ROBUST DATA-COLLECTION POLICY LEARNING FOR LOW-VARIANCE ONLINE POLICY EVALUATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In reinforcement learning policy evaluation, classic on-policy methods often suffer from high variance when estimating policy performance. To mitigate this issue, *behavior policy search* has been proposed to learn data-collecting policies tailored to reduce online evaluation variance. However, these approaches do not account for uncertainties in the transition functions. In practice, simulator transitions often differ from the real world due to modeling errors or approximation limitations. As a result, behavior policies trained in simulation may still yield high variance when deployed in real environments, leading to costly reliance on real-world evaluation samples. In this work, we propose a double-loop gradient-based algorithm for learning behavior policies that are both efficient and robust to transition uncertainty. Theoretically, we derive novel transition-variance gradient expressions and establish global convergence guarantees for the algorithm. Numerically, we demonstrate that our method is less sensitive to transition perturbations than existing approaches, providing supportive evidence for its practical utility.

1 INTRODUCTION

Reinforcement learning (RL) has achieved remarkable success in recent years across domains such as robotics, healthcare, recommendation systems, and natural language processing (Mnih et al., 2015; Silver et al., 2017; Jumper et al., 2021). A central component in these advances is policy evaluation, the task of estimating the performance of a policy. The most direct approach is the on-policy Monte Carlo method, which collects trajectories from the target policy and estimates its value by averaging the observed returns. Although conceptually simple and widely used, this method often suffers from high evaluation variance, limiting the reliability of the resulting estimates.

To improve efficiency, a growing body of work has investigated learning a separate data-collecting behavior policy to reduce evaluation variance via off-policy evaluation (Hanna et al., 2017; Zhong et al., 2022; Liu & Zhang, 2024; Liu et al., 2025a). This line of research, known as behavior policy search (BPS), optimizes a variance-reducing behavior policy so that the collected trajectories yield more informative evaluations. With an appropriately chosen behavior policy, BPS has been shown to achieve lower variance than naive on-policy evaluation. By contrast, in the standard formulation of off-policy evaluation (OPE), the data are assumed to be pre-logged by a fixed behavior policy and the focus is on designing improved estimators. In comparison, BPS explicitly optimizes the behavior policy itself to reduce variance and is broadly applicable across different OPE estimators.

Despite this progress, existing BPS methods generally optimize behavior policies under prespecified transition functions, without accounting for underlying uncertainties. In practice, the true transition functions often deviate from the assumed model due to approximation errors, adversarial perturbations, or partial observability. Such discrepancies can compromise evaluation reliability: a behavior policy optimized under simulator transition may still yield high variance when deployed in the real environment. Consequently, prior methods may continue to require a large number of costly real-world samples to achieve accurate evaluation.

To address these two challenges—variance reduction and transition mismatch—we propose an **efficient and robust policy evaluation** framework that explicitly accounts for transition uncertainty. Our method formulates BPS as a minimax optimization problem, where an adversarial transition model seeks to maximize the evaluation variance while the behavior policy is optimized to minimize it. Our contributions are summarized as follows: **(1)** We introduce a novel adversarial framework for

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

robust behavior policy search in policy evaluation (Section 3.3); **(2)** We derive analytical transition-gradient expressions for the on-transition setting, and provide convergence guarantees for the inner-loop adversarial optimization (Section 4); **(3)** We propose a double-loop robust gradient algorithm and provide global convergence guarantee for variance-minimizing behavior policy search (Section 5); **(4)** We numerically show that our method is less sensitive to transition perturbations than existing approaches (Section 6), verifying the utility of our theoretical results.

2 Related Work

Behavior Policy Search. Behavior policy search (BPS) reduces evaluation variance by optimizing the data-collecting policy. Hanna et al. (2017) formulate this as an optimization problem, using stochastic gradient descent to outperform standard Monte Carlo methods. Zhong et al. (2022) extend this via adaptive behavior policies that prioritize under-sampled regions. However, these approaches overlook transition uncertainty; consequently, behavior policies optimized in simulation may still induce high variance under real-world dynamics. In contrast, our method explicitly models transition uncertainty to ensure the learned behavior policy remains effective even under perturbed dynamics.

Liu & Zhang (2024) also study the variance reducing problem, deriving a closed-form, offline-learnable behavior policy with theoretical guarantees. However, their formulation relies on pre-logged data with prespecified and fixed transition probabilities, and cannot adapt when these probabilities shift at deployment, leaving it vulnerable to modeling errors and sim-to-real mismatches. Our method fills this gap by integrating adversarial transition modeling into behavior policy search, combining efficiency with robustness to transition shifts. Similarly, while Russo & Pacchiano (2025) optimize adaptive exploration for multi-policy evaluation, they assume fixed dynamics; in contrast, our framework proactively targets robustness to adversarial transition shifts.

Robust Policy Evaluation. Recent work has begun addressing robustness in reinforcement learning policy evaluation. For example, Katdare et al. (2023) and Voloshin et al. (2021) propose techniques to improve robustness under simulator mismatch via estimator modification or robust model learning. However, these methods either rely on access to real-world data or focus on minimizing worst-case prediction errors, and do not directly address the high-variance issue central to policy evaluation. In contrast, our work proactively reduces variance by designing behavior policies that are robust to adversarial transition shifts, without requiring target-environment data.

While robust MDP (RMDP) frameworks (Iyengar, 2005; Nilim & El Ghaoui, 2005) also consider robustness under transition uncertainty, they are typically designed for reward maximization and rely on linear programming techniques. These approaches (e.g., Wang et al. (2023a; 2024)) do not apply to our setting, where the goal is to minimize the variance of policy value estimators, a fundamentally non-linear objective. We fill this gap by proposing a novel adversarial transition variance gradient method that explicitly targets variance reduction under transition uncertainty.

3 Background

3.1 Markov Decision Process

We study a finite-horizon Markov Decision Process (MDP, Puterman (2014)) with finite state space $\mathcal{S}$ and finite action space $\mathcal{A}$. For a finite set $\mathcal{X}$, we denote the probability simplex over $\mathcal{X}$ by $\Delta(\mathcal{X}) \doteq \{p : \mathcal{X} \to [0,1] \mid \sum_{x\in\mathcal{X}} p(x) = 1\}$. The MDP consists of a transition probability function $p : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, a reward function $r : \mathcal{S} \times \mathcal{A} \to [0,1]$, an initial state distribution $p_0 \in \Delta(\mathcal{S})$, and a fixed horizon length T. To simplify notation, we consider the undiscounted setting without loss of generality. Our method naturally applies to the discounted setting as long as the horizon is fixed and finite (Puterman, 2014).

A policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ maps each state to a probability distribution over actions. We consider the parameterized policies π_θ, where the parameters $\theta \in \Theta$ is a vector with $\Theta \subseteq \mathbb{R}^n$ for some constant n. Likewise, we parameterize the transition function $p_\omega : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ by a parameter $\omega \in \Omega$, where $\Omega \subseteq \mathbb{R}^m$ and is compact. Unless stated otherwise, all norms used in this work are Euclidean (i.e., $\|x\| = \|x\|_2$).

The MDP process begins at time step 0, where an initial state S_0 is sampled from p_0. At each time step $t \in [T-1]$, an action A_t is sampled based on $\pi(\cdot \mid S_t)$. Then, a finite reward $R_{t+1} \doteq r(S_t, A_t)$ is given by the environment and a successor state S_{t+1} is obtained based on $p(\cdot \mid S_t, A_t)$. After T steps, the agent's interaction with the environment terminates. If the agent reaches any terminal state before time step T, it stays there and receives zero reward.

We use $h \doteq \{S_0, A_0, R_1, S_1, A_1, ...S_{T-1}, A_{T-1}, R_T, S_T\}$ to denote the trajectory of this MDP. We then define the *return* of h as $g(h) \doteq \sum_{t=0}^{T-1} R_{t+1}$. For any policy, we have a distribution over the trajectory as $\Pr(H = h|\pi)$, where H is a random variable used to denote the trajectory. Lastly, we define the *value* of a policy as $v(\pi) \doteq \mathbb{E}_{H\sim\pi}[g(H)]$. To simplify notations, we also define $\ell_{p_\omega} \doteq \sum_{t=0}^{T-1} \log(p_\omega(S_{t+1}|S_t, A_t))$.

3.2 Variance Reduction in Policy Evaluation

We consider the task of reinforcement learning policy evaluation, where the goal is to estimate the value of an interested policy π_e, called the *target policy*. The traditional *on-policy* Monte Carlo (MC) method estimates $v(\pi_e)$ by repeatedly executing the target policy π_e online and averaging the observed returns. That is, $\mathrm{MC}(\pi_e, H) \doteq \frac{1}{n}\sum_{i=0}^{n-1} g(H_i)$ for all $H_i \sim \pi_e$. However, in practice, this straightforward method can induce high evaluation variance, leading to less reliable results (Liu & Zhang, 2024; Liu et al., 2025b;a; Chen et al., 2025).

To mitigate this challenge, recent work has proposed to use *off-policy* evaluation method to reduce variance, where we execute a different policy π_θ (the *behavior policy*), to collect data. For wide applicability, we consider a general off-policy estimator $\mathrm{OPE}(\pi_e, \pi_\theta, H)$, which estimates the value of π_e using trajectories H from π_θ. A standard example is importance sampling, $\mathrm{IS}(\pi_e, \pi_\theta, H) \doteq g(H)\prod_{t=0}^{T-1} \frac{\pi_e(A_t|S_t)}{\pi_\theta(A_t|S_t)}$. Prior work has shown that with a properly designed behavior policy π_θ, one can achieve lower evaluation variance with an off-policy estimator than the traditional on-policy MC method (Hanna et al., 2017; Zhong et al., 2022). This is known as the *Behavior Policy Search* (BPS) problem, where we aim to solve $\min_{\theta\in\Theta} \mathbb{V}_{H\sim\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, H)]$.

3.3 Robust Behavior Policy Search

Standard behavior policy search methods typically assume fixed transition probabilities, but in practice there are often discrepancies between simulators and real environments. After learning a variance-reducing behavior policy in simulation, practitioners typically deploy it to evaluate the target policy in the real system. However, because *robustness to dynamics shifts* is not considered during the behavior policy search phase, the resulting policy may still induce high variance when collecting real-world data. Consequently, achieving reliable evaluation often requires a large amount of costly real-world samples, motivating a robustness-aware formulation.

To address this, we formulate the robust behavior policy search problem as a minimax optimization:

$$\min_{\theta\in\Theta} \max_{\omega\in\Omega} \mathbb{V}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, H)\right], \tag{1}$$

where the inner maximization identifies worst-case transition perturbations, and the outer minimization seeks a behavior policy that mitigates this adversarial effect. Such min–max formulations are standard in the robust RL literature for modeling adversarial dynamics (Katdare et al., 2023; Voloshin et al., 2021). Following standard practice in robust RL (Iyengar, 2005; Nilim & El Ghaoui, 2005; Ho et al., 2021; Wang et al., 2023a), our robustness guarantees are defined with respect to a user-specified transition uncertainty set, enabling practitioners to encode transition uncertainties appropriate to their application.

To further analyze the min-max problem in (1), we can write it as the following equivalent problem

$$\min_{\theta\in\Theta}\{\Phi(\theta) \doteq \max_{\omega\in\Omega} \mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, H)]\}, \tag{2}$$

which minimizes the worst-case evaluation variance (Jin et al., 2020). Notably, the function Φ is not differentiable, and is neither convex nor concave. Thus, we are unable to solve the problem through direct gradient descent on the function Φ, which motivates our double-loop approach (Algorithm 2) in Section 5 with global convergence guarantee.

4 SOLVING THE INNER LOOP

In this section, we study the inner loop of the optimization problem (1), which identifies adversarial dynamics that maximize evaluation variance. We derive analytical gradient expressions of the variance with respect to the transition probability in the *on-transition* case. In the on-transition case, the simulator transition can be modified, so trajectories are sampled directly from the evolving p_ω at each iteration. We introduce Algorithm 1, which adapts p_ω to maximize evaluation variance, serving as the adversarial player against the robust behavior policy π_θ (Section 5). We provide theoretical convergence guarantees for this algorithm. To the best of our knowledge, this is the first work to develop adversarial transition-gradient methods for variance objectives in reinforcement learning.

We begin with the *on-transition* case, analogous to the on-policy setting, where the simulator transition can be directly modified to follow the target transition p_ω at each iteration. Given a fixed behavior policy π_θ, we look for the variance-maximizing adversarial transition p_ω. Formally, we need to solve

$$\max_{\omega\in\Omega} \mathbb{V}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\right].$$

In the following theorem, we give a gradient expression of the evaluation variance. Importantly, this analytical form is general and applies to any off-policy evaluation estimator, forming the foundation of our transition-gradient method.

Theorem 4.1 (Transition Gradient of the Variance)**.** *For a fixed behavior policy π_θ,*

$$\begin{aligned}
&\frac{\partial}{\partial\omega}\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}\ell_{p_\omega}\right]\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right].
\end{aligned}$$

Its proof is in Appendix A.2. This gradient expression is in expectation forms and can be estimated unbiasedly from sampled trajectories without additional structural assumptions on the OPE estimator. Building on Theorem 4.1, we now present the On-transition Variance Gradient method in Algorithm 1. We instantiate our algorithm with the importance sampling estimator (IS), but our framework is ready to accommodate any off-policy evaluation estimator.

To discuss the convergence property of Algorithm 1, we impose the standard Robbins-Monro step-size condition $\sum_{i=0}^{\infty}\alpha_i=\infty$ and $\sum_{i=0}^{\infty}\alpha_i^2<\infty$ (Robbins & Monro, 1951). We assume the importance sampling ratio $\frac{\pi_e(a|s)}{\pi_\theta(a|s)}$ exists and is bounded above for all s, a, and θ (Hanna et al., 2024). Besides, we require the transition p_ω to be twice-differentiable with respect to ω with uniformly bounded first- and second-order derivatives. These conditions for p_ω hold, for example, when it is parameterized by a neural network with smooth activations and a softmax output layer, and are commonly adopted in policy gradient literature (e.g., Hanna et al. (2017; 2024)). Then, we have the following lemma for the convergence of Algorithm 1, whose proof is in Appendix A.3.

Lemma 4.2 (Transition Gradient Convergence)**.** *For a fixed behavior policy π_θ, Algorithm 1 converges. That is, $\mathbb{V}_{H_i\sim p_{\omega_i},\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]$ converges to a finite value and* $\lim_{i\to\infty}\frac{\partial}{\partial\omega}\mathbb{V}_{H_i\sim p_{\omega_i},\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]=0$.

In practice, although discrepancies often exist between the transition probability in the deployment environment and the original simulator, the simulator typically remains a reasonable approximation. Thus, to ensure the learned adversarial transition remains realistic rather than overly pessimistic, we also offer an optional Kullback–Leibler (KL) divergence penalty that discourages large deviations between p_ω and the initial simulator transition p_{ω_0}. Given a behavior policy π_θ, we consider the following inner-loop optimization problem under KL regularization:

$$\max_{\omega\in\Omega}\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]-\eta\mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0})),$$

where $\eta>0$ is the regularization coefficient and the KL-divergence term is defined as $\mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0}))\doteq\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\log\frac{\Pr(H|p_\omega)}{\Pr(H|p_{\omega_0})}\right]$. We provide the gradient expression of this regularized optimization problem in the following theorem.

Algorithm 1 On-Transition Variance Gradient.

1: **Input:** an initial transition parameter ω_0, a target policy π_e, a fixed behavior policy π_θ, a number of iteration n, a batch size k (a positive even integer), a step-size α_i for each i
2: **Output:** a final adversarial transition parameter ω_n
3: **For all** $i \in 0, ..., n-1$ **do**
4: Sample k trajectories $H \sim \pi_\theta, p_{\omega_i}$
5: $$\omega_{i+1} = \omega_i + \frac{\alpha_i}{k}\sum_{j=1}^{k}\left(\mathrm{IS}(\pi_e,\pi_\theta,H^j)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log\big(p^j_{\omega_i}(S_{t+1}|S_t,A_t)\big)\right)$$
$$-\frac{8\alpha_i}{k^2}\sum_{j=1}^{\frac{k}{2}}\mathrm{IS}(\pi_e,\pi_\theta,H^j)\sum_{j=\frac{k}{2}+1}^{k}\left(\mathrm{IS}(\pi_e,\pi_\theta,H^j)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log\big(p^j_{\omega_i}(S_{t+1}|S_t,A_t)\big)\right).$$
6: **End for**
7: **Return:** ω_n

Theorem 4.3 (Transition Gradient of Variance with KL)**.** *For a fixed behavior policy π_θ and a regularization coefficient $\eta > 0$,*

$$\begin{aligned}
&\frac{\partial}{\partial\omega}\Big[\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)] - \eta\mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0}))\Big]\\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}\ell_{p_\omega}\right]\\
-&2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right]\\
&-\eta\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\left(\frac{\partial}{\partial\omega}\ell_{p_\omega}\right)\left(1+\ell_{p_\omega}-\ell_{p_{\omega_0}}\right)\right].
\end{aligned}$$

Its proof is in Appendix A.4. This regularization balances robustness with realism, ensuring that the learned adversary remains close to plausible dynamics.

5 Solving the Outer Loop

In this section, we propose a behavior policy search (BPS) method that is robust to potential discrepancies in the environment. Specifically, we adopt a policy gradient approach to search for a variance-reducing behavior policy under an adversarial transition probability. We first introduce this algorithm, theoretically analyzing its global convergence guarantee. Then, in Section 6, we demonstrate its empirical robustness under perturbed transition probabilities.

5.1 Double-Loop Robust Variance Gradient

To begin with, recall that in (1), our goal is to solve the min-max objective

$$\min_{\theta\in\Theta}\max_{\omega\in\Omega}\mathbb{V}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\right].$$

In Section 4 and Algorithm 1, we present methods to solve the inner maximization problem by performing gradient ascent on the transition parameter ω. In this section, we focus on performing gradient descent for the variance objective on the policy parameter θ. This is also known as the *behavior policy search* problem in off-policy evaluation (OPE) community (Hanna et al., 2017; 2024), which aims at finding a variance minimizing behavior policy to collect data through gradient based methods. In Lemma 5.1, we present the gradient expression for variance with respect to the behavior policy adopted from Hanna et al. (2017).

Lemma 5.1 (Variance Gradient Expression)**.** *With a fixed transition probability p_ω, $\forall\theta$,*

$$\frac{\partial}{\partial\theta}\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)] = \mathbb{E}_{H\sim p_\omega,\pi_\theta}[-\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\theta}\log\pi_\theta(A_t|S_t)].$$

Importantly, this lemma shows that we can estimate the gradient with trajectories sampled from the behavior policy π_θ. With this analytical expression, we are now ready to present our double loop algorithm, named *Double-Loop Robust Variance Gradient* (DRVG).

Algorithm 2 Double-Loop Robust Variance Gradient (DRVG)

1: **Input:** a target policy parameter θ_e, an initial parameter $\theta_0 \in \Theta$, a number of iteration n, a batch size K_{out}, a step-size α, tolerance sequence $\{\epsilon_i\}$
2: **Output:** an averaged behavior policy parameter $\bar{\theta}$
3: **For all** $i = 0, ..., n-1$ **do**
4: Find p_{ω_i} s.t. $\mathbb{V}_{H\sim p_{\omega_i},\pi_{\theta_i}}[\text{IS}(\pi_e, \pi_{\theta_i}, H)] \geq \max_{p_\omega} \mathbb{V}_{H\sim p_\omega,\pi_{\theta_i}}[\text{IS}(\pi_e, \pi_{\theta_i}, H)] - \epsilon_i$.
5: After selecting ω_i, sample a fresh batch of K_{out} conditionally independent trajectories $H_{i,j} \sim p_{\omega_i}, \pi_{\theta_i}$.
6: $\widehat{\mathcal{G}}_i = -\frac{1}{K_{\text{out}}} \sum_{j=1}^{K_{\text{out}}} \text{IS}(\pi_e, \pi_{\theta_i}, H_{i,j})^2 \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(A_t^{i,j}|S_t^{i,j})\Big|_{\theta=\theta_i}$.
7: $\theta_{i+1} = \text{Proj}_\Theta[\theta_i - \alpha\widehat{\mathcal{G}}_i]$.
8: **End for**
9: **Return:** $\bar{\theta} \doteq \frac{1}{n}\sum_{i=0}^{n-1} \theta_i$.

The double loop algorithm DRVG iteratively takes gradient steps on the evaluation variance objective to solve the min-max problem in (1). Specifically, the inner loop of DRVG returns a worst-case transition probability p_{ω_i} up to a precision ϵ_i. Algorithm 1 serves as a practical inner solver. The outer-loop convergence guarantee assumes that its returned transitions satisfy the stated inner-accuracy condition. Such a sequence $\{\epsilon_i\}$ introduces more flexibility to this double-loop algorithm, allowing for quick policy updates without hurting the global convergence property. This convention of using an ϵ_i-optimal inner solution to analyze outer-loop convergence is also adopted by prior work in the robust MDP community (Ho et al., 2021; Wang et al., 2023a).

In the outer loop, DRVG takes a *projected gradient step* to minimize evaluation variance within the feasible parameter set Θ. A well-known proximal representation of projected gradient in Bertsekas (1995) is $\theta_{i+1} \in \text{argmin}_{\theta\in\Theta}\langle\widehat{\mathcal{G}}_i, \theta - \theta_i\rangle + \frac{1}{2\alpha}\|\theta - \theta_i\|^2 = \text{Proj}_\Theta[\theta_i - \alpha\widehat{\mathcal{G}}_i]$, where Proj_Θ is the projection operator onto Θ. In other words, it is identical to taking a plain gradient step, and then using the closest feasible point in Euclidean distance within the feasible set. Notably, when the feasible set Θ is convex, this projected gradient step can be implemented by a convex optimization solver with a quadratic objective (Wang et al., 2023a). Together, this double-loop algorithm yields a robust behavior policy for off-policy evaluation under environment uncertainty.

5.2 Global Convergence Analysis

In this subsection, we present the global convergence analysis for Algorithm 2. For the widely-studied policy gradient methods in reinforcement learning *policy improvement*, the objective function is the *performance* of a given target policy. In our *policy evaluation* setting, however, in order to minimize the ultimate online samples needed in the real-world evaluation, the objective function is the *performance's variance*,

$$\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\text{OPE}(\pi_e, \pi_\theta, H)] = \mathbb{E}_{H\sim p_\omega,\pi_\theta}[\text{OPE}(\pi_e, \pi_\theta, H)^2] - \mathbb{E}_{H\sim p_\omega,\pi_\theta}[\text{OPE}(\pi_e, \pi_\theta, H)]^2.$$

The non-linear nature of this variance objective introduces additional difficulties, making the min-max optimization problem (1) nonconvex-nonconcave, which is widely known to be challenging (Jin et al., 2020; Nouiehed et al., 2019; Lin et al., 2020). Besides, the objective function $\Phi(\theta)$ in the equivalent expression (2) is generally non-differentiable and nonconvex, making the theoretical analysis to our Algorithm 2 even more challenging. In fact, even without the inner maximization problem, finding the global optima of such nonconvex objectives is already NP-hard in the worst case (Jin et al., 2020).

In *policy improvement* regime without robustness consideration (i.e., a single-loop performance maximization problem), recent work has shown that some algorithms are guaranteed to converge to a globally-optimal policy with a non-convex objective function in *tabular* MDPs (Agarwal et al., 2021; Bhandari & Russo, 2021). When robustness is introduced via a min–max formalization, only recently was the first generic algorithm with global convergence proposed (Wang et al., 2023a). However,

since their inner maximization objective (policy performance) reduces to a linear program in each update, the setting is considerably simpler than our variance-based objective.

In Section 6, we demonstrate the empirical performance of our Algorithm 2 under *a neural network policy parameterization*. While in this section, for the theoretical analysis of Algorithm 2, we adopt a linear-softmax parameterization for the behavior policy π_θ, $\pi_\theta(a|s) \doteq \frac{\exp\left(\theta_a^\top \phi(s)\right)}{\sum_{a' \in \mathcal{A}} \exp\left(\theta_{a'}^\top \phi(s)\right)}$, where $\phi : s \to \mathbb{R}^d$ is a state feature function, and $\theta_a \in \mathbb{R}^d$ is the parameter associated with action $a \in \mathcal{A}$. In this section, we assume that the parameters' feasible set Θ to be closed and convex with a diameter D (i.e., $\forall \theta, \theta' \in \Theta$, $\|\theta - \theta'\| \leq D$), and assume the linear feature to be bounded (i.e., $\forall s$, $\|\phi(s)\| \leq B$ for $B \in \mathbb{R}$) . This choice enables generalization across states through shared features, and makes the variance objective convex in θ. This assumption has been widely adopted in recent theoretical work on policy gradient (Agarwal et al., 2021; Yuan et al., 2022; Cayci et al., 2024).

With the smoothness of this linear-softmax parameterization, we first establish Lemma 5.2, which characterizes the behavior of the objective function $\mathbb{V}$ with respect to the policy parameter θ. This lemma then helps to derive the Lipschitz continuity and convexity of the otherwise non-convex and non-differentiable objective function Φ in (2).

Lemma 5.2. *Under linear-softmax policy parameterization, the objective function* $\mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ *is* L_Θ*-Lipschitz,* ℓ_Θ*-smooth, and convex in* θ *with* $L_\Theta = \sqrt{2} B C^{2T} T^3$ *and* $\ell_\Theta = B^2 C^{2T} T^3 (5 + 8T)$*, where* C *denotes an upper bound on the importance sampling ratio with* $\frac{\pi_e(a|s)}{\pi_\theta(a|s)} \leq C, \forall (s, a), \forall \theta \in \Theta$.

Its proof is in Appendix A.5. We assume bounded importance-sampling ratios, as is standard in off-policy evaluation to ensure finite variance (Hanna et al., 2017; 2024), which can be simply satisfied by bounding the behavior policy away from zero. While the theoretical constants scale with horizon T, this dependence is intrinsic to importance sampling–based approaches and has also appeared in prior OPE analyses (Liu et al., 2018; 2020). Our result shows that global convergence still holds with finite-sample guarantees despite this scaling. With Lemma 5.2, we further obtain the desired properties of Φ, which serve as key building blocks for the global convergence of Algorithm 2.

Lemma 5.3. *The function* $\Phi(\theta)$ (2) *is* L_Θ*-Lipschitz and convex in* θ.

Its proof is in Appendix A.6. Equipped with Lemma 5.2 and Lemma 5.3, we are now ready to establish the convergence analysis despite the inherent nondifferentiability. The following theorem provides a finite-sample convergence guarantee for our double-loop algorithm.

Theorem 5.4 (Double loop global convergence)**.** *With a constant step size* $\alpha \doteq \frac{D}{\sqrt{n(L_\Theta^2 + \sigma_\theta^2 / K_{\text{out}})}}$,

$$\mathbb{E}[\Phi(\bar{\theta})] - \min_{\theta \in \Theta} \Phi(\theta) \leq \frac{D L_\Theta}{\sqrt{n}} + \frac{D \sigma_\theta}{\sqrt{n K_{\text{out}}}} + \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i]. \tag{3}$$

Here K_{out} *is the outer trajectory batch size; for averages of independent on-transition trajectory gradients from Lemma 5.1, one may take* $\sigma_\theta^2 = L_\Theta^2 = 2B^2 C^{4T} T^6$.

Its proof is in Appendix A.7. This result shows that Algorithm 2 converges to an $\epsilon-$optimal solution in expectation at a rate of $\mathcal{O}(\frac{1}{\sqrt{n}})$, provided $\frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i] = \mathcal{O}(n^{-1/2})$, where n is the number of iterations. This rate matches the optimal rate of projected gradient descent in convex optimization, although our min–max variance objective is more challenging than the performance-based objectives studied in prior work (Agarwal et al., 2021; Bhandari & Russo, 2021; Wang et al., 2023a). The error bound consists of three parts: the first term $\frac{D L_\Theta}{\sqrt{n}}$ reflects the convergence rate of projected gradient descent, the second term $D\sigma_\theta / \sqrt{n K_{\text{out}}}$ accounts for outer-gradient sampling error, while the third term $\frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i]$ accounts for the chosen precision in the inner maximization. To our knowledge, this is the *first* global convergence guarantee for variance-minimizing behavior policy search under adversarial transitions, filling an important gap between classical off-policy evaluation and robust reinforcement learning. Finally, we note that double-loop adversarial optimization is standard in

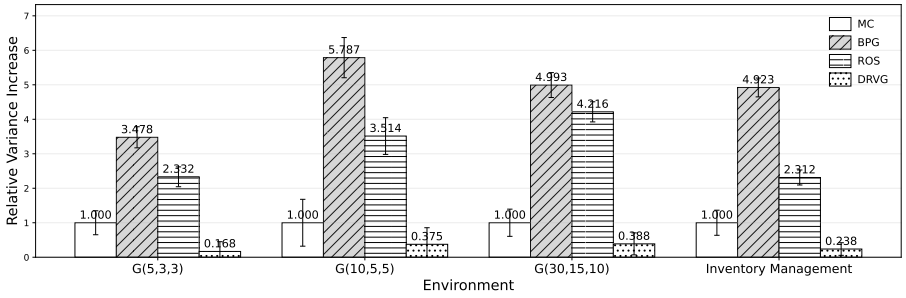


Figure 1: Relative variance increase of each method under its tailored adversarial transition, compared to the variance under the original simulator transition. All values are normalized by the variance increase of the on-policy Monte Carlo (MC) method in the same environment. More details are provided in Appendix B. Error bars denote the standard error.

robust reinforcement learning (e.g., Wang et al. (2023a); Ho et al. (2021); Wang et al. (2024)). As in prior work, our method trades additional, low-cost simulator computation for improved robustness and reliability under transition uncertainty.

6 Numerical Results

In this section, we provide numerical results to validate the utility of our efficient and robust evaluation framework. Our primary goal is to examine two key questions: **(1)** Is our method robust to adversarial transition perturbations? **(2)** Does it give lower evaluation variance under perturbed transitions compared with standard on-policy Monte Carlo?

We evaluate these questions on two environments. Garnet MDPs (Archibald et al., 1995) provide a class of randomly generated abstract MDPs that allow controlled investigation of robustness properties. A Garnet instance $G(|\mathcal{S}|, |\mathcal{A}|, b)$ is parameterized by the number of states $|\mathcal{S}|$, number of actions $|\mathcal{A}|$, and a branching factor b that controls the connectivity of transitions. Owing to this flexibility, Garnets are a standard setting for analyzing robustness in controlled MDP studies (Tarbouriech & Lazaric, 2019; Wang et al., 2023a;b). Inventory management (Porteus, 2002; Ho et al., 2018) is a classical stochastic control problem where a retailer makes ordering decisions under uncertain demand. It provides a natural testbed for evaluating policy performance under transition uncertainties. Notably, compared with recent related work in robust reinforcement learning, our experimental environments operate at comparable or higher complexity (see Table 1).

Method	**Garnet**	**Inventory**
Ours	$\mathbf{G(30, 15)}$	✓
(Wang et al., 2023a) ICML'23	$G(15, 8)$	✓
(Sun et al., 2024) NeurIPS'24	$G(10, 5)$	✗

Table 1: Environments used in recent robust-RL works. Larger $G(\cdot, \cdot)$ settings indicate more challenging Garnet tasks. Branching factors are omitted as they are not specified in the related work.

To contextualize the results, we compare our approach with several representative methods: the on-policy Monte Carlo estimator (MC), the behavior policy gradient estimator (BPG, Hanna et al. (2017)), and the robust on-policy sampling estimator (ROS, Zhong et al. (2022)). All methods are trained with the same initial transition function to obtain their behavior policies. We parameterize our behavior policy with a neural network and use the final iterate behavior policy from Algorithm 2 to collect evaluation data. Further experimental details are provided in Appendix B.

We note that several related works consider settings that are not directly comparable to our online behavior-policy search framework. In particular, Liu & Zhang (2024) study a fully offline setting with fixed transitions, where the behavior policy is computed from pre-logged data and cannot adapt to transition shifts at deployment. Other robust evaluation approaches (Katdare et al., 2023; Voloshin et al., 2021) focus on estimator robustness or model learning under transition uncertainty, rather than learning variance-minimizing behavior policies for data collection. We therefore include baselines (BPG and ROS) that explicitly target the same behavior-policy optimization objective as our method.

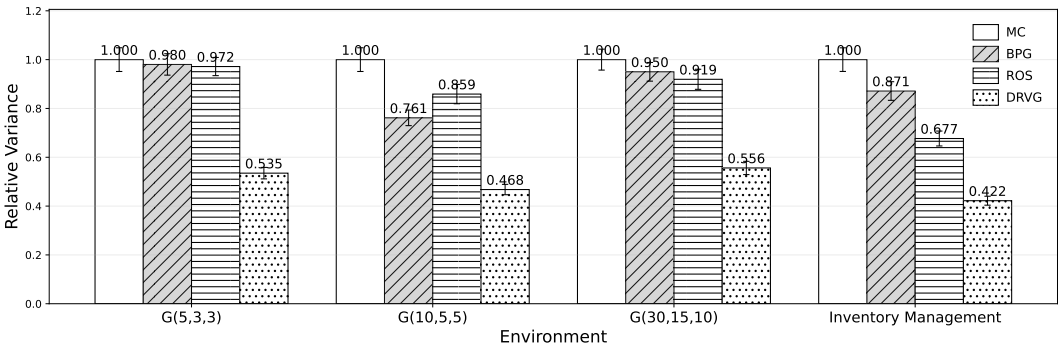


Figure 2: Relative variance of each method under the same perturbed transition. Values are normalized by the variance of the on-policy Monte Carlo (MC) method in the same environment. Error bars denote the standard error.

6.1 Variance Increase under Tailored Adversarial Transitions

To answer the first question, we examine the robustness of each behavior policy when exposed to its own most adversarial transition. For each method, we run Algorithm 1 to obtain the transition that maximizes its evaluation variance. Then, we use each behavior policy to collect data under its *method-specific adversarial transition*. We report the relative variance increase compared to the original simulator transition, highlighting each method's vulnerability to adversarial perturbations. As shown in Figure 1, our method exhibits the smallest variance increase, illustrating its robustness to adversarial transitions. Notably, although designed for variance reduction, BPG and ROS incur larger variance increases than the on-policy Monte Carlo baseline when the deployment transition is perturbed, underscoring the necessity of our robustness-aware behavior policy search framework.

6.2 Variance Comparison under Shared and Perturbed Transition

To address the second question, we evaluate all methods under a shared adversarial target transition identified by Algorithm 1 for the on-policy baseline. We compare the variance of all four methods under this *same perturbed transition*. This setup contrasts with Section 6.1, where each method faced its own tailored adversary. As shown in Figure 2, our method (DRVG) indeed achieves lower evaluation variance. This demonstrates that explicitly accounting for transition uncertainty enables more reliable policy evaluation under perturbed environments.

7 Conclusion

In this work, we present an efficient and robust behavior policy search framework that tackles two central challenges in real-world policy evaluation: variance reduction and transition mismatch. Our method learns variance-reducing behavior policies while explicitly accounting for transition uncertainty through a minimax formulation over adversarial dynamics. Theoretically, we derive novel transition-variance gradient expressions, establish convergence guarantees for the adversarial inner loop, and prove global convergence of our proposed double-loop algorithm. Numerically, our method demonstrates increased robustness under transition perturbations. Taken together, these results unify variance reduction with robustness to transition shifts, offering a promising step toward reliable policy evaluation under uncertainty.

AI Use Statement

Generative AI tools were used solely for grammar correction in the manuscript. They were not used for research ideation, methodology, theoretical development or proofs, code, experiments, data analysis, or interpretation of results. All AI-assisted edits were reviewed by the authors, who take responsibility for the final content.

REFERENCES

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. URL http://jmlr.org/papers/v22/19-736.html. 6, 7, 13

Thomas Welsh Archibald, KIM McKinnon, and Lyn C Thomas. On the generation of markov decision processes. *Journal of the Operational Research Society*, 46(3):354–361, 1995. 8, 28

Dimitri P Bertsekas and John N Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM Journal on Optimization*, 10(3):627–642, 2000. 15, 19

D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1995. 6

Jalaj Bhandari and Daniel Russo. On the linear convergence of policy gradient methods for finite mdps. In Arindam Banerjee and Kenji Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2386–2394. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/bhandari21a.html. 6, 7, 13

Semih Cayci, Niao He, and Rayadurgam Srikant. Convergence of entropy-regularized natural policy gradient with linear function approximation. *SIAM Journal on Optimization*, 34(3):2729–2755, 2024. 7

Claire Chen, Shuze Liu, and Shangtong Zhang. Efficient policy evaluation with safety constraint for reinforcement learning. In *Proceedings of the International Conference on Learning Representations*, 2025. 3

Josiah P Hanna, Philip S Thomas, Peter Stone, and Scott Niekum. Data-efficient policy evaluation through behavior policy search. In *Proceedings of the International Conference on Machine Learning*, 2017. 1, 2, 3, 4, 5, 7, 8, 13, 27

Josiah P Hanna, Yash Chandak, Philip S Thomas, Martha White, Peter Stone, and Scott Niekum. Data-efficient policy evaluation through behavior policy search. *Journal of Machine Learning Research*, 25(313):1–58, 2024. 4, 5, 7, 13, 23, 27

Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Fast bellman updates for robust mdps. In *International Conference on Machine Learning*, pp. 1979–1988. PMLR, 2018. 8, 28

Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Partial policy iteration for l1-robust markov decision processes. *Journal of Machine Learning Research*, 22(275):1–46, 2021. URL http://jmlr.org/papers/v22/20-445.html. 3, 6, 8

Garud N Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2): 257–280, 2005. 2, 3

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *International conference on machine learning*, pp. 4880–4889. PMLR, 2020. 3, 6

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 2021. 1

Pulkit Katdare, Nan Jiang, and Katherine Rose Driggs-Campbell. Marginalized importance sampling for off-environment policy evaluation. In *Conference on Robot Learning*, pp. 3778–3788. PMLR, 2023. 2, 3, 8

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015. 27

Matthieu Komorowski, Leo A Celi, Omar Badawi, Anthony C Gordon, and A Aldo Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature medicine*, 24(11):1716–1720, 2018. 29

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6083–6093. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/lin20a.html. 6

Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, 2018. 7

Shuze Liu and Shangtong Zhang. Efficient policy evaluation with offline data informed behavior policy design. In *Proceedings of the International Conference on Machine Learning*, 2024. 1, 2, 3, 8

Shuze Liu, Claire Chen, and Shangtong Zhang. Doubly optimal policy evaluation for reinforcement learning. In *Proceedings of the International Conference on Learning Representations*, 2025a. 1, 3

Shuze Liu, Yuxin Chen, and Shangtong Zhang. Efficient multi-policy evaluation for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025b. 3

Yao Liu, Pierre-Luc Bacon, and Emma Brunskill. Understanding the curse of horizon in off-policy evaluation via conditional importance sampling. In *International Conference on Machine Learning*, pp. 6184–6193. PMLR, 2020. 7

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 2015. 1

Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005. 2, 3

Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/25048eb6a33209cb5a815bff0cf6887c-Paper.pdf. 6

Evan L Porteus. *Foundations of stochastic inventory theory*. Stanford University Press, 2002. 8, 28

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014. 2

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 1951. 4

Alessio Russo and Aldo Pacchiano. Adaptive exploration for multi-reward multi-policy evaluation. *arXiv preprint arXiv:2502.02516*, 2025. 2

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017. 1

Zhongchang Sun, Sihong He, Fei Miao, and Shaofeng Zou. Policy optimization for robust average reward mdps. *Advances in Neural Information Processing Systems*, 37:17348–17372, 2024. 8

Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction (2nd Edition)*. MIT press, 2018. 28

Jean Tarbouriech and Alessandro Lazaric. Active exploration in markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 974–982. PMLR, 2019. 8, 28

Cameron Voloshin, Nan Jiang, and Yisong Yue. Minimax model learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 1612–1620. PMLR, 2021. 2, 3, 8

Qiuhao Wang, Chin Pang Ho, and Marek Petrik. Policy gradient in robust mdps with global convergence guarantee. In *International Conference on Machine Learning*, pp. 35763–35797. PMLR, 2023a. 2, 3, 6, 7, 8, 13, 21, 28

Qiuhao Wang, Shaohang Xu, Chin Pang Ho, and Marek Petrik. Policy gradient for robust markov decision processes. *arXiv preprint arXiv:2410.22114*, 2024. 2, 8

Yue Wang, Alvaro Velasquez, George Atia, Ashley Prater-Bennette, and Shaofeng Zou. Robust average-reward markov decision processes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 15215–15223, 2023b. 8, 28

Rui Yuan, Simon S Du, Robert M Gower, Alessandro Lazaric, and Lin Xiao. Linear convergence of natural policy gradient methods with log-linear policies. *arXiv preprint arXiv:2210.01400*, 2022. 7

Rujie Zhong, Duohan Zhang, Lukas Schäfer, Stefano V. Albrecht, and Josiah P. Hanna. Robust on-policy sampling for data-efficient policy evaluation in reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 8, 27

A PROOF

A.1 DEFINITIONS

In this section, we show the standard optimization definitions used in our work. Consider an optimization problem

$$\min_{x \in \mathcal{X}} f(x)$$

where $\mathcal{X} \subseteq \mathbb{R}^d$ is nonempty and closed, and $f : \mathbb{R}^d \to \mathbb{R}$. We have the following definitions for Lipschitz continuity and smoothness.

Definition A.1 (Lipschitz Continuity)**.** The function $f : \mathcal{X} \to \mathbb{R}$ is $L-$Lipschitz if $\forall x_1, x_2 \in \mathcal{X}$,

$$\|f(x_1) - f(x_2)\| \leq L\|x_1 - x_2\|.$$

Definition A.2 (Smoothness)**.** The function $f : \mathcal{X} \to \mathbb{R}$ is $\ell-$smooth if $\forall x_1, x_2 \in \mathcal{X}$,

$$\|\nabla f(x_1) - \nabla f(x_2)\| \leq \ell\|x_1 - x_2\|.$$

Our theoretical results are established within a standard analytical framework consistent with prior work in behavior-policy search and robust reinforcement learning (Hanna et al., 2017; Agarwal et al., 2021; Wang et al., 2023a; Hanna et al., 2024). To ensure the existence of well-defined gradients and the validity of our global convergence analysis, we consider environments and parameterizations that satisfy the following standard regularity conditions:

Smoothness of the Transition Model We consider a class of transition functions $p_\omega(s'|s, a)$ that are twice-differentiable with respect to their parameters ω. This is a standard property naturally satisfied by typical softmax or neural-network parameterizations with smooth activation functions (Agarwal et al., 2021; Wang et al., 2023a).

Compactness of the Parameter Spaces Consistent with established results in minimax optimization and projected gradient descent, the transition uncertainty set Ω and the behavior policy parameter space Θ are assumed to be compact and convex sets (Agarwal et al., 2021).

Finite Evaluation Variance To ensure the stability of the behavior-policy search, we assume the importance sampling ratios are uniformly bounded by a constant C. This is the standard condition for ensuring finite evaluation variance in off-policy evaluation (Hanna et al., 2017; 2024). In practice, this is satisfied by choosing a behavior policy parameter space Θ that keeps the data-collection policy bounded away from zero.

Bounded Feature Representations For the global convergence guarantees established in Section 5.2, we assume that the state features $\phi(s)$ are bounded (Agarwal et al., 2021; Bhandari & Russo, 2021).

With these definitions in hand, we are now ready to present the proofs.

A.2 PROOF OF THEOREM 4.1

Theorem 4.1 (Transition Gradient of the Variance)**.** *For a fixed behavior policy π_θ,*

$$\begin{aligned}
&\frac{\partial}{\partial \omega} \mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, H)] \\
=&\mathbb{E}_{H \sim p_\omega, \pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, H)^2 \frac{\partial}{\partial \omega} \ell_{p_\omega}\right] \\
-&2\mathbb{E}_{H \sim p_\omega, \pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, H)]\mathbb{E}_{H \sim p_\omega, \pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right].
\end{aligned}$$

Proof. To prove Theorem 4.1, we aim at decomposing the term $\Pr(H = h \mid p_\omega)$ into two parts: one that depends on p_ω and one that does not. By the standard trajectory factorization for a fixed

initial-state distribution p_0 and behavior policy π_θ,

$$\Pr(H = h \mid p_\omega) = p_0(S_0) \prod_{t=0}^{T-1} \pi_\theta(A_t|S_t) \prod_{t=0}^{T-1} p_\omega(S_{t+1}|S_t, A_t).$$

We separate the ω-dependent transition factor from the ω-independent factors by defining

$$m_{p_\omega}(h) \doteq \prod_{t=0}^{T-1} p_\omega(S_{t+1}|S_t, A_t) \tag{4}$$

and

$$p(h) \doteq p_0(S_0) \prod_{t=0}^{T-1} \pi_\theta(A_t|S_t). \tag{5}$$

Note that $p(h)$ is independent of ω since p_0 is the fixed initial-state distribution and π_θ does not depend on ω. We thus obtain the decomposition

$$\Pr(H = h \mid p_\omega) = p(h) m_{p_\omega}(h). \tag{6}$$

Next, we manipulate the term $\frac{\partial}{\partial \omega} m_{p_\omega}(h)$.

$$\begin{aligned}
&\frac{\partial}{\partial \omega} m_{p_\omega}(h) \\
=&\frac{\partial}{\partial \omega} \prod_{t=0}^{T-1} p_\omega(S_{t+1}|S_t, A_t) \\
=&\sum_{t=0}^{T-1} \left(\prod_{i \neq t} p_\omega(S_{i+1}|S_i, A_i) \frac{\partial p_\omega(S_{t+1}|S_t, A_t)}{\partial \omega} \right) \\
=&\sum_{t=0}^{T-1} \left(\frac{\prod_{i=0}^{T-1} p_\omega(S_{i+1}|S_i, A_i)}{p_\omega(S_{t+1}|S_t, A_t)} \cdot \frac{\partial p_\omega(S_{t+1}|S_t, A_t)}{\partial \omega} \right) \\
=&\prod_{i=0}^{T-1} p_\omega(S_{i+1}|S_i, A_i) \cdot \sum_{t=0}^{T-1} \left(\frac{1}{p_\omega(S_{t+1}|S_t, A_t)} \frac{\partial p_\omega(S_{t+1}|S_t, A_t)}{\partial \omega} \right) \\
\overset{(a)}{=}&\prod_{i=0}^{T-1} p_\omega(S_{i+1}|S_i, A_i) \cdot \sum_{t=0}^{T-1} \left(\frac{1}{p_\omega(S_{t+1}|S_t, A_t)} p_\omega(S_{t+1}|S_t, A_t) \frac{\partial \log p_\omega(S_{t+1}|S_t, A_t)}{\partial \omega} \right) \\
=&\prod_{i=0}^{T-1} p_\omega(S_{i+1}|S_i, A_i) \sum_{t=0}^{T-1} \left(\frac{\partial}{\partial \omega} \log p_\omega(S_{t+1}|S_t, A_t) \right) \\
=&m_{p_\omega}(h) \sum_{t=0}^{T-1} \left(\frac{\partial}{\partial \omega} \log p_\omega(S_{t+1}|S_t, A_t) \right) \\
=&m_{p_\omega}(h) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t))
\end{aligned} \tag{7}$$

Here, (a) follows from the fact that

$$\begin{aligned}
\frac{\partial}{\partial x} \log f(x) &= \frac{1}{f(x)} \frac{\partial f(x)}{\partial x} \\
\implies \frac{\partial f(x)}{\partial x} &= f(x) \cdot \frac{\partial \log f(x)}{\partial x}.
\end{aligned}$$

Then, we decompose the variance objective

$$
\begin{aligned}
&\frac{\partial}{\partial\omega}\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\\
=&\frac{\partial}{\partial\omega}\left(\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)^2]-\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,h)]^2\right)\\
=&\frac{\partial}{\partial\omega}\sum_h \Pr(H=h|p_\omega)\mathrm{OPE}(\pi_e,\pi_\theta,h)^2\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\frac{\partial}{\partial\omega}\sum_h \Pr(H=h|p_\omega)\mathrm{OPE}(\pi_e,\pi_\theta,h)\\
=&\sum_h p(h)\mathrm{OPE}(\pi_e,\pi_\theta,h)^2\frac{\partial}{\partial\omega}m_{p_\omega}(h)\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\sum_h p(h)\mathrm{OPE}(\pi_e,\pi_\theta,h)\frac{\partial}{\partial\omega}m_{p_\omega}(h) \qquad \text{(By (6))}
\end{aligned}
$$

$$
\begin{aligned}
=&\sum_h p(h)\mathrm{OPE}(\pi_e,\pi_\theta,h)^2 m_{p_\omega}(h)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\sum_h p(h)\mathrm{OPE}(\pi_e,\pi_\theta,h)m_{p_\omega}(h)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(By (7))}\\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))]\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right].
\end{aligned}
$$

□

A.3 Proof of Lemma 4.2

Lemma 4.2 (Transition Gradient Convergence)**.** *For a fixed behavior policy π_θ, Algorithm 1 converges. That is, $\mathbb{V}_{H_i\sim p_{\omega_i},\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]$ converges to a finite value and $\lim_{i\to\infty}\frac{\partial}{\partial\omega}\mathbb{V}_{H_i\sim p_{\omega_i},\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]=0$.*

Proof. The proof leverages Proposition 3 in Bertsekas & Tsitsiklis (2000), for which we have to show that Algorithm 1 satisfies the following conditions:

1. $\mathbb{V}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]$ is continuously differentiable w.r.t. ω.
2. The gradient of the variance objectives, $\frac{\partial}{\partial\omega}\mathbb{V}[\mathrm{IS}(\pi_e,\pi_\theta,H_i)]$, is Lipschitz continuous w.r.t. ω.
3. The variance of the gradient estimate used by Algorithm 1 is bounded.

The other conditions of Proposition 3 in Bertsekas & Tsitsiklis (2000) are satisfied because of the unbiasedness of the gradient estimates in Algorithm 1. Additionally, since V is bounded above, equivalently $-V$ is bounded below, the divergence case in Proposition 3 applied to $-V$ is excluded (Bertsekas & Tsitsiklis, 2000).

By assumptions, we have p_ω is twice-differentiable, and quotient $\frac{w_{\pi_e}}{w_{\pi_\theta}}$ and the estimator $\mathrm{IS}(\pi_e,\pi_\theta,H)$ always exist. Therefore, by the gradient expression in Lemma 4.1, we conclude that $\frac{\partial}{\partial\omega}V_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]$ is continuously differentiable, verifying condition 1.

Next, we show the Lipschitz continuity of $\frac{\partial}{\partial\omega}V_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]$ by verifying the boundedness of its second derivative.

$$
\begin{aligned}
&\frac{\partial^2}{\partial\omega^2}\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]\\
=&\frac{\partial}{\partial\omega}\Bigg(\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right]\\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{IS}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right]\Bigg)\\
=&\frac{\partial}{\partial\omega}\left(\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right)\right.\\
&-2\sum_h(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\\
&\left.\cdot\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right)\right)\quad\text{(By Lemma 4.1 and (6))}
\end{aligned}
$$

$$
\begin{aligned}
=&\frac{\partial}{\partial\omega}\left(\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}\log m_{p_\omega}(h)\right)\right.\\
&\left.-2\sum_h(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}\log m_{p_\omega}(h)\right)\right)\\
=&\frac{\partial}{\partial\omega}\left(\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)^2\frac{1}{m_{p_\omega}(h)}\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right.\\
&\left.-2\sum_h(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{1}{m_{p_\omega}(h)}\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right)\\
=&\frac{\partial}{\partial\omega}\left(\sum_h\left(p(h)\mathrm{IS}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right.\\
&\left.-2\sum_h(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right)\\
=&\sum_h p(h)\left(\underbrace{\mathrm{IS}(\pi_e,\pi_\theta,H)^2}_{(1)}\underbrace{\frac{\partial^2}{\partial\omega^2}m_{p_\omega}(h)}_{(2)}\right)\\
&-2\frac{\partial}{\partial\omega}\left[\sum_h(p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right].
\end{aligned}
$$

We further decompose the term in the square brackets.

$$
\begin{aligned}
&\frac{\partial}{\partial\omega}\left[\sum_h (p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\right]\\
=&\sum_h p(h)\frac{\partial}{\partial\omega}(m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h\left(p(h)\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\\
&+\sum_h (p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H))\cdot\sum_h p(h)\frac{\partial}{\partial\omega}\left(\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\\
=&\sum_h p(h)\left(\underbrace{\mathrm{IS}(\pi_e,\pi_\theta,H)}_{(3)}\underbrace{\frac{\partial}{\partial\omega}m_{p_\omega}(h)}_{(4)}\right)\cdot\sum_h p(h)\left(\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\\
&+\sum_h p(h)\left(\underbrace{m_{p_\omega}(h)}_{(5)}\mathrm{IS}(\pi_e,\pi_\theta,H)\right)\cdot\sum_h p(h)\left(\mathrm{IS}(\pi_e,\pi_\theta,H)\frac{\partial^2}{\partial\omega^2}m_{p_\omega}(h)\right).
\end{aligned}
$$

Notice that since $p(h)=p_0(S_0)\prod_{t=0}^{T-1}\pi_\theta(A_t|S_t)\le 1$ by (5), $p(h)$ is bounded above. We then analyze the boundedness of $\frac{\partial^2}{\partial\omega^2}V_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]$ through the above 5 terms.

For (1) and (3), the quotient $\frac{\pi_e(a|s)}{\pi_\theta(a|s)}$ is bounded above by assumption. Besides, since the reward is bounded, so is $g(h)$. Therefore, both (1), $\mathrm{IS}(\pi_e,\pi_\theta,H)^2$ and (3) $\mathrm{IS}(\pi_e,\pi_\theta,H)$ are bounded.

For (5), it is bounded because $m_{p_\omega}(h)=\prod_{t=0}^{T-1}p_\omega(S_{t+1}|S_t,A_t)\le 1$. Then, for (4),

$$
\begin{aligned}
\frac{\partial}{\partial\omega}m_{p_\omega}(h)=&\frac{\partial}{\partial\omega}\prod_{t=0}^{T-1}p_\omega(S_{t+1}|S_t,A_t)\\
=&\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}p_\omega(S_{t+1}|S_t,A_t)\frac{\prod_{i=0}^{T-1}p_\omega(S_{i+1}|S_i,A_i)}{p_\omega(S_{t+1}|S_t,A_t)}.
\end{aligned}
$$

Here, $\frac{\partial}{\partial\omega}p_\omega(S_{t+1}|S_t,A_t)$ is bounded by construction and $\frac{\prod_{i=0}^{T-1}p_\omega(S_{i+1}|S_i,A_i)}{p_\omega(S_{t+1}|S_t,A_t)}\le 1$. Thus, (4) is bounded. Lastly, for (2)

$$
\begin{aligned}
&\frac{\partial^2}{\partial\omega^2}m_{p_\omega}(h)\\
=&\frac{\partial}{\partial\omega}\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}p_\omega(S_{t+1}|S_t,A_t)\frac{\prod_{i=0}^{T-1}p_\omega(S_{i+1}|S_i,A_i)}{p_\omega(S_{t+1}|S_t,A_t)}\\
=&\frac{\partial}{\partial\omega}\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}p_\omega(S_{t+1}|S_t,A_t)\prod_{i\neq t}p_\omega(S_{i+1}|S_i,A_i)\\
=&\sum_{t=0}^{T-1}\Big[\frac{\partial^2}{\partial\omega^2}p_\omega(S_{t+1}|S_t,A_t)\prod_{i\neq t}p_\omega(S_{i+1}|S_i,A_i)+\frac{\partial}{\partial\omega}p_\omega(S_{t+1}|S_t,A_t)\\
&\cdot\sum_{i\neq t}\frac{\partial}{\partial\omega}p_\omega(S_{i+1}|S_i,A_i)\prod_{j\neq t,i}p_\omega(S_{j+1}|S_j,A_j)\Big],
\end{aligned}
$$

which is bounded because p_ω is constructed to be twice differentiable with bounded first and second derivatives.

Therefore, we conclude that the gradient objective $\frac{\partial}{\partial\omega}V_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]$ is Lipschitz continuous w.r.t. ω, verifying condition 2.

Finally, we show that the variance of the gradient estimate used by Algorithm 1 is bounded. According to Algorithm 1, we use the unbiased estimate as

$$\begin{aligned}
&\frac{\partial}{\partial\omega} V_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)] \\
\approx &\underbrace{\mathrm{IS}(\pi_e,\pi_\theta,H)^2 \sum_{t=0}^{T-1} \frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))}_{A} \\
&\underbrace{-2\mathrm{IS}(\pi_e,\pi_\theta,H')\mathrm{IS}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))}_{B}.
\end{aligned}$$

Here H and H' are independent trajectories from (p_ω,π_θ). We work coordinatewise for the variance calculation; the boundedness of these terms also bounds their batch averages. Then, the variance of the estimate is decomposed into

$$\mathbb{V}[A]+\mathbb{V}[B]-2\mathrm{Cov}[A,B],$$

where $|\mathrm{Cov}[A,B]| \le \sqrt{\mathbb{V}[A]}\cdot\sqrt{\mathbb{V}[B]}$ by the Cauchy-Schwarz inequality. Thus, it is sufficient to show the boundedness of $\mathbb{V}[A]$ and $\mathbb{V}[B]$. For $\mathbb{V}[A]$, since the variance of a bounded random variable is bounded, we aim to demonstrate that for any trajectory h, the term $\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))$ is bounded.

$$\begin{aligned}
&\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t)) \\
=&\mathrm{IS}(\pi_e,\pi_\theta,H)^2\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t)) \\
=&\mathrm{IS}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}\log m_{p_\omega}(h) \\
=&\mathrm{IS}(\pi_e,\pi_\theta,H)^2\frac{\frac{\partial}{\partial\omega}m_{p_\omega}(h)}{m_{p_\omega}(h)}.
\end{aligned} \tag{8}$$

The boundedness of $\mathrm{IS}(\pi_e,\pi_\theta,H)^2$ and $\frac{\partial}{\partial\omega}m_{p_\omega}(h)$ is shown by the argument above for terms (3) and (4). For the boundedness of $\frac{1}{m_{p_\omega}(h)} = \frac{1}{\prod_{t=0}^{T-1}p_\omega(S_{t+1}|S_t,A_t)}$, we invoke the Extreme Value Theorem. Since $p_\omega(s'|s,a)$ is strictly positive (under softmax parameterization) and continuous on the finite state-action space $\mathcal{S}\times\mathcal{A}\times\mathcal{S}$ and the compact parameter space Ω (compactness of Ω is assumed in our background), the Extreme Value Theorem ensures that p_ω attains a strictly positive minimum $c \doteq \min_{s,a,s',\omega} p_\omega(s'|s,a) > 0$. Since the trajectory length T is finite, we have $\frac{1}{m_{p_\omega}(h)} \le (1/c)^T < \infty$ uniformly in h and ω. Thus, we conclude that $\mathbb{V}[A]$ is bounded.

Next, we decompose term B into two parts because of the different samples used to estimate them:

$$B = 2\underbrace{\mathrm{IS}(\pi_e,\pi_\theta,H')}_{C}\underbrace{\mathrm{IS}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))}_{D}.$$

Since in Algorithm 1 the cross-product estimator is computed by sample splitting (the first $k/2$ trajectories estimate C and the second $k/2$ trajectories estimate D), C and D are independent. Consequently, by the standard variance identity for products of independent random variables,

$$\mathbb{V}[B] = 4\mathbb{V}[CD] = 4\big(\mathbb{E}[C^2]\mathbb{V}[D]+\mathbb{E}[D]^2\mathbb{V}[C]\big).$$

We show their boundedness term by term.

$$\mathbb{E}[C^2] = \mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)^2] = \sum_h p(h)m_{p_\omega}(h)\mathrm{IS}(\pi_e,\pi_\theta,H)^2, \tag{9}$$

where each term is shown to be bounded above. Next, by Jensen's inequality and the derivation from (8),

$$\mathbb{E}[D]^2 \leq \mathbb{E}[D^2] = \sum_h p(h) m_{p_\omega}(h) \mathrm{IS}(\pi_e, \pi_\theta, H)^2 \left(\frac{\frac{\partial}{\partial \omega} m_{p_\omega}(h)}{m_{p_\omega}(h)} \right)^2,$$

where the boundedness of the right-hand side follows from the analysis of (9) and (8), and hence $\mathbb{E}[D]^2$ is bounded as well.

As for the two variance terms, $\mathbb{V}[C]$ and $\mathbb{V}[D]$, we show the boundedness of the random variable C and D for each trajectory h, where $\mathrm{IS}(\pi_e, \pi_\theta, H)$ is shown to be bounded in term (3) above, and the boundedness of $\sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t))$ is incorporated in (8).

Therefore, we conclude that the variance of our estimate is bounded. By far, we show that the three conditions of Proposition 3 in Bertsekas & Tsitsiklis (2000) are satisfied, demonstrating the convergence of Algorithm 1.

□

A.4 Proof of Theorem 4.3

Proof.

Theorem 4.3 (Transition Gradient of Variance with KL)**.** *For a fixed behavior policy π_θ and a regularization coefficient $\eta > 0$,*

$$\begin{aligned}
&\frac{\partial}{\partial \omega}\Big[\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)] - \eta \mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0}))\Big] \\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)^2\frac{\partial}{\partial\omega}\ell_{p_\omega}\right] \\
&-2\mathbb{E}_{H\sim p_\omega,\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,H)]\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\mathrm{OPE}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right] \\
&-\eta\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\left(\frac{\partial}{\partial\omega}\ell_{p_\omega}\right)\left(1+\ell_{p_\omega}-\ell_{p_{\omega_0}}\right)\right].
\end{aligned}$$

We begin by manipulating the KL-divergence term.

$$\begin{aligned}
\mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0})) =&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\log\frac{\Pr(H|p_\omega)}{\Pr(H|p_{\omega_0})}\right] \\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\log\frac{m_{p_\omega}(H)}{m_{p_{\omega_0}}(H)}\right] && \text{(By (6))} \\
=&\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\log m_{p_\omega}(H) - \log m_{p_{\omega_0}}(H)\right].
\end{aligned}$$

Next, we decompose the following gradient:

$$\begin{aligned}
&\frac{\partial}{\partial\omega}\log m_{p_\omega}(H) && (10)\\
=&\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log p_\omega(S_{t+1}|S_t,A_t) \\
=&\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t)). && \text{(By definition)}
\end{aligned}$$

Then, we take the gradient of the KL-divergence with respect to ω:

$$\begin{aligned}
&\frac{\partial}{\partial\omega}\mathrm{KL}(\Pr(H|p_\omega)\|\Pr(H|p_{\omega_0})) && (11)\\
=&\frac{\partial}{\partial\omega}\mathbb{E}_{H\sim p_\omega,\pi_\theta}\left[\log m_{p_\omega}(H) - \log m_{p_{\omega_0}}(H)\right] \\
=&\frac{\partial}{\partial\omega}\sum_h \Pr(H=h|p_\omega)\left[\log m_{p_\omega}(h) - \log m_{p_{\omega_0}}(h)\right] \\
=&\frac{\partial}{\partial\omega}\sum_h p(h)m_{p_\omega}(h)\left[\log m_{p_\omega}(h) - \log m_{p_{\omega_0}}(h)\right] && \text{(By (6))} \\
=&\sum_h p(h)\left[\frac{\partial}{\partial\omega}m_{p_\omega}(h)\log m_{p_\omega}(h) + m_{p_\omega}(h)\frac{\partial}{\partial\omega}\log m_{p_\omega}(h) - \log m_{p_{\omega_0}}(h)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right] \\
=&\sum_h p(h)\left[\log m_{p_\omega}(h)\frac{\partial}{\partial\omega}m_{p_\omega}(h) + m_{p_\omega}(h)\frac{\partial}{\partial\omega}\log m_{p_\omega}(h)\right. \\
&\left.-\log m_{p_{\omega_0}}(h)m_{p_\omega}(h)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right] && \text{(By (7))}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_h p(h) \Bigg[\log m_{p_\omega}(h) m_{p_\omega}(h) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \\
&\quad + m_{p_\omega}(h) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \\
&\quad - \log m_{p_{\omega_0}}(h) m_{p_\omega}(h) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \Bigg] && \text{(By (7) (10))} \\
&= \sum_h p(h) m_{p_\omega}(h) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \left[\log m_{p_\omega}(h) + 1 - \log m_{p_{\omega_0}}(h) \right] \\
&= \sum_h \Pr(H = h|p_\omega) \sum_{t=0}^{T-1} \left[\frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \right] \left[\log m_{p_\omega}(h) + 1 - \log m_{p_{\omega_0}}(h) \right] && \text{(By (6))} \\
&= \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\left(\frac{\partial}{\partial \omega} \ell_{p_\omega} \right) \left(1 + \ell_{p_\omega} - \ell_{p_{\omega_0}} \right) \right]. && \text{(By (4))}
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\frac{\partial}{\partial \omega} \left[\mathbb{V}_{H \sim p_\omega, \pi_\theta} \left[\mathrm{OPE}(\pi_e, \pi_\theta, H) \right] - \eta \mathrm{KL}(\Pr(H|p_\omega) \| \Pr(H|p_{\omega_0})) \right] \\
&= \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\mathrm{OPE}(\pi_e, \pi_\theta, H)^2 \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \right] - 2 \mathbb{E}_{H \sim p_\omega, \pi_\theta} [\mathrm{OPE}(\pi_e, \pi_\theta, H)] \\
&\quad \cdot \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\mathrm{OPE}(\pi_e, \pi_\theta, H) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \right] \\
&\quad - \eta \frac{\partial}{\partial \omega} \mathrm{KL}(\Pr(H|p_\omega) \| \Pr(H|p_{\omega_0})) && \text{(By Lemma 4.1)} \\
&= \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\mathrm{OPE}(\pi_e, \pi_\theta, H)^2 \frac{\partial}{\partial \omega} \ell_{p_\omega} \right] \\
&\quad - 2 \mathbb{E}_{H \sim p_\omega, \pi_\theta} [\mathrm{OPE}(\pi_e, \pi_\theta, H)] \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\mathrm{OPE}(\pi_e, \pi_\theta, H) \frac{\partial}{\partial \omega} \ell_{p_\omega} \right] \\
&\quad - \eta \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[\left(\frac{\partial}{\partial \omega} \ell_{p_\omega} \right) \left(1 + \ell_{p_\omega} - \ell_{p_{\omega_0}} \right) \right]. && \text{(By (11))}
\end{aligned}
$$

□

A.5 Proof of Lemma 5.2

Proof. By Lemma 5.1,

$$
\frac{\partial}{\partial \theta} \mathbb{V}_{H \sim p_\omega, \pi_\theta} [\mathrm{IS}(\pi_e, \pi_\theta, H)] = \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[-\mathrm{IS}(\pi_e, \pi_\theta, H)^2 \sum_{t=0}^{T-1} \frac{\partial}{\partial \theta} \log \pi_\theta(A_t|S_t) \right].
$$

To prove the Lipschitz property, we bound each term in the RHS. First, we aim to bound $\left\| \frac{\partial}{\partial \theta} \log \pi_\theta(A_t|S_t) \right\|$. Remember that we define

$$
\pi_\theta(a|s) \doteq \frac{\exp\left(\theta_a^\top \phi(s)\right)}{\sum_{a' \in \mathcal{A}} \exp\left(\theta_{a'}^\top \phi(s)\right)},
$$

where we assumed the linear features $\|\phi(s)\|$ to be bounded by a constant B. Here, $\theta = \{\theta_a\}_{a \in \mathcal{A}}$ is the whole parameter matrix, and θ_a is the column for action a specifically. From Wang et al. (2023a),

we know that

$$\left\| \frac{\partial}{\partial \theta} \log \pi_\theta(a|s) \right\|^2 = \sum_{a' \in \mathcal{A}} \left\| \frac{\partial}{\partial \theta_{a'}} \log \pi_\theta(a|s) \right\|^2.$$

Further decomposing, we get

$$\begin{aligned} \left\| \frac{\partial}{\partial \theta} \log \pi_\theta(a|s) \right\| =& \left[\|\phi(s)\|_2^2 \left(1 - 2\pi_\theta(a|s) + \sum_{a' \in \mathcal{A}} \pi_\theta(a'|s)^2 \right) \right]^{\frac{1}{2}} \\ \leq& \sqrt{2} B. \end{aligned}$$

Thus, we have

$$\left\| \sum_{t=0}^{T-1} \frac{\partial}{\partial \theta} \log \pi_\theta(A_t|S_t) \right\| \leq \sqrt{2} B T. \tag{12}$$

We also make the standard assumption that the quotient $\frac{\pi_e(a|s)}{\pi_\theta(a|s)}$ is bounded above by a constant C for all s, a, and θ.

$$\left\| \mathrm{IS}(\pi_e, \pi_\theta, H)^2 \right\| = \left\| \left(\frac{\prod_{t=0}^{T-1} \pi_e(A_t|S_t)}{\prod_{t=0}^{T-1} \pi_\theta(A_t|S_t)} g(H) \right)^2 \right\| \leq C^{2T} T^2, \tag{13}$$

since we assume the reward is bounded above by 1.

Then,

$$\left\| \frac{\partial}{\partial \theta} V_{H \sim p_\omega, \pi_\theta} [\mathrm{IS}(\pi_e, \pi_\theta, H)] \right\| \leq \sqrt{2} B C^{2T} T^3.$$

Thus, the objective function $\mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is L_Θ-Lipschitz in θ with $L_\Theta = \sqrt{2} B C^{2T} T^3$.

Next, we aim to show that the objective function $\mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is ℓ_Θ-smooth in θ.

For fixed ω, write $V(\theta, \omega) \doteq \mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ and define

$$\begin{aligned} w_{\pi_\theta}(h) \doteq \prod_{t=0}^{T-1} \pi_\theta(A_t|S_t), \qquad \tilde{p}(h) \doteq \frac{\Pr(H = h \mid p_\omega, \pi_\theta)}{w_{\pi_\theta}(h)}, \\ X(H) \doteq \mathrm{IS}(\pi_e, \pi_\theta, H), \qquad z(H) \doteq \nabla_\theta \log w_{\pi_\theta}(H), \qquad J(H) \doteq \nabla_\theta^2 \log w_{\pi_\theta}(H). \end{aligned}$$

The transition factor $\tilde{p}(h)$ is independent of θ; the normalized weights are $\tilde{p}(h) w_{\pi_\theta}(h)$, so

$$\sum_h \tilde{p}(h) w_{\pi_\theta}(h) = 1.$$

The IS mean is independent of the behavior-policy parameter. Therefore, writing that mean as $m(\omega)$,

$$V(\theta, \omega) = \sum_h \frac{\tilde{p}(h) w_{\pi_e}(h)^2 g(h)^2}{w_{\pi_\theta}(h)} - m(\omega)^2.$$

Since $\nabla_\theta(w_{\pi_\theta}^{-1}) = -w_{\pi_\theta}^{-1} z$ and $\nabla_\theta^2(w_{\pi_\theta}^{-1}) = w_{\pi_\theta}^{-1}(zz^\top - J)$, differentiation gives

$$\nabla_\theta^2 V(\theta, \omega) = \mathbb{E}_{H \sim p_\omega, \pi_\theta} \left[X(H)^2 \big(z(H) z(H)^\top - J(H) \big) \right]. \tag{14}$$

For completeness, let $\theta_\alpha = \theta + \alpha u$, $v_s(a) = \langle u_a, \phi(s) \rangle$, and $\mu_s = \sum_b \pi_\theta(b|s) v_s(b)$. The softmax directional second derivative is

$$\left. \frac{d^2}{d\alpha^2} \pi_{\theta_\alpha}(a|s) \right|_{\alpha=0} = \pi_\theta(a|s) \left[(v_s(a) - \mu_s)^2 - \sum_b \pi_\theta(b|s) (v_s(b) - \mu_s)^2 \right].$$

In particular, a common shift of all logits gives zero. For the log probability,

$$u^\top \nabla_\theta^2 \log \pi_\theta(a|s) u = -\sum_b \pi_\theta(b|s)(v_s(b) - \mu_s)^2 \le 0,$$
$$\left|u^\top \nabla_\theta^2 \log \pi_\theta(a|s) u\right| \le \sum_b \pi_\theta(b|s) v_s(b)^2 \le \|v_s\|_2^2 \le B^2 \|u\|_2^2.$$

Summing over the T time steps and using (12) and (13), we obtain

$$J(H) \preceq 0, \qquad \|J(H)\|_{\mathrm{op}} \le B^2 T, \qquad \|z(H)\| \le \sqrt{2} BT, \qquad X(H)^2 \le C^{2T} T^2.$$

Applying the operator norm to (14) yields

$$\begin{aligned}\|\nabla_\theta^2 V(\theta, \omega)\|_{\mathrm{op}} &\le \mathbb{E}_{H \sim p_\omega, \pi_\theta}\left[X(H)^2 \left(\|z(H)\|^2 + \|J(H)\|_{\mathrm{op}}\right)\right] \\ &\le B^2 C^{2T} T^3 (1 + 2T) \le B^2 C^{2T} T^3 (5 + 8T).\end{aligned}$$

Thus, the stated smoothness constant $\ell_\Theta = B^2 C^{2T} T^3 (5 + 8T)$ remains valid. Lastly, $zz^\top - J \succeq 0$ in (14), so $V(\cdot, \omega)$ is convex, consistently with Lemma 2 of Hanna et al. (2024).

□

A.6 Proof of Lemma 5.3

Proof. We first show that $\Phi(\theta)$ is L_Θ-Lipschitz in θ. By Lemma 5.2, we know that $\mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is L_Θ-Lipschitz.
$\forall \theta_1, \theta_2 \in \Theta$, define $p_{\omega_1} \doteq \operatorname{argmax}_{p_\omega} \mathbb{V}_{H \sim p_\omega, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)]$, $p_{\omega_2} \doteq \operatorname{argmax}_{p_\omega} \mathbb{V}_{H \sim p_\omega, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)]$. Then,

$$\begin{aligned}\Phi(\theta_1) - \Phi(\theta_2) =& \max_{p_\omega} \mathbb{V}_{H \sim p_\omega, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)] - \max_{p_\omega} \mathbb{V}_{H \sim p_\omega, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)] \\ =& \mathbb{V}_{H \sim p_{\omega_1}, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)] - \mathbb{V}_{H \sim p_{\omega_2}, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)] \\ \le& \mathbb{V}_{H \sim p_{\omega_1}, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)] - \mathbb{V}_{H \sim p_{\omega_1}, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)] \\ \le& L_\Theta \|\theta_1 - \theta_2\|. \qquad \text{(By Lemma 5.2)}\end{aligned}$$

By symmetry, we also have

$$\Phi(\theta_2) - \Phi(\theta_1) \le L_\Theta \|\theta_1 - \theta_2\|.$$

Thus,

$$|\Phi(\theta_1) - \Phi(\theta_2)| \le L_\Theta \|\theta_1 - \theta_2\|,$$

which shows the Lipschitz property.
Next, from Lemma 5.2, we also know that $\mathbb{V}_{H \sim p_\omega, \pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is convex in θ under the linear softmax parameterization of the behavior policy π_θ. Thus, $\forall \theta_1, \theta_2 \in \Theta$ and $t \in [0, 1]$,

$$\begin{aligned}&\Phi(t\theta_1 + (1-t)\theta_2) \\ =& \max_{p_\omega} \mathbb{V}_{H \sim p_\omega, \pi_{(t\theta_1 + (1-t)\theta_2)}}[\mathrm{IS}(\pi_e, \pi_{(t\theta_1 + (1-t)\theta_2)}, H)] \\ \le& \max_{p_\omega} [t \mathbb{V}_{H \sim p_\omega, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)] + (1-t) \mathbb{V}_{H \sim p_\omega, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)]] \qquad \text{(By Lemma 5.2)} \\ \le& t \max_{p_\omega} [\mathbb{V}_{H \sim p_\omega, \pi_{\theta_1}}[\mathrm{IS}(\pi_e, \pi_{\theta_1}, H)] + (1-t) \max_{p_{\omega'}} \mathbb{V}_{H \sim p_{\omega'}, \pi_{\theta_2}}[\mathrm{IS}(\pi_e, \pi_{\theta_2}, H)] \\ =& t\Phi(\theta_1) + (1-t)\Phi(\theta_2).\end{aligned}$$

Therefore, we show that $\Phi(\theta)$ is convex in θ. □

A.7 Proof of Theorem 5.4

Exact-gradient case.

Proof. To begin with, we define $\theta^* \doteq \operatorname{argmin}_{\theta\in\Theta} \Phi(\theta)$. Since the set Θ is closed and convex, the Euclidean projection is nonexpansive. That is, $\forall u \in \mathbb{R}^d, z \in \Theta$,

$$\|\mathrm{Proj}_\Theta(u) - z\|^2 \le \|u - z\|^2.$$

With $u \doteq \theta_i - \alpha\mathcal{G}_i$, $z \doteq \theta^*$, we have

$$\mathrm{Proj}_\Theta(u) = \theta_{i+1}.$$

Thus,

$$\begin{aligned}\|\theta_{i+1} - \theta^*\|^2 &\le \|\theta_i - \alpha\mathcal{G}_i - \theta^*\|^2 \\ &= \|\theta_i - \theta^*\|^2 - 2\alpha\langle\mathcal{G}_i, \theta_i - \theta^*\rangle + \alpha^2\|\mathcal{G}_i\|^2. \end{aligned} \tag{15}$$

From here, we first bound the last term, $\|\mathcal{G}_i\|^2$. By Lemma 5.2 we know that the objective function $\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is L_Θ-Lipschitz and convex in θ. Thus we have

$$\|\mathcal{G}_i\| \le L_\Theta \implies \|\mathcal{G}_i\|^2 \le L_\Theta^2. \tag{16}$$

Next, since the gradient objective $\mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)]$ is differentiable and convex in θ, we have the subgradient inequality that

$$\left\langle \mathcal{G}_i, \theta_i - \theta^* \right\rangle \ge \mathbb{V}_{H\sim p_{\omega_i},\pi_{\theta_i}}[\mathrm{IS}(\pi_e, \pi_{\theta_i}, H)] - \mathbb{V}_{H\sim p_{\omega_i},\pi_{\theta^*}}[\mathrm{IS}(\pi_e, \pi_{\theta^*}, H)].$$

Remember that we defined

$$\Phi(\theta) \doteq \max_{p_\omega} \mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e, \pi_\theta, H)].$$

Thus,

$$\mathbb{V}_{H\sim p_{\omega_i},\pi_{\theta^*}}[\mathrm{IS}(\pi_e, \pi_{\theta^*}, H)] \le \Phi(\theta^*),$$

and by Algorithm 2,

$$\max_{p_\omega} \mathbb{V}_{H\sim p_\omega,\pi_{\theta_i}}[\mathrm{IS}(\pi_e, \pi_{\theta_i}, H)] = \Phi(\theta_i) \le \mathbb{V}_{H\sim p_{\omega_i},\pi_{\theta_i}}[\mathrm{IS}(\pi_e, \pi_{\theta_i}, H)] + \epsilon_i.$$

Therefore,

$$\left\langle \mathcal{G}_i, \theta_i - \theta^* \right\rangle \ge \Phi(\theta_i) - \epsilon_i - \Phi(\theta^*). \tag{17}$$

Putting it all together, by (15), (16), and (17), we have

$$\|\theta_{i+1} - \theta^*\|^2 \le \|\theta_i - \theta^*\|^2 - 2\alpha(\Phi(\theta_i) - \epsilon_i - \Phi(\theta^*)) + \alpha^2 L_\Theta^2.$$

Rearranging the terms, we get

$$2\alpha[\Phi(\theta_i) - \Phi(\theta^*)] \le \|\theta_i - \theta^*\|^2 - \|\theta_{i+1} - \theta^*\|^2 + 2\alpha\epsilon_i + \alpha^2 L_\Theta^2.$$

Taking the summation over i,

$$2\alpha \sum_{i=0}^{n-1} \big[\Phi(\theta_i) - \Phi(\theta^*)\big] \le \|\theta_0 - \theta^*\|^2 + 2\alpha \sum_{i=0}^{n-1} \epsilon_i + n\alpha^2 L_\Theta^2.$$

Since $\theta_0, \theta^* \in \Theta$, and we defined $\mathrm{diam}(\Theta) \le D$ where

$$\mathrm{diam}(\Theta) \doteq \max_{\theta,\theta'\in\Theta} \|\theta - \theta'\|,$$

we have

$$\|\theta_0 - \theta^*\|^2 \le D^2.$$

Thus,

$$\frac{1}{n}\sum_{i=0}^{n-1} \Phi(\theta_i) - \Phi(\theta^*) \le \frac{D^2}{2\alpha n} + \frac{\alpha L_\Theta^2}{2} + \frac{1}{n}\sum_{i=0}^{n-1} \epsilon_i. \tag{18}$$

According to Algorithm 2,

$$\bar{\theta} \doteq \frac{1}{n}\sum_{i=0}^{n-1}\theta_i.$$

By Lemma 5.3, $\Phi(\theta)$ is convex in θ. Thus, by induction with the basic convex property, with the nonnegative weight $\frac{1}{n}$ and the fact that $\sum_{i=0}^{n-1}(1/n)=1$, we obtain

$$\Phi(\bar{\theta}) = \Phi\left(\frac{1}{n}\sum_{i=0}^{n-1}\theta_i\right) \le \frac{1}{n}\sum_{i=0}^{n-1}\Phi(\theta_i).$$

Subtracting $\Phi(\theta^*)$ from both sides, we get

$$\Phi(\bar{\theta}) - \Phi(\theta^*) \le \frac{1}{n}\sum_{i=0}^{n-1}\Phi(\theta_i) - \Phi(\theta^*).$$

Plugging it into (18),

$$\Phi(\bar{\theta}) - \Phi(\theta^*) \le \frac{D^2}{2\alpha n} + \frac{\alpha L_\Theta^2}{2} + \frac{1}{n}\sum_{i=0}^{n-1}\epsilon_i.$$

With the definition $\theta^* \doteq \operatorname{argmin}_{\theta\in\Theta}\Phi(\theta)$ and $\alpha \doteq \frac{D}{L_\Theta\sqrt{n}}$, we then have

$$\Phi(\bar{\theta}) - \min_{\theta\in\Theta}\Phi(\theta) \le \frac{DL_\Theta}{\sqrt{n}} + \frac{1}{n}\sum_{i=0}^{n-1}\epsilon_i.$$

□

Stochastic-gradient case.

Proof. Write $V(\theta,\omega) \doteq \mathbb{V}_{H\sim p_\omega,\pi_\theta}[\mathrm{IS}(\pi_e,\pi_\theta,H)]$ and $\Phi^* \doteq \min_{\theta\in\Theta}\Phi(\theta)$. Let $\mathcal{F}_i$ contain the history through the selection of ω_i, including θ_i and the nonnegative, integrable inner tolerance ϵ_i. The population inner-accuracy condition in Algorithm 2 is $V(\theta_i,\omega_i) \ge \Phi(\theta_i) - \epsilon_i$ almost surely. Set $\mathcal{G}_i = \nabla_\theta V(\theta_i,\omega_i)$ and use the sampled outer update $\theta_{i+1} = \mathrm{Proj}_\Theta[\theta_i - \alpha\widehat{\mathcal{G}}_i]$, where a fresh batch of K_{out} trajectories gives

$$\mathbb{E}[\widehat{\mathcal{G}}_i \mid \mathcal{F}_i] = \mathcal{G}_i, \qquad \mathbb{E}[\|\widehat{\mathcal{G}}_i - \mathcal{G}_i\|^2 \mid \mathcal{F}_i] \le \frac{\sigma_\theta^2}{K_{\mathrm{out}}}.$$

Let $\theta^* \in \operatorname{argmin}_{\theta\in\Theta}\Phi(\theta)$. Conditional on $\mathcal{F}_i$, θ_i and ω_i are fixed. Convexity of $V(\cdot,\omega_i)$ and the population inner-accuracy assumption give

$$\begin{aligned}\langle \mathcal{G}_i, \theta_i - \theta^*\rangle &\ge V(\theta_i,\omega_i) - V(\theta^*,\omega_i)\\ &\ge \Phi(\theta_i) - \Phi^* - \epsilon_i.\end{aligned}$$

By conditional unbiasedness and Lemma 5.2,

$$\begin{aligned}\mathbb{E}[\|\widehat{\mathcal{G}}_i\|^2 \mid \mathcal{F}_i] &= \|\mathcal{G}_i\|^2 + \mathbb{E}[\|\widehat{\mathcal{G}}_i - \mathcal{G}_i\|^2 \mid \mathcal{F}_i]\\ &\le L_\Theta^2 + \frac{\sigma_\theta^2}{K_{\mathrm{out}}}.\end{aligned}$$

Nonexpansiveness of Euclidean projection, followed by conditional expectation, therefore yields

$$\begin{aligned}\mathbb{E}[\|\theta_{i+1} - \theta^*\|^2 \mid \mathcal{F}_i] &\le \|\theta_i - \theta^*\|^2 - 2\alpha\langle \mathcal{G}_i, \theta_i - \theta^*\rangle + \alpha^2\mathbb{E}[\|\widehat{\mathcal{G}}_i\|^2 \mid \mathcal{F}_i]\\ &\le \|\theta_i - \theta^*\|^2 - 2\alpha(\Phi(\theta_i) - \Phi^* - \epsilon_i) + \alpha^2\left(L_\Theta^2 + \frac{\sigma_\theta^2}{K_{\mathrm{out}}}\right).\end{aligned}$$

Taking total expectations, rearranging, and summing over $i = 0, \dots, n-1$ gives

$$2\alpha \sum_{i=0}^{n-1} \big(\mathbb{E}[\Phi(\theta_i)] - \Phi^*\big) \le \mathbb{E}[\|\theta_0 - \theta^*\|^2] - \mathbb{E}[\|\theta_n - \theta^*\|^2] + 2\alpha \sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i] + n\alpha^2 \left(L_\Theta^2 + \frac{\sigma_\theta^2}{K_{\text{out}}}\right) \le D^2 + 2\alpha \sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i] + n\alpha^2 \left(L_\Theta^2 + \frac{\sigma_\theta^2}{K_{\text{out}}}\right).$$

The set Θ is convex, so $\bar{\theta} \in \Theta$. Convexity of Φ gives $\mathbb{E}[\Phi(\bar{\theta})] \le n^{-1} \sum_{i=0}^{n-1} \mathbb{E}[\Phi(\theta_i)]$. Dividing the preceding inequality by $2\alpha n$, for any constant $\alpha > 0$,

$$\mathbb{E}[\Phi(\bar{\theta})] - \Phi^* \le \frac{D^2}{2\alpha n} + \frac{\alpha}{2}\left(L_\Theta^2 + \frac{\sigma_\theta^2}{K_{\text{out}}}\right) + \frac{1}{n}\sum_{i=0}^{n-1} \mathbb{E}[\epsilon_i].$$

Substituting the stated step size and using $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$ for $a, b \ge 0$ proves (3).

To verify the variance constant, write the single-trajectory gradient from Lemma 5.1 as

$$Z_i(H) \doteq -\mathrm{IS}(\pi_e, \pi_{\theta_i}, H)^2 \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(A_t|S_t)|_{\theta=\theta_i}.$$

For $H \sim p_{\omega_i}, \pi_{\theta_i}$, this lemma gives $\mathbb{E}[Z_i(H) \mid \mathcal{F}_i] = \mathcal{G}_i$. The bounded rewards, features, and importance ratios imply, by (12) and (13),

$$\|Z_i(H)\| \le C^{2T}T^2 \cdot \sqrt{2}BT = \sqrt{2}BC^{2T}T^3 = L_\Theta.$$

If $H_{i,1}, \dots, H_{i,K_{\text{out}}}$ are conditionally independent trajectories from this distribution, sampled after ω_i is selected, set $\widehat{\mathcal{G}}_i = K_{\text{out}}^{-1} \sum_{j=1}^{K_{\text{out}}} Z_i(H_{i,j})$. Conditional independence and unbiasedness give

$$\mathbb{E}[\|\widehat{\mathcal{G}}_i - \mathcal{G}_i\|^2 \mid \mathcal{F}_i] = \frac{1}{K_{\text{out}}}\mathbb{E}[\|Z_i(H_{i,1}) - \mathcal{G}_i\|^2 \mid \mathcal{F}_i] \le \frac{1}{K_{\text{out}}}\mathbb{E}[\|Z_i(H_{i,1})\|^2 \mid \mathcal{F}_i] \le \frac{L_\Theta^2}{K_{\text{out}}}.$$

Thus $\sigma_\theta^2 = L_\Theta^2 = 2B^2C^{4T}T^6$ is a valid choice. □

Inner-loop empirical decomposition. If $\widehat{V}_i$ estimates $V(\theta_i, \cdot)$ and $\sup_{\omega\in\Omega} \widehat{V}_i(\omega) - \widehat{V}_i(\omega_i) \le \widehat{\epsilon}_i$, the population inner gap satisfies

$$\epsilon_i^{\text{pop}} \doteq \Phi(\theta_i) - V(\theta_i, \omega_i) \le \widehat{\epsilon}_i + 2\sup_{\omega\in\Omega} |\widehat{V}_i(\omega) - V(\theta_i, \omega)|.$$

To prove this, let $\delta_i \doteq \sup_{\omega\in\Omega} |\widehat{V}_i(\omega) - V(\theta_i, \omega)|$. If $\sup_{\omega\in\Omega} \widehat{V}_i(\omega) - \widehat{V}_i(\omega_i) \le \widehat{\epsilon}_i$, then for every $\omega \in \Omega$,

$$V(\theta_i, \omega) - V(\theta_i, \omega_i) \le \widehat{V}_i(\omega) - \widehat{V}_i(\omega_i) + 2\delta_i \le \widehat{\epsilon}_i + 2\delta_i.$$

Taking the supremum over ω proves the claim. When these terms are integrable, the outer bound therefore holds with $\mathbb{E}[\epsilon_i] \le \mathbb{E}[\widehat{\epsilon}_i] + 2\mathbb{E}[\delta_i]$ by taking $\epsilon_i = \epsilon_i^{\text{pop}}$. This algebraic decomposition does not assume that a local empirical solver attains a global optimization gap, or supply a uniform concentration rate for a general neural transition class.

B NUMERICAL STUDIES

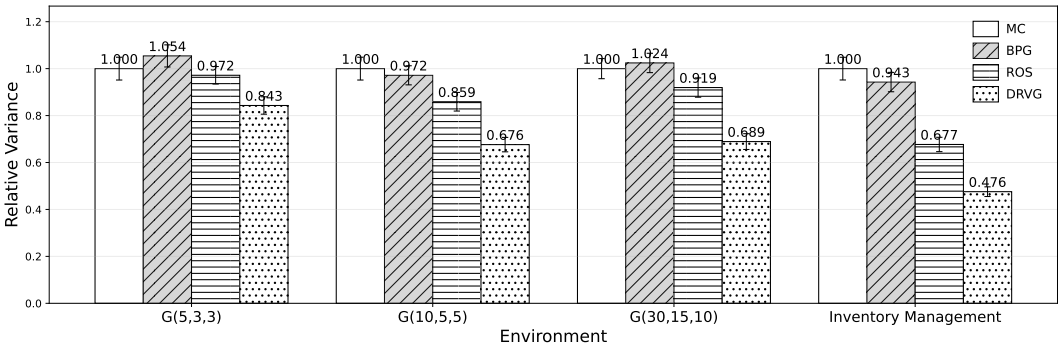


Figure 3: Supplementary figure for Section 6.1. Relative variance of each method under its tailored adversarial transition. All values are normalized by the variance of the on-policy Monte Carlo (MC) method (under its tailored adversarial transition) in the same environment. Error bars denote the standard error over 900 total runs per environment (30 target policies with 30 independent runs each).

In our numerical studies, we leverage a wide range of target policies ranging from completely random to highly deterministic. Specifically, a policy π_{train} is computed as the optimal policy of the MDP model, and π_{random} is randomly generated. Then, the target policies π_e are set to be $(1-\beta)\pi_{\text{train}} + \beta\pi_{\text{random}}$ with $\beta \in \{\frac{1}{30}, \frac{2}{30}, ..., 1\}$. For each of the 30 target policies, we have 30 independent runs, resulting in a total of 900 runs for each value.

In Section 6.1, we generate method-specific adversarial transitions by running Algorithm 1 separately for each method, yielding one adversarial transition per behavior policy. For ROS (Zhong et al., 2022), which adapts its behavior policy, we generate the adversarial transition by treating the target policy as the nominal behavior policy in Algorithm 1. We first measure each method's variance under the original simulator transition p_0, and then under its method-specific adversarial transition p_{adv}. Note that p_{adv} differs across methods. We report the relative variance of each method under its own p_{adv} in Figure 3, where each value is normalized by the variance of MC (also under its p_{adv}). Finally, we report the variance increase for each method in Figure 1, defined as the difference between its variance under p_{adv} and under p_0. In Section 6.2, we evaluate all methods under a shared adversarial target transition. This transition is constructed by applying Algorithm 1 to the on-policy Monte Carlo baseline. We then report each method's variance under this transition in Figure 2.

B.1 EXPERIMENTAL SETUP

To ensure reproducibility and to isolate the source of variance reduction across methods, we adopt a uniform experimental protocol across all four environments. Our setup follows the standard conventions established in the behavior policy search literature (Hanna et al., 2017; 2024; Zhong et al., 2022). We parameterize the behavior policy π_θ as a two-layer multilayer perceptron (MLP) with 64 hidden units per layer and tanh activations, followed by a softmax output head over the discrete action space. We adopt the same architecture for the adversarial transition model p_ω, with its softmax head defined over the discrete next-state space. This shared architecture, deployed identically across environments and across all four methods, removes parameterization choices as a source of confounding variance in our comparisons.

We optimize all network parameters with the Adam optimizer (Kingma & Ba, 2015), using a learning rate of 10^{-3} and the default momentum coefficients $\beta_1 = 0.9$ and $\beta_2 = 0.999$. For BPG (Hanna et al., 2017) and ROS (Zhong et al., 2022), we adopt the hyperparameters reported in their original publications, ensuring that each baseline is evaluated under conditions favorable to its own design. All four methods (MC, BPG, ROS, and DRVG) are trained on the same initial simulator transition and evaluated against the same set of target policies, so any performance differences are attributable to the variance-reduction strategy itself rather than to environment configuration.

To establish the statistical reliability of our reported numbers, we evaluate each method across 30 target policies generated by the mixing scheme $\pi_e = (1-\beta)\pi_{\text{train}} + \beta\pi_{\text{random}}$ described in Appendix B, with 30 independent runs per target policy. This yields 900 total runs per reported value. Error bars in Figures 1, 2, and 3 denote the standard error of the mean computed across these 900 runs. This protocol provides sufficient resolution to detect consistent variance differences between methods, which is essential given the high baseline variance of policy evaluation under transition perturbations.

All experiments were conducted on a single shared compute node equipped with two AMD EPYC 7663 processors (56 cores per socket, two sockets, two threads per core, yielding 224 logical CPUs of which 222 are allocatable to jobs) and 1024 GB of system memory (1000 GB allocatable).

B.2 Garnet Examples

A Garnet environment (Archibald et al., 1995) is represented by three integers $(|S|, |A|, b)$, denoting the number of states, actions, and the branching factor, respectively. By varying b, one controls the degree of stochasticity: small b yields sparse transitions, while large b approaches fully connected transitions. This flexibility makes Garnets particularly suitable for stress-testing reinforcement learning algorithms across a wide spectrum of transition structures (Tarbouriech & Lazaric, 2019; Wang et al., 2023a;b). We evaluate the four methods on three Garnet instances—$G(5, 3, 3)$, $G(10, 5, 5)$, and $G(30, 15, 10)$—which span increasing environment sizes and connectivity levels.

B.3 Inventory Management

Inventory management (Porteus, 2002; Ho et al., 2018) is a classical stochastic control problem under transition uncertainty. The state corresponds to inventory levels, actions represent order quantities, and stochastic demand drives the state transitions. In our inventory management example, we adopt radial-type basis functions as introduced in Sutton & Barto (2018), defined for state s and feature index i as $\phi_i(s) = \exp\left(-\frac{\|s-c_i\|^2}{2\sigma_i^2}\right)$, where c_i and σ_i denote the deterministic center and scaling parameter of the i-th feature, respectively. This nonlinear parameterization captures variations in state representation while controlling the expressive capacity of the model under uncertainty.

C SUPPLEMENTARY OFF-TRANSITION GRADIENT RESULTS

In the *off-transition* case, analogous to the off-policy setting, simulator transitions are fixed at p_{ω_0} and may differ from the target transition p_ω. This situation arises naturally when using black-box simulators that permit data collection but do not allow modifying transition probabilities (e.g., Komorowski et al. (2018)). In this case, we introduce a transition importance sampling ratio to reweight the collected trajectories, mirroring the familiar correction used in off-policy evaluation for policies. For a general off-policy estimator OPE, we overload the notation as

$$\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H) \doteq \frac{\prod_{t=0}^{T-1} p_\omega(S_{t+1}|S_t, A_t)}{\prod_{t=0}^{T-1} p_{\omega_0}(S_{t+1}|S_t, A_t)} \mathrm{OPE}(\pi_e, \pi_\theta, H).$$

We omit the input p_{ω_0} in $\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)$ to simplify notations. Similar to Theorem 4.1, we first give an analytical gradient expression of the evaluation variance.

Theorem C.1 (Off-Transition Gradient of Variance)**.** *When $p_\omega \neq p_{\omega_0}$, for a fixed behavior policy π_θ,*

$$\frac{\partial}{\partial \omega} \mathbb{V}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] = 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}^2(\pi_e, \pi_\theta, p_\omega, H)\frac{\partial}{\partial \omega}\ell_{p_\omega}\right]$$
$$- 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)]\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)\frac{\partial}{\partial \omega}\ell_{p_\omega}\right].$$

Its proof is in Appendix C.1. In the next theorem, we incorporate a KL-divergence term to penalize large deviations of p_ω from the simulator's transition p_{ω_0}, with the KL direction chosen so that the expectation aligns with the available sampling distribution p_{ω_0}. This design ensures the realism of the learned adversarial transition.

Theorem C.2 (Off-transition Gradient of Variance with KL)**.** *For a fixed behavior policy π_θ and a regularization coefficient $\eta > 0$,*

$$\frac{\partial}{\partial \omega}\left[\mathbb{V}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] - \eta \mathrm{KL}(\Pr(H|p_{\omega_0})\|\Pr(H|p_\omega))\right]$$
$$=2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}^2(\pi_e, \pi_\theta, p_\omega, H)\frac{\partial}{\partial \omega}\ell_{p_\omega}\right] - 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)]$$
$$\cdot \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)\frac{\partial}{\partial \omega}\ell_{p_\omega}\right] - \eta\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[-\frac{\partial}{\partial \omega}\ell_{p_\omega}\right].$$

Its proof is in Appendix C.2. Note that the gradient expression in the off-transition setting (Theorem C.2) differs from that in the on-transition case (Theorem 4.3), reflecting the distinct data sampling mechanisms.

C.1 PROOF OF THEOREM C.1

$H \sim p_{\omega_0}, \pi_\theta$ denotes fixed-simulator sampling throughout this proof and Appendix C.2.

Proof. For simplification, we define $w_\pi(h) \doteq \prod_{t=0}^{T-1} \pi(A_t|S_t)$ under trajectory h. Then,

$$
\begin{aligned}
&\frac{\partial}{\partial\omega}\mathbb{V}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\\
=&\frac{\partial}{\partial\omega}\left(\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}^2(\pi_e,\pi_\theta,p_\omega,H)]-\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]^2\right)\\
=&\frac{\partial}{\partial\omega}\left(\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\frac{m^2_{p_\omega}(H)}{m^2_{p_{\omega_0}}(H)}\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\right]-\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\frac{m_{p_\omega}(H)}{m_{p_{\omega_0}}(H)}\mathrm{OPE}(\pi_e,\pi_\theta,H)\right]^2\right)\\
=&\frac{\partial}{\partial\omega}\sum_h\left(\Pr(H=h|p_{\omega_0})\frac{m^2_{p_\omega}(H)}{m^2_{p_{\omega_0}}(H)}\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\right)-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\\
&\frac{\partial}{\partial\omega}\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\frac{m_{p_\omega}(H)}{m_{p_{\omega_0}}(H)}\mathrm{OPE}(\pi_e,\pi_\theta,H)\right]\\
=&\sum_h\left(p(h)\frac{1}{m_{p_{\omega_0}}(H)}\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m^2_{p_\omega}(h)\right)\\
&-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\\
&\cdot\frac{\partial}{\partial\omega}\sum_h\left(p(h)m_{p_{\omega_0}}(h)\frac{m_{p_\omega}(h)}{m_{p_{\omega_0}}(h)}\mathrm{OPE}(\pi_e,\pi_\theta,H)\right) && \text{(By (6))}\\
=&2\sum_h\left(p(h)\frac{m_{p_\omega}(h)}{m_{p_{\omega_0}}(h)}\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right)\\
&-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\sum_h\left(p(h)\mathrm{OPE}(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}m_{p_\omega}(h)\right) && \text{(By (6))}\\
=&2\sum_h\left(p(h)\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\frac{m_{p_\omega}(h)}{m_{p_{\omega_0}}(h)}m_{p_\omega}(h)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right)\\
&-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\sum_h\left(p(h)\mathrm{OPE}(\pi_e,\pi_\theta,H)m_{p_\omega}(h)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right) && \text{(By (7))}\\
=&2\sum_h\left(p(h)m_{p_{\omega_0}}(h)\frac{m^2_{p_\omega}(h)}{m^2_{p_{\omega_0}}(h)}\mathrm{OPE}^2(\pi_e,\pi_\theta,H)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right)\\
&-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]\\
&\cdot\sum_h\left(p(h)m_{p_{\omega_0}}(h)\frac{m_{p_\omega}(h)}{m_{p_{\omega_0}}(h)}\mathrm{OPE}(\pi_e,\pi_\theta,H)\sum_{t=0}^{T-1}\frac{\partial}{\partial\omega}\log(p_\omega(S_{t+1}|S_t,A_t))\right) && \text{(By (7))}\\
=&2\sum_h\left(\Pr(H=h|p_{\omega_0})\mathrm{OPE}^2(\pi_e,\pi_\theta,p_\omega,H)\frac{\partial}{\partial\omega}\ell_{p_\omega}\right)\\
&-2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e,\pi_\theta,p_\omega,H)]
\end{aligned}
$$

$$\cdot \sum_h \left(\Pr(H=h|p_{\omega_0}) \mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H) \sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t)) \right) \qquad \text{(By (6))}$$

$$= 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}^2(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right]$$

$$- 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right].$$

□

C.2 Proof of Theorem C.2

The KL-divergence between two probability distribution p and q is defined as $\mathrm{KL}(p\|q) \doteq \mathbb{E}_{X\sim p}\left[\log \frac{p(X)}{q(X)}\right]$. Therefore, the KL-divergence between the trajectory distribution of the target transition p_ω and the simulator's transition p_{ω_0} is given by

$$\begin{aligned}
\mathrm{KL}(\Pr(H|p_{\omega_0}) \| \Pr(H|p_\omega)) =& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\log \frac{\Pr(H|p_{\omega_0})}{\Pr(H|p_\omega)}\right] \\
=& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\log \frac{m_{p_{\omega_0}}(H)}{m_{p_\omega}(H)}\right] \qquad \text{(By (6))} \\
=& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\log m_{p_{\omega_0}}(H) - \log m_{p_\omega}(H)\right].
\end{aligned}$$

We take the gradient of the KL-divergence with respect to ω:

$$\begin{aligned}
\frac{\partial}{\partial \omega}\mathrm{KL}(\Pr(H|p_{\omega_0}) \| \Pr(H|p_\omega)) =& \frac{\partial}{\partial \omega}\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\log m_{p_{\omega_0}}(H) - \log m_{p_\omega}(H)\right] \\
=& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[-\frac{\partial}{\partial \omega} \log m_{p_\omega}(H)\right] \\
=& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[-\sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log p_\omega(S_{t+1}|S_t, A_t)\right] \\
=& \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[-\sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t))\right]. \qquad (19)
\end{aligned}$$

Thus,

$$\begin{aligned}
&\frac{\partial}{\partial \omega}\left[\mathbb{V}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] - \eta \mathrm{KL}(\Pr(H|p_{\omega_0}) \| \Pr(H|p_\omega))\right] \\
=& 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}^2(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right] \\
&- 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right] \\
&- \frac{\partial}{\partial \omega}\left[\eta \mathrm{KL}(\Pr(H|p_{\omega_0}) \| \Pr(H|p_\omega))\right] \\
=& 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}^2(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right] \\
&- 2\mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H)] \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[\mathrm{OPE}(\pi_e, \pi_\theta, p_\omega, H) \frac{\partial}{\partial \omega} \ell_{p_\omega}\right] \\
&- \eta \mathbb{E}_{H\sim p_{\omega_0},\pi_\theta}\left[-\sum_{t=0}^{T-1} \frac{\partial}{\partial \omega} \log(p_\omega(S_{t+1}|S_t, A_t))\right]. \qquad \text{(By (19))}
\end{aligned}$$