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]Computer Science and Automation, Indian Institute of Science, Bengaluru, India
[2]NVIDIA Research, Israel
{eshwarsr, gthoppe, ananyabratab, aditya}@iisc.ac.in, galdl20@gmail.com

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, 2022). 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, each new policy's value function is guaranteed to 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

[†]Pre-doc supported by Walmart Centre for Tech Excellence at IISc.

Safe PI (SPI) (Metelli et al., 2021) currently lack FA variants that retain their per-step improvement guarantees. In approximate settings, PI often learns quickly but can stall or oscillate, whereas conservative variants such as CPI and SPI make more measured updates and empirically reach better policies in some domains (Metelli et al., 2021).

This raises two central questions: (i) Can PI be generalized beyond the tabular case without losing value-estimate monotonicity? (ii) Can CPI and SPI be extended to FA while retaining tabular-style performance guarantees? Because CPI underpins TRPO (Schulman et al., 2015a) and PPO (Schulman et al., 2017)—the modern RL workhorses—answering these questions is crucial for robust, scalable RL.

In this work, we develop Reliable Policy Iteration (RPI) and its conservative variant (CRPI) to answer both these questions in the affirmative. Our work's highlights can be summarized as follows:

1. **Algorithm:** We introduce RPI (Algorithm 1) and CRPI (Algorithm 2). Both adopt a novel policy-evaluation principle: compute a value estimate that is *farthest* from the previous estimate 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—which are tailored for the tabular setting—our methods apply under general FA.

2. **RPI Theoretical Guarantees:** We show that RPI recovers the classical PI guarantees of per-step improvement and convergence under arbitrary FA (Section 3.2)—where these properties, so far, were out of reach. Specifically, we prove that RPI's value estimates are non-decreasing and lower bound the true policy values. Moreover, RPI's value estimates 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 policy. Finally, we establish RPI as a true generalization of classical PI and show that, with an ℓ_1-type norm, its policy evaluation step admits a constrained projection interpretation.

3. **CRPI Theoretical Guarantees**: We derive a generalization of the performance-difference lemma to arbitrary FA (Section 3.3), that (i) enables working with FA-estimates of the true advantage function and (ii) incorporates errors arising due to FA. We show that CRPI's policy update maximizes a lower bound on this performance gap, leading to the first per-step improvement guarantees under FA.

4. **Simulations**: In inventory control (dense reward), RPI outperforms approximate variants of PI, CPI, and SPI, both in learning speed and terminal policy quality. In contrast, in chain walk (sparse reward), CRPI learns conservatively, but often succeeds in identifying better terminal policies.

2 Related Work

The earliest use of dynamic programming with 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 is based on changing only the policy evaluation step, leaving the greedy policy update unchanged. For evaluation, these methods take one of three paths: (i) minimize mean-squared projection error, as in TD(1) (Tsitsiklis and Van Roy, 1996); (ii) solve for the projected Bellman fixed point, as in LSPI and TD(λ) (Lagoudakis and Parr, 2003; Tsitsiklis and Van Roy, 1996); or (iii) minimize the Bellman 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 assume that a more accurate value approximation will yield a better policy—a reasonable but by no means self-evident hypothesis." These methods often diverge, oscillate between worse policies, or exhibit unreliability in learning a good policy (Bertsekas, 2011; Patterson et al., 2024; Gopalan and Thoppe, 2022; Young and Sutton, 2020). A theoretically sound approach addressing these concerns has remained elusive so far.

PI with Conservative Policy Update. A complementary line of research focuses on making policy updates cautious, either by (i) bounding the KL divergence between successive policies or (ii) forming a convex mixture between the current and the full greedy policy. This idea dates back to Conservative PI (Kakade and Langford, 2002)—and its recent extension Safe PI (Metelli et al., 2021)—and underpins several influential algorithms,

including TRPO (Schulman et al., 2015a), PPO (Schulman et al., 2017). (Metelli et al., 2021, Remark 3) shows that such cautious updates can yield strong per-step improvement bounds. Also, (Metelli et al., 2021, Figures 9, 13) show that SPI often learns conservatively, but reaches better policies terminally. However, the theoretical guarantees presuppose access to accurate value estimates for *every* state–action pair—an assumption that cannot be met in realistic FA settings.

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.

Finally, there is classification-based PI (Lazaric et al., 2010) that treats the greedy policy step as a supervised learning problem, but forfeits per-iteration guarantees. Linear programming has also been proposed to frame dynamic programming with linear FA (De Farias and Van Roy, 2003). This idea may seem similar to our proposed RPI, but the resemblance is only skin-deep. Theirs is a single pass approach that yields a lower bound on optimal Q-values. In contrast, our method arises from a new policy-evaluation within PI. It is iterative and lower bounds the current policy's value.

3 Problem Formulation, Proposed Algorithms, and Main Results

We formalize the setting and our goals in Section 3.1, and then present RPI and CRPI—together with their theoretical guarantees—in Sections 3.2 and 3.3, respectively. RPI introduces our new policy-evaluation step for FA, while CRPI additionally shows our novel FA-based conservative policy update. Section 4 provides proofs of our main results, while those of the technical propositions are in the Appendix.

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.

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

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

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

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 $\mathbb{R}^{SA}$. Hence, we have that $\mathcal{F} \subseteq \mathbb{R}^{SA}$.

3.2 Reliable Policy Iteration (RPI)

RPI's description is given in Algorithm 1. The inequalities there are meant to hold coordinate wise and we retain this meaning of $\geq$ throughout. Like other PI implementations, RPI also interleaves policy evaluations and policy updates. Its novelty is in its policy evaluation approach. Unlike standard techniques, observe that

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

RPI does not try to find an arbitrary close approximation to Q_{μ_k}, or minimize the projected-Bellman error, or even be conservatively close to f_k. Instead, RPI finds $f \in \mathcal{F}$ that is farthest away from f_k under a given norm $\|\cdot\|$, subject to the constraint $T_{\mu_k} f \geq f \geq f_k$. RPI's policy update step simply involves replacing the current policy μ_k with the one that is greedy with respect to f_{k+1}. Such policy updates are also used 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 derive 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 $Q \geq_{\text{p}} Q'$ 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_* denotes the optimal Q-value function, then*

$$\|Q_{\mu_\infty} - Q_*\|_\infty \leq \frac{2\|T f_\infty - f_\infty\|_\infty}{1-\gamma} = \frac{2\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}.$$

4. *Additionally suppose $\|\cdot\|$ is strictly monotone: $Q \geq Q' \geq 0$ and $Q \neq Q'$ imply $\|Q\| > \|Q'\|$ Also, the function class $\mathcal{F}$ has room to improve in the positive orthant centered at f_∞. That is, suppose 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, $T_{\mu_\infty} f_\infty = T f_\infty \geq_{\text{p}} f_\infty$, 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, $f_{k+1} = Q_{\mu_k}$ $\forall k \geq 0$, $f_\infty = Q_*$, 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, $f_{k+1} \in \arg\min_{f \in \mathcal{F}: T_{\mu_k} f \geq f \geq f_k} \|f - Q_{\mu_k}\|_{w,1}$.*

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}

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

Policy Evaluation:

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

Policy Improvement:

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

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

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 policy Bellman equation for μ_∞. While multiple such fixed points may exist (a fundamental FA limitation), RPI naturally aligns with the core RL goal of solving the unprojected Bellman equations. Moreover, unlike projection-based schemes that can oscillate between poor policies (Bertsekas, 2011), RPI guarantees that the limiting policy's value is never below f_∞.

3. **Geometric interpretation in weighted-ℓ_1**. Proposition 3.3 reveals that under any weighted-ℓ_1 norm, RPI's evaluation step is equivalent to projecting Q_{μ_k} onto the carefully chosen constrained set $\{f \in \mathcal{F} : T_{\mu_k} f \geq f \geq f_k\}$. Unlike a full-space projection, this selective projection embeds the Bellman inequalities directly. While this exact projection interpretation does not generalize to arbitrary norms, the coordinate-wise monotonicity guarantees hold for any norm, making our constrainted optimization a faithful extension of tabular PI to the FA setting.

In summary, by embedding the Bellman inequalities directly into the evaluation program in (1), RPI fully restores the classical PI guarantees of monotonic improvement and convergence—properties unattainable by any previous FA-based PI variant.

3.3 Conservative RPI (CRPI)

While RPI and vanilla PI employ greedy policy updates, CPI and SPI adopt conservative ones. In the presence of FA-based errors and sampling noise, such CPI/SPI-style updates have empirically led to superior final performance. Further, in tabular or near-tabular regimes, CPI/SPI admit per-step policy-improvement guarantees, obtained by maximizing a lower bound on the performance-difference gap. This section explains how CPI/SPI-style conservative policy updates can be incorporated within the RPI framework.

CRPI's description is given in Algorithm 2. It retains RPI's policy evaluation step, but differs in the policy update. Specifically, following CPI and SPI, CRPI sets μ_{k+1} to a convex combination of the current policy μ_k and the one greedy with respect to f_{k+1}. When $\alpha < 1$, using Theorem 3.9 below, this choice can be seen as the policy that maximizes suitable lower bounds on a *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.,

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

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 $x \in \mathbb{R}^{SA}$, $\mathrm{sp}(x) := \max_{s,a} x(s,a) - \min_{s,a} x(s,a)$ 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).** Kakade and Langford (2002)'s original lemma (in the Q-value-function version) states that $\nu^\top Q_{\mu'} - \nu^\top Q_\mu = \frac{\gamma}{1-\gamma} d_{\mu'}^\top a_\mu^{\mu'}(Q_\mu)$. When $Q_\mu = f$ such as in tabular settings, our result in (7) matches with the above since $T_\mu f = f$ then, making our second term vanish. Crucially, our result extends to practical FA-regimes, where Q_μ can only approximately known. Specifically, for any f with $T_\mu f \geq f$ (hence $Q_\mu \geq f$; see Claim 4.1), our result gives a computable lower bound on an approximate performance gap between μ' and μ; the approximation is due to the use of f, a known lower bound on Q_μ, instead of Q_μ itself. Also note that our result includes additional $(T_\mu f - f)$-type error terms that arise solely from FA, and the bound tightens 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

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

and

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

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

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

and

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

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

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

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 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 conclusions of Theorem 3.1 hold. In particular, $T_{\mu_k} f_{k+1} \geq f_{k+1}$ for any $k \geq 0$. The latter implies that the performance bounds from Theorem 3.8 apply to $\nu^\top Q_{\mu_{k+1}} - \nu^\top f_{k+1}$ for any $k \geq 0$ and any initial distribution ν on $\mathcal{S} \times \mathcal{A}$.*

Remark 3.10. Ideally, at iteration k, we would have liked an improvement guarantee over $\nu^\top Q_{\mu_k}$. However, with FA, Q_{μ_k} can only be known approximately. Our result therefore guarantees improvement over 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 a simpler—albeit looser—lower bound on the performance-improvement gap that holds for arbitrary (not just mixture) policies.

Proposition 3.11. *Let μ and μ' be arbitrary stationary policies and ν be any initial distribution. Further, suppose $f \in \mathbb{R}^{SA}$ is such that $T_\mu f \geq f$. Then,*

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \frac{\gamma}{1-\gamma} A_\mu^{\mu'}(f) - \frac{2\gamma^2}{(1-\gamma)^2} \|f\|_\infty D_{\mathrm{KL}}^{\max}(\mu', \mu) + \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f],$$

where $D_{\mathrm{KL}}^{\max}(\mu', \mu) = \max_{s \in \mathcal{S}} D_{\mathrm{KL}}\big(\mu'(\cdot \mid s) \,\|\, \mu(\cdot \mid s)\big)$.

Remark 3.12. In tabular settings, where $f = Q_\mu$ and thus $T_\mu f = f$, a related bound underpins the TRPO algorithm design (Schulman et al., 2015b). Our result extends this bound to the FA setting and includes the additional $(T_\mu f - f)$-based error terms. Developing a TRPO-style algorithm that leverages these terms in the FA setting is an interesting direction for future work.

4 Proofs

We prove Theorem 3.1, Lemma 3.4, and (part of) Theorem 3.9 here. Other proofs are in Appendix A.

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

Claim 4.1. *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 4.1 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, the above claim shows that

$$\{f \in \mathcal{F} : T_{\mu_k} f \geq f \geq f_k\} = \mathcal{F} \cap \{f \in \mathbb{R}^{SA} : T_{\mu_k} f \geq f \geq f_k\} \subseteq \mathcal{F} \cap \{f \in \mathbb{R}^{SA} : Q_{\mu_k} \geq f \geq f_k\}. \tag{14}$$

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

$$T_{\mu_k} f_{k+1} \geq f_{k+1} \geq f_k. \tag{15}$$

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

$$T_{\mu_{k+1}} f_{k+1} = T f_{k+1} \geq T_{\mu_k} f_{k+1} \geq f_{k+1}, \tag{16}$$

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 (15), 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 4.1 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 $f_k \leq Q_{\mu_k} \leq Q_*$, it follows that $f_\infty := \lim_{k\to\infty} f_k$ exists. Therefore,

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

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 $Tf \geq T_\mu f$ for any μ and f, while (e) holds since f_k satisfies the constraints in (1) as shown in our first statement.

Next, since $TQ_* = Q_*$, observe that

$$\|f_\infty - Q_*\|_\infty \leq \|Tf_\infty - f_\infty\|_\infty + \|Tf_\infty - TQ_*\|_\infty \leq \|Tf_\infty - f_\infty\|_\infty + \gamma\|f_\infty - Q_*\|_\infty,$$

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

$$\|f_\infty - Q_*\|_\infty \leq \frac{\|Tf_\infty - f_\infty\|_\infty}{1-\gamma}. \tag{17}$$

Similarly, since T_μ also is a γ-contraction for any μ, we have

$$\|f_\infty - Q_{\mu_\infty}\|_\infty \leq \frac{\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}. \tag{18}$$

Finally, since $Tf_\infty = T_{\mu_\infty} f_\infty$, it follows from (17) and (18) that

$$\|Q_{\mu_\infty} - Q_*\|_\infty \leq \|Q_{\mu_\infty} - f_\infty\|_\infty + \|f_\infty - Q_*\|_\infty \leq \frac{2\|Tf_\infty - f_\infty\|_\infty}{1-\gamma} = \frac{2\|T_{\mu_\infty} f_\infty - f_\infty\|_\infty}{1-\gamma}.$$

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

Proof of Lemma 3.4. First observe that

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

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

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

Finally, observe that

$$\nu^\top [\mathbb{I} - \gamma P_{\mu'}]^{-1}[T_\mu f - f] = \sum_{h=0}^{\infty} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f].$$

Also, $T_\mu f \geq f$, and $P_{\mu'}^h$ and ν are made up of non-negative entries; therefore,

$$\sum_{h=H+1}^{\infty} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f] \geq 0.$$

The desired result now follows. □

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 (16), 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.

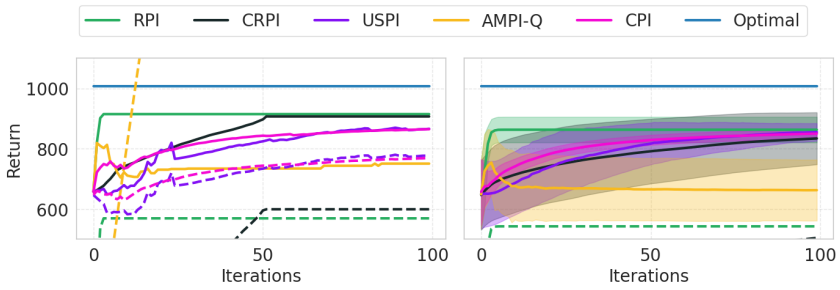
By reasoning analogous to the proof of Theorem 3.1, the conclusions of its first three statements can now be shown to hold for CRPI as well. The fourth statement does not carry over as directly: RPI generates deterministic policies, whereas CRPI may produce stochastic ones. Nevertheless, the argument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

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 $T_{\mu_k} f_{k+1} \geq f_{k+1}$. This condition holds because $T_{\mu_k} f_k \geq f_k$ (shown above) ensures (3) is feasible; hence a solution f_{k+1} exists and satisfies the desired condition. □

5 Experiments

This section presents empirical evaluations of our proposed algorithms against three standard benchmarks: CPI, USPI, and AMPI-Q. We conduct experiments on two distinct environments: the standard inventory control problem (Bertsekas, 2012), a dense-reward setting, and the chain walk (Lagoudakis and Parr, 2003), a sparse-reward setting. For each environment, we present learning curves and compare the terminal policy performance. We quantify learning efficiency using the Area Under the Curve (AUC) of the learning curve, where a higher AUC reflects greater cumulative reward accrued during training.

For brevity, we defer ablation analyses and full experimental details—including computational resources, and solvers—to the appendix.



Algo.	AUC ($\times 10^3$) (mean $\pm$ std)	Terminal Perf. ($\times 10^2$) (mean $\pm$ std)
RPI	$\mathbf{85.1 \pm 4.2}$	$\mathbf{8.6 \pm 0.4}$
CRPI	77.2 ± 10.1	8.3 ± 0.9
USPI	78.4 ± 6.4	$\mathbf{8.6 \pm 0.3}$
AMPI-Q	66.3 ± 9.8	6.6 ± 1.0
CPI	80.2 ± 1.6	8.5 ± 0.2

Figure 1: Inventory Control with linear function approximation. **Left:** Training curve of a single representative run (solid: true return, dashed: estimated return). **Center:** Averaged training curves over 100 runs (solid: mean return, shaded: mean return $\pm$ 1 std). **Right:** Key metrics table (AUC and terminal performance). **Summary**: RPI converges fastest, outperforms baselines on average across runs, and achieves a higher AUC—indicating faster, more sample-efficient learning.

5.1 Inventory Control

We conducted our experiments on the standard inventory control problem (Bertsekas, 2012) with the following parameters: maximum inventory capacity $M = 49$ ($|\mathcal{S}| = |\mathcal{A}| = 50$), unit cost $c = 5$, holding cost $h = 1$, and selling price $p = 10$. We presume the daily demand follows a uniform distribution over $\{0, \ldots, M\}$ and the discount factor $\gamma = 0.9$. We consider a linear function-approximation setting for this problem, i.e., $\Phi \in \mathbb{R}^{SA \times d}$ is the feature matrix. In all our experiments, $d = 75$ and the entries of Φ, i.e., the features, are sampled uniformly from the interval $[1, 5]$.

Observations: Figure 1 gives the performance on this inventory control task. The left panel displays one representative run. Solid lines plot true policy value ($\mathbb{E}_{(s,a)\sim\nu}[Q_{\mu_k}(s,a)]$) and dashed lines show their estimates ($\mathbb{E}_{(s,a)\sim\nu}[\Phi(s,a)^\top \theta_k]$). The center panel aggregates results over 100 random seeds, with shaded bands representing ± 1 standard deviation.

Our experiments on the inventory control problem highlight the critical impact of function approximation on different policy iteration algorithms. While methods like USPI and CPI offer performance improvement guarantees in the tabular setting, these assurances do not extend to cases with function approximation. Consequently, the approximation errors can cause policy performance to degrade intermittently, as observed in the left plot. AMPI-Q updates policies greedily with respect to its value function estimates, a process that can lead to policy degradation and cause value estimates to dangerously overshoot the optimal value, yet this overestimation fails to translate into a high-performing policy.

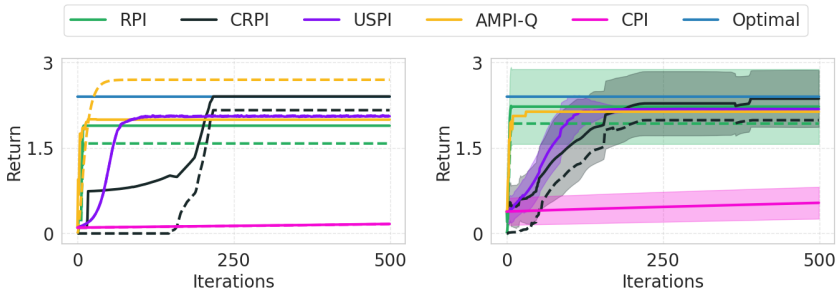
In contrast, RPI is designed to circumvent these issues. By maintaining a certified lower bound on the policy's value function rather than an exact estimate, it creates a reliable and stable signal for policy improvement. This fundamental difference allows RPI to learn the fastest and converge to a higher-performing terminal policy, demonstrating its robustness in the presence of approximation errors.

The conservative variant, CRPI, also proves to be an effective, albeit slower, method. Due to its highly conservative updates, CRPI's progress is more gradual but remarkably steady, eventually converging to a terminal performance comparable to RPI's. Notably, its upper confidence band indicates that in some runs, it can also outperform all other baselines.

In summary, this experiment demonstrates the distinct advantage of RPI in a dense reward setting using a linear function approximation class. Not only does RPI converge to a higher policy value on average, but it also learns considerably faster than other baseline methods. We note that even though the broader family of conservative policy iteration algorithms (like CPI and USPI) typically approach RPI's terminal performance, they do so at a markedly slower rate.

5.2 Chain Walk

In the sparse-reward setting, we consider Chain Walk (Lagoudakis and Parr, 2003), an MDP that consists of $N = 50$ states arranged in a linear chain with states labeled 1 to N. The agent's actions ('Left', 'Right') are



Algo.	AUC ($\times 10^2$) (mean $\pm$ std)	Terminal Perf. (mean $\pm$ std)
RPI	9.9 ± 3.2	2.2 ± 0.7
CRPI	9.1 ± 2.2	$\mathbf{2.4 \pm 0.5}$
USPI	9.7 ± 0.5	2.2 ± 0.0
AMPI-Q	$\mathbf{10.5 \pm 0.0}$	2.1 ± 0.0
CPI	2.3 ± 1.3	0.5 ± 0.3

Figure 2: Chain Walk with linear function approximation. **Left:** A single training run. RPI and CRPI both maintain a monotonic lower bound. This contrasts with the highly accurate estimates of USPI and CPI and the severe overestimation by AMPI-Q, far exceeding the optimal value. **Center:** Averaged training curves over 25 runs (solid: mean return, shaded: mean return $\pm$ 1 std). **Right:** Key performance metrics. **Summary:** Although AMPI-Q achieves the highest AUC, indicating rapid initial learning, CRPI's conservative updates lead to the best and most stable terminal performance.

stochastic: the agent moves one step in the intended direction with probability $p = 0.9$ and in the opposite direction with probability $1 - p$. The reward is sparse, with a positive reward granted only in two states located a distance of $N/4$ from each end of the chain. This setup is challenging due to sparse feedback and long horizons.

We use a linear function approximation class with a feature dimension of $d = 90$ and a discount factor $\gamma = 0.9$. We observed that, unlike inventory control task, algorithm performance in this domain was sensitive to the choice of features. To ensure a fair comparison, we select the best-performing feature matrix for each algorithm from 10 random candidates.

Observations: The results for the Chain Walk task are presented in Figure 2. To ensure a robust analysis of the 25 seeds, which contained outliers (due to bad initializations), the aggregated statistics are computed after removing extreme observations using Tukey's outlier detection rule (Tukey, 1977). The results highlight the principal advantage of CRPI in this challenging sparse-reward environment. It is seen that on average CRPI settled at a higher policy value compared to other baselines. The table reveals a crucial trade-off between learning efficiency and final policy quality: while some baselines achieve a higher AUC, CRPI's conservative updates enable it to robustly converge to a superior terminal policy. This success stems directly from its novel design. Unlike CPI and USPI, CRPI's performance bounds are uniquely constructed to be reliable under function approximation. It is this principled approach of handling function-approximation errors that provides the stability needed to find high-performing policies in this challenging sparse-reward setting as shown in the plots.

6 Conclusion

We have introduced Reliable Policy Iteration (RPI) and its conservative variant, CRPI, to address a foundational challenge in reinforcement learning: the failure of classical policy iteration guarantees under function approximation. Our methods restore properties like monotonic value estimates by using a novel Bellman-constrained optimization for policy evaluation, producing a certified lower bound on the true return. While RPI uses a standard greedy update, CRPI incorporates a conservative step that maximizes a new performance-difference lower bound. This bound, derived to account for function approximation errors, allows CRPI to admit per-step improvement guarantees.

The practical advantages of our principled approach were evident in the experiments. RPI demonstrated faster learning and higher terminal performance in the dense-reward inventory control task, while CRPI's stability and conservative design enabled it to gradually attain superior terminal policies in the challenging sparse-reward Chain Walk environment. By successfully extending tabular guarantees to the function approximation setting, RPI and CRPI offer a robust foundation for developing more scalable and theoretically sound reinforcement learning agents under function approximation.

References

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

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.

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.

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.

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. (2022). Does DQN learn? *arXiv preprint arXiv:2205.13617*.

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.

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., 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. (2015a). Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015b). 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.

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

Tukey, J. W. (1977). Exploratory data analysis. *Reading/Addison-Wesley*.

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*.

Appendix

A Proofs

A.1 Proof of Proposition 3.2

Since $\mathcal{F} = \mathbb{R}^{SA}$, we have that $Q_{\mu_k} \in \mathcal{F}$. On the other hand, Claim 4.1 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, $f_{k+1} = Q_{\mu_k}$. Invoking the classical convergence proof for PI (Bertsekas and Tsitsiklis, 1996) now shows that $f_\infty = Q_*$ and μ_∞ is the optimal policy, as desired.

A.2 Proof of Proposition 3.3

For any f satisfying the constraints in (1), we have from Claim 4.1 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

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

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

A.3 Proof of Proposition 3.6

We first prove the following technical result that mirrors Lemma 1 of (Metelli et al., 2021). Let M_μ be the $S \times SA$-dimensional matrix whose $(s,(s',a'))$-th entry is $\mu(a'|s')$ if $s = s'$ and 0 otherwise.

Lemma A.1. *For any initial distribution ν and stationary policies μ', μ, we have*

$$\|d_{\mu'} - d_\mu\|_1 \leq \frac{\gamma}{1-\gamma}\|\mu' - \mu\|_{1,\delta_\mu}. \tag{19}$$

Proof. We have

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

Hence, it follows that

$$\begin{aligned} \|d_{\mu'} - d_\mu\|_1 &= \|d_{\mu'}^\top - d_\mu^\top\|_\infty \\ &\leq \gamma\|d_\mu^\top P[M_{\mu'} - M_\mu]\|_\infty \, \|[\mathbb{I} - \gamma P_\mu]^{-1}\|_\infty \\ &= \frac{\gamma}{1-\gamma}\|d_\mu^\top P[M_{\mu'} - M_\mu]\|_\infty. \end{aligned} \tag{20}$$

Now,

$$\begin{aligned}
\|d_\mu^\top P[M_{\mu'} - M_\mu]\|_\infty &= \sum_{s',a'} \Big| \sum_{s,a} d_\mu(s,a) P(s'|s,a)[\mu'(a'|s) - \mu(a'|s)] \Big| \\
&\leq \sum_{s',a'} \sum_{s,a} d_\mu(s,a) P(s'|s,a) \Big| \mu'(a'|s') - \mu(a'|s') \Big| \\
&= \sum_{s,a,s'} d_\mu(s,a) P(s'|s,a) \sum_{a'} \Big| \mu'(a'|s') - \mu(a'|s') \Big| \\
&\leq \sum_{s,a,s'} d_\mu(s,a) P(s'|s,a) \|\mu'(\cdot|s') - \mu(\cdot|s')\|_1 \\
&= \|\mu' - \mu\|_{1,\delta_\mu}. \qquad (21)
\end{aligned}$$

where δ_μ is defined as in Section 3.3. The desired result now follows from (20) and (21). □

Proof of Proposition 3.6. Starting from Lemma 3.4 we get

$$\frac{1-\gamma}{\gamma}(\nu^\top Q_{\mu'} - \nu^\top f) \geq d_{\mu'}^\top a_\mu^{\mu'}(f) + \frac{1-\gamma}{\gamma} \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]. \qquad (22)$$

Now, for any μ_b, observe that

$$\begin{aligned}
d_{\mu'}^\top a_\mu^{\mu'}(f) &= d_{\mu_b}^\top a_\mu^{\mu'}(f) + (d_{\mu'}^\top - d_{\mu_b}^\top) a_\mu^{\mu'}(f) \\
&\overset{(a)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \big|(d_{\mu'} - d_{\mu_b})^\top a_\mu^{\mu'}(f)\big| \\
&\overset{(b)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \|(d_{\mu'} - d_{\mu_b})^\top\|_1 \frac{\mathrm{sp}(a_\mu^{\mu'}(f))}{2} \\
&\overset{(c)}{\geq} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \frac{\gamma}{1-\gamma} \|\mu' - \mu_b\|_{1,\delta_{\mu_b}} \frac{\mathrm{sp}(a_\mu^{\mu'}(f))}{2}, \qquad (23)
\end{aligned}$$

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.1.

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

$$\begin{aligned}
d_{\mu_b}^\top a_\mu^{\mu'}(f) &= \alpha d_\mu^\top [P_{\bar{\mu}} - P_\mu] f = \alpha A_\mu^{\bar{\mu}}(f) \\
\|\mu'(\cdot|s) - \mu(\cdot|s)\|_1 &= \alpha \|\bar{\mu}(\cdot|s) - \mu(\cdot|s)\|_1 \qquad (24) \\
\mathrm{sp}(a_\mu^{\mu'}(f)) &= \alpha\ \mathrm{sp}(a_\mu^{\bar{\mu}}(f)).
\end{aligned}$$

Hence, by combining (22), (23), and (24), it follows that

$$\frac{1-\gamma}{\gamma}(\nu^\top Q_{\mu'} - \nu^\top f) \geq \alpha A_\mu^{\bar{\mu}}(f) - \frac{\gamma}{2(1-\gamma)} \alpha^2 \|\bar{\mu} - \mu\|_{1,\delta_\mu} \mathrm{sp}(a_\mu^{\bar{\mu}}(f)) + \frac{1-\gamma}{\gamma} \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f]. \qquad (25)$$

The desired results are now easy to see.

□

A.4 Proof of Theorem 3.8

$\Psi_1(\alpha)$ and $\Psi_0(\alpha)$, being concave quadratic functions of α are maximized at $\alpha_1^* = \frac{(1-\gamma)A_\mu^{\bar{\mu}}(f) + (1-\gamma)^2 \nu^\top a_\mu^{\bar{\mu}}(T_\mu f - f)}{\gamma \|\bar{\mu} - \mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar{\mu}}(f))}$ and $\alpha_0^* = \frac{(1-\gamma)A_\mu^{\bar{\mu}}(f)}{\gamma \|\bar{\mu} - \mu\|_{1,\delta_\mu}\ \mathrm{sp}(a_\mu^{\bar{\mu}}(f))}$ respectively.

α^* is defined as in Equation(13) and the new mixture policy is updated as $\mu' = \alpha\bar{\mu} + (1-\alpha)\mu$ where $\alpha = \min\{1, \alpha^*\}$.

If $\alpha_1^* > 1$ we take a full greedy step (i.e., $\alpha = 1$) and the improvement is at least $\Psi_1(1)$ (setting $\alpha = 1$ in Equation(9)).

If $\alpha_1^* \in [0,1]$, the improvement is at least the maximum value of (9), given by $\Psi_1(\alpha_1^*)$.

If $\alpha_1^* < 0$ and $\alpha_0^* > 1$, we take a full greedy step and the improvement is at least $\Psi_0(1)$ (setting $\alpha = 1$ in Equation(10)).

Finally, if $\alpha_1^* < 0$ and $\alpha_0^* \leq 1$, the improvement is at least the maximum value of (10), given by $\Psi_0(\alpha_0^*)$.

A.5 Proof of Theorem 3.9

Let (f_k, μ_k) be generated by CRPI in a finite MDP with discount $\gamma \in [0,1)$. It was earlier proved that for each k the evaluation step produces $T_{\mu_k} f_{k+1} \geq f_{k+1} \geq f_k$ and $f_k \uparrow f_\infty$ componentwise.

We assume the following:

(A1) (*Strictly monotone norm*) If $x \geq y \geq 0$ and $x \neq y$ then $\|x\| > \|y\|$.

(A2) (*Room to improve at f_∞*) There exists $\delta_0 > 0$ such that for every $0 < \delta \leq \delta_0$ there is $f \in \mathcal{F}$ with $f > f_\infty$ componentwise and $\|f - f_\infty\|_\infty < \delta$.

We claim that for any subsequential limit μ_∞ of the policy sequence $\{\mu_k\}$ (so $\mu_{k_j} \to \mu_\infty$ for some k_j) and recalling that $f_{k_j} \to f_\infty$, we get

$$T_{\mu_\infty} f_\infty \ \geq_p \ f_\infty,$$

i.e., $T_{\mu_\infty} f_\infty \geq f_\infty$ componentwise and equality holds in at least one coordinate.

By the CRPI constraint and compactness of the policy space, pick a subsequence k_j such that $\mu_{k_j} \to \mu_\infty$ and $f_{k_j} \to f_\infty$. Passing to the limit in $T_{\mu_{k_j}} f_{k_j+1} \geq f_{k_j+1}$ and using joint continuity of $T_\mu f$ in (μ, f) yields $T_{\mu_\infty} f_\infty \geq f_\infty$ componentwise.

It remains to show *tightness* in at least one coordinate. Suppose, for contradiction, that the inequality is strict everywhere:

$$T_{\mu_\infty} f_\infty \ > \ f_\infty \qquad \text{(componentwise)}. \tag{26}$$

By (A2), pick $f \in \mathcal{F}$ with $f > f_\infty$ and $\|f - f_\infty\|_\infty < \delta_0$. Using the same reasoning as in the proof of Theorem 3.1, it can be shown that for such an f we have $T_{\mu_\infty} f > f$ componentwise.

Since $\mu_{k_j} \to \mu_\infty$ and f is fixed, from continuity of T_μ in μ, we have

$$\|T_{\mu_{k_j}} f - T_{\mu_\infty} f\|_\infty \ \longrightarrow \ 0.$$

Now fix such a large j. Since $T_{\mu_\infty} f > f$, for large j we also have $T_{\mu_{k_j}} f > f$ componentwise. Also, because $f_k \uparrow f_\infty$ from below and $f > f_\infty$ we have $f > f_{k_j}$ componentwise. Therefore f is feasible for the CRPI evaluation step at iteration k_j, i.e.,

$$T_{\mu_{k_j}} f > f > f_{k_j}.$$

Let f_{k_j+1} be the maximizer returned by the CRPI evaluation step at iteration k_j. Because $f > f_{k_j}$ with a strict improvement in at least one coordinate, the strictly monotone norm (A1) yields

$$\|f - f_{k_j}\| \ > \ 0.$$

Since f is feasible, maximality of f_{k_j+1} implies

$$\|f_{k_j+1} - f_{k_j}\| \geq \|f - f_{k_j}\| > 0.$$

Since $f > f_\infty$, we have $f - f_{k_j} \geq f_\infty - f_{k_j}$ and $f - f_{k_j} \neq f_\infty - f_{k_j}$; by (A1),

$$\|f - f_{k_j}\| \ > \ \|f_\infty - f_{k_j}\|.$$

Thus, we have

$$\|f_{k_j+1} - f_{k_j}\| \;\geq\; \|f - f_{k_j}\| \;>\; \|f_\infty - f_{k_j}\|. \tag{27}$$

Now, $f_{k_j+1} \leq f_\infty$ componentwise, so $0 \leq f_{k_j+1} - f_{k_j} \leq f_\infty - f_{k_j}$ and $f_{k_j+1} - f_{k_j} \neq f_\infty - f_{k_j}$, so by (A1),

$$\|f_{k_j+1} - f_{k_j}\| \;<\; \|f_\infty - f_{k_j}\|,$$

contradicting (27). Therefore $f_{k_j+1} \not\leq f_\infty$, i.e.

$$\exists (s,a): \quad f_{k_j+1}(s,a) \;>\; f_\infty(s,a).$$

This is a contradiction, since $f_k \uparrow f_\infty$ from below cannot overshoot f_∞ for large k.

Hence the assumption (26) is false, and $T_{\mu_\infty} f_\infty \geq_p f_\infty$.

A.6 Proof of Proposition 3.11

First, we prove the following technical result.

Theorem A.2. *Suppose μ, μ_b, and μ' are arbitrary stationary policies and $f \in \mathbb{R}^{SA}$ is an arbitrary vector such that $T_\mu f \geq f$. Further, let ν be arbitrary distribution on $\mathcal{S} \times \mathcal{A}$. Then,*

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

Proof. The desired claim follows from (22) and (23). □

Proof of Proposition 3.11. From Pinsker's inequality, it follows that, for any fixed state s,

$$\big\|\mu'(\cdot \mid s) - \mu(\cdot \mid s)\big\|_1 = 2\,\mathrm{TV}\big(\mu'(\cdot \mid s), \mu(\cdot \mid s)\big) \leq \sqrt{2\,\mathrm{D_{KL}}\big(\mu'(\cdot \mid s) \,\|\, \mu(\cdot \mid s)\big)}.$$

Now, from Theorem A.2, we have

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

Clearly,

$$\|\mu' - \mu_b\|_{1,\,\delta_{\mu_b}} = \mathbb{E}_{s\sim\delta_{\mu_b}}\Big[\|\mu'(\cdot \mid s) - \mu_b(\cdot \mid s)\|_1\Big] \leq \mathbb{E}_s\Big[\sqrt{2\,\mathrm{D_{KL}}\big(\mu'(\cdot \mid s) \,\|\, \mu_b(\cdot \mid s)\big)}\Big] \leq \sqrt{2D_{\mathrm{KL}}^{\max}(\mu', \mu_b)}.$$

Separately, since P is row stochastic, we have

$$\begin{aligned}
\frac{\operatorname{sp}\big(a_\mu^{\mu'}(f)\big)}{2} &\leq \big\|a_\mu^{\mu'}(f)\big\|_\infty \\
&= \big\|P(M'_\mu - M_\mu) f\big\|_\infty \\
&\leq \big\|(M'_\mu - M_\mu) f\big\|_\infty \\
&= \max_{s\in\mathcal{S}} \Big| \sum_{a\in\mathcal{A}} \big(\mu'(a \mid s) - \mu(a \mid s)\big) f(s,a) \Big| \\
&\leq \max_{s\in\mathcal{S}} \Big\{ \Big\|\mu'(\cdot \mid s) - \mu(\cdot \mid s)\Big\|_1 \frac{\operatorname{sp}\big(f(s,\cdot)\big)}{2} \Big\} \\
&\leq \max_{s\in\mathcal{S}} \sqrt{2\mathrm{D_{KL}}\big(\mu'(\cdot \mid s) \| \mu(\cdot \mid s)\big)} \max_{s\in\mathcal{S}} \frac{\operatorname{sp}\big(f(s,\cdot)\big)}{2} \\
&\leq \sqrt{2D_{\mathrm{KL}}^{\max}(\mu', \mu)} \max_{s\in\mathcal{S}} \frac{\operatorname{sp}\big(f(s,\cdot)\big)}{2} \\
&\leq \sqrt{2D_{\mathrm{KL}}^{\max}(\mu', \mu)}\, \|f\|_\infty.
\end{aligned}$$

Therefore, it follows that

$$\nu^\top Q_{\mu'} - \nu^\top f \geq \frac{\gamma}{1-\gamma} d_{\mu_b}^\top a_\mu^{\mu'}(f) - \frac{2\gamma^2}{(1-\gamma)^2} \|f\|_\infty \sqrt{D_{\mathrm{KL}}^{\max}(\mu', \mu_b).D_{\mathrm{KL}}^{\max}(\mu', \mu)} + \sum_{h=0}^{H} \gamma^h \nu^\top P_{\mu'}^h [T_\mu f - f].$$

By setting $\mu_b = \mu$, the desired result now follows.

□

Remark A.3. In tabular settings, where $f = Q_\mu$ and thus $T_\mu f = f$, a related bound underpins the TRPO algorithm design (Schulman et al., 2015b). Our result extends this bound to the FA setting and includes the additional $(T_\mu f - f)$-based error terms.

B 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.

B.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 $s_t \in \mathcal{S} = \{0, 1, \ldots, M\}$, represents the number of items currently in stock. The action space is $\mathcal{A} = \{0, 1, \ldots, M\}$, 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.

B.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.

C Additional Experiments on Chain Walk

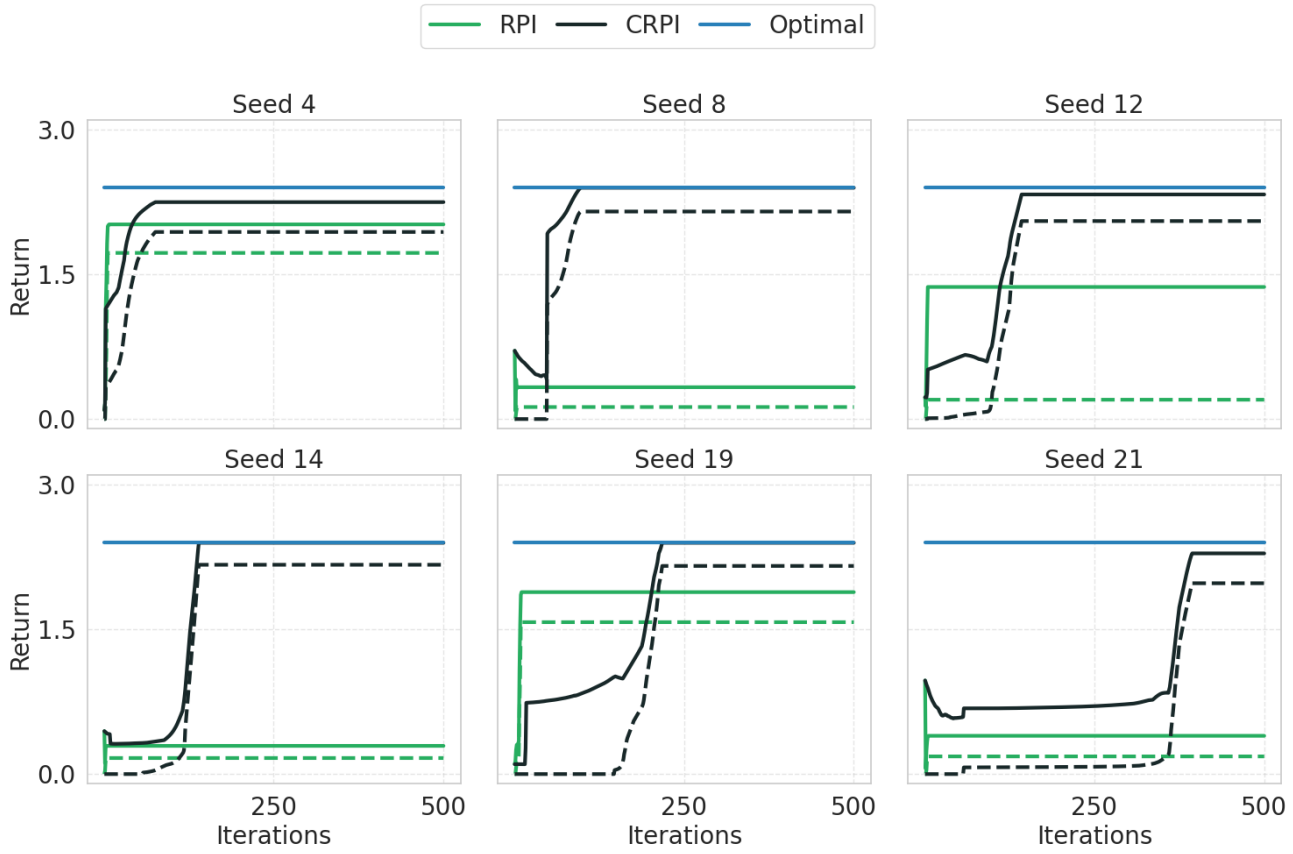


Figure 3: Performance comparison of CRPI against RPI on Chain Walk over 6 representative runs (solid: true values, dashed: estimated values)

C.1 The Benefit of Conservative Updates in CRPI

We now analyze how CRPI's conservative step selection can provide an advantage over RPI's purely greedy updates in a function approximation setting. While RPI always takes a full greedy step ($\alpha = 1$), CRPI seeks to select a step-size α that maximizes a theoretical lower bound on policy improvement, wherever possible. This conservative approach allows it to find more reliable, locally optimal steps, to potentially converge at a superior policy.

Figure 3 compares the performance of CRPI and RPI across several representative runs, showing both the true policy value (solid lines) and its linear approximation (dotted lines). A closer look at some individual iteration steps of these runs in Figure 4 reveals how CRPI could gain an advantage by not being purely greedy.

Conservative updates: RPI's greedy update ($\alpha = 1$) may result in minimal policy improvement. In contrast, CRPI could select a more conservative step ($\alpha \leq 1$) that would capture a larger local gain. This is evident in the iteration steps shown in Figure 4a and Figure 4d.

Greedy when necessary: CRPI is not simply conservative. It can also take purely greedy steps. When the lower bound maximization does not lead to a feasible step-size choice, CRPI reverts to the next best option, i.e., a purely greedy choice $\alpha = 1$, matching RPI. This scenario is shown in Figure 4c, where its choice aligns with the maximum of the approximate performance difference curve.

We also observe a case where CRPI is unable to choose the actual locally optimal step-size, but instead chooses a "near-optimal" step-size as can be seen in Figure 4b. This once again highlights the conservative nature of CRPI. Thus even though CRPI, in some iterations, could potentially fail to choose the exact optimal step-size,

it could still choose prudent, locally near optimal step-sizes, that yield steady, reliable progress. The benefit of which is evident in *seed 8* of Figure 3, where the purely greedy RPI settled far below CRPI, whereas CRPI continued to improve and, in this instance, eventually reached the optimal policy value.

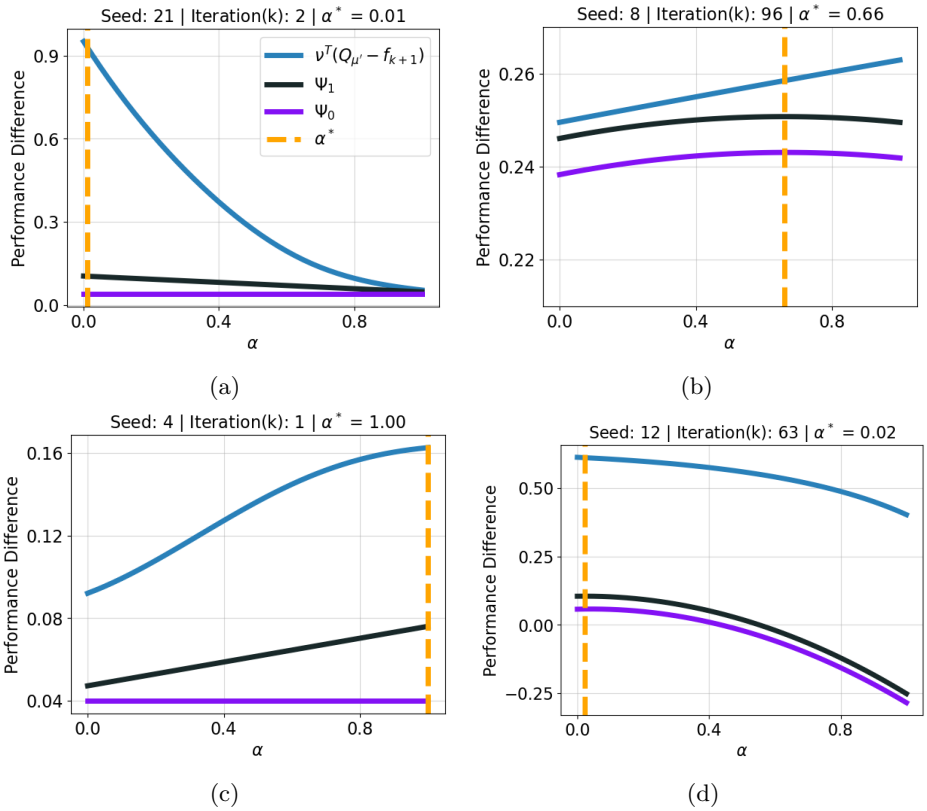


Figure 4: Comparison of Ψ_1 and Ψ_0 curves against approximate performance-difference curves across different runs and iteration stages.

C.2 Stability When Initialized at the Optimal Policy

To assess the stability of the algorithms, we investigated whether starting from the optimal policy, μ^*, could lead to performance degradation due to function approximation errors. Each algorithm was initialized at μ^* and its performance was averaged over 10 choices of feature matrix Φ.

The results in Figure 5 demonstrate a clear distinction between the algorithms.

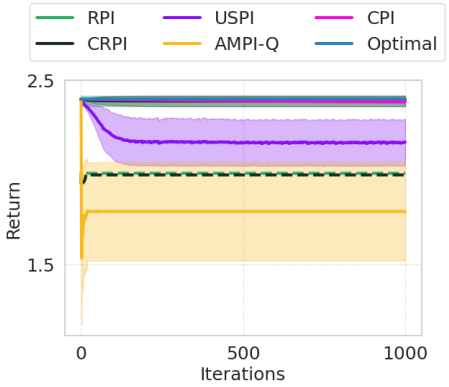
Stable Algorithms: CRPI, RPI, and CPI all recognized that no further improvements were necessary and their performance remained stable near the optimal value.

Unstable Algorithms: In stark contrast, USPI and AMPI-Q proved to be unstable. Their policy values degraded substantially, indicating that they not only fail to recognize the optimal policy but can actively move away from it.

This experiment provides strong evidence that in linear function approximation setting, RPI and CRPI are robust and stable. In contrast to USPI and AMPI-Q, these methods avert destructive updates to high-performing policies, rendering them the more reliable alternative.

D Computational Resources

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



Algo.	**AUC** ($\times 10^2$) (mean $\pm$ std)	**Terminal Performance** (mean $\pm$ std)
RPI	23.84 ± 0.29	$\mathbf{2.39 \pm 0.03}$
CRPI	23.85 ± 0.27	$\mathbf{2.39 \pm 0.03}$
USPI	21.74 ± 1.18	2.16 ± 0.13
AMPI-Q	17.85 ± 2.67	1.79 ± 0.27
CPI	$\mathbf{23.86 \pm 0.05}$	2.38 ± 0.01

Figure 5: Chain Walk starting from the optimal policy. **Left:** Averaged training curves over 10 runs for different feature matrices (solid: mean return, shaded: mean return $\pm$ 1 std). **Right:** Key metrics table (AUC and terminal performance). **Summary:** When initialized at the optimal policy, RPI, CRPI, and CPI maintain their performance, remaining stable near the optimum. In contrast, both AMPI-Q and USPI exhibit a significant degradation in performance.

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.