Efficient Policy Evaluation with Offline Data Informed Behavior Policy Design

Shuze Liu [1] **Shangtong Zhang** [1]

Abstract

Most reinforcement learning practitioners evaluate their policies with online Monte Carlo estimators for either hyperparameter tuning or testing different algorithmic design choices, where the policy is repeatedly executed in the environment to get the average outcome. Such massive interactions with the environment are prohibitive in many scenarios. In this paper, we propose novel methods that improve the data efficiency of online Monte Carlo estimators while maintaining their unbiasedness. We first propose a tailored closed-form behavior policy that provably reduces the variance of an online Monte Carlo estimator. We then design efficient algorithms to learn this closed-form behavior policy from previously collected offline data. Theoretical analysis is provided to characterize how the behavior policy learning error affects the amount of reduced variance. Compared with previous works, our method achieves better empirical performance in a broader set of environments, with fewer requirements for offline data.

1. Introduction

Reinforcement Learning (RL, Sutton & Barto (2018)) has recently demonstrated great success in solving sequential decision-making problems. For example, AlphaStar (Vinyals et al., 2019) defeats the best human StarCraft II players and is ranked at the GrandMaster level in the StarCraft ladder. The canonical RL paradigm behind the success, however, requires massive active interactions with the environment to obtain data (Sutton, 1988; Watkins & Dayan, 1992; Sutton et al., 1999; Mnih et al., 2015). Those data are called online data, and this paradigm is called online RL. Requiring massive online data is, however, prohibitive in many scenarios. First, obtaining massive online data can be both expensive and slow in the real world (Li, 2019; Zhang, 2023). Second, even if a simulator is available, obtaining massive online data can still be prohibitively slow for high-fidelity simulation (Chervonyi et al., 2022).

Offline RL (Ernst et al., 2005; Lange et al., 2012; Fujimoto et al., 2019; Levine et al., 2020) attacks this issue using existing, previously logged data, called offline data. Compared with online data, offline data is cheaper and safer (Li, 2019; Zhang, 2023). Offline RL also demonstrates great success. For example, Mathieu et al. (2023) train an offline AlphaStar, which uses only existing human replays without any interaction with the StarCraft II simulator during training. The offline AlphaStar obtains over 90% win rates against the supervised learning agent in Vinyals et al. (2019).

However, most RL practitioners, even offline RL practitioners, still heavily rely on online Monte Carlo estimators. For example, Mathieu et al. (2023) repeatedly execute their trained offline AlphaStar agents in the StartCraft II simulator and use the win rates as the performance metric for hyperparameter tuning and evaluating different algorithmic design choices. This evaluation practice is the straightforward online Monte Carlo evaluation and requires massive online data. There are indeed offline evaluation methods, most of which, however, still rely on online Monte Carlo evaluation for hyperparameter tuning and testing different algorithmic design choices (see, e.g., Fu et al. (2020); Gülçehre et al. (2020); Schrittwieser et al. (2021); Mathieu et al. (2023)).

Improving the sample efficiency of online Monte Carlo estimators while maintaining their unbiasedness is thus a need for both online and offline RL practitioners. We emphasize *unbiasedness* because it is arguably one of the key reasons that make Monte Carlo estimators so dominating. In this paper, we make three contributions toward fulfilling this need. **First**, we propose tailored closed-form behavior policies that *provably* reduce the variance of online Monte Carlo estimators. **Second**, we design efficient algorithms to learn the closed-form behavior policies from offline data. Theoretical analysis is provided to characterize how the behavior policy learning error affects the amount of reduced variance. Notably, this learning error does not introduce any bias in the estimation. **Third**, we conduct thorough empirical studies in a broad set of environments. Compared with previous works, our method achieves better empirical

---

[1] Department of Computer Science, University of Virginia. Correspondence to: Shuze Liu <shuzeliu@virginia.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

performance while being less restrictive on offline data.

2. Background

We consider a finite horizon Markov Decision Process (MDP, Puterman (2014)) with a finite state space $\mathcal{S}$, a finite action space $\mathcal{A}$, a reward function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, a transition probability function $p : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to [0, 1]$, an initial distribution $p_0 : \mathcal{S} \to [0, 1]$, and a constant horizon length T. Without loss of generality, we consider the undiscounted setting for simplifying notations. Our results naturally apply to the discounted setting (Puterman, 2014) as long as the horizon is fixed and finite. For any integer n, we define as shorthand $[n] \doteq \{0, 1, \ldots, n\}$. At time step 0, an initial state S_0 is sampled from p_0. At time step $t \in [T-1]$, an action A_t is sampled according to $\pi_t(\cdot \mid S_t)$ where $\pi_t : \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the policy at time step t. A finite reward $R_{t+1} \doteq r(S_t, A_t)$ is then emitted and a successor state S_{t+1} is sampled from $p(\cdot \mid S_t, A_t)$. We define abbreviations $\pi_{i:j} \doteq \{\pi_i, \pi_{i+1}, \ldots, \pi_j\}$ and $\pi \doteq \pi_{0:T-1}$. The return at time step t is defined as $G_t \doteq \sum_{i=t+1}^{T} R_i$, which allows defining the state- and action-value functions as $v_{\pi,t}(s) \doteq \mathbb{E}_\pi \left[G_t \mid S_t = s\right]$ and $q_{\pi,t}(s, a) \doteq \mathbb{E}_\pi \left[G_t \mid S_t = s, A_t = a\right]$. We use the total rewards performance metric (Puterman, 2014) to measure the performance of the policy π, which is defined as $J(\pi) \doteq \sum_s p_0(s) v_{\pi,0}(s)$. In this paper, we focus on Monte Carlo methods introduced by Kakutani (1945) to estimate the total rewards $J(\pi)$. Among its variants, the most straightforward and widely used way is to draw samples of $J(\pi)$ by executing the policy π online. As the number of samples increases, the empirical average of the sampled returns converges to $J(\pi)$. This idea is called on-policy learning (Sutton 1988) because it estimates a policy π by executing itself.

From now on, we consider off-policy learning, where we estimate the total rewards $J(\pi)$ of an interested policy π, called the target policy, by executing a different policy μ, called the behavior policy. In off-policy learning, each trajectory $\{S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S_{T-1}, A_{T-1}, R_T\}$ is generated by a behavior policy μ with $S_0 \sim p_0, A_t \sim \mu_t(\cdot|S_t)$, $t \in [T-1]$. Let

$$\tau_{t:T-1}^{\mu_{t:T-1}} \doteq \{S_t, A_t, R_{t+1}, \ldots, S_{T-1}, A_{T-1}, R_T\}$$

be a shorthand for a segment of a random trajectory generated by the behavior policy μ from the time step t to the time step $T-1$ inclusively. In off-policy learning, we use the importance sampling ratio to reweight rewards collected by μ in order to give an estimate of $J(\pi)$. The importance sampling ratio at time step t is defined as $\rho_t \doteq \frac{\pi_t(A_t|S_t)}{\mu_t(A_t|S_t)}$. The product of importance sampling ratios from time t to $t' \geq t$ is defined as $\rho_{t:t'} \doteq \prod_{k=t}^{t'} \frac{\pi_k(A_k|S_k)}{\mu_k(A_k|S_k)}$. There are various ways to use the importance sampling ratios in off-policy learning (Geweke, 1988; Hesterberg, 1995; Koller & Friedman, 2009; Thomas, 2015). We start with the per-decision importance sampling estimator (PDIS, Precup et al. (2000)) in this work and leave the investigation of others for future work. The PDIS Monte Carlo estimator is defined as

$$G^{\text{PDIS}}(\tau_{t:T-1}^{\mu_{t:T-1}}) \doteq \sum_{k=t}^{T-1} \rho_{t:k} R_{k+1} \tag{1}$$

and is unbiased for any behavior policy μ that covers target policy π (Precup et al., 2000). In other words, when $\forall s, \forall a$, $\mu_t(a|s) = 0 \implies \pi_t(a|s) = 0$, we have $\forall t, \forall s$,

$$\mathbb{E}[G^{\text{PDIS}}(\tau_{t:T-1}^{\mu_{t:T-1}}) \mid S_t = s] = v_{\pi,t}(s).$$

We intensively use the recursive form of the PDIS estimator:

$$\begin{aligned} &G^{\text{PDIS}}(\tau_{t:T-1}^{\mu_{t:T-1}}) \\ =&\begin{cases} \rho_t \left(R_{t+1} + G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu_{t+1:T-1}})\right) & t \in [T-2], \\ \rho_t R_{t+1} & t = T-1. \end{cases} \end{aligned} \tag{2}$$

Since the PDIS estimator is unbiased, reducing its variance is sufficient for improving its sample efficiency. We achieve this variance reduction by designing and learning proper behavior policies.

3. Variance Reduction in Statistics

In this section, we provide the mathematical foundation for variance reduction with importance sampling ratios. The notations here are independent of the rest of this paper. We use similar notations only for easy interpretation in later sections. Consider a discrete random variable A taking values from a finite space $\mathcal{A}$ according to a probability mass function $\pi : \mathcal{A} \to [0, 1]$ and a function $q : \mathcal{A} \to \mathbb{R}$ mapping a value in $\mathcal{A}$ to a real number. We are interested in estimating $\mathbb{E}_{A\sim\pi}[q(A)]$. The ordinary Monte Carlo methods then sample $\{A_1, \ldots, A_N\}$ from π and use the empirical average $\frac{1}{N}\sum_{i=1}^{N} q(A_i)$ as the estimate. In statistics, importance sampling is introduced as a variance reduction technique for Monte Carlo methods (Rubinstein 1981). The main idea is to sample $\{A_i, \ldots, A_N\}$ from a different distribution μ and use $\frac{1}{N}\sum_{i=1}^{N} \rho(A_i) q(A_i)$ as the estimate, where $\rho(A) \doteq \frac{\pi(A)}{\mu(A)}$ is the importance sampling ratio. Assuming μ covers π, i.e.,

$$\forall a, \mu(a) = 0 \implies \pi(a) = 0, \tag{3}$$

the importance sampling ratio weighted empirical average is then unbiased, i.e.,

$$\mathbb{E}_{A\sim\pi}[q(A)] = \mathbb{E}_{A\sim\mu}[\rho(A) q(A)].$$

If the sampling distribution μ is carefully designed, the variance can also be reduced. To adapt this idea for RL, we relax the condition (3) in this section. We formulate this problem of searching a variance-reducing sampling distribution as an optimization problem:

$$\min_{\mu\in\Lambda_+} \quad \mathbb{V}_{A\sim\mu}(\rho(A)q(A)). \tag{4}$$

Here Λ_+ denotes the set of all the policies that give unbiased estimations, i.e.,

$$\Lambda_+ \doteq \{\mu \in \Delta(\mathcal{A}) \mid \mathbb{E}_{A\sim\mu}\left[\rho(A)q(A)\right] = \mathbb{E}_{A\sim\pi}\left[q(A)\right]\},$$

where $\Delta(\mathcal{X})$ denotes the set of all probability distributions on the set $\mathcal{X}$. Solving (4) is actually very challenging. To see this, consider a concrete example where $\mathcal{A} = \{a_1, a_2, a_3\}$ and

$$\begin{cases} q(a_1) = -10 \\ q(a_2) = 2 \\ q(a_3) = 2 \end{cases}, \quad \begin{cases} \pi(a_1) = 0.1 \\ \pi(a_2) = 0.5 \\ \pi(a_3) = 0.4 \end{cases}, \quad \begin{cases} \mu(a_1) = 0 \\ \mu(a_2) = 0 \\ \mu(a_3) = 1 \end{cases}. \tag{5}$$

It can be computed that $\mathbb{E}_{A\sim\pi}\left[q(A)\right] = 0.8$ and $\mathbb{E}_{A\sim\mu}\left[\rho(A)q(A)\right] = 0.8$. In other words, we could sample A from μ and use $\rho(A)q(A)$ as an estimator. This estimator is unbiased. But apparently, this μ does not cover π. Moreover, since μ is deterministic, the variance of this estimator is 0. Then μ is an optimal sampling distribution. However, μ is hand-crafted based on the knowledge that $q(a_1)\pi(a_1) + q(a_2)\pi(a_2) = 0$. Without such knowledge, we argue that there is little hope to find this μ. This example suggests that searching over the entire Λ_+ might be too ambitious. One natural choice presented by Rubinstein (1981) is to restrict the search to

$$\Lambda_- \doteq \{\mu \in \Delta(\mathcal{A}) \mid \forall a, \mu(a) = 0 \implies \pi(a) = 0\}. \tag{6}$$

In other words, we aim to find a variance-minimizing sampling distribution among all distributions that cover π. Because coverage implies unbiasedness, we have $\Lambda_- \subseteq \Lambda_+$. In this work, we enlarge Λ_- to Λ defined as

$$\Lambda \doteq \{\mu \in \Delta(\mathcal{A}) \mid \forall a, \mu(a) = 0 \implies \pi(a)q(a) = 0\}. \tag{7}$$

following Owen (2013). The space Λ weakens the assumption in (6). Owen (2013) proves that any distribution μ in Λ gives unbiased estimation, though μ may not cover π.

Lemma 1. $\forall \mu \in \Lambda, \mathbb{E}_{A\sim\mu}\left[\rho(A)q(A)\right] = \mathbb{E}_{A\sim\pi}\left[q(A)\right].$

For completeness, its proof is in Appendix A.1. We now consider the variance minimization problem on Λ, i.e.,

$$\min_{\mu\in\Lambda} \quad \mathbb{V}_{A\sim\mu}(\rho(A)q(A)). \tag{8}$$

The following lemma from Owen (2013) gives an optimal solution μ^* to the optimization problem (8).

Lemma 2. *Define $\mu^*(a) \propto \pi(a)|q(a)|$. Then μ^* is an optimal solution to (8).*

For completeness, its proof is detailed in Appendix A.2. Here by

$$\mu(a) \propto \pi(a)w(a)$$

with some non-negative $w(a)$, we mean

$$\mu(a) \doteq \pi(a)w(a)/\textstyle\sum_b \pi(b)w(b).$$

The reader may notice that if $\pi(a)w(a) = 0$ for all a, the above "reweighted" distribution is not well defined. We then use the convention to interpret $\mu(a)$ as a uniform distribution, i.e., $\mu(a) = 1/|\mathcal{A}|$. *We adopt this convention in using $\propto$ in the rest of the paper to simplify the presentation.* The following lemma gives intuition on the optimality of μ^*, whose proof is in Appendix A.3.

Lemma 3. *If $\forall a \in \mathcal{A}, q(a) \geq 0$ or $\forall a \in \mathcal{A}, q(a) \leq 0$, then $\Lambda = \Lambda_+$, and the μ^* defined in Lemma 2 gives a zero variance, i.e., $\mathbb{V}_{A\sim\mu^*}(\rho(A)q(A)) = 0$.*

An optimal sampling distribution proportional to $\pi(a)|q(a)|$ dates back to Kahn & Marshall (1953); Rubinstein (1981); Benjamin Melamed (1998) and is commonly used in RL (Carpentier et al., 2015; Mukherjee et al., 2022). We, however, make two remarks. **First**, we show such a sampling distribution can be suboptimal in Λ_+. For (5), such a sampling distribution incurs strictly positive variance, but μ in (5) has a zero variance and is also unbiased. **Second**, different from existing literature in RL (Carpentier et al., 2015; Sutton & Barto, 2018; Mukherjee et al., 2022), our μ^* defined in Lemma 2 does not need to cover π. Nevertheless, we note that Lemma 1 still ensures that μ^* gives unbiased estimation (Owen, 2013) and extend unbiasedness to RL settings in Theorem 1.

4. Variance Reduction in Reinforcement Learning

We now apply the techniques in Section 3 in RL. In particular, we seek to reduce the variance $\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{0:T-1}}_{0:T-1})\right)$ by designing a proper behavior policy μ. Of course, we need to ensure that the PDIS estimator with this behavior policy is unbiased. In other words, ideally we should search over

$$\Lambda_+ \doteq \left\{\mu \in \Delta(\mathcal{A})^T \mid \mathbb{E}\left[G^{\text{PDIS}}(\tau^{\mu_{0:T-1}}_{0:T-1})\right] = J(\pi)\right\}.$$

As discussed in Section 3, this is too ambitious without domain-specific knowledge. Instead, we can search over all policies that cover π, i.e.,

$$\Lambda_- \doteq \{\mu \mid \forall t, s, a, \mu_t(a|s) = 0 \implies \pi_t(a|s) = 0\}.$$

The set Λ_- contains all policies that satisfy the policy coverage constraint in off-policy learning (Sutton & Barto 2018). Similar to (7), we can also enlarge Λ_- to

$$\Lambda \doteq \{\mu \mid \forall t, s, a, \mu_t(a|s) = 0 \implies \pi_t(a|s)q_{\pi,t}(s,a) = 0\}.$$

The following theorem ensures the desired unbiasedness, which is proved in Appendix A.4.

Theorem 1 (Unbiasedness)**.** $\forall \mu \in \Lambda, \forall t, \forall s$, $\mathbb{E}\left[G^{PDIS}(\tau_{t:T-1}^{\mu_{t:T-1}}) \mid S_t = s\right] = v_{\pi,t}(s)$.

One immediate consequence of Theorem 1 is that $\forall \mu \in \Lambda, \mathbb{E}\left[G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}})\right] = J(\pi)$. In this paper, we consider a set Λ^* such that $\Lambda_- \subseteq \Lambda^* \subseteq \Lambda$. This Λ^* inherits the unbiasedness property of Λ and is less restrictive than Λ_-, the classical search space of behavior policies. This Λ^* will be defined shortly. We now formulate our problem as

$$\min_{\mu \in \Lambda^*} \quad \mathbb{V}\left(G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}})\right). \tag{9}$$

By the law of total variance, for any $\mu \in \Lambda^*$, we decompose the variance of the PDIS estimator as

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}})\right) \\
=&\mathbb{E}_{S_0}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}}) \mid S_0\right)\right] \\
&+ \mathbb{V}_{S_0}\left(\mathbb{E}\left[G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}}) \mid S_0\right]\right) \\
=&\mathbb{E}_{S_0}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}}) \mid S_0\right)\right] + \mathbb{V}_{S_0}\left(v_{\pi,0}(S_0)\right). \\
&\text{(by Theorem 1)}
\end{aligned}$$

The second term $\mathbb{V}_{S_0}\left(v_{\pi,0}(S_0)\right)$ is a constant given a target policy π and is unrelated to the choice of μ. In the first term, the expectation is taken over S_0 that is determined by the initial probability distribution p_0. Consequently, to solve the problem (9), it is sufficient to solve for each s,

$$\min_{\mu \in \Lambda^*} \quad \mathbb{V}\left(G^{\text{PDIS}}(\tau_{0:T-1}^{\mu_{0:T-1}}) \mid S_0 = s\right). \tag{10}$$

Denote the variance of the state value for the next state given the current state-action pair (s, a) as $\nu_{\pi,t}(s, a)$. We have $\nu_{\pi,t}(s, a) = 0$ for $t = T - 1$ and otherwise

$$\nu_{\pi,t}(s, a) \doteq \mathbb{V}_{S_{t+1}}\left(v_{\pi,t+1}(S_{t+1}) \mid S_t = s, A_t = a\right). \tag{11}$$

We now construct a behavior policy μ^* as

$$\mu_t^*(a|s) \propto \pi_t(a|s)\sqrt{u_{\pi,t}(s, a)}, \tag{12}$$

where $u_{\pi,t}(s, a) \doteq q_{\pi,t}^2(s, a)$ for $t = T - 1$ and otherwise

$$\begin{aligned}
&u_{\pi,t}(s, a) = q_{\pi,t}^2(s, a) + \nu_{\pi,t}(s, a) \\
&+ \textstyle\sum_{s'} p(s'|s, a)\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu^*_{t+1:T-1}}) \mid S_{t+1} = s'\right).
\end{aligned} \tag{13}$$

Notably, μ_t^* and $u_{\pi,t}$ are defined backwards and alternatively, i.e., they are defined in the order of $u_{\pi,T-1}, \mu_{T-1}^*, u_{\pi,T-2}, \mu_{T-2}^*, \dots, u_{\pi,0}, \mu_0^*$. We prove μ^* is optimal in the following sense.

Theorem 2 (Optimal Behavior Policy)**.** *For any t and s, the behavior policy $\mu_t^*(a|s)$ defined above is an optimal solution to the following problem*

$$\min_{\mu_t \in \Lambda_t, \dots, \mu_{T-1} \in \Lambda_{T-1}} \quad \mathbb{V}\left(G^{PDIS}(\tau_{t:T-1}^{\mu_{t:T-1}}) \mid S_t = s\right),$$

where $\Lambda_t \doteq \{\mu_t \in \Delta(\mathcal{A}) \mid \forall s, a, \mu_t(a|s) = 0 \implies \pi_t(a|s)u_{\pi,t}(s, a) = 0\}$.

Its proof is in Appendix A.5. We are now ready to define $\Lambda^* \doteq \Lambda_0 \times \cdots \times \Lambda_{T-1}$. Theorem 2 indicates that μ^* achieves optimality for the optimization problem (10). Since $u_{\pi,t}(s, a) = 0 \implies q_{\pi,t}(s, a) = 0$ by the non-negativity of the summands in (13), we have $\Lambda^* \subseteq \Lambda$. If $\mu_t(a|s) = 0 \implies \pi_t(a|s) = 0$, it follows immediately that $\mu_t(a|s) = 0 \implies \pi_t(a|s)u_{\pi,t}(s, a) = 0$. This indicates $\Lambda_- \subseteq \Lambda^*$. This means that the set of policies Λ^* considered in Theorem 2 are unbiased and includes at least all the policies that cover the target policy, which is the classical behavior policy search space Λ_- (Precup et al., 2000; Maei, 2011; Sutton et al., 2016; Zhang, 2022).

Unfortunately, empirically implementing μ_t^* requires knowledge of $u_{\pi,t}$ (13) that contains the transition function p. Approximating the transition function is very challenging in MDPs with large stochasticity and function approximation (cf. model-based RL (Sutton, 1990; Sutton et al., 2008; Deisenroth & Rasmussen, 2011; Chua et al., 2018)). Thus, we seek to build another policy $\hat{\mu}$ that can be easily implemented without direct knowledge of the transition function p (cf. model-free RL (Sutton, 1988; Watkins, 1989)).

We achieve this by aiming at one-step optimality instead of global optimality. We try to find the best μ_t assuming in the future we follow $\pi_{t+1}, \dots, \pi_{T-1}$, instead of $\mu_{t+1}^*, \dots, \mu_{T-1}^*$. We refer to this one-step optimal behavior policy as $\hat{\mu}_t$. Similarly, to define optimality, we first need to specify the set of policies we are concerned about. To this end, we define

$$\hat{q}_{\pi,t}(s, a) \doteq q_{\pi,t}^2(s, a) \tag{14}$$

for $t = T - 1$ and otherwise

$$\begin{aligned}
&\hat{q}_{\pi,t}(s, a) \doteq q_{\pi,t}^2(s, a) + \nu_{\pi,t}(s, a) \\
&+ \textstyle\sum_{s'} p(s'|s, a)\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t+1:T-1}^{\pi_{t+1:T-1}}) \mid S_{t+1} = s'\right).
\end{aligned} \tag{15}$$

Notably, $\hat{q}_{\pi,t}(s, a)$ is always non-negative since all the summands are non-negative. Accordingly, we define for $t \in [T - 1]$, $\hat{\Lambda}_t \doteq \{\mu_t \in \Delta(\mathcal{A}) \mid \forall s, a, \mu_t(a|s) = 0 \implies \pi_t(a|s)\hat{q}_{\pi,t}(s, a) = 0\}$. Comparing (13) and (15), the optimality of μ^* implies that $\forall s, a, t$, we have $\hat{q}_{\pi,t}(s, a) \geq u_{\pi,t}(s, a) \geq 0$. As a result, if $\mu_t \in \hat{\Lambda}_t$, we have

$$\begin{aligned}
\mu_t(a|s) = 0 &\implies \pi_t(a|s)\hat{q}_{\pi,t}(a|s) = 0 \\
&\implies \pi_t(a|s)u_{\pi,t}(a|s) = 0,
\end{aligned}$$

indicating $\mu_t \in \Lambda_t$. In other words, we have $\hat{\Lambda}_t \subseteq \Lambda_t$. To search for $\hat{\mu}_{0:T-1}$, we work on $\hat{\Lambda} \doteq \hat{\Lambda}_0 \times \cdots \times \hat{\Lambda}_{T-1}$. To summarize, we have $\Lambda_- \subseteq \hat{\Lambda} \subseteq \Lambda^* \subseteq \Lambda \subseteq \Lambda_+$. Recall that Λ_+ is the set of all behavior policies such that the corresponding PDIS estimator is unbiased. Λ is a sufficient but not necessary condition to ensure such unbiasedness (Theorem 1). Λ^* is a restriction of Λ such that we are able to find an optimal solution. We restrict Λ^* to $\hat{\Lambda}$, aiming for a sub-optimal but implementable policy. $\hat{\Lambda}$ is still larger than Λ_-, which is the space with the coverage assumption (3) that previous works (Precup et al., 2000; Maei, 2011; Sutton et al., 2016; Sutton & Barto, 2018; Zhang, 2022) consider.

After confirming the space of behavior policies, we formulate the optimization problem for designing an efficient behavior policy to achieve one-step optimality as

$$\min_{\mu_t \in \hat{\Lambda}_t} \quad \mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\{\mu_t, \pi_{t+1}, \dots, \pi_{T-1}\}}) \mid S_t = s\right). \tag{16}$$

According to the recursive expression of the variance in Lemma 4 in Appendix A.5, we rewrite (16) as

$$\begin{aligned}\min_{\mu_t \in \hat{\Lambda}_t} \mathbb{E}_{A_t \sim \mu_t} \big[\rho_t^2 \big(\mathbb{E}_{S_{t+1}} \big[\mathbb{V}\big(G^{\text{PDIS}}(\tau_{t+1:T-1}^{\pi_{t+1:T-1}}) \mid S_{t+1}\big) \\ \mid S_t, A_t\big] + \nu_{\pi,t}(S_t, A_t) + q_{\pi,t}^2(S_t, A_t)\big) \mid S_t\big],\end{aligned} \tag{17}$$

where the objective can be further simplified as

$$\begin{aligned}&\mathbb{E}_{A_t \sim \mu_t} \big[\rho_t^2 \big(\mathbb{E}_{S_{t+1}} \big[\mathbb{V}\big(G^{\text{PDIS}}(\tau_{t+1:T-1}^{\pi_{t+1:T-1}}) \mid S_{t+1}\big) \\ &\quad \mid S_t, A_t\big] + \nu_{\pi,t}(S_t, A_t) + q_{\pi,t}^2(S_t, A_t)\big) \mid S_t\big] \\ =&\mathbb{E}_{A_t \sim \mu_t} \left[\rho_t^2 \hat{q}_{\pi,t}(S_t, A_t) \mid S_t\right] \qquad \text{(By (15))} \\ =&\mathbb{V}_{A_t \sim \mu_t}\left(\rho_t \sqrt{\hat{q}_{\pi,t}(S_t, A_t)} \mid S_t\right) \\ &- \mathbb{E}_{A_t \sim \pi_t}^2 \left[\sqrt{\hat{q}_{\pi,t}(S_t, A_t)} \mid S_t\right]. \\ &\qquad \text{(Lemma 1 and } \mu_t \in \hat{\Lambda}_t)\end{aligned}$$

Since the second term is unrelated to μ_t, it is equivalent to solving

$$\min_{\mu_t \in \hat{\Lambda}_t} \quad \mathbb{V}_{A_t \sim \mu_t}\left(\rho_t \sqrt{\hat{q}_{\pi,t}(S_t, A_t)} \mid S_t\right).$$

According to Lemma 2,

$$\hat{\mu}_t(a|s) \propto \pi_t(a|s)\sqrt{\hat{q}_{\pi,t}(s,a)}. \tag{18}$$

is an optimal solution to (17). We now present our main result that $\hat{\mu}$ *provably* reduces variance.

Theorem 3 (Variance Reduction)**.** *For any t and s,*

$$\begin{aligned}&\mathbb{V}\left(G^{PDIS}(\tau_{t:T-1}^{\hat{\mu}_{t:T-1}}) \mid S_t = s\right) \\ \leq&\mathbb{V}\left(G^{PDIS}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t = s\right) - \epsilon_t(s).\end{aligned}$$

To define $\epsilon_t(s)$, first define $c_t(s) =$

$$\textstyle\sum_a \pi_t(a|s)\hat{q}_{\pi,t}(s,a) - \left(\sum_a \pi_t(a|s)\sqrt{\hat{q}_{\pi,t}(s,a)}\right)^2.$$

Then we define $\epsilon_t(s) \doteq c_t(s)$ for $t = T-1$ and otherwise

$$\epsilon_t(s) \doteq c_t(s) + \mathbb{E}_{A_t \sim \hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1})|s, A_t\right]\right]. \tag{19}$$

Its proof is in Appendix A.6. Notably, this c_t is always non-negative by Jensen's inequality, ensuring the non-negativity of ϵ_t and thus the variance reduction property. Moreover, $c_t(s) = 0$ occurs only when all actions have the same $\hat{q}_{\pi,t}$ on the state s. It is reasonable to conjecture that this is rare in practice. So, $c_t(s)$ is likely to be strictly positive. This shows the variance of the PDIS estimator with $\hat{\mu}$ at a state s is *provably* smaller than or equal to that with π, the straightforward on-policy Monte Carlo estimator, by at least $\epsilon_t(s)$. The magnitude of $\epsilon_t(s)$ depends on a specific target policy and the environment. We empirically show the variance reduction is significant in commonly used benchmarks in Section 7.

5. Learning Closed-Form Behavior Policies

We now present efficient algorithms to learn the closed-form behavior policy $\hat{\mu}$. Despite that $\hat{q}_{\pi,t}$ in (15) has a complicated definition, we prove that it has a concise representation. It is exactly the action value function of the policy π with the same transition function p but a different reward function $\hat{r}$.

Theorem 4. *Define*

$$\hat{r}_{\pi,t}(s,a) \doteq 2r(s,a)q_{\pi,t}(s,a) - r^2(s,a). \tag{20}$$

Then $\hat{q}_{\pi,t}(s,a) = \hat{r}_{\pi,t}(s,a)$ for $t = T-1$ and otherwise

$$\begin{aligned}&\hat{q}_{\pi,t}(s,a) \\ =&\hat{r}_{\pi,t}(s,a) + \textstyle\sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a').\end{aligned} \tag{21}$$

Its proof is in Appendix A.7. *This observation makes it possible to apply any off-the-shelf offline policy evaluation methods to learn $\hat{q}$,* after which the behavior policy $\hat{\mu}$ can be computed easily with (18). For generality, we consider the behavior policy agnostic offline learning setting (Nachum et al., 2019), where the offline data in the form of $\{(t_i, s_i, a_i, r_i, s_i')\}_{i=1}^m$ consists of m previously logged data tuples. In the i-th data tuple, t_i is the time step, s_i is the state at time step t_i, a_i is the action executed on state s_i, r_i is the sampled reward, and s_i' is the successor state. Those tuples can be generated by one or more, known or unknown behavior policies. Those tuples do not need to form a complete trajectory.

In this paper, we choose Fitted Q-Evaluation (FQE, Le et al. (2019)) as a demonstration, but our framework is ready to incorporate any state-of-the-art offline policy evaluation methods to approximate $\hat{q}$. To learn $\hat{r}$, it is sufficient to learn r and q. FQE can be used to learn q, and learning r is a simple regression problem. FQE is then invoked again w.r.t. the learned $\hat{r}$ to learn an approximation of $\hat{q}$. We refer the reader to Algorithm 1 for a detailed exposition of our algorithm. We split the offline data into training

Algorithm 1 Offline Data Informed (ODI) algorithm

```
Input: Estimators $r(s,a)$, $q_{\pi,t}(s,a)$, $\hat{q}_{\pi,t}(s,a)$,
  a target policy $\pi$,
  an offline dataset $\mathcal{D} = \{(t_i, s_i, a_i, r_i, s_i)\}_{i=1}^m$
Output: a behavior policy $\hat{\mu}$
Approximate $r$ from $\mathcal{D}$ using supervised learning
Approximate $q_{\pi,t}$ from $\mathcal{D}$ using any offline RL method
  (e.g. Fitted Q-Evaluation)
Compute $\hat{r}_i$ by (20) for each data pair in $\mathcal{D}$
Construct $\mathcal{D}_{\hat{r}} \doteq \{(t_i, s_i, a_i, \hat{r}_i, s_i)\}_{i=1}^m$ by plugging $\hat{r}_i$
  into $\mathcal{D}$
Approximate $\hat{q}_{\pi,t}$ from $\mathcal{D}_{\hat{r}}$ by (21) using any offline RL
  method (e.g. Fitted Q-Evaluation)
Return: $\hat{\mu}_t(a|s) \propto \pi_t(a|s)\sqrt{\hat{q}_{\pi,t}(s,a)}$
```

sets and test sets to tune all the hyperparameters offline in Algorithm 1, based on the supervised learning loss or the FQE loss on the test set. We remark that FQE loss on the test set is known to be an inaccurate signal (Fujimoto et al., 2022) so our $\hat{q}$ estimation would be poorly tuned in this sense. We, however, notice that even with such a poorly tuned estimation, the variance reduction in the tested environments is still significant. This suggests that $\epsilon_t(s)$ in Theorem 3 is likely to be large and demonstrates the robustness of our approach. Since $\hat{q}_{\pi,t}(s,a)$ is proved to be always non-negative (cf. (15)), we use positive function class for FQE in approximating $\hat{q}$, e.g., a neural network with softplus as the last activation function.

In the following, we theoretically analyze how the error in approximating $\hat{q}$ affects the amount of reduced variance in Theorem 3. We assume $\hat{q}_{\pi,t}(s,a)$ is not only non-negative but also positive. Given its non-negative summands in (15), we argue that this positivity assumption is not restrictive at all. We use $q^+_{\pi,t}(s,a) > 0$ to denote our approximation to $\hat{q}_{\pi,t}(s,a)$. The approximation error can then be captured by

$$\eta_{\pi,t}(s,a) \doteq \hat{q}^+_{\pi,t}(s,a) / \hat{q}_{\pi,t}(s,a) > 0. \tag{22}$$

If $\eta_{\pi,t}(s,a)$ is 1, there is no approximation error for (s,a,t). The actual learned behavior policy is then denoted by

$$\hat{\mu}^+_t(a|s) \propto \pi_t(a|s)\sqrt{\hat{q}^+_{\pi,t}(s,a)}. \tag{23}$$

Then, we generalize Theorem 3 to the following theorem.

Theorem 5. *For any t and s,*

$$\begin{aligned}&\mathbb{V}(G^{PDIS}(\tau_{t:T-1}^{\hat{\mu}^+_{t:T-1}}) \mid S_t = s)\\ \le&\mathbb{V}(G^{PDIS}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t = s) - \epsilon_t^+(s).\end{aligned}$$

To define $\epsilon_t^+(s)$, first define

$$\begin{aligned}c_t^+(s) \doteq& \textstyle\sum_a \pi_t(a|s)\hat{q}_{\pi,t}(s,a) - \\ &\left(\textstyle\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right)\\ &\times\left(\textstyle\sum_a \pi_t(a|S_t)\frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right).\end{aligned}$$

Then we define $\epsilon_t^+(s) \doteq c_t^+(s)$ for $t = T-1$ and otherwise

$$\begin{aligned}&\epsilon_t^+(s)\\ \doteq& c_t^+(s) + \mathbb{E}_{A_t \sim \hat{\mu}_t^+}[\rho_t^2 \mathbb{E}_{S_{t+1}}[\epsilon_{t+1}^+(S_{t+1}) | s, A_t]].\end{aligned} \tag{24}$$

Its proof is in Appendix A.8. When there is no estimation error, i.e., $\eta_{\pi,t}(s,a) = 1$, c_t^+ and ϵ_t^+ reduce to c_t and ϵ_t in Theorem 3, which is non-negative by Jensen's inequality. As discussed earlier, it is reasonable to conjecture that $c_t(s)$ is likely to be strictly positive. This leaves room to tolerate estimation errors such that $c_t^+(s)$ can still be positive even if $\eta_t(s,a) \neq 1$. Because the sign of c_t^+ only depends on the current $\eta_{\pi,t}$, the estimation error in the future step does not affect current c_t. Notably, even if some $\epsilon_{t+1}^+(S_{t+1}) < 0$, $\epsilon_t^+(S_t)$ can still be positive. This is because $\epsilon_t^+(s)$ depends on the expectation of the $\epsilon_{t+1}^+(S_{t+1})$, not a single value, and c_t^+ can still be positive. This makes our approach robust to the approximation error. *It is important to note that the PDIS estimator with $\hat{\mu}_t(a|s)$ is always unbiased, regardless of the approximation error η.*

Theorem 5 makes it straightforward to analyze how the offline data affects the amount of the reduced variance. For example, if FQE is used, one can resort to Munos (2003); Antos et al. (2008); Munos & Szepesvári (2008); Chen & Jiang (2019) to connect offline data and the approximation error η. Theorem 5 then directly relays the approximation error to the amount of reduced variance. We, however, omit such analysis since it deviates from our main contribution.

6. Related Work

Monte Carlo methods. Reducing the variance of Monte Carlo estimators via learning a proper behavior policy has been explored before. Hanna et al. (2017) model the problem of finding a variance-reducing behavior policy as an optimization problem and thus rely on stochastic gradient descent to update a parameterized behavior policy. In particular, Hanna et al. (2017) consider the ordinary importance sampling. *By contrast, we consider the per-decision importance sampling, which is fundamentally better (Precup et al., 2000).* Moreover, Hanna et al. (2017) require new online data to learn this behavior policy. *By contrast, our method works with offline data and does not need any online data for behavior policy learning.* Hanna et al. (2017) also require the online data to be complete trajectories. *By contrast, our method copes well with incomplete offline tuples.* Mukherjee et al. (2022) also investigate variance-reducing behavior policies for the per-decision importance sampling estimator. Their results, however, apply to only tree-structured MDPs,

	MDP	Data to learn μ	Parameterization of π	Gridworld size	Other environments
Ours	**general**	**offline data**	**no assumption**	**27,000**	**MuJoCo robotics**
BPS (Hanna et al., 2017)	**general**	online data	need to be known	1,600	CartPole, Acrobot
ROS (Zhong et al., 2022)	**general**	online data	need to be known	1,600	CartPole
ReVar (Mukherjee et al., 2022)	tree	**offline data**	**no assumption**	1,600	15 states tree-MDP

Table 1. Our methods impose weaker assumptions on the data, and our empirical study covers more challenging tasks.

which is rather restrictive because many MDPs of interest are not tree-structured. For example, in finite horizon MDPs considered in this paper, if two states at time t can transit to the same successor state at time $t + 1$, then this MDP is not tree-structured. Moreover, Mukherjee et al. (2022) require to directly approximate the transition function of the MDP by counting, making it essentially a model-based approach. Mukherjee et al. (2022), therefore, suffer from all canonical challenges in model learning (Sutton, 1990; Sutton et al., 2008; Deisenroth & Rasmussen, 2011; Chua et al., 2018). *By contrast, we work on general MDPs without making any assumption regarding their underlying structures, and we do not need to approximate the transition function. Our approach is model-free.* Zhong et al. (2022) adjust the behavior policy by encouraging under-sampled data. Their offline data, however, has to be complete trajectories generated by known policies. In their experiments, they also require the policies for generating offline data to be similar to the target policy since they do not have any importance sampling. *By contrast, our method copes well with offline data in the form of incomplete segments from probably unknown behavior policies that can be arbitrarily different from the target policy*. Moreover, there is no theoretical guarantee that the estimates made by Zhong et al. (2022) are unbiased or consistent. *By contrast, our estimate is always provably unbiased*.

Other attempts for variance reduction in Monte Carlo evaluation mostly use control variates based on value functions (Zinkevich et al., 2006; White & Bowling, 2009; Jiang & Li, 2016). Such control variates can be integrated into our estimator, which we, however, save for future work. Notably, our work differs from the doubly robust method in Jiang & Li (2016) in that they assume the behavior policy is fixed and given while we use the fact that we have the freedom to choose a behavior policy in many settings. Moreover, to account for the stochasticity from the transition function, they require to learn a model of the MDP accurately, while we achieve this in a model-free way. Finally, they do not confirm a reduced variance compared with the on-policy estimator while we do.

Model-based offline evaluation. One straightforward way to exploit offline data for policy evaluation is to learn a model of the MDP first, probably with supervised learning (Jiang & Li, 2016; Paduraru, 2013; Zhang et al., 2021), and then execute Monte Carlo methods inside the learned model. Learning a high-fidelity model is, however, sometimes even more challenging than evaluating the policy itself (Li, 2019). And the model prediction error can easily compound over time steps during model rollouts (Wan et al., 2019). *Nevertheless, if a good model could somehow be learned, our work still helps reduce the required rollouts when Monte Carlo is applied within the learned model.*

Model-free offline evaluation. Model-free offline evaluation methods rely on learning other quantities for policy evaluation, including density ratio (a.k.a. marginalized importance sampling ratio, Liu et al. (2018); Nachum et al. (2019); Li (2019); Xie et al. (2019); Zhang et al. (2020); Mousavi et al. (2020); Uehara et al. (2020); Yang et al. (2020)) and state-action value function (Harutyunyan et al., 2016; Munos et al., 2016; Farajtabar et al., 2018; Le et al., 2019; Precup et al., 2000). But those learning processes bring in bias, either due to the misspecification of the function class or due to the complexity of optimization. Consequently, the estimation they make is biased, and it is hard to quantify such bias without restrictive assumptions. *To our knowledge, the only practical way in general settings to certify that their estimation is indeed accurate is to compare those estimations with Monte Carlo estimations.*

Furthermore, those learning algorithms also have hyperparameters to tune (i.e., model selection), for which most offline RL practitioners (see, e.g., Liu et al. (2018); Nachum et al. (2019); Li (2019); Xie et al. (2019); Mousavi et al. (2020); Uehara et al. (2020); Yang et al. (2020); Zhang et al. (2020)) usually use Monte Carlo with online data. The online data comes from either a simulator or a learned model. As a result, *our work helps reduce the online data used in model selection by those model-free offline evaluation methods.*

Efforts have been made to perform model selection with only offline data without explicitly learning a model as well (Paine et al., 2020; Kumar et al., 2021; Xie & Jiang, 2021; Zhang & Jiang, 2021). Those offline model selection methods, however, rarely have a correctness guarantee without restrictive assumptions. To summarize, if obtaining online data is entirely impossible, existing offline evaluation methods without using any online data might be the only choices. These include model-based methods and model-free methods augmented by offline model selection. However, in many scenarios, it is practical to assume that a small amount of online data is available. If, in addition, evaluation correctness should be honored, then the improved Monte Carlo method in this work might be a better choice. Using offline data to help online model selection is previously explored by Konyushova et al. (2021). In particular, they use offline data to decide which policy, among a given set of policies, should be given priority to evaluate. When it comes to the actual online evaluation, Konyushova et al. (2021) still uses the ordinary online Monte Carlo methods. *Konyushova et al. (2021), therefore, again benefit from the improved Monte Carlo method in this paper.*

7. Empirical Results

In this section, we present empirical results comparing our methods against three baselines: **(1)** the canonical on-policy Monte Carlo estimator, **(2)** off-policy Monte Carlo estimator with behavior policy search (BPS, Hanna et al. (2017)), and **(3)** robust on-policy sampling (ROS, Zhong et al. (2022)). We do not implement ReVar (Mukherjee et al., 2022) because it will incur infinite loops if the MDP is not tree-structured. Our method first learns a behavior policy with given offline data using Algorithm 1, then the PDIS Monte Carlo estimator (1) is used to estimate the performance of the target policy, where the learned behavior policy is used to interact with the environment. We call our method Offline Data Informed (ODI) algorithm. Our implementation is made publicly available to facilitate future research[1]. Our method is superior in data requirements and applicability as summarized in Table 1.

Gridworld: We first conduct experiments with linear function approximation in Gridworld with n^3 states, i.e., it is an $n \times n$ grid with the time horizon also being n. Specifically, we use Gridworld with $n^3 = 1,000$ and $n^3 = 27,000$. We use randomly generated reward functions with 30 randomly generated target policies. The offline data is generated by selecting random actions on uniformly random state distribution. We report the *normalized estimation error* of the four methods against the number of environment interactions (steps). Intuitively, this normalized estimation error is the estimation error of an estimator normalized by that of the on-policy Monte Carlo estimator. Precisely speaking, define the *estimation error* at step t as the absolute difference between an estimator and the ground truth divided by the ground truth. The *normalized estimation error* is then the estimation error divided by the average estimation error of the on-policy Monte Carlo estimator after the first episode. Thus, the normalized estimation error of the on-policy Monte Carlo estimator starts from 1.

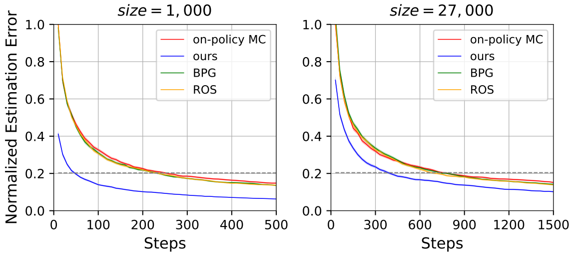

Figure 1. Results on Gridworld. The curves are averaged over 900 trials (30 target policies, each having 30 independent runs). The shaded regions denote standard errors and are invisible for some curves because they are too small.

As shown in Figure 1, our method outperforms baselines by a large margin. In particular, as shown by the dotted line, in Gridworld with size $1,000$, to achieve the same estimation error that the on-policy Monte Carlo estimator achieves with 250 steps, our methods only need around 50 steps. In Gridworld with size $27,000$, to achieve the same estimation error that on-policy Monte Carlo estimator achieves with 750 steps, our methods only need around 400 steps, saving more than 40% of online iteractions. The improvement in environments with size $= 27,000$ is smaller than environments with size $= 1,000$ because the amount of offline data is the same for both environments, i.e., the offline data coverage is worse for the Gridworld with size $= 27,000$. In fact, the offline data coverage for the Gridworld with size $= 1,000$ and size $= 27,000$ are 62.5% and 2.3%, respectively. More experiment details are in Appendix B.1.

On-policy MC	Ours with 2.3% offline data coverage	Ours with 4.6% offline data coverage	Ours with 18.4% offline data coverage	BPG	ROS
300	150	90	60	300	300
600	330	180	120	540	540
1200	540	420	270	990	990

Table 2. The above table is an extension of Figure 1 by adding experiments with $4.6\%/18.4\%$ data coverage for our algorithm in Gridworld with size $= 27,000$. Each number is the number of steps needed to achieve the same estimation accuracy that the naive Monte Carlo achieves with $300/600/1200$ steps. All numbers are averaged from 900 different runs over a wide range of policies. Standard errors are visualized in Figure 1 of our paper and are invisible for some algorithm curves because they are too small.

---

[1] https://github.com/ShuzeLiu/Behavior-Policy-Design-for-Policy-Evaluation

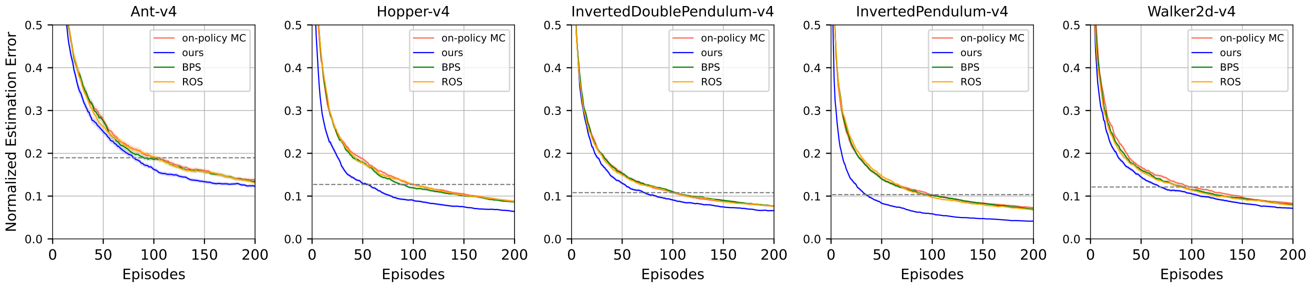

Figure 2. Results on Mujoco environments. Each curve is averaged over 900 trials (30 target policies, each having 30 independent runs). The shaded regions denote standard errors and are invisible for some curves because they are too small.

	On-policy MC	Ours	BPG	ROS	Improvement in Saved Episodes
Ant	100	**81**	91	103	(100-81)/(100-91)≈211.1%
Hopper	100	**54**	89	100	(100-54)/(100-89)≈418.2%
I. Pendulum	100	**72**	103	99	(100-72)/(100-99)=2800%
I. D. Pendulum	100	**35**	95	90	(100-35)/(100-90)=650%
Walker	100	**70**	92	91	(100-70)/(100-91)≈333.3%

Table 3. Episodes needed to achieve the same of estimation accuracy that on-policy Monte Carlo achieves with 100 episodes.

We also show our algorithm scales with offline data. As we increase the data coverage in the Gridwolrd with size $= 27,000$ by adding more offline data generated from many different distributions, our method improves the saved samples from $55\% = (1200 - 540)/1200$ to $77.5\% = (1200 - 270)/1200$ in the last row of Table 2. By comparison, the best over all previous state-of-the-art algorithms only saves $17.5\% = (1200 - 990)/1200$ samples and do not have a mechanism to use offline data because they can only utilize online trajectory.

MuJoCo: We then conduct experiments with neural network function approximation in MuJoCo (Todorov et al., 2012) robot simulation tasks. Since our methods are designed for discrete action space, we discretize the MuJoCo action space. Details about action space discretization, target policy generation, and offline data generation are provided in Appendix B.2. We report the normalized estimator error in Figure 2, where our methods are consistently better than baselines. In particular, as shown by the dotted line in Figure 2 and Table 3, our methods need much fewer episodes (save up to 65% episodes) to achieve the estimation error that the on-policy Monte Carlo estimator achieves with 100 episodes. Recognizing episodes may have different lengths in MuJoCo, we also provide in Appendix B.2 a version of Figure 2 with the x-axis being steps, where our methods are still consistently better.

It is worth mentioning that all hyperparameters of our methods required to learn $\hat{\mu}$ are tuned offline and are the same across all MuJoCo and Gridworld experiments.

8. Conclusion

Monte Carlo methods are the most dominant approach for evaluating a policy. The development and deployment of almost all RL algorithms, including offline RL algorithms, implicitly or explicitly depend on Monte Carlo methods more or less. For example, when an RL researcher wants to plot a curve of the agent performance against training steps, Monte Carlo methods are usually the first choice. Our method improves the online data efficiency of Monte Carlo evaluation while maintaining its unbiasedness by learning a tailored behavior policy from offline data. The two main contributions are the provably better closed-form behavior policy (Theorem 3) and its alternative representation (Theorem 4). Extending them to temporal difference learning (Sutton, 1988) is a possible future work.

Acknowledgements

We thank Yuxin Chen for warm revising comments and Haifeng Xu and Zhengkun Xiao for insightful discussions. This work is supported in part by the US National Science Foundation under grants III-2128019 and SLES-2331904.

Impact Statement

This paper advances the field of reinforcement learning and machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 2008.

Benjamin Melamed, R. Y. R. *Modern Simulation and Modeling (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 1998.

Carpentier, A., Munos, R., and Antos, A. Adaptive strategy for stratified monte carlo sampling. *Journal of Machine Learning Research*, 2015.

Chen, J. and Jiang, N. Information-theoretic considerations in batch reinforcement learning. In *Proceedings of International Conference on Machine Learning*, 2019.

Chervonyi, Y., Dutta, P., Trochim, P., Voicu, O., Paduraru, C., Qian, C., Karagozler, E., Davis, J. Q., Chippendale, R., Bajaj, G., et al. Semi-analytical industrial cooling system model for reinforcement learning. *arXiv preprint arXiv:2207.13131*, 2022.

Chua, K., Calandra, R., McAllister, R., and Levine, S. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In *Advances in Neural Information Processing Systems*, 2018.

Deisenroth, M. P. and Rasmussen, C. E. PILCO: A model-based and data-efficient approach to policy search. In *Proceedings of the International Conference on Machine Learning*, 2011.

Ernst, D., Geurts, P., and Wehenkel, L. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 2005.

Farajtabar, M., Chow, Y., and Ghavamzadeh, M. More robust doubly robust off-policy evaluation. In *Proceedings of the International Conference on Machine Learning*, 2018.

Fu, J., Kumar, A., Nachum, O., Tucker, G., and Levine, S. D4RL: datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Fujimoto, S., Meger, D., and Precup, D. Off-policy deep reinforcement learning without exploration. In *Proceedings of the International Conference on Machine Learning*, 2019.

Fujimoto, S., Meger, D., Precup, D., Nachum, O., and Gu, S. S. Why should i trust you, bellman? the bellman error is a poor replacement for value error. *arXiv preprint arXiv:2201.12417*, 2022.

Geweke, J. Antithetic acceleration of monte carlo integration in bayesian inference. *Journal of Econometrics*, 1988.

Gülçehre, Ç., Wang, Z., Novikov, A., Paine, T., Colmenarejo, S. G., Zolna, K., Agarwal, R., Merel, J., Mankowitz, D. J., Paduraru, C., Dulac-Arnold, G., Li, J., Norouzi, M., Hoffman, M., Heess, N., and de Freitas, N. RL unplugged: A collection of benchmarks for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, 2020.

Hanna, J. P., Thomas, P. S., Stone, P., and Niekum, S. Data-efficient policy evaluation through behavior policy search. In *Proceedings of the International Conference on Machine Learning*, 2017.

Harutyunyan, A., Bellemare, M. G., Stepleton, T., and Munos, R. Q(λ) with off-policy corrections. In *Proceedings of the International Conference on Algorithmic Learning Theory*, 2016.

Hesterberg, T. Weighted average importance sampling and defensive mixture distributions. *Technometrics*, 1995.

Huang, S., Dossa, R. F. J., Ye, C., Braga, J., Chakraborty, D., Mehta, K., and Araújo, J. G. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 2022.

Jiang, N. and Li, L. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2016.

Kahn, H. and Marshall, A. W. Methods of reducing sample size in monte carlo computations. *Journal of the Operations Research Society of America*, 1953.

Kakutani, S. Markoff process and the dirichlet problem. In *Proceedings of the Japan Academy*, 1945.

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

Koller, D. and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. Mit Press, 2009.

Konyushova, K., Chen, Y., Paine, T., Gulcehre, C., Paduraru, C., Mankowitz, D. J., Denil, M., and de Freitas, N. Active offline policy selection. In *Advances in Neural Information Processing Systems*, 2021.

Kumar, A., Singh, A., Tian, S., Finn, C., and Levine, S. A workflow for offline model-free robotic reinforcement learning. In *Proceedings of the Annual Conference on Robot Learning*, 2021.

Lange, S., Gabel, T., and Riedmiller, M. Batch reinforcement learning. *Reinforcement learning: State-of-the-art*, 2012.

Le, H. M., Voloshin, C., and Yue, Y. Batch policy learning under constraints. In *Proceedings of the International Conference on Machine Learning*, 2019.

Levine, S., Kumar, A., Tucker, G., and Fu, J. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Li, L. A perspective on off-policy evaluation in reinforcement learning. *Frontiers of Computer Science*, 2019.

Liu, Q., Li, L., Tang, Z., and Zhou, D. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, 2018.

Maei, H. R. *Gradient temporal-difference learning algorithms*. PhD thesis, University of Alberta, 2011.

Mathieu, M., Ozair, S., Srinivasan, S., Gulcehre, C., Zhang, S., Jiang, R., Paine, T. L., Powell, R., Żołna, K., Schrittwieser, J., Choi, D., Georgiev, P., Toyama, D., Huang, A., Ring, R., Babuschkin, I., Ewalds, T., Bordbar, M., Henderson, S., Colmenarejo, S. G., van den Oord, A., Czarnecki, W. M., de Freitas, N., and Vinyals, O. Alphastar unplugged: Large-scale offline reinforcement learning. *arXiv preprint arXiv:2308.03526*, 2023.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M. A., Fidjeland, A., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 2015.

Mousavi, A., Li, L., Liu, Q., and Zhou, D. Black-box off-policy estimation for infinite-horizon reinforcement learning. In *Proceedings of the International Conference on Learning Representations*, 2020.

Mukherjee, S., Hanna, J. P., and Nowak, R. D. Revar: Strengthening policy evaluation via reduced variance sampling. In *Proceedings of the Conference in Uncertainty in Artificial Intelligence*, 2022.

Munos, R. Error bounds for approximate policy iteration. In *Proceedings of the International Conference on Machine Learning*, 2003.

Munos, R. and Szepesvári, C. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 2008.

Munos, R., Stepleton, T., Harutyunyan, A., and Bellemare, M. Safe and efficient off-policy reinforcement learning. In *Advances in Neural Information Processing Systems*, 2016.

Nachum, O., Chow, Y., Dai, B., and Li, L. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. In *Advances in Neural Information Processing Systems*, 2019.

O'Donoghue, B., Osband, I., Munos, R., and Mnih, V. The uncertainty bellman equation and exploration. In *Proceedings of the International Conference on Machine Learning*, 2018.

Owen, A. B. *Monte Carlo theory, methods and examples*. 2013.

Paduraru, C. *Off-policy evaluation in Markov decision processes*. PhD thesis, McGill University, 2013.

Paine, T. L., Paduraru, C., Michi, A., Gulcehre, C., Zolna, K., Novikov, A., Wang, Z., and de Freitas, N. Hyperparameter selection for offline reinforcement learning. *arXiv preprint arXiv:2007.09055*, 2020.

Precup, D., Sutton, R. S., and Singh, S. P. Eligibility traces for off-policy policy evaluation. In *Proceedings of the International Conference on Machine Learning*, 2000.

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

Rubinstein, R. Y. *Simulation and the Monte Carlo Method*. Wiley, 1981.

Schrittwieser, J., Hubert, T., Mandhane, A., Barekatain, M., Antonoglou, I., and Silver, D. Online and offline reinforcement learning by planning with a learned model. In *Advances in Neural Information Processing Systems*, 2021.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Sherstan, C., Bennett, B., Young, K., Ashley, D. R., White, A., White, M., and Sutton, R. S. Directly estimating the variance of the λ-return using temporal-difference methods. *arXiv preprint arXiv:1801.08287*, 2018.

Sutton, R. S. Learning to predict by the methods of temporal differences. *Machine Learning*, 1988.

Sutton, R. S. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the International Conference on Machine Learning*, 1990.

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

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, 1999.

Sutton, R. S., Szepesvári, C., Geramifard, A., and Bowling, M. H. Dyna-style planning with linear function approximation and prioritized sweeping. In *Proceedings of the Conference in Uncertainty in Artificial Intelligence*, 2008.

Sutton, R. S., Mahmood, A. R., and White, M. An emphatic approach to the problem of off-policy temporal-difference learning. *Journal of Machine Learning Research*, 2016.

Tamar, A., Castro, D. D., and Mannor, S. Learning the variance of the reward-to-go. *Journal of Machine Learning Research*, 2016.

Thomas, P. S. *Safe reinforcement learning*. PhD thesis, University of Massachusetts Amherst, 2015.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *Proceedings of the International Conference on Intelligent Robots and Systems*, 2012.

Uehara, M., Huang, J., and Jiang, N. Minimax weight and q-function learning for off-policy evaluation. In *Proceedings of the International Conference on Machine Learning*, 2020.

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., Oh, J., Horgan, D., Kroiss, M., Danihelka, I., Huang, A., Sifre, L., Cai, T., Agapiou, J. P., Jaderberg, M., Vezhnevets, A. S., Leblond, R., Pohlen, T., Dalibard, V., Budden, D., Sulsky, Y., Molloy, J., Paine, T. L., Gülçehre, Ç., Wang, Z., Pfaff, T., Wu, Y., Ring, R., Yogatama, D., Wünsch, D., McKinney, K., Smith, O., Schaul, T., Lillicrap, T. P., Kavukcuoglu, K., Hassabis, D., Apps, C., and Silver, D. Grandmaster level in starcraft II using multi-agent reinforcement learning. *Nature*, 2019.

Wan, Y., Zaheer, M., White, A., White, M., and Sutton, R. S. Planning with expectation models. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2019.

Watkins, C. J. and Dayan, P. Q-learning. *Machine Learning*, 1992.

Watkins, C. J. C. H. *Learning from delayed rewards*. PhD thesis, King's College, Cambridge, 1989.

White, M. and Bowling, M. H. Learning a value analysis tool for agent evaluation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2009.

Xie, T. and Jiang, N. Batch value-function approximation with only realizability. In *Proceedings of the International Conference on Machine Learning*, 2021.

Xie, T., Ma, Y., and Wang, Y. Towards optimal off-policy evaluation for reinforcement learning with marginalized importance sampling. In *Advances in Neural Information Processing Systems*, 2019.

Yang, M., Nachum, O., Dai, B., Li, L., and Schuurmans, D. Off-policy evaluation via the regularized lagrangian. In *Advances in Neural Information Processing Systems*, 2020.

Zhang, M. R., Paine, T. L., Nachum, O., Paduraru, C., Tucker, G., Wang, Z., and Norouzi, M. Autoregressive dynamics models for offline policy evaluation and optimization. In *Proceedings of the International Conference on Learning Representations*, 2021.

Zhang, S. *Breaking the deadly triad in reinforcement learning*. PhD thesis, University of Oxford, 2022.

Zhang, S. A new challenge in policy evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Zhang, S. and Jiang, N. Towards hyperparameter-free policy selection for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, 2021.

Zhang, S., Liu, B., and Whiteson, S. GradientDICE: Rethinking generalized offline estimation of stationary values. In *Proceedings of the International Conference on Machine Learning*, 2020.

Zhong, R., Zhang, D., Schäfer, L., Albrecht, S. V., and Hanna, J. P. Robust on-policy sampling for data-efficient policy evaluation in reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022.

Zinkevich, M., Bowling, M., Bard, N., Kan, M., and Billings, D. Optimal unbiased estimators for evaluating agent performance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2006.

A. Proofs

A.1. Proof of Lemma 1

Proof.

$$\begin{aligned}
\mathbb{E}_{A\sim\mu}\left[\rho(A)q(A)\right] =& \sum_{a\in\{a|\mu(a)>0\}} \mu(a)\frac{\pi(a)}{\mu(a)}q(a) \\
=& \sum_{a\in\{a|\mu(a)>0\}} \pi(a)q(a) \\
=& \sum_{a\in\{a|\mu(a)>0\}} \pi(a)q(a) + \sum_{a\in\{a|\mu(a)=0\}} \pi(a)q(a) && (\mu \in \Lambda) \\
=& \sum_{a} \pi(a)q(a) \\
=& \mathbb{E}_{A\sim\pi}\left[q(A)\right].
\end{aligned}$$

The intuition in the third equation is that the sample a where μ does not cover π must satisfy $q(a) = 0$, i.e., this sample does not contribute to the expectation anyway. ☐

A.2. Proof of Lemma 2

Proof.
For a given π and q, define

$$\mathcal{A}_+ \doteq \{a \mid \pi(a)q(a) \neq 0\}.$$

For any $\mu \in \Lambda$, we expand the variance as

$$\begin{aligned}
& \mathbb{V}_{A\sim\mu}(\rho(A)q(A)) \\
=& \mathbb{E}_{A\sim\mu}[(\rho(A)q(A))^2] - \mathbb{E}^2_{A\sim\mu}[\rho(A)q(A)] \\
=& \mathbb{E}_{A\sim\mu}[(\rho(A)q(A))^2] - \mathbb{E}^2_{A\sim\pi}[q(A)] && \text{(Lemma 1)} \\
=& \sum_{a\in\{a|\mu(a)>0\}} \frac{\pi^2(a)q^2(a)}{\mu(a)} - \mathbb{E}^2_{A\sim\pi}[q(A)] \\
=& \sum_{a\in\{a|\mu(a)>0\}\cap\mathcal{A}_+} \frac{\pi^2(a)q^2(a)}{\mu(a)} - \mathbb{E}^2_{A\sim\pi}[q(A)] && (\pi(a)q(a) = 0, \forall a \notin \mathcal{A}_+) \\
=& \sum_{a\in\mathcal{A}_+} \frac{\pi^2(a)q^2(a)}{\mu(a)} - \mathbb{E}_{A\sim\pi}[q(A)]^2 && (\mu \in \Lambda)
\end{aligned}$$

The second term is a constant and is unrelated to μ. Solving the optimization problem (8) is, therefore, equivalent to solving

$$\min_{\mu\in\Lambda} \quad \sum_{a\in\mathcal{A}_+} \frac{\pi^2(a)q^2(a)}{\mu(a)}. \tag{25}$$

Case 1: $|\mathcal{A}_+| = 0$
In this case, the variance is always 0 so any $\mu \in \Lambda$ is optimal. In particular, $\mu^*(a) = \frac{1}{\mathcal{A}}$ is optimal.
Case 2: $|\mathcal{A}_+| > 0$
The definition of Λ in (7) can be equivalently expressed, using contraposition, as

$$\Lambda = \{\mu \in \Delta(\mathcal{A}) \mid \forall a, a \in \mathcal{A}_+ \implies \mu(a) > 0\}.$$

The optimization problem (25) can then be equivalently written as

$$\min_{\mu\in\Delta(\mathcal{A})} \quad \sum_{a\in\mathcal{A}_+} \frac{\pi^2(a)q^2(a)}{\mu(a)} \tag{26}$$

$$\text{s.t.} \quad \mu(a) > 0 \quad \forall a \in \mathcal{A}_+.$$

If for some μ we have $\sum_{a \in \mathcal{A}_+} \mu(a) < 1$, then there must exist some $a_0 \notin \mathcal{A}_+$ such that $\mu(a_0) > 0$. Since a_0 does not contribute to the summation in the objective function of (26), we can move the probability mass on a_0 to some other $a_1 \in \mathcal{A}_+$ to increase $\mu(a_1)$ to further decrease the objective. In other words, any optimal solution μ to (26) must put all its mass on $\mathcal{A}_+$. This motivates the following problem

$$\begin{aligned} \min\nolimits_{z \in \Delta(\mathcal{A}_+)} \quad & \sum_{a \in \mathcal{A}_+} \frac{\pi^2(a) q^2(a)}{z(a)} \\ \text{s.t.} \quad & z(a) > 0 \quad \forall a \in \mathcal{A}_+. \end{aligned} \tag{27}$$

In particular, if z_* is an optimal solution to (27), then an optimal solution to (26) can be constructed as

$$\mu_*(a) = \begin{cases} z_*(a) & a \in \mathcal{A}_+ \\ 0 & \text{otherwise}. \end{cases} \tag{28}$$

Let $\mathbb{R}_{++} \doteq (0, +\infty)$.

According to the Cauchy-Schwarz inequality, for any $z \in \mathbb{R}_{++}^{|\mathcal{A}_+|}$, we have

$$\left(\sum_{a \in \mathcal{A}_+} \frac{\pi^2(a) q^2(a)}{z(a)} \right) \left(\sum_{a \in \mathcal{A}_+} z(a) \right) \geq \left(\sum_{a \in \mathcal{A}_+} \frac{\pi(a) |q(a)|}{\sqrt{z(a)}} \sqrt{z(a)} \right)^2 = \left(\sum_{a \in \mathcal{A}_+} \pi(a) |q(a)| \right)^2.$$

It can be easily verified that the equality holds for

$$z^*(a) \doteq \frac{\pi(a)|q(a)|}{\sum_b \pi(b)|q(b)|} > 0.$$

Since $\sum_{a \in \mathcal{A}_+} z^*(a) = 1$, we conclude that z^* is an optimal solution to (27). An optimal solution μ_* to (8) can then be constructed according to (28). Making use of the fact that $\pi(a)|q(a)| = 0$ for $a \notin \mathcal{A}_+$, this μ_* can be equivalently expressed as

$$\mu_*(a) = \frac{\pi(a)|q(a)|}{\sum_{b \in \mathcal{A}} \pi(b) q(b)},$$

which completes the proof. □

A.3. Proof of Lemma 3

Proof. We start by showing $\Lambda = \Lambda_+$. Lemma 1 ensures that $\mu \in \Lambda \implies \mu \in \Lambda_+$. We now show that $\mu \in \Lambda_+ \implies \mu \in \Lambda$. For any $\mu \in \Lambda_+$, we have

$$\sum_{a \in \{a | \mu(a) > 0\}} \mu(a) \frac{\pi(a)}{\mu(a)} q(a) = \sum_a \pi(a) q(a).$$

This indicates that

$$\sum_{a \in \{a | \mu(a) = 0\}} \pi(a) q(a) = 0.$$

Since $\pi(a) \geq 0$ and all $q(a)$ has the same sign, we must have

$$\pi(a) q(a) = 0, \forall a \in \{a \mid \mu(a) = 0\}.$$

This is exactly $\mu(a) = 0 \implies \pi(a) q(a) = 0$, yielding $\mu \in \Lambda$. This completes the proof of $\Lambda_+ = \Lambda$.

We now show the zero variance. When $\forall a \in \mathcal{A}, q(a) \geq 0$, if $\exists a_0, \pi_0(a_0)q(a_0) \neq 0$, we have $\forall a \in \mathcal{A}$

$$\mu^*(a) = \frac{\pi(a)|q(a)|}{c}$$

and $c > 0$ is a normalizing constant. Plugging μ^* to $\rho(A)q(A)$, we get $\forall a \in \mathcal{A}$

$$\rho(a)q(a) = \frac{\pi(a)}{\mu^*(a)}q(a) = \frac{\pi(a)}{\frac{\pi(a)|q(a)|}{c}}q(a) = c.$$

This means in this setting, with the optimal distribution μ^*, the random variable $\rho(\cdot)q(\cdot)$ is a constant function. Thus,

$$\mathbb{V}_{A\sim\mu^*}(\rho(A)q(A)) = 0.$$

When $\forall a \in \mathcal{A}, q(a) \geq 0$, if $\forall a_0, \pi_0(a_0)q(a_0) = 0$, we have $\forall a \in \mathcal{A}$

$$\mu^*(a) = \frac{1}{|\mathcal{A}|}.$$

Plugging μ^* to $\rho(A)q(A)$, we get $\forall a \in \mathcal{A}$

$$\rho(a)q(a) = \frac{\pi(a)}{\mu^*(a)}q(a) = \frac{\pi(a)q(a)}{\frac{1}{|\mathcal{A}|}} = 0.$$

This shows $\rho(A)q(A)$ is also a constant. Thus,

$$\mathbb{V}_{A\sim\mu^*}(\rho(A)q(A)) = 0.$$

The proof is similar for $\forall a \in \mathcal{A}, q(a) \leq 0$ and is thus omitted.

□

A.4. Proof of Theorem 1

Proof. We proceed via induction. For $t = T-1$, we have

$$\begin{aligned}
\mathbb{E}\left[G^{\text{PDIS}}(\tau_{t:T-1}^{\mu_{t:T-1}}) \mid S_t\right] =& \mathbb{E}\left[\rho_t R_{t+1} \mid S_t\right] = \mathbb{E}\left[\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right] \\
=& \mathbb{E}_{A_t\sim\pi_t(\cdot|S_t)}\left[q_{\pi,t}(S_t, A_t) \mid S_t\right] && \text{(Lemma 1)} \\
=& v_{\pi,t}(S_t).
\end{aligned}$$

For $t \in [T-2]$, we have

$$\begin{aligned}
&\mathbb{E}\left[G^{\text{PDIS}}(\tau_{t:T-1}^{\mu_{t:T-1}}) \mid S_t\right] \\
=&\mathbb{E}\left[\rho_t R_{t+1} + \rho_t G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu_{t+1:T-1}}) \mid S_t\right] \\
=&\mathbb{E}\left[\rho_t R_{t+1} \mid S_t\right] + \mathbb{E}\left[\rho_t G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu_{t+1:T-1}}) \mid S_t\right] \\
=&\mathbb{E}\left[\rho_t R_{t+1} \mid S_t\right] + \mathbb{E}_{A_t\sim\mu_t(\cdot|S_t), S_{t+1}\sim p(\cdot|S_t,A_t)}\left[\mathbb{E}\left[\rho_t G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu_{t+1:T-1}}) \mid S_t, A_t, S_{t+1}\right] \mid S_t\right] && \text{(Law of total expectation)} \\
=&\mathbb{E}\left[\rho_t R_{t+1} \mid S_t\right] + \mathbb{E}_{A_t\sim\mu_t(\cdot|S_t), S_{t+1}\sim p(\cdot|S_t,A_t)}\left[\rho_t \mathbb{E}\left[G^{\text{PDIS}}(\tau_{t+1:T-1}^{\mu_{t+1:T-1}}) \mid S_{t+1}\right] \mid S_t\right] \\
& && \text{(Conditional independence and Markov property)} \\
=&\mathbb{E}\left[\rho_t R_{t+1} \mid S_t\right] + \mathbb{E}_{A_t\sim\mu_t(\cdot|S_t), S_{t+1}\sim p(\cdot|S_t,A_t)}\left[\rho_t v_{\pi,t+1}(S_{t+1}) \mid S_t\right] && \text{(Inductive hypothesis)} \\
=&\mathbb{E}_{A_t\sim\mu_t(\cdot|S_t)}\left[\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right] && \text{(Definition of } q_{\pi,t}\text{)} \\
=&\mathbb{E}_{A_t\sim\pi_t(\cdot|S_t)}\left[q_{\pi,t}(S_t, A_t) \mid S_t\right] && \text{(Lemma 1)} \\
=&v_{\pi,t}(S_t),
\end{aligned}$$

which completes the proof. □

A.5. Proof of Theorem 2

To prove Theorem 2, we rely on a recursive expression of the PDIS Monte Carlo estimator summarized by the following lemma.

Lemma 4 (Recursive Expression of Variance)**.** *For any $\mu \in \Lambda$, for $t = T-1$,*

$$\mathbb{V}\left(G^{\textit{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right) = \mathbb{E}_{A_t \sim \mu_t}\left[\rho_t^2 q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t),$$

for $t \in [T-2]$,

$$\begin{aligned}
&\mathbb{V}\left(G^{\textit{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{E}_{A_t \sim \mu_t}\left[\rho_t^2 \left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\textit{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t\right) \mid S_t, A_t\right] + \nu_{\pi,t}(S_t, A_t) + q^2_{\pi,t}(S_t, A_t)\right) \mid S_t\right] \\
&- v^2_{\pi,t}(S_t).
\end{aligned}$$

Proof. When $t \in [T-2]$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right) && (29) \\
=&\mathbb{E}_{A_t}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t, A_t\right) \mid S_t\right] + \mathbb{V}_{A_t}\left(\mathbb{E}\left[G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t, A_t\right] \mid S_t\right) && \text{(Law of total variance)} \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \mathbb{V}\left(r(S_t, A_t) + G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right) \mid S_t\right] \\
&+ \mathbb{V}_{A_t}\left(\rho_t \mathbb{E}\left[r(S_t, A_t) + G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right] \mid S_t\right) && \text{(Using (2))} \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right) \mid S_t\right] + \mathbb{V}_{A_t}\left(\rho_t \mathbb{E}\left[r(S_t, A_t) + G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right] \mid S_t\right) \\
& && \text{(Deterministic reward } r\text{)} \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right) \mid S_t\right] + \mathbb{V}_{A_t}\left(\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right).
\end{aligned}$$

Further decomposing the first term, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t\right) && (30) \\
=&\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t, S_{t+1}\right) \mid S_t, A_t\right] \\
&+ \mathbb{V}_{S_{t+1}}\left(\mathbb{E}\left[G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_t, A_t, S_{t+1}\right] \mid S_t, A_t\right) && \text{(Law of total variance)} \\
=&\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \mathbb{V}_{S_{t+1}}\left(\mathbb{E}\left[G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right] \mid S_t, A_t\right) && \text{(Markov property)} \\
=&\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \mathbb{V}_{S_{t+1}}\left(v_{\pi,t+1}(S_{t+1}) \mid S_t, A_t\right). && \text{(Theorem 1)}
\end{aligned}$$

With $\nu_{\pi,t}$ defined in (11), plugging (30) back to (29) yields

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t)\right) \mid S_t\right] \\
&+ \mathbb{V}_{A_t}\left(\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right) \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t)\right) \mid S_t\right] \\
&+ \mathbb{E}_{A_t}\left[\rho_t^2 q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - \left(\mathbb{E}_{A_t}\left[\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right]\right)^2 \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 \left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t)\right) \mid S_t\right] \\
&+ \mathbb{E}_{A_t}\left[\rho_t^2 q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t). && \text{(Lemma 1)}
\end{aligned}$$

When $t = T-1$, we have

$$\begin{aligned}
\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right) =&\mathbb{V}\left(\rho_t r(S_t, A_t) \mid S_t\right) \\
=&\mathbb{V}\left(\rho_t q_{\pi,t}(S_t, A_t) \mid S_t\right) \\
=&\mathbb{E}_{A_t}\left[\rho_t^2 q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t),
\end{aligned}$$

which completes the proof. □

We restate and present the main proof of Theorem 2.

Theorem 2 (Optimal Behavior Policy)**.** *For any t and s, the behavior policy $\mu_t^*(a|s)$ defined above is an optimal solution to the following problem*

$$\min_{\mu_t\in\Lambda_t,\dots,\mu_{T-1}\in\Lambda_{T-1}} \quad \mathbb{V}\left(G^{\textit{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t = s\right),$$

where $\Lambda_t \doteq \{\mu_t \in \Delta(\mathcal{A}) \mid \forall s, a, \mu_t(a|s) = 0 \implies$
$\pi_t(a|s)u_{\pi,t}(s,a) = 0\}$.

Proof. We proceed via induction. When $t = T-1$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{T-1:T-1}}_{T-1:T-1}) \mid S_{T-1} = s\right)\\
=&\mathbb{V}_{A_{T-1}}\left(\rho_{T-1} r(s, A_{T-1}) \mid S_{T-1} = s\right)\\
=&\mathbb{V}_{A_{T-1}}\left(\rho_{T-1} q_{\pi,T-1}(s, A_{T-1}) \mid S_{T-1} = s\right).
\end{aligned}$$

The definition of μ^*_{T-1} in (12) and Lemma 2 ensure that μ^*_{T-1} is an optimal solution to

$$\min_{\mu_{T-1}\in\Lambda_{T-1}} \quad \mathbb{V}\left(G^{\text{PDIS}}\left(\tau^{\mu_{T-1}}_{T-1}\right) \mid S_{T-1} = s\right).$$

Now, suppose for some $t \in [T-2]$, $\mu^*_{t+1:T-1}$ is an optimal solution to

$$\min_{\mu_{t+1}\in\Lambda_{t+1},\dots,\mu_{T-1}\in\Lambda_{T-1}} \quad \mathbb{V}\left(G^{\text{PDIS}}\left(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}\right) \mid S_{t+1} = s\right).$$

To complete induction, we proceed to proving that $\mu^*_{t:T-1}$ is an optimal solution to

$$\min_{\mu_t\in\Lambda_t,\dots,\mu_{T-1}\in\Lambda_{T-1}} \quad \mathbb{V}\left(G^{\text{PDIS}}\left(\tau^{\mu_{t:T-1}}_{t:T-1}\right) \mid S_t = s\right). \tag{31}$$

In the rest of this proof, we omit the domain $\Lambda_t, \dots, \Lambda_{T-1}$ for simplifying notations. For any $\mu_{t:T-1}$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t:T-1}}_{t:T-1}) \mid S_t\right)\\
=&\mathbb{E}_{A_t}\left[\rho_t^2\left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t) + q^2_{\pi,t}(S_t, A_t)\right) \mid S_t\right]\\
&- v^2_{\pi,t}(S_t) && \text{(By Lemma 4)}\\
\overset{(a)}{\geq}&\mathbb{E}_{A_t}\left[\rho_t^2\left(\mathbb{E}_{S_{t+1}}\left[\min_{\mu'_{t+1:T-1}} \mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu'_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t) + q^2_{\pi,t}(S_t, A_t)\right) \mid S_t\right]\\
&- v^2_{\pi,t}(S_t) && \text{(Monotonically non-increasing in } \mathbb{V}(\cdot))\\
=&\mathbb{E}_{A_t}\left[\rho_t^2\left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\mu^*_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_t(S_t, A_t) + q^2_{\pi,t}(S_t, A_t)\right) \mid S_t\right]\\
&- v^2_{\pi,t}(S_t) && \text{(Inductive hypothesis)}\\
=&\mathbb{E}_{A_t}\left[\rho_t^2 u_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) && \text{(By (13))}\\
=&\mathbb{V}_{A_t}\left(\rho_t\sqrt{u_{\pi,t}(S_t, A_t)} \mid S_t\right) + \mathbb{E}_{A_t}\left[\rho_t\sqrt{u_{\pi,t}(S_t, A_t)} \mid S_t\right]^2 - v^2_{\pi,t}(S_t) && \text{(Definition of variance)}\\
=&\mathbb{V}_{A_t}\left(\rho_t\sqrt{u_{\pi,t}(S_t, A_t)} \mid S_t\right) + \mathbb{E}_{A_t\sim\pi_t(\cdot|S_t)}\left[\sqrt{u_{\pi,t}(S_t, A_t)} \mid S_t\right]^2 - v^2_{\pi,t}(S_t) && \text{(Lemma 1 and } \mu_t \in \Lambda_t)\\
\overset{(b)}{\geq}&\mathbb{E}_{A_t\sim\pi_t(\cdot|S_t)}\left[\sqrt{u_{\pi,t}(S_t, A_t)} \mid S_t\right]^2 - v^2_{\pi,t}(S_t). && \text{(Non-negativity of variance)}
\end{aligned}$$

According to the inductive hypothesis, the equality in (a) can be achieved when $\mu_{t+1:T-1} = \mu^*_{t+1:T-1}$. According to the construction of μ^*_t in (12) and Lemma 3, the equality in (b) can be achieved when $\mu_t = \mu^*_t$. This suggests that $\mu^*_{t:T-1}$ achieves the lower bound and is thus an optimal solution to (31), which completes the induction and thus completes the proof. □

A.6. Proof of Theorem 3

To prove the variance reduction property of $\hat\mu$, we express $\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t = s\right)$, the variance of the on-policy Monte Carlo estimator, in the form of a Bellman equation (Tamar et al., 2016; O'Donoghue et al., 2018; Sherstan et al., 2018). Define

$$\tilde r_{\pi,t}(s,a) \doteq \nu_{\pi,t}(s,a) + q^2_{\pi,t}(s,a) - v^2_{\pi,t}(s) \quad \forall t \in [T-1], \tag{32}$$

$$\tilde q_{\pi,t}(s,a) \doteq \begin{cases} \tilde r_{\pi,t}(s,a) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\tilde q_{\pi,t+1}(s',a') & \text{if } t \in [T-2] \\ \tilde r_{\pi,t}(s,a) & \text{if } t = T-1 \end{cases}. \tag{33}$$

We have

Lemma 5 (Variance Equality)**.**

$$\mathbb{V}\left(G^{PDIS}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t = s\right) = \sum_a \pi_t(a|s)\tilde q_{\pi,t}(s,a) \quad \forall t, s.$$

Proof. We proceed via induction. When $t = T-1$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{V}_{A_t}\left(\rho_t r(S_t, A_t) \mid S_t\right) \\
=&\mathbb{V}_{A_t}\left(r(S_t, A_t) \mid S_t\right) && \text{(By on-policy)} \\
=&\mathbb{V}_{A_t}\left(q_{\pi,t}(S_t, A_t) \mid S_t\right) \\
=&\mathbb{E}_{A_t}\left[q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) \\
=&\sum_a \pi_t(a|S_t)\tilde q_{\pi,t}(S_t, a). && \text{(By (33) and } \nu_{\pi,T-1}(s,a) = 0\text{)}
\end{aligned}$$

For $t \in [T-2]$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{E}_{A_t}\left[\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + q^2_{\pi,t}(S_t, A_t) + \nu_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) \\
& && \text{(Lemma 4 and on-policy)} \\
=&\sum_a \pi_t(a|S_t)\left(\sum_{s'} p(s'|S_t, a)\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1} = s'\right) + \tilde r(S_t, a)\right) \\
=&\sum_a \pi_t(a|S_t)\left(\sum_{s'} p(s'|S_t, a)\sum_{a'} \pi_{t+1}(a'|s')\tilde q_{\pi,t+1}(s', a') + \tilde r(S_t, a)\right) && \text{(Inductive hypothesis)} \\
=&\sum_a \pi_t(a|S_t)\tilde q_{\pi,t}(S_t, a), && \text{(By (33))}
\end{aligned}$$

which completes the proof. ☐

Here, this $\tilde q$ is exactly the state-action value function of the target policy π in the MDP w.r.t. to a new reward function $\tilde r$. Manipulating (15) then yields

$$\begin{aligned}
\hat q_{\pi,t}(s,a) =&\sum_{s'} p(s'|s,a)\sum_{a'} \pi_{t+1}(a'|s')\tilde q_{\pi,t+1}(s',a') + \nu_t(s,a) + q^2_{\pi,t}(s,a) \\
=&\tilde q_{\pi,t}(s,a) + v^2_{\pi,t}(s).
\end{aligned} \tag{34}$$

Now, we restate and present the main proof of Theorem 3.

Theorem 3 (Variance Reduction)**.** *For any t and s,*

$$\mathbb{V}\left(G^{PDIS}(\tau^{\hat\mu_{t:T-1}}_{t:T-1}) \mid S_t = s\right)$$

$$\leq \mathbb{V}\left(G^{PDIS}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t = s\right) - \epsilon_t(s).$$

To define $\epsilon_t(s)$, first define $c_t(s) =$

$$\sum\nolimits_a \pi_t(a|s)\hat{q}_{\pi,t}(s,a) - \left(\sum\nolimits_a \pi_t(a|s)\sqrt{\hat{q}_{\pi,t}(s,a)}\right)^2.$$

Then we define $\epsilon_t(s) \doteq c_t(s)$ for $t = T-1$ and otherwise

$$\epsilon_t(s) \doteq c_t(s) + \mathbb{E}_{A_t \sim \hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1})|s, A_t\right]\right]. \tag{19}$$

Proof. We proceed via induction. For $t = T-1$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\hat{\mu}_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 q^2_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) && \text{(Lemma 4)} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 \hat{q}_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) && \text{(Definition of } \hat{q} \text{ (14))} \\
=&\mathbb{V}_{A_t\sim\hat{\mu}_t}\left(\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right) + \mathbb{E}^2_{A_t\sim\hat{\mu}_t}\left[\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right] - v^2_{\pi,t}(S_t) \\
& && \text{(Definition of variance and non-negativity of } \hat{q}\text{)} \\
=&\mathbb{V}_{A_t\sim\hat{\mu}_t}\left(\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t, a)}\right)^2 - v^2_{\pi,t}(S_t) && \text{(Lemma 1)} \\
=&\left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t, a)}\right)^2 - v^2_{\pi,t}(S_t) && \text{(Definition of } \hat{\mu} \text{ (18) and Lemma 3)} \\
=&\sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t, a) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t, a)}\right)^2 - \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t, a) - v^2_{\pi,t}(S_t) \\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t\right) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t, a)}\right)^2 - \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t, a) && \text{(By (34) and Lemma 5)} \\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\pi_{t:T-1}}_{t:T-1}) \mid S_t\right) - \epsilon_t(S_t). && \text{(Definition of } \epsilon \text{ (19))}
\end{aligned}$$

For $t \in [T-2]$, we have

$$\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\hat{\mu}_{t:T-1}}_{t:T-1}) \mid S_t\right) \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau^{\hat{\mu}_{t+1:T-1}}_{t+1:T-1}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_{\pi,t}(S_t, A_t) + q^2_{\pi,t}(S_t, A_t)\right) \mid S_t\right] \\
&- v^2_{\pi,t}(S_t) && \text{(Lemma 4)} \\
\leq&\mathbb{E}_{A_t\sim\hat{\mu}_t}\Big[\rho_t^2\Big(\mathbb{E}_{S_{t+1}}\Big[\sum_{a'}\pi_{t+1}(a'|S_{t+1})\tilde{q}_{\pi,t+1}(S_{t+1}, a') \mid S_t, A_t\Big] + \nu_{\pi,t}(S_t, A_t) \\
&+ q^2_{\pi,t}(S_t, A_t)\Big) \mid S_t\Big] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Inductive hypothesis and Lemma 5)} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\left(\tilde{q}_{\pi,t}(S_t, A_t) + v^2_{\pi,t}(S_t)\right) \mid S_t\right] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] \\
& && \text{(Definition of } \tilde{q} \text{ (33))} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\hat{q}_{\pi,t}(S_t, A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Definition of } \hat{q} \text{ (34))} \\
=&\mathbb{V}_{A_t\sim\hat{\mu}_t}\left(\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right) + \mathbb{E}^2_{A_t\sim\hat{\mu}_t}\left[\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right] - v^2_{\pi,t}(S_t) \\
&- \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2\mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Definition of variance and non-negativity of } \hat{q}\text{)} \\
=&\mathbb{V}_{A_t\sim\hat{\mu}_t}\left(\rho_t\sqrt{\hat{q}_{\pi,t}(S_t, A_t)}|S_t\right) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t, a)}\right)^2 - v^2_{\pi,t}(S_t)
\end{aligned}$$

$$
\begin{aligned}
&- \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Lemma 1)}\\
=&\left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right)^2 - v_{\pi,t}^2(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] \\
& && \text{(Definition of } \hat{\mu} \text{ (18) and Lemma 3)}\\
=&\sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) - v_{\pi,t}^2(S_t) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right)^2 - \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a)\\
&- \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right]\\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) + \left(\sum_a \pi_t(a|S_t)\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right)^2 - \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a)\\
&- \mathbb{E}_{A_t\sim\hat{\mu}_t}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(By (34) and Lemma 5)}\\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) - \epsilon_t(S_t). && \text{(Definition of } \epsilon \text{ (19))}
\end{aligned}
$$

□

A.7. Proof of Theorem 4

Proof. For $t = T-1$, we have

$$
\begin{aligned}
\hat{q}_{\pi,t}(s,a) &= q_{\pi,t}^2(s,a) && \text{(Definition of } \hat{q}_{\pi,t} \text{ (14))}\\
&= \hat{r}_{\pi,t}(s,a). && \text{(By } q_{\pi,T-1}(s,a) = r(s,a) \text{ and Theorem 4)}
\end{aligned}
$$

For $t \in [T-2]$, we have

$$
\begin{aligned}
&\hat{q}_{\pi,t}(s,a)\\
=&\tilde{q}_{\pi,t}(s,a) + v_{\pi,t}^2(s) && \text{(By (34))}\\
=&\tilde{r}_{\pi,t}(s,a) + v_{\pi,t}^2(s) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\tilde{q}_{\pi,t+1}(s',a') && \text{(Definition of } \tilde{q} \text{ (33))}\\
=&\tilde{r}_{\pi,t}(s,a) + v_{\pi,t}^2(s) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')(\tilde{q}_{\pi,t+1}(s',a') + v_{\pi,t+1}^2(s') - v_{\pi,t+1}^2(s'))\\
=&\tilde{r}_{\pi,t}(s,a) + v_{\pi,t}^2(s) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')(\hat{q}_{\pi,t+1}(s',a') - v_{\pi,t+1}^2(s')) && \text{(By (34))}\\
=&\nu_{\pi,t}(s,a) + q_{\pi,t}^2(s,a) - \sum_{s'} p(s'|s,a)v_{\pi,t+1}^2(s') + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a') && \text{(Definition of } \tilde{r} \text{ (32))}\\
=&-\left(\mathbb{E}[v_{\pi,t+1}(S_{t+1}) \mid S_t = s, A_t = a]\right)^2 + q_{\pi,t}^2(s,a) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a') && \text{(Definition of } \nu \text{ (11))}\\
=&-\left(q_{\pi,t}(s,a) - r(s,a)\right)^2 + q_{\pi,t}^2(s,a) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a')\\
=&2r(s,a)q_{\pi,t}(s,a) - r^2(s,a) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a')\\
=&\hat{r}_{\pi,t}(s,a) + \sum_{s',a'} p(s'|s,a)\pi_{t+1}(a'|s')\hat{q}_{\pi,t+1}(s',a'), && \text{(By Theorem 4)}
\end{aligned}
$$

which completes the proof. □

A.8. Proof of Theorem 5

Proof. We first derive an important equality. $\forall t$,

$$
\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[{\rho_t^+}^2 \hat{q}_{\pi,t}(S_t,A_t) \mid S_t\right]
$$

$$
\begin{aligned}
&= \sum_a \frac{\pi_t^2(a|S_t)}{\hat{\mu}_t^+(a|S_t)} \hat{q}_{\pi,t}(S_t,a) \\
&= \sum_a \frac{\pi_t^2(a|S_t)}{\frac{\pi_t(a|S_t)\sqrt{\hat{q}^+_{\pi,t}(S_t,a)}}{\sum_b \pi_t(b|S_t)\sqrt{\hat{q}^+_{\pi,t}(S_t,b)}}} \hat{q}_{\pi,t}(S_t,a) && \text{(by (23))} \\
&= \left[\sum_a \pi_t(a|S_t)\sqrt{\hat{q}^+_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{\hat{q}_{\pi,t}(S_t,a)}{\sqrt{\hat{q}^+_{\pi,t}(S_t,a)}}\right] \\
&= \left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}} \sqrt{\hat{q}_{\pi,t}(S_t,a)}\right]. && \text{(By (22))} \qquad (35)
\end{aligned}
$$

We proceed via induction. For $t = T-1$, we have

$$
\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\hat{\mu}^+_{t:T-1}}) \mid S_t\right) \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[{\rho_t^+}^2 q^2_{\pi,t}(S_t,A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) && \text{(Lemma 4)} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[{\rho_t^+}^2 \hat{q}_{\pi,t}(S_t,A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) && \text{(Definition of } \hat{q} \text{ (14))} \\
=&\left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}} \sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] - v^2_{\pi,t}(S_t) && \text{(By (35))} \\
=&\sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) + \left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}} \sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \\
&- \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) - v^2_{\pi,t}(S_t) \\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) \\
&-\left(\sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) - \left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}} \sqrt{\hat{q}_{\pi,t}(S_t,a)}\right]\right) \\
& && \text{(By (34) and Lemma 5)} \\
=&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) - \epsilon_t^+(S_t). && \text{(Definition of } \epsilon^+ \text{ (24))}
\end{aligned}
$$

For $t \in [T-2]$, we have

$$
\begin{aligned}
&\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\hat{\mu}^+_{t:T-1}}) \mid S_t\right) \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2\left(\mathbb{E}_{S_{t+1}}\left[\mathbb{V}\left(G^{\text{PDIS}}(\tau_{t+1:T-1}^{\hat{\mu}^+_{t+1:T-1}}) \mid S_{t+1}\right) \mid S_t, A_t\right] + \nu_{\pi,t}(S_t,A_t) + q^2_{\pi,t}(S_t,A_t)\right) \mid S_t\right] \\
&- v^2_{\pi,t}(S_t) && \text{(Lemma 4)} \\
\leq&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\Bigg[\rho_t^2\Bigg(\mathbb{E}_{S_{t+1}}\Bigg[\sum_{a'} \pi_{t+1}(a'|S_{t+1})\tilde{q}_{\pi,t+1}(S_{t+1},a') \mid S_t, A_t\Bigg] + \nu_{\pi,t}(S_t,A_t) \\
&+ q^2_{\pi,t}(S_t,A_t)\Bigg) \mid S_t\Bigg] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon^+_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Inductive hypothesis and Lemma 5)} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2\left(\tilde{q}_{\pi,t}(S_t,A_t) + v^2_{\pi,t}(S_t)\right) \mid S_t\right] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon^+_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] \\
& && \text{(Definition of } \tilde{q} \text{ (33))} \\
=&\mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2 \hat{q}_{\pi,t}(S_t,A_t) \mid S_t\right] - v^2_{\pi,t}(S_t) - \mathbb{E}_{A_t\sim\hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon^+_{t+1}(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(Definition of } \hat{q} \text{ (15))} \\
=&\left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \left[\sum_a \pi_t(a|S_t) \frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}} \sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] - v^2_{\pi,t}(S_t)
\end{aligned}
$$

$$
\begin{aligned}
&- \mathbb{E}_{A_t \sim \hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}^+(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(By (35))} \\
=& \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) - v_{\pi,t}^2(S_t) \\
&+ \left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right]\left[\sum_a \pi_t(a|S_t)\frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] - \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) \\
&- \mathbb{E}_{A_t \sim \hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}^+(S_{t+1}) \mid S_t, A_t\right]\right] \\
=& \mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) + \left[\sum_a \pi_t(a|S_t)\sqrt{\eta_{\pi,t}(S_t,a)}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right]\left[\sum_a \pi_t(a|S_t)\frac{1}{\sqrt{\eta_{\pi,t}(S_t,a)}}\sqrt{\hat{q}_{\pi,t}(S_t,a)}\right] \\
&- \sum_a \pi_t(a|S_t)\hat{q}_{\pi,t}(S_t,a) - \mathbb{E}_{A_t \sim \hat{\mu}_t^+}\left[\rho_t^2 \mathbb{E}_{S_{t+1}}\left[\epsilon_{t+1}^+(S_{t+1}) \mid S_t, A_t\right]\right] && \text{(By (34) and Lemma 5)} \\
=& \mathbb{V}\left(G^{\text{PDIS}}(\tau_{t:T-1}^{\pi_{t:T-1}}) \mid S_t\right) - \epsilon_t^+(S_t). && \text{(Definition of } \epsilon^+ \text{ (24))}
\end{aligned}
$$

□

B. Experiment Details

B.1. GridWorld

For a Gridworld with size n, its width, height, and time horizon T are all set to n. There are four possible actions: up, down, left, and right. After taking an action, the agent has a 0.9 probability of moving accordingly and a 0.1 probability of moving uniformly at random. If the agent runs into a boundary, the agent stays in its current location. The reward function $r(s, a)$ is randomly generated and fixed after generation. We normalize the rewards across all (s, a) such that $\max_{s,a} r(s, a) = 1$. We consider a set of randomly generated target policies. The ground truth policy performance is estimated using the on-policy Monte Carlo method by running each target policy for 10^6 episodes. We test two different sizes of the Gridworld with a number of $1,000$ and $27,000$ states. The offline dataset contains $m = 10^5$ randomly generated tuples. For a Gridworld of size n, the total amount of possible (s, t, a, r, s') tuples is $n \times n \times n \times 4 \times 4 = 16n^3$. The offline data coverages for the Gridworld of size $1,000$ and $27,000$ are then 62.5% and 2.3%.

We use a one-hot vector representing the position of the agent and a real number representing the current time step as features for the state. We execute Algorithm 1 to approximate function r, q, and $\hat{q}$. As shown in Algorithm 1, we train r using supervised learning by batch stochastic gradient descent. We train q and $\hat{q}$ using fitted Q-learning. We split the offline data into a training set and a test set. We tune all hyperparameters offline based on the supervised learning loss and fitted Q-learning loss on the test set. With the Adam optimizer (Kingma & Ba, 2015), we search the learning rates in $\left\{2^{-20}, 2^{-18}, \cdots, 2^{0}\right\}$ to minimize the loss on the offline data and use the learning rate 2^{-10} on all learning processes. For the behavior policy search (BPS, Hanna et al. (2017)) and robust on-policy sampling (ROS, Zhong et al. (2022)) algorithms, we use the reported parameters from Hanna et al. (2017) and Zhong et al. (2022), since it is not clear how to do hyperparameter turning for BPS and ROS with only offline data.

B.2. MuJoCo

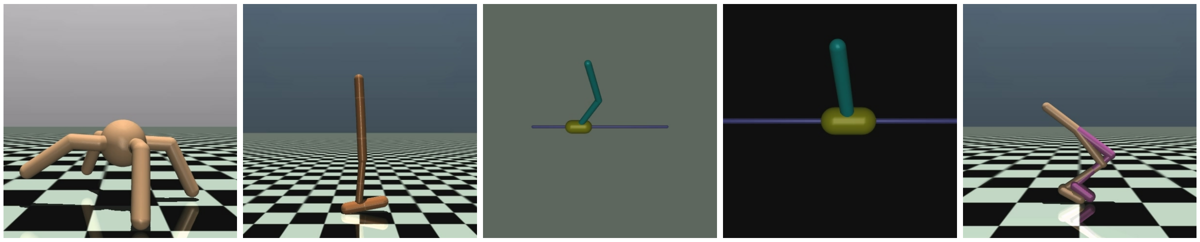

Figure 3. MuJoCo (Todorov et al., 2012) robot simulation tasks. MuJoCo is a physics engine for robotics simulation and contains various stochastic environments. The goal in each environment is to control a robot to achieve different behaviors such as walking, jumping, and balancing. Environments from the left to the right are Ant, Hopper, InvertedDoublePendulum, InvertedPendulum, and Walker. We conducted experiments on those five environments with results reported in Section 7.

Figure 3 is an introduction to the MuJoCo environments. We construct 150 policies (30 policies in each environment) with a wide range of performance using the proximal policy optimization (PPO) algorithm (Schulman et al., 2017) and the default PPO implementation in Huang et al. (2022). Since our methods are designed for discrete action space, we discretize the first dimension of MuJoCo action space in our experiments. The remaining dimensions are controlled by the PPO policy and are deemed as part of the environment. We run each compared algorithm 30 times for each policy and compute the average and standard error to plot curves in Figure 2. To generate offline data, we add different levels of noise to the target policy and run noisy target policies for 2000 episodes. The noise is in the form of a uniformly random policy, and its weight is uniformly randomly sampled from $(0, 0.1]$. This data generation process simulates the data generated during the training of a policy. Notably, compared with previous works, we do not need data to be complete trajectories or generated by known policies. We leave the investigation of entirely irrelevant offline data in the MuJoCo domain for future work. Our algorithm is robust on hyperparameters. All learning rates in Algorithm 1 are tuned offline and are the same 2^{-10} across all MuJoCo and Gridworld experiments.

In MuJcCo, the episode length varies because of stochasticity in policies and environments. Because the length of each episode is not fixed, episodes in off-policy estimation may be longer than episodes in on-policy estimation. In the main text, we use episodes instead of steps as the x-axis mainly to improve readability. Because after running 100 steps, we might already have a good estimate for a target policy with a length of 10 but may still not finish a single episode for a target policy with a length of 250. Due to the diversity of our target policies, averaging using steps as the x-axis makes the plot conceptually hard to interpret.

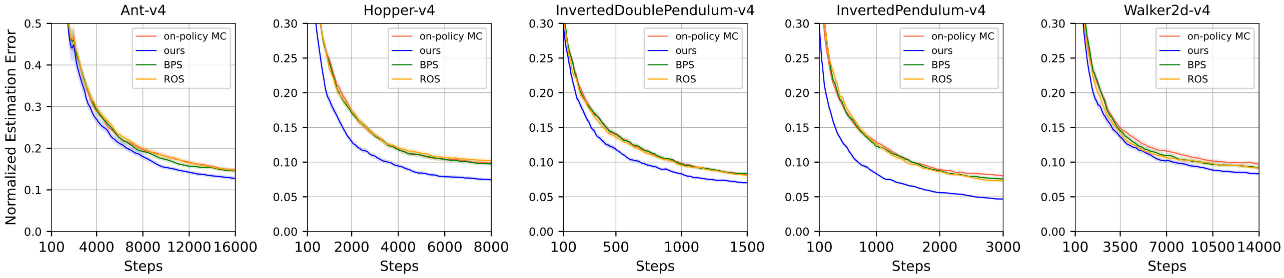

Figure 4. MuJoCo results using steps as the x-axis. We draw each curve from step 100 because some policies need more than 100 steps to finish the first episode. All curves are averaged over 900 trials (30 target policies, each having 30 independent runs). The shaded regions denote standard errors and are invisible because they are too small.

We anyway show the figure with steps as the x-axis in Figure 4. Setting steps as the x-axis, we linearly interpolate the estimation error across episodes. At each step, we average the estimation error for all tests that have completed the first episode and, thus, have an estimate. The estimation error is divided by the first estimate of the on-policy estimation to get the normalized estimation error. Although the normalized estimation error for the on-policy estimation starts from 1, it may be unstable until around 1000 steps because different policies get the first estimate at different steps. However, it is still clear that our off-policy estimator achieves the same accuracy with fewer online steps.