GLOBALLY CONVERGENT OFFLINE REINFORCEMENT LEARNING WITH SMOOTHED BELLMAN RESIDUAL MINIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Bellman Residual Minimization (BRM) enforces pointwise Bellman consistency and remains stable under neural parameterization, yet it is rarely used in offline reinforcement learning. Recent work shows that the debiased objective, though non-convex minimax, admits a Polyak–Łojasiewicz–strongly-concave landscape (Kang et al., 2025), but only for offline inverse RL, and only up to the residual, which is not known to control value error. We propose `Off-GLADIUS`, which carries this geometry to offline RL, and establish global convergence for the procedure actually run. We then close the remaining gap by proving a local metric-subregularity inequality for the soft Bellman fixed point that converts residual excess risk into a finite-sample bound on value error under the optimal policy, assuming realizability and single-policy concentrability. Empirically, `Off-GLADIUS` is competitive with CQL and ahead of OptiDICE and prior BRM methods on three discrete-action control benchmarks. Carrying no pessimism term, density ratio, or target network, it also degrades far more gracefully as the logged data turns adversarial.

1 INTRODUCTION

Modern sequential decision-making systems, ranging from large language models (Ouyang et al., 2022) to autonomous driving (Kiran et al., 2021), select actions whose consequences unfold over time. Reinforcement learning (RL) provides a principled framework for modeling and optimizing such systems. However, in many applications, online interaction with the environment is unethical, unsafe, or prohibitively expensive (Komorowski et al., 2018; Raghu et al., 2017). In these settings, *offline reinforcement learning (offline RL)* (Levine et al., 2020; Prudencio et al., 2023) is used, where learning is performed using a fixed, pre-collected dataset without further interaction with the environment.

In domains where offline RL is most compelling, the state space is typically too large for tabular methods, so effective learning hinges on function approximation (Levine et al., 2020). Function approximation is also where the difficulty concentrates: the interaction between bootstrapping, off-policy data, and approximation is what makes offline value estimation unstable in the first place.

To address the "offline + large state" setup using function approximation, prior research has largely coalesced around three prominent paradigms. These include marginalized importance-weighting (MIS) methods (Uehara et al., 2020; Jiang and Huang, 2020) that evaluate policies by correcting state-action distributions, fitted fixed-point methods (e.g., Fitted Q-Iteration (Ernst et al., 2005) and Fitted Value Iteration (Munos and Szepesvári, 2008)) that enforce consistency on the approximation functions, and projected mean-squared Bellman error (MSPBE) minimization methods (Patterson et al., 2022). A fourth line, Bellman Residual Minimization (BRM), has long been viewed as a theoretically superior alternative (Jiang and Xie, 2025). By minimizing the mean-squared Bellman error directly via gradient descent, BRM uniformly enforces strict pointwise Bellman consistency across all states and actions without relaxation. Most notably, this direct optimization approach circumvents the "deadly triad" of instability that frequently plagues traditional DP-based methods (Tsitsiklis and Van Roy, 1996b; Van Hasselt et al., 2018; Sutton and Barto, 2018).

Despite possessing these ideal characteristics for offline reinforcement learning, BRM requires solving a complex minimax optimization problem (Antos et al., 2008), and establishing its global convergence has remained a long-standing open question (Jiang and Xie, 2025). Even the closest prior work,

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

SBEED (Dai et al., 2018), which offered strong empirical effectiveness and tight sample-complexity analysis (Touati and Vincent, 2020), left the establishment of global convergence unresolved. Kang et al. (2025) recently showed that the debiased smoothed objective admits a Polyak–Łojasiewicz–strongly-concave landscape, but for offline inverse RL, and with the guarantee stated for the Bellman residual rather than the value error. This context leads us to a fundamental question:

Can BRM achieve global convergence to the optimum in offline reinforcement learning?

In this paper, we give an affirmative answer for one route to it. We propose `Off-GLADIUS`, which extends GLADIUS (Kang et al., 2025) from offline inverse RL to the offline RL setting, and establish its global convergence. Specifically, our main contributions are as follows:

- We propose `Off-GLADIUS`, adapting smoothed BRM with primal–dual debiasing (Dai et al., 2018; Kang et al., 2025) to offline RL through a Q-function-centric reformulation.
- We establish global convergence for `Off-GLADIUS` in entropy-regularized MDPs with finite action spaces, for the procedure actually run rather than an assumed minimizer.
- We convert this residual guarantee into a value-function bound. A local metric-subregularity inequality for the soft Bellman fixed point bounds $\|\widehat{Q} - Q^*\|_{2,\pi^*}^2$ by the residual risk, under realizability and single-policy concentrability. This identifies conditions under which the conversion holds. Fujimoto et al. (2022) show that it fails in general.
- Empirically, on three discrete-action control benchmarks, `Off-GLADIUS` is competitive with CQL (Kumar et al., 2020) and ahead of OptiDICE (Lee et al., 2021). Against the two prior BRM methods, SBEED (Dai et al., 2018) and BRM-FPI (Antos et al., 2008), matched in capacity and training budget, `Off-GLADIUS` is the only one of the three that is stable on every task, isolating the optimization gap that their guarantees leave open. `Off-GLADIUS` also degrades far more gracefully than CQL as the logged data turns adversarial.

Section 2 introduces the offline RL setting, and Section 3 formulates the debiased BRM objective. Section 4 presents `Off-GLADIUS`, and Section 5 contains our main theoretical results. Section 6 reports experiments. Appendix B surveys the broader related work, and Appendix E situates our guarantee within the BRM literature through an error decomposition.

2 Problem Settings

2.1 Setup

Notations. Let Δ_A denote the probability simplex over a set A, and $\Delta_A^B := \{f : B \to \Delta_A\}$ be the set of probability kernels.

Markov Decision Process (MDP). We formulate the decision-making problem as a discounted Markov Decision Process described by the tuple $(\mathcal{S}, \mathcal{A}, P, \nu_0, \gamma, r)$. Here, $\mathcal{S}$ is a measurable state space, $\mathcal{A}$ denotes a finite action space, and $P : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$ is a Markovian transition kernel. The initial state distribution is given by $\nu_0 \in \Delta_{\mathcal{S}}$, and $\gamma \in (0, 1)$ is the discount factor. For any state-action pair (s, a), the function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ gives the deterministic and bounded immediate reward. We study the entropy-regularized control problem with a regularization coefficient $\lambda > 0$. Specifically, we employ the Shannon entropy for a distribution $q \in \Delta_{\mathcal{A}}$, defined as $H(q) := -\sum_{a\in\mathcal{A}} q(a) \log q(a)$.

Given the initial distribution ν_0, we define the distribution (and expectation) of state-action sequences for policy π over the sample space $(\mathcal{S} \times \mathcal{A})_\infty := \{(s_0, a_0, s_1, a_1, ...) : s_h \in \mathcal{S}, a_h \in \mathcal{A}, h \in \mathbb{N}\}$ as $\mathbb{P}_\pi$ (and $\mathbb{E}_\pi$, respectively). For any policy π, define the time-h state–action visitation distribution

$$d_h^\pi(s, a) := \mathbb{P}_\pi(s_h = s,\ a_h = a),$$

and the discounted state–action occupancy measure

$$d^\pi(s, a) := (1 - \gamma) \sum_{h=0}^{\infty} \gamma^h\, d_h^\pi(s, a).$$

We also define the corresponding weighted L_2 norm

$$\|f\|_{2,\pi}^2 := \mathbb{E}_{(s,a)\sim d^\pi}[f(s, a)^2].$$

Policy and Value Functions. A stationary Markov policy, denoted $\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}$, maps each state s to a probability distribution over the action space $\mathcal{A}$, written as $\pi(\cdot \mid s)$. Consequently, when occupying state s_h at timestep h, the agent selects an action a_h by sampling from $\pi(\cdot \mid s_h)$. Then we can define the entropy-regularized optimal policy and its associated value functions as follows:

$$\pi^* := \text{argmax}_{\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}} \mathbb{E}_{\nu_0,\pi}\left[\sum_{h=0}^{\infty} \gamma^h \Big(r(s_h, a_h) + \lambda H(\pi(\cdot \mid s_h)) \Big)\right],$$

$$V^*(s) := \max_{\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}} \mathbb{E}_{\nu_0,\pi}\left[\sum_{h=0}^{\infty} \gamma^h \Big(r(s_h, a_h) + \lambda H(\pi(\cdot \mid s_h)) \Big) \;\middle|\; s_0 = s\right],$$

$$Q^*(s,a) := \max_{\pi \in \Delta_{\mathcal{A}}^{\mathcal{S}}} \mathbb{E}_{\nu_0,\pi}\left[r(s_0, a_0) + \sum_{h=1}^{\infty} \gamma^h \Big(r(s_h, a_h) + \lambda H(\pi(\cdot \mid s_h)) \Big) \;\middle|\; s_0 = s, a_0 = a\right].$$

It can be demonstrated that the optimal policy π^* and the value functions adhere to the following optimality equations (see Appendix B.4 in Kang et al. (2025)):

$$\pi^*(a \mid s) = \frac{\exp\left(Q^*(s,a)/\lambda\right)}{\sum_{a' \in \mathcal{A}} \exp\left(Q^*\left(s,a'\right)/\lambda\right)}, \quad \forall a \in \mathcal{A}, \tag{1}$$

$$V^*(s) = \lambda \ln\left[\sum_{a \in \mathcal{A}} \exp\left(Q^*(s,a)/\,\lambda\right)\right], \tag{2}$$

$$Q^*(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim P(s,a)}\left[\lambda \ln \sum_{a' \in \mathcal{A}} \exp\left(Q^*\left(s',a'\right)/\lambda\right) \;\middle|\; s, a\right]. \tag{3}$$

Observe that these optimality conditions are mathematically equivalent to those found in entropy-regularized reinforcement learning (Haarnoja et al., 2017; 2018)[1]. For notational convenience, we set $\lambda = 1$ in the subsequent derivations.

2.2 Offline Reinforcement Learning (Offline RL)

We study the *offline* (off-policy) setting, where the learner is *not* allowed to interact with the MDP. Instead, the learner is given a fixed dataset $\mathcal{D}_N^b := \{(s_i, a_i, r(s_i, a_i), s_i')\}_{i=1}^N$ that was collected by executing a stationary Markov *behavioral policy* (a.k.a. behavior policy) $\pi_b : \mathcal{S} \to \Delta_{\mathcal{A}}$. Concretely, under π_b, a trajectory evolves as $s_0 \sim \nu_0$, $a_h \sim \pi_b(\cdot \mid s_h)$, $s_{h+1} \sim P(\cdot \mid s_h, a_h)$, and each datapoint in $\mathcal{D}_N^b$ is a transition tuple $(s_h, a_h, r(s_h, a_h), s_{h+1})$. We do not assume π_b is known explicitly; the algorithm only uses samples in $\mathcal{D}_N^b$.

Given $\mathcal{D}_N^b$, the goal of offline RL is often defined as finding a policy $\hat{\pi}$ whose value is close to the optimal policy π^*. In our setting, as seen in Equation (1), the optimal policy π^* is characterized by Q^*. Therefore, as in Hu et al. (2021), instead of finding $\hat{\pi}$, we find $\hat{Q}$ that is close to Q^* in order to recover a near-optimal policy. Specifically, a good offline RL algorithm finds $\hat{Q}$ that minimizes

$$\mathbb{E}_{\mathcal{D}_N^b}\left[\left\|Q^* - \hat{Q}\right\|_{2,\pi^*}^2\right].$$

3 Bellman Residual Minimization (BRM)

In this section, we describe Bellman Residual Minimization (BRM), but in the format that motivates Algorithm 1 we will describe in Section 4. Throughout, given a function Q, we define

$$V^Q(s) := \ln\left[\sum_{a \in \mathcal{A}} \exp\left(Q(s,a)\right)\right]$$

[1]This equivalence is frequently noted in the literature on Inverse Reinforcement Learning (Ermon et al., 2015; Zeng et al., 2024) and Dynamic Discrete Choice (Geng et al., 2020; Kang et al., 2025). Refer to Kang et al. (2025) for a detailed discussion. In the experiments, we use the entropy regularization parameter λ for hyperparameter tuning.

and

$$\pi^Q(a|s) := \frac{\exp\left(Q(s,a)\right)}{\sum_{a\in\mathcal{A}} \exp\left(Q(s,a)\right)}.$$

For the space of Q function $\mathcal{Q}_\infty := \{Q : \mathcal{S}\times\mathcal{A} \to \mathbb{R} \mid \|Q\|_\infty < \infty\}$, we define the *Bellman operator* and the *sampled Bellman operator* for Q function as follows:

$$\mathcal{T}Q(s,a) := r(s,a) + \gamma \cdot \mathbb{E}_{s'\sim P(s,a)}\left[V^Q(s')\right], \quad \hat{\mathcal{T}}Q(s,a,s') := r(s,a) + \gamma \cdot V^Q(s').$$

According to the Bellman equation for Q function in Equation (3), one can prove that $\mathcal{T}Q^*(s,a) = Q^*(s,a)$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$, which means that Q^* is a fixed point for the operator $\mathcal{T}$; in fact, Q^* is the unique solution for $\mathcal{T}Q = Q$. Motivated by this, we define

$$\mathcal{L}_{\text{BE}}(Q)(s,a) := \left(\mathcal{T}Q(s,a) - Q(s,a)\right)^2$$

and call it a *pointwise squared Bellman error loss.* For Q^*, this quantity must be 0 for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$. We define *mean squared Bellman error (MSBE) loss* for π over (s,a) as

$$\overline{\mathcal{L}_{\text{BE}}}(Q;\pi) = \mathbb{E}_\pi\left[\mathcal{L}_{\text{BE}}(Q)(s,a)\right] \tag{4}$$

Bellman Residual Minimization (BRM) finds Q that minimizes the MSBE. Unfortunately, we cannot compute $\mathcal{T}$ directly since the learner does not know the transition kernel P. In practice, one should use *sampled Bellman operator* $\hat{\mathcal{T}}$ instead, and get the quantity we often call the *temporal difference*

$$\mathcal{L}_{\text{TD}}(Q)(s,a,s') := (\hat{\mathcal{T}}Q(s,a,s') - Q(s,a))^2.$$

Naively applying the expectation over this temporal difference is known to cause bias, often called the *double sampling* issue (Antos et al., 2008; Jiang and Xie, 2025). Fortunately, there is a debiasing technique called *bi-conjugate trick*, which is formally written in the Lemma 3.1.

Lemma 3.1 (Bi-conjugate trick)**.** *(Antos et al., 2008; Kang et al., 2025)*

$$\mathcal{L}_{BE}(Q)(s,a) = \mathbb{E}_{s'\sim P(s,a)}\left[\mathcal{L}_{TD}(Q)(s,a,s')\right] - \gamma^2 \min_{\xi\in\mathbb{R}^{S\times A}} D(Q)(s,a,\xi)$$

where $D(Q)(s,a,\xi) := \mathbb{E}_{s'\sim P(s,a)}\left[(V^Q(s') - \xi)^2\right]$.

Now we can rewrite the BRM objective, i.e., the MSBE minimization, as finding Q that minimizes

$$\overline{\mathcal{L}_{\text{BE}}}(Q,\pi) = \min_{Q\in\mathcal{Q}_\infty} \max_{\xi\in\mathbb{R}^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_\pi\left[\mathbb{E}_{s'\sim P(s,a)}\left[\mathcal{L}_{\text{TD}}(Q)(s,a,s') - \gamma^2(V^Q(s') - \xi(s,a))^2\right]\right] \tag{5}$$

and now our setting reduces to the saddle-point problem between parameter Q and ξ. Empirically, this reduces to finding Q that minimizes

$$\frac{1}{N}\sum_{(s,a,s')\in\mathcal{D}_N^b}\left[\mathcal{L}_{\text{TD}}(Q)(s,a,s') - \gamma^2\left(V^Q(s') - \bar{\xi}(s,a)\right)^2\right] \tag{6}$$

where $\bar{\xi} := \text{argmin}_{\xi\in\mathbb{R}^{S\times A}} \frac{1}{N}\sum_{(s,a,s')\in\mathcal{D}_N^b}\left[\left(V^Q(s') - \xi(s,a)\right)^2\right]$.

Note that the formulation in Equation (5) (and its empirical counterpart in Equation (6)) circumvents the double sampling issue. The role of this dual correction is both theoretical and empirical: deleting the $\gamma^2(V^Q(s') - \bar{\xi}(s,a))^2$ term has a negligible impact under deterministic transitions, where the bias it removes vanishes, but prevents a severe performance degradation under moderate transition noise (Appendix D.1).

However, it faces a new challenge: it is now a minimax optimization problem, for which finding the global optimum is nontrivial. In Section 4, we will discuss the algorithm we propose to solve this minimax optimization problem.

4 ALGORITHM

In this section, we propose an algorithm for Bellman Residual Minimization (BRM), which we call `Off-GLADIUS` (Algorithm 1). Intuitively, `Off-GLADIUS` solves the BRM by adapting alternating gradient ascent and descent, a traditional approach for the saddle-point problem (Yang et al., 2020).

Let $Q_{\theta_1} \in \mathcal{Q}$ and $\xi_{\theta_2} \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ be parameterized representations of Q and ξ, respectively, using either linear classes or sufficiently wide neural networks as specified in Section 5 and Appendix A.2. The algorithm starts with a random initialization of parameters θ_1 and θ_2. For each iteration, the algorithm draws two batches, Z_Q and Z_ξ, from the offline dataset $\mathcal{D}$. The two batches are used for the descent step on θ_1 and the ascent step on θ_2, respectively.

Algorithm 1 `Off-GLADIUS`

Require: Offline dataset $\mathcal{D} = \{(s, a, r, s')\}$, horizon T, ascent steps K_ξ
Ensure: $\widehat{Q}, \widehat{\xi}$

1: Initialize $Q_{\theta_1}, \xi_{\theta_2}$; $t \leftarrow 1$
2: **while** $t \leq T$ **do**
3: **Ascent step (update ξ_{θ_2}):**
4: **for** $k = 1, \ldots, K_\xi$ **do**
5: Sample mini-batch $Z_\xi \subset \mathcal{D}$
6: $D_{\theta_2} \leftarrow \frac{1}{|Z_\xi|} \sum_{(s,a,s')\in Z_\xi} \big(V_{\theta_1}(s') - \xi_{\theta_2}(s,a)\big)^2$
7: $\theta_2 \leftarrow \theta_2 - \tau_{2,t} \nabla_{\theta_2} D_{\theta_2}$
8: **end for**
9: **Descent step (update Q_{θ_1}):**
10: Sample mini-batch $Z_Q \subset \mathcal{D}$
11: $\overline{\mathcal{L}_{\mathrm{BE}}} \leftarrow \frac{1}{|Z_Q|} \sum_{(s,a,s')\in Z_Q} \Big[\mathcal{L}_{\mathrm{TD}}(Q_{\theta_1})(s,a,s') - \gamma^2 \big(V_{\theta_1}(s') - \xi_{\theta_2}(s,a)\big)^2\Big]$
12: $\theta_1 \leftarrow \theta_1 - \tau_{1,t} \nabla_{\theta_1} \overline{\mathcal{L}_{\mathrm{BE}}}$
13: $t \leftarrow t + 1$
14: **end while**
15: **Return** $\widehat{Q} \leftarrow Q_{\theta_1}, \widehat{\xi} \leftarrow \xi_{\theta_2}$

Lines 3–8 correspond to the ascent step for θ_2. Here Q_{θ_1} is fixed, so $V^{Q_{\theta_1}}$ is a fixed function as well. Among the terms of Equation (5), only $\gamma^2(V^Q(s') - \xi(s,a))^2$ involves ξ. The learner therefore needs only to consider $D_{\theta_2} := \frac{1}{|Z_\xi|} \sum_{(s,a,s')\in Z_\xi} (V^{Q_{\theta_1}}(s') - \xi_{\theta_2}(s,a))^2$. Note that the algorithm minimizes D_{θ_2} by gradient descent: this term enters Equation (5) with a negative sign, so descending D_{θ_2} is the ascent direction of the objective. We repeat this update K_ξ times, drawing a fresh mini-batch each time, so that the inner maximization is sufficiently solved before Q_{θ_1} is updated.

Lines 9–12 are the descent step for θ_1, where we fix ξ_{θ_2} and update θ_1. Here we use all terms inside the expectation of Equation (5) to compute the gradient. The choice of K_ξ, the step sizes $\tau_{1,t}, \tau_{2,t}$, and the remaining implementation details are given in Appendix C.3.

5 THEORETICAL ANALYSIS

Consider a parameterized subclass of $\mathcal{Q}_\infty$,

$$\mathcal{Q} = \Big\{Q_\theta : \mathbb{R}^{\dim(\mathcal{S})+\dim(\mathcal{A})} \to \mathbb{R} \mid \theta \in \Theta \subseteq \mathbb{R}^{d_\theta}, Q_\theta \in \mathcal{F}\Big\}$$

where $\mathcal{F}$ denotes a class of functions such as a linear, polynomial, or deep neural network class parameterized by θ.

Assumption 5.1 (Bellman realizability)**.** The function class contains the soft-optimal action-value function: there exists $\theta^\star \in \Theta$ such that $Q_{\theta^\star} = Q^*$.

Assumption 5.1 is standard in offline RL and value-function approximation (Chen and Jiang, 2019; Xie and Jiang, 2021; Zhan et al., 2022; Zanette, 2023; Jiang and Xie, 2025). It is the condition

under which the population Bellman residual can be driven to zero by some element of the function class. In practice, this assumption is typically interpreted as an approximation-theoretic idealization: sufficiently wide neural networks can approximate broad classes of continuous functions arbitrarily well (Cybenko, 1989; Hornik, 1991; Leshno et al., 1993), and over-parameterized architectures are commonly used precisely to reduce approximation bias (Zhang et al., 2016; Allen-Zhu et al., 2019; Dar et al., 2021).

As in standard offline RL theory, we also require the logged data to cover the target state-action distribution under which we evaluate the learned Q-function.

Assumption 5.2 (Single-policy concentrability)**.** There exists a finite positive constant C_Π such that, for every measurable set $B \subseteq \mathcal{S} \times \mathcal{A}$,

$$d^{\pi^*}(B) \leq C_\Pi d^{\pi_b}(B).$$

Equivalently, when densities exist, $d^{\pi^*}(s,a) \leq C_\Pi d^{\pi_b}(s,a)$ for d^{π_b}-almost every (s,a).

For $Q \in \mathcal{Q}$, define its Bellman residual and behavior-distribution BRM risk by

$$b_Q(s,a) := \mathcal{T}Q(s,a) - Q(s,a), \qquad \mathcal{R}_b(Q) := \mathbb{E}_{(s,a)\sim d^{\pi_b}}[b_Q(s,a)^2].$$

$\mathcal{R}_b(Q)$ is the MSBE of Equation (4) weighted by the occupancy of the behavior policy. For any target state-action distribution η, define the weighted norm

$$\|f\|_{2,\eta}^2 := \mathbb{E}_{(s,a)\sim\eta}[f(s,a)^2].$$

When $\eta = d^{\pi^*}$, this is the norm $\|\cdot\|_{2,\pi^*}$ introduced in Section 2. We also use the target-support essential sup norm

$$\|f\|_{\infty,*} := \operatorname*{ess\,sup}_{s\sim d_S^{\pi^*}} \max_{a\in\mathcal{A}} |f(s,a)|.$$

where $d_S^\pi(U) := d^\pi(U \times \mathcal{A})$ denotes the state marginal of d^π.

Finally, we make a regularity condition of Q and ξ for our analysis, which holds for a broad range of function classes.

Assumption 5.3 (Regularity conditions)**.** Let Z_Q and Z_ξ be the mini-batches drawn in the descent and ascent steps of Algorithm 1, with $M_Q := |Z_Q|$ and $M_\xi := |Z_\xi|$, and let $B_Q \subseteq \Theta$ and B_ξ be compact convex parameter sets containing $\theta^\star$ and every iterate (θ_1, θ_2) of Algorithm 1. We require the following on $B_Q \times B_\xi$, verified for linear and sufficiently wide neural classes in Lemma A.10.

(i) **Smoothness and bounded derivatives.** For every $z \in Z_Q$, the map $\theta_1 \mapsto Q_{\theta_1}(z)$ is twice continuously differentiable on B_Q, and there exist constants $B_{Q,1}, B_{Q,2} < \infty$ such that

$$\sup_{\theta_1\in B_Q} \sup_{z\in Z_Q} \|\nabla_{\theta_1} Q_{\theta_1}(z)\|_2 \leq B_{Q,1}, \qquad \sup_{\theta_1\in B_Q} \sup_{z\in Z_Q} \|\nabla_{\theta_1}^2 Q_{\theta_1}(z)\|_{\mathrm{op}} \leq B_{Q,2}.$$

For every $\bar{z} \in Z_\xi$, the map $\theta_2 \mapsto \xi_{\theta_2}(\bar{z})$ is twice continuously differentiable on B_ξ, and analogous finite constants $B_{\xi,1}, B_{\xi,2} < \infty$ exist. Additionally, the Q-parameterization admits a finite global first-derivative bound B on B_Q:

$$\sup_{\theta_1\in B_Q} \sup_{(s,a)\in\mathcal{S}\times\mathcal{A}} \|\nabla_{\theta_1} Q_{\theta_1}(s,a)\|_2 \leq B.$$

(ii) **Empirical-output Q, ξ-Jacobian conditioning.** There exist $\mu_Q, \mu_\xi > 0$ such that, for every $\theta_1 \in B_Q$ and $\theta_2 \in B_\xi$,

$$J_Q(\theta_1; Z_Q) J_Q(\theta_1; Z_Q)^\top \succeq \mu_Q I_{M_Q}, \qquad J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_{M_\xi}$$

where $J_Q(\theta_1; Z_Q)$ stacks the rows $\nabla_{\theta_1} Q_{\theta_1}(z)^\top$, and J_ξ is defined analogously.

Condition (ii) constrains the parameterization, but the Bellman-residual tangent map is nondegenerate as a consequence of Assumption 5.2, not an additional NTK assumption (Corollary A.6). The main theorem below converts the population Bellman-residual excess-risk guarantee into a Q^*-estimation guarantee.

Theorem 5.4 (Bellman-subregular Q^*-estimation bound, informal)**.** *Suppose Assumptions 5.1, 5.2, and 5.3 hold. Let $Q_{\hat{\theta}_T}$ denote the Q-function returned after T iterations of Algorithm 1. Let $\tau_{1,t} = \tau_{2,t} = \frac{c_1}{c_2+t}$ for some appropriate constants $c_1 > 0, c_2 \geq 1$. Then,*

$$\mathbb{E}\left[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2\right] = O\big((c_2+T)^{-\alpha}\big) + O\left(\frac{1}{N}\right) + O\left(\frac{1}{T}\right),$$

where $\alpha = \min\{1/2, 3cc1/8\} > 0$ depends on the step-size constants and the PL and strong-concavity moduli (Lemma A.1), and the constants hidden in the $O(\cdot)$ notation are independent of N and T. The formal statement and proof are in Appendix A.7.

Sketch of Proof.

(I) Population Bellman-residual excess risk. Using the bi-conjugate reformulation (Lemma 3.1), the empirical BRM problem is a smooth minimax objective in the primal Q_θ and auxiliary ξ variables. The two-sided PL structure required by Yang et al. (2020) is established in Kang et al. (2025) under their Jacobian conditioning, whose role is played here by Assumption 5.3 (Lemma A.10). We combine this structure with the stability/generalization result of Kang and Jang (2025) to obtain the population residual-risk bound

$$\mathbb{E}_{\mathcal{D}_N^b,\mathrm{alg}}\left[\mathcal{R}_b(Q_{\hat{\theta}_T}) - \mathcal{R}_b^\star\right] \leq \Delta_{N,T} = O((c_2+T)^{-\alpha}) + O(\frac{1}{N}) + O(\frac{1}{T}),$$

where $\mathcal{R}_b^\star := \inf_{Q\in\mathcal{Q}} \mathcal{R}_b(Q)$. Under Bellman realizability, $\mathcal{R}_b^\star = 0$. Steps *(II)* and *(III)* then convert this residual guarantee into a statement about Q^*.

(II) Exact identification and entrance into a local neighborhood. If $\mathcal{R}_b(Q) = 0$, Assumption 5.2 implies $b_Q = 0$ on the d^{π^*} support. Because the soft optimal policy has full action support, the target state support is closed under all one-step transitions from actions in $\mathcal{A}$. The soft Bellman contraction then implies that $Q = Q^*$ on the d^{π^*} support. In the appendix proof, compactness of B_Q and Lipschitz continuity of $\theta \mapsto Q_\theta$ are used to define a target-support identification modulus that vanishes at the origin. Hence sufficiently small Bellman residual forces the returned Q-function into a local target-support L_∞ neighborhood of Q^*.

(III) Local Bellman subregularity and constant absorption. Inside this local neighborhood, the soft Bellman equation yields a metric-subregularity inequality of the form

$$\|Q - Q^*\|_{2,\pi^*}^2 \leq C_{\mathrm{loc}}\, \mathcal{R}_b(Q),$$

for a finite local constant C_{loc}. On the complementary event, boundedness of B_Q and Markov's inequality control the contribution by another finite constant times $\mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})]$. Combining these two event-wise bounds gives

$$\mathbb{E}\left[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2\right] \leq C_{\mathrm{Bell}}\, \mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})] \leq C_{\mathrm{Bell}}\, \Delta_{N,T},$$

for a finite constant C_{Bell} independent of N and T. Absorbing this constant into the big-O terms proves Theorem 5.4.

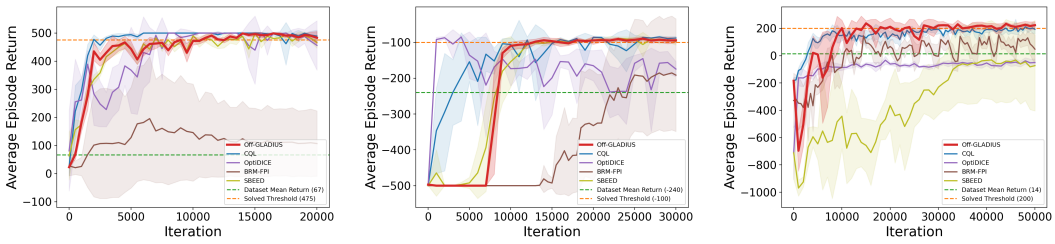
6 Experiments

Section 6.1 compares `Off-GLADIUS` with four baselines on three control tasks, Section 6.2 probes behavior when data coverage fails, and Section 6.3 isolates the optimization behavior of the BRM methods on a tabular MDP where Q^* is known.

6.1 Overall performance

Environments & Datasets. We evaluate `Off-GLADIUS` on three classic control tasks from Gymnasium (Towers et al., 2025): CartPole-v1, Acrobot-v1, and LunarLander-v3. For each we construct a mixed offline dataset of 10^6 transitions, 70% from a sub-optimal PPO policy (Schulman et al., 2017) and 30% uniform-random, following D4RL (Fu et al., 2020). Results are mean $\pm$ std over 5 seeds on the final 20% of evaluations. Details are in Appendix C.

Baselines. We compare `Off-GLADIUS` against four methods spanning the paradigms discussed in Appendix B. CQL (Kumar et al., 2020) is a pessimism-based value method and OptiDICE (Lee et al., 2021) is a marginalized importance sampling method. The remaining two are the prior BRM methods. BRM-FPI (Antos et al., 2008) is the earliest debiased BRM method but its analysis assumes that each inner minimax problem is solved exactly. SBEED (Dai et al., 2018) optimizes a smoothed, debiased objective by two-time-scale primal–dual SGDA, with a guarantee that stops at a stationary point. We match both to `Off-GLADIUS` in network capacity and training budget, so that the gaps in Table 1 isolate optimization difficulty rather than capacity. `Off-GLADIUS` itself is a direct implementation of Algorithm 1 and uses no target network, no conservatism term, and no density-ratio estimation. Full descriptions of all five methods together with their hyperparameter grids appear in Appendix C.3–C.7, and ablations over λ and K_ξ in Appendix D.2.



(a) CartPole-v1 (b) Acrobot-v1 (c) LunarLander-v3

Figure 1: Learning curves across three control tasks. Dataset Mean Return denotes the average episode reward of the dataset itself, and Solved Threshold the recommended (or maximum) achievable reward for each task.

Table 1: Final return on the mixed dataset. Mean $\pm$ std over 5 seeds, computed over the final 20% of evaluations. Bold values indicate the highest mean return.

Environment	CQL	OptiDICE	BRM-FPI	SBEED	`Off-GLADIUS`
CartPole-v1	490.8 ± 16.9	484.1 ± 25.0	110.6 ± 119.4	482.4 ± 12.8	$\mathbf{491.4 \pm 5.6}$
Acrobot-v1	-94.8 ± 11.6	-180.2 ± 34.2	-250.6 ± 195.2	-95.0 ± 6.3	$\mathbf{-93.5 \pm 5.6}$
LunarLander-v3	192.5 ± 11.5	-56.8 ± 19.3	83.2 ± 47.8	-50.4 ± 292.0	$\mathbf{215.9 \pm 12.6}$

Figure 1 and Table 1 report learning curves and final returns on the mixed dataset. `Off-GLADIUS` clears the solved threshold on all three tasks. CQL is indistinguishable from it within seed variance on CartPole-v1 and Acrobot-v1, and falls just short of the threshold on LunarLander-v3, so we read the two as competitive rather than separated. OptiDICE trails on the two harder tasks. SBEED matches `Off-GLADIUS` on CartPole-v1 and Acrobot-v1 but diverges on LunarLander-v3, and BRM-FPI is unstable on every task.

6.2 Behavior under coverage failure

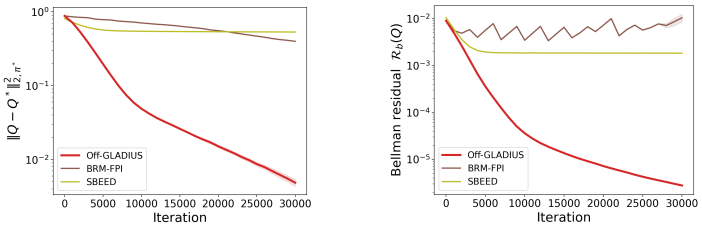
Assumption 5.2 requires the logged data to cover d^{π^*}, and the mixed dataset above satisfies it comfortably. To probe what happens outside that regime, we evaluate on three LunarLander-v3 datasets that strain it along different axes (Appendix C.1). The first contains only uniform-random data, the second contains anti-expert data collected under a policy that inverts a near-optimal expert's action preferences (softmax of negated logits), and the third mixes the two in a 5:5 ratio. We use LunarLander-v3 as it has the largest state dimension and longest horizon of our three tasks, making the gap between the behavior data and π^* widest.

As the LunarLander-v3 data degrades from uniform-random to pure anti-expert, CQL loses roughly 320 return while `Off-GLADIUS` moves by under 50. This is consistent with `Off-GLADIUS` carrying no mechanism calibrated to a particular level of distribution shift; no density ratio, no fixed conservatism coefficient, whereas a behavior-anchored penalty tracks π_b whatever its quality (Appendix F).

Table 2: Final return under coverage failure (LunarLander-v3, 5 seeds, final 20% of online evaluations). Anti-expert denotes a negative-softmax policy ($\mathrm{softmax}(-\ell(s))$, where $\ell(s)$ denotes the expert's action logits).

Dataset	CQL	OptiDICE	Off-GLADIUS
Uniform-random	-124.1 ± 38.8	-237.9 ± 65.6	$\mathbf{-91.4 \pm 68.0}$
Anti-expert : random (5:5)	-291.9 ± 101.9	-140.5 ± 46.5	$\mathbf{-130.3 \pm 9.3}$
Anti-expert only	-440.8 ± 24.0	-241.4 ± 172.3	$\mathbf{-141.1 \pm 10.9}$

6.3 Verification on a Tabular MDP



(a) Distance to Q^*.

(b) Bellman residual.

Figure 2: Tabular FrozenLake over 5 seeds, where Q^* is computed exactly. Off-GLADIUS continues to approach Q^* throughout training, while both prior BRM methods stop short. Under deterministic transitions $\mathcal{R}_b(Q^*) = 0$ exactly.

On the control tasks the optimization term cannot be isolated, because Q^* is unavailable. We therefore run the three BRM methods on a deterministic 8×8 FrozenLake MDP with a tabular parameterization, where the entropy-regularized optimum is computed exactly by soft value iteration. All three share the dataset and a budget of $180,000$ gradient updates, so approximation and statistical error are held fixed and only the formulation and solution of the minimax problem differ.

Off-GLADIUS keeps approaching Q^* throughout training. SBEED plateaus after roughly $5,000$ iterations, as expected of a method certified only to reach a stationary point. BRM-FPI decreases linearly at the pace of policy iteration, and its residual does not fall monotonically because each round minimizes the evaluation residual of a frozen policy rather than $\mathcal{R}_b$. Appendix E reports the final values and the residual-to-value-error relation of Lemma A.5.

7 Conclusion

This paper addressed a long-standing question in offline reinforcement learning: whether smoothed Bellman residual minimization can achieve global convergence under off-policy data and yield a meaningful Q^*-estimation guarantee. We proposed Off-GLADIUS and established global convergence of its population Bellman-residual excess risk. A local metric-subregularity inequality for the soft Bellman fixed point then converts this residual guarantee into an explicit Q^*-estimation bound under realizability and single-policy concentrability. Empirically, Off-GLADIUS matches CQL and outperforms OptiDICE on three control benchmarks. Among BRM methods it is the only one that remains stable on every task and, where Q^* is known exactly, the only one whose iterate continues to approach it. It also degrades far more gracefully than CQL as data coverage fails. Together, these results supply the optimization guarantee that prior BRM analyses assumed or left open, and identify a regime in which a small Bellman residual does certify a small value error.

Our guarantee is limited to finite action spaces, where the log-sum-exp operator is exact, and our experiments to discrete-action control tasks. Extending either is the natural next step. It would also be valuable to examine whether model-based BRM approaches converge to the global optimum, which remains an open and important question. In addition, as Jiang and Xie (2025) note, a reliable BRM solver such as Off-GLADIUS could serve as a building block for version-space pessimistic methods like PSPI (Cheng et al., 2022).

REPRODUCIBILITY STATEMENT

We provide an anonymous implementation of `Off-GLADIUS` and the code used for the reported experiments at `https://anonymous.4open.science/r/off-gladius-code-632A`. Algorithm 1 specifies the complete ascent–descent procedure. For the theoretical results, the assumptions and the main guarantee are stated in Section 5, while Appendix A provides the complete proofs and Appendix A.2 details the conditions under which the required regularity assumptions hold for linear and sufficiently wide neural-network classes. For the empirical results, Appendix C reports the computational environment, offline-dataset construction, network architecture and initialization, training and evaluation protocols, and random-seed aggregation. Appendices C.3–C.7 provide the implementation details and hyperparameter search spaces for `Off-GLADIUS` and all baselines, and Appendix C.8 reports the selected configuration for every method and environment. Additional dataset constructions and ablation protocols are documented in Appendices C.1 and D.2. Together with the released code, these details are intended to enable reproduction of both the theoretical claims and the empirical results.

AI USE DISCLOSURE

In this work, we used generative AI tools to assist in the writing of proofs, propose or refine hypotheses, and assist with translation. We have not used generative AI tools for the other research tasks such as generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset and support qualitative and thematic data analysis, interpret results. Additionally, we used generative AI tools to create or edit software code, draft parts of the research paper, summarize or analyze existing literature, and edit the paper to improve readability.

We have reviewed all AI-assisted work. In particular, we checked AI-assisted research ideas and literature analyses for potential plagiarism and unsupported attribution through a manual literature survey. AI-generated or AI-edited experimental code was verified and tested for correctness by at least two authors. All AI-assisted proofs and mathematical arguments were independently reviewed and verified by the authors, and all participating authors are fully familiar with the final proofs and experimental implementation.

We take responsibility for the final content of this work, including text, claims, code, or other artifacts produced with the aid of generative AI.

REFERENCES

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International conference on machine learning*, pages 242–252. PMLR, 2019.

András Antos, Csaba Szepesvári, and Rémi Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71:89–129, 2008.

Kavosh Asadi and Michael L Littman. An alternative softmax operator for reinforcement learning. In *International Conference on Machine Learning*, pages 243–252. PMLR, 2017.

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine learning proceedings 1995*, pages 30–37. Elsevier, 1995.

Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. In *International conference on machine learning*, pages 1042–1051. PMLR, 2019.

Ching-An Cheng, Tengyang Xie, Nan Jiang, and Alekh Agarwal. Adversarially trained actor critic for offline reinforcement learning. In *International Conference on Machine Learning*, pages 3852–3878. PMLR, 2022.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, Jianshu Chen, and Le Song. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International conference on machine learning*, pages 1125–1134. PMLR, 2018.

Yehuda Dar, Vidya Muthukumar, and Richard G Baraniuk. A farewell to the bias-variance tradeoff? an overview of the theory of overparameterized machine learning. *arXiv preprint arXiv:2109.02355*, 2021.

Stefano Ermon, Yexiang Xue, Russell Toth, Bistra Dilkina, Richard Bernstein, Theodoros Damoulas, Patrick Clark, Steve DeGloria, Andrew Mude, Christopher Barrett, et al. Learning large-scale dynamic discrete choice models of spatio-temporal preferences with application to migratory pastoralism in east africa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research (JMLR)*, 6:503–556, 2005.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Scott Fujimoto, David Meger, Doina Precup, Ofir Nachum, and Shixiang Shane Gu. Why should i trust you, bellman? the bellman error is a poor replacement for value error. *arXiv preprint arXiv:2201.12417*, 2022.

Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34: 4028–4039, 2021.

Sinong Geng, Houssam Nassif, Carlos Manzanares, Max Reppen, and Ronnie Sircar. Deep pqr: Solving inverse reinforcement learning using anchor actions. In *International Conference on Machine Learning*, pages 3431–3441. PMLR, 2020.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International conference on machine learning*, pages 1352–1361. PMLR, 2017.

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

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2): 251–257, 1991.

Yichun Hu, Nathan Kallus, and Masatoshi Uehara. Fast rates for the regret of offline reinforcement learning. *arXiv preprint arXiv:2102.00479*, 2021.

David Yu-Tung Hui, Aaron C Courville, and Pierre-Luc Bacon. Double gumbel q-learning. *Advances in Neural Information Processing Systems*, 36:2580–2616, 2023.

Nan Jiang and Jiawei Huang. Minimax value interval for off-policy evaluation and policy optimization. *Advances in Neural Information Processing Systems*, 33:2747–2758, 2020.

Nan Jiang and Tengyang Xie. Offline reinforcement learning in large state spaces: Algorithms and guarantees. *Statistical Science*, 40(4):570–596, 2025.

Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline rl? In *International Conference on Machine Learning*, pages 5084–5096. PMLR, 2021.

Enoch H Kang and Kyoungseok Jang. Stability and generalization for bellman residuals. *arXiv preprint arXiv:2508.18741*, 2025.

Enoch Hyunwook Kang, Hema Yoganarasimhan, and Lalit Jain. An empirical risk minimization approach for offline inverse reinforcement learning and dynamic discrete choice models. In *Proceedings of the 26th ACM Conference on Economics and Computation*, pages 341–341, 2025.

B. R. Kiran, Ibrahim Sobh, Victor Talpaert, et al. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

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

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in neural information processing systems*, 32, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 1179–1191, 2020.

Jongmin Lee, Wonseok Jeon, Byungjun Lee, Joelle Pineau, and Kee-Eung Kim. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, pages 6120–6130. PMLR, 2021.

Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Chenran Li, Chen Tang, Haruki Nishimura, Jean Mercat, Masayoshi Tomizuka, and Wei Zhan. Residual q-learning: Offline and online policy customization without value. *Advances in Neural Information Processing Systems*, 36:61857–61869, 2023.

Chaoyue Liu, Libin Zhu, and Mikhail Belkin. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *Applied and Computational Harmonic Analysis*, 59: 85–116, 2022.

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

Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9:815–857, 2008.

Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *Advances in Neural Information Processing Systems*, 2019a.

Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans. Algaedice: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019b.

Long Ouyang, Jeffrey Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 2022.

Andrew Patterson, Adam White, and Martha White. A generalized projected bellman error for off-policy value estimation in reinforcement learning. *Journal of Machine Learning Research*, 23 (145):1–61, 2022.

Ricardo F. Prudencio, Matheus R. O. A. Maximo, and Esther L. Colombini. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

Antonin Raffin. Rl baselines3 zoo. `https://github.com/DLR-RM/rl-baselines3-zoo`, 2020.

Aniruddh Raghu, Matthieu Komorowski, Shashank Singh, et al. Continuous state-space models for optimal sepsis treatment: A deep reinforcement learning approach. *Machine Learning for Healthcare Conference*, 2017.

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

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

Richard S Sutton, Hamid Reza Maei, Doina Precup, Shalabh Bhatnagar, David Silver, Csaba Szepesvári, and Eric Wiewiora. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *International Conference on Machine Learning*, pages 993–1000, 2009.

Ahmed Touati and Pascal Vincent. Sharp analysis of smoothed bellman error embedding. *arXiv preprint arXiv:2007.03749*, 2020.

Mark Towers, Ariel Kwiatkowski, John Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Kallinteris Andreas, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, J K Terry, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A standard interface for reinforcement learning environments. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4916.

John Tsitsiklis and Benjamin Van Roy. Analysis of temporal-diffference learning with function approximation. *Advances in neural information processing systems*, 9, 1996a.

John N. Tsitsiklis and Benjamin Van Roy. Feature-based methods for large scale dynamic programming. *Machine Learning*, 22(1-3):59–94, 1996b.

Masatoshi Uehara, Jiawei Huang, and Nan Jiang. Minimax weight and q-function learning for off-policy evaluation. In *International Conference on Machine Learning*, pages 9659–9668. PMLR, 2020.

Hado Van Hasselt, Yotam Doron, Florian Strub, Matteo Hessel, Nicolas Sonnerat, and Joseph Modayil. Deep reinforcement learning and the deadly triad. *arXiv preprint arXiv:1812.02648*, 2018.

Tengyang Xie and Nan Jiang. Batch value-function approximation with only realizability. In *International Conference on Machine Learning*, pages 11404–11413. PMLR, 2021.

Junchi Yang, Negar Kiyavash, and Niao He. Global convergence and variance-reduced optimization for a class of nonconvex-nonconcave minimax problems. *arXiv preprint arXiv:2002.09621*, 2020.

Andrea Zanette. When is realizability sufficient for off-policy reinforcement learning? In *International Conference on Machine Learning*, pages 40637–40668. PMLR, 2023.

Siliang Zeng, Mingyi Hong, and Alfredo Garcia. Structural estimation of markov decision processes in high-dimensional state space with finite-time guarantees. *Operations Research*, 73(2):720–737, 2024. doi: 10.1287/opre.2022.0511.

Wenhao Zhan, Baihe Huang, Audrey Huang, Nan Jiang, and Jason Lee. Offline reinforcement learning with realizability and single-policy concentrability. In *Conference on Learning Theory*, pages 2730–2775. PMLR, 2022.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Ruiyi Zhang, Bo Dai, Lihong Li, and Dale Schuurmans. Gendice: Generalized offline estimation of stationary values. *International Conference on Learning Representations*, 2020.

A TECHNICAL PROOFS

A.1 PROOF OF THEOREM 5.4

We first isolate the four ingredients used by the main theorem: a population Bellman-residual excess-risk input, exact target-support identification, a compactness-based entrance modulus, and a local Bellman metric-subregularity inequality. Throughout this subsection,

$$b_Q(s,a) := \mathcal{T}Q(s,a) - Q(s,a), \qquad \mathcal{R}_b(Q) := \mathbb{E}_{(s,a)\sim d^{\pi_b}}[b_Q(s,a)^2].$$

Let $B_Q \subseteq \Theta$ denote the compact finite-radius parameter region used in Lemma A.10; it contains the realizability parameter $\theta^\star$ and the iterates of Algorithm 1. Write

$$\mathcal{Q}_B := \{Q_\theta : \theta \in B_Q\}$$

and define the target-support identification modulus

$$\omega_\infty(\varepsilon) := \sup\{\|Q - Q^*\|_{\infty,*} : Q \in \mathcal{Q}_B, \mathcal{R}_b(Q) \le \varepsilon\}.$$

Lemma A.4 below proves that exact Bellman fixed-point identification plus compactness imply

$$\omega_\infty(\varepsilon) \to 0 \qquad \text{as } \varepsilon \downarrow 0.$$

Thus one may choose $\bar{\varepsilon} > 0$ such that

$$\delta_{\bar{\varepsilon}} := \omega_\infty(\bar{\varepsilon}) < \frac{1}{2}\log\frac{1}{\gamma}.$$

Lemma A.1 (Population BRM excess-risk input)**.** *Let $\hat{\theta}_T$ be the output of Algorithm 1 after T iterations. Under the empirical two-sided-PL SGDA and stability regime of Yang et al. (2020); Kang and Jang (2025),*

$$\mathbb{E}_{\mathcal{D}_N^b,\mathrm{alg}}\left[\mathcal{R}_b(Q_{\hat{\theta}_T}) - \mathcal{R}_b^\star\right] \le \Delta_{N,T},$$

where

$$\mathcal{R}_b^\star := \inf_{Q\in\mathcal{Q}} \mathcal{R}_b(Q),$$

$$\Delta_{N,T} = (1 + L/\rho)G\left[O\big((c_2+T)^{-\alpha}\big) + \frac{C}{N}\right] + \frac{D_0}{\Gamma_0 + T}, \qquad \alpha := \min\left\{\frac{1}{2}, \frac{3cc_1}{8}\right\}.$$

If Assumption 5.1 holds, then $\mathcal{R}_b^\star = 0$.

Remark 1 (Reading the constants). All constants above are those of Kang and Jang (2025); Yang et al. (2020), instantiated on the empirical minimax objective

$$F_N(\theta_1,\theta_2) := \frac{1}{N}\sum_{(s,a,s')\in\mathcal{D}_N^b}\left[\mathcal{L}_{\mathrm{TD}}(Q_{\theta_1})(s,a,s') - \gamma^2\big(V^{Q_{\theta_1}}(s') - \xi_{\theta_2}(s,a)\big)^2\right]$$

optimized by Algorithm 1, and on its primal profile $\Phi_N(\theta_1) := \max_{\theta_2\in B_\xi} F_N(\theta_1,\theta_2)$, over the compact region $B_Q \times B_\xi$ of Lemma A.10. By Lemma 3.1, Φ_N is the empirical BRM risk. None of the constants depends on N or T. They fall into three groups.

- **Landscape constants** of F_N and Φ_N, finite and positive by Assumption 5.3 and compactness, uniformly over datasets of size N and their replace-one neighbors:

$$\underbrace{L}_{\text{smoothness of } F_N}, \quad \underbrace{\rho}_{\text{strong concavity in } \theta_2}, \quad \underbrace{G}_{\text{per-sample gradient bound}}, \quad \underbrace{\mu_{\mathrm{PL}},\ \mu_{\mathrm{QG}}}_{\text{PL and quadratic growth of } \Phi_N}.$$

- **Algorithmic constants**, fixed by the harmonic step-size schedule $\tau_{1,t} = \tau_{2,t} = c_1/(c_2+t)$ of Algorithm 1. These are the step-size constants $c_1 > 0$ and $c_2 \ge 1$, together with the induced contraction rate $c := \min\{\mu_{\mathrm{PL}}/2, \rho/2\}$, which give the decay exponent α.

- **Imported constants** from the two source results:
  - $(1 + L/\rho)G$ transfers stability into generalization (Kang and Jang, 2025, Lem. 5).
  - C is the $O(1/N)$ stability constant (Kang and Jang, 2025, Thm. 3).

– D_0 and Γ_0 bound the empirical primal suboptimality by $D_0/(\Gamma_0 + T)$. They are, respectively, the constants ν and γ of Yang et al. (2020, Thm. 3.3), written here in upper case because the latter would otherwise clash with the discount factor γ.

Proof. The biconjugate identity in Lemma 3.1 expresses the population BRM risk as the profile of a minimax objective whose empirical counterpart is optimized by Algorithm 1. The empirical PL result gives the finite-T SGDA optimization term, and the algorithmic stability/generalization theorem for Bellman residual minimization converts empirical optimization into population excess-risk control. This yields the displayed bound. If $Q^* \in \mathcal{Q}$, then $\mathcal{T}Q^* = Q^*$, so $b_{Q^*} \equiv 0$ and $\mathcal{R}_b^\star = 0$. □

Lemma A.2 (Target-support closure). *If $U \subseteq \mathcal{S}$ is measurable with $d_S^{\pi^*}(U) = 0$, then there exists a $d_S^{\pi^*}$-null set $E \subseteq \mathcal{S}$ such that*

$$P(U \mid s, a) = 0 \qquad \text{for all } s \notin E \text{ and all } a \in \mathcal{A}.$$

Proof. Step 1 (flow identity). Write $d_{h,S}^{\pi^*}$ for the state marginal of $d_h^{\pi^*}$. By the Markov property, for every measurable $U \subseteq \mathcal{S}$ and $h \geq 0$,

$$d_{h+1,S}^{\pi^*}(U) = \mathbb{E}_{s \sim d_{h,S}^{\pi^*}}\Big[\sum_{a \in \mathcal{A}} \pi^*(a \mid s)\, P(U \mid s, a)\Big].$$

Multiplying by $(1-\gamma)\gamma^{h+1}$, summing over $h \geq 0$, and adding the $h = 0$ term $(1-\gamma)\nu_0(U)$ yields, by the definition $d^{\pi^*} = (1-\gamma)\sum_{h \geq 0} \gamma^h d_h^{\pi^*}$,

$$d_S^{\pi^*}(U) = (1-\gamma)\nu_0(U) + \gamma\, \mathbb{E}_{s \sim d_S^{\pi^*}}\big[g(s)\big], \qquad g(s) := \sum_{a \in \mathcal{A}} \pi^*(a \mid s)\, P(U \mid s, a).$$

Step 2. Suppose $d_S^{\pi^*}(U) = 0$. Both terms on the right are nonnegative, so $\mathbb{E}_{s \sim d_S^{\pi^*}}[g(s)] = 0$. Since $g \geq 0$, this forces $g(s) = 0$ for all s outside a $d_S^{\pi^*}$-null set E.

Step 3. Fix $s \notin E$. The summands $\pi^*(a \mid s)P(U \mid s, a)$ are nonnegative and sum to $g(s) = 0$, so each equals zero. As π^* is the softmax policy of the bounded function Q^* in Equation (1), $\pi^*(a \mid s) > 0$ for every a, hence $P(U \mid s, a) = 0$ for every $a \in \mathcal{A}$. □

Lemma A.3 (Exact target-support Bellman identification). *Let Q be bounded. If*

$$b_Q(s, a) = 0 \qquad d^{\pi^*}\text{-a.e.},$$

then

$$\|Q - Q^*\|_{\infty,*} = 0.$$

Consequently, under Assumption 5.2,

$$\mathcal{R}_b(Q) = 0 \quad \Longrightarrow \quad \|Q - Q^*\|_{\infty,*} = 0.$$

Proof. Step 1 (from d^{π^*}-a.e. to all actions). For every measurable $U \subseteq \mathcal{S}$ and $a \in \mathcal{A}$,

$$d^{\pi^*}(U \times \{a\}) = \mathbb{E}_{s \sim d_S^{\pi^*}}\big[\mathbf{1}_U(s)\, \pi^*(a \mid s)\big],$$

and $\pi^*(a \mid s) > 0$ for every (s, a). Hence a set is d^{π^*}-null iff its state projection is $d_S^{\pi^*}$-null for each a. Since $\mathcal{A}$ is finite, the union of these null sets is null, so there is a $d_S^{\pi^*}$-null set E_0 with

$$\mathcal{T}Q(s, a) = Q(s, a) \qquad \text{for all } s \notin E_0,\ a \in \mathcal{A}.$$

Step 2 (error recursion). Let $e := Q - Q^*$ and

$$M := \|e\|_{\infty,*} = \operatorname*{ess\,sup}_{s \sim d_S^{\pi^*}} \max_{a \in \mathcal{A}} |e(s, a)|,$$

which is finite since Q and Q^* are bounded. Using $Q^* = \mathcal{T}Q^*$, for $s \notin E_0$ and every $a \in \mathcal{A}$,

$$e(s, a) = \gamma\, \mathbb{E}_{s' \sim P(\cdot \mid s, a)}\big[V^Q(s') - V^{Q^*}(s')\big].$$

The log-sum-exp map is 1-Lipschitz in the sup norm, so

$$|V^Q(s') - V^{Q^*}(s')| \le \max_{a' \in \mathcal{A}} |e(s', a')|.$$

Step 3 (the bound propagates to s'). Let

$$U := \Big\{ s \in \mathcal{S} : \max_{a \in \mathcal{A}} |e(s, a)| > M \Big\}.$$

By definition of the essential supremum, $d_S^{\pi^*}(U) = 0$. Lemma A.2 therefore gives a $d_S^{\pi^*}$-null set E_1 with $P(U \mid s, a) = 0$ for all $s \notin E_1$ and all a. For $s \notin E_0 \cup E_1$, the integrand in Step 2 is thus bounded by M for $P(\cdot \mid s, a)$-almost every s', and

$$|e(s, a)| \le \gamma M \qquad \text{for all } s \notin E_0 \cup E_1,\ a \in \mathcal{A}.$$

Step 4 (contraction). Since $E_0 \cup E_1$ is $d_S^{\pi^*}$-null, taking the essential supremum in the last display gives $M \le \gamma M$. As $\gamma < 1$, this forces $M = 0$.

Second claim. If $\mathcal{R}_b(Q) = 0$, then $b_Q = 0$ d^{π_b}-a.e. By Assumption 5.2,

$$d^{\pi^*}(W) \le C_\Pi \, d^{\pi_b}(W) \qquad \text{for every measurable } W \subseteq \mathcal{S} \times \mathcal{A},$$

so every d^{π_b}-null set is d^{π^*}-null. Thus $b_Q = 0$ d^{π^*}-a.e., and the first claim applies. □

Lemma A.4 (Compact identification modulus)**.** *Under Assumptions 5.2 and 5.3, recall the target-support identification modulus*

$$\omega_\infty(\varepsilon) = \sup\big\{ \|Q - Q^*\|_{\infty,*} \ :\ Q \in \mathcal{Q}_B,\ \mathcal{R}_b(Q) \le \varepsilon \big\}, \qquad \mathcal{Q}_B = \{Q_\theta : \theta \in B_Q\}.$$

Then

$$\omega_\infty(\varepsilon) \to 0 \qquad \textit{as } \varepsilon \downarrow 0.$$

Proof. Step 1 (continuity of $\theta \mapsto \mathcal{R}_b(Q_\theta)$ on B_Q). Assumption 5.3(i) provides the global bound

$$\sup_{\theta \in B_Q} \sup_{(s,a) \in \mathcal{S} \times \mathcal{A}} \|\nabla_\theta Q_\theta(s, a)\|_2 \le B.$$

Since B_Q is a ball, hence convex, the mean value theorem gives

$$\|Q_{\theta_1} - Q_{\theta_2}\|_\infty \le B \, \|\theta_1 - \theta_2\|_2 \qquad \text{for all } \theta_1, \theta_2 \in B_Q.$$

The log-sum-exp map is 1-Lipschitz in the sup norm, so $\|\mathcal{T}Q_{\theta_1} - \mathcal{T}Q_{\theta_2}\|_\infty \le \gamma \|Q_{\theta_1} - Q_{\theta_2}\|_\infty$ and therefore

$$\|b_{Q_{\theta_1}} - b_{Q_{\theta_2}}\|_\infty \le (1 + \gamma) B \, \|\theta_1 - \theta_2\|_2.$$

Moreover, b_{Q_θ} is uniformly bounded on B_Q: writing $b_{Q_\theta} = (\mathcal{T}Q_\theta - \mathcal{T}Q^*) + (Q^* - Q_\theta)$ and using $\mathcal{T}Q^* = Q^*$,

$$\bar{b} := \sup_{\theta \in B_Q} \|b_{Q_\theta}\|_\infty \le (1 + \gamma) \sup_{\theta \in B_Q} \|Q_\theta - Q^*\|_\infty < \infty,$$

the last supremum being finite because $\theta \mapsto Q_\theta$ is continuous on the compact set B_Q and Q^* is bounded. Hence

$$\big|\mathcal{R}_b(Q_{\theta_1}) - \mathcal{R}_b(Q_{\theta_2})\big| \le \mathbb{E}_{d^{\pi_b}}\Big[\, |b_{Q_{\theta_1}} - b_{Q_{\theta_2}}| \, \big(|b_{Q_{\theta_1}}| + |b_{Q_{\theta_2}}| \big) \Big] \le 2\bar{b} \, (1 + \gamma) B \, \|\theta_1 - \theta_2\|_2,$$

so $\theta \mapsto \mathcal{R}_b(Q_\theta)$ is continuous on B_Q.

Step 2 (contradiction). Suppose that the conclusion fails. Since ω_∞ does not decrease, there exist $\eta > 0$, $\varepsilon_n \downarrow 0$, and $\theta_n \in B_Q$ with

$$\mathcal{R}_b(Q_{\theta_n}) \le \varepsilon_n, \qquad \|Q_{\theta_n} - Q^*\|_{\infty,*} \ge \eta.$$

By compactness of B_Q, a subsequence (not relabeled) converges to some $\theta_\infty \in B_Q$. Step 1 gives

$$\mathcal{R}_b(Q_{\theta_\infty}) = \lim_{n \to \infty} \mathcal{R}_b(Q_{\theta_n}) = 0,$$

and Lemma A.3 then yields $\|Q_{\theta_\infty} - Q^*\|_{\infty,*} = 0$. By the triangle inequality and $\|\cdot\|_{\infty,*} \le \|\cdot\|_\infty$,

$$\eta \le \|Q_{\theta_n} - Q^*\|_{\infty,*} \le \|Q_{\theta_n} - Q_{\theta_\infty}\|_\infty + \|Q_{\theta_\infty} - Q^*\|_{\infty,*} \le B \, \|\theta_n - \theta_\infty\|_2 \longrightarrow 0,$$

a contradiction. □

Lemma A.5 (Local Bellman metric subregularity)**.** *Let $Q \in \mathcal{Q}_B$ satisfy*

$$\|Q - Q^*\|_{\infty,*} \le \delta \qquad \textit{for some} \qquad \delta < \frac{1}{2}\log\frac{1}{\gamma}.$$

Then

$$\|Q - Q^*\|_{2,\pi^*}^2 \le \frac{C_\Pi}{\left(1 - \sqrt{\gamma e^{2\delta}}\right)^2}\mathcal{R}_b(Q).$$

For entropy temperature $\lambda \neq 1$, the factor $e^{2\delta}$ becomes $e^{2\delta/\lambda}$.

Proof. Throughout, write $e_Q := Q - Q^*$, so that $\|e_Q\|_{\infty,*} \le \delta$ by hypothesis, and note that e_Q is bounded since $Q \in \mathcal{Q}_B$ and Q^* are.

Step 1 (linearization of the soft Bellman operator). For $u \in [0,1]$ let $Q_u := Q^* + u\, e_Q$ and let π_{Q_u} be the softmax policy of Q_u. Since $\nabla_x \log \sum_a e^{x_a}$ is the softmax vector,

$$\frac{d}{du}V^{Q_u}(s) = \sum_{a\in\mathcal{A}} \pi_{Q_u}(a \mid s)\, e_Q(s,a),$$

and integrating over $u \in [0,1]$ gives

$$V^{Q}(s) - V^{Q^*}(s) = \sum_{a\in\mathcal{A}} \bar{\pi}_Q(a \mid s)\, e_Q(s,a), \qquad \bar{\pi}_Q(a \mid s) := \int_0^1 \pi_{Q_u}(a \mid s)\, du.$$

Note that $\bar{\pi}_Q(\cdot \mid s)$ is a probability distribution on $\mathcal{A}$ for every s. Defining, for bounded f,

$$(P_{\bar{\pi}_Q} f)(s,a) := \mathbb{E}_{s'\sim P(\cdot|s,a)}\Big[\sum_{a'\in\mathcal{A}} \bar{\pi}_Q(a' \mid s')\, f(s',a')\Big],$$

we obtain the pointwise identity $\mathcal{T}Q - \mathcal{T}Q^* = \gamma P_{\bar{\pi}_Q} e_Q$ on $\mathcal{S}\times\mathcal{A}$. Since $Q^* = \mathcal{T}Q^*$,

$$e_Q = (Q - \mathcal{T}Q) + (\mathcal{T}Q - \mathcal{T}Q^*) = -\,b_Q + \gamma P_{\bar{\pi}_Q} e_Q.$$

Step 2 (density-ratio bound on the target support). Let $U := \{s \in \mathcal{S} : \max_a |e_Q(s,a)| > \delta\}$, so that $d_S^{\pi^*}(U) = 0$ by definition of $\|\cdot\|_{\infty,*}$. Fix $s \notin U$ and $u \in [0,1]$. Since $|Q_u(s,a) - Q^*(s,a)| = u|e_Q(s,a)| \le \delta$ for every a,

$$\pi_{Q_u}(a \mid s) = \frac{e^{Q_u(s,a)}}{\sum_{a'} e^{Q_u(s,a')}} \le \frac{e^{\delta}\, e^{Q^*(s,a)}}{e^{-\delta}\sum_{a'} e^{Q^*(s,a')}} = e^{2\delta}\,\pi^*(a \mid s).$$

Averaging over u,

$$\bar{\pi}_Q(a \mid s) \le e^{2\delta}\,\pi^*(a \mid s) \qquad \text{for all } s \notin U,\ a \in \mathcal{A}.$$

Step 3 (the bound propagates to s'). By Lemma A.2 applied to U, there is a $d_S^{\pi^*}$-null set E with $P(U \mid s,a) = 0$ for all $s \notin E$ and all a. Hence, for every bounded f and every (s,a) with $s \notin E$, the bound of Step 2 holds for $P(\cdot \mid s,a)$-almost every s', and

$$(P_{\bar{\pi}_Q} f^2)(s,a) \le e^{2\delta}\,(P_{\pi^*} f^2)(s,a),$$

where P_{π^*} is defined as $P_{\bar{\pi}_Q}$ with π^* in place of $\bar{\pi}_Q$. Since E is $d_S^{\pi^*}$-null, this inequality holds d^{π^*}-a.e.

Step 4 (flow inequality). Applying the argument of Lemma A.2, Step 1, to state–action pairs, for every bounded f,

$$\mathbb{E}_{(s,a)\sim d^{\pi^*}}\big[f^2(s,a)\big] = (1-\gamma)\,\mathbb{E}_{s\sim\nu_0,\, a\sim\pi^*(\cdot|s)}\big[f^2(s,a)\big] + \gamma\,\mathbb{E}_{(s,a)\sim d^{\pi^*}}\big[(P_{\pi^*} f^2)(s,a)\big].$$

Dropping the nonnegative first term,

$$\mathbb{E}_{d^{\pi^*}}\big[P_{\pi^*} f^2\big] \le \frac{1}{\gamma}\,\|f\|_{2,\pi^*}^2.$$

Step 5 ($\gamma P_{\bar{\pi}_Q}$ is a contraction in $L^2(d^{\pi^*})$). For bounded f, Jensen's inequality for the conditional expectation $P_{\bar{\pi}_Q}$ gives $(P_{\bar{\pi}_Q} f)^2 \le P_{\bar{\pi}_Q} f^2$ pointwise. Combining with Steps 3 and 4,

$$\|\gamma P_{\bar{\pi}_Q} f\|_{2,\pi^*}^2 \le \gamma^2 \, \mathbb{E}_{d^{\pi^*}}\left[P_{\bar{\pi}_Q} f^2\right] \le \gamma^2 e^{2\delta} \, \mathbb{E}_{d^{\pi^*}}\left[P_{\pi^*} f^2\right] \le \gamma e^{2\delta} \, \|f\|_{2,\pi^*}^2 .$$

Thus $\|\gamma P_{\bar{\pi}_Q} f\|_{2,\pi^*} \le \sqrt{\gamma e^{2\delta}} \, \|f\|_{2,\pi^*}$, and $\sqrt{\gamma e^{2\delta}} < 1$ precisely because $\delta < \frac{1}{2} \log \frac{1}{\gamma}$.

Step 6 (conclusion). Taking $\|\cdot\|_{2,\pi^*}$ norms in the identity of Step 1 and applying Step 5 with $f = e_Q$,

$$\|e_Q\|_{2,\pi^*} \le \|b_Q\|_{2,\pi^*} + \sqrt{\gamma e^{2\delta}} \, \|e_Q\|_{2,\pi^*},$$

and since $\|e_Q\|_{2,\pi^*} < \infty$,

$$\|Q - Q^*\|_{2,\pi^*} \le \frac{1}{1 - \sqrt{\gamma e^{2\delta}}} \, \|b_Q\|_{2,\pi^*}.$$

Finally, Assumption 5.2 gives $\|b_Q\|_{2,\pi^*}^2 \le C_\Pi \|b_Q\|_{2,\pi_b}^2 = C_\Pi \, \mathcal{R}_b(Q)$, and squaring the previous display proves the lemma. □

Corollary A.6 (Derived population Bellman-tangent nondegeneracy)**.** *Let $\theta^\star \in B_Q$ satisfy $Q_{\theta^\star} = Q^*$ and define, for any parameter direction v,*

$$h_v(s,a) := \nabla_\theta Q_{\theta^\star}(s,a)^\top v,$$

$$\dot{b}_v := D_\theta(\mathcal{T} Q_\theta - Q_\theta)|_{\theta=\theta^\star}[v] = \gamma P_{\pi^*} h_v - h_v.$$

Then

$$\|\dot{b}_v\|_{2,\pi_b}^2 \ge \frac{(1-\sqrt{\gamma})^2}{C_\Pi} \|h_v\|_{2,\pi^*}^2, \qquad \forall v.$$

Thus the relevant population Bellman-residual tangent map is nondegenerate in the target Q-function metric as a consequence of the Bellman structure and Assumption 5.2; it is not an additional NTK assumption.

Proof. At $Q = Q^*$, the interpolation policy in the proof of Lemma A.5 is exactly π^* and $\delta = 0$. Repeating the contraction argument gives

$$\|h_v\|_{2,\pi^*} \le \frac{1}{1-\sqrt{\gamma}} \|h_v - \gamma P_{\pi^*} h_v\|_{2,\pi^*} = \frac{1}{1-\sqrt{\gamma}} \|\dot{b}_v\|_{2,\pi^*}.$$

Assumption 5.2 gives

$$\|\dot{b}_v\|_{2,\pi^*}^2 \le C_\Pi \|\dot{b}_v\|_{2,\pi_b}^2.$$

Combining the two displays proves the claim. □

Theorem A.7 (Bellman-subregular Q^*-estimation bound, formal)**.** *Suppose Assumptions 5.1, 5.2, and 5.3 hold, and let $Q_{\hat{\theta}_T}$ be the Q-function returned by Algorithm 1 after T iterations with step sizes $\tau_{1,t} = \tau_{2,t} = c_1/(c_2 + t)$, where $c_1 > 0$ and $c_2 \ge 1$ satisfy the conditions of Lemma A.1. By Lemma A.4, fix any $\bar{\varepsilon} > 0$ with*

$$\delta_{\bar{\varepsilon}} := \omega_\infty(\bar{\varepsilon}) < \tfrac{1}{2} \log \tfrac{1}{\gamma},$$

and let $D_Q := \sup_{\theta \in B_Q} \|Q_\theta - Q^*\|_{2,\pi^*}^2$, which is finite by compactness of B_Q. Then

$$\mathbb{E}\big[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2\big] \le C_{\mathrm{Bell}} \, \Delta_{N,T}, \qquad C_{\mathrm{Bell}} := \frac{C_\Pi}{\big(1 - \sqrt{\gamma e^{2\delta_{\bar{\varepsilon}}}}\big)^2} + \frac{D_Q}{\bar{\varepsilon}},$$

where $\Delta_{N,T}$ is the population excess-risk bound of Lemma A.1. In particular,

$$\mathbb{E}\big[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2\big] = O\big((c_2 + T)^{-\alpha}\big) + O(1/N) + O(1/T),$$

with C_{Bell} and the constants in $\Delta_{N,T}$ independent of N and T.

Proof of Theorem A.7. By Lemma A.4, choose $\bar{\varepsilon} > 0$ such that

$$\delta_{\bar{\varepsilon}} := \omega_\infty(\bar{\varepsilon}) < \frac{1}{2}\log\frac{1}{\gamma}.$$

Let

$$A := \{\mathcal{R}_b(Q_{\hat{\theta}_T}) \le \bar{\varepsilon}\}.$$

On A, the definition of ω_∞ gives

$$\|Q_{\hat{\theta}_T} - Q^*\|_{\infty,*} \le \delta_{\bar{\varepsilon}}.$$

Therefore Lemma A.5 yields

$$\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2 \mathbf{1}_A \le \frac{C_\Pi}{\left(1 - \sqrt{\gamma e^{2\delta_{\bar{\varepsilon}}}}\right)^2} \mathcal{R}_b(Q_{\hat{\theta}_T}) \mathbf{1}_A.$$

Taking expectations and dropping $\mathbf{1}_A$ on the right gives

$$\mathbb{E}[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2 \mathbf{1}_A] \le \frac{C_\Pi}{\left(1 - \sqrt{\gamma e^{2\delta_{\bar{\varepsilon}}}}\right)^2} \mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})].$$

On A^c, compactness gives

$$\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2 \le D_Q := \sup_{\theta \in B_Q} \|Q_\theta - Q^*\|_{2,\pi^*}^2 < \infty.$$

By Markov's inequality,

$$\mathbb{P}(A^c) \le \frac{\mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})]}{\bar{\varepsilon}}.$$

Thus

$$\mathbb{E}[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2 \mathbf{1}_{A^c}] \le \frac{D_Q}{\bar{\varepsilon}} \mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})].$$

Define

$$K_{\mathrm{Bell}}(\bar{\varepsilon}) := \frac{C_\Pi}{\left(1 - \sqrt{\gamma e^{2\delta_{\bar{\varepsilon}}}}\right)^2} + \frac{D_Q}{\bar{\varepsilon}}.$$

Combining the bounds on A and A^c gives

$$\mathbb{E}[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2] \le K_{\mathrm{Bell}}(\bar{\varepsilon}) \mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})].$$

By Lemma A.1 and Assumption 5.1,

$$\mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T})] = \mathbb{E}[\mathcal{R}_b(Q_{\hat{\theta}_T}) - \mathcal{R}_b^\star] \le \Delta_{N,T}.$$

Thus

$$\mathbb{E}[\|Q_{\hat{\theta}_T} - Q^*\|_{2,\pi^*}^2] \le K_{\mathrm{Bell}}(\bar{\varepsilon})\, \Delta_{N,T}.$$

The theorem follows by taking $C_{\mathrm{Bell}} := K_{\mathrm{Bell}}(\bar{\varepsilon})$ and absorbing this finite constant into the big-O bound. □

Corollary A.8 (Uniform Bellman-residual bound)**.** *For every bounded Q,*

$$\|Q - Q^*\|_\infty \le \frac{1}{1-\gamma} \|\mathcal{T}Q - Q\|_\infty.$$

Consequently,

$$\|Q - Q^*\|_{2,\pi^*}^2 \le \frac{1}{(1-\gamma)^2} \|\mathcal{T}Q - Q\|_\infty^2.$$

Proof. Using the contraction of the soft Bellman optimality operator in the sup norm,

$$\begin{aligned} \|Q - Q^*\|_\infty &= \|Q - \mathcal{T}Q + \mathcal{T}Q - \mathcal{T}Q^*\|_\infty \\ &\le \|Q - \mathcal{T}Q\|_\infty + \|\mathcal{T}Q - \mathcal{T}Q^*\|_\infty \\ &\le \|\mathcal{T}Q - Q\|_\infty + \gamma \|Q - Q^*\|_\infty. \end{aligned}$$

Rearranging proves the first display; the $L_2(d^{\pi^*})$ statement follows because every probability-weighted L_2 norm is bounded by the sup norm. □

Definition A.9 (Polyak-Lojasiewicz (PL) condition)**.** A function $g : \Theta \mapsto \mathbb{R}$ is said to satisfy the Polyak-Łojasiewicz (PL) condition with respect to the ℓ_2 norm if g has a nonempty solution set and a finite minimal value $g(\theta^*)$ for $\theta^* \in \Theta \subseteq \mathbb{R}^d$, and there exists $\mu_{PL} > 0$ such that

$$\frac{1}{2}\|\nabla g(\theta)\|_2^2 \ge \mu_{PL}\big(g(\theta) - g(\theta^*)\big), \qquad \forall \theta \in \Theta.$$

A.2 NEURAL NETWORK CLASS AND LINEAR CLASS

Lemma A.10 (Compact empirical Jacobian regularity for linear and sufficiently wide neural classes)**.** *Let $Z_Q = \{z_i\}_{i=1}^{M_Q}$ and $Z_\xi = \{\bar{z}_i\}_{i=1}^{M_\xi}$ be finite empirical evaluation sets generated by $\mathcal{D}_N^b$, with $M_Q := |Z_Q|$ and $M_\xi := |Z_\xi|$ as in Assumption 5.3, and let*

$$J_Q(\theta_1; Z_Q) := \begin{bmatrix} \nabla_{\theta_1} Q_{\theta_1}(z_1)^\top \\ \vdots \\ \nabla_{\theta_1} Q_{\theta_1}(z_{M_Q})^\top \end{bmatrix}, \qquad J_\xi(\theta_2; Z_\xi) := \begin{bmatrix} \nabla_{\theta_2} \xi_{\theta_2}(\bar{z}_1)^\top \\ \vdots \\ \nabla_{\theta_2} \xi_{\theta_2}(\bar{z}_{M_\xi})^\top \end{bmatrix}$$

denote the empirical output Jacobians. Let $B_Q = B(\theta_{1,0}, R_Q)$ and $B_\xi = B(\theta_{2,0}, R_\xi)$ be closed finite-radius parameter balls containing the initial points, the optimization iterates, the relevant empirical minimizers, and, under Assumption 5.1, a realizability parameter $\theta^\star$ with $Q_{\theta^\star} = Q^*$.

Suppose one of the following parameterization conditions holds.

(a) ***Linear classes.*** *The functions are linear,*

$$Q_{\theta_1}(z) = \theta_1^\top \varphi_Q(z), \qquad \xi_{\theta_2}(\bar{z}) = \theta_2^\top \varphi_\xi(\bar{z}),$$

the features are uniformly bounded on their domains, and the empirical feature matrices have full row rank.

(b) ***Sufficiently wide smooth neural networks.*** *The functions Q_{θ_1} and ξ_{θ_2} are feedforward networks with C^2 hidden activations, fixed depth $H = O(1)$, and hidden width*

$$m = \widetilde{\Omega}\left(\frac{n_{\mathrm{eval}} R_\star^{6H+2}}{\lambda_\star^2}\right),$$

where $n_{\mathrm{eval}} := M_Q + M_\xi$, $R_\star := \max\{R_Q, R_\xi, 1\}$, and

$$\lambda_\star := \min\left\{\lambda_{\min}(K_{Q,\infty}(Z_Q)), \lambda_{\min}(K_{\xi,\infty}(Z_\xi))\right\} > 0.$$

Here $K_{Q,\infty}(Z_Q)$ and $K_{\xi,\infty}(Z_\xi)$ are the corresponding limiting tangent kernels at random initialization, and the $\widetilde{\Omega}(\cdot)$ notation suppresses logarithmic and fixed problem-dependent factors. Equivalently, for non-fixed depth H, the required width scales as the displayed polynomial order in n_{eval} and $\lambda_\star^{-1}$ with the depth dependence $R_\star^{6H+2}$.

Then, in the linear case deterministically and in the neural-network case with high probability over the initialization, the following properties hold on B_Q and B_ξ.

(i) ***Smoothness and bounded derivatives.*** *For every $z \in Z_Q$, the map $\theta_1 \mapsto Q_{\theta_1}(z)$ is twice continuously differentiable on B_Q, and there exist constants $B_{Q,1}, B_{Q,2} < \infty$ such that*

$$\sup_{\theta_1 \in B_Q} \sup_{z \in Z_Q} \|\nabla_{\theta_1} Q_{\theta_1}(z)\|_2 \le B_{Q,1}, \qquad \sup_{\theta_1 \in B_Q} \sup_{z \in Z_Q} \|\nabla_{\theta_1}^2 Q_{\theta_1}(z)\|_{\mathrm{op}} \le B_{Q,2}.$$

For every $\bar{z} \in Z_\xi$, the map $\theta_2 \mapsto \xi_{\theta_2}(\bar{z})$ is twice continuously differentiable on B_ξ, and analogous finite constants $B_{\xi,1}, B_{\xi,2} < \infty$ exist. Additionally, the Q-parameterization admits a finite global first-derivative bound B on B_Q:

$$\sup_{\theta_1 \in B_Q} \sup_{(s,a) \in \mathcal{S} \times \mathcal{A}} \|\nabla_{\theta_1} Q_{\theta_1}(s,a)\|_2 \le B.$$

(ii) ***Empirical-output Q-Jacobian conditioning.*** *There exists $\mu_Q > 0$ such that, for every $\theta_1 \in B_Q$,*

$$J_Q(\theta_1; Z_Q) J_Q(\theta_1; Z_Q)^\top \succeq \mu_Q I_{M_Q}.$$

Equivalently, for every $u \in \mathbb{R}^{M_Q}$,

$$\|J_Q(\theta_1; Z_Q)^\top u\|_2^2 \ge \mu_Q \|u\|_2^2.$$

(iii) ***Empirical-output auxiliary Jacobian conditioning.*** *There exists $\mu_\xi > 0$ such that, for every $\theta_2 \in B_\xi$,*

$$J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_{M_\xi}.$$

Proof. The argument is finite-dimensional because Z_Q and Z_ξ are finite empirical evaluation sets.

Linear parameterizations. Let

$$\Phi_Q(Z_Q) := \begin{bmatrix} \varphi_Q(z_1)^\top \\ \vdots \\ \varphi_Q(z_{M_Q})^\top \end{bmatrix}, \qquad \Phi_\xi(Z_\xi) := \begin{bmatrix} \varphi_\xi(\bar{z}_1)^\top \\ \vdots \\ \varphi_\xi(\bar{z}_{M_\xi})^\top \end{bmatrix}.$$

Then

$$\mathbf{Q}_{\theta_1}(Z_Q) = \Phi_Q(Z_Q)\theta_1, \qquad J_Q(\theta_1; Z_Q) = \Phi_Q(Z_Q),$$

and analogously

$$\boldsymbol{\xi}_{\theta_2}(Z_\xi) = \Phi_\xi(Z_\xi)\theta_2, \qquad J_\xi(\theta_2; Z_\xi) = \Phi_\xi(Z_\xi).$$

Thus the Jacobian conditioning properties hold whenever

$$\Phi_Q(Z_Q)\Phi_Q(Z_Q)^\top \succeq \mu_Q I_{M_Q}, \qquad \Phi_\xi(Z_\xi)\Phi_\xi(Z_\xi)^\top \succeq \mu_\xi I_{M_\xi}$$

for some $\mu_Q, \mu_\xi > 0$, which is exactly the full-row-rank condition. The derivative bounds are immediate: the first derivatives are the feature vectors and the second derivatives are zero. The global constant B is finite when the Q-features are uniformly bounded on $\mathcal{S} \times \mathcal{A}$.

The compact-region condition is not specific to neural networks. For a linear class, once the optimization is analyzed on a closed finite-radius parameter domain, for example through projected SGDA, bounded regularized iterates, or any finite run whose iterates are known to remain bounded, the relevant parameter region is compact in finite-dimensional Euclidean space. No transition-to-linearity or lazy-training argument is needed in the linear case because the empirical Jacobians are constant over the entire parameter space. The neural-network argument below is used only to ensure that the empirical tangent kernels remain well conditioned throughout a finite-radius neighborhood of random initialization.

Wide smooth neural networks. We give the argument for Q_{θ_1}; the proof for ξ_{θ_2} is identical after replacing $(Z_Q, \theta_1, M_Q, R_Q)$ by $(Z_\xi, \theta_2, M_\xi, R_\xi)$. Write

$$K_Q(\theta_1; Z_Q) := J_Q(\theta_1; Z_Q) J_Q(\theta_1; Z_Q)^\top.$$

For the finite set Z_Q, assume the limiting tangent kernel at initialization is strictly positive definite:

$$\lambda_Q^\star := \lambda_{\min}(K_{Q,\infty}(Z_Q)) > 0.$$

Finite-width tangent-kernel concentration gives, after increasing the width m if necessary,

$$\|K_Q(\theta_{1,0}; Z_Q) - K_{Q,\infty}(Z_Q)\|_2 \le \frac{\lambda_Q^\star}{4}$$

with high probability. By Weyl's inequality,

$$\lambda_{\min}(K_Q(\theta_{1,0}; Z_Q)) \ge \frac{3\lambda_Q^\star}{4}.$$

It remains to keep the kernel well conditioned on the whole ball $B_Q = B(\theta_{1,0}, R_Q)$. The transition-to-linearity/Hessian-control result of Liu et al. (2022) gives a high-probability bound

$$\sup_{\theta_1 \in B_Q} \max_{i \in [M_Q]} \|\nabla^2_{\theta_1} Q_{\theta_1}(z_i)\|_{\mathrm{op}} \le C_{Q,R}(m), \qquad C_{Q,R}(m) \to 0$$

as $m \to \infty$; in the notation of Liu et al. (2022),

$$C_{Q,R}(m) = \widetilde{O}(R_Q^{3H}/\sqrt{m}).$$

Hence, for every $\theta_1 \in B_Q$,

$$\|J_Q(\theta_1; Z_Q) - J_Q(\theta_{1,0}; Z_Q)\|_F \le \sqrt{M_Q}\, C_{Q,R}(m) R_Q.$$

The displayed width order makes this Jacobian perturbation small enough that the induced kernel perturbation is at most $\lambda_Q^\star/4$ uniformly on B_Q, up to the same logarithmic and fixed problem-dependent factors suppressed in $\widetilde{\Omega}(\cdot)$. Another application of Weyl's inequality yields

$$K_Q(\theta_1; Z_Q) \succeq \frac{\lambda_Q^\star}{2} I_{M_Q}, \qquad \forall \theta_1 \in B_Q.$$

Thus the Q-Jacobian conditioning holds with $\mu_Q = \lambda_Q^\star/2$. If the network has a smooth scalar output activation σ_{out} applied to a linear-output network and

$$\rho_Q := \inf_{\theta_1 \in B_Q} \min_{z \in Z_Q} |\sigma'_{\text{out}}(\widetilde{Q}_{\theta_1}(z))| > 0,$$

then the same argument gives the admissible constant $\mu_Q = \rho_Q^2 \lambda_Q^\star/2$.

The identical construction for the auxiliary network gives

$$J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_{M_\xi} \qquad \forall \theta_2 \in B_\xi,$$

with $\mu_\xi = \lambda_\xi^\star/2$ for a linear-output auxiliary network, or $\mu_\xi = \rho_\xi^2 \lambda_\xi^\star/2$ when a smooth output activation with derivative lower bound ρ_ξ is used. Finally, because the networks are C^2 and the parameter balls and empirical evaluation sets are finite, the local derivative bounds are finite. The global derivative constant B is finite under the standard bounded-input/compact-domain condition for the Q network. This proves all stated properties. □

B Related Works

Bellman Residual Minimization (BRM). The BRM approach (Antos et al., 2008) frames offline RL from the perspective of supervised learning, where the squared Bellman error plays the role of the supervised loss. It has been considered the only known offline RL method that can stably enforce Bellman consistency under nonlinear (e.g., neural network) parameterization (Jiang and Xie, 2025), but it suffers from the double sampling problem (Baird, 1995). Several works resolve this with a debiasing (dual) correction term (Antos et al., 2008; Dai et al., 2018), at the cost of turning BRM into a nonconvex minimax problem. Prior work has addressed the two halves of the resulting difficulty separately: Antos et al. (2008) control the statistical error but take the empirical minimizer as given; SBEED (Dai et al., 2018), together with its sharp sample-complexity analysis (Touati and Vincent, 2020), supplies an implementable SGDA procedure yet certifies convergence only to a stationary point; and Kang et al. (2025) identify a PL–strongly-concave geometry that yields global SGDA convergence, but for offline inverse RL rather than offline RL, and without controlling the resulting Q-estimation error. Our contribution is precisely the missing optimization guarantee for offline RL and we defer the formal error decomposition to Appendix E.

Paradigms that relax pointwise Bellman consistency. Three other families dominate offline RL under function approximation, and each buys tractability by weakening the consistency requirement that BRM enforces. Projected MSBE (MSPBE) methods (Sutton et al., 2009; Patterson et al., 2022) project the residual onto the function class before squaring, discarding the orthogonal component, so a small MSPBE does not certify pointwise consistency. Fitted fixed-point methods and their TD-based practical variants (Ernst et al., 2005; Munos and Szepesvári, 2008), retain the semi-gradient objective and handle distribution shift by other means—policy constraints (Fujimoto et al., 2019; Kumar et al., 2019) or pessimism (Jin et al., 2021)—and remain exposed to the deadly triad (Tsitsiklis and Van Roy, 1996a; Van Hasselt et al., 2018; Sutton and Barto, 2018); our baseline Conservative Q-Learning (CQL) (Kumar et al., 2020) is a widely used representative, adding a conservatism penalty whose magnitude is set by a hyperparameter rather than by the Bellman structure. Marginalized importance sampling and the DICE family (Liu et al., 2018; Nachum et al., 2019a;b; Zhang et al., 2020) instead constrain the stationary distribution: the Bellman error enters only through the convex conjugate f^* of an f-divergence regularizer, certifying distribution-level matching rather than pointwise Bellman consistency. Our baseline, OptiDICE (Lee et al., 2021), reduces the alternating SGDA to a single convex minimization over ν. However, evaluating f^* on the single-sample residual $\hat{e}_\nu(s, a, s')$ is biased under stochastic transitions — the same double-sampling issue that our dual variable ξ removes (Section 3). Our guarantee is for the squared Bellman error itself.

C EXPERIMENTAL DETAILS

In this section, we describe the common training configurations, network architectures, and evaluation protocols shared across all evaluated algorithms (`Off-GLADIUS`, CQL, OptiDICE, BRM-FPI, and SBEED) for a fair comparison.

Computational Resources. All experiments were conducted on a server equipped with three NVIDIA RTX PRO 5000 Blackwell GPUs, each with 48 GB memory, of which two were used for our experiments. The CUDA version reported by `nvidia-smi` was 13.0 (driver 580.173.02). We implemented all methods in Python 3.11 using PyTorch 2.13.0 (CUDA 13.0 build).

C.1 DATASETS

Mixed dataset (Section 6.1). As these are standard online environments, we constructed offline datasets to simulate the offline RL setting. For each environment we first trained a PPO agent (Schulman et al., 2017) to obtain a sub-optimal behavior policy π_{ppo}, then aggregated 700,000 transitions from π_{ppo} with 300,000 from a uniform random policy, yielding 10^6 tuples $(s, a, r(s, a), s')$ of *medium*-level expertise with diverse exploratory coverage, following the mixed-dataset protocol of D4RL (Fu et al., 2020). The behavior policy is never exposed to any algorithm, and all methods access the data only through these samples.

Coverage-failure datasets (Section 6.2). The three datasets of Table 2 are LunarLander-v3 datasets of $N = 10^6$ transitions each, generated under a common protocol. Coverage of d^{π^*} is degraded by replacing part or all of the data with transitions from an *anti-expert*, a negative-softmax policy that inverts the action preferences of a near-optimal expert while retaining full action support. One dataset per condition, with the five seeds in Table 2 being training seeds.

Behavior policies. The expert is a PPO agent trained with the tuned LunarLander-v3 hyperparameters of RL Baselines3 Zoo (Raffin, 2020). Training stops once the deterministic evaluation return reaches 250, or after 2×10^6 environment steps. Under the stochastic action sampling used for data collection, its best checkpoint attains a mean episodic return of 227.9. The anti-expert requires no additional training. It samples actions from $\mathrm{softmax}(-\ell(s))$, where $\ell(s)$ denotes the expert's action logits, so it assigns the lowest probability to the action the expert prefers most while keeping every action possible. Its mean episodic return is -760.1, far below the -182.0 of a uniform-random policy.

Dataset construction. For a behavior fraction $p \in \{0,\ 0.5,\ 1\}$, we collect $\mathrm{round}(pN)$ transitions from the anti-expert and the remaining transitions from a uniform-random policy, sampling actions stochastically in both cases. The two sets are pooled and shuffled so that mini-batches mix both sources. The three values of p give the three datasets of Table 3. We distinguish environment termination from time-limit truncation, and zero the bootstrap target $V(s')$ only at termination.

Coverage. The three datasets fail to cover d^{π^*} in different ways. Uniform-random data are broad in support but place little mass on the states that π^* visits. Anti-expert data concentrate on trajectories that π^* avoids, and the 5:5 dataset lies between the two. Because the anti-expert never assigns zero probability to any action, Assumption 5.2 is not formally violated, but C_Π is large enough to make Theorem 5.4 vacuous (Section 6.2).

Table 3: Coverage-failure datasets (LunarLander-v3, $N = 10^6$ transitions each). BC is the mean episodic return of each dataset. In the 5:5 mixture, the anti-expert and uniform-random episodes return -760.4 and -179.8 on average, respectively.

Dataset	behavior : random	Episodes	BC
Uniform-random	0:10	10,751	-182.0
Anti-expert : random (5:5)	5:5	10,844	-472.2
Anti-expert only	10:0	10,960	-760.1

C.2 Shared Experimental Setup

Evaluation. Although training is performed purely offline, we periodically evaluate the learned policy online, every $1{,}000$ training iterations, and report the average episodic return. All results are averaged over 5 random seeds and reported as mean $\pm$ standard deviation, computed over the final 20% of evaluations.

Network Architecture and Initialization. For all algorithms, we parameterized the value and auxiliary networks using an MLP with two hidden layers of 128 units. To accommodate the ReLU activation functions used in the hidden layers, we initialized all neural network weights using orthogonal initialization with a gain of $\sqrt{2}$.

Common Training Hyperparameters. Across all algorithms and environments, we set the discount factor to $\gamma = 0.99$ and optimized all neural networks using the Adam optimizer. The total number of training iterations was tailored to the complexity of each environment: 20k iterations for CartPole-v1, 30k for Acrobot-v1, and 50k for LunarLander-v3.

C.3 Off-GLADIUS

Ascent-Descent Iteration. For `Off-GLADIUS`, we report learning curves with the x-axis measured in *iteration*. One such iteration is a complete update cycle: K_ξ consecutive updates of the auxiliary network ξ (the ascent step, lines 3–8 of Algorithm 1), followed by a single update of the Q network (the descent step).

In our implementation, the quantities τ_2, τ_1, λ, and K_ξ correspond to `lr_xi`, `lr_q`, `lmbda`, and `xi_steps`, respectively. These were tuned by grid search over:

- Temperature λ (`lmbda`): $\{0.01, 0.1, 1.0\}$
- Ascent step learning rate τ_2 (`lr_xi`): $\{10^{-4}, 3\times10^{-4}, 3\times10^{-3}\}$
- Descent step learning rate τ_1 (`lr_q`): $\{10^{-5}, 3\times10^{-5}, 3\times10^{-4}\}$
- Number of ascent steps K_ξ (`xi_steps`): $\{1, 5, 10\}$

C.4 Conservative Q-Learning (CQL)

CQL augments the standard Bellman error with a regularizer that pushes down the Q-values of out-of-distribution (OOD) actions while pushing up the Q-values of actions observed in the dataset, yielding the objective

$$\mathcal{L}(Q) = \tfrac{1}{2}\,\mathbb{E}_{\mathcal{D}}\Big[(Q(s,a)-y)^2\Big] + \alpha_{\text{CQL}}\,\mathbb{E}_{\mathcal{D}}\left[\log\sum_{a'}\exp Q(s,a') - Q(s,a)\right],$$
$$y := r + \gamma(1-d)\max_{a'} Q_{\bar\theta}(s',a'),$$

where $Q_{\bar\theta}$ denotes a target network synchronized every `target_update_freq` steps and d is the termination flag. The parameter α_{CQL} weights the penalty on OOD actions and therefore controls the degree of conservatism. Since CQL learns a single Q-network, only one learning rate `lr` is required. We searched over the following configurations.

- Conservatism parameter (α_{CQL}): $\{0.01, 0.1, 1.0, 5.0\}$
- Q-network learning rate (`lr`): $\{10^{-4}, 3\times10^{-4}, 10^{-3}\}$

C.5 OptiDICE

OptiDICE casts offline RL as a stationary distribution matching problem, where α_{OptiDICE} controls the strength of the f-divergence regularizer that keeps the learned state-action distribution close to the offline dataset. For the χ^2-divergence the inner maximizer admits the closed form

$$w^*_\nu(s,a,s') = \max\left(0,\ \frac{\hat e_\nu(s,a,s')}{\alpha_{\text{OptiDICE}}} + 1\right), \qquad \hat e_\nu(s,a,s') := r + \gamma\nu(s') - \nu(s),$$

which collapses the minimax problem into a single minimization over the value network ν. For policy extraction over a finite action space, we additionally fit an advantage regression network $\phi(s,a)$ to the target $\hat{e}_\nu(s,a,s')$ under an MSE objective and recover the policy through $w_\phi(s,a) = \max(0, \phi(s,a)/\alpha_{\text{OptiDICE}} + 1)$. Two learning rates are therefore required, `lr_nu` and `lr_phi`. We searched over the following configurations.

- Regularization parameter (α_{OptiDICE}): $\{0.01, 0.1, 1.0\}$
- Value network learning rate (`lr_nu`): $\{10^{-5}, 3\times10^{-5}, 3\times10^{-4}\}$
- Advantage regression network learning rate (`lr_phi`): $\{10^{-4}, 3\times10^{-4}, 3\times10^{-3}\}$

C.6 Bellman Residual Minimization based Fitted Policy Iteration (BRM-FPI)

BRM-FPI (Antos et al., 2008) minimizes the Bellman residual without entropy smoothing and removes the double-sampling bias with an auxiliary function h in place of our dual variable ξ. Each outer round freezes the evaluated policy $\pi_k = \text{greedy}(Q)$ and runs `inner_steps` of

$$\min_Q \max_h \ \hat{\mathbb{E}}_{\mathcal{D}}\Big[\big(y - Q(s,a)\big)^2 - \big(h(s,a) - y\big)^2\Big], \qquad y = r + \gamma\, Q\big(s', \pi_k(s')\big),$$

where `h_steps_per_q` ascent steps on h precede each descent step on Q, in the same role that `xi_steps` plays in `Off-GLADIUS`. We fix $K \times$ `inner_steps` $= T$ and `h_steps_per_q` $=$ `xi_steps`, so that both methods issue the same number of gradient updates, namely 120k, 180k and 550k for CartPole-v1, Acrobot-v1 and LunarLander-v3, and share the same x-axis of primal Q updates. Since `lr_h` and `lr_q` play the roles of τ_2 and τ_1, we reuse the `Off-GLADIUS` grid and add the two baseline-specific hyperparameters. We searched over the following configurations.

- Number of outer rounds (K): $\{1, 5, 10, 25\}$
- Number of ascent steps (`h_steps_per_q`): $\{1, 5, 10\}$
- Ascent step learning rate (`lr_h`): $\{10^{-4},\ 3\times10^{-4},\ 3\times10^{-3}\}$
- Descent step learning rate (`lr_q`): $\{10^{-5},\ 3\times10^{-5},\ 3\times10^{-4}\}$

C.7 SBEED

SBEED (Dai et al., 2018) minimizes the smoothed Bellman error through a primal–dual reformulation in which the policy is a separate network. With $\delta(s,a,s') := r + \gamma V(s') - \lambda \log \pi(a \mid s)$, the objective is

$$\mathcal{L}_\eta(V, \pi; \rho) = \mathbb{E}_{\mathcal{D}}\Big[\big(\delta(s,a,s') - V(s)\big)^2\Big] - \eta\, \mathbb{E}_{\mathcal{D}}\Big[\big(\delta(s,a,s') - \rho(s,a)\big)^2\Big], \tag{7}$$

where the dual $\rho(s,a)$ regresses onto $\mathbb{E}_{s'}[\delta(s,a,s')]$ and the second term removes the double-sampling bias, playing the role that ξ plays in `Off-GLADIUS`. Following Algorithm 2 of Dai et al. (2018), each iteration takes one stochastic gradient step on ρ, then one on V and one on π, with the primal–dual step sizes on two time scales; the policy is updated by the advantage-weighted gradient $-2\lambda\, \mathbb{E}_{\mathcal{D}}[A(s,a) \nabla \log \pi(a \mid s)]$ of Theorem 4 in that paper, with $A(s,a) = (1-\eta)\delta + \eta\rho(s,a) - V(s)$ standardized within the batch. The three networks share the two-hidden-layer architecture of Appendix C.2, with policy logits clipped to $[-10, 10]$. We fix $\eta = 0.1$ and the policy learning rate at 10^{-4}, match the training budget of `Off-GLADIUS` in primal iterations (20k, 30k and 50k for CartPole-v1, Acrobot-v1 and LunarLander-v3), and searched over the following configurations, sweeping λ first at the learning rates of Table 4 and then the two learning rates at the selected λ:

- Entropy-regularization parameter (λ): $\{0.01, 0.1, 1.0\}$
- Value network learning rate (`lr_v`): $\{10^{-5},\ 3\times10^{-5},\ 3\times10^{-4}\}$
- Dual network learning rate (`lr_rho`): $\{10^{-4},\ 3\times10^{-4},\ 3\times10^{-3}\}$

C.8 Selected Hyperparameters

Table 4 collects the configuration used for each method and environment, selected by the grid searches described above.

Table 4: Selected hyperparameters for each method and environment. All methods use a batch size of 1024, which was optimal across all three environments.

Method	Hyperparameter	CartPole-v1	Acrobot-v1	LunarLander-v3
Off-GLADIUS	Temperature λ (lmbda)	1.0	0.1	0.1
	Ascent lr τ_2 (lr_xi)	3×10^{-3}	3×10^{-4}	3×10^{-3}
	Descent lr τ_1 (lr_q)	3×10^{-4}	3×10^{-4}	3×10^{-4}
	Ascent steps K_ξ (xi_steps)	5	5	10
CQL	Conservatism (α_{CQL})	5.0	1.0	1.0
	Learning rate (lr)	3×10^{-4}	3×10^{-4}	1×10^{-3}
OptiDICE	Regularization (α_{OptiDICE})	1.0	1.0	1.0
	Value net lr (lr_nu)	3×10^{-4}	3×10^{-4}	3×10^{-4}
	Advantage net lr (lr_phi)	3×10^{-4}	3×10^{-4}	3×10^{-3}
BRM-FPI	Policy iterations K (K)	1	10	25
	Auxiliary lr (lr_h)	3×10^{-3}	3×10^{-4}	3×10^{-3}
	Q-network lr (lr_q)	3×10^{-4}	3×10^{-4}	3×10^{-4}
	Ascent steps (h_steps_per_q)	5	5	10
SBEED	Temperature λ (lam)	0.1	0.01	0.01
	Dual weight η (eta)	0.1	0.1	0.1
	Value net lr (lr_v)	3×10^{-4}	3×10^{-5}	3×10^{-4}
	Dual net lr (lr_rho)	3×10^{-3}	3×10^{-3}	3×10^{-4}

D Additional Ablations

D.1 Effect of the Dual Correction

The dual variable ξ is not an optimization heuristic layered onto BRM. As shown in Lemma 3.1, it is what makes the empirical objective an unbiased estimate of the MSBE. Since the bias it removes is exactly $\gamma^2 \text{Var}_{s' \sim P(\cdot|s,a)}[V^Q(s')]$, the theory makes a sharp prediction: the correction should have no effect under deterministic transitions, and should matter precisely as transitions become stochastic. We test this by removing the $\gamma^2(V^Q(s') - \xi(s,a))^2$ term, which reduces the empirical objective to a single-sample squared TD loss. We then systematically vary the transition noise via the enable_wind setting in LunarLander-v3, regenerating the dataset for each condition (Appendix C.1).

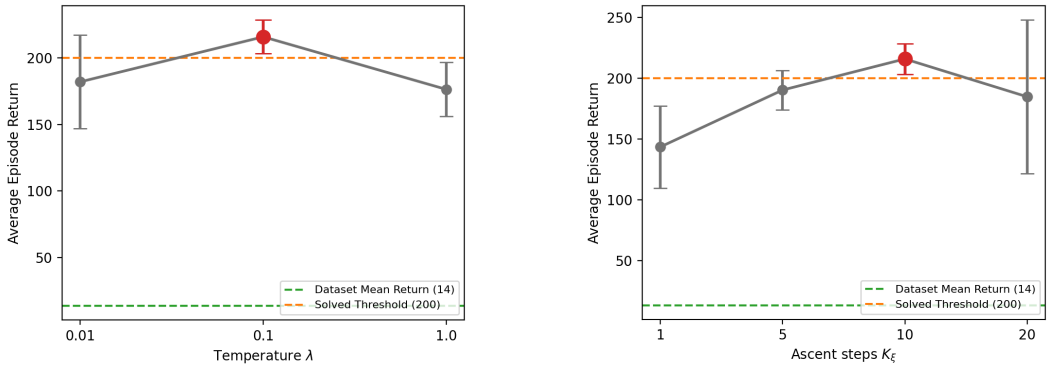
Table 5: Effect of the dual correction under increasing transition stochasticity (LunarLander-v3, 5 seeds). Naive BRM deletes the correction term, leaving a single-sample squared TD objective.

	wind_power		
Method	0	10	20
Off-GLADIUS	215.9 ± 12.6	$\mathbf{44.6 \pm 81.4}$	$\mathbf{-89.9 \pm 62.1}$
Naive BRM	198.6 ± 42.4	-112.5 ± 76.7	-169.0 ± 103.5

Table 5 matches the prediction. With deterministic transitions (wind_power $= 0$), the gap between the two objectives (17.4 return) is well within seed variance, consistent with there being no bias for the correction to remove. Once transitions become stochastic, the correction separates them by a wide margin, most clearly at wind_power $= 10$; at wind_power $= 20$, the task becomes sufficiently difficult that the performance of both methods degrades significantly, so the remaining gap is best read as a direction rather than a magnitude. We note that the auxiliary network here has far less capacity than would be needed to interpolate the 10^6 logged transitions. Consequently, ξ properly estimates a conditional expectation rather than fitting each observed s' exactly, which is precisely the regime in which the correction is meaningful.

D.2 Ablation Studies

Figure 3 varies the two hyperparameters specific to `Off-GLADIUS`, holding the remaining configuration fixed at the values of Table 4. Both exhibit an interior optimum rather than a monotone trend.



(a) Temperature λ (at $K_\xi = 10$). (b) Ascent steps K_ξ (at $\lambda = 0.1$).

Figure 3: Ablations for `Off-GLADIUS` on LunarLander-v3. Mean $\pm$ standard deviation over 5 seeds, computed over the final 20% of evaluations. The selected setting is shown in red.

For the entropy regularization parameter, $\lambda = 0.1$ is best, and both a weaker and a stronger setting lose roughly the same amount. This matches the role λ plays in the soft value $V^Q(s) = \lambda \ln \sum_a \exp(Q(s,a)/\lambda)$: as $\lambda \to 0$ the operator approaches the hard max and the smoothing our analysis relies on vanishes, whereas a large λ moves the soft Bellman fixed point away from Q^*.

For the number of ascent steps, performance improves from $K_\xi = 1$ to $K_\xi = 10$ as the dual variable ξ tracks its inner optimum more closely, then degrades at $K_\xi = 20$, where the across-seed standard deviation more than quadruples. Part of the gap at $K_\xi = 1$ reflects budget rather than inner-loop depth, since T is held fixed and each iteration costs $K_\xi + 1$ gradient updates (Table 6). We extend the sweep beyond the tuning grid of Appendix C.3 to $K_\xi = 20$ to confirm that the optimum is interior. $K_\xi = 10$ is the only setting whose error bar lies entirely above the solved threshold, and we adopt it for all LunarLander-v3 experiments.

D.3 Wall-Clock Cost

`Off-GLADIUS` performs K_ξ ascent updates for every descent update, so its per-iteration cost is higher than that of single-objective methods by design. Table 6 reports seconds per 1,000 ascent–descent iterations, i.e. the x-axis of Figure 1.

Table 6: Wall-clock cost per 1,000 ascent–descent iterations (seconds), with the multiplier relative to CQL in parentheses. Measured at the main-experiment configuration on the hardware described in Appendix C.

Method	CartPole-v1	Acrobot-v1	LunarLander-v3
`Off-GLADIUS`	5.220 (5.82×)	5.492 (5.74×)	9.898 (10.64×)
CQL	0.897 (1.00×)	0.957 (1.00×)	0.930 (1.00×)
OptiDICE	1.978 (2.21×)	2.024 (2.11×)	1.982 (2.13×)

The overhead relative to CQL tracks K_ξ almost exactly: CartPole-v1 and Acrobot-v1 use $K_\xi = 5$ and cost roughly 5.8×, while LunarLander-v3 uses $K_\xi = 10$ and costs 10.6×. This is because the per-update cost is nearly identical across all three methods (0.87–1.01 ms). The auxiliary network ξ shares the architecture of the Q network, so a ξ update costs what a Q update costs. The multiplier is therefore a single explicit knob, linear in cost. We pay it deliberately: solving the inner maximization more accurately before each descent step yields better and more stable returns, as the sweep over K_ξ in Appendix D.2 shows.

E Detailed Positioning within the BRM Literature

Table 7: Positioning of `Off-GLADIUS` within the BRM line of work. Opt. guarantee refers to the procedure actually run, not to an assumed oracle. Q^* bound indicates whether a bound on the value-function estimation error is available *for that procedure*.

Family (representative)	Objective	Opt. guarantee	Q^* bound
BRM, naive (Baird, 1995)	MSBE	—	—
BRM, ERM oracle (Antos et al., 2008)	debiased MSBE	*assumed*	oracle-conditional †
BRM, SGDA (SBEED) (Dai et al., 2018)	smoothed MSBE	stationary point	—
BRM, SGDA (Kang et al., 2025)†	smoothed MSBE	global (PL)	—
`Off-GLADIUS` (ours)	smoothed MSBE	**global (PL)**	**unconditional**

† Established for offline inverse RL / dynamic discrete choice rather than offline RL. *Oracle-conditional* means the value-error bound holds for the empirical minimizer, whose computation is outside the guarantee.

To locate our contribution precisely within the BRM line of work, it is useful to decompose the error of a BRM estimator. Let $\hat{\theta}_T$ denote the parameter returned after T iterations of a first-order method run on a dataset of N transitions, let $\mathcal{R}_b(Q_\theta)$ be the population objective defined in Section 5, and let $\theta^\star \in \operatorname{argmin}_{\theta\in\Theta} \mathcal{R}_b(Q_\theta)$. Then

$$\left\|Q^* - Q_{\hat{\theta}_T}\right\| \lesssim \underbrace{\varepsilon_{\mathrm{approx}}}_{\text{expressiveness of } \mathcal{Q}} + \underbrace{\varepsilon_{\mathrm{stat}}(N)}_{\widehat{\mathcal{R}}_N \to \mathcal{R}_b} + \underbrace{\varepsilon_{\mathrm{opt}}(T)}_{\hat{\theta}_T \to \theta^\star}, \tag{8}$$

where $\varepsilon_{\mathrm{approx}}$ vanishes under realizability (Assumption 5.1). The three surviving questions—how rich $\mathcal{Q}$ must be, what happens as $N \to \infty$, and what happens as $T \to \infty$—have been addressed to markedly different degrees in the literature.

Antos et al. (2008) established convergence of the statistical term: under a single β-mixing sample path, they gave finite-sample bounds showing that the minimizer of the empirical debiased BRM objective converges to its population counterpart as N grows. Their analysis, however, takes that empirical minimizer as given; because the debiased objective is a nonconvex minimax problem under nonlinear parameterization, the question of whether an implementable first-order method actually reaches it—i.e., the behavior of $\varepsilon_{\mathrm{opt}}(T)$—lies outside their guarantee. SBEED (Dai et al., 2018) substantially advanced the problem on the algorithmic side, combining Nesterov smoothing of the max operator with a Fenchel dual reformulation to obtain a smooth, single-sample-estimable saddle objective that can be optimized by stochastic gradient descent–ascent under neural function approximation, together with a sharp sample-complexity analysis (Touati and Vincent, 2020). Its optimization guarantee, however, certifies convergence only to a stationary point of the resulting nonconvex–concave problem, so $\varepsilon_{\mathrm{opt}}(T)$ is not driven to zero and global convergence remained unresolved (Jiang and Xie, 2025).

Our contribution is precisely the missing term. Building on the loss-landscape analysis of Kang et al. (2025), we use the fact that the smoothed BRM objective satisfies a Polyak–Łojasiewicz condition in parameter space under the Jacobian conditioning of Assumption 5.3, which we establish for linear and sufficiently wide classes in Lemma A.10; this rules out spurious stationary points and yields $\varepsilon_{\mathrm{opt}}(T) \to 0$ at an explicit rate for the SGDA updates of Algorithm 1. Theorem 5.4 then controls both remaining terms of Equation (8) simultaneously, with the T-dependent term being the novel component and the $\mathcal{O}(1/N)$ term following from the stability–generalization analysis of Kang and Jang (2025). In this sense our result is complementary to, rather than in competition with, the statistical guarantees of Antos et al. (2008): it supplies the optimization guarantee that their analysis assumes and that SBEED left open.

This conversion also speaks to a line of criticism directed at the Bellman error itself. Fujimoto et al. (2022) document that, in deep RL, a small Bellman error is empirically a poor predictor of a small value error, and argue on that basis against treating the Bellman residual as a surrogate objective. Our result does not contradict that finding, which concerns the general case with no assumption on the function class or on data coverage; rather, it isolates conditions under which the conversion does hold. Lemma A.5 shows that once the returned Q-function lies in a local target-support neighborhood of Q^*, the soft Bellman fixed-point structure yields a metric-subregularity

inequality bounding $\|Q - Q^*\|^2_{2,\pi^*}$ by a constant multiple of the residual risk, with the constant degrading as $\gamma \to 1$ and as the entropy temperature $\lambda \to 0$. Realizability (Assumption 5.1) and single-policy concentrability (Assumption 5.2) are what place the iterate inside that neighborhood. Read together, the two results delimit rather than contradict each other: the residual is an unreliable proxy in general, and a controlled one under the entropy-regularized, realizable, covered regime we analyze.

Empirical evidence for the optimization term. Section 6.3 runs the three BRM methods on a deterministic 8×8 FrozenLake MDP with a tabular parameterization, where Q^*_λ is computed exactly by soft value iteration, so $\varepsilon_{\text{approx}}$ vanishes exactly. The offline dataset holds $544{,}723$ transitions covering 211 of the 256 state-action pairs and is shared by all three methods, so $\varepsilon_{\text{stat}}(N)$ is common. All three consume it under an identical budget of $180{,}000$ gradient updates, namely $30{,}000$ primal steps with 5 dual steps each, which for BRM-FPI is organized as $K = 10$ outer rounds of $3{,}000$ iterations. Because the transitions are deterministic, the dual correction removes no bias and $\mathcal{R}_b(Q^*) = 0$ exactly, leaving the formulation and solution of the minimax problem as the only difference between the methods.[2]

Table 8: Final values on the tabular MDP of Figure 2. Mean $\pm$ std over 5 seeds. Soft value iteration is the exact solution and serves as a reference rather than a competing method. †End-of-round value. The residual of BRM-FPI is a sawtooth whose troughs range from 3.4×10^{-3} (iteration 13k) to 7.2×10^{-3} (final round).

Method	$\|Q - Q^*\|^2_{2,\pi^*}$	$\mathcal{R}_b(Q)$	Discounted return
Soft VI (exact)	0	0	0.8775
`Off-GLADIUS`	$\mathbf{0.00486 \pm 0.00052}$	$\mathbf{(2.78 \pm 0.09) \times 10^{-6}}$	0.8775
SBEED	0.53249 ± 0.00034	$(1.83 \pm 0.005) \times 10^{-3}$	0.8775
BRM-FPI	0.39855 ± 0.00269	$(1.04 \pm 0.20) \times 10^{-2\,\dagger}$	0.0000

Table 8 reports the final values, where `Off-GLADIUS` reaches a Bellman residual $660\times$ below that of SBEED. Since the other two error terms are held fixed by construction, this gap reflects how each method formulates and solves the minimax problem, not approximation or statistical error. Three entries deserve comment. The return of BRM-FPI is exactly zero because it never reaches the goal state, while `Off-GLADIUS` and SBEED share the same return because the MDP is small enough that the greedy ordering is recovered well before Q itself is, so the return column carries little information here. Finally, the residual of BRM-FPI exceeds that of SBEED although its value error is smaller. $\mathcal{R}_b$ is the soft optimality residual, which BRM-FPI does not minimize directly. Each outer round minimizes the evaluation residual of the frozen policy π_k, so $\mathcal{R}_b$ rises within a round as Q approaches Q^{π_k} and drops when π_k is refreshed, and the tabulated value is an end-of-round peak (Figure 2, right).

Figure 4 plots value error against Bellman residual along the same trajectories. For the two methods that minimize $\mathcal{R}_b$ directly, the two quantities move together over three orders of magnitude, and SBEED halts partway along the curve `Off-GLADIUS` traces to its lower end, with a value error within 17% of ours at equal residual. BRM-FPI sits below this curve, since its residual is an end-of-round peak rather than a quantity it drives down. What separates `Off-GLADIUS` from SBEED is therefore how far along this relation each optimizer gets, not the relation itself.

F Further Discussions

Naturality of the Finite Action Space Assumption. In our entropy-regularized Bellman Residual Minimization (BRM) framework, assuming a finite action space $\mathcal{A}$ is mathematically essential for establishing global convergence. Specifically, the log-sum-exp operator in soft Bellman equations

[2]The value of λ cannot be set freely here. The soft optimum collects an entropy bonus of order $\lambda \log |\mathcal{A}|/(1-\gamma)$, which exceeds the optimal discounted return of 0.877 once $\lambda \gtrsim 6 \times 10^{-3}$, beyond which the soft-optimal policy avoids termination and the task changes. BRM-FPI has no such parameter because its target is a hard maximum, but this does not disadvantage it here, as $\|Q^*_{\text{hard}} - Q^*_\lambda\|^2_{2,\pi^*} = 4 \times 10^{-6}$ on this MDP, three orders of magnitude below the smallest of the gaps in Table 8.

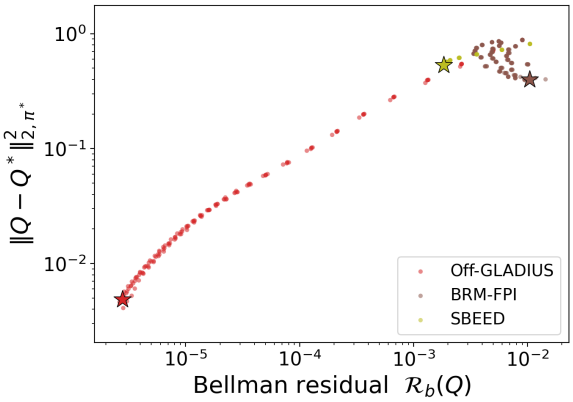


Figure 4: Value error against Bellman residual along the training trajectories of Figure 2, over 5 seeds. Stars mark the final iterate of each method.

(Equation (1), Equation (2) and Equation (3)) is exactly computable only under a finite action space. Extending this to continuous spaces necessitates sampling-based approximations that would break the strict, pointwise Bellman consistency `Off-GLADIUS` enforces. Finally, this exact computation is a strict prerequisite for establishing the empirical Jacobian conditioning (Lemma A.10) and the resulting Polyak-Łojasiewicz (PL) condition, which together drive our population residual-risk guarantee and the Bellman-subregular Q^* conversion in Theorem 5.4.

Empirical Validation on Discrete Control Benchmarks. We evaluated `Off-GLADIUS` on classic discrete control tasks (CartPole-v1, Acrobot-v1, and LunarLander-v3) to strictly align our empirical setup with our theoretical framework. This selection ensures structural consistency with our theory, which relies on exact computation of the finite-action log-sum-exp operator. Furthermore, evaluating on these specific environments is standard practice in the literature for validating the theoretical properties of both soft Bellman operators and minimax-based offline RL (Asadi and Littman, 2017; Uehara et al., 2020; Garg et al., 2021; Hui et al., 2023; Li et al., 2023).

Behavior under Coverage Failure. Table 2 shows that, as the LunarLander-v3 data degrades from uniform-random to pure anti-expert, CQL loses roughly 320 return while `Off-GLADIUS` moves by under 50. We record here a mechanistic account of that asymmetry, with the caveat stated in Section 6.2. It is a hypothesis about the objectives being optimized, not a consequence of Theorem 5.4, which is vacuous in this regime.

The BRM estimand does not depend on the behavior policy. The population risk $\mathcal{R}_b(Q) = \mathbb{E}_{d^{\pi_b}}[b_Q(s,a)^2]$ is a weighted average of a pointwise fixed-point condition whose solution is π_b-independent: $b_{Q^*} \equiv 0$, and hence $\mathcal{R}_b(Q^*) = 0 = \min_Q \mathcal{R}_b(Q)$ for every behavior policy with full support, whatever its return. The behavior distribution enters only as a weighting: it determines *where* the residual is driven to zero first, but never what minimizes the objective. Its effect is confined to the constants and rates of Theorem 5.4, through C_Π. Degrading π_b removes information from the problem; it does not move the target.

Behavior-anchored penalties move the target with π_b. The conservatism term of CQL,

$$\mathcal{C}(Q) := \mathbb{E}_{(s,a)\sim d^{\pi_b}}\left[\log \sum_{a'} \exp Q(s,a') - Q(s,a)\right],$$

does not share this property: in the tabular case

$$\frac{\partial \mathcal{C}(Q)}{\partial Q(s,a)} = d_S^{\pi_b}(s)\big(\pi_Q(a \mid s) - \pi_b(a \mid s)\big),$$

which vanishes exactly when $\pi_Q = \pi_b$ on the data support. The penalty is an anchor at the behavior policy, and hyperparameter α_{CQL} interpolates between the TD fixed point and behavior cloning of π_b; Q^* is stationary for the CQL objective only if $\alpha_{\mathrm{CQL}} = 0$ or $\pi_{Q^*} = \pi_b$. The induced bias therefore

points *at* π_b, with a magnitude fixed by a coefficient chosen without reference to the quality of π_b. On the mixed dataset of Section 6.1 this is benign, since π_b is a reasonable policy; on the anti-expert dataset the same anchor points at a policy of return -760.1 (Table 3), and the observed -440.8 falls between the anti-expert and the uniform-random reference. The χ^2 regularizer of OptiDICE has the same structural form, penalizing deviation of $w = d^\pi / d^{\pi_b}$ from $w \equiv 1$, that is, anchoring at d^{π_b} itself.

The two failure modes differ in kind. All three `Off-GLADIUS` entries in Table 2 (-91.4, -130.3, -141.1) sit at or above the -182.0 return of the uniform random policy, and far above the anti-expert's -760.1. Where the logged data does not constrain the residual, the returned Q is simply unconstrained there and the extracted softmax policy is correspondingly uninformed; nothing in the objective rewards agreement with π_b, so it is not pulled toward the anti-optimal action. Bad data costs `Off-GLADIUS` the ability to learn, rather than teaching it the wrong thing. A behavior-anchored objective, by contrast, converts adversarial data into an adversarial policy, with regret scaling in how bad π_b is.

Scope of these results. Three caveats limit what Table 2 shows. First, these runs lie outside the reach of our theory: C_Π remains finite because the anti-expert assigns positive probability to every action, but it is large enough that Theorem 5.4 is vacuous at any realistic N. Table 2 is therefore not evidence that Assumption 5.2 can be dropped, and our account is a hypothesis rather than a consequence of our analysis. Second, the robustness is relative rather than absolute, since every return in Table 2 is negative and far below the solved threshold of 200 (Figure 1). What distinguishes `Off-GLADIUS` is where it fails: all three of its returns stay above the -182.0 of a uniform-random policy (Table 3), whereas CQL falls below that reference on both datasets containing anti-expert data. Third, OptiDICE does not fit this account. Its return is non-monotone in dataset quality, peaking on the 5:5 mixture (-140.5) rather than on uniform-random data (-237.9), and its seed deviation reaches ± 172.3 on the anti-expert dataset. This pattern points to optimization and policy-extraction instability rather than to the data, so Table 2 supports no conclusion about marginalized importance sampling methods in general. Our observations suggest that, for OptiDICE, a second factor interacts with or outweighs the anchoring effect. We leave these to future work.