Globally Convergent Offline Reinforcement Learning with Smoothed Bellman Residual Minimization

Byungjun Park
Department of AI
Chung Ang University
joshua106@cau.ac.kr

Minhyeok Park
Department of AI
Chung Ang University
nyissha07@cau.ac.kr

Enoch Hyunwook Kang
Foster School of Business
University of Washington
ehwkang@uw.edu

Kyoungseok Jang
Department of AI
Chung Ang University
ksjang@cau.ac.kr

Abstract

Bellman Residual Minimization (BRM) is a scalable gradient-based approach for offline reinforcement learning. While BRM is conceptually appealing in that it directly enforces Bellman consistency and admits stable optimization under neural network parametrization, existing BRM methods have not gained widespread adoption, largely because no prior work has established global convergence guarantees. In this work, we propose `Off-GLADIUS`, an offline reinforcement learning algorithm that provably converges to the global optimum. Our theoretical analysis builds on a reinterpretation of the recent loss-landscape results of Kang et al. [2025], which show that the BRM objective satisfies a Polyak–Łojasiewicz (PL) condition, thereby implying global optimality and linear convergence under gradient-based optimization. Empirically, in proof-of-concept experiments, our algorithm compares favorably with the prominent baselines Conservative Q-Learning (CQL)[Kumar et al., 2020] and OptiDICE[Lee et al., 2021], and outperforms the behavior policy on standard control benchmarks.

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, we often employ *offline reinforcement learning (Offline RL)* [Levine et al., 2020, Prudencio et al., 2023], where learning is performed using a fixed, pre-collected dataset without further interaction with the environment.

In many domains where offline RL is most compelling, e.g., autonomous driving with many raw sensory streams or language-based decision-making with rich textual contexts, the state space is often large, continuous, and high-dimensional. As tabular assumptions break down in such settings, effective learning on practical problems often hinges on function approximation and representation learning (e.g., using neural networks) to generalize beyond a small, finite state space setup [Levine et al., 2020].

Accepted at ICML 2026 Workshop on Decision-Making from Offline Datasets to Online Adaptation: Black-Box Optimization to Reinforcement Learning.

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]. Among these various approaches, 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 inherently circumvents the "deadly triad" of instability that frequently plagues traditional DP-based methods [Tsitsiklis and Van Roy, 1996, 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, 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. This context leads us to a fundamental question:

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

In this paper, we answer this question affirmatively by proposing a novel BRM algorithm, `Off-GLADIUS`, and establishing its global optimality convergence. Specifically, our main contributions are as follows:

- We propose `Off-GLADIUS`, a novel BRM algorithm for offline reinforcement learning. Its key distinction lies in adapting smoothed Bellman Residual Minimization with a primal-dual optimization-based debiasing [Dai et al., 2018, Kang et al., 2025] to the offline reinforcement learning setting, and in interpreting it through a Q-function-centric perspective.
- We prove that `Off-GLADIUS` is a BRM algorithm with global optimality convergence, which has been an open problem in offline reinforcement learning [Jiang and Xie, 2025]. Specifically, we establish that the Bellman residual satisfies the Polyak–Łojasiewicz condition for linear and sufficiently wide neural-network function classes, which guarantees global convergence of `Off-GLADIUS` despite the lack of convex–concave structure.
- We further demonstrate the practical effectiveness of `Off-GLADIUS` in offline settings through proof-of-concept empirical simulations. On standard control benchmarks, `Off-GLADIUS` achieves performance comparable to or exceeding that of prominent offline reinforcement learning baselines, including Conservative Q-Learning (CQL) [Kumar et al., 2020] and OptiDICE [Lee et al., 2021].

Section 2 provides an overview of related work. Section 3 introduces the offline reinforcement learning setting. Section 4 formally defines the expected mean-squared Bellman error minimization problem underlying our approach. In Section 5, we present the `Off-GLADIUS` algorithm, which formulates Bellman residual minimization as a primal–dual optimization problem and solves it via an ascent–descent procedure. Section 6 contains our main theoretical results. Section 7 reports experimental results on benchmark tasks. Finally, Section 8 concludes with a discussion of limitations and directions for future work.

2 Related Works

Bellman Residual Minimization (BRM) Bellman Residual Minimization approach frames offline RL from the perspective of supervised learning (SL), where the squared Bellman error loss serves as an analog to the loss function in SL. BRM has been considered the only known offline RL method that can stably enforce Bellman consistency under non-parametric (e.g., neural network) parametrization [Jiang and Xie, 2025]. Unfortunately, it suffers from a statistical challenge known as the double sampling problem [Antos et al., 2008]. To address this challenge, several works introduce a debiasing (dual) correction term [Antos et al., 2008, Dai et al., 2018]. Although this correction resolves the double-sampling issue, it introduces another problem: the debiasing term is a solution to an inner maximization, turning BRM into a nonconvex *minimax* problem. As a result, despite substantial

progress, BRM-style methods have been considered lacking global convergence guarantees in the offline RL setting [Jiang and Xie, 2025]. In this paper, we address this open problem by proposing `Off-GLADIUS`, the first BRM method to provably achieve global convergence.

Projected mean-squared Bellman error (MSPBE) An alternative to directly minimizing the Bellman residual is to *project* the residual back into the approximation space before squaring it. This method is called the *projected mean-squared Bellman error* (MSPBE) minimization [Patterson et al., 2022]. In BRM, we minimize the mean-squared Bellman error (MSBE), which measures the full size of this Bellman inconsistency. In contrast, MSPBE replaces the Bellman inconsistency by its projection onto the function class. Intuitively, MSPBE asks only that the Bellman residual be small in directions the function class can represent, which can make the objective easier to optimize. However, MSPBE minimization only minimizes the component of the Bellman equation's residual that lies in the function class (the projected part) and ignores the component that is orthogonal to the approximation space.

Fitted Q-Iteration (FQI) Ernst et al. [2005] introduced Fitted Q-Iteration (FQI) as a framework for offline reinforcement learning. Despite its simplicity, iterative value-based methods are well known to be unstable. FQI can diverge even under favorable conditions, such as infinite data and exact regression with a linear function class. The root cause is that FQI repeatedly solves regression problems with targets that change across iterations. Because the temporal-difference targets depend on the current value estimate, the learning problem becomes inherently unstable. More generally, the interaction of function approximation, bootstrapping, and off-policy data leads to what is commonly referred to as the deadly triad [Tsitsiklis and Van Roy, 1996, Van Hasselt et al., 2018, Sutton and Barto, 2018].

Implicit Q-Learning(IQL) Kostrikov et al. [2022] introduced Implicit Q-Learning (IQL), which avoids evaluating out-of-distribution actions entirely by formulating state-value estimation as expectile regression. While IQL successfully mitigates extrapolation error without requiring explicit pessimistic regularization in the action space, its optimization framework fundamentally differs from Bellman Residual Minimization (BRM). Because IQL relies on standard temporal difference (TD) learning coupled with expectiles rather than minimizing the exact squared Bellman error, it does not enforce strict point-wise Bellman consistency. Consequently, it lacks the global convergence guarantees achieved by `Off-GLADIUS`.

Marginalized Importance Sampling (MIS) Marginalized importance sampling (MIS) methods formulate offline reinforcement learning as a minimax optimization problem, where marginalized density ratios are learned as discriminators to minimize the worst-case reweighted Bellman error [Nachum et al., 2019, Zhang et al., 2020, Liu et al., 2018]. Unlike Bellman residual minimization, these approaches optimize linear rather than squared Bellman errors, thereby avoiding the double-sampling issue. However, because MIS-based methods minimize signed average Bellman errors, cancellation between positive and negative residuals may occur. This can sometimes lead to overly optimistic estimates, presenting a limitation in enforcing strict point-wise Bellman consistency [Nachum et al., 2019, Zhang et al., 2020].

3 Problem Settings

3.1 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 immediate reward. Rather than assuming stochastic perturbations on the rewards, we directly 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:

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

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

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

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.

3.2 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), 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

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

[1]This equivalence is frequently noted in the literature on Inverse Reinforcement Learning [Ermon et al., 2015, Zeng et al., 2025] 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.

4 Bellman Residual Minimization (BRM)

In this section, we describe Bellman Residual Minimization (BRM), but in the format that motivates the Algorithm 1 we will describe in the Section 5. Throughout, given a function Q, we define

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

and

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

For a space of Q function $\mathcal{Q} := \{Q : \mathcal{S}\times\mathcal{A} \to \mathbb{R} \mid \|Q\|_\infty < \infty\}$, we define *Bellman operator* and *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 Bellman equation for Q function (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 *point-wise 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

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

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*

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

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 4.1 [Kang et al., 2025].

Lemma 4.1 (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

$$\overline{\mathcal{L}_{\text{BE}}}(Q,\pi) = \min_{Q\in\mathcal{Q}} \max_{\xi\in\mathbb{R}^{S\times A}} \mathbb{E}_\pi\left[\mathbb{E}_{s'\sim P(s,a)}\left[\mathcal{L}_{\text{TD}}(Q)(s,a,s') - \gamma^2(V^Q(s') - \xi(s,a))^2\right]\right] \tag{4}$$

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

$$\frac{1}{N}\sum_{(s,a,s')\in\mathcal{D}_N^b}\left[\mathcal{L}_{\text{TD}}(Q)(s,a,s') - \gamma^2\left(V^Q(s') - \bar{\xi}(s,a)\right)^2\right] \tag{5}$$

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 minimization of Equation (4) (or minimization of (5), empirically) does not suffer the double sampling issue. However, it faces a new challenge: it is now a minimax optimization problem, which is non-trivial to find the global optimum. In Section 5, we will discuss the algorithm we propose to solve this minimax optimization problem.

Algorithm 1 `Off-GLADIUS`

Require: Offline dataset $\mathcal{D} = \{(s, a, r, s')\}$, horizon T

Ensure: $\widehat{Q}, \widehat{\xi}$

1: Initialize $Q_{\theta_1}, \xi_{\theta_2}$; $t \leftarrow 1$
2: **while** $t \leq T$ **do**
3: Sample mini-batches $B_1, B_2 \subset \mathcal{D}$
4: **Ascent step (update ξ_{θ_2}):**
5: $D_{\theta_2} \leftarrow \frac{1}{|B_2|} \sum_{(s,a,s') \in B_2} \big(V_{\theta_1}(s') - \xi_{\theta_2}(s,a)\big)^2$
6: $\theta_2 \leftarrow \theta_2 - \tau_{2,t} \nabla_{\theta_2} D_{\theta_2}$
7: **Descent step (update Q_{θ_1}):**
8: $\overline{\mathcal{L}_{\mathrm{BE}}} \leftarrow \frac{1}{|B_1|} \sum_{(s,a,s') \in B_1} \Big[\mathcal{L}_{\mathrm{TD}}(Q)(s,a,s') - \gamma^2 \big(V_{\theta_1}(s') - \xi_{\theta_2}(s,a)\big)^2\Big]$
9: $\theta_1 \leftarrow \theta_1 - \tau_{1,t} \nabla_{\theta_1} \overline{\mathcal{L}_{\mathrm{BE}}}$
10: $t \leftarrow t + 1$
11: **end while**
12: **Return** $\widehat{\xi} \leftarrow \xi_{\theta_2}$, $\widehat{Q} \leftarrow Q_{\theta_1}$

5 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 6 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, B_1 and B_2, from the offline dataset $\mathcal{D}$. Each dataset will be used for the optimization of θ_1 and θ_2, respectively.

Line 4-6 corresponds to the ascent step for θ_2, the first optimization step in each batch. In this step, the function Q_{θ_1} is fixed, and $V^{Q_{\theta_1}}$ will also be a fixed function induced by Q_{θ_1}. Since only the second term of the Eq. (4), $\gamma^2(V^Q(s') - \xi(s,a))$, has the parameter ξ, in this ascent step the learner should only care about the loss $D_{\theta_2} := \sum_{(s,a,s') \in B_2} (V^Q(s') - \xi(s,a))^2$. Then, the algorithm updates the weight by gradient ascent.

Line 7-9 is about the descent step for θ_1, where we fix the function ξ_{θ_2} and update θ_1. Here, we have to use the whole terms inside the expectation of Eq. (4) to compute the gradient.

6 Theoretical Analysis

Consider a set of parametrized functions

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

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

Assumption 6.1 (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 6.1 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-parametrized 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 6.2 (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

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

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 3. We also use the target-support essential sup norm

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

Finally, we make a regularity condition of Q and ξ for our analysis, which holds for broad range of function classes such as wide neural networks and linear models (see Lemma A.9 for the proofs).

Assumption 6.3 (Regularity conditions)**.** Let Z_Q and Z_ξ be the set of minibatches in line 3 of Algorithm 1. Q and ξ satisfies the following properties:

(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 exists $\mu_Q > 0$ and μ_ξ respectively, such that, for every $\theta_1 \in B_Q$,

$$J_Q(\theta_1; Z_Q) J_Q(\theta_1; Z_Q)^\top \succeq \mu_Q I_M \qquad J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_G.$$

The main theorem below converts the population Bellman-residual excess-risk guarantee into a Q^*-estimation guarantee.

Theorem 6.4 (Bellman-subregular Q^*-estimation bound, informal)**.** *Suppose Assumptions 6.1, 6.2, and 6.3 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 $c_1 > 0, c_2 > 1$. 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 α and the constants hidden in the $O(\cdot)$ notation are problem-dependent constants, independent of N and T.

Sketch of Proof.

(I) Population Bellman-residual excess risk. Using the bi-conjugate reformulation (Lemma 4.1), the empirical BRM problem is a smooth minimax objective in the primal Q_θ and auxiliary ξ variables. The empirical landscape satisfies the two-sided-PL structure required for SGDA, and

the stability/generalization result for BRM converts empirical optimization into the population residual-risk bound

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

where $\mathcal{R}_b^\star := \inf_{Q\in\mathcal{Q}} \mathcal{R}_b(Q)$, and $\Delta_{N,T}$ is a risk upper bound term defined in Lemma A.1. Under Bellman realizability, $\mathcal{R}_b^\star = 0$.

(II) Exact identification and entrance into a local neighborhood. If $\mathcal{R}_b(Q) = 0$, Assumption 6.2 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 uniform 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

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

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

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

for a finite constant C_{Bell} independent of N and T. Absorbing this constant into the big-O terms proves Theorem 6.4.

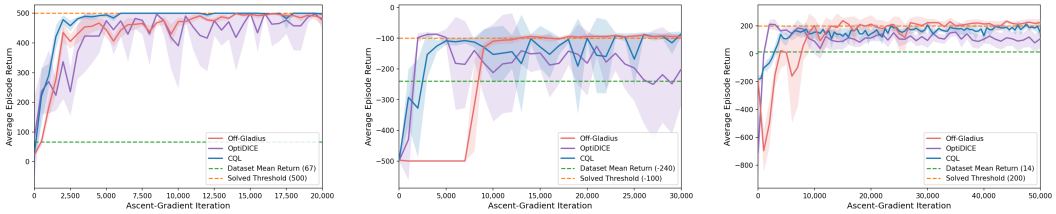
7 Experiments

Experimental Setup The goal of our experiments is to provide proof-of-concept experimentation results that validate the empirical efficiency and stability of `Off-GLADIUS` in offline reinforcement learning settings. We aim to demonstrate that `Off-GLADIUS` can achieve competitive performance purely via gradient-based Bellman residual optimization, consistent with our population residual-risk and Bellman-subregular Q^*-estimation guarantees.

Environments & Datasets We evaluated our algorithm on classic control tasks from the OpenAI Gymnasium benchmark [Brockman, 2016], which is distributed under the MIT License: CartPole-v1, Acrobot-v1, and LunarLander-v3. Since these are standard online environments, we constructed offline datasets to simulate the offline RL setting. For each environment, we first trained an on-policy agent using Proximal Policy Optimization (PPO) [Schulman et al., 2017] to obtain a behavioral policy π_{ppo} that achieves reasonable but sub-optimal performance. To ensure sufficient coverage of the state-action space, we constructed a mixed dataset by aggregating transitions from two distinct sources. Specifically, we collected 700,000 transitions by rolling out the sub-optimal policy π_{ppo}, and an additional 300,000 transitions using a uniform random policy. This process yielded a combined dataset of 1,000,000 transitions characterized by "medium" level expertise with diverse exploratory trajectories, suitable for evaluating the stability and convergence of offline RL algorithms. This mixed-dataset approach aligns with standard offline RL benchmarks like D4RL [Fu et al., 2020].

Baselines Each graph in Figure 1 presents the results of training the following three models on OpenAI Gym environments (CartPole-v1, Acrobot-v1, and LunarLander-v3). Experimental Details can be found in Appendix B.

- **Off-GLADIUS**: A direct implementation of Algorithm 1 that strictly adheres to the gradient-based Bellman residual minimization objective. This variant updates both the Q-function and the dual variable ξ simultaneously via gradient descent/ascent, without incorporating additional techniques such as target networks.



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

Figure 1: Performance comparison of the proposed method across three control tasks. Dataset Mean Return denotes the average episode reward of the dataset, and Solved Threshold refers to the recommended (or maximum) achievable reward for each control task.

- **Conservative Q-Learning (CQL)**: A prominent value-based offline RL algorithm proposed by Kumar et al. [2020]. CQL mitigates extrapolation error by learning a lower-bound conservative Q-estimate for out-of-distribution actions. While it is empirically highly effective and widely adopted as a standard baseline, it does not directly optimize the exact squared Bellman residual, nor does it provide the population Bellman-residual and Bellman-subregular Q^* guarantees established for `Off-GLADIUS`.

- **OptiDICE**: A marginalized importance sampling (MIS) method introduced by Lee et al. [2021]. OptiDICE addresses offline policy optimization by estimating stationary distribution corrections, successfully avoiding the double-sampling issue. However, unlike `Off-GLADIUS`, it optimizes a linear average Bellman error rather than the strict squared Bellman residual, and thus does not strictly enforce point-wise Bellman consistency.

As observed in Figure 1, `Off-GLADIUS` consistently outperforms the Dataset Mean Return baseline across all environments. Furthermore, when compared with other offline RL algorithms, our method demonstrates highly competitive and often superior asymptotic performance. While CQL sometimes exhibits faster initial learning, `Off-GLADIUS` reliably converges to the optimal solved thresholds with remarkable stability. Additionally, it consistently yields higher returns and significantly lower variance than OptiDICE, which struggles with severe instability across these benchmark tasks.

Ultimately, this suggests that achieving rigorous point-wise Bellman consistency does not require sacrificing empirical effectiveness. Instead, our formulation demonstrates that smoothed Bellman residual minimization is a theoretically grounded and practically effective framework for offline control.

8 Conclusion

This paper addressed a fundamental gap in the theory of offline reinforcement learning: whether smoothed Bellman error minimization can achieve global convergence under off-policy data and yield a meaningful Q-estimation guarantee. By proposing the novel algorithm `Off-GLADIUS`, we established global convergence in population Bellman-residual excess risk and converted this residual guarantee into an explicit Q^*-estimation bound through a Bellman-subregularity argument derived from the soft Bellman fixed-point structure. Empirically, we demonstrated that our method achieves performance comparable to prominent baselines such as CQL and OptiDICE, while offering a theoretically grounded alternative to conventional value-based and MIS-based methods. These results bridge a key theoretical divide in the field and open the door for further exploration of stable and optimal offline RL methods grounded in smoothed Bellman error.

As a future research direction, it would be valuable to examine whether model-based BRM approaches also converge to the global optimum, which remains an open and important question. In addition, as noted in Jiang and Xie [2025], empirical research on BRM-type algorithms such as `Off-GLADIUS` may also serve as a foundation for designing more reliable version-space-based pessimistic algorithms, such as PSPI [Cheng et al., 2022], making this an interesting direction for future work.

Acknowledgments and Disclosure of Funding

K. Jang and B. Park were supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [RS-2021-II211341, Artificial Intelligence Graduate School Program (Chung-Ang University)] and by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT).

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.

G Brockman. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

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.

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.

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.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=68n2s9ZJWF8`.

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, Zhaoran 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*, 2019.

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.

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.

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

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

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, 2025.

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.

Ziyu Zhang, Yao Liu, Bo Dai, and Lihong Li. Gendice: Generalized offline estimation of stationary values. *International Conference on Learning Representations*, 2020.

A Technical Proofs

A.1 Proof of Theorem 6.4.

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.9; 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

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

Kang and Jang [2025] proved that under the Assumptions 6.1, 6.2, and 6.3, we have the following properties:

- Empirical minimax objective (equation inside the minimax of Equation 4) is L-smooth with respect to parameter θ_1 and ρ-strong concave with respect to parameter θ_2.
- Per-sample gradient (gradient of terms inside the summation of Equation 5) is uniformly bounded by some constant G over the effective parameter domain.
- Empirical minimization objective (equation inside the minimization of Equation 4) satisfies μ_{PL}-Polyak-Łojasiewicz condition and μ_{QG}-Quadratic Growth condition.

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], let $c := \min\{\mu_{PL}/2, \rho/2\}$ and $\tau_{t,1} = \tau_{t,2} = c_1/(c_2+t)$ for some appropriate constants $c_1 > 0$ and $c_2 > 1$ such that $\tau_{t,1}, \tau_{t,2} \le \min\{\frac{1}{4L}, \frac{1}{\rho}\}$. Then,*

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

where

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

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

Here, C, D_0, and Γ_0 are aggregate coefficients encapsulating the algorithmic stability hitting probabilities, stochastic gradient variance, and initialization bias. Note that if Assumption 6.1 holds, then $\mathcal{R}_b^\star = 0$.

Proof. The biconjugate identity in Lemma 4.1 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)**.** *Let $d_S^{\pi^*}$ be the discounted state occupancy of the entropy-regularized optimal policy. If $B \subseteq \mathcal{S}$ satisfies $d_S^{\pi^*}(B) = 0$, then*

$$P(B \mid s,a) = 0 \qquad d_S^{\pi^*}\text{-a.e. } s, \quad \forall a \in \mathcal{A}.$$

Proof. The discounted state occupancy satisfies the flow identity

$$d_S^{\pi^*}(B) = (1-\gamma)\nu_0(B) + \gamma \int \sum_{a\in\mathcal{A}} \pi^*(a \mid s) P(B \mid s,a)\, d_S^{\pi^*}(s).$$

If $d_S^{\pi^*}(B) = 0$, then the nonnegative integral on the right-hand side is zero. The softmax policy satisfies $\pi^*(a \mid s) > 0$ for every $a \in \mathcal{A}$ on the target state support. Therefore each nonnegative integrand $\pi^*(a \mid s)P(B \mid s,a)$ is zero $d_S^{\pi^*}$-almost surely, which implies the claim. □

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

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

then

$$Q = Q^* \qquad d^{\pi^*}\text{-a.s.}$$

Consequently, under Assumption 6.2,

$$\mathcal{R}_b(Q) = 0 \quad \Longrightarrow \quad Q = Q^* \quad d^{\pi^*}\text{-a.s.}$$

Proof. Because $\pi^*(a \mid s) > 0$ for every $a \in \mathcal{A}$ on the target state support, $b_Q = 0$ d^{π^*}-a.s. implies that

$$\mathcal{T}Q(s,a) = Q(s,a) \qquad \forall a \in \mathcal{A}, \quad d_S^{\pi^*}\text{-a.e. } s.$$

Let $e := Q - Q^*$. Since $Q^* = \mathcal{T}Q^*$, for $d_S^{\pi^*}$-a.e. s and every $a \in \mathcal{A}$,

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

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

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

By Lemma A.2, the next state s' remains in the target state support for $d_S^{\pi^*}$-almost every current state and every action. Hence, with

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

we obtain

$$|e(s,a)| \le \gamma M \qquad d_S^{\pi^*}\text{-a.e. } s, \quad \forall a \in \mathcal{A}.$$

Taking the essential supremum gives $M \le \gamma M$. Since $\gamma < 1$, $M = 0$, so $Q = Q^*$ d^{π^*}-a.s.

If $\mathcal{R}_b(Q) = 0$, then $b_Q = 0$ d^{π_b}-a.s. Assumption 6.2 implies that every d^{π_b}-null set is also d^{π^*}-null. Therefore $b_Q = 0$ d^{π^*}-a.s., and the first part of the lemma applies. □

Lemma A.4 (Compact identification modulus)**.** *Under Assumptions 6.1 and 6.2 and the compact bounded parametrization in Lemma A.9,*

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

Proof. Lemma A.9 gives a compact parameter region B_Q and a uniform derivative bound

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

Therefore

$$\|Q_{\theta_1} - Q_{\theta_2}\|_\infty \le B\|\theta_1 - \theta_2\|_2,$$

so $\theta \mapsto Q_\theta$ is continuous in the target-support sup norm. Since the log-sum-exp map is 1-Lipschitz in the sup norm,

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

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

Suppose the conclusion fails. Then there exist $\eta > 0$, $\varepsilon_n \downarrow 0$, and $\theta_n \in B_Q$ such that

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

By compactness of B_Q, a subsequence, not relabeled, satisfies $\theta_n \to \theta_\infty \in B_Q$. Continuity gives

$$\mathcal{R}_b(Q_{\theta_\infty}) = 0.$$

By Lemma A.3, $Q_{\theta_\infty} = Q^*$ d^{π^*}-a.s., and therefore

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

But the uniform Lipschitz bound gives

$$\|Q_{\theta_n} - Q_{\theta_\infty}\|_{\infty,*} \to 0,$$

which contradicts $\|Q_{\theta_n} - Q^*\|_{\infty,*} \geq \eta$. Thus the modulus vanishes at the origin. □

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

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

Then

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

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

Proof. Let

$$e_Q(s,a) := Q(s,a) - Q^*(s,a).$$

For $u \in [0,1]$, define $Q_u := Q^* + u(Q - Q^*)$ and let π_{Q_u} be the softmax policy induced by Q_u. Define the interpolation policy

$$\bar{\pi}_Q(a \mid s) := \int_0^1 \pi_{Q_u}(a \mid s)\, du.$$

By the fundamental theorem of calculus applied to the log-sum-exp value map,

$$V^Q(s) - V^{Q^*}(s) = \sum_{a\in\mathcal{A}} \bar{\pi}_Q(a \mid s) e_Q(s,a).$$

Therefore

$$\mathcal{T}Q - \mathcal{T}Q^* = \gamma P_{\bar{\pi}_Q} e_Q,$$

where

$$(P_{\bar{\pi}_Q} f)(s,a) := \mathbb{E}_{s'\sim P(\cdot|s,a),\, a'\sim\bar{\pi}_Q(\cdot|s')}[f(s',a')].$$

Since $Q^* = \mathcal{T}Q^*$,

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

We next show that $\gamma P_{\bar{\pi}_Q}$ is a contraction in $L_2(d^{\pi^*})$ in the local neighborhood. For $d_S^{\pi^*}$-a.e. s and every $u \in [0,1]$, the bound $\|Q - Q^*\|_{\infty,*} \leq \delta$ implies

$$\frac{\pi_{Q_u}(a \mid s)}{\pi^*(a \mid s)} \leq e^{2\delta}, \qquad \forall a \in \mathcal{A}.$$

Indeed, each numerator term changes by at most e^δ and the softmax denominator changes by at least $e^{-\delta}$. Averaging over u gives

$$\bar{\pi}_Q(a \mid s) \leq e^{2\delta}\pi^*(a \mid s) \qquad d_S^{\pi^*}\text{-a.e. } s, \quad \forall a.$$

By Lemma A.2, this comparison is valid at the next states reached from the target support.

For every square-integrable f, Jensen's inequality gives

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

The discounted occupancy flow identity is

$$d^{\pi^*} = (1-\gamma)d_0^{\pi^*} + \gamma P_{\pi^*}^\top d^{\pi^*},$$

so $\gamma P_{\pi^*}^\top d^{\pi^*} \le d^{\pi^*}$. Hence

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

Combining the last two displays yields

$$\|\gamma P_{\bar{\pi}_Q} f\|_{2,\pi^*} \le \sqrt{\gamma e^{2\delta}}\,\|f\|_{2,\pi^*}.$$

Since $\delta < (1/2)\log(1/\gamma)$, the factor $\sqrt{\gamma e^{2\delta}}$ is strictly smaller than one. Applying this contraction bound to $e_Q = -b_Q + \gamma P_{\bar{\pi}_Q} e_Q$ gives

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

Thus

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

Finally, Assumption 6.2 gives

$$\|b_Q\|_{2,\pi^*}^2 \le C_\Pi \|b_Q\|_{2,\pi_b}^2 = C_\Pi \mathcal{R}_b(Q).$$

Squaring 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 6.2; 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 6.2 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 6.4 *Suppose Assumptions 6.1, 6.2, and 6.3 hold. Let $Q_{\hat{\theta}_T}$ denote the Q-function returned after T iterations of Algorithm 1. Then,*

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

where the hidden constant is finite and independent of N and T.

Proof. 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

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

By Markov's inequality,

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

Thus

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

Define

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

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 6.1,

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

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.7 (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.8 (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.9 (Compact empirical Jacobian regularity for linear and sufficiently wide neural classes)**.** *Let $Z_Q = \{z_i\}_{i=1}^M$ and $Z_\xi = \{\bar{z}_i\}_{i=1}^G$ be finite empirical evaluation sets generated by $\mathcal{D}_N^b$, and let*

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

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 6.1, 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*

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

where $n_{\text{eval}} := |Z_Q| + |Z_\xi|$, $R_\star := \max\{R_Q, R_\xi, 1\}$, and

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

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*

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

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 $\theta_1 \in B_Q$,*

$$J_Q(\theta_1; Z_Q) J_Q(\theta_1; Z_Q)^\top \succeq \mu_Q I_M.$$

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

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

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

$$J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_G.$$

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

Linear parameterizations. Let

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

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

$$\Phi_Q(Z_Q)\Phi_Q(Z_Q)^\top \succeq \mu_Q I_M, \qquad \Phi_\xi(Z_\xi)\Phi_\xi(Z_\xi)^\top \succeq \mu_\xi I_G$$

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, θ_1, M, R_Q) by $(Z_\xi, \theta_2, G, R_\xi)$. 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

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

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 $\theta_1 \in B_Q$,

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

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

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

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

$$J_\xi(\theta_2; Z_\xi) J_\xi(\theta_2; Z_\xi)^\top \succeq \mu_\xi I_G \qquad \forall \theta_2 \in B_\xi,$$

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 Experimental Details

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

Computational Resources All experiments were conducted on a GPU server equipped with 4× NVIDIA Tesla V100 (DGX) GPUs, each with 32 GB memory (Tesla V100-DGXS-32GB). The CUDA version reported by `nvidia-smi` was 12.8 (driver 570.133.20). We implemented all methods in Python using PyTorch.

B.1 Shared Experimental Setup

Evaluation Although training is performed purely offline, we periodically evaluate the learned policy online in the corresponding environment. We evaluate every 1,000 ascent–descent iterations and report the average episodic return. All results are averaged over 5 random seeds (we report the mean $\pm$ standard deviation across seeds).

Network Architecture and Initialization For all algorithms, we parameterized the value functions and policies using a 3-layer multilayer perceptron (MLP) with 128 hidden nodes per layer. 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 (ascent–descent steps) was tailored to the complexity of each environment: 20k iterations for CartPole-v1, 30k for Acrobot-v1, and 50k for LunarLander-v3. The batch size for each environment was selected via a grid search over the set $\{256, 512, 1024\}$.

B.2 Off-GLADIUS

Ascent-Descent Iteration For `Off-GLADIUS`, we report learning curves with the x-axis measured in *ascent–descent iteration*. We define one ascent–descent iteration as a complete update cycle consisting of (i) `xi_steps` consecutive updates of the auxiliary network ξ (the ascent step) followed by (ii) a single update of the Q network (the descent step). To ensure sufficient maximization during the ascent phase before moving to the descent step, we perform multiple ascent updates per cycle. Specifically, we set `xi_steps` $= 5$ for both CartPole-v1 and Acrobot-v1, and `xi_steps` $= 10$ for LunarLander-v3. Note that in our implementation, τ_2, τ_1, and λ correspond to `lr_xi`, `lr_q`, and `lmbda`, respectively. To identify the optimal configuration, we tuned the following hyperparameters via grid search:

- Temperature (`lmbda`): $\{0.01, 0.1, 1.0\}$
- Ascent step learning rate (`lr_xi`): $\{10^{-4}, 3 \times 10^{-4}, 3 \times 10^{-3}\}$
- Descent step learning rate (`lr_q`): $\{10^{-5}, 3 \times 10^{-5}, 3 \times 10^{-4}\}$

Off-GLADIUS Hyperparameters Through our evaluation, a batch size of 1024 consistently yielded the best performance across all three environments. For the descent step learning rate, `lr_q` $= 3 \times 10^{-4}$ was universally optimal. The optimal ascent step learning rate was found to be `lr_xi` $= 3 \times 10^{-4}$ for Acrobot-v1, while a larger learning rate of `lr_xi` $= 3 \times 10^{-3}$ was preferred for both CartPole-v1 and LunarLander-v3. Finally, the optimal temperature parameter was `lmbda` $= 1.0$ for CartPole-v1, whereas `lmbda` $= 0.1$ achieved the best results in the remaining two environments.

B.3 Conservative Q-Learning (CQL)

CQL Conservatism Parameter (α_{CQL}) In CQL, the α_{CQL} parameter controls the degree of conservatism by weighting the penalty for out-of-distribution (OOD) actions. A higher α_{CQL} mitigates overestimation bias but risks overly pessimistic policies, while a lower α_{CQL} may lead to severe value

extrapolation errors. Thus, tuning α_{CQL} is essential to balance robustness and optimization. We searched over the following configurations:

- Conservatism parameter (α_{CQL}): $\{0.01, 0.1, 1.0, 5.0\}$
- Learning rate: $\{10^{-4}, 3 \times 10^{-4}, 10^{-3}\}$

CQL Hyperparameters Based on the search spaces listed above, we performed a grid search to identify the optimal configuration for each environment. We found that a batch size of 1024 consistently yielded the best performance across all three environments. For the learning rate, the optimal value was 1×10^{-3} for LunarLander-v3, and 3×10^{-4} for both CartPole-v1 and Acrobot-v1. Regarding the conservatism parameter, a higher penalty of $\alpha_{\text{CQL}} = 5.0$ yielded the best performance in CartPole-v1, whereas a lower value of $\alpha_{\text{CQL}} = 1.0$ was optimal for the remaining two environments.

B.4 OptiDICE

OptiDICE Regularization Parameter (α_{OptiDICE}) and Learning Rates While CQL enforces conservatism directly in the value space, OptiDICE achieves it in the distribution space. Specifically, OptiDICE casts offline RL as a stationary distribution matching problem, where the α_{OptiDICE} parameter controls the strength of the f-divergence (e.g., χ^2-divergence) regularization. This regularizer constrains the learned policy's state-action distribution to stay close to the offline dataset. Tuning α_{OptiDICE} is critical, as it determines the trade-off between maximizing the reward and matching the behavioral data. Furthermore, rather than relying on unstable alternating Stochastic Gradient Descent-Ascent (SGDA) for minimax optimization, OptiDICE derives a closed-form solution for the density ratios. This reduces the main objective to a single convex minimization problem for the value network (ν), while the weight network (w) is jointly trained via gradient descent to match this closed-form target. Consequently, we tuned separate learning rates, denoted as `lr_v` and `lr_w`, respectively. We considered the following search spaces:

- Regularization parameter (α_{OptiDICE}): $\{0.01, 0.1, 1.0\}$
- Weight network learning rate (`lr_w`): $\{10^{-4}, 3 \times 10^{-4}, 3 \times 10^{-3}\}$
- Value network learning rate (`lr_v`): $\{10^{-5}, 3 \times 10^{-5}, 3 \times 10^{-4}\}$

OptiDICE Hyperparameters Based on the search spaces listed above, we performed a grid search to identify the optimal configuration for each environment. Through our evaluation, we found that a batch size of 1024, a regularization parameter of $\alpha_{\text{OptiDICE}} = 1.0$, and an value network learning rate of `lr_v` $= 3 \times 10^{-4}$ consistently yielded the best performance across all three environments. For the weight network learning rate, a higher value of `lr_w` $= 3 \times 10^{-3}$ was optimal for LunarLander-v3, while `lr_w` $= 3 \times 10^{-4}$ was preferred for both CartPole-v1 and Acrobot-v1.

C 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 (Equations 1, 2 and 3) is exactly computable only under a finite action space. Extending this to continuous spaces necessitates sampling-based approximations that would break the strict, point-wise Bellman consistency `Off-GLADIUS` enforces. Finally, this exact computation is a strict prerequisite for establishing the empirical Jacobian conditioning (Lemma A.9) and the resulting Polyak-Łojasiewicz (PL) condition, which together drive our population residual-risk guarantee and the Bellman-subregular Q^* conversion in Theorem 6.4.

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 requires 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].