Monotone and Conservative Policy Iteration Beyond the Tabular Case

S.R. Eshwar[1] **Gugan Thoppe**[1] **Ananyabrata Barua**[1] **Aditya Gopalan**[1] **Gal Dalal**[2]
[1] Indian Institute of Science, Bengaluru, India
[2]NVIDIA Research, Israel

Abstract

We introduce Reliable Policy Iteration (RPI) and Conservative RPI (CRPI), variants of Policy Iteration (PI) and Conservative PI (CPI), that retain tabular guarantees under function approximation. RPI uses a novel Bellman-constrained optimization for policy evaluation. We show that RPI restores the textbook *monotonicity* of value estimates and that these estimates provably *lower-bound* the true return; moreover, their limit partially satisfies the *unprojected* Bellman equation. CRPI shares RPI's evaluation, but updates policies conservatively by maximizing a new performance-difference *lower bound* that explicitly accounts for function-approximation-induced errors. CRPI inherits RPI's guarantees and, crucially, admits per-step improvement bounds. In initial simulations, RPI and CRPI outperform PI and its variants. Our work addresses a foundational gap in RL: popular algorithms such as TRPO and PPO derive from tabular CPI yet are deployed with function approximation, where CPI's guarantees often fail-leading to divergence, oscillations, or convergence to suboptimal policies. By restoring PI/CPI-style guarantees for *arbitrary* function classes, RPI and CRPI provide a principled basis for next-generation RL.

1 INTRODUCTION

Function Approximation (FA) in Reinforcement Learning (RL) creates a fundamental challenge: while necessary for handling large state-action spaces, it also results in issues such as divergence, policy and value oscillations, training instability, and convergence to sub-optimal policies—sometimes even to the worst policy (Aethelios, 2025; Patterson et al., 2024; Baird, 1995; Young and Sutton, 2020; Henderson et al., 2018; Gopalan and Thoppe, 2025). A closer look at these issues reveals a gap between practice and theory: while mainstream RL algorithms are deployed with (possibly high-capacity) FA, the associated guarantees apply only in tabular or near-tabular regimes (Bertsekas and Tsitsiklis, 1996; Kakade and Langford, 2002; Hasselt, 2010; Haarnoja et al., 2018; Metelli et al., 2021).

The roots of this gap stem from the fact that the textbook (model-based) Policy Iteration (PI) method (Howard, 1960)—the foundation of all actor–critic-type approaches—lacks a FA variant that preserves its core guarantees. Recall that PI alternates between policy evaluation and policy update. In tabular settings, the value functions of the successive policies improve *monotonically* and *converge* to the optimum. With FA, however, even in the model-based setting—or with access to infinite data—approximation errors during policy evaluation can corrupt the policy updates, causing divergence, oscillations, and learning instability (Bertsekas, 2011). Therefore, it is not surprising that these pathologies also pervade model-free actor–critic methods wherein we must also contend with the epistemic uncertainty of working with only sampled data (Thrun and Schwartz, 1993; Van Hasselt et al., 2016; Fujimoto et al., 2018). In a similar vein, Conservative PI (CPI) (Kakade and Langford, 2002) and its extension Safe PI (SPI) (Metelli et al., 2021) currently lack FA variants that retain their per-step improvement guarantees. It has been noted that PI, CPI, and SPI differ markedly in how they respond to approximation errors during policy updates: PI often learns quickly but can stall or oscillate, whereas CPI and SPI make more measured updates and, in some domains, reach better policies (Metelli et al., 2021).

Together, these observations raise two central ques-

Proceedings of the 29[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

tions: (i) Can FA variants of PI be built without losing value-estimate monotonicity? (ii) Can FA variants of CPI and SPI be designed while retaining their per-step improvement guarantees? These questions bear directly on widely used value-based and policy-optimization methods, including DQN (Mnih et al., 2015), TRPO (Schulman et al., 2015), and PPO (Schulman et al., 2017), and are therefore central to placing modern RL on firmer theoretical foundations.

We answer both questions in the affirmative. Our work's key contributions are as follows:

1. **Algorithms:** We introduce Reliable Policy Iteration (RPI) and its conservative variant, CRPI, in Algorithms 1 and 2. Both use a novel policy-evaluation rule: compute the value estimate *farthest* from the previous one subject to a linear Bellman-inequality constraint. The two methods differ in the policy-update step: RPI uses the standard greedy update, whereas CRPI uses a conservative one akin to CPI/SPI. Unlike PI, CPI, and SPI, our methods apply under general FA.

2. **RPI Theoretical Guarantees:** We show that RPI extends the classical PI guarantees of per-step improvement and convergence to arbitrary FA (Section 3.2), where such guarantees were previously out of reach. Specifically, we show that RPI's value estimates are non-decreasing, lower bound the true policy values, and converge to a vector that partially satisfies the *unprojected* Bellman equation. We also bound the performance gap between RPI's terminal policy and the optimal one. Finally, we show that RPI generalizes classical PI and that its policy-evaluation step admits a constrained projection interpretation under an ℓ_1-type norm.

3. **CRPI Theoretical Guarantees:** We develop a generalization of the performance-difference lemma to arbitrary FA (Section 3.3) that (i) accommodates FA-based estimates of the true advantage function and (ii) explicitly accounts for approximation errors. Building on this, we show that CRPI's policy update maximizes a lower bound on the performance gap, yielding the first per-step improvement guarantees under FA. We further prove that CRPI's suboptimality converges at $O(1/k)$ rate to an ϵ-neighborhood of its limiting value.

4. **Simulations**: In model-based inventory control with dense rewards, RPI outperforms approximate variants of PI, CPI, and SPI in both learning speed and terminal policy quality. By contrast, in chain walk with sparse rewards, CRPI learns more conservatively but often identifies better terminal policies. In deep RL experiments on MuJoCo robotic tasks, RPI-based methods are either competitive with or outperform standard baselines, and their critic estimates often display near-monotone behavior while frequently lower bounding the Monte Carlo returns.

2 RELATED WORK

The earliest use of FA can be traced to Samuel's checkers program (Samuel, 1959, 1967). It selected moves via multistage lookahead while evaluating positions with a value function expressed as a linear combination of handcrafted features. Since then, many attempts have been made to extend PI to the FA setting, which can be categorized and summarized as follows.

PI with Approximate Evaluation. A large body of work modifies only the policy-evaluation step, while leaving the greedy policy update unchanged. These methods typically follow one of four approaches: (i) minimizing the mean-squared projection error, as in TD(1) (Tsitsiklis and Van Roy, 1996); (ii) solving the projected Bellman fixed-point equation, as in LSPI and TD(λ) (Lagoudakis and Parr, 2003; Tsitsiklis and Van Roy, 1996); (iii) minimizing the Bellman residual error, e.g., (Scherrer, 2010); or (iv) minimizing the Bellman backup error, as in Fitted Q-Iteration and AMPI-Q (Ernst et al., 2005; Mnih et al., 2015; Scherrer et al., 2015). However, to paraphrase Bertsekas (2011), these methods presume that a more accurate value approximation should yield a better policy—a plausible but far from self-evident assumption. In practice, they diverge, oscillate among worse policies, or otherwise behave unreliably (Bertsekas, 2011; Patterson et al., 2024; Gopalan and Thoppe, 2025; Young and Sutton, 2020). A theoretically sound resolution of these issues has remained elusive so far.

PI with Conservative Policy Updates. A complementary line of work makes policy updates cautious, either by constraining the KL divergence between successive policies or by interpolating between the current policy and a greedy one. This idea goes back to conservative PI (Kakade and Langford, 2002) and is further developed in safe PI (Metelli et al., 2021); it is also reflected in influential methods such as TRPO (Schulman et al., 2015) and PPO (Schulman et al., 2017). Such cautious updates can yield strong per-step improvement bounds (Metelli et al., 2021, Remark 3), and empirically they often learn more conservatively yet attain better terminal policies (Metelli et al., 2021, Figures 9, 13). However, their theory relies on accurate value estimates for essentially every state–action pair, an assumption that breaks down under FA.

Policy Iteration with Multi-step Lookahead. In classical PI, the actor jumps to a policy that is greedy with respect to the current policy's value function.

Multistep-lookahead variants instead choose a policy that is greedy over an h-step rollout. While this still ensures monotonic improvement in tabular MDPs (Efroni et al., 2018a), the picture becomes bleak once FA enters. Winnicki et al. (2021) show that with least squares approximation during evaluation, the value estimates can diverge unless h exceeds a problem-dependent threshold. In fact, lookahead faces inherent limitations even in tabular settings: partial policy evaluation may lead to divergence (Efroni et al., 2019), and conservative PI loses its one-step monotonicity as soon as $h > 1$ (Efroni et al., 2018b). These results point to a need to redesign policy evaluation and policy update to get better guarantees.

PI-Adjacent Alternatives. There is also classification-based PI (Lazaric et al., 2010) which treats policy-update as a supervised learning problem, but forfeits per-iteration guarantees. There is also linear-programming-based approaches for dynamic programming with linear FA (De Farias and Van Roy, 2003). Unlike RPI, they are single-shot methods that directly seek a lower bound on the optimal value.

Policy-Optimization Beyond PI. While several recent works study policy-gradient, trust-region, and mirror-descent methods under FA, their guarantees differ fundamentally from those considered here. These methods typically rely on realizability assumptions or uniform bounds on critic approximation error across all intermediate policies. For example, Liu et al. (2019) assume that the function class contains Q_π for every policy π, while Agarwal et al. (2021) and Alfano et al. (2023) assume uniform bounds on critic approximation error across iterations. Consequently, their guarantees bound quantities such as the minimum or average suboptimality over T iterations in terms of the approximation-error constant (e.g., ϵ_{bias} or ϵ_{approx}). When this constant is large, the resulting bounds can be large and therefore do not preclude oscillations or convergence to suboptimal policies.

3 PROBLEM FORMULATION, PROPOSED ALGORITHMS, AND MAIN RESULTS

We formalize our setup and goals in Section 3.1, and then present RPI and CRPI—along with their theoretical guarantees—in Sections 3.2 and 3.3, respectively. Appendix A provides detailed proofs of all our results.

3.1 Setup and Problem Formulation

We have a stationary MDP $\mathcal{M} \equiv (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$. Here, $\mathcal{S}$ and $\mathcal{A}$ are finite state and action spaces, respectively, with $|\mathcal{S}| := S$ and $|\mathcal{A}| := A$. Further, $\mathcal{P}$ is the transition kernel and $\mathcal{P}(s'|s,a)$ specifies the probability of reaching state s' from state s under action a. Finally, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the per-step reward function, and $\gamma \in [0, 1)$ the discount factor.

Algorithm 1 Reliable Policy Iteration (RPI)

Input: FA class $\mathcal{F}$, policy μ_0, an initial approximation $f_0 \in \mathcal{F}$ of Q_{μ_0}, and a norm $\|\cdot\|$

for $k = 0, 1, 2 \ldots$ until convergence **do**

Policy Evaluation:

$$f_{k+1} \in \arg\max_{f \in \mathcal{F}} \quad \|f - f_k\| \quad \text{s.t.} \quad T_{\mu_k} f \geq f \geq f_k \tag{1}$$

Policy Improvement:

$$\mu_{k+1} \in \{\mu : \mu \text{ is a deterministic greedy policy w.r.t. } f_{k+1}\} \tag{2}$$

end for

For any set $\mathcal{U}$, let $\Delta(\mathcal{U})$ be the set of distributions on it. Now, for a stationary policy $\mu : \mathcal{S} \to \Delta(\mathcal{A})$, define its Q-value function $Q_\mu : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ by $Q_\mu(s,a) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s_0 = s, a_0 = a\right]$, where $s_{t+1} \sim \mathcal{P}(\cdot|s_t, a_t)$ and $a_{t+1} \sim \mu(\cdot|s_{t+1})$ for all $t \geq 0$. Further, let $T_\mu : \mathbb{R}^{SA} \to \mathbb{R}^{SA}$ and $T : \mathbb{R}^{SA} \to \mathbb{R}^{SA}$ be the Bellman operators given by

$$T_\mu Q(s,a) = r(s,a) + \gamma \sum_{s',a'} \mathcal{P}(s'|s,a)\mu(a'|s')Q(s',a')$$

and

$$TQ(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s'|s,a) \max_{a'} Q(s',a').$$

Our goal here is twofold: (i) modify tabular PI's policy-evaluation step to restore its monotonicity and convergence guarantees under FA; and (ii) adapt the policy-update step to maximize an FA-based performance-improvement gap, akin to CPI and SPI.

We use $\mathcal{F} = \{f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$ to denote a given FA space for representing the Q-value functions. Clearly, each $f \in \mathcal{F}$ can also be interpreted as a vector in $\mathbb{R}^{SA}$.

3.2 Reliable Policy Iteration (RPI)

RPI's description is given in Algorithm 1. The inequalities there are coordinate wise and we retain this convention for vector inequalities throughout. Like other PI implementations, RPI also interleaves policy evaluations and policy updates, but its novelty lies in the evaluation step. Rather than seeking a close approximation to Q_{μ_k}, or minimizing the projected-Bellman error, or even being conservatively close to f_k, RPI chooses $f \in \mathcal{F}$ that is farthest away from f_k under a given norm $\|\cdot\|$, subject to $T_{\mu_k} f \geq f \geq f_k$. The policy-update step is standard: it replaces the current policy μ_k with the one that is greedy with respect to f_{k+1}, as in classical PI and in existing FA variants such as approximate PI (Bertsekas and Tsitsiklis, 1996) and AMPI-Q (Scherrer et al., 2015).

We now state RPI's performance guarantees under general FA, covering both linear and non-linear settings. For any two vectors $Q, Q' \in \mathbb{R}^{SA}$, we write $Q \geq_{\mathrm{p}} Q'$ to imply that $Q(s,a) \geq Q'(s,a)$ for all (s,a), with equality holding on *at least one coordinate*.

Theorem 3.1 (**RPI properties with general FA**). *Suppose the FA space $\mathcal{F}$ is a closed subset of $\mathbb{R}^{SA}$ and the initial policy and value estimates satisfy $T_{\mu_0} f_0 \geq f_0$. Then, the following claims hold.*

1. *For any $k \geq 0$, f_k satisfies the constraints in (1); hence, a solution to (1) always exists.*
2. *$(f_k)_{k\geq 0}$ is non-decreasing and $Q_{\mu_k} \geq f_k$ $\forall k \geq 0$.*
3. *$f_\infty := \lim_{k\to\infty} f_k$ exists and satisfies $T_{\mu_\infty} f_\infty = T f_\infty \geq f_\infty$, where μ_∞ is any policy that is greedy with respect to f_∞. Furthermore, if $Q^\star$ denotes the optimal Q-value function, then*
$$\|Q_{\mu_\infty} - Q^\star\|_\infty \leq \frac{\|T f_\infty - f_\infty\|_\infty}{1-\gamma} = \frac{\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}.$$
4. *Additionally suppose $\|\cdot\|$ is strictly monotone: $Q \geq Q' \geq 0$ and $Q \neq Q'$ imply $\|Q\| > \|Q'\|$. Also suppose the function class $\mathcal{F}$ has room to improve in the positive orthant centered at f_∞: there is a $\delta_0 > 0$ such that, for every $0 < \delta \leq \delta_0$, the function class $\mathcal{F}$ contains at least one f satisfying $\|f - f_\infty\| < \delta$ and $f > f_\infty$ (coordinate-wise). Then, $T_{\mu_\infty} f_\infty = T f_\infty \geq_{\mathrm{p}} f_\infty$, i.e., f_∞ partially satisfies the Bellman and Bellman optimality equations.*

We next show that RPI mimics PI in the tabular case. Thus, it is a true generalization of PI.

Proposition 3.2 (**RPI generalizes PI**). *Suppose $\mathcal{F} = \mathbb{R}^{SA}$, $T_{\mu_0} f_0 \geq f_0$, and the norm $\|\cdot\|$ is strictly monotone (as defined in Theorem 3.1). Then, $f_{k+1} = Q_{\mu_k}$ $\forall k \geq 0$, $f_\infty = Q^\star$, and μ_∞ is optimal.*

Our next result shows that RPI's evaluation step has a projection interpretation under ℓ_1-type norms.

Proposition 3.3 (**Projection view under ℓ_1-type-norm**). *Suppose $\mathcal{F}$ is closed, $T_{\mu_0} f_0 \geq f_0$, and the norm in (1) is some w-weighted ℓ_1-norm $\|\cdot\|_{w,1}$. That is, let $w \in \mathbb{R}^{SA}$ be made up of strictly positive values and $\|f\|_{w,1} := \sum_{s,a} w(s,a)|f(s,a)|$. Then, $f_{k+1} \in \underset{f\in\mathcal{F}: T_{\mu_k} f \geq f \geq f_k}{\arg\min} \|f - Q_{\mu_k}\|_{w,1}$.*

Discussion: We highlight the following three key implications of the above results.

1. **Monotonic reliability under FA**. Theorem 3.1 shows that (f_k) is coordinate-wise non-decreasing and lower bounds the true Q-values of the corresponding policies. Even if the policy estimate degrades, i.e., Q_{μ_k} decreases, the drop in performance will not go below f_k, which is non-decreasing. RPI is the first FA-variant of PI that comes with a monotonic improvement guarantee for its value estimates. Recall that existing methods like PI (Howard, 1960), CPI (Kakade and Langford, 2002), or SPI (Metelli et al., 2021) require accurate Q-value estimates across all SA-many state-action pairs to provide such guarantees.
2. **Convergence to Bellman-consistent points**. Theorem 3.1 also shows that f_k converges to a limit f_∞ that *partially* satisfies both the optimality Bellman equation and the Bellman equation for μ_∞. While multiple such fixed points may exist—a fundamental limitation of FA—RPI nevertheless remains aligned with the central RL objective of solving the unprojected Bellman equations. In contrast, projection-based schemes typically target projected surrogates instead, and can oscillate between poor policies (Bertsekas, 2011).
3. **Geometric interpretation in weighted-ℓ_1**. Proposition 3.3 shows that, under any weighted-ℓ_1 norm, RPI's evaluation step is equivalent to projecting Q_{μ_k} onto the constrained set $\{f \in \mathcal{F} : T_{\mu_k} f \geq f \geq f_k\}$. Unlike standard full-space projections, this projection incorporates the Bellman inequalities directly into the feasible set. Although this exact geometric interpretation need not extend to arbitrary norms, the coordinate-wise monotonicity guarantees continue to hold for any norm. Thus, our constrained optimization retains the key monotonic structure of tabular PI even in the FA setting.

In summary, by embedding the Bellman inequalities directly into the evaluation program in (1), RPI fully restores the classical PI guarantees.

3.3 Conservative RPI (CRPI)

While vanilla PI uses greedy policy updates, CPI and SPI employ conservative updates instead. Under FA and sampling noise, such CPI/SPI-style updates have empirically been shown to produce better terminal performance (Metelli et al., 2021). In addition, in

Algorithm 2 Conservative RPI (CRPI)

Input: FA class $\mathcal{F}$, initial policy μ_0, an initial approximation $f_0 \in \mathcal{F}$ of Q_{μ_0}, distribution ν for sampling the initial (s, a)-pair, and a norm $\|\cdot\|$

for $k = 0, 1, 2$ ... until convergence **do**

Policy Evaluation:

$$\begin{aligned} f_{k+1} \in \arg\max_{f \in \mathcal{F}} \quad & \|f - f_k\| \\ \text{s.t.} \quad & T_{\mu_k} f \geq f \geq f_k \end{aligned} \tag{3}$$

Policy Improvement:

$$\begin{aligned} \bar{\mu}_k &\in \{\mu : \mu \text{ is a greedy policy w.r.t. } f_{k+1}\} \\ \mu_{k+1} &\leftarrow \alpha_k \bar{\mu}_k + (1 - \alpha_k)\mu_k, \end{aligned} \tag{4}$$

where $\alpha_k = \min\{1, \alpha^*(f_{k+1}; \mu_k, \bar{\mu}_k)\}$ and $\alpha^*(f; \mu, \bar{\mu})$ is as defined in (13).

end for

tabular or near-tabular settings, CPI/SPI admit per-iteration policy-improvement guarantees by maximizing a lower bound on the performance-difference gap. This section shows how to bring such conservative policy updates into the RPI framework.

Algorithm 2 describes CRPI. It keeps RPI's policy-evaluation step but replaces the greedy policy update with a conservative one: μ_{k+1} is chosen as a convex combination of μ_k and the policy greedy with respect to f_{k+1}. For $\alpha < 1$, Theorem 3.9 shows that this update maximizes suitable lower bounds on an *approximate performance-difference gap.*

We now formally define the various notations used in Algorithm 2. For a policy μ, P_μ is the $SA \times SA$ matrix, whose $((s, a), (s', a'))$-th entry is $P(s'|s, a)\mu(a'|s')$. We use $\nu \in \mathbb{R}^{SA}$ for an arbitrary but fixed initial distribution on the $\mathcal{S} \times \mathcal{A}$ space, and $d_\mu^\top := (1-\gamma)\nu^\top[\mathbb{I}-\gamma P_\mu]^{-1}$ for the SA-dimensional discounted state–action occupancy measure associated with μ. For a policy μ, $\delta_\mu := d_\mu^\top P$. Further, for policies μ, μ' and vector $f \in \mathbb{R}^{SA}$,

$$a_\mu^{\mu'}(f) := [P_{\mu'} - P_\mu] f \quad \text{and} \quad A_\mu^{\mu'}(f) := d_\mu^\top a_\mu^{\mu'}(f). \tag{5}$$

Note that the (s, a)-th coordinate of $a_\mu^{\mu'}(f)$, i.e.,

$$\begin{aligned} a_\mu^{\mu'}(f)(s, a) = &\sum_{s', a'} P(s'|s, a)\mu'(a'|s') \\ &\times \Big(f(s', a') - \langle \mu(\cdot|s'),\, f(s', \cdot)\rangle\Big), \end{aligned} \tag{6}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. Thus, if f is an estimate of Q_μ, then $a_\mu^{\mu'}(f)$ approximates the advantage function of μ, relative to μ'. Finally, for $x \in \mathbb{R}^{SA}$, $\text{sp}(x) := \max_{s,a} x(s, a) - \min_{s,a} x(s, a)$ denotes the span semi-norm of x, and $\|\mu_1 - \mu_2\|_{1,\delta_\mu} := \sum_s \delta_\mu(s)\ \|\mu_1(\cdot|s) - \mu_2(\cdot|s)\|_1$.

We now present our main results on CRPI. The first gives a FA generalization of the performance-difference lemma—the backbone of CPI, SPI, TRPO, and PPO.

Lemma 3.4 (Approximate Performance-Difference Lemma)**.** *Suppose μ and μ' are arbitrary stationary policies and $f \in \mathbb{R}^{SA}$ is a arbitrary vector such that $T_\mu f \geq f$. Then, for any $H \geq 0$,*

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \frac{\gamma}{1-\gamma} d_{\mu'}^\top a_\mu^{\mu'}(f) + \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]. \tag{7}$$

Remark 3.5. **Comparison with (Kakade and Langford, 2002).** The original performance-difference lemma (in the Q-value-function version) states that $\nu^\top Q_{\mu'} - \nu^\top Q_\mu = \frac{\gamma}{1-\gamma} d_{\mu'}^\top a_\mu^{\mu'}(Q_\mu)$. In tabular settings, where $Q_\mu = f$ and hence $T_\mu f = f$, our bound in (7) reduces to the above, since the second term vanishes. Our result's main benefit is that it also applies in practical FA regimes, where Q_μ is not known exactly. In particular, for any f satisfying $T_\mu f \geq f$—and therefore $Q_\mu \geq f$ by Claim A.1—our result yields a computable lower bound on an approximate performance gap between μ' and μ. We say approximate since we use $\nu^\top f$ in place of $\nu^\top Q_\mu$. Our bound also contains additional error terms involving $T_\mu f - f$, which arise purely due to FA, and these terms become smaller as the truncation horizon H increases.

Next, for mixture policies, we derive two performance-gap bounds, each quadratic in the mixing parameter α. These follow by using (7) with $H = 0$ and $H = 1$. Additional polynomial bounds follow by using higher values of H, but we do not pursue this here.

Proposition 3.6. *Let μ and $\bar{\mu}$ be arbitrary stationary policies and $f \in \mathbb{R}^{SA}$ be an arbitrary vector such that $T_\mu f \geq f$. Then, for any $\alpha \in [0, 1]$ and the mixture policy $\mu' = \alpha\bar{\mu} + (1 - \alpha)\mu$, we have*

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_1(\alpha) \geq \Psi_0(\alpha), \tag{8}$$

where

$$\begin{aligned} \Psi_1(\alpha) &= \Psi_1(\alpha; f, \mu, \bar{\mu}) \\ &= -\frac{\alpha^2\gamma^2}{2(1-\gamma)^2} \|\bar{\mu} - \mu\|_{1,\delta_\mu}\ \text{sp}(a_\mu^{\bar{\mu}}(f)) \\ &\quad + \alpha\left[\frac{\gamma}{1-\gamma} A_\mu^{\bar{\mu}}(f) + \gamma\nu^\top a_\mu^{\bar{\mu}}(T_\mu f - f)\right] \\ &\quad + \nu^\top(\mathbb{I} + \gamma P_\mu)[T_\mu f - f] \end{aligned} \tag{9}$$

and

$$\begin{aligned}\Psi_0(\alpha) &= \Psi_0(\alpha; f, \mu, \bar{\mu}) \\ &= -\frac{\alpha^2\gamma^2}{2(1-\gamma)^2}\|\bar{\mu}-\mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar{\mu}}(f)) \\ &\quad + \frac{\alpha\gamma}{1-\gamma}A_\mu^{\bar{\mu}}(f) + \nu^\top[T_\mu f - f]. \qquad (10)\end{aligned}$$

Remark 3.7. **Comparison with (Metelli et al., 2021).** Our bound in (10) generalizes equation (P.6) of (Metelli et al., 2021): (i) it uses $A_\mu^{\bar{\mu}}(f)$ and $a_\mu^{\bar{\mu}}(f)$ instead of the true advantage function estimates and (ii) it introduces an additional $(T_\mu f - f)$ term that captures FA error. Moreover, the specialization (9) dominates (10) (i.e., is pointwise larger) and incorporates FA-dependent coefficients, yielding extra FA-specific guidance for choosing the mixture parameter.

Our next result provides a per-step improvement guarantee for CRPI under general FA.

For arbitrary stochastic policies μ and $\bar{\mu}$ and a arbitrary vector $f \in \mathbb{R}^{SA}$, let

$$\alpha_1^* \equiv \alpha_1^*(f, \mu, \bar{\mu}) := \arg\max_{\alpha\in\mathbb{R}} \Psi_1(\alpha) = \frac{\eta_1 + \eta_2}{\partial_1} \qquad (11)$$

and

$$\alpha_0^* \equiv \alpha_0^*(f, \mu, \bar{\mu}) := \arg\max_{\alpha\in\mathbb{R}} \Psi_0(\alpha) = \frac{\eta_1}{\partial_1}, \qquad (12)$$

where $\eta_1 \equiv \eta_1(f, \mu, \bar{\mu}) = (1-\gamma)A_\mu^{\bar{\mu}}(f)$, $\eta_2 \equiv \eta_2(f, \mu, \bar{\mu}) = (1-\gamma)^2\nu^\top a_\mu^{\bar{\mu}}(T_\mu f - f)$, and $\partial_1 \equiv \partial_1(f, \mu, \bar{\mu}) = \gamma\|\bar{\mu}-\mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar{\mu}}(f))$. Finally, let

$$\alpha^* \equiv \alpha^*(f, \mu, \bar{\mu}) = \begin{cases}\alpha_1^* & \text{if } \alpha_1^* > 0, \\ \alpha_0^* & \text{otherwise.}\end{cases} \qquad (13)$$

Theorem 3.8. *Let μ be an arbitrary stochastic policy and $f \in \mathbb{R}^{SA}$ an arbitrary vector such that $T_\mu f \geq f$. Also, let $\bar{\mu}$ be a greedy policy with respect to f and ν an arbitrary initial distribution on $\mathcal{S}\times\mathcal{A}$. Then, $\alpha_0^* \geq 0$ and the following bounds hold for the mixture policy $\mu' = \alpha\bar{\mu} + (1-\alpha)\mu$ where $\alpha = \min\{1, \alpha^*\}$.*

1. *If $\alpha_1^* > 1$, then $\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_1(1)$.*
2. *If $\alpha_1^* \in [0,1]$, then $\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_1(\alpha_1^*)$.*
3. *If $\alpha_1^* < 0$ and $\alpha_0^* > 1$, then $\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_0(1)$.*
4. *If $\alpha_1^* < 0$ and $\alpha_0^* \leq 1$, then $\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_0(\alpha_0^*)$.*

Theorem 3.9. *Suppose $\mathcal{F} \subseteq \mathbb{R}^{SA}$ is closed and the initial pair (μ_0, f_0) satisfies $T_{\mu_0} f_0 \geq f_0$. Then all conclusions of Theorem 3.1 hold for the CRPI sequence $(f_k, \mu_k)_{k\geq 0}$. In particular, for every $k \geq 0$, $T_{\mu_k} f_{k+1} \geq f_{k+1}$. Consequently, Theorem 3.8 applies to the gap $\nu^\top Q_{\mu_{k+1}} - \nu^\top f_{k+1}$ for every $k \geq 0$ and every initial distribution ν on $\mathcal{S}\times\mathcal{A}$.*

Remark 3.10. Ideally, at iteration k, one would prefer an improvement guarantee on $\nu^\top Q_{\mu_{k+}} - \nu^\top Q_{\mu_k}$. However, with FA, Q_{μ_k} can only be known approximately. Our result therefore guarantees improvement in terms of f_{k+1}, the certified underestimator to Q_{μ_k}. These bounds are, to our knowledge, the first per-step guarantees for FA and coincide with (Metelli et al., 2021, Corollary 5) in the tabular case where $f_{k+1} = Q_{\mu_k}$.

We end with CRPI's convergence rate, where we show that the suboptimality gap converges at $O(1/k)$ rate to a neighborhood of 0 and stays there thereafter.

Assumption 3.11. There exists $\Delta > 0$ such that $d_{\mu_k}(s,a) \geq \Delta$ for $k \geq 0$ and (s,a) with $d_{\mu^*}(s,a) > 0$.

Theorem 3.12. *Suppose that $T_{\mu_0} f_0 \geq f_0 \geq 0$, and that $\|\cdot\|$ is strictly monotone. Further suppose that, for every (μ, g) with $T_\mu g \geq g$, the set $\mathcal{H}(\mu, g) := \{f \in \mathcal{F} : T_\mu f \geq f \geq g\}$ has a greatest element $f^{\max}(\mu, g)$ satisfying $Q_\mu - \epsilon\mathbf{1} \leq f^{\max}(\mu, g) \leq Q_\mu$ for some $\epsilon > 0$, where $\mathbf{1}$ is all ones vector. Then, under Assumption 3.11, if CRPI is run with $\alpha_k = \frac{(1-\gamma)^2 A_\mu^{\bar{\mu}}(f_{k+1})}{2R_{\max}}$, $g_k := \nu^\top(Q^\star - Q_{\mu_k})$ satisfies $g_k \leq \max\{2C_1/k, \delta\}$, where $\delta = \sqrt{2C_1(C_2\epsilon^2 + \epsilon)} + C_2\epsilon^2 + \epsilon = O(\sqrt{\epsilon} + \epsilon^2)$ with $C_1 = \frac{16R_{\max}\gamma}{(1-\gamma)^3\Delta^2}$ and $C_2 = \frac{(1-\gamma)\Delta^2}{4R_{\max}\gamma}$. Moreover, the interval $[0, \delta]$ is invariant: $g_k \leq \delta \implies g_{k+1} \leq \delta$.*

4 PROOF SKETCHES

Here we give an outline of our proofs for Theorems 3.1, 3.9, and 3.12, and a complete proof of Lemma 3.4. All the other details can be found in Appendix A.

Sketch of Proof of Theorem 3.1. We first show in Claim A.1 that the constraint $T_\mu f \geq f$ implies $Q_\mu \geq f$. This follows from the monotonicity of T_μ and the fact that $(T_\mu)^m f \to Q_\mu$ as $m \to \infty$. Using this result, we prove the four statements as follows.

(1) Feasibility. Using induction, we show that the constraint set in (1) remains feasible at every iteration. The initialization ensures feasibility at $k = 0$. Assuming f_k is feasible, Claim A.1 then yields $f_k \leq Q_{\mu_k}$, which bounds the feasible set. Since the constraint set is closed and bounded, it is compact, and hence an optimizer f_{k+1} exists. Moreover, the optimizer satisfies

$$T_{\mu_k} f_{k+1} \geq f_{k+1} \geq f_k. \qquad (14)$$

Now, the policy μ_{k+1} is greedy with respect to f_{k+1}, so $T_{\mu_{k+1}} f_{k+1} = T f_{k+1}$. Using $Tf \geq T_{\mu_k} f$, we get

$$T_{\mu_{k+1}} f_{k+1} = T f_{k+1} \geq T_{\mu_k} f_{k+1} \geq f_{k+1}, \qquad (15)$$

which shows that f_{k+1} satisfies the constraints in (1) for the $k+1$ iteration, as desired.

(2) Monotonicity and lower bound. The inequality $f_{k+1} \geq f_k$ in (14) implies that $(f_k)_{k\geq 0}$ is monotonically non-decreasing. Separately, since $T_{\mu_k} f_k \geq f_k$ holds at every iteration as shown in (15), applying Claim A.1 yields $Q_{\mu_k} \geq f_k$ for all k, as desired.

(3) Convergence and sub-optimality gap. Since $(f_k)_{k\geq 0}$ is monotonically non-decreasing and bounded above by $Q^\star$ through $f_k \leq Q_{\mu_k} \leq Q^\star$, the limit $f_\infty := \lim_{k\to\infty} f_k$ exists. Using continuity of the Bellman operator and feasibility of (f_k), we pass to the limit in the constraint $T_{\mu_k} f_k \geq f_k$ to obtain $T_{\mu_\infty} f_\infty = T f_\infty \geq f_\infty$. We then use contraction of T and T_{μ_∞} to relate f_∞ to $Q^\star$ and Q_{μ_∞}. In particular, we show that $\|f_\infty - Q^\star\|_\infty \leq \frac{\|Tf_\infty - f_\infty\|_\infty}{1-\gamma}$. Now $Tf_\infty = T_{\mu_\infty} f_\infty$ and $f_\infty \leq Q_{\mu_\infty} \leq Q^\star$ yield

$$\|Q_{\mu_\infty} - Q^\star\|_\infty \leq \|f_\infty - Q^\star\|_\infty \leq \frac{\|Tf_\infty - f_\infty\|_\infty}{1-\gamma} = \frac{\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}.$$

(4) Partial Bellman optimality. Since the number of deterministic policies is finite, there is a policy μ that appears infinitely often in (μ_k). So, there is a subsequence (k_n) such that $\mu_{k_n} = \mu$ for all n. At these indices, feasibility implies $Tf_{k_n} = T_\mu f_{k_n} \geq f_{k_n}$. Passing to the limit using continuity of T and T_μ, we get $Tf_\infty = T_\mu f_\infty \geq f_\infty$. For the sake of contradiction, suppose that $T_\mu f_\infty > f_\infty$ holds strictly. Then, by continuity and the room-to-improve property of $\mathcal{F}$ at f_∞, there exists a function $f \in \mathcal{F}$ such that $f > f_\infty$ and $T_\mu f > f$. Consequently, at any index k with $\mu_k = \mu$, the monotonicity of $\|\cdot\|$ would imply that the solution f_{k+1} of the optimization problem in (1) would have satisfied $f_{k+1} > f_\infty$, a contradiction. $\square$

We now prove Lemma 3.4, the approximate performance-difference lemma.

Proof of Lemma 3.4. First observe that

$$\begin{aligned} & Q_{\mu'} - f \\ &= Q_{\mu'} - T_\mu f + T_\mu f - f \\ &\stackrel{(a)}{=} T_{\mu'} Q_{\mu'} - T_\mu f + T_\mu f - f \\ &\stackrel{(b)}{=} \gamma P_{\mu'}[Q_{\mu'} - f] + \gamma a_\mu^{\mu'}(f) + T_\mu f - f \\ &\stackrel{(c)}{=} \gamma[\mathbb{I} - \gamma P_{\mu'}]^{-1} a_\mu^{\mu'}(f) + [\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f], \end{aligned}$$

where (a) follows since $T_{\mu'} Q_{\mu'} = Q_{\mu'}$, (b) follows from the definitions of $T_{\mu'}$, T_μ, and $a_\mu^{\mu'}(f)$, and (c) follows by taking the first term to the left and then multiplying $[\mathbb{I} - \gamma P_{\mu'}]^{-1}$ on both sides. Multiplying the last relation on both sides by $\nu^\top$ then gives

$$\begin{aligned} & \nu^\top [Q_{\mu'} - f] \\ &= \frac{\gamma}{1-\gamma} d_{\mu'}^\top a_\mu^{\mu'}(f) + \nu^\top [\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f]. \end{aligned} \quad (16)$$

Now, $\nu^\top [\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f] = \sum_{h=0}^\infty \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]$. Also, $T_\mu f \geq f$, and $P_{\mu'}^h$ and ν are made up of non-negative entries; hence, $\sum_{h=H+1}^\infty \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f] \geq 0$. The desired result now follows. $\square$

Sketch of Proof of Theorem 3.9. We use induction to first show that CRPI's (f_k, μ_k) pairs satisfy $T_{\mu_k} f_k \geq f_k$, $k \geq 0$. The $k = 0$ case holds due to initialization. Suppose $T_{\mu_k} f_k \geq f_k$ for some $k \geq 0$. Then, the solution f_{k+1} to (3) exists and satisfies $T_{\mu_k} f_{k+1} \geq f_{k+1}$. Using arguments as in (15), it then follows that $T_{\bar{\mu}_k} f_{k+1} \geq f_{k+1}$. Since $\mu_{k+1} = \alpha \bar{\mu}_k + (1-\alpha)\mu_k$ for some $\alpha \in [0,1]$, we then have $\alpha T_{\bar{\mu}_k} f_{k+1} + (1-\alpha) T_{\mu_k} f_{k+1} \geq f_{k+1}$. From the definition of T_{μ_k} and $T_{\bar{\mu}_k}$ and using the fact that $\alpha P_{\mu_k} + (1-\alpha) P_{\bar{\mu}_k} = P_{\mu_{k+1}}$, it then follows that $T_{\mu_{k+1}} f_{k+1} \geq f_{k+1}$, as desired.

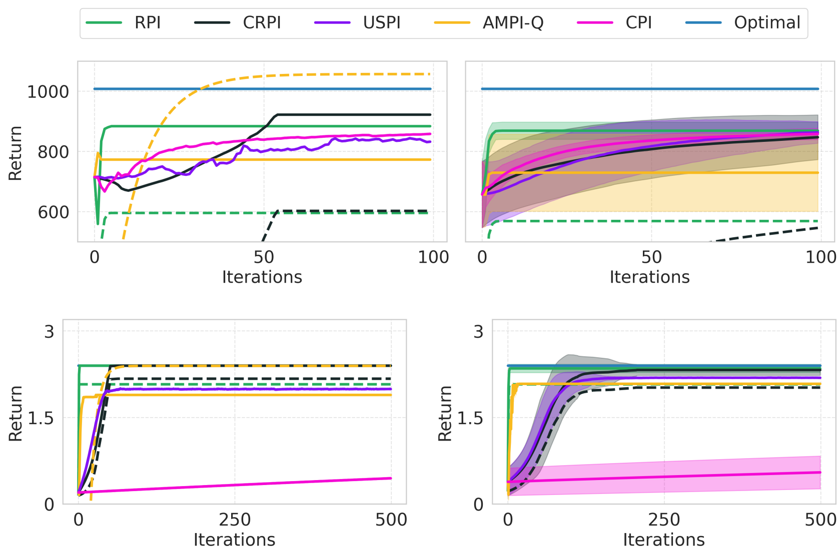
By reasoning analogous to the proof of Theorem 3.1, the conclusions of its first three statements can be shown to hold for CRPI as well. The fourth statement does not carry over as directly: RPI generates deterministic policies, whereas CRPI may produce stochastic ones. Nevertheless, the argument establishing $Tf_\infty \geq_{\mathrm{p}} f_\infty$ in Theorem 3.1's proof adapts to CRPI due to the compactness of the policy simplex.

Finally, for any $k \geq 0$, to invoke Theorem 3.8 for $\nu^\top Q_{\mu_{k+1}} - \nu^\top f_{k+1}$, we only need $T_{\mu_k} f_{k+1} \geq f_{k+1}$, which we have already established above. $\square$

5 EXPERIMENTS

We evaluate our methods against three model-based baselines: CPI, USPI, and AMPI-Q. We conduct our experiments on two environments with linear FA: the inventory control problem (Bertsekas, 2012) (dense rewards) and the chain walk (Lagoudakis and Parr, 2003) (sparse rewards). We also test a model-free deep RL instantiation of RPI using the DDPG framework (Lillicrap et al., 2015) on MuJoCo continuous-control tasks (Brockman et al., 2016; Todorov et al., 2012).

For each environment we report learning curves, terminal performance, and the area under the learning curve (AUC) as a measure of sample efficiency. Additional results on CartPole-v1 and InvertedPendulum-v5 appear in our earlier work (Eshwar et al., 2025). Implementation details—such as initialization (B), environment descriptions (D), and computational resources and solvers (F)—are provided in the appendix. All



Env: Inventory Control		
Algo.	**AUC** $(\times 10^3)$	**Terminal Perf.** $(\times 10^2)$
RPI	$\mathbf{85.70 \pm 2.80}$	$\mathbf{8.69 \pm 0.29}$
CRPI	78.44 ± 8.97	8.47 ± 0.74
USPI	78.78 ± 7.56	8.63 ± 0.36
AMPI-Q	72.08 ± 12.47	7.29 ± 1.28
CPI	80.89 ± 1.82	8.58 ± 0.19

Env: Chain Walk		
Algo.	**AUC** $(\times 10^2)$	**Terminal Perf.**
RPI	$\mathbf{11.71 \pm 0.36}$	$\mathbf{2.35 \pm 0.07}$
CRPI	10.42 ± 0.79	2.32 ± 0.09
USPI	9.94 ± 0.34	2.19 ± 0.01
AMPI-Q	10.32 ± 0.01	2.08 ± 0.00
CPI	2.34 ± 1.31	0.55 ± 0.28

Figure 1: Inventory Control and Chain Walk with linear function approximation. **Left:** Training curve of a single representative run (solid: true return, dashed: estimated return). **Center:** Averaged training curves across random seeds (100 for Inventory Control, 25 for Chain Walk; solid: mean return, shaded: mean $\pm$ 1 std). **Right:** Key metrics table reporting terminal performance and AUC (mean $\pm$ std). **Summary:** RPI and CRPI maintains value estimates that lower bound the true returns, while those of CPI and USPI are very close to true returns. Nevertheless, RPI achieves the best average terminal performance and AUC across runs, indicating faster and more sample-efficient learning. CRPI exhibits larger variance across runs in Inventory Control, which leads to trajectories that outperform RPI, as illustrated in the representative run. AMPI-Q tends to overestimate and can exceed the optimal value, while CPI performs poorly in the sparse-reward Chain Walk setting.

code and experimental outputs are publicly available at `https://github.com/EshwarSR/RPI`.

5.1 Model-Based Experiments

Here we describe our inventory control (dense reward) and chain walk (sparse reward) experiments.

Inventory Control. We consider the inventory-control setup of Bertsekas (2012) with $M = 49$ (so $|\mathcal{S}| = |\mathcal{A}| = 50$), unit cost $c = 5$, holding cost $h = 1$, selling price $p = 10$, and discount factor $\gamma = 0.9$. Demand is uniform on $\{0, \dots, M\}$. We use linear FA with feature dimension $d = 75$, where the entries of $\Phi \in \mathbb{R}^{SA \times d}$ are sampled uniformly from $[1, 5]$.

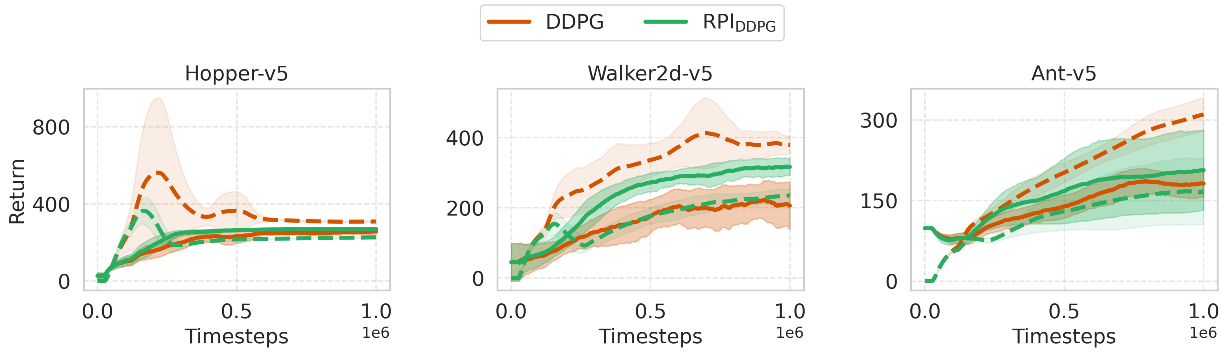
Figure 1 (top row) shows the results. The main highlight is that, under FA, preserving a lower-bound structure can be more useful than closely tracking true values. Although CPI and USPI produce estimates often very close to the true returns, they do not translate this into better policy performance. By contrast, RPI attains the best average terminal performance and AUC, suggesting that its conservative updates lead to faster and more reliable improvement. CRPI remains competitive but exhibits higher variance across runs.

Chain Walk. We next consider the Chain Walk domain (Lagoudakis and Parr, 2003), a sparse-reward MDP with $N = 50$ states arranged in a linear chain and actions `Left` and `Right`. The agent moves in the intended direction with probability 0.9 and in the opposite direction with probability 0.1. Rewards are sparse, obtained only in two states located $N/4$ from the left and right boundaries. We use linear FA with feature dimension $d = 90$, where entries are sampled uniformly from $[1, 5]$, and discount factor $\gamma = 0.9$. Performance in this domain is sensitive to feature choice. For a fair comparison, we sample 10 candidate feature matrices, evaluate each algorithm over 25 random seeds on each matrix, and report results corresponding to the best-performing matrix.

Figure 1 (bottom) reports results for this task. RPI achieves the highest terminal performance and AUC on average, while CRPI is competitive. AMPI-Q again exhibits value overestimation, and CPI performs poorly due to very slow learning.

5.2 Model-Free Experiments

We now demonstrate the applicability of our framework in deep RL settings, for which we develop RPI_{DDPG}. This method replaces the standard Mean-



Metric	Algorithm	Swimmer	Hopper	HalfCheetah	Walker2d	Ant
Terminal Performance	DDPG	19.4 ± 5.8	257.6 ± 9.7	564.8 ± 64.7	200.3 ± 34.1	182.8 ± 7.6
	RPI_{DDPG}	**22.3 ± 1.8**	**270.9 ± 2.0**	**632.0 ± 50.4**	**319.9 ± 17.1**	**208.2 ± 73.5**
AUC ($\times 10^6$)	DDPG	20.1 ± 2.7	219.3 ± 10.3	483.7 ± 58.2	165.7 ± 4.8	144.4 ± 8.4
	RPI_{DDPG}	**21.0 ± 2.6**	**241.5 ± 6.5**	**550.2 ± 48.2**	**245.0 ± 10.6**	**163.4 ± 50.5**

Figure 2: Performance comparison between DDPG and RPI_{DDPG} on MuJoCo environments. **Top:** Learning curves for Hopper-v5, Walker2d-v5, and Ant-v5 (mean ± 1 std; solid: return, dashed: critic estimate). Remaining environments are in Appendix E. **Bottom:** Terminal performance and AUC (mean ± std) across all environments. **Summary:** RPI_{DDPG} maintains lower-bound value estimates, while DDPG often overestimates. RPI_{DDPG} matches DDPG on simpler tasks and outperforms DDPG on harder environments.

Squared-Bellman-Error (MSBE) critic loss of DDPG with a custom loss function which enforces RPI's Bellman-constrained optimization. Specifically, our inequality constraints are incorporated into the critic loss using a penalty formulation, where the penalty weight is adjusted dynamically during training. Implementation details are provided in Appendix C.

We evaluate RPI_{DDPG} against the standard DDPG algorithm on five tasks: Swimmer-v5, Hopper-v5, HalfCheetah-v5, Walker2d-v5, and Ant-v5. Figure 2 shows the learning curves for Hopper-v5, Walker2d-v5, and Ant-v5 averaged over five seeds. The plots of remaining environments are provided in Appendix E.

The results support the same message as in our linear-FA experiments: enforcing a conservative critic improves reliability without sacrificing performance. Across environments, the critic estimates produced by RPI_{DDPG} mostly remain below the empirical returns, in line with the lower-bound structure induced by our method; when small violations occur early in training, the adaptive penalty quickly suppresses them. By contrast, DDPG often exhibits overestimation, particularly in Walker2d-v5 and Ant-v5.

Overall, RPI_{DDPG} performs competitively with or better than DDPG across all tasks. In simpler environments such as Swimmer-v5 and Hopper-v5 both methods achieve similar performance, while in the remaining environments RPI_{DDPG} attains higher returns and learns faster. The summary table confirms this trend: RPI_{DDPG} achieves higher terminal performance and larger AUC across all environments.

6 CONCLUSION

We introduced RPI and its conservative variant, CRPI, to address the breakdown of classical PI guarantees under FA. The core idea is a Bellman-constrained policy-evaluation step that produces certified lower bounds and restores monotonic value estimates. We further derive a new performance-difference lower bound that allows CRPI to carry out conservative policy updates with per-step improvement guarantees. Empirically, these ideas translate into strong performance. RPI performs best overall, achieving the highest average terminal performance and AUC across Inventory Control and Chain Walk. CRPI remains competitive: it closely tracks RPI in Chain Walk, while in Inventory Control its conservative updates lead to higher variance but can also outperform RPI in some runs. Overall, our work offers a principled bridge between tabular-style reliability guarantees and practical RL with FA. Our guarantees, however, impose a constraint: when the function class admits no feasible improving direction, the method may stagnate. However, this behavior is not a limitation but a consequence of enforcing reliability—it prevents harmful updates. We illustrate this phenomenon in Appendix G.

Acknowledgements

We thank anonymous reviewers for their valuable comments to improve our work. We also thank Aniruddha Mukherjee for helping run some of the simulations. Eshwar's research is supported in part by grant from the National Payments Corporation of India (NPCI) to Indian Institute of Science (IISc) and Prime Minister's Research Fellowship. Gugan Thoppe's research is supported in part by grants from the Walmart Centre for Tech Excellence at IISc, the Indo-French Centre for the Promotion of Advanced Research Grant (CEFIPRA; Project 7102-1), Kotak IISc FinTech Grant, DST-SERB's Core Research Grant (CRG/2021/008330), and the Pratiksha Trust Young Investigator Award. Aditya Gopalan's research is supported in part by the Centre for Networked Intelligence (a Cisco Corporate Social Responsibility (CSR) Initiative), IISc. Both Gugan and Aditya are also supported by the ANRF ARG grant (ANRF/ARG/2025/011952/ENS).

References

Aethelios (2025). Beyond hype: The brutal truth about deep reinforcement learning. *Medium*. Blog post.

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

Agrawal, A., Verschueren, R., Diamond, S., and Boyd, S. (2018). A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60.

Alfano, C., Yuan, R., and Rebeschini, P. (2023). A novel framework for policy mirror descent with general parameterization and linear convergence. *Advances in Neural Information Processing Systems*, 36:30681–30725.

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., et al. (2024). Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *Proceedings of the 29th ACM international conference on architectural support for programming languages and operating systems, volume 2*, pages 929–947.

Baird, L. (1995). Residual algorithms: Reinforcement learning with function approximation. In *Proceedings of the twelfth international conference on machine learning*, pages 30–37.

Bertsekas, D. (2012). *Dynamic programming and optimal control*, volume 1. Athena scientific.

Bertsekas, D. and Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Athena Scientific.

Bertsekas, D. P. (2011). Approximate policy iteration: A survey and some new methods. *Journal of Control Theory and Applications*, 9(3):310–335.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. (2016). Openai gym. *arXiv preprint arXiv:1606.01540*.

De Farias, D. P. and Van Roy, B. (2003). The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865.

Diamond, S. and Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5.

Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2018a). Beyond the one-step greedy approach in reinforcement learning. In *International Conference on Machine Learning*, pages 1387–1396. PMLR.

Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2018b). Multiple-step greedy policies in approximate and online reinforcement learning. *Advances in neural information processing systems*, 31.

Efroni, Y., Dalal, G., Scherrer, B., and Mannor, S. (2019). How to combine tree-search methods in reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3494–3501.

Ernst, D., Geurts, P., and Wehenkel, L. (2005). Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6.

Eshwar, S., Mukherjee, A., Saha, K., Agarwal, K., Thoppe, G., Gopalan, A., and Dalal, G. (2025). Reliable policy iteration: Performance robustness across architecture and environment perturbations. In *2025 Eleventh Indian Control Conference (ICC-11)*.

Fujimoto, S., Hoof, H., and Meger, D. (2018). Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR.

Gopalan, A. and Thoppe, G. (2025). Does DQN learn? *IEEE Transaction on Automatic Control (to appear)*.

Gurobi Optimization, LLC (2024). Gurobi Optimizer Reference Manual.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. Pmlr.

Hasselt, H. (2010). Double q-learning. *Advances in neural information processing systems*, 23.

Haviv, M. and Van der Heyden, L. (1984). Perturbation bounds for the stationary probabilities of a finite markov chain. *Advances in Applied Probability*, 16(4):804–818.

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. (2018). Deep reinforcement learning that matters. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Howard, R. A. (1960). Dynamic programming and markov processes.

Kakade, S. and Langford, J. (2002). Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, pages 267–274.

Lagoudakis, M. G. and Parr, R. (2003). Least-squares policy iteration. *Journal of machine learning research*, 4(Dec):1107–1149.

Lazaric, A., Ghavamzadeh, M., and Munos, R. (2010). Analysis of a classification-based policy iteration algorithm. In *ICML-27th International Conference on Machine Learning*, pages 607–614. Omnipress.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Liu, B., Cai, Q., Yang, Z., and Wang, Z. (2019). Neural trust region/proximal policy optimization attains globally optimal policy. *Advances in neural information processing systems*, 32.

Metelli, A. M., Pirotta, M., Calandriello, D., and Restelli, M. (2021). Safe policy iteration: A monotonically improving approximate policy iteration approach. *Journal of Machine Learning Research*, 22(97):1–83.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Patterson, A., Neumann, S., White, M., and White, A. (2024). Empirical design in reinforcement learning. *Journal of Machine Learning Research*, 25(318):1–63.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229.

Samuel, A. L. (1967). Some studies in machine learning using the game of checkers. ii—recent progress. *IBM Journal of research and development*, 11(6):601–617.

Scherrer, B. (2010). Should one compute the temporal difference fix point or minimize the bellman residual? the unified oblique projection view. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 959–966.

Scherrer, B., Ghavamzadeh, M., Gabillon, V., Lesner, B., and Geist, M. (2015). Approximate modified policy iteration and its application to the game of tetris. *Journal of Machine Learning Research*, 16(49):1629–1676.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015). Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR.

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

Thrun, S. and Schwartz, A. (1993). Issues in using function approximation for reinforcement learning. In *Proceedings of the 1993 connectionist models summer school*, pages 255–263.

Todorov, E., Erez, T., and Tassa, Y. (2012). Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE.

Tsitsiklis, J. and Van Roy, B. (1996). Analysis of temporal-diffference learning with function approximation. *Advances in neural information processing systems*, 9.

Van Hasselt, H., Guez, A., and Silver, D. (2016). Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Winnicki, A., Lubars, J., Livesay, M., and Srikant, R. (2021). The role of lookahead and approximate policy evaluation in policy iteration with linear value function approximation. *CoRR*.

Young, K. and Sutton, R. S. (2020). Understanding the pathologies of approximate policy evaluation when combined with greedification in reinforcement learning. *arXiv preprint arXiv:2010.15268*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. **Yes**

   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. **Yes**
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. **Yes**

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. **Yes**
   (b) Complete proofs of all theoretical results. **Yes**
   (c) Clear explanations of any assumptions. **Yes**

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). **Yes**
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). **Yes**
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). **Yes**
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). **Yes**

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. **Yes**
   (b) The license information of the assets, if applicable. **Yes**
   (c) New assets either in the supplemental material or as a URL, if applicable. **Yes**
   (d) Information about consent from data providers/curators. **Not Applicable**
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. **Not Applicable**

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. **Not Applicable**
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. **Not Applicable**
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. **Not Applicable**

Appendix

A PROOFS

We begin with Theorem 3.1's proof. First, we derive a key relation from the constraints in (1).

Claim A.1. *For a policy μ and a vector $f \in \mathbb{R}^{SA}$, the condition $T_\mu f \geq f$ implies $Q_\mu \geq f$.*

Proof. The given condition and the monotonicity of T_μ imply $(T_\mu)^m f \geq \cdots \geq T_\mu f \geq f$ for any $m \geq 0$. Hence, $Q_\mu = \lim_{m\to\infty}(T_\mu)^m f \geq f$, as desired. □

Proof of Theorem 3.1. We now use Claim A.1 and induction to prove the first statement. The condition $T_{\mu_0} f_0 \geq f_0$ ensures that f_0 is feasible with respect to the constraints in (1) for $k = 0$. Now, suppose $f = f_k$ satisfies the constraints in (1) for some arbitrary $k \geq 0$. Then, Claim A.1 shows that

$$\begin{aligned} \{f \in \mathcal{F} :& T_{\mu_k} f \geq f \geq f_k\} \\ &= \mathcal{F} \cap \{f \in \mathbb{R}^{SA} : T_{\mu_k} f \geq f \geq f_k\} \\ &\subseteq \mathcal{F} \cap \{f \in \mathbb{R}^{SA} : Q_{\mu_k} \geq f \geq f_k\}. \end{aligned} \tag{17}$$

Since $\mathcal{F}$ and $\{f \in \mathbb{R}^{SA} : T_{\mu_k} f \geq f \geq f_k\}$ are closed in $\mathbb{R}^{SA}$ and $\{f \in \mathbb{R}^{SA} : Q_{\mu_k} \geq f \geq f_k\}$ is bounded, the constraint set $\{f \in \mathcal{F} : T_{\mu_k} f \geq f \geq f_k\}$ itself is closed and bounded and, hence, compact. Because the objective function is continuous, compactness implies that a solution f_{k+1} to (1) exists and it satisfies

$$T_{\mu_k} f_{k+1} \geq f_{k+1} \geq f_k. \tag{18}$$

The new policy μ_{k+1} obtained subsequently from f_{k+1} then satisfies

$$T_{\mu_{k+1}} f_{k+1} = T f_{k+1} \geq T_{\mu_k} f_{k+1} \geq f_{k+1}, \tag{19}$$

which shows that f_{k+1} satisfies the constraints in (1) for the $k+1$ iteration, as desired.

Now consider the second statement. From the rightmost inequality in (18), it follows that $(f_k)_{k\geq 0}$ is non-decreasing. On the other hand, for any $k \geq 0$, our first statement shows that f_k satisfies the constraints in (1); Claim A.1 then shows that $Q_{\mu_k} \geq f_k$. Therefore, f_k is a lower bound on Q_{μ_k} for any $k \geq 0$, and $(f_k)_{k\geq 0}$ is monotonically non-decreasing, as desired.

With regard to the third statement, since $(f_k)_{k\geq 0}$ is monotonically non-decreasing and $f_k \leq Q_{\mu_k} \leq Q^\star$, it follows that $f_\infty := \lim_{k\to\infty} f_k$ exists. Separately, for any $k \geq 0$, since $f_{k+1} \geq f_k$ and since T_{μ_k} is monotone, we have $T_{\mu_k} f_{k+1} \geq T_{\mu_k} f_k$. This, when combined with (19), then shows that $(T_{\mu_k} f_k)_{k\geq 0}$ is monotone. Therefore,

$$T_{\mu_\infty} f_\infty \overset{(a)}{=} T f_\infty \overset{(b)}{=} T(\lim_{k\to\infty} f_k) \overset{(c)}{=} \lim_{k\to\infty} T f_k \overset{(d)}{\geq} \lim_{k\to\infty} T_{\mu_k} f_k \overset{(e)}{\geq} \lim_{k\to\infty} f_k \overset{(f)}{=} f_\infty, \tag{20}$$

where (a) holds because μ_∞ is greedy with respect f_∞, (b) and (f) hold from the definition of f_∞, (c) holds since T is continuous, (d) holds since $T f_k \geq T_{\mu_k} f_k$ and $(T_{\mu_k} f_k)_{k\geq 0}$ is monotone and, hence, convergent, while (e) holds since f_k satisfies the constraints in (1) as shown in our first statement.

Next, since $TQ^\star = Q^\star$, observe that

$$\|f_\infty - Q^\star\|_\infty \leq \|Tf_\infty - f_\infty\|_\infty + \|Tf_\infty - TQ^\star\|_\infty \leq \|Tf_\infty - f_\infty\|_\infty + \gamma\|f_\infty - Q^\star\|_\infty,$$

where the last inequality follows since T is a contraction. Hence,

$$\|f_\infty - Q^\star\|_\infty \leq \frac{\|Tf_\infty - f_\infty\|_\infty}{1-\gamma}. \tag{21}$$

Separately, from (20), we have $T_{\mu_\infty} f_\infty \geq f_\infty$. Hence,

$$f_\infty \leq Q_{\mu_\infty} \leq Q^\star, \tag{22}$$

where the first inequality follows from Claim A.1. Therefore,

$$\|Q^\star - Q_{\mu_\infty}\|_\infty \overset{(a)}{\leq} \|Q^\star - f_\infty\|_\infty \overset{(b)}{\leq} \frac{\|Tf_\infty - f_\infty\|_\infty}{1-\gamma} \overset{(c)}{=} \frac{\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}$$

where (a) follows from (22), (b) from (21), while (c) since $Tf_\infty = T_{\mu_\infty} f_\infty$.

Finally, we discuss the fourth statement on the partial satisfiability of the Bellman equations by f_∞. Since the number of state-action pairs is finite, we only have finitely many deterministic policies. Hence, among $(\mu_k)_{k\geq 0}$, there exists a deterministic policy (say μ) that repeats infinitely often. That is, there is a subsequence $(k_n)_{n\geq 0}$ such that $\mu_{k_n} = \mu$ for all $n \geq 0$. Now, since $\lim_{n\to\infty} f_{k_n} = f_\infty$ and, at iteration k_n, $Tf_{k_n} = T_\mu f_{k_n} = T_{\mu_{k_n}} f_{k_n} \geq f_{k_n}$, the continuity of T and T_μ implies $Tf_\infty = T_\mu f_\infty \geq f_\infty$. Suppose $Tf_\infty = T_\mu f_\infty > f_\infty$, i.e., the strict inequality holds on all coordinates. Let $\eta := \min_{s,a} |T_\mu f_\infty(s,a) - f_\infty(s,a)|$, where $T_\mu f_\infty(s,a)$ and $f_\infty(s,a)$ denote the (s,a)-th coordinate of $T_\mu f_\infty$ and f_∞, respectively. Then, from the continuity of T_μ, it follows that there exist some δ and ϵ such that $0 < \delta, \epsilon < \eta/2$ and, for any $f \in \mathbb{R}^{SA}$ satisfying $\|f - f_\infty\|_\infty \leq \delta$, we have $\|T_\mu f - T_\mu f_\infty\|_\infty \leq \epsilon$. Now, the given condition that $\mathcal{F}$ has room to improve at f_∞ implies there exists a $f \in \mathcal{F}$ such that $f > f_\infty$ and $\|f - f_\infty\|_\infty \leq \delta$. Hence, for this f, we have $T_\mu f > f$. Consequently, at any index k with $\mu_k = \mu$, the monotonicity of $\|\cdot\|$ would imply that the solution f_{k+1} of the optimization problem in (1) would have satisfied $f_{k+1} > f_\infty$, a contradiction. This shows that $Tf_\infty \geq_{\mathrm{p}} f_\infty$, as desired. □

We next derive Propositions 3.2 and 3.3.

Proof of Proposition 3.2. Since $\mathcal{F} = \mathbb{R}^{SA}$, we have that $Q_\mu \in \mathcal{F}$ for any μ. Now, for iteration $k \geq 0$ of RPI, Claim A.1 implies that every element in $\{f \in \mathcal{F} : T_{\mu_k} f \geq f \geq f_k\}$ satisfies $Q_{\mu_k} \geq f \geq f_k$ or, equivalently, $Q_{\mu_k} - f_k \geq f - f_k \geq 0$. The strict monotonicity of $\|\cdot\|$ then implies that $\|Q_{\mu_k} - f_k\| \geq \|f - f_k\|$. Hence, $f_{k+1} = Q_{\mu_k}$. Thus, RPI mirrors vanilla PI. Therefore, by invoking the classical convergence proof for PI (Bertsekas and Tsitsiklis, 1996), we now get $f_\infty = Q^\star$ and μ_∞ is the optimal policy, as desired. □

Proof of Proposition 3.3. For any f satisfying the constraints in (1), we have from Claim A.1 that $Q_{\mu_k} \geq f \geq f_k$. Hence, for any weight vector $w \in \mathbb{R}^{SA}$ with strictly positive values, we have

$$\begin{aligned}
\|f - f_k\|_{w,1} &= \sum_{s,a} w(s,a)[f(s,a) - f_k(s,a)] \\
&= \sum_{s,a} w(s,a)[f(s,a) - Q_{\mu_k}(s,a) + Q_{\mu_k}(s,a) - f_k(s,a)] \\
&= -\|Q_{\mu_k} - f\|_{w,1} + \|f_k - Q_{\mu_k}\|_{w,1}.
\end{aligned}$$

Since the rightmost term in the last expression is independent of f, the desired result follows. □

Next we prove Proposition 3.6. To this end, we need the following technical result that mirrors Lemma 1 of (Metelli et al., 2021).

Lemma A.2. *For any initial distribution ν and stationary policies μ', μ, we have*

$$\|d_{\mu'} - d_\mu\|_1 \leq \frac{\gamma}{1-\gamma} \|\mu' - \mu\|_{1,\delta_\mu}. \tag{23}$$

Proof. We have

$$\begin{aligned}
d_{\mu'}^\top - d_\mu^\top &= (1-\gamma)\nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1} - (1-\gamma)\nu^\top[\mathbb{I} - \gamma P_\mu]^{-1} \\
&= (1-\gamma)\nu^\top[\mathbb{I} + \gamma(\mathbb{I} - \gamma P_{\mu'})^{-1} P'_\mu] - (1-\gamma)\nu^\top[\mathbb{I} + \gamma(\mathbb{I} - \gamma P_\mu)^{-1} P_\mu] \\
&= \gamma[d_{\mu'}^\top P_{\mu'} - d_\mu^\top P_\mu] \\
&= \gamma[d_{\mu'}^\top - d_\mu^\top] P_{\mu'} + \gamma d_\mu^\top [P_{\mu'} - P_\mu] \\
&= \gamma d_\mu^\top [P_{\mu'} - P_\mu][\mathbb{I} - \gamma P_{\mu'}]^{-1}.
\end{aligned}$$

Hence, it follows that

$$\begin{aligned}\|d_{\mu'} - d_\mu\|_1 &= \|d_{\mu'}^\top - d_\mu^\top\|_\infty \\ &\leq \gamma \|d_\mu^\top [P'_\mu - P_\mu]\|_\infty \ \|[\mathbb{I} - \gamma P_\mu]^{-1}\|_\infty \\ &= \frac{\gamma}{1-\gamma} \|d_\mu^\top [P'_\mu - P_\mu]\|_\infty, \end{aligned} \tag{24}$$

where the last inequality follows from the fact that $\|(\mathbb{I} - \gamma P_\mu)^{-1}\|_\infty \leq \sum_{k=0}^{\infty} \gamma^k \|(P_\mu)^k\|_\infty = 1/(1-\gamma)$.

Now,

$$\begin{aligned}\|d_\mu^\top [P'_\mu - P_\mu]\|_\infty &= \sum_{s',a'} \left| \sum_{s,a} d_\mu(s,a) P(s'|s,a)[\mu'(a'|s') - \mu(a'|s')] \right| \\ &\leq \sum_{s',a'} \sum_{s,a} d_\mu(s,a) P(s'|s,a) \left| \mu'(a'|s') - \mu(a'|s') \right| \\ &= \sum_{s,a,s'} d_\mu(s,a) P(s'|s,a) \sum_{a'} \left| \mu'(a'|s') - \mu(a'|s') \right| \\ &\leq \sum_{s,a,s'} d_\mu(s,a) P(s'|s,a) \|\mu'(\cdot|s') - \mu(\cdot|s')\|_1 \\ &= \|\mu' - \mu\|_{1,\delta_\mu}. \end{aligned} \tag{25}$$

where δ_μ is defined as in Section 3.3. The desired result now follows from (24) and (25). □

Proof of Proposition 3.6. Starting from Lemma 3.4, we get

$$\frac{1-\gamma}{\gamma}(\nu^\top Q_{\mu'} - \nu^\top f) \geq d_{\mu'}^\top a_\mu^{\mu'}(f) + \frac{1-\gamma}{\gamma} \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]. \tag{26}$$

Now, for any μ_b, observe that

$$\begin{aligned} d_{\mu'}^\top a_\mu^{\mu'}(f) &= d_{\mu_b}^\top a_\mu^{\mu'}(f) + (d_{\mu'}^\top - d_{\mu_b}^\top) a_\mu^{\mu'}(f) \\ &\overset{(a)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \left|(d_{\mu'} - d_{\mu_b})^\top a_\mu^{\mu'}(f)\right| \\ &\overset{(b)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \|(d_{\mu'} - d_{\mu_b})^\top\|_1 \frac{\mathrm{sp}(a_\mu^{\mu'}(f))}{2} \\ &\overset{(c)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \frac{\gamma}{1-\gamma} \|\mu' - \mu_b\|_{1,\delta_{\mu_b}} \frac{\mathrm{sp}(a_\mu^{\mu'}(f))}{2}, \end{aligned} \tag{27}$$

where (a) follows from the fact that $x + y \geq x - |y|$ for any real numbers x and y, (b) follows from (Haviv and Van der Heyden, 1984, Corollary 2.4), and (c) follows from Lemma A.2.

Separately, for $\mu_b = \mu$ and $\mu' = \alpha\bar{\mu} + (1-\alpha)\mu$, observe that $P_{\mu'} = \alpha P_{\bar{\mu}} + (1-\alpha) P_\mu$ and, hence,

$$\begin{aligned} d_{\mu_b}^\top a_\mu^{\mu'}(f) &= \alpha d_\mu^\top [P_{\bar{\mu}} - P_\mu] f = \alpha A_\mu^{\bar{\mu}}(f) \\ \|\mu'(\cdot|s) - \mu(\cdot|s)\|_1 &= \alpha \|\bar{\mu}(\cdot|s) - \mu(\cdot|s)\|_1 \\ \mathrm{sp}(a_\mu^{\mu'}(f)) &= \alpha \ \mathrm{sp}(a_\mu^{\bar{\mu}}(f)). \end{aligned} \tag{28}$$

Hence, by combining (26), (27), and (28), it follows that

$$\frac{1-\gamma}{\gamma}(\nu^\top Q_{\mu'} - \nu^\top f) \geq \alpha A_\mu^{\bar{\mu}}(f) - \frac{\gamma}{2(1-\gamma)} \alpha^2 \|\bar{\mu} - \mu\|_{1,\delta_\mu} \mathrm{sp}(a_\mu^{\bar{\mu}}(f)) + \frac{1-\gamma}{\gamma} \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]. \tag{29}$$

By substituting $H = 0$ and 1, the desired results are now easy to see. □

We next prove Theorems 3.8 and 3.9.

Proof of Theorem 3.8. For $\alpha \in [0,1]$, $\mu' = \alpha\bar{\mu} + (1-\alpha)\mu$. Since $T_\mu f \geq f$, Proposition 3.6 shows that

$$\nu^\top Q_{\mu'_\alpha} - \nu^\top f \geq \Psi_1(\alpha) \geq \Psi_0(\alpha)$$

for every $\alpha \in [0,1]$. Further, Ψ_1 and Ψ_0, being concave quadratic functions of α, attain their maxima at

$$\alpha_1^* = \frac{(1-\gamma)A_\mu^{\bar{\mu}}(f) + (1-\gamma)^2\nu^\top a_\mu^{\bar{\mu}}(T_\mu f - f)}{\gamma\|\bar{\mu} - \mu\|_{1,\delta_\mu}\, \mathrm{sp}(a_\mu^{\bar{\mu}}(f))} \quad \text{and} \quad \alpha_0^* = \frac{(1-\gamma)A_\mu^{\bar{\mu}}(f)}{\gamma\|\bar{\mu} - \mu\|_{1,\delta_\mu}\, \mathrm{sp}(a_\mu^{\bar{\mu}}(f))},$$

respectively. Since $\bar{\mu}$ is greedy with respect to f, we have $A_\mu^{\bar{\mu}}(f) \geq 0$, and hence $\alpha_0^* \geq 0$.

The rest of the proof reduces to locating these maximizers relative to the admissible interval $[0,1]$. Recall from (13) that the theorem chooses

$$\alpha = \min\{1, \alpha^*\}, \qquad \text{where} \qquad \alpha^* = \begin{cases} \alpha_1^*, & \text{if } \alpha_1^* > 0, \\ \alpha_0^*, & \text{otherwise.} \end{cases}$$

We now consider the four possible cases.

1. If $\alpha_1^* > 1$, then the maximizer of Ψ_1 lies to the right of 1. Hence, Ψ_1 is increasing on $[0,1]$, and the best admissible choice is the full greedy step $\alpha = 1$, which is exactly the coefficient selected by the theorem. Therefore,
$$\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_1(1).$$

2. If $\alpha_1^* \in [0,1]$, then the maximizer of Ψ_1 lies inside the admissible interval. Thus, the best admissible choice is the partial step $\alpha = \alpha_1^*$, and so
$$\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_1(\alpha_1^*).$$

3. If $\alpha_1^* < 0$ and $\alpha_0^* > 1$, then Ψ_1 is decreasing on $[0,1]$, and hence does not suggest a positive step. In this regime, we instead appeal to the weaker but still valid lower bound Ψ_0. Since its maximizer lies to the right of 1, Ψ_0 is increasing on $[0,1]$, so the best admissible choice is again $\alpha = 1$, which is what the theorem selects. Therefore,
$$\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_0(1).$$

4. If $\alpha_1^* < 0$ and $\alpha_0^* \leq 1$, then, as in the previous case, we work with Ψ_0. Now its maximizer lies in $[0,1]$, so the best admissible choice is the partial step $\alpha = \alpha_0^*$. Hence,
$$\nu^\top Q_{\mu'} - \nu^\top f \geq \Psi_0(\alpha_0^*).$$

This completes the proof. □

Proof of Theorem 3.9. As argued in Section 4, we already know that

$$T_{\mu_k} f_k \geq f_k \qquad \forall k \geq 0.$$

Consequently, the first three conclusions of Theorem 3.1 hold. It remains to prove the fourth one.

Let $f_\infty := \lim_{k\to\infty} f_k$, and let μ_∞ be an arbitrary subsequential limit of (μ_k); that is, for some subsequence (k_j), we have $\mu_{k_j} \to \mu_\infty$. Since $f_k \to f_\infty$, we also have $f_{k_j} \to f_\infty$. Moreover, $T_{\mu_{k_j}} f_{k_j} \geq f_{k_j}$, for all $j \geq 0$. Passing to the limit gives $T_{\mu_\infty} f_\infty \geq f_\infty$.

We now show that equality must hold in at least one coordinate. Suppose, for contradiction, that

$$T_{\mu_\infty} f_\infty > f_\infty \qquad \text{componentwise.} \tag{30}$$

By continuity of T_{μ_∞} and the room-to-improve assumption, it follows by arguing as in the proof of Theorem 3.1 that there exists $f \in \mathcal{F}$ such that $f > f_\infty$ and $T_{\mu_\infty} f > f$. Since $\mu_{k_j} \to \mu_\infty$ and f is fixed, continuity of $T_\mu f$ in μ

implies that, for all sufficiently large j, $T_{\mu_{k_j}} f > f$. Also, because $f_k \uparrow f_\infty$ componentwise and $f > f_\infty$, we have $f > f_{k_j}$ for all large j. Thus, for such j, the vector f is feasible for the CRPI evaluation step at iteration k_j.

Let f_{k_j+1} be the optimizer returned at iteration k_j. By optimality,

$$\|f_{k_j+1} - f_{k_j}\| \geq \|f - f_{k_j}\|. \tag{31}$$

On the other hand, since $f_{k_j} \leq f_{k_j+1} \leq f_\infty$, we have

$$0 \leq f_{k_j+1} - f_{k_j} \leq f_\infty - f_{k_j}.$$

Further, $f > f_\infty$ implies

$$f - f_{k_j} > f_\infty - f_{k_j} \geq 0.$$

Hence, by the monotonicity property of the norm,

$$\|f - f_{k_j}\| > \|f_\infty - f_{k_j}\| \quad \text{and} \quad \|f_{k_j+1} - f_{k_j}\| \leq \|f_\infty - f_{k_j}\|. \tag{32}$$

Combining the inequalities in (31) and (32) yields

$$\|f_{k_j+1} - f_{k_j}\| \geq \|f - f_{k_j}\| > \|f_\infty - f_{k_j}\| \geq \|f_{k_j+1} - f_{k_j}\|,$$

a contradiction.

Therefore, (30) is impossible. This completes the proof. □

We now prove Theorem 3.12. We begin with a simple inequality.

Lemma A.3. *Let* $[t]_+ := \max\{t, 0\}$. *Then, for any* $x \geq 0$ *and* $c \geq 0$,

$$[x - c]_+^2 \;\geq\; \frac{x^2}{4} - c^2.$$

Proof. We consider two cases.

Case 1: $0 \leq x \leq c$**.** In this case, $[x - c]_+ = 0$. Since $0 \leq x \leq c$, we have $x^2/4 \leq c^2$, and hence

$$[x - c]_+^2 = 0 \;\geq\; \frac{x^2}{4} - c^2.$$

Case 2: $x > c$**.** Here, $[x - c]_+ = x - c$. A direct calculation shows that

$$(x - c)^2 - \left(\frac{x^2}{4} - c^2\right) = \frac{3}{4}x^2 - 2cx + 2c^2 = \frac{3}{4}\left(x - \frac{4c}{3}\right)^2 + \frac{2}{3}c^2 \;\geq\; 0.$$

Thus, $(x - c)^2 \geq \frac{x^2}{4} - c^2$.

Combining the two cases completes the proof. □

We now show a relation between the relative advantage function estimate and performance difference between the two policies.

Lemma A.4. *Let* μ *and* μ' *be arbitrary policies. Further, let* f *satisfy* $T_\mu f \geq f$, *and let* $\bar{\mu}$ *be greedy with respect to* f. *Define*

$$\beta(\mu, \mu') := \min_{(s,a):\, d_{\mu'}(s,a) > 0} \frac{d_\mu(s,a)}{d_{\mu'}(s,a)}. \tag{33}$$

Then,

$$A_\mu^{\bar{\mu}}(f) \geq \beta(\mu, \mu') \frac{1 - \gamma}{\gamma} \Big[\nu^\top Q_{\mu'} - \nu^\top f - \nu^\top [I - \gamma P_{\mu'}]^{-1} [T_\mu f - f]\Big]. \tag{34}$$

Moreover, if $\|Q_\mu - f\|_\infty \leq \epsilon$, *then*

$$A_\mu^{\bar{\mu}}(f) \geq \beta(\mu, \mu') \frac{1 - \gamma}{\gamma} \Big[\nu^\top Q_{\mu'} - \nu^\top Q_\mu - \frac{\epsilon}{1 - \gamma}\Big]. \tag{35}$$

Proof. The relation in (34) follows by observing that

$$\begin{aligned}
A_\mu^{\bar\mu}(f) &\overset{(a)}{=} \sum_{s,a} d_\mu(s,a) \sum_{s',a'} P(s'|s,a)\bar\mu(a'|s')\Big(f(s',a') - \langle \mu(\cdot|s'),\, f(s',\cdot)\rangle\Big) \\
&\overset{(b)}{=} \sum_{s,a} d_\mu(s,a) \sum_{s'} P(s'|s,a)\max_{a'}\Big(f(s',a') - \langle \mu(\cdot|s'),\, f(s',\cdot)\rangle\Big) \\
&\overset{(c)}{\geq} \beta(\mu,\mu') \sum_{s,a} d_{\mu'}(s,a) \sum_{s'} P(s'|s,a)\max_{a'}\Big(f(s',a') - \langle \mu(\cdot|s'),\, f(s',\cdot)\rangle\Big) \\
&\overset{(d)}{\geq} \beta(\mu,\mu') \sum_{s,a} d_{\mu'}(s,a) \sum_{s',a'} P(s'|s,a)\mu'(a'|s')\Big(f(s',a') - \langle \mu(\cdot|s'),\, f(s',\cdot)\rangle\Big) \\
&\overset{(e)}{=} \beta(\mu,\mu') d_{\mu'}^\top a_\mu^{\mu'}(f) \\
&\overset{(f)}{=} \beta(\mu,\mu') \frac{1-\gamma}{\gamma}\Big[\nu^\top Q_{\mu'} - \nu^\top f - \nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f]\Big],
\end{aligned}$$

where (a) follows from the definition of $A_\mu^{\bar\mu}(f)$, (b) follows since $\bar\mu$ is greedy with respect to f, (c) follows from the fact that the inner sum is non-negative, (d) holds since the maximum is larger than any average, (e) follows from the definition of $a_\mu^{\mu'}(f)$, while (f) follows from (16).

The relation in (35) follows by additionally observing that

$$\begin{aligned}
\nu^\top Q_{\mu'} - &\nu^\top f - \nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f] \\
&\overset{(a)}{=} \nu^\top Q_{\mu'} - \nu^\top Q_\mu + \nu^\top[Q_\mu - f] - \nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f] \\
&\overset{(b)}{\geq} \nu^\top Q_{\mu'} - \nu^\top Q_\mu - \nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f] \\
&\overset{(c)}{\geq} \nu^\top Q_{\mu'} - \nu^\top Q_\mu - \frac{\epsilon}{1-\gamma},
\end{aligned}$$

where (a) follows by adding and subtracting $\nu^\top Q_\mu$, (b) follows since $Q_\mu \geq f$, while (c) holds since $(1-\gamma)\nu^\top[\mathbb{I} - \gamma P_{\mu'}]^{-1}$ is a distribution, and since $\|Q_\mu - f\|_\infty \leq \epsilon$ implies $Q_\mu(s,a) - \epsilon \leq f(s,a) \leq T_\mu f(s,a) \leq Q_\mu(s,a)$ and, hence, $T_\mu f(s,a) - f(s,a) \leq \epsilon$.

The desired results now follow. ☐

Lemma A.5. *Let $R_{\max} = \max_{s,a} r(s,a)$. Also, let f and μ be such that $0 \leq f \leq Q_\mu$. Finally, let $\bar\mu$ be the greedy policy with respect to f and μ' be the mixture policy $\tilde\alpha\bar\mu + (1-\tilde\alpha)\mu$, where*

$$\tilde\alpha = \frac{(1-\gamma)^2 A_\mu^{\bar\mu}(f)}{2R_{\max}}.$$

Then, $0 \leq \tilde\alpha \leq 1$, and

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \frac{(1-\gamma)\gamma\Big(A_\mu^{\bar\mu}(f)\Big)^2}{4R_{\max}}.$$

Proof. First, let $\alpha \in [0,1]$ be arbitrary, and let $\mu' = \alpha\bar\mu + (1-\alpha)\mu$. Then,

$$\nu^\top Q_{\mu'} - \nu^\top f \overset{(a)}{\geq} -\frac{\alpha^2\gamma^2}{2(1-\gamma)^2}\|\bar\mu - \mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar\mu}(f)) + \frac{\alpha\gamma}{1-\gamma}A_\mu^{\bar\mu}(f) + \nu^\top[T_\mu f - f] \tag{36}$$

$$\overset{(b)}{\geq} -\frac{\alpha^2\gamma^2}{2(1-\gamma)^2}\|\bar\mu - \mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar\mu}(f)) + \frac{\alpha\gamma}{1-\gamma}A_\mu^{\bar\mu}(f), \tag{37}$$

where (a) follows from (10), while (b) follows since $T_\mu f \geq f$.

We now simplify the expression in (37). For any policies μ, μ', and any state $s \in \mathcal{S}$,

$$\|\mu'(\cdot|s) - \mu(\cdot|s)\|_1 \leq 2 \tag{38}$$

and, hence,

$$\|\mu' - \mu\|_{1,\delta_\mu} \leq 2. \tag{39}$$

Next, we show that, for f and μ such that $0 \leq f \leq Q_\mu$ and $\bar{\mu}$ greedy with respect to f, we have

$$\mathrm{sp}\big(a_\mu^{\bar{\mu}}(f)\big) \leq \frac{R_{\max}}{1-\gamma}. \tag{40}$$

Clearly, for any $0 \leq f$, μ, and $s \in \mathcal{S}$,

$$\max f(s,a) - \langle \mu(\cdot|s), f(s,\cdot)\rangle \in [0, \mathrm{sp}(f)]. \tag{41}$$

Now, since $\bar{\mu}$ is greedy with respect to f, it follows from the definition given in (6) that

$$a_\mu^{\bar{\mu}}(f)(s,a) \in [0, \mathrm{sp}(f)]$$

for every state-action pair (s,a). Therefore,

$$\mathrm{sp}(a_\mu^{\bar{\mu}}(f)) \leq \mathrm{sp}(f) \leq \|f\|_\infty,$$

where the last inequality uses the fact that $0 \leq f$. Finally, since $0 \leq f \leq Q_\mu$ and $\|Q_\mu\|_\infty \leq \dfrac{R_{\max}}{1-\gamma}$, we obtain

$$\|f\|_\infty \leq \frac{R_{\max}}{1-\gamma}.$$

Using this relation in the previous display yields (40).

By substituting (39) and (40) in (37), we then get

$$\nu^\top Q_{\mu'} - \nu^\top f \geq -\frac{\alpha^2\gamma^2 R_{\max}}{(1-\gamma)^3} + \frac{\alpha\gamma}{1-\gamma} A_\mu^{\bar{\mu}}(f). \tag{42}$$

While the expression on the RHS is maximized at

$$\alpha = \frac{(1-\gamma)^2 A_\mu^{\bar{\mu}}(f)}{2\gamma R_{\max}},$$

this value can potentially be larger than 1. Hence, we pick the value stated in the statement, i.e.,

$$\tilde{\alpha} = \frac{(1-\gamma)^2 A_\mu^{\bar{\mu}}(f)}{2R_{\max}} \leq \frac{1-\gamma}{2} \leq 1.$$

Substituting this value of $\tilde{\alpha}$ in (42) then gives

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \frac{(1-\gamma)\gamma(2-\gamma)\Big(A_\mu^{\bar{\mu}}(f)\Big)^2}{4R_{\max}}.$$

Since $2 - \gamma \geq 1$, we get the desired result. □

Proof of Theorem 3.12. We first claim that, for every iteration $k \geq 0$,

$$Q_{\mu_k} \leq f_{k+1} + \epsilon\mathbf{1} \leq Q_{\mu_k} + \epsilon\mathbf{1} \leq Q^\star + \epsilon\mathbf{1}, \tag{43}$$

where $\mathbf{1}$ is the all-ones vector. The last inequality holds since $Q_\mu \leq Q^\star$ for any policy μ, and the middle inequality follows from $f_{k+1} \leq Q_{\mu_k}$ by Theorem 3.9.

It remains to verify the first inequality. Let $\hat{f} := f^{\max}(\mu_k, f_k)$ denote the greatest element of $\mathcal{H}(\mu_k, f_k)$. By assumption,

$$Q_{\mu_k} - \epsilon\mathbf{1} \leq \hat{f} \leq Q_{\mu_k}.$$

Moreover, $\hat{f} \in \mathcal{H}(\mu_k, f_k)$, so $T_{\mu_k}\hat{f} \geq \hat{f} \geq f_k$.

For any $f \in \mathcal{H}(\mu_k, f_k)$, we have $f \leq \hat{f}$, and hence

$$0 \leq f - f_k \leq \hat{f} - f_k.$$

If $f \neq \hat{f}$, the inequality is strict in at least one coordinate. By strict monotonicity of $\|\cdot\|$,

$$\|f - f_k\| < \|\hat{f} - f_k\|.$$

Thus, $\hat{f}$ is the unique maximizer of $\|f - f_k\|$ over $\mathcal{H}(\mu_k, f_k)$, and hence $f_{k+1} = \hat{f}$. Therefore,

$$f_{k+1} \geq Q_{\mu_k} - \epsilon\mathbf{1},$$

which proves the claim.

We now establish a recurrence for $g_k := \nu^\top(Q^\star - Q_{\mu_k})$. We show that

$$g_{k+1} \leq g_k - \frac{g_k^2}{C_1} + C_2\epsilon^2 + \epsilon, \tag{44}$$

where $C_1 = \frac{16R_{\max}\gamma}{(1-\gamma)^3\Delta^2}$ and $C_2 = \frac{(1-\gamma)\Delta^2}{4R_{\max}\gamma}$.

To this end, we first show that

$$\left(A_{\mu_k}^{\bar{\mu}_k}(f_{k+1})\right)^2 \geq C_3 g_k^2 - C_4\epsilon^2, \tag{45}$$

where $C_3 = \frac{1}{4}\left(\frac{(1-\gamma)\Delta}{\gamma}\right)^2$, and $C_4 = \frac{\Delta^2}{\gamma^2}$.

We start with (45). Recall that $[x]_+ := \max\{x, 0\}$. Then,

$$\begin{aligned}
A_{\mu_k}^{\bar{\mu}_k}(f_{k+1}) &\overset{(a)}{\geq} \frac{(1-\gamma)\beta(\mu_k, \mu^\star)}{\gamma}\left(\nu^\top Q_{\mu^\star} - \nu^\top Q_{\mu_k} - \frac{\epsilon}{1-\gamma}\right) \\
&\overset{(b)}{\geq} \frac{(1-\gamma)\Delta}{\gamma}\left(\nu^\top Q_{\mu^\star} - \nu^\top Q_{\mu_k} - \frac{\epsilon}{1-\gamma}\right) \\
&\overset{(c)}{\geq} \frac{(1-\gamma)\,\Delta}{\gamma}\left(g_k - \frac{\epsilon}{1-\gamma}\right) \\
&\overset{(d)}{\geq} \frac{(1-\gamma)\,\Delta}{\gamma}\left[g_k - \frac{\epsilon}{1-\gamma}\right]_+,
\end{aligned}$$

where (a) follows from (35), (b) follows since $d_{\mu^\star}(s,a) \leq 1$ and since Assumption 3.11 implies $d_{\mu_k}(s,a) \geq \Delta$ whenever $d_{\mu^\star}(s,a) > 0$, (c) follows from the definition of g_k, while (d) follows since $\bar{\mu}_k$ is greedy with respect to f_{k+1}, and thus $A_{\mu_k}^{\bar{\mu}_k}(f_{k+1}) \geq 0$. Therefore, it follows that

$$\begin{aligned}
\left(A_{\mu_k}^{\bar{\mu}_k}(f_{k+1})\right)^2 &\geq \left(\frac{(1-\gamma)\,\Delta}{\gamma}\right)^2\left[g_k - \frac{\epsilon}{1-\gamma}\right]_+^2 \\
&\overset{(a)}{\geq} \frac{1}{4}\left(\frac{(1-\gamma)\,\Delta}{\gamma}\right)^2 g_k^2 - \left(\frac{(1-\gamma)\,\Delta}{\gamma}\right)^2\left(\frac{\epsilon}{1-\gamma}\right)^2,
\end{aligned}$$

where (a) follows from Lemma A.3. This proves (45).

Next, using the stepsize choice and Lemma A.5, we get

$$\begin{aligned}
\nu^\top Q_{\mu_{k+1}} - \nu^\top f_{k+1} &\geq \frac{(1-\gamma)\gamma}{4R_{\max}}\left(A_{\mu_k}^{\bar{\mu}_k}(f_{k+1})\right)^2 \\
&\geq \frac{g_k^2}{C_1} - C_2\epsilon^2.
\end{aligned} \tag{46}$$

Now,

$$\begin{aligned} g_{k+1} &= \nu^\top Q^\star - \nu^\top Q_{\mu_{k+1}} \\ &= g_k - (\nu^\top Q_{\mu_{k+1}} - \nu^\top Q_{\mu_k}) \\ &\leq g_k - (\nu^\top Q_{\mu_{k+1}} - \nu^\top f_{k+1}) + \epsilon \\ &\leq g_k - \frac{g_k^2}{C_1} + C_2\epsilon^2 + \epsilon, \end{aligned}$$

where (a) follows (43), while (b) follows from (46). This establishes (44).

Let $b := C_2\epsilon^2 + \epsilon$ and define

$$\delta' := \sqrt{2C_1 b}, \qquad \delta := \delta' + b.$$

Step 1: Decay outside the δ'-region. If $g_k \geq \delta'$, then $b \leq g_k^2/(2C_1)$, and hence

$$g_{k+1} \leq g_k - \frac{g_k^2}{2C_1}.$$

This implies $g_{k+1} \leq g_k$, and

$$\frac{1}{g_{k+1}} - \frac{1}{g_k} \geq \frac{1}{2C_1}.$$

Summing yields

$$g_k \leq \frac{2C_1}{k}.$$

Step 2: Invariance of the δ-region. We show that $[0, \delta]$ is invariant.

If $g_k < \delta'$, then

$$g_{k+1} \leq g_k + b < \delta' + b = \delta.$$

If $\delta' \leq g_k \leq \delta$, then

$$g_{k+1} \leq g_k - \frac{g_k^2}{C_1} + b \leq g_k - \frac{g_k^2}{2C_1} \leq g_k \leq \delta.$$

Thus, in all cases,

$$g_k \leq \delta \;\Rightarrow\; g_{k+1} \leq \delta.$$

Step 3: Global bound. Let $\tau := \inf\{k \geq 0 : g_k < \delta'\}$.

If $\tau > N$, then $g_j \geq \delta'$ for all $j \leq N$, so

$$g_N \leq \frac{2C_1}{N}.$$

If $\tau \leq N$, then $g_\tau < \delta' < \delta$, and by invariance,

$$g_N \leq \delta.$$

Therefore, for all $N \geq 1$,

$$g_N \leq \max\left\{\frac{2C_1}{N}, \delta\right\}.$$

This completes the proof. □

B INITIALIZATION UNDER LINEAR FUNCTION APPROXIMATION

Our theoretical guarantees require the initial value estimate to satisfy

$$T_{\mu_0} f_0 \;\geq\; f_0.$$

In this section, we present a simple initialization procedure that ensures this condition under linear FA.

Suppose $\mathcal{F} = \{f_\theta : \theta \in \mathbb{R}^d\}$, where $f_\theta = \Phi\theta$ for a feature matrix $\Phi \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}| \times d}$. Then, for any (s, a),

$$f_\theta(s, a) = \phi(s, a)^\top \theta,$$

where $\phi(s, a) \in \mathbb{R}^d$ denotes the feature vector corresponding to (s, a). We assume that the first column of Φ is a bias feature, i.e.,

$$\Phi((s, a), 1) = 1 \quad \text{for all } (s, a).$$

Let $\hat{f}_0 = \Phi\hat{\theta}$ be any initial estimate. We construct a shifted parameter vector

$$\theta = \hat{\theta} + b e_1,$$

where $e_1 = (1, 0, \ldots, 0) \in \mathbb{R}^d$ and $b \in \mathbb{R}$ is a scalar. The corresponding function is

$$f_0 := \Phi\theta = \hat{f}_0 + b\mathbf{1},$$

where $\mathbf{1}$ denotes the all-ones vector.

We now choose b so that $T_{\mu_0} f_0 \geq f_0$. Using the affine property of the Bellman operator (Bertsekas and Tsitsiklis, 1996, Lemma 2.4), we have

$$T_{\mu_0}(\hat{f}_0 + b\mathbf{1})(s, a) = T_{\mu_0}\hat{f}_0(s, a) + \gamma b.$$

Thus, the condition $T_{\mu_0} f_0 \geq f_0$ is equivalent to

$$T_{\mu_0}\hat{f}_0(s, a) + \gamma b \;\geq\; \hat{f}_0(s, a) + b, \quad \forall (s, a).$$

Rearranging, we obtain

$$T_{\mu_0}\hat{f}_0(s, a) - \hat{f}_0(s, a) \;\geq\; (1 - \gamma) b.$$

Therefore, it suffices to choose

$$b \;\leq\; \min_{(s,a)} \frac{T_{\mu_0}\hat{f}_0(s, a) - \hat{f}_0(s, a)}{1 - \gamma}.$$

With this choice, the shifted initialization $f_0 = \hat{f}_0 + b\mathbf{1}$ satisfies

$$T_{\mu_0} f_0 \;\geq\; f_0,$$

as required.

C DEEP RL IMPLEMENTATION OF RPI

We now describe how the policy evaluation step of RPI can be implemented in a deep reinforcement learning setting. The resulting method replaces the constrained optimization in (1) with a stochastic surrogate objective that can be optimized using standard deep RL infrastructure.

Recall that the policy evaluation step of RPI computes

$$f_{k+1} \in \arg\max_{f \in \mathcal{F}} \|f - f_k\| \quad \text{s.t.} \quad T_{\mu_k} f \geq f \geq f_k.$$

The constraint $T_{\mu_k} f \geq f$ enforces $Q_{\mu_k} \geq f_{k+1}$, while $f \geq f_k$ ensures monotonic improvement of the value estimates.

Convex surrogate. We use a convex surrogate objective that minimizes the Bellman residual while retaining the same constraints, that is

$$\min_{f\in\mathcal{F}} \|T_{\mu_k} f - f\|_2^2 \quad \text{s.t.} \quad T_{\mu_k} f \geq f \geq f_k.$$

Intuitively, the original RPI objective maximizes the gap between the constraints $f \geq f_k$ while being feasible. In contrast, the surrogate objective minimizes the gap between $T_{\mu_k} f \geq f$ while being feasible.

Lagrangian relaxation. We now focus on the problem with only the one sided Bellman constraint. We enforce the Bellman inequality constraint $f - T_\mu f \leq 0$ using a Lagrangian formulation. The resulting objective takes the form

$$L(f, \lambda) = \|T_\mu f - f\|_2^2 + \lambda^\top (f - T_\mu f), \quad \lambda \geq 0. \tag{47}$$

In large state–action spaces maintaining a separate multiplier for every constraint is impractical. Instead, we introduce a shared scalar multiplier λ_1 and approximate the constraint penalty using the ReLU function $[x]_+ = \max(x, 0)$. This yields the penalty approximation

$$\tilde{L}(f, \lambda) = \|T_\mu f - f\|_2^2 + \lambda_1 \mathbf{1}^\top [f - T_\mu f]_+,$$

where $\mathbf{1}$ is an all-ones vector.

The resulting objective closely resembles standard critic losses used in deep reinforcement learning, with the addition of the penalty term enforcing the RPI constraint. As in off-policy deep RL algorithms such as DDPG, we use sampled $\mathcal{B}-$sized minibatch of transitions (s_i, a_i, r_i, s'_i) from the replay buffer to approximate the Bellman operator using the target $y_i = r_i + \gamma Q_{\text{target}}(s'_i, a'_i)$. Substituting this into the relaxed objective yields the critic loss

$$\mathcal{L}_{\text{critic}}^{\text{RPI}} = \frac{1}{|\mathcal{B}|} \sum_i \Big((Q_{\text{curr}}(s_i, a_i) - y_i)^2 + \lambda_1 [Q_{\text{curr}}(s_i, a_i) - y_i]_+ \Big).$$

Importantly, this loss is very similar to existing deep RL critic losses, making it straightforward to incorporate into standard algorithms.

Adaptive penalty update. The multiplier λ_1 controls the strength of constraint enforcement. In the true Lagrangian formulation (47), the dual variable would be updated using gradient ascent as

$$\lambda^{t+1} = \left[\lambda^t + \eta(f - T_\mu f)\right]_+ .$$

This update increases the multiplier when the constraint is violated and decreases it when the constraint is satisfied. Maintaining such updates for every state–action pair is infeasible in deep RL. Instead, we adopt a computationally efficient heuristic that mimics the qualitative behavior of dual ascent.

Specifically, we monitor the average constraint satisfaction rate $\bar{s}$ computed over a sliding window of recent training iterations. Let s_{target} denote a desired target satisfaction rate. The shared multiplier is then updated as

$$\lambda_1 \leftarrow \begin{cases} \lambda_1 \alpha_\uparrow & \text{if } \bar{s} < s_{\text{target}} \\ \lambda_1 \alpha_\downarrow & \text{otherwise,} \end{cases}$$

where $\alpha_\uparrow > 1$ and $\alpha_\downarrow < 1$ are scale-up and scale-down hyperparameters controlling how aggressively the penalty is adjusted. Finally, we clamp the multiplier to a reasonable range $\lambda_1 \in [\lambda_{\min}, \lambda_{\max}]$, where $\lambda_{\min} \geq 0$.

Integration with deep RL algorithms. The proposed critic loss can be directly integrated into standard deep RL pipelines. In particular, replacing the critic loss in algorithms such as DDPG with $\mathcal{L}_{\text{critic}}^{\text{RPI}}$ produces a modified algorithm that enforces the RPI Bellman inequality during critic training. We refer to this variant as RPI_{DDPG}. This modification requires no changes to the actor update or the overall training pipeline, making the proposed method easy to incorporate into existing deep RL implementations.

Implementation details. We implement RPI_{DDPG} by modifying the publicly available DDPG implementation from the repository accompanying (Fujimoto et al., 2018).[1] Specifically, we replace the standard critic loss with $\mathcal{L}^{\text{RPI}}_{\text{critic}}$ while keeping all other components unchanged.

The additional hyperparameters introduced by our method are as follows: the initial penalty weight $\lambda_1 = 0.01$, window size $n = 100$, target constraint satisfaction rate $s_{\text{target}} = 0.75$, and λ_1 is updated every 10^4 iterations. Here, the window size denotes the number of recent iterations over which the average constraint satisfaction is computed. The penalty weight is adapted multiplicatively with scale factors 2 and 0.5, and is constrained to lie in the range $[10^{-3}, 10^2]$.

D ENVIRONMENT DETAILS

To evaluate the performance of our proposed algorithms, RPI and CRPI, we benchmark them against USPI, AMPI-Q, and CPI on two environments: Inventory Control and Chain Walk. The details of these environments are described below.

D.1 Inventory Control

In this work we consider a variant of the classical inventory control problem, modeled as a Markov Decision Process, where the objective is to maximize the expected long-term reward obtained from managing inventory in the presence of stochastic demand.

The inventory system has a fixed maximum capacity of M units. The state on day t, denoted by $s_t \in \mathcal{S} = \{0, 1, \dots, M\}$, represents the number of items currently in stock. The action space is $\mathcal{A} = \{0, 1, \dots, M\}$, where an action $a_t \in \mathcal{A}$ denotes the number of items ordered at the beginning of day t. Each day yields a reward composed of three components as follows

- **Procurement cost** for the items ordered,
- **Holding cost** for leftover inventory, and
- **Revenue** from items sold.

Let c denote the unit procurement cost, h the holding cost per unit of unsold inventory, and p the unit selling price. Let d_t denote the demand on day t, sampled from a predefined demand distribution. Define the post-order inventory level as

$$\hat{s}_t = \min(s_t + a_t, M),$$

which represents the total inventory available for sale on day t after ordering. The system evolves according to the transition rule

$$s_{t+1} = \max(\hat{s}_t - d_t, 0),$$

reflecting the remaining inventory after meeting demand. The immediate reward is given by

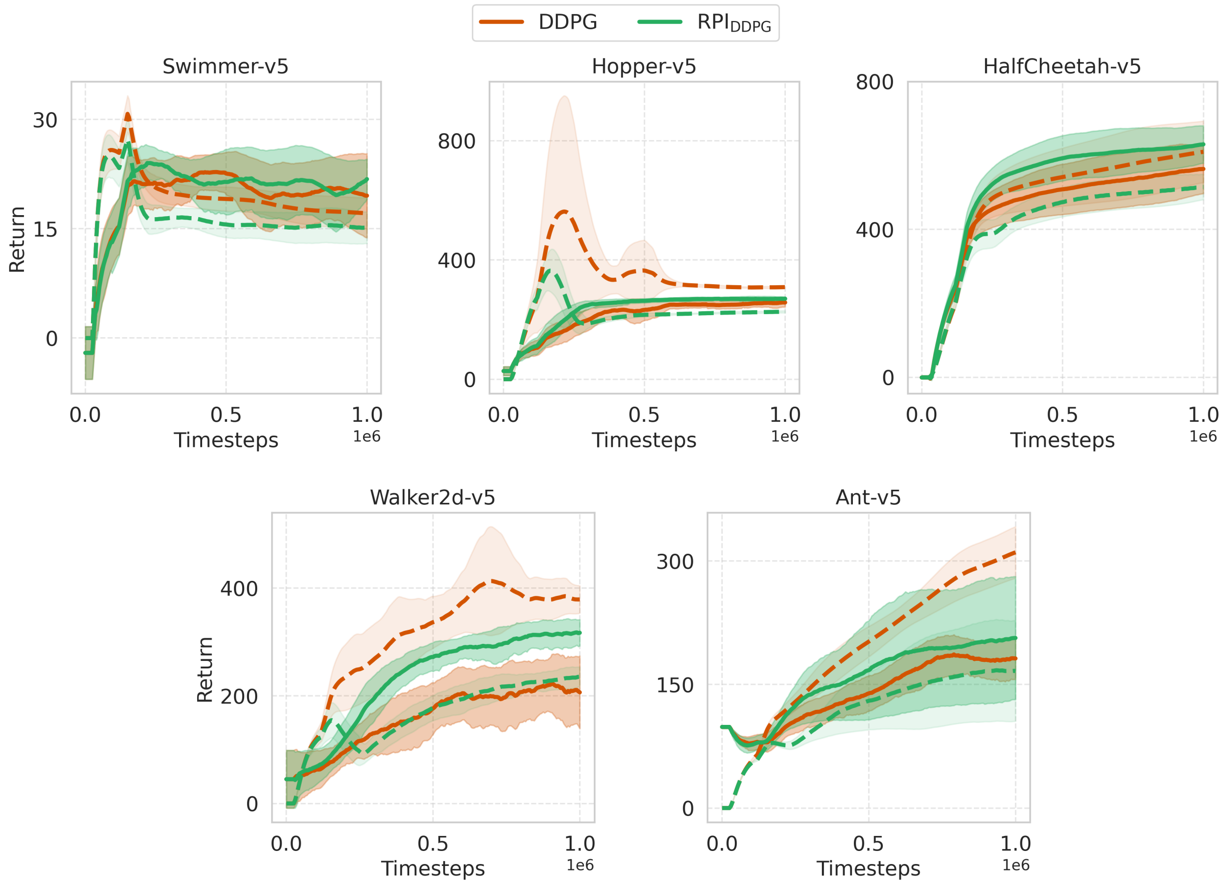
$$R(s_t, a_t, d_t) = p \cdot \min(\hat{s}_t, d_t) - c \cdot a_t - h \cdot \max(\hat{s}_t - d_t, 0),$$

where the first term captures revenue from sales, the second term penalizes procurement, and the third penalizes excess inventory.

D.2 Chain Walk

The chain walk environment is modelled as an N-state linear chain with states labeled 1 to N (with $N = 50$ in our experiments). At each step the agent chooses an action "Left" (L) or "Right" (R), moving in the intended direction with probability p and in the opposite direction with probability $1-p$ (with $p = 0.9$ in our experiments). A reward of $+1$ is granted only upon entering either of the two target states located $N/4$ steps from each end of the chain; all other transitions yield zero reward. No states are terminal, so the chain can be traversed indefinitely. The initial distribution is taken to be uniform over the state-action space.

[1] `https://github.com/sfujim/TD3`



Metric	Algorithm	Swimmer	Hopper	HalfCheetah	Walker2d	Ant
Terminal Performance	DDPG	19.4 ± 5.8	257.6 ± 9.7	564.8 ± 64.7	200.3 ± 34.1	182.8 ± 7.6
	RPI_{DDPG}	**22.3 ± 1.8**	**270.9 ± 2.0**	**632.0 ± 50.4**	**319.9 ± 17.1**	**208.2 ± 73.5**
AUC ($\times 10^6$)	DDPG	20.1 ± 2.7	219.3 ± 10.3	483.7 ± 58.2	165.7 ± 4.8	144.4 ± 8.4
	RPI_{DDPG}	**21.0 ± 2.6**	**241.5 ± 6.5**	**550.2 ± 48.2**	**245.0 ± 10.6**	**163.4 ± 50.5**

Figure 3: Performance comparison between DDPG and RPI_{DDPG} on MuJoCo environments. **Top:** Learning curves for Swimmer-v5, Hopper-v5, HalfCheetah-v5, Walker2d-v5, and Ant-v5 (mean ± 1 std; solid: return, dashed: critic estimate). **Bottom:** Terminal performance and AUC (mean ± std) across all environments. **Summary:** RPI_{DDPG} maintains lower-bound value estimates, while DDPG often overestimates. RPI_{DDPG} matches DDPG on simpler tasks and outperforms DDPG on harder environments.

E ADDITIONAL RESULTS ON MUJOCO ENVIRONMENTS

This section provides a consolidated view of all MuJoCo environments discussed in Section 5.2. In particular, we include results for Swimmer-v5 and HalfCheetah-v5, which were omitted from the main paper due to space constraints.

Figure 3 presents these results. The plots follow the same format as Figure 2. Solid curves denote Monte Carlo returns averaged over 25 trajectories, while dashed curves denote the critic estimates for the initial state–action pair, averaged over the same trajectories. All curves are averaged over five random seeds, with shaded regions indicating ±1 standard deviation.

The results are consistent with the observations in the main text. RPI_{DDPG} maintains value estimates that largely remain below the empirical returns after an initial transient phase, while DDPG exhibits noticeable overestimation in most environments. In simpler environments such as Swimmer-v5 and Hopper-v5, both methods

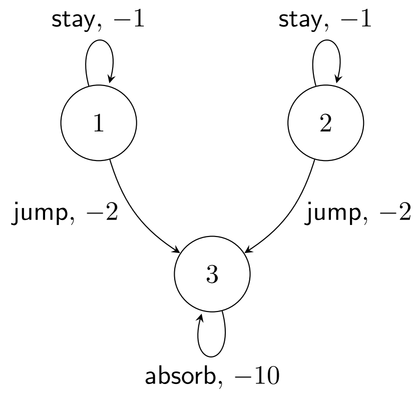


Figure 4: Three-state MDP illustrating stagnation vs. degradation.

achieve similar performance, while in the remaining environments $\mathrm{RPI}_{\mathrm{DDPG}}$ attains higher returns and learns faster. The summary table confirms this trend: $\mathrm{RPI}_{\mathrm{DDPG}}$ achieves higher terminal performance and larger AUC across all environments.

F COMPUTATIONAL RESOURCES AND SOLVERS

All experiments were run on two high-performance computing machines. The first machine is powered by an AMD EPYC 7763 64-Core Processor and has 1 TB of RAM, running Ubuntu 20.04. The second machine is equipped with an AMD EPYC 9554 64-Core Processor, 188 GB of RAM, and two NVIDIA GeForce RTX 5080 GPUs, running Ubuntu 22.04.

All optimization problems were solved using either the CVXPY (Diamond and Boyd, 2016; Agrawal et al., 2018) or Gurobi (Gurobi Optimization, LLC, 2024) Python libraries. We utilized an academic license for the Gurobi optimizer. The deep reinforcement learning experiments were implemented in PyTorch (Ansel et al., 2024), with GPU acceleration used where available.

G STAGNATION VS. DEGRADATION UNDER FUNCTION APPROXIMATION

Under FA, policy evaluation can fundamentally alter the behavior of policy iteration. Projection-based methods may introduce uncontrolled errors on unvisited state-action pairs, leading to policy degradation. In contrast, RPI enforces Bellman-consistent lower bounds, which prevent such degradation. However, this conservatism can restrict the set of feasible updates: even when the current estimate is feasible, there may be no strictly improving direction within the function class. In such cases, the algorithm stagnates. This section illustrates that such stagnation is not a failure, but a consequence of maintaining reliable value estimates under approximation.

MDP description. Consider an MDP with states $S = \{1, 2, 3\}$ and discount $\gamma = 0.9$. The initial-state distribution is uniform over states 1 and 2, i.e., $\nu = (0.5, 0.5, 0)$. In states 1 and 2, the actions are $\{\mathsf{stay}, \mathsf{jump}\}$, while state 3 has a single action absorb. The agent transitions and rewards are as follows.

- stay: reward -1, self-loop,
- jump: reward -2, transitions to state 3,
- absorb: reward -10, self-loop in state 3.

A schematic of this MDP is shown in Figure 4. Let π_0 (resp. π_1) choose stay (resp. jump) in states $1, 2$.

Exact values. We compute Q_{π_0} directly from the Bellman equations.

For the absorbing state,

$$Q_{\pi_0}(3, \mathsf{absorb}) = -10 + 0.9\, Q_{\pi_0}(3, \mathsf{absorb}),$$

which gives

$$Q_{\pi_0}(3, \mathsf{absorb}) = -100.$$

For states 1 and 2, under π_0 the action stay is always taken, so

$$Q_{\pi_0}(1, \mathsf{stay}) = -1 + 0.9\, Q_{\pi_0}(1, \mathsf{stay}), \quad Q_{\pi_0}(2, \mathsf{stay}) = -1 + 0.9\, Q_{\pi_0}(2, \mathsf{stay}),$$

which yields

$$Q_{\pi_0}(1, \mathsf{stay}) = Q_{\pi_0}(2, \mathsf{stay}) = -10.$$

For the action jump,

$$Q_{\pi_0}(1, \mathsf{jump}) = -2 + 0.9\, Q_{\pi_0}(3, \mathsf{absorb}) = -2 + 0.9(-100) = -92,$$

and similarly for state 2.

Thus,

$$Q_{\pi_0} = (-10, -92, -10, -92, -100)^\top,$$

with coordinates ordered as

$$(1, \mathsf{stay}),\ (1, \mathsf{jump}),\ (2, \mathsf{stay}),\ (2, \mathsf{jump}),\ (3, \mathsf{absorb}).$$

The expected return under the initial state-distribution $\nu = (1/2, 1/2, 0)$ is

$$J_\nu(\pi) := \mathbb{E}_{s\sim\nu, a\sim\pi(\cdot|s)}[Q_\pi(s, a)],$$

and hence

$$J_\nu(\pi_0) = \tfrac{1}{2}(-10) + \tfrac{1}{2}(-10) = -10.$$

Similarly, under π_1, both states transition immediately to state 3, giving

$$Q_{\pi_1}(1, \mathsf{jump}) = Q_{\pi_1}(2, \mathsf{jump}) = -92,$$

while

$$Q_{\pi_1}(3, \mathsf{absorb}) = -100.$$

Thus,

$$Q_{\pi_1} = (-10, -92, -10, -92, -100)^\top,$$

with the same coordinate ordering as above, and

$$J_\nu(\pi_1) = -92 < J_\nu(\pi_0).$$

Function class. We consider a linear FA class of the form

$$\mathcal{F} = \{\Phi\theta : \theta \in \mathbb{R}^2\},$$

where $\theta = (p, 1)^\top$ and the feature matrix $\Phi \in \mathbb{R}^{5\times 2}$ is given by

$$\Phi = \begin{bmatrix} 1 & 0 \\ 2 & 15 \\ 1 & 0 \\ 2 & 15 \\ 2 & 15 \end{bmatrix}.$$

Thus, for any $\theta = (p, 1)^\top$, we obtain

$$f(p) := \Phi\theta = (p,\ 2p + 15,\ p,\ 2p + 15,\ 2p + 15)^\top.$$

We initialize at

$$\theta_0 = (-57.5, 1)^\top, \quad f_0 = f(-57.5) = \Phi\theta_0 = (-57.5, -100, -57.5, -100, -100)^\top.$$

Failure of projected/TD-style updates. We now analyze the update produced by projected (on-policy) policy evaluation of π_0. Under π_0, only the state-action pairs $(1, \mathsf{stay})$ and $(2, \mathsf{stay})$ are visited. Thus, the projected Bellman update minimizes the mean-squared Bellman error over these visited coordinates:

$$\min_{p \in \mathbb{R}} \ \mathbb{E}_{(s,a)\sim d_{\pi_0}} \big(f(p)(s,a) - T_{\pi_0} f(p)(s,a)\big)^2.$$

Since d_{π_0} is supported only on $(1, \mathsf{stay})$ and $(2, \mathsf{stay})$, and both coordinates share the same value $f(p) = p$, this reduces to

$$\min_{p \in \mathbb{R}} \ \big(p - (-1 + 0.9p)\big)^2 = (0.1p + 1)^2.$$

The minimizer is obtained by setting the derivative to zero.

$$0.1p + 1 = 0 \ \Rightarrow \ p = -10.$$

Substituting this value,

$$f(-10) = (-10, -5, -10, -5, -5)^\top.$$

Thus, the projected update fits the Bellman equation exactly on the visited coordinates:

$$f(-10)(s, \mathsf{stay}) = -10 = Q_{\pi_0}(s, \mathsf{stay}),$$

but distorts the unvisited ones. In particular,

$$Q_{\pi_0}(1, \mathsf{jump}) = -92 \quad \text{while} \quad f(-10)(1, \mathsf{jump}) = -5.$$

Hence, the jump action is severely overestimated. Greedy improvement therefore selects π_1, even though $J_\nu(\pi_1) < J_\nu(\pi_0)$. This example shows that projected policy evaluation minimizes Bellman error only on the data distribution d_{π_0}, and can therefore produce misleading estimates on unvisited state-action pairs, leading to policy degradation.

Behavior of RPI. We now verify that f_0 satisfies the RPI feasibility condition. For each coordinate, we check that $T_{\pi_0} f_0 \geq f_0$.

At states 1 and 2, π_0 selects stay, so

$$T_{\pi_0} f_0(1, \mathsf{stay}) = -1 + 0.9\, f_0(1, \mathsf{stay}) = -1 + 0.9(-57.5) = -52.75 \ \geq \ -57.5,$$

and similarly for $(2, \mathsf{stay})$.

For the jump action,

$$T_{\pi_0} f_0(1, \mathsf{jump}) = -2 + 0.9\, f_0(3, \mathsf{absorb}) = -2 + 0.9(-100) = -92 \ \geq \ -100,$$

and similarly for $(2, \mathsf{jump})$.

Finally, at state 3,

$$T_{\pi_0} f_0(3, \mathsf{absorb}) = -10 + 0.9(-100) = -100,$$

which equals $f_0(3, \mathsf{absorb})$.

Thus, $T_{\pi_0} f_0 \geq f_0$ holds coordinate-wise. Moreover, f_0 correctly ranks actions at states 1 and 2:

$$f_0(s, \mathsf{stay}) = -57.5 \ > \ -100 = f_0(s, \mathsf{jump}),$$

so the greedy policy with respect to f_0 coincides with π_0.

We now show that f_0 is the *only* feasible solution. Consider any $f(p) \in \mathcal{F}$. The constraint $T_{\pi_0} f \geq f$ must hold for all coordinates.

Focusing on the absorbing coordinate,

$$T_{\pi_0} f(3, \mathsf{absorb}) = -10 + 0.9\,(2p + 15).$$

The feasibility condition requires

$$-10 + 0.9(2p + 15) \geq 2p + 15.$$

Rearranging,

$$-10 + 1.8p + 13.5 \geq 2p + 15 \iff 1.8p + 3.5 \geq 2p + 15 \iff p \leq -57.5.$$

On the other hand, the lower-bound constraint $f \geq f_0$ implies $p \geq -57.5$. Therefore, the only feasible value is $p = -57.5$, i.e.,

$$\{f \in \mathcal{F} : T_{\pi_0} f \geq f \geq f_0\} = \{f_0\}.$$

Thus, the RPI update admits no improving direction within $\mathcal{F}$, and the algorithm returns $f_1 = f_0$. The greedy policy remains π_0 in subsequent iterations, and the algorithm stagnates at this solution.

Takeaway. This example highlights a fundamental trade-off. When the function class poorly represents critical state-action pairs, unconstrained projection can produce overly optimistic estimates and lead to policy degradation. In contrast, RPI enforces lower-bound updates that prevent such degradation, but may stagnate when no feasible improving direction exists.

In general, intermediate policies generated by RPI may exhibit degradation in their true value. However, the learned estimates remain certified lower bounds, i.e., $f_k \leq Q_{\mu_k}$ for all k. Thus, even when performance fluctuates, the estimates provide a reliable signal that does not overestimate the true values. Such reliability is not guaranteed in projected Bellman-type evaluation methods, where the estimates can be arbitrarily optimistic.