_{s_{t+1}}Q_\theta^{\pi_\phi}(s_{t+1},a_{t+1}) = \nabla_s Q_\theta^{\pi_\phi}(s_{t+1},a_{t+1}) \;+\; \nabla_a Q_\theta^{\pi_\phi}(s_{t+1},a_{t+1})\,\nabla_{s_{t+1}}\pi_\phi(s_{t+1},\xi_{t+1}). \tag{60}$$

$$\nabla_\phi Q_\theta^{\pi_\phi}(s,a) = \mathbb{E}\Bigg[\sum_{t=0}^{\infty}\gamma^{t+1}\,\nabla_{a_{t+1}}Q_\theta^{\pi_\phi}\big(s_{t+1},a_{t+1}\big)\,\nabla_\phi\pi_\phi(s_{t+1},\xi_{t+1})\;\Bigg|\; s_0=s,\; a_0=a\Bigg]. \tag{61}$$

A.3 Alternative Sensitivity estimation for the policy evaluation

Lemma 4 (Sensitivities of Q^π and V^π)**.** *Given the Bellman conditions,*

$$Q^\pi(s,a) = R(s,a) + \gamma\mathbb{E}_{s_+\sim f}[V^\pi(s_+)], \quad \textit{and} \quad V^\pi(s) = \mathbb{E}_{a\sim\pi}[Q^\pi(s,a)],$$

the sensitivity of Q^π and V^π are given by:

$$\partial Q^\pi(s,a) = \partial R(s,a) + \gamma\mathbb{E}_{s_+\sim f}\left[\partial V^\pi(s_+) + V^\pi(s_+)\partial\log f(s_+|s,a)\right], \tag{62a}$$

$$\partial V^\pi(s) = \mathbb{E}_{a\sim\pi}\left[^\pi(s,a) + Q^\pi(s,a)\partial\log\pi(a|s)\right]. \tag{62b}$$

Proof. The simulation-based critic ideally satisfies the following equations:

$$Q_\theta^{\pi_\phi}(s,a) = R_\theta(s,a) + \gamma\mathbb{E}_{s_+\sim\mathcal{P}_\theta}\left[V_\theta^{\pi_\phi}(s_+)\,\middle|\,s,a\right] \tag{63a}$$

$$V_\theta^{\pi_\phi}(s) = \mathbb{E}_{a\sim\pi_\phi}\left[Q_\theta^{\pi_\phi}(s,a)\right] \tag{63b}$$

To compute the sensitivities "safely", we observe that (63) in full reads as:

$$Q_\theta^{\pi_\phi}(s,a) = R_\theta(s,a) + \gamma\int V_\theta^{\pi_\phi}(s_+)\,\mathcal{P}_\theta(s_+|s,a)\,\mathrm{d}s_+ \tag{64a}$$

$$V_\theta^{\pi_\phi}(s) = \int Q_\theta^{\pi_\phi}(s,a)\,\pi_\phi(a|s)\,\mathrm{d}a \tag{64b}$$

Their sensitivities are then given by:

$$\begin{aligned}\partial Q_\theta^{\pi_\phi}(s,a) = {}& \partial R_\theta(s,a) + \gamma \int \partial V_\theta^{\pi_\phi}(s_+)\,\mathcal{P}_\theta(s_+|s,a)\,\mathrm{d}s_+ \\ & + \gamma \int V_\theta^{\pi_\phi}(s_+)\,\mathcal{P}_\theta(s_+|s,a)\,\partial \log \mathcal{P}_\theta(s_+|s,a)\,\mathrm{d}s_+ \end{aligned} \tag{65a}$$

$$\partial V_\theta^{\pi_\phi}(s) = \int \partial Q_\theta^{\pi_\phi}(s,a)\,\pi_\phi(a|s)\,\mathrm{d}a + \int Q_\theta^{\pi_\phi}(s,a)\,\pi_\phi(a|s)\,\partial \log \pi_\phi(a|s)\,\mathrm{d}a \tag{65b}$$

In the compact "expected value" format, they then read as:

$$\partial Q_\theta^{\pi_\phi}(s,a) = \partial R_\theta(s,a) + \gamma \mathbb{E}_{s_+\sim\mathcal{P}_\theta(s,a)}\left[\partial V_\theta^{\pi_\phi}(s_+) + V_\theta^{\pi_\phi}(s_+)\,\partial \log \mathcal{P}_\theta(s_+|s,a)\right] \tag{66a}$$

$$\partial V_\theta^{\pi_\phi}(s) = \mathbb{E}_{a\sim\pi_\phi(a|s)}\left[\partial Q_\theta^{\pi_\phi}(s,a) + Q_\theta^{\pi_\phi}(s,a)\,\partial \log \pi_\phi(a|s)\right] \tag{66b}$$

□

Note that this "format" of sensitivity (Q and V split) is possibly best for a critic that has function approximators dedicated to V and Q separately (or rather V and A).

Sensitivity of the Policy Evaluation w.r.t. θ and ϕ

We want to derive separately the two sensitivities

$$\nabla_\theta Q_\theta^{\pi_\phi} \quad \text{and} \quad \nabla_\phi Q_\theta^{\pi_\phi},$$

starting from the general ("compact") policy-evaluation equations (66): Here, *each* of the two sensitivity questions below assumes that the *other* parameter is held fixed.

1. Sensitivity w.r.t. θ, keeping ϕ fixed

We want

$$\nabla_\theta Q_\theta^{\pi_\phi}(s,a) \quad \text{and} \quad \nabla_\theta V_\theta^{\pi_\phi}(s).$$

Starting from (66):

$$\nabla_\theta Q_\theta^{\pi_\phi}(s,a) \;=\; \nabla_\theta R_\theta(s,a) + \gamma \mathbb{E}_{s_+\sim\mathcal{P}_\theta(\cdot|s,a)}\Big[\nabla_\theta V_\theta^{\pi_\phi}(s_+) + V_\theta^{\pi_\phi}(s_+)\,\nabla_\theta \log \mathcal{P}_\theta(s_+ \mid s,a)\Big]. \tag{67}$$

Since, π_ϕ is fixed w.r.t. θ, hence: $\nabla_\theta \log \pi_\phi(a \mid s) \;=\; 0$. We get

$$\nabla_\theta V_\theta^{\pi_\phi}(s) = \mathbb{E}_{a\sim\pi_\phi(\cdot|s)}\big[\nabla_\theta Q_\theta^{\pi_\phi}(s,a)\big]. \tag{68}$$

These two coupled equations show how to backpropagate the effect of changing θ (the model or cost parameters) *under a fixed policy* π_ϕ.

2. Sensitivity w.r.t. ϕ, keeping θ fixed

Now we want

$$\nabla_\phi Q_\theta^{\pi_\phi}(s,a) \quad \text{and} \quad \nabla_\phi V_\theta^{\pi_\phi}(s),$$

with θ fixed. That means

$$\nabla_\phi \log \mathcal{P}_\theta(s_+ \mid s,a) \;=\; 0\,, \tag{69}$$

and the reward sensitivity is

$$\nabla_\phi R_\theta(s,a) \;=\; 0\,. \tag{70}$$

From (66) again:

$$\nabla_\phi Q_\theta^{\pi_\phi}(s,a) \;= \gamma\, \mathbb{E}_{s_+\sim\mathcal{P}_\theta(\cdot|s,a)}\Big[\nabla_\phi V_\theta^{\pi_\phi}(s_+)\Big]\,, \tag{71}$$

$$\nabla_\phi V_\theta^{\pi_\phi}(s) \;= \mathbb{E}_{a\sim\pi_\phi(\cdot|s)}\Big[\nabla_\phi Q_\theta^{\pi_\phi}(s,a) \;+\; Q_\theta^{\pi_\phi}(s,a)\,\nabla_\phi \log \pi_\phi(a \mid s)\Big]. \tag{72}$$

Here, we iteratively compute the required gradients, $\nabla_\theta Q_\theta^{\pi_\phi}, \nabla_\theta V_\theta^{\pi_\phi}, \nabla_\phi Q_\theta^{\pi_\phi}, \nabla_\phi Q_\theta^{\pi_\phi}$, similar to temporal difference learning in value-based methods. For the illustrative examples, we calculate these values till convergence, but they can be approximated using a generic function approximator for bigger problems.

A.4 Equivalency between the two critic sensitivity approximations

We first treat the two partial derivatives with the *other* parameter held fixed.

Sensitivity w.r.t. θ (policy π_ϕ fixed). Differentiating the "compact" policy-evaluation equations yields

$$\nabla_\theta Q_\theta^{\pi_\phi}(s,a) = \nabla_\theta R_\theta(s,a) + \gamma\,\mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot|s,a)}\Big[\nabla_\theta V_\theta^{\pi_\phi}(s') + V_\theta^{\pi_\phi}(s')\,\nabla_\theta \log \mathcal{P}_\theta(s' \mid s,a)\Big], \tag{73}$$

$$\nabla_\theta V_\theta^{\pi_\phi}(s) = \mathbb{E}_{a\sim\pi_\phi(\cdot|s)}\big[\nabla_\theta Q_\theta^{\pi_\phi}(s,a)\big]. \tag{74}$$

Substitute (74) into (73) and recognize the operator $P_\theta^{\pi_\phi}$ and the score identity:

$$\nabla_\theta Q_\theta^{\pi_\phi} = \nabla_\theta R_\theta + \gamma\,P_\theta^{\pi_\phi}\nabla_\theta V_\theta^{\pi_\phi} + \gamma\Big(\nabla_\theta P_\theta^{\pi_\phi}\Big)Q_\theta^{\pi_\phi} \tag{75}$$

$$= \nabla_\theta R_\theta + \gamma\,P_\theta^{\pi_\phi}\nabla_\theta Q_\theta^{\pi_\phi} + \gamma\Big(\nabla_\theta P_\theta^{\pi_\phi}\Big)Q_\theta^{\pi_\phi}. \tag{76}$$

The equality $P_\theta^{\pi_\phi}\nabla_\theta V_\theta^{\pi_\phi} = P_\theta^{\pi_\phi}\nabla_\theta Q_\theta^{\pi_\phi}$ is proved in Lemma 5 below. By rearranging, we get the Bellman fixed point equation,

$$(I - \gamma P_\theta^{\pi_\phi})\,\nabla_\theta Q_\theta^{\pi_\phi} \;=\; \nabla_\theta R_\theta \;+\; \gamma\Big(\nabla_\theta P_\theta^{\pi_\phi}\Big)Q_\theta^{\pi_\phi}. \tag{77}$$

Sensitivity w.r.t. ϕ (dynamics $\mathcal{P}_\theta$ fixed). Similarly, with $\nabla_\phi R_\theta \equiv 0$ and $\nabla_\phi \log \mathcal{P}_\theta \equiv 0$,

$$\nabla_\phi Q_\theta^{\pi_\phi}(s,a) = \gamma\,\mathbb{E}_{s'\sim\mathcal{P}_\theta(\cdot|s,a)}\Big[\nabla_\phi V_\theta^{\pi_\phi}(s')\Big], \tag{78}$$

$$\nabla_\phi V_\theta^{\pi_\phi}(s) = \mathbb{E}_{a\sim\pi_\phi(\cdot|s)}\Big[\nabla_\phi Q_\theta^{\pi_\phi}(s,a) + Q_\theta^{\pi_\phi}(s,a)\,\nabla_\phi \log \pi_\phi(a \mid s)\Big]. \tag{79}$$

Substitute (79) into (78) we get the Bellman fixed point equation:

$$\nabla_\phi Q_\theta^{\pi_\phi} = \gamma\,P_\theta^{\pi_\phi}\nabla_\phi V_\theta^{\pi_\phi} + \gamma\Big(\nabla_\phi P_\theta^{\pi_\phi}\Big)Q_\theta^{\pi_\phi}, \quad\Longleftrightarrow\quad (I - \gamma P_\theta^{\pi_\phi})\,\nabla_\phi Q_\theta^{\pi_\phi} \;=\; \gamma\Big(\nabla_\phi P_\theta^{\pi_\phi}\Big)Q_\theta^{\pi_\phi}. \tag{80}$$

Proving $P_\theta^{\pi_\phi}\nabla_\theta V_\theta^{\pi_\phi} = P_\theta^{\pi_\phi}\nabla_\theta Q_\theta^{\pi_\phi}$

Lemma 5. *Let $Q_\theta^{\pi_\phi} : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ be bounded and define $V_\theta^{\pi_\phi}(s) := \mathbb{E}_{a\sim\pi_\phi(\cdot|s)}[Q_\theta^{\pi_\phi}(s,a)]$. Then*

$$(P_\theta^{\pi_\phi}V_\theta^{\pi_\phi})(s,a) \;=\; (P_\theta^{\pi_\phi}Q_\theta^{\pi_\phi})(s,a) \qquad \forall (s,a)\in\mathcal{S}\times\mathcal{A}.$$

Proof. By the definition (38) of $P_\theta^{\pi_\phi}$,

$$(P_\theta^{\pi_\phi}V_\theta^{\pi_\phi})(s,a) = \int_{\mathcal{S}} V_\theta^{\pi_\phi}(s')\,\mathcal{P}_\theta(\mathrm{d}s' \mid s,a) = \int_{\mathcal{S}}\Big(\int_{\mathcal{A}} Q_\theta^{\pi_\phi}(s',a')\,\pi_\phi(\mathrm{d}a' \mid s')\Big)\,\mathcal{P}_\theta(\mathrm{d}s' \mid s,a)$$

$$= \int_{\mathcal{S}}\int_{\mathcal{A}} Q_\theta^{\pi_\phi}(s',a')\,\pi_\phi(\mathrm{d}a' \mid s')\,\mathcal{P}_\theta(\mathrm{d}s' \mid s,a) = (P_\theta^{\pi_\phi}Q_\theta^{\pi_\phi})(s,a).$$

$P_\theta^{\pi_\phi}$ already does *both* "go to the next state" and "sample the next action under π_ϕ"; irrespective of if the average is taken over a' inside $V_\theta^{\pi_\phi}$ first, or let $P_\theta^{\pi_\phi}$ average over a' directly, the result is the same. □

Coupled (Q/V) critic sensitivity recursions and the Poisson/operator sensitivity equation describe the *same* linear system. Lemma 5 explains why we can eliminate ∇V and obtain the compact operator form $(I - \gamma P_\theta^{\pi_\phi})\nabla_\eta Q_\theta^{\pi_\phi} = b_\eta$.

B Convergence Analysis of Bi-Level SPG via Three Timescales

In this section, we establish almost-sure convergence of the full three-timescale scheme where (i) the critic and its sensitivities are estimated on the *fastest* timescale, (ii) the inner (in-sim) policy parameter ϕ on a *medium* timescale, and (iii) the outer simulation parameter θ on the *slowest* timescale. The two-timescale and nested/single-level cases are immediate corollaries.

Lemma 6 (Affine contraction for sensitivities)**.** *Fix (ϕ, θ) and define, on the Banach space $\big(\mathcal{B}_\infty, \|\cdot\|_\infty\big)$ of bounded measurable functions on $\mathcal{S} \times \mathcal{A}$, the affine map*

$$\mathcal{C}_\eta : \ \nabla_\eta Q_\theta^{\pi_\phi} \ \mapsto \ b_\eta \ + \ \gamma\, P_\theta^{\pi_\phi}\, \nabla_\eta Q_\theta^{\pi_\phi}, \qquad b_\eta \ := \ \nabla_\eta R_\theta \ + \ \gamma\, (\nabla_\eta P_\theta^{\pi_\phi})\, Q_\theta^{\pi_\phi},$$

where $\eta \in \{\phi, \theta\}$ and $\gamma \in (0, 1)$. Then $\mathcal{C}_\eta$ is a γ-contraction in $\|\cdot\|_\infty$, hence admits a unique fixed point $\nabla_\eta Q_\theta^{\pi_\phi} \in \mathcal{B}_\infty$ satisfying

$$\nabla_\eta Q_\theta^{\pi_\phi} \ = \ (I - \gamma P_\theta^{\pi_\phi})^{-1} b_\eta \ = \ \sum_{t=0}^{\infty} (\gamma P_\theta^{\pi_\phi})^t \, b_\eta.$$

Proof. Contraction in $\|\cdot\|_\infty$. Let $G_1, G_2 \in \mathcal{B}_\infty$ be two candidate sensitivity functions for $\nabla_\eta Q_\theta^{\pi_\phi}$. Since b_η does not depend on the input function, we have

$$\mathcal{C}_\eta(G_1) - \mathcal{C}_\eta(G_2) = \gamma\, P_\theta^{\pi_\phi}\, (G_1 - G_2).$$

The policy–induced Markov operator $P_\theta^{\pi_\phi}$ is an averaging operator, hence non-expansive in the sup–norm, i.e. $\|P_\theta^{\pi_\phi} f\|_\infty \le \|f\|_\infty$ for all $f \in \mathcal{B}_\infty$. Therefore

$$\|\mathcal{C}_\eta(G_1) - \mathcal{C}_\eta(G_2)\|_\infty = \gamma\, \|P_\theta^{\pi_\phi}(G_1 - G_2)\|_\infty \ \le \ \gamma\, \|G_1 - G_2\|_\infty.$$

Since $\gamma \in (0, 1)$, $\mathcal{C}_\eta$ is a contraction with modulus γ.

Because $(\mathcal{B}_\infty, \|\cdot\|_\infty)$ is complete, Banach's fixed–point theorem implies that $\mathcal{C}_\eta$ has a unique fixed point $G^\star \in \mathcal{B}_\infty$, and $G^{(k+1)} = \mathcal{C}_\eta(G^{(k)})$ converges to $G^\star$ for any start $G^{(0)}$ (Granas et al., 2003). By construction, $G^\star = \nabla_\eta Q_\theta^{\pi_\phi}$, the fixed–point identity is:

$$\nabla_\eta Q_\theta^{\pi_\phi} = (I - \gamma P_\theta^{\pi_\phi})^{-1} b_\eta = \sum_{t=0}^{\infty} (\gamma P_\theta^{\pi_\phi})^t b_\eta,$$

where the Neumann series converges because $\|\gamma P_\theta^{\pi_\phi}\| \le \gamma < 1$. □

Fast-layer mean field from three Bellman residuals. Let us represent the variables in the fast layer estimating critic and its sensitivities by $z = (w, u, v)$, where w estimates $Q_\theta^{\pi_\phi}$ and u, v estimate $\nabla_\phi Q_\theta^{\pi_\phi}$, $\nabla_\theta Q_\theta^{\pi_\phi}$. Their residuals with respect to the fixed points can be written as:

$$\begin{aligned} H_w(w; \phi, \theta) &:= R_\theta + \gamma P_\theta^{\pi_\phi} Q_\theta^{\pi_\phi} - Q_\theta^{\pi_\phi}, \\ H_u(u, w; \phi, \theta) &:= b_\phi(w; \phi, \theta) + \gamma P_\theta^{\pi_\phi} u - u, \\ H_v(v, w; \phi, \theta) &:= b_\theta(w; \phi, \theta) + \gamma P_\theta^{\pi_\phi} v - v, \end{aligned}$$

and the stacked mean field

$$H(z; \phi, \theta) \ := \ \big(H_w(w; \phi, \theta), \ H_u(u, w; \phi, \theta), \ H_v(v, w; \phi, \theta)\big).$$

Lemma 7 (Existence, stability, and Lipschitz dependence of $z^\star$)**.** *For each (ϕ, θ), consider the fast-layer ODE*

$$\dot{z} \ = \ H(z; \phi, \theta), \qquad z = (w, u, v),$$

with the stacked drift

$$H(z; \phi, \theta) := \big(\underbrace{R_\theta + \gamma P_\theta^{\pi_\phi} w - w}_{=: H_w(w)}, \ \underbrace{b_\phi(w; \phi, \theta) + \gamma P_\theta^{\pi_\phi} u - u}_{=: H_u(u, w)}, \ \underbrace{b_\theta(w; \phi, \theta) + \gamma P_\theta^{\pi_\phi} v - v}_{=: H_{U_\theta}(v, w)}\big).$$

Here $\gamma \in (0,1)$, $P_\theta^{\pi_\phi}$ is the policy-induced Markov operator, R_θ the reward function, and

$$b_\eta(w;\phi,\theta) := \nabla_\eta R_\theta \;+\; \gamma\,(\nabla_\eta P_\theta^{\pi_\phi})\,w, \qquad \eta \in \{\phi,\theta\}.$$

Assume: (i) $P_\theta^{\pi_\phi}$ is a contraction in the chosen norm ($\|P_\theta^{\pi_\phi} f\| \le \|f\|$), (ii) $\nabla_\eta P_\theta^{\pi_\phi}$ and $\nabla_\eta R_\theta$ are locally Lipschitz in (ϕ,θ), and (iii) all iterates/limits are restricted to a compact set. Then:

1. *Existence and uniqueness of $z^\star$: for each (ϕ,θ) there is a unique equilibrium $z^\star(\phi,\theta)$ with $H\big(z^\star(\phi,\theta);\phi,\theta\big) = 0$.*
2. *Dissipativity and stability: there exists $\mu > 0$ (independent of z on compacts) such that for all z,*

$$\big\langle H(z;\phi,\theta) - H(z^\star(\phi,\theta);\phi,\theta)\,,\; z - z^\star(\phi,\theta)\big\rangle \;\le\; -\,\mu\,\|z - z^\star(\phi,\theta)\|^2, \tag{81}$$

   where $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ are the inner product and norm induced by the chosen (Hilbert) metric.
3. *Lipschitz dependence: there exists a constant $L_{z^\star} < \infty$ such that for all $(\phi,\theta),(\tilde\phi,\tilde\theta)$ in a compact set,*

$$\big\|z^\star(\phi,\theta) - z^\star(\tilde\phi,\tilde\theta)\big\| \;\le\; L_{z^\star}\big(\|\phi-\tilde\phi\| + \|\theta-\tilde\theta\|\big).$$

Proof. For fixed (ϕ,θ), the maps

$$T(w) := R_\theta + \gamma P_\theta^{\pi_\phi} w, \qquad \mathcal{C}_\eta(u;w) := b_\eta(w;\phi,\theta) + \gamma P_\theta^{\pi_\phi} u$$

are γ-contractions in w and u respectively (Lemma 6), with contraction modulus $\gamma \in (0,1)$ and bounded affine terms on compacts.

(1) Existence and uniqueness of $z^\star$. The stacked system is *upper triangular* in (w,u,v):

- Since T is a γ-contraction, the Banach fixed-point theorem gives a unique $w^\star(\phi,\theta)$ such that $w^\star = T(w^\star)$.
- Given $w^\star$, the u-block satisfies $H_u(u,w^\star) = 0 \iff u = \mathcal{C}_\phi(u;w^\star)$, and $\mathcal{C}_\phi(\cdot;w^\star)$ is a γ-contraction in u. Thus there is a unique $u^\star(\phi,\theta) = (I - \gamma P_\theta^{\pi_\phi})^{-1} b_\phi(w^\star;\phi,\theta)$.
- Similarly, $v^\star(\phi,\theta) = (I - \gamma P_\theta^{\pi_\phi})^{-1} b_\theta(w^\star;\phi,\theta)$ is unique.

Hence $z^\star(\phi,\theta) := (w^\star,u^\star,v^\star)$ is the unique solution of $H\big(z;\phi,\theta\big) = 0$.

(2) Dissipativity and stability. Let $e_w := w - w^\star$, $e_u := u - u^\star$, $e_v := v - v^\star$, and $e := (e_w,e_u,e_v)$. We work in a Hilbert norm so that $\langle\cdot,\cdot\rangle$ is the associated inner product and Cauchy–Schwarz holds.

Since T is a γ-contraction,

$$\begin{aligned}\big\langle H_w(w) - H_w(w^\star),\, e_w\big\rangle &= \big\langle T(w) - T(w^\star) - (w - w^\star),\, e_w\big\rangle \\ &\le \|T(w) - T(w^\star)\|\,\|e_w\| - \|e_w\|^2 \le (\gamma - 1)\|e_w\|^2.\end{aligned}$$

Thus $\langle H_w(w) - H_w(w^\star), e_w\rangle \le -(1-\gamma)\|e_w\|^2$.

Using $H_u(u,w) = b_\phi(w) + \gamma P_\theta^{\pi_\phi} u - u$ and $H_u(u^\star,w^\star) = 0$,

$$H_u(u,w) - H_u(u^\star,w^\star) = \underbrace{\big(b_\phi(w) - b_\phi(w^\star)\big)}_{\text{depends on } e_w} + \gamma P_\theta^{\pi_\phi} e_u - e_u.$$

Now taking the inner product with e_u and using the non-expansiveness of $P_\theta^{\pi_\phi}$:

$$\langle H_u(u,w) - H_u(u^\star,w^\star),\, e_u\rangle \le (\gamma-1)\|e_u\|^2 + \|b_\phi(w) - b_\phi(w^\star)\|\,\|e_u\|.$$

By local Lipschitzness of $b_\phi(\cdot;\phi,\theta)$ in w on compacts: $\|b_\phi(w) - b_\phi(w^\star)\| \le L_\phi\|e_w\|$. Applying Young's inequality with any $\varepsilon_\phi > 0$ yields:

$$\|b_\phi(w) - b_\phi(w^\star)\|\,\|e_u\| \le \frac{\varepsilon_\phi}{2}\|e_u\|^2 + \frac{L_\phi^2}{2\varepsilon_\phi}\|e_w\|^2.$$

Hence

$$\langle H_u(u,w) - H_u(u^\star,w^\star),\, e_u\rangle \le -\big(1-\gamma-\tfrac{\varepsilon_\phi}{2}\big)\|e_u\|^2 \;+\; \frac{L_\phi^2}{2\varepsilon_\phi}\,\|e_w\|^2.$$

The same argument gives, for some L_θ and $\varepsilon_\theta > 0$,

$$\langle H_{U_\theta}(v,w) - H_{U_\theta}(v^\star,w^\star),\, e_v\rangle \le -\big(1-\gamma-\tfrac{\varepsilon_\theta}{2}\big)\|e_v\|^2 \;+\; \frac{L_\theta^2}{2\varepsilon_\theta}\,\|e_w\|^2.$$

Summing the bounds for H_w, H_u and H_v gives:

$$\big\langle H(z;\phi,\theta)-H(z^\star;\phi,\theta),\, e\big\rangle \le -(1-\gamma)\|e_w\|^2-\Big(1-\gamma-\tfrac{\varepsilon_\phi}{2}\Big)\|e_u\|^2-\Big(1-\gamma-\tfrac{\varepsilon_\theta}{2}\Big)\|e_v\|^2+C_w\,\|e_w\|^2,$$

with $C_w := \frac{L_\phi^2}{2\varepsilon_\phi} + \frac{L_\theta^2}{2\varepsilon_\theta}$. Choose $\varepsilon_\phi, \varepsilon_\theta > 0$ small enough and use compactness to ensure $C_w < (1-\gamma)/2$. Then there exists $\mu \in (0, 1-\gamma)$ such that

$$\big\langle H(z;\phi,\theta) - H(z^\star;\phi,\theta),\, z-z^\star\big\rangle \le -\,\mu\,\big(\|e_w\|^2 + \|e_u\|^2 + \|e_v\|^2\big) = -\,\mu\,\|z-z^\star\|^2.$$

This is (81). In particular, the ODE $\dot z = H(z;\phi,\theta)$ is globally (on the compact set) asymptotically stable at $z^\star$.

(3) Lipschitz dependence $(\phi,\theta) \mapsto z^\star(\phi,\theta)$. We use a fixed-point perturbation argument.

Critic $w^\star$. For (ϕ,θ) and $(\tilde\phi,\tilde\theta)$, let $T := R_\theta + \gamma P_\theta^{\pi_\phi}(\cdot)$ and $\tilde T := R_{\tilde\theta} + \gamma P_{\tilde\theta}^{\pi_{\tilde\phi}}(\cdot)$. Both are γ-contractions. A standard bound for fixed points of contractions gives

$$\|w^\star - \tilde w^\star\| \;\le\; \frac{1}{1-\gamma}\,\big\|T(\tilde w^\star) - \tilde T(\tilde w^\star)\big\| \;\le\; \frac{1}{1-\gamma}\,\Big(\|R_\theta - R_{\tilde\theta}\| + \gamma\,\|P_\theta^{\pi_\phi} - P_{\tilde\theta}^{\pi_{\tilde\phi}}\|\,\|\tilde w^\star\|\Big).$$

On a compact parameter set, $\|\tilde w^\star\|$ is bounded and $(\phi,\theta) \mapsto (R_\theta, P_\theta^{\pi_\phi})$ is locally Lipschitz; hence $\|w^\star - \tilde w^\star\| \le L_w(\|\phi-\tilde\phi\| + \|\theta-\tilde\theta\|)$.

Sensitivities $(u^\star, v^\star)$. With $u^\star = (I-\gamma P_\theta^{\pi_\phi})^{-1} b_\phi(w^\star;\phi,\theta)$, we have

$$\|u^\star - \tilde u^\star\| \le \big\|(I-\gamma P_\theta^{\pi_\phi})^{-1}\big\|\,\big\|b_\phi(w^\star;\phi,\theta) - b_\phi(\tilde w^\star;\tilde\phi,\tilde\theta)\big\| + \big\|(I-\gamma P_\theta^{\pi_\phi})^{-1} - (I-\gamma P_{\tilde\theta}^{\pi_{\tilde\phi}})^{-1}\big\|\,\|b_\phi(\tilde w^\star;\tilde\phi,\tilde\theta)\|.$$

The resolvent is uniformly bounded by $(1-\gamma)^{-1}$, and $(\phi,\theta,w) \mapsto b_\phi(w;\phi,\theta)$ is locally Lipschitz. The resolvent difference is locally Lipschitz in P on compacts. Using the bound for $\|w^\star - \tilde w^\star\|$ and Lipschitzness of $(\phi,\theta) \mapsto P_\theta^{\pi_\phi}$, we conclude $\|u^\star - \tilde u^\star\| \le L_u(\|\phi-\tilde\phi\| + \|\theta-\tilde\theta\|)$. An identical argument applies to $v^\star$.

Combining the three blocks yields the stated Lipschitz bound for $z^\star$. □

Smoothness and projected smoothness. On a compact Θ, if F is C^1 with L-Lipschitz gradient, then for any $x, y \in \Theta$, $F(y) \ge F(x) + \nabla F(x)^\top (y-x) - \frac{L}{2}\|y-x\|^2$ (Nesterov, 2013). If Π_Θ is the metric projection, then $F(\Pi_\Theta(x+u)) \ge F(x) + \nabla F(x)^\top u - \frac{L}{2}\|u\|^2$.

Robbins–Siegmund. We use the standard almost-supermartingale theorem: if $V_{k+1} \le (1-a_k)V_k + b_k - \xi_k$, with $a_k \in [0,1]$, $\sum a_k = \infty$, $\sum b_k < \infty$, $V_k \ge 0$, then V_k converges and $\sum a_k V_k < \infty$ (Borkar and Borkar, 2008).

B.1 ODE Approach

Stochastic-approximation (SA) viewpoint Each update in Algorithm 1 is a noisy Euler step for a mean-field drift; with projections keeping iterates bounded. Concretely, with stepsizes (γ_k) for the fast layer $z = (w,u,v)$, (β_k) for the medium layer ϕ, and (α_k) for the slow layer θ, the projected recursions implement

$$\text{(fast)} \quad z_{k+1} = \Pi_{\mathcal{Z}}\big(z_k + \gamma_k[\,H(z_k;\phi_k,\theta_k) + M_{k+1}^{(z)}\,]\big), \tag{82}$$

$$\text{(medium)} \quad \phi_{k+1} = \Pi_\Phi\big(\phi_k + \beta_k[\,\hat\varphi(\phi_k,\theta_k;z_k) + M_{k+1}^{(\phi)}\,]\big), \tag{83}$$

$$\text{(slow)} \quad \theta_{k+1} = \Pi_\Theta\big(\theta_k + \alpha_k[\,\varphi(\phi_k,\theta_k;z_k) + M_{k+1}^{(\theta)}\,]\big). \tag{84}$$

Outer-level gradient. Define

$$F(\theta) := J_{\text{real}}\big(\pi_{\phi^\star(\theta)}\big), \qquad \nabla_\theta F(\theta) = \varphi\big(\phi^\star(\theta), \theta;\, z(\phi^\star(\theta), \theta)\big). \tag{85}$$

Assumptions (Three Timescales). Let $\mathcal{Z}, \Phi, \Theta$ be compact convex; all norms are Euclidean/operator.

1. Regularity/IFT for inner best response. $\phi^\star(\theta)$ is locally unique with nonsingular $\nabla_\phi \hat{\varphi}(\phi^\star(\theta), \theta)$; thus $\phi^\star \in C^1$.
2. Smoothness / finite moments. Rewards are bounded; $\log \pi_\phi, \mathcal{P}_\theta, R_\theta$ are C^1 (or C^2 as used), locally Lipschitz on $\Phi \times \Theta$. On-policy Markov chains (sim and real) are geometrically ergodic; all MC estimators have bounded second moments.
3. Fast-layer contraction & Lipschitz. Lemma 6 and Lemma 7 hold (for critic *and* sensitivities, with projection if using linear FA).
4. Inner-layer stability. For each fixed θ, there exists $m > 0$ s.t. for ϕ near $\phi^\star(\theta)$, $\langle \hat{\varphi}(\phi, \theta) - \hat{\varphi}(\phi^\star, \theta),\, \phi - \phi^\star \rangle \le -m\|\phi - \phi^\star\|^2$.
5. Stepsizes and separation. Let γ, β and α) represents the step sizes for fast, medium and slow ODEs. $\sum_k \gamma_k = \sum_k \beta_k = \sum_k \alpha_k = \infty$, $\sum_k \gamma_k^2, \sum_k \beta_k^2, \sum_k \alpha_k^2 < \infty$, and $\beta_k/\gamma_k \to 0$, $\alpha_k/\beta_k \to 0$.
6. Noise. $M_k^{(z)}, M_k^{(\phi)}, M_k^{(\theta)}$ are square-integrable martingale differences with uniformly bounded conditional second moments.
7. Projection. Updates are projected onto $\mathcal{Z}, \Phi, \Theta$; projections are nonexpansive.

Under **A1**–**A7** (regularity, bounded moments, contraction/stability, stepsize separation, projections), the corresponding mean-field ODEs are

$$\dot{z} = H(z; \phi, \theta), \qquad \dot{\phi} = \hat{\varphi}(\phi, \theta; z), \qquad \dot{\theta} = \varphi(\phi, \theta; z).$$

Steps that follows To establish this three timestep convergence, we need to establish:

1. *Fast equilibrium is well-posed, stable, and smooth in* (ϕ, θ). This is provided by Lemma 7 using affine γ-contraction of Bellman/Poisson operators for the critic and its sensitivities.
2. *Fast tracking with a moving target.* Despite the slow motion of (ϕ_k, θ_k), the error e_k^z contracts up to a small *moving-target* term $\Delta_{k+1}^z := z^\star(\phi_{k+1}, \theta_{k+1}) - z^\star(\phi_k, \theta_k)$, which is square-summable because $\|\Delta_{k+1}^z\| = O(\beta_k) + O(\alpha_k)$ and $\sum_k (\beta_k^2 + \alpha_k^2) < \infty$. This is provided by Lemma 8.
3. *Medium tracking with a moving target and a fast-layer perturbation.* Define the reduced inner drift $h^\star$ and note $\hat{\varphi}(\phi_k, \theta_k; z_k) = h^\star(\phi_k, \theta_k) + \delta_k^z$ with $\delta_k^z := \hat{\varphi}(\phi_k, \theta_k; z_k) - \hat{\varphi}(\phi_k, \theta_k; z^\star(\phi_k, \theta_k))$. The perturbation obeys $\|\delta_k^z\| \le L\|e_k^z\|$ by Lipschitzness, while the moving target $\Delta_{k+1}^\phi := \phi^\star(\theta_{k+1}) - \phi^\star(\theta_k)$ is $O(\alpha_k)$. A Lyapunov (quadratic) drift argument with Young's inequality and **A4** (local strong stability) proves Lemma 9.
4. *Slow layer as SGD with summable bias.* Write the slow estimator as

$$\widehat{g}_k = \varphi(\phi_k, \theta_k; z_k) + M_{k+1}^{(\theta)} = g(\theta_k) + b_k + M_{k+1}^{(\theta)},$$

where $g(\theta) := \varphi\big(\phi^\star(\theta), \theta;\, z(\phi^\star(\theta), \theta)\big)$ and the bias b_k satisfies $\|b_k\| \le C(\|e_k^\phi\| + \|e_k^z\|)$. Timescale separation gives $\sum_k \alpha_k \|b_k\|^2 < \infty$ (Lemma 10). The standard smooth-SGD inequality with projection and Robbins–Siegmund then yield convergence to the stationary set $\{\nabla_\theta F = 0\}$.

Error notations used in the lemmas. We introduce the tracking errors and the "moving-target" increments for the fast and medium layers:

$$e_k^z := z_k - z_k^\star, \qquad \Delta_{k+1}^z := z_{k+1}^\star - z_k^\star, \tag{86}$$

$$e_k^\phi := \phi_k - \phi^\star(\theta_k), \qquad \Delta_{k+1}^\phi := \phi^\star(\theta_{k+1}) - \phi^\star(\theta_k). \tag{87}$$

The * notation represents the target at an iterate k. The lemmas quantify how e_k^z and e_k^ϕ contract up to the small (square-summable) perturbations $\Delta_{k+1}^z, \Delta_{k+1}^\phi$ and the martingale noises, and how this yields a summable bias for the slow layer.

B.2 Lemmas and Main Theorem

Lemma 8 (Fast tracking of critic and sensitivities). *Work in the D_μ–Hilbert norm $\|f\|_{D_\mu}^2 = \sum_{s,a} \mu(s,a) f(s,a)^2$ (or its block product for $z = (w,u,v)$), with inner product $\langle \cdot, \cdot \rangle_{D_\mu}$. Under **A2**–**A7** and Lemma 7, for the fast recursion*

$$z_{k+1} = \Pi_{\mathcal{Z}}\Big(z_k + \gamma_k \big[\, H(z_k; \phi_k, \theta_k) + M_{k+1}^{(z)} \big]\Big), \quad e_k^z := z_k - z_k^\star,$$

we have $e_k^z \to 0$ almost surely. In particular, $\sum_{k=0}^\infty \gamma_k \, \mathbb{E}\|e_k^z\|_{D_\mu}^2 < \infty$ and $\|e_k^z\|_{D_\mu} \to 0$ a.s.

Proof. Fix k and abbreviate $z_k^\star := z(\phi_k, \theta_k)$, $H_k := H(z_k; \phi_k, \theta_k)$, $D_k := H_k - H_k^\star = H(z_k; \phi_k, \theta_k) - H(z_k^\star; \phi_k, \theta_k)$, and $\Delta_{k+1}^z := z_{k+1}^\star - z_k^\star$.

From Lemma 7 (proved via Bellman/Poisson contractions):

$$\text{(dissipativity)} \qquad \langle D_k,\, e_k^z \rangle_{D_\mu} \;\le\; -\,\mu\, \|e_k^z\|_{D_\mu}^2, \tag{88}$$

$$\text{(Lipschitz in } z\text{)} \qquad \|D_k\|_{D_\mu} \;\le\; L_H\, \|e_k^z\|_{D_\mu}, \tag{89}$$

for some $\mu > 0$ and $L_H < \infty$, uniform on the compact set where iterates live.

By nonexpansiveness of $\Pi_{\mathcal{Z}}$, adding and subtracting $H_k^\star$ and using $H_k^\star = 0$ we can write,

$$\begin{aligned} \|e_{k+1}^z\|_{D_\mu} &= \big\|\Pi_{\mathcal{Z}}\big(z_k + \gamma_k(H_k + M_{k+1}^{(z)})\big) - z_{k+1}^\star\big\|_{D_\mu} \\ &\le \big\|z_k + \gamma_k(H_k + M_{k+1}^{(z)}) - z_{k+1}^\star\big\|_{D_\mu} \\ &= \big\|e_k^z + \gamma_k D_k + \gamma_k M_{k+1}^{(z)} - \Delta_{k+1}^z\big\|_{D_\mu}. \end{aligned} \tag{90}$$

Now by the martingale-difference property (Assumption **A6**),

$$\mathbb{E}\langle e_k^z + \gamma_k D_k - \Delta_{k+1}^z,\; M_{k+1}^{(z)} \mid \mathcal{F}_k \rangle_{D_\mu} = 0, \quad \mathbb{E}\|M_{k+1}^{(z)}\|_{D_\mu}^2 \mid \mathcal{F}_k \le \sigma_z^2.$$

Using this to take the conditional expectation $\mathbb{E}[\cdot \mid \mathcal{F}_k]$ of the square of (90) gives:

$$\mathbb{E}\Big[\|e_{k+1}^z\|_{D_\mu}^2 \mid \mathcal{F}_k\Big] \le \|e_k^z + \gamma_k D_k - \Delta_{k+1}^z\|_{D_\mu}^2 + \gamma_k^2 \sigma_z^2. \tag{91}$$

According to Young's inequality, for any $\eta > 0$ and any vectors x, y, $\|x-y\|^2 \le (1+\eta)\|x\|^2 + (1+1/\eta)\|y\|^2$. Apply with $x = e_k^z + \gamma_k D_k$ and $y = \Delta_{k+1}^z$, then take $\mathbb{E}[\cdot \mid \mathcal{F}_k]$:

$$\mathbb{E}\Big[\|e_{k+1}^z\|_{D_\mu}^2 \mid \mathcal{F}_k\Big] \le (1+\eta)\, \|e_k^z + \gamma_k D_k\|_{D_\mu}^2 \;+\; \Big(1 + \tfrac{1}{\eta}\Big)\, \mathbb{E}\Big[\|\Delta_{k+1}^z\|_{D_\mu}^2 \mid \mathcal{F}_k\Big] \;+\; \gamma_k^2 \sigma_z^2. \tag{92}$$

Now let us consider only the term $\|e_k^z + \gamma_k D_k\|^2$. By dissipativity and Lipschitzness of H in z as shown in (88) and (89), we get:

$$\begin{aligned} \|e_k^z + \gamma_k D_k\|_{D_\mu}^2 &= \|e_k^z\|_{D_\mu}^2 + 2\gamma_k \langle e_k^z, D_k \rangle_{D_\mu} + \gamma_k^2 \|D_k\|_{D_\mu}^2 \\ &\le \big(1 - 2\mu\gamma_k + L_H^2 \gamma_k^2\big)\, \|e_k^z\|_{D_\mu}^2 \;\le\; \big(1 - 2\mu\gamma_k + C\gamma_k^2\big)\, \|e_k^z\|_{D_\mu}^2. \end{aligned} \tag{93}$$

By the Lipschitz dependence of $z(\cdot,\cdot)$ (Lemma 7) and projected bounded steps (Assumptions **A2**, **A7**),

$$\|\Delta_{k+1}^z\|_{D_\mu} = \|z^\star(\phi_{k+1}, \theta_{k+1}) - z^\star(\phi_k, \theta_k)\|_{D_\mu} \le L_{z^\star}\big(\|\phi_{k+1} - \phi_k\| + \|\theta_{k+1} - \theta_k\|\big).$$

Taking $\mathbb{E}[\cdot \mid \mathcal{F}_k]$ and using $\mathbb{E}\|\phi_{k+1} - \phi_k\|^2 \le C\beta_k^2$ and $\mathbb{E}\|\theta_{k+1} - \theta_k\|^2 \le C\alpha_k^2$,

$$\mathbb{E}\Big[\|\Delta_{k+1}^z\|_{D_\mu}^2 \mid \mathcal{F}_k\Big] \;\le\; C\big(\beta_k^2 + \alpha_k^2\big). \tag{94}$$

Plug (93) and (94) into (92), and choose $\eta = \mu\gamma_k$ (small for large k):

$$\mathbb{E}\Big[\|e^z_{k+1}\|^2_{D_\mu} \mid \mathcal{F}_k\Big] \le (1+\mu\gamma_k)\big(1-2\mu\gamma_k+C\gamma_k^2\big)\,\|e^z_k\|^2_{D_\mu} \;+\; \Big(1+\tfrac{1}{\mu\gamma_k}\Big)\,C(\beta_k^2+\alpha_k^2) \;+\; \gamma_k^2\sigma_z^2$$
$$= \big(1-\mu\gamma_k+C\gamma_k^2\big)\,\|e^z_k\|^2_{D_\mu} \;+\; C(\beta_k^2+\alpha_k^2) + \tfrac{C}{\mu}\tfrac{\beta_k^2+\alpha_k^2}{\gamma_k} \;+\; \gamma_k^2\sigma_z^2,$$

where we expanded $(1+\mu\gamma_k)(1-2\mu\gamma_k+C\gamma_k^2)$ and absorbed higher-order terms into $C\gamma_k^2$.

$$(1+\mu\gamma_k)\big(1-2\mu\gamma_k+C\gamma_k^2\big) = 1 \;-\; \mu\gamma_k \;+\; (C-2\mu^2)\gamma_k^2 \;+\; C\mu\,\gamma_k^3 \;\le\; 1 \;-\; \mu\gamma_k \;+\; C\gamma_k^2,$$

for a (possibly larger) constant $C < \infty$ (since $\gamma_k^3 \le \gamma_k^2$ for all k large enough and we can absorb finitely many initial indices into C). Thus there exists $c \in (0, \mu]$ such that

$$(1+\mu\gamma_k)\big(1-2\mu\gamma_k+C\gamma_k^2\big) \;\le\; 1-c\gamma_k+C\gamma_k^2. \tag{95}$$

Keep the squared step-size term as is and bound the factor:

$$\Big(1+\tfrac{1}{\mu\gamma_k}\Big)\,C(\beta_k^2+\alpha_k^2) \;\le\; C(\beta_k^2+\alpha_k^2) \;+\; \frac{C}{\mu}\,\frac{\beta_k^2+\alpha_k^2}{\gamma_k}.$$

Under the timescale separation $\beta_k/\gamma_k \to 0$ and $\alpha_k/\gamma_k \to 0$, we have $(\beta_k^2+\alpha_k^2)/\gamma_k = o(\gamma_k)$; hence, for all sufficiently large k,

$$\frac{\beta_k^2+\alpha_k^2}{\gamma_k} \;\le\; C_1\,\gamma_k \;\le\; C_2\,\gamma_k^2 \;+\; C_3\,(\beta_k^2+\alpha_k^2),$$

where the last inequality holds after (i) possibly enlarging constants to account for a finite prefix of indices, and (ii) using that $\beta_k^2+\alpha_k^2 = O(\gamma_k^2)$ under separation. Absorbing constants, we obtain the per–step bound

$$\Big(1+\tfrac{1}{\mu\gamma_k}\Big)\,C(\beta_k^2+\alpha_k^2) \;\le\; C\,\gamma_k^2 \;+\; C\,(\beta_k^2+\alpha_k^2). \tag{96}$$

Now combining (91), (94), and the Young bound, and taking $\varepsilon > 0$ small so that $1-2\mu\gamma_k \mapsto 1-c\gamma_k$ for some $c > 0$ (absorbing constants), we obtain

$$\mathbb{E}\|e^z_{k+1}\|^2_{D_\mu} \;\le\; \big(1-c\gamma_k+C\gamma_k^2\big)\,\mathbb{E}\|e^z_k\|^2_{D_\mu} \;+\; C\gamma_k^2 \;+\; C(\beta_k^2+\alpha_k^2). \tag{97}$$

To apply Robbins–Siegmund to (97) we set

$$V_k := \mathbb{E}\|e^z_k\|^2_{D_\mu}, \quad a_k := c\gamma_k - C\gamma_k^2, \quad b_k := C\gamma_k^2 + C(\beta_k^2+\alpha_k^2),$$

so that

$$V_{k+1} \;\le\; (1-a_k)V_k + b_k.$$

Assumption **A5** (namely, $\sum_k \gamma_k = \infty$ and $\sum_k(\gamma_k^2+\alpha_k^2+\beta_k^2) < \infty$) implies $\sum_k a_k = \infty$ and $\sum_k b_k < \infty$, with $a_k \in [0,1]$ for large k. Robbins–Siegmund then yields that V_k converges and $\sum_k a_k V_k < \infty$.

Since

$$a_k \;=\; c\gamma_k - C\gamma_k^2 \;=\; c\gamma_k\Big(1-\tfrac{C}{c}\gamma_k\Big),$$

and by Assumption **A5** we have $\gamma_k \to 0$, there exists K such that $\gamma_k \le \frac{c}{2C}$ for all $k \ge K$. Thus, for $k \ge K$,

$$1-\tfrac{C}{c}\gamma_k \;\ge\; \tfrac{1}{2}, \qquad \text{so} \qquad a_k \;\ge\; \tfrac{c}{2}\gamma_k.$$

Since $a_k \ge \frac{c}{2}\gamma_k$ for all sufficiently large k and $\sum_{k=0}^{\infty}\gamma_k = \infty$, it follows that

$$\sum_{k=0}^{\infty} a_k = \infty.$$

Suppose, for contradiction, that $\lim_{k\to\infty} V_k = L > 0$. Then there exists K such that $V_k \ge \frac{L}{2}$ for all $k \ge K$. Hence

$$\sum_{k=0}^{\infty} a_k V_k \;\ge\; \sum_{k=K}^{\infty} a_k\,\tfrac{L}{2} \;=\; \tfrac{L}{2}\sum_{k=K}^{\infty} a_k \;=\; \infty,$$

which contradicts the Robbins–Siegmund conclusion that $\sum_k a_k V_k < \infty$. Therefore $L = 0$, i.e. $V_k \to 0$. Finally, the standard martingale argument (Doob + Borel–Cantelli) implies $\|e_k^z\|_{D_\mu} \to 0$ almost surely (Duflo, 2013).

□

Lemma 9 (Medium tracking of the inner-level (in-sim)). *Under **A1**, **A2**, **A4**–**A7**, and Lemma 8, we get:*

$$\sum_{k=0}^{\infty} \beta_k \, \mathbb{E}\|e_k^\phi\|^2 < \infty \quad \Longrightarrow \quad e_k^\phi \to 0 \quad a.s.$$

Proof. Work in the Euclidean norm on Φ. Let

$$e_k^\phi := \phi_k - \phi^\star(\theta_k), \quad h(\phi, \theta) := \hat{\varphi}(\phi, \theta; z(\phi, \theta)).$$

Define

$$D_k^\phi := h(\phi_k, \theta_k) - h^\star(\phi^\star(\theta_k), \theta_k), \qquad \delta_k^z := \hat{\varphi}(\phi_k, \theta_k; z_k) - \hat{\varphi}(\phi_k, \theta_k; z_k^\star), \qquad \Delta_{k+1}^\phi := \phi^\star(\theta_{k+1}) - \phi^\star(\theta_k).$$

From **A4** (inner-layer stability), there exists $m > 0$ such that, for k large enough (iterates in a compact neighborhood by projection **A7**),

$$\langle e_k^\phi, \, D_k^\phi \rangle \le -\, m \, \|e_k^\phi\|^2, \qquad \|D_k^\phi\| \le L_h \|e_k^\phi\| \quad \text{(local Lipschitzness from \textbf{A2}).} \tag{98}$$

By Lipschitzness of $\hat{\varphi}$ in z on compacts (**A2**),

$$\|\delta_k^z\| \le L_z \|z_k - z_k^\star\| = L_z \|e_k^z\|. \tag{99}$$

By **A1** (IFT) $\phi^\star \in C^1$, thus Lipschitz on compact Θ:

$$\|\Delta_{k+1}^\phi\| \le L_{\phi^\star} \, \|\theta_{k+1} - \theta_k\|. \tag{100}$$

Using nonexpansiveness of Π_Φ and $\phi^\star(\theta_{k+1}) \in \Phi$,

$$\begin{aligned} \|e_{k+1}^\phi\| &= \left\| \Pi_\Phi\big(\phi_k + \beta_k[\hat{\varphi}(\phi_k, \theta_k; z_k) + M_{k+1}^{(\phi)}]\big) - \phi^\star(\theta_{k+1}) \right\| \\ &\le \left\| \phi_k + \beta_k \big(h^\star(\phi_k, \theta_k) + \delta_k^z + M_{k+1}^{(\phi)}\big) - \phi^\star(\theta_{k+1}) \right\| \\ &= \left\| e_k^\phi + \beta_k D_k^\phi + \beta_k \delta_k^z + \beta_k M_{k+1}^{(\phi)} - \Delta_{k+1}^\phi \right\|. \end{aligned} \tag{101}$$

Squaring (101) and taking $\mathbb{E}[\cdot \mid \mathcal{F}_k]$, the martingale-difference property (**A6**) gives

$$\mathbb{E}\langle e_k^\phi + \beta_k D_k^\phi + \beta_k \delta_k^z - \Delta_{k+1}^\phi, \; M_{k+1}^{(\phi)} \mid \mathcal{F}_k \rangle = 0, \quad \mathbb{E}\|M_{k+1}^{(\phi)}\|^2 \mid \mathcal{F}_k \le \sigma_\phi^2.$$

Hence

$$\mathbb{E}\Big[\|e_{k+1}^\phi\|^2 \mid \mathcal{F}_k\Big] \le \|e_k^\phi + \beta_k D_k^\phi - \Delta_{k+1}^\phi\|^2 \; + \; \beta_k^2 \, \mathbb{E}\|\delta_k^z\|^2 \; + \; \beta_k^2 \, \sigma_\phi^2. \tag{102}$$

Expanding and applying (98):

$$\|e_k^\phi + \beta_k D_k^\phi\|^2 = \|e_k^\phi\|^2 + 2\beta_k \langle e_k^\phi, D_k^\phi \rangle + \beta_k^2 \|D_k^\phi\|^2 \le \big(1 - 2m\beta_k + L_h^2 \beta_k^2\big) \, \|e_k^\phi\|^2. \tag{103}$$

Using Young's inequality as in the fast-layer proof (Step 4 there), for any $\varepsilon \in (0, 1)$,

$$-2\langle e_k^\phi + \beta_k D_k^\phi, \; \Delta_{k+1}^\phi \rangle \; \le \; \varepsilon \, \|e_k^\phi + \beta_k D_k^\phi\|^2 \; + \; \frac{1}{\varepsilon} \, \|\Delta_{k+1}^\phi\|^2.$$

By (100) and the projected slow step (**A7**) with bounded second moments (**A2**),

$$\mathbb{E}\|\Delta_{k+1}^\phi\|^2 \le L_{\phi^\star}^2 \, \mathbb{E}\|\theta_{k+1} - \theta_k\|^2 \le C \, \alpha_k^2. \tag{104}$$

By (99) and boundedness of $\|e_k^z\|$ on $\mathcal{Z}$ (projection),

$$\mathbb{E}\|\delta_k^z\|^2 \le C \, \mathbb{E}\|e_k^z\|^2 \le C. \tag{105}$$

Plugging (103) and the Young bound into (102), and choosing $\varepsilon > 0$ small, we obtain

$$\mathbb{E}\Big[\|e_{k+1}^\phi\|^2 \mid \mathcal{F}_k\Big] \; \le \; \big(1 - c\,\beta_k + C\beta_k^2\big) \, \|e_k^\phi\|^2 \; + \; C \, \beta_k^2 \, \mathbb{E}\|\delta_k^z\|^2 \; + \; C \, \mathbb{E}\|\Delta_{k+1}^\phi\|^2 \; + \; C \, \beta_k^2,$$

for some $c \in (0,1)$ and finite C (uniform on the compact set). Now apply (105) and (104):

$$\mathbb{E}\|e_{k+1}^{\phi}\|^2 \;\le\; (1 - c\,\beta_k + C\beta_k^2)\,\mathbb{E}\|e_k^{\phi}\|^2 \;+\; C\,\beta_k^2 \;+\; C\,\alpha_k^2 \;+\; C\,\beta_k^2\,\mathbb{E}\|e_k^{z}\|^2.$$

Let $V_k := \mathbb{E}\|e_k^{\phi}\|^2$, $a_k := c\beta_k - C\beta_k^2$, and $b_k := C\beta_k^2(1+\mathbb{E}\|e_k^z\|^2) + C\alpha_k^2$. By **A5**, $\sum_k a_k = \infty$ and $\sum_k b_k < \infty$ (since $\sum \beta_k^2, \sum \alpha_k^2 < \infty$ and $\|e_k^z\|$ is bounded by projection). Robbins–Siegmund implies V_k converges and $\sum_k a_k V_k < \infty$. As $a_k \ge \frac{c}{2}\beta_k$ for large k and $\sum_k \beta_k = \infty$, we get $\lim_{k\to\infty} V_k = 0$.

Finally, the assumption $\sum_k \beta_k\,\mathbb{E}\|e_k^{\phi}\|^2 < \infty$ together with the boundedness of $\{e_k^{\phi}\}$ standardly yields $e_k^{\phi} \to 0$ almost surely (see the same martingale/Borel–Cantelli argument used after Lemma 8). □

Lemma 10 (Convergence of outer-level with slow timescale)**.** *Define the drift term of the slow ODE*

$$g(\theta) \;:=\; \varphi\big(\phi^\star(\theta), \theta;\, z(\phi^\star(\theta), \theta)\big) \;=\; \nabla_\theta F(\theta),$$

and the actual estimator

$$\widehat{g}_k \;:=\; \varphi(\phi_k, \theta_k; z_k) \;+\; M_{k+1}^{(\theta)} \;=\; g(\theta_k) \;+\; b_k \;+\; M_{k+1}^{(\theta)},$$

with the bias

$$b_k \;:=\; \varphi(\phi_k, \theta_k; z_k) - \varphi(\phi^\star(\theta_k), \theta_k; z(\phi^\star(\theta_k), \theta_k)).$$

*Under **A1**–**A7** and Lemmas 8–9:*

1. $\|b_k\| \le C\big(\|e_k^{\phi}\| + \|e_k^{z}\|\big)$ *and hence* $\displaystyle\sum_k \alpha_k \|b_k\|^2 < \infty$ *a.s.;*
2. *the projected slow update* $\theta_{k+1} = \Pi_\Theta\big(\theta_k + \alpha_k \widehat{g}_k\big)$ *satisfies*

$$F(\theta_k) \text{ converges a.s.}, \qquad \sum_{k=0}^{\infty} \alpha_k\, \|\nabla_\theta F(\theta_k)\|^2 < \infty \text{ a.s.},$$

and every limit point of $\{\theta_k\}$ *lies in* $\mathcal{E} := \{\theta : \nabla_\theta F(\theta) = 0\}$.

Proof. We work in the Euclidean norm on Θ. On compact Θ, ∇F is L-Lipschitz (by **A1**–**A2**). $M_{k+1}^{(\theta)}$ is a martingale difference with $\mathbb{E}[\|M_{k+1}^{(\theta)}\|^2 \mid \mathcal{F}_k] \le \sigma_\theta^2$ (**A6**).

Local Lipschitzness of φ in (ϕ, z) on the compact $\Phi \times \Theta \times \mathcal{Z}$ (**A2**) gives

$$\|b_k\| \;\le\; L_\varphi\Big(\|\phi_k - \phi^\star(\theta_k)\| + \|z_k - z(\phi^\star(\theta_k), \theta_k)\|\Big) \;\le\; C\big(\|e_k^{\phi}\| + \|e_k^{z}\|\big).$$

Therefore $\|b_k\|^2 \le C\big(\|e_k^{\phi}\|^2 + \|e_k^{z}\|^2\big)$.

From Lemma 9: $\sum_k \beta_k\,\mathbb{E}\|e_k^{\phi}\|^2 < \infty$ a.s. From Lemma 8: $\sum_k \gamma_k\,\mathbb{E}\|e_k^{z}\|^2 < \infty$ a.s. Since $\alpha_k/\beta_k \to 0$ and $\alpha_k/\gamma_k \to 0$ (**A5**), there exist c_ϕ, c_z and K s.t. for all $k \ge K$, $\alpha_k \le c_\phi \beta_k$ and $\alpha_k \le c_z \gamma_k$. Therefore, it proves (i).

$$\sum_{k=0}^{\infty} \alpha_k\,\mathbb{E}\|b_k\|^2 \;\le\; C\sum_k \alpha_k\big(\mathbb{E}\|e_k^{\phi}\|^2 + \mathbb{E}\|e_k^{z}\|^2\big) \;<\; \infty \quad \text{a.s.}$$

By projected smoothness (same inequality as used in Lemma 9):

$$F(\theta_{k+1}) \;\ge\; F(\theta_k) + \alpha_k\,\langle \nabla F(\theta_k),\, \widehat{g}_k\rangle \;-\; \tfrac{L}{2}\,\alpha_k^2 \|\widehat{g}_k\|^2.$$

Take $\mathbb{E}[\cdot \mid \mathcal{F}_k]$ and use $\mathbb{E}[M_{k+1}^{(\theta)} \mid \mathcal{F}_k] = 0$:

$$\mathbb{E}[F(\theta_{k+1}) \mid \mathcal{F}_k] \;\ge\; F(\theta_k) + \alpha_k\,\langle \nabla F(\theta_k),\, g(\theta_k)\rangle + \alpha_k\,\langle \nabla F(\theta_k),\, b_k\rangle - \tfrac{L}{2}\alpha_k^2\,\mathbb{E}\|g(\theta_k) + b_k + M_{k+1}^{(\theta)}\|^2.$$

But $g(\theta_k) = \nabla F(\theta_k)$, hence the main progress term is $\alpha_k \|\nabla F(\theta_k)\|^2$. Also

$$\mathbb{E}\|g(\theta_k) + b_k + M_{k+1}^{(\theta)}\|^2 \le C\big(\|\nabla F(\theta_k)\|^2 + \|b_k\|^2 + \sigma_\theta^2\big)\,.$$

By Young's inequality (same as in the medium-layer proof), for $\eta = \frac{1}{2}$,

$$\alpha_k \langle \nabla F(\theta_k), b_k \rangle \;\geq\; -\frac{\alpha_k}{4} \|\nabla F(\theta_k)\|^2 \;-\; \alpha_k \|b_k\|^2.$$

Absorb $C\alpha_k^2 \|\nabla F(\theta_k)\|^2$ into the main term for large k (since $\alpha_k \to 0$). Thus there exist k_0 and $c \in (0,1)$ such that for all $k \geq k_0$,

$$\mathbb{E}[F(\theta_{k+1}) \mid \mathcal{F}_k] \;\geq\; F(\theta_k) \;+\; c\,\alpha_k \|\nabla F(\theta_k)\|^2 \;-\; \alpha_k \|b_k\|^2 \;-\; C\,\alpha_k^2.$$

Define the almost-supermartingale

$$X_k \;:=\; F(\theta_k) - \sum_{i=0}^{k-1} \big(\alpha_i \|b_i\|^2 + C\alpha_i^2\big).$$

Since $\sum_k \alpha_k \|b_k\|^2 < \infty$ a.s., and $\sum_k \alpha_k^2 < \infty$ (**A5**), so $\sum_k (\alpha_k \|b_k\|^2 + C\alpha_k^2) < \infty$ a.s. Hence

$$\mathbb{E}[X_{k+1} \mid \mathcal{F}_k] \;\geq\; X_k \;+\; c\,\alpha_k \|\nabla F(\theta_k)\|^2.$$

Robbins–Siegmund implies: (i) X_k converges a.s., so $F(\theta_k)$ converges a.s.; (ii) $\sum_k \alpha_k \|\nabla F(\theta_k)\|^2 < \infty$ a.s. Therefore $\liminf_k \|\nabla F(\theta_k)\| = 0$, and any limit point $\theta^\star$ satisfies $\nabla F(\theta^\star) = 0$ by continuity of ∇F. This proves (ii). □

Theorem 2 (Three–timescale bi-level SPG: direct almost-sure convergence)**.** *Assume **A1**–**A7** and let the iterates (z_k, ϕ_k, θ_k) be generated by the three–timescale recursions in (82)–(84). Define the fast- and medium-layer errors*

$$e_k^z \;:=\; z_k - z^\star(\phi_k, \theta_k), \qquad e_k^\phi \;:=\; \phi_k - \phi^\star(\theta_k),$$

and the slow-layer stationary set

$$\mathcal{E} \;:=\; \big\{\theta \in \Theta :\; \nabla_\theta F(\theta) = 0\big\}, \qquad F(\theta) = J_{\text{real}}\big(\pi_{\phi^\star(\theta)}\big).$$

Then, almost surely,

$$e_k^z \longrightarrow 0, \qquad e_k^\phi \longrightarrow 0, \qquad \operatorname{dist}\big(\theta_k, \mathcal{E}\big) \longrightarrow 0.$$

Equivalently, every limit point of the triple (z_k, ϕ_k, θ_k) belongs to

$$\Big\{\big(z^\star(\phi^\star(\theta), \theta),\; \phi^\star(\theta),\; \theta\big) \;:\; \theta \in \mathcal{E}\Big\}.$$

Proof. By Lemma 7, the fast ODE has a unique, globally asymptotically stable equilibrium $z^\star(\phi, \theta)$; Lemma 8 implies $e_k^z \to 0$ a.s. Using this and the inner-layer stability in **A4**, Lemma 9 yields $e_k^\phi \to 0$ a.s. Finally, with $z_k \approx z^\star(\phi_k, \theta_k)$ and $\phi_k \approx \phi^\star(\theta_k)$, the slow drift equals $\nabla_\theta F(\theta_k)$ up to a summable bias, and Lemma 10 gives $\operatorname{dist}(\theta_k, \mathcal{E}) \to 0$ a.s. □

Corollary 1 (Two timescales (critic exact or faster))**.** *If $z_k \equiv z_k^\star$ (exact) or the fast layer is strictly faster so that $e_k^z \to 0$ with summable bias, Lemma 8 is obvious and Theorem 2 follows from Lemmas 9–10.*

Corollary 2 (Nested (inner exact at each outer step))**.** *If $z_k = z(\phi^\star(\theta_k), \theta_k)$ and $\phi_k = \phi^\star(\theta_k)$, then $\theta_{k+1} = \Pi_\Theta(\theta_k + \alpha_k[\nabla_\theta F(\theta_k) + M_{k+1}^{(\theta)}])$, and the standard single-level SPG convergence to $\mathcal{E}$ follows from Lemma 10.*

C Efficient Adjoint Computation of the Outer Gradient

This appendix provides an equivalent but more computationally efficient way to compute the outer gradient in the bi-level RL formulation of Section 5 (cf. Eq. (5)–Eq. (7a)). The key idea is to avoid forming the full Jacobian $\nabla_\theta \phi(\theta)$ (or any dense $d_\phi \times d_\theta$ object) and instead compute the required scalar outer gradient via a single *adjoint* linear solve. This adjoint form is mathematically equivalent to the Implicit Function Theorem (IFT)-based sensitivity expression in (6).

Recall the in-sim SPG convergence (Eq. (4))

$$\hat{\varphi}(\phi,\theta) := \mathbb{E}_{s\sim\hat{\rho}_\theta^{\pi_\phi}}\Big[\nabla_\phi \log \pi_\phi(a|s)\, Q_\theta^{\pi_\phi}(s,a)\Big], \qquad \hat{\varphi}(\phi,\theta) = 0. \tag{106}$$

Under the local regularity assumptions stated in Section 5 (local isolation and nonsingularity of $\nabla_\phi\hat{\varphi}$), the stationarity condition implicitly defines a locally unique differentiable solution branch $\phi = \phi(\theta)$.

Let the outer objective be the real-world return (5), written abstractly as a scalar function

$$\mathcal{L}(\theta) := \mathbb{E}_{\tau\sim\pi_{\phi(\theta)},\mathcal{P}}\Big[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\Big]. \tag{107}$$

This outer objective may, in general, have both (i) *implicit dependence* through $\phi(\theta)$ and (ii) *explicit dependence* on θ (e.g., if additional regularizers are added to the outer level). We keep both possibilities in the derivation.

Avoiding forming $\nabla_\theta\phi$ for scalar objectives

Differentiate $\mathcal{L}(\theta)$ using the chain rule:

$$\nabla_\theta\mathcal{L}(\theta) = \underbrace{\partial_\theta\mathcal{L}(\phi,\theta)}_{c(\theta)} + \underbrace{\big(\nabla_\theta\phi(\theta)\big)^\top\nabla_\phi\mathcal{L}(\phi,\theta)}_{(\nabla_\theta\phi)^\top g}. \tag{108}$$

Let's define the following objects evaluated at (ϕ,θ):

$$A(\phi,\theta) := \nabla_\phi\hat{\varphi}(\phi,\theta) \in \mathbb{R}^{d_\phi\times d_\phi}, \qquad B(\phi,\theta) := \nabla_\theta\hat{\varphi}(\phi,\theta) \in \mathbb{R}^{d_\phi\times d_\theta}, \tag{109}$$

$$g(\phi,\theta) := \nabla_\phi\mathcal{L}(\phi,\theta) \in \mathbb{R}^{d_\phi}, \qquad c(\phi,\theta) := \partial_\theta\mathcal{L}(\phi,\theta) \in \mathbb{R}^{d_\theta}. \tag{110}$$

From the Implicit Function Theorem (IFT) (6), we have

$$\nabla_\theta\phi(\theta) = -A(\phi,\theta)^{-1}B(\phi,\theta). \tag{111}$$

Substituting (111) into (108) yields

$$\nabla_\theta\mathcal{L}(\theta) = c(\phi,\theta) - B(\phi,\theta)^\top A(\phi,\theta)^{-\top}g(\phi,\theta). \tag{112}$$

Therefore, to compute the scalar outer gradient, we do *not* need the full matrix $\nabla_\theta\phi(\theta)$; we only need the vector $A^{-\top}g$.

Adjoint formulation

Introduce the adjoint variable $\lambda \in \mathbb{R}^{d_\phi}$ defined as the solution of the linear system

$$A(\phi,\theta)^\top\lambda = g(\phi,\theta). \tag{113}$$

Then (112) becomes

$$\boxed{\nabla_\theta\mathcal{L}(\theta) = c(\phi,\theta) - B(\phi,\theta)^\top\lambda.} \tag{114}$$

In the common case where the outer objective depends on θ only through the deployed policy $\pi_{\phi(\theta)}$ (so that $c(\phi,\theta)\approx 0$ by construction), we obtain the simplified expression

$$\boxed{\nabla_\theta\mathcal{L}(\theta) = -B(\phi,\theta)^\top\lambda, \qquad A(\phi,\theta)^\top\lambda = g(\phi,\theta).} \tag{115}$$

(114) is the most efficient reverse-mode form of the bi-level gradient for scalar outer objectives: it avoids constructing any dense $d_\phi \times d_\theta$ sensitivity matrix and requires only (i) one linear solve in d_ϕ dimensions and (ii) one vector–Jacobian product to obtain $B^\top\lambda$.

Obtaining $A^\top v$ and $B^\top v$ without forming A or B

The adjoint system (113) and the gradient formula (114) only require the ability to apply the linear operators

$$v \mapsto A(\phi,\theta)^\top v, \qquad v \mapsto B(\phi,\theta)^\top v.$$

These can be computed without forming A or B explicitly using reverse-mode automatic differentiation as *vector–Jacobian products (VJPs)* applied to the residual $\hat{\varphi}(\phi, \theta)$ in (106).

Given a vector $v \in \mathbb{R}^{d_\phi}$:

$$A(\phi,\theta)^\top v = \nabla_\phi\Big(\langle \hat{\varphi}(\phi,\theta), v\rangle\Big), \qquad B(\phi,\theta)^\top v = \nabla_\theta\Big(\langle \hat{\varphi}(\phi,\theta), v\rangle\Big). \tag{116}$$

Thus, iterative Krylov solvers (e.g., GMRES/BiCGSTAB) can be used to solve (113) using only the operator $v \mapsto A^\top v$.

Additionally the operator $A = \nabla_\phi \hat{\varphi}$ need not be symmetric or positive definite. Therefore, a nonsymmetric Krylov method (e.g., GMRES or BiCGSTAB) is generally appropriate. Damping can be used for numerical stability by solving

$$\big(A(\phi,\theta)^\top + \lambda I\big)\lambda = g(\phi,\theta), \qquad \lambda > 0. \tag{117}$$

The adjoint form in (114) is fully consistent with Theorem 1 while avoiding the computational burden of forming $\nabla_\theta \phi$ explicitly. In particular,

$$B(\phi,\theta)^\top \lambda = \Big(\nabla_\theta \hat{\varphi}(\phi,\theta)\Big)^\top \lambda$$

can be estimated using the same trajectory-based estimators used for Theorem 1 (e.g., Lemma 1 and Lemma 2).

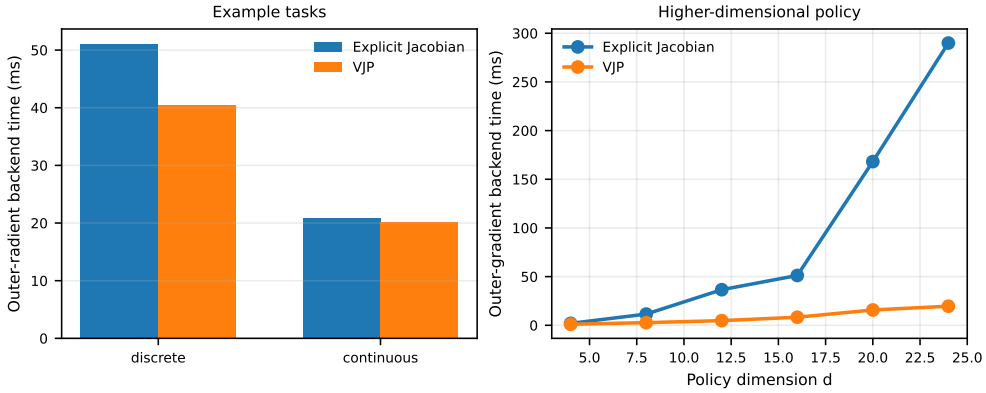

Figure 5: Comparison of computational time between the proposed VJP-based method and explicit Jacobian computation. The left panel shows results for low-dimensional problems (a 3-state discrete MDP and a 1-dimensional continuous MDP), while the right panel reports results for a higher-dimensional NN policy.

Evaluation

A comparison between the VJP approach and explicitly forming the Jacobian $\nabla_\theta \phi$ using a PyTorch implementation is shown in Fig. 5. For the low-dimensional discrete task and the continuous MDP with a NN policy containing six parameters used in Section 6, the difference in computational time is not substantial due to the simplicity of the policies. However, as the size of the policy network increases, the computational advantage becomes significant. For NN policies with more than 20 parameters, the VJP-based method is more than two orders of magnitude faster than explicitly computing the Jacobian. The reported timings focus only on the outer-gradient computation after trajectory batches have already been generated. Since the rollout data are shared between both methods, excluding rollout time isolates the computational cost of the gradient backend and highlights the efficiency gain from avoiding explicit Jacobian construction.

D Details on Illustrative Examples

D.1 Example 1: Discrete MDP

We considered a finite MDP with state space $\mathcal{S} = \{0, 1, 2\}$ and action space $\mathcal{A} = \{0, 1\}$. The real and model MDP parameters have the following form:

Transition logits

$$\theta_{\text{trans}}[s, a] \;=\; \big[\theta_{\text{trans}}[s, a, 0],\, \theta_{\text{trans}}[s, a, 1],\, \theta_{\text{trans}}[s, a, 2]\big]$$

so that

$$P(s' \mid s, a) = \text{softmax}\big(\theta_{\text{trans}}[s, a]\big)_{s'}.$$

The real-MDP transition logits are

$$\theta^f_{\text{real}} = \big\{ \underbrace{[0.5, 2.0, 0.5],\, [1.0, 1.5, 0.5]}_{s=0},\; \underbrace{[1.0, 1.0, 1.0],\, [1.5, 1.0, 0.5]}_{s=1},\; \underbrace{[0.5, 1.0, 0.1],\, [1.0, 0.5, 1.0]}_{s=2} \big\}.$$

$$R(s, a) = \theta^R[s, a],$$

with

$$\theta^R_{\text{real}} = \begin{bmatrix} 1.0 & 0.5 \\ 0.0 & 3.0 \\ 0.01 & 2.0 \end{bmatrix}.$$

A similar structure holds for the simulation MDP, where the simulation model and reward parameters for the trials shown in Fig. 3 were initialized at random in $[0, 5]$.

The in-sim policy is solved using DP with a convergence threshold of 1×10^{-2}, which provides a deterministic policy, which is then converted into a stochastic policy. After solving the discrete MDP by soft value iteration, we obtain a state–action value table

$$Q^\star(s, a) \;=\; R(s, a) \;+\; \gamma \sum_{s'} P(s' \mid s, a)\, V^\star(s'),$$

where $P(s' \mid s, a)$ and $R(s, a)$ come from the MDP parameters. A greedy policy would be

$$\pi_{\text{det}}(s) = \arg\max_a Q^\star(s, a),$$

but this is non-differentiable. Instead, we define a *soft* (stochastic) policy

$$\pi(a \mid s) \;=\; \frac{\exp\big(Q^\star(s, a)/\tau\big)}{\sum_b \exp\big(Q^\star(s, b)/\tau\big)},$$

where $\tau > 0$ is a temperature parameter controlling exploration, which we set as 2.0. Because every entry of $\pi(\cdot \mid s)$ is a smooth function of the underlying $Q^\star$, and $Q^\star$ itself depends differentiably on the simulation parameters θ, this construction makes the whole inner-level fully differentiable.

Once $Q^\star$ is computed, we set the policy logits as:

$$\text{logits}(s, a) \;=\; \log\big(\pi(a \mid s)\big) \quad\Longrightarrow\quad \pi(a \mid s) = \text{softmax}\big(\text{logits}(s, \cdot)\big).$$

Using an on-policy, in-sim trajectory of length 1000 drawn from the DP solution, we compute the Markov-chain sensitivities (Eq. 12 and 13) and the critic sensitivities (Eq. 9 and 10). Each sensitivity table is represented in tabular form and updated iteratively until convergence with a threshold of 1×10^{-2}. For each outer-level iteration, we use a trajectory of length 1000 generated by applying the in-sim policy to the real MDP to update the model and reward parameters using SPG steps according to Eq. (7a) with a learning rate of 0.1.

D.2 Example 2: Continuous MDP

We use a one-dimensional linear system with additive Gaussian noise and an exponential reward:

$$s_{t+1} = \theta_s\, s_t \;+\; \theta_a\, a_t \;+\; \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2), \tag{118}$$

$$R_\theta(s_t, a_t) = \exp\big(-\lambda\,[\,\theta_q\, s_t^2 + \theta_r\, a_t^2\,]\big). \tag{119}$$

The true system uses $\theta_s = \theta_a = \theta_q = \theta_r = 1$, while all five simulation parameters $\{\theta_s, \theta_a, \theta_q, \theta_r\}$ are initialized uniformly at random in $[0, 1]$.

By taking logarithms of the stage reward, we obtain an equivalent quadratic stage cost $c(s, a) = \lambda\,(q\, s^2 + r\, a^2)$, with $q = \theta_q$, $r = \theta_r$. In the infinite-horizon, γ-discounted setting, the optimal control law is linear,

$$a_t^\star = -K\, s_t,$$

where the gain K and value-function curvature P solve the discrete Algebraic Riccati Equation (DARE)

$$P = \lambda q + \gamma\,(\theta_s - \theta_a\, K)^2\, P, \qquad K = \frac{\theta_a\, P\, \theta_s}{r + \theta_a^2\, P}.$$

Once P is found, the optimal value function is $V^\star(s) = P\, s^2$ and the greedy policy is $a = -K\, s$.

To test the robustness of our sensitivity estimates, we fit two small feed-forward NN of one hidden layer of size 6 for the policy and 64 for the value function:

$$V_\psi(s) \approx V^\star(s), \qquad \hat{\pi}_\phi(a \mid s) \approx \delta(a + K\, s),$$

where $\hat{\pi}_\phi$ is trained to match the exact linear feedback as a stochastic Gaussian policy. The learned V and $\hat{\pi}$ are then used in place of the true LQR quantities in the sensitivity computations inner level.

Using an on-policy, in-sim trajectory of length 1000 drawn from the approximated NN policy, we compute the Markov-chain sensitivities (Eq. 12 and 13) and the critic sensitivities (Eq. 9 and 10). Sensitivities are computed for each sample until convergence of 1×10^{-2}. For each outer-level iteration, we use 20 trajectories of length 200 generated by applying the in-sim NN approximated policy to the real MDP to update the model and reward parameters using SPG steps according to Eq. (7a) with a learning rate of 0.1.

D.3 Example 3: Quadrotor

To evaluate the proposed bi-level PPO scheme in a nonlinear continuous-control setting, we consider the planar quadrotor (Quad2D) environment implemented in `safe-control-gym` (Yuan et al., 2022). The objective is to stabilize the quadrotor at a desired reference position as quickly as possible while minimizing control effort.

The system state is defined as

$$s = [x, \dot{x}, z, \dot{z}, \phi, \dot{\phi}]^T,$$

where (x, z) denote the planar position of the vehicle and ϕ is the pitch angle. The control inputs correspond to the thrusts generated by the two rotors, $a = [u_1, u_2]^T$.

The simplified dynamics of the system are given by

$$m\ddot{z} = (u_1 + u_2)\cos\phi - mg, \tag{120}$$

$$m\ddot{x} = -(u_1 + u_2)\sin\phi, \tag{121}$$

$$I_{yy}\ddot{\phi} = l(u_2 - u_1), \tag{122}$$

where m denotes the quadrotor mass, l the arm length, and I_{yy} the moment of inertia around the pitch axis. These dynamics capture the key nonlinear coupling between translational motion and attitude.

The reward function follows an exponential quadratic form similar to Example 2:

$$r_t = \exp\big(-(s_t - s_{ref,t})^T W_s (s_t - s_{ref,t}) - (a_t - a_{ref,t})^T W_a (a_t - a_{ref,t})\big)\,, \tag{123}$$

where $s_{ref,t}$ denotes the desired reference state, $a_{ref,t}$ is the hover thrust, and W_s, W_a are the cost weights. This formulation encourages accurate tracking while penalizing excessive control effort.

For the outer task, the quadrotor mass and arm length are $m_0 = 0.033\,kg, l_0 = 0.039\,m$ and the cost weights are $W_s = diag([5, 0.1, 5, 0.1, 0.1, 0.001]), W_a = diag([0.1, 0.1])$. For the in-sim task, the tunable simulation parameter is $\theta = [\theta_m, \theta_l, \theta_{ws}, \theta_{wa}]^T$ such that quadrotor mass $m = \theta_m m_0$, the arm length $l = \theta_l l_0$, and the cost weights $W_s = diag(\theta_{ws}), W_a = diag(\theta_{wa})$. The in-sim parameter θ is initialized as $\theta = [0.5, 1.0, 5, 0.1, 5, 0.1, 0.1, 0.001, 0.1, 0.1]$.

Bi-level PPO details

Since no closed-form optimal policy exists for this nonlinear system, both the policy and value function are approximated using NNs trained with PPO. The actor is implemented as a multi-layer perceptron with two hidden layers of 64 units each, while the critic uses two hidden layers of 128 units. Both networks use *Tanh* activations.

The in-sim RL performs standard PPO updates in simulation to improve the policy under the current simulator parameter θ. Each PPO update uses 10 optimization epochs with a learning rate of 3×10^{-4}. Additional hyperparameters include a discount factor $\gamma = 0.99$, GAE parameter $\lambda = 0.95$, and entropy coefficient 0.01.

To estimate the Markov-chain sensitivities required for the outer-level updates, 40 trajectories are sampled from the simulator using the current policy. These samples are used to compute the sensitivities according to Eq. 11, Eq. 12, and Eq. 13. Note that a first-order approximation of Eq. 11 is used in the quadrotor example, wherein the Hessian term is neglected for computational reasons.

For each outer-level iteration, n in-sim PPO updates are carried out before applying a simulator-parameter update using the stochastic policy-gradient step in Eq. (7a). Results are reported for $n = 20$. The outer-loop learning rate is set to 10^{-3}.

E Related Works

Sim-to-real gap

The problem of the sim-to-real gap has been commonly addressed by either enriching the simulator fidelity or exposing agents to diverse simulated scenarios. Domain randomization is a common approach that perturbs the simulation parameters during training to improve robustness, such that the resulting in-sim policy generalizes better to the real-world (Tobin et al., 2017; Duan et al., 2024). *Domain adaptation* approaches instead aligns feature or parameter distributions between the simulator and real world, commonly via adversarial training on observations (Chen et al., 2020; Long et al., 2018; Pei et al., 2018; Höfer et al., 2021; Eysenbach et al., 2020). In *meta RL*, the agent is trained on a distribution of environments with the goal that the agent can then quickly adapt to the real world (Finn et al., 2017). Other approaches include grounded action methods to adjust simulator dynamics to match observed real trajectories (Hanna and Stone, 2017; Desai et al., 2020), and distributionally robust RL formulates the problem as maximizing worst-case returns under bounded transition shifts (Liu and Xu, 2024). Most of these approaches are robustness-focused rather than performance-oriented; they seek to reduce the sim-to-real gap by optimizing for proxy objectives (e.g., simulator accuracy, variability) rather than the real-world performance directly (Da et al., 2025). As a result, they can improve stability under distribution shift but do not offer a guarantee on optimal performance. Few works have explored embedding task performance in domain adaptation. For example, RL-CycleGAN fine-tunes scene parameters by minimizing an RL-consistency loss on a learned Q-function (Rao et al., 2020), yet this is an indirect surrogate for real-world performance. In contrast, the sensitivity analysis provided in this paper enables the simulation adaptation directly with the real-world performance of the in-sim policy.