Augmented Lagrangian Method for Last-Iterate Convergence for Constrained MDPs

Michael Lu
Simon Fraser University
michael_lu_3@sfu.ca

Max Qiushi Lin
Simon Fraser University
maxqslin@gmail.com

Mo Chen
Canada CIFAR AI Chair, Amii,
Simon Fraser University
mochen@cs.sfu.ca

Sharan Vaswani
Simon Fraser University
vaswani.sharan@gmail.com

Abstract

We study policy optimization for infinite-horizon, discounted constrained Markov decision processes (CMDPs). While existing theoretical guarantees typically hold for the mixture policy, deploying such a policy is computationally and memory intensive. This leads to a practical mismatch where a single (last-iterate) policy must be deployed. Recent theoretical works have thus focused on proving last-iterate convergence, but are largely limited to the tabular setting or to algorithmic variants that are rarely used in practice. To address this, we use the classic inexact augmented Lagrangian (AL) method from constrained optimization, and propose a general framework with provable last-iterate convergence for CMDPs. We first focus on the tabular setting and propose to solve the AL sub-problem with projected Q-ascent (PQA). Combining the theoretical guarantees of PQA and the standard AL analysis enables us to establish global last-iterate convergence. We generalize these results to handle log-linear policies, and demonstrate that an efficient, projected variant of PQA can achieve last-iterate convergence with comparable guarantees as prior work. Finally, we demonstrate that our framework scales to complex non-linear policies, and evaluate it on continuous control tasks.

1 Introduction

Reinforcement Learning (RL) has found success in a variety of applications such as video games [Mnih et al., 2015], robotics [Kober et al., 2013] and fine-tuning large language models [Uc-Cetina et al., 2023]. While standard RL algorithms are typically designed to optimize a single performance objective, real-world systems must also operate under operational and safety constraints. For instance, in a wireless communication system, a network controller aims to maximize the bitrate delivered to users while ensuring that total energy consumption does not exceed a given budget [Julian et al., 2002, Buratti et al., 2009]. A common approach is to model such problems as a constrained Markov decision process (CMDP) [Altman, 2021]. In this setting, an agent must maximize a cumulative reward (e.g. bitrate), while ensuring that the cumulative constraint reward (e.g. energy consumption) satisfies a prescribed threshold. Unlike standard MDPs, the optimal policies for CMDPs are stochastic, making it challenging to design algorithms that reliably recover a near-optimal policy.

A typical algorithmic template [Jain et al., 2022, Liu et al., 2025b, Ding et al., 2020, Bura et al., 2022] to solve CMDPs is to form the Lagrangian and iteratively perform alternating updates of the policy (primal variable) and the associated Lagrange multipliers (dual variable). While such primal–dual (PD) methods are guaranteed to output a policy that maximizes the reward while satisfying the

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

constraints, such guarantees generally only hold for the *mixture policy*. In particular, PD methods iteratively generate a sequence of policies, and maintaining a mixture policy requires storing all intermediate policies, making this approach memory intensive. Moreover, in order to deploy a mixture policy, a policy is randomly sampled from the set of intermediate policies, and the theoretical guarantees hold in expectation over this random selection. From a safety perspective, relying on such a mixture policy can be dangerous, because an individual sampled policy may significantly violate the constraints. Consequently, practitioners use PD methods and deploy the final policy, trading theoretical guarantees for ease of implementation. Unfortunately, this practical variant is typically sensitive to hyper-parameters and the final policy can exhibit oscillatory behavior (see Fig. 1).

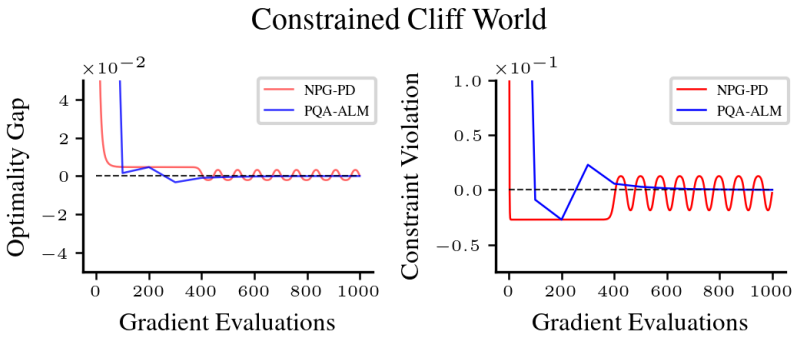

Figure 1: Comparing the proposed AL method (PQA−ALM) to the primal-dual approach (NPG−PD) [Ding et al., 2020] on a tabular environment. While the iterates of NPG−PD oscillate, PQA−ALM results in smooth convergence to zero optimality gap and constraint violation.

This concern has motivated recent works [Ding et al., 2023, Mondal and Aggarwal, 2026, Müller et al., 2024, Moskovitz et al., 2023, Ying et al., 2022, Montenegro et al., 2024] that aim to design iterative approaches that have theoretical guarantees for the final stochastic policy (see Table 1 for a detailed comparison). Such *last-iterate convergence* is desirable from both a theoretical and empirical perspective. All such recent works rely on adding additional regularization on either the policy, dual variables or to both. Despite their theoretical appeal, these last-iterate algorithms are either difficult to implement in practice, or are designed for simplified settings, limiting their applicability. In fact, to the best of our knowledge, there is no last-iterate algorithm that is both principled (has guarantees in simplified settings) and practical (can be efficiently implemented for large problems).

Table 1: Comparison of prior last-iterate convergence guarantees. To enable a fair comparison, the convergence rate is characterized in terms of the number of policy gradient evaluations. Assumption 1 denotes an exploration condition requiring the initial state distribution to be bounded away from zero, i.e. $\min_s \rho(s) > 0$. For function approximation, ϵ_{bias} denotes the irreducible approximation error of the restricted policy class (e.g., Assumption 5), CFA denotes the requirement of the compatible function approximation framework [Agarwal et al., 2021] and FND denotes the requirement of the Fisher non-degenerate assumption [Zhang et al., 2020] (fails to hold for log-linear policies [Ni and Kamgarpour, 2023, Proposition C.4]).

Method	Constraints	Horizon	Rate	Key Assumptions & Caveats
Tabular Setting				
Ying et al. [2022, Corollary 5.4]	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^2)$	Assumption 1
Ying et al. [2022, Corollary 6.2]	Single	Infinite	$\mathcal{O}(\log(1/\varepsilon))$	Assumption 1
Ding et al. [2023, Theorem 6]	Single	Infinite	$\tilde{\mathcal{O}}(1/\varepsilon^6)$	Assumption 1, requires projection onto a restricted simplex
Ding et al. [2023, Corollary 7]	Single	Infinite	$\mathcal{O}(\log(1/\varepsilon))$	Assumption 1, rate depends on potentially unbounded constants
Gladin et al. [2023, Corollary 4.5]	Multiple	Infinite	$\mathcal{O}(\log(1/\varepsilon))$	Requires the induced Markov chain to be uniformly ergodic
Montenegro et al. [2024, Theorem 3.2]	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^2)$	Assumption 1, rate depends on potentially unbounded constants
Moskovitz et al. [2023, Theorem 3.1]	Multiple	Infinite	–	Rate unspecified
Müller et al. [2024, Theorem 4.1]	Multiple	Finite	$\mathcal{O}(\text{poly}(A,H,n)/\varepsilon^{10})$	Polynomial dependence on A, H, n left unspecified
Corollary 1 (Ours)	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^6)$	Assumption 1
General Policy Parameterization				
Ding et al. [2023, Corollary 5]	Single	Infinite	$\mathcal{O}(1/\varepsilon^6)$	Log-linear policies, $\epsilon_{\text{bias}} \leq \mathcal{O}(\varepsilon^8)$, CFA
Montenegro et al. [2024, Theorem 3.2]	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^6)$	CFA, FND, ϵ_{bias} depends on FND constant
Mondal and Aggarwal [2026, Theorem 1]	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^4)$	Assumption 1, $\epsilon_{\text{bias}} \leq \mathcal{O}(\varepsilon^6)$, CFA, FND
Corollary 2 (Ours)	Multiple	Infinite	$\mathcal{O}(1/\varepsilon^6)$	Assumption 1, log-linear policies $\epsilon_{\text{bias}} \leq \mathcal{O}(\varepsilon^{16})$

Motivated by this gap, and in contrast to the previous work on last-iterate convergence, we employ the *augmented Lagrangian (AL) method*. Such methods are a classic approach in the constrained

optimization literature, and are widely used in practice (see Deng et al. [2025] for a recent survey). Under suitable assumptions, AL methods enjoy strong theoretical guarantees, including last-iterate global convergence [Xu, 2021, Liu et al., 2019]. Empirically, AL methods tend to be stable and more robust when compared to common PD approaches [Birgin and Martínez, 2014]. For example, Fig. 1 shows that the final policy produced by our proposed AL-based approach attains convergence to the optimal policy (the optimality gap is small) and satisfies the constraints more reliably when compared to the PD approach in Ding et al. [2020]. We note that such AL methods have been previously used for solving CMDPs. However, prior work either uses the AL method only as inspiration, without providing theoretical guarantees [Dai et al., 2023, Li et al., 2021], or is restricted to simplistic settings and relies on computationally expensive procedures that do not scale in practice [Müller et al., 2023]. Consequently, our objective is to develop a *general, practical and theoretically principled* AL *framework for last-iterate convergence in CMDPs*. To this end, we make the following contributions.

Contribution 1: In Section 3, we introduce a generic inexact AL framework for solving CMDPs. Each iteration of the algorithm requires computing an approximate solution to a non-concave AL *subproblem*. Assuming access to an inexact solver for the subproblem, we prove that this framework requires $\mathcal{O}(1/\varepsilon^2)$ iterations to output an ε-optimal policy (which simultaneously achieves an ε optimality-gap and constraint violation) for the CMDP. This modular design separates the AL analysis from the choice of the subproblem solver, enabling flexibility across a range of settings.

Contribution 2: We first consider the *tabular setting* where the number of parameters scales with the size of the state-action space, and propose to solve the AL subproblem using policy optimization. In Section 4.1, we show the policy gradient of the AL subproblem involves policy-dependent, non-stationary *pseudo-rewards*. We use projected Q-ascent (PQA) [Xiao, 2022, Liu et al., 2025a] which yields last-iterate convergence guarantees for the AL subproblem. In Section 4.2, we combine these results with the standard AL analysis [Liu et al., 2019], resulting in last-iterate guarantees for the CMDP. Specifically, for a target $\varepsilon > 0$, the proposed algorithm requires $\mathcal{O}(1/\varepsilon^6)$ policy gradient evaluations to obtain an ε-optimal stochastic policy.

Contribution 3: To extend our framework to practical settings with large state-action spaces, we use function approximation and consider a parametric class of policies. Such policies can be succinctly expressed using fewer parameters compared to the state-action space. For our AL framework, we solve each subproblem with a restricted policy class. In Section 5, we derive the AL policy gradient and show that it can be estimated as efficiently as in standard RL. As a result, our framework allows common PG methods such as PPO [Schulman et al., 2017] or SPMA [Asad et al., 2024] to be used to solve the AL subproblem. Using the AL policy gradient, we follow Vaswani et al. [2021], Asad et al. [2024] and develop a surrogate optimization approach to update the parameterized policy.

In Section 5.1 we specialize this approach to log-linear policies [Yuan et al., 2022a, Agarwal et al., 2021] and prove that a projected variant of PQA requires $\mathcal{O}(1/\varepsilon^6)$ gradient evaluations to output an ε-optimal final stochastic policy. Unlike Ding et al. [2023], our method does not involve non-standard algorithmic modifications and works for multiple constraints. Furthermore, unlike Mondal and Aggarwal [2026], Montenegro et al. [2024], it is computationally efficient to implement in practice.

Contribution 4: In Appendix G, we evaluate our approach on large-scale constrained continuous control tasks from Safety-Gymnasium [Ji et al., 2023]. We instantiate our framework using PPO and SPMA. Our experiments demonstrate that the proposed approach performs comparably to strong heavily-engineered baselines such as PPO-Lag [Ray et al., 2019] and CPO [Achiam et al., 2017].

2 Problem Formulation & Background

We consider an infinite-horizon discounted constrained Markov decision process (CMDP) [Altman, 2021] defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \rho, \gamma, r, \{c_i\}_{i=1}^m, \{b_i\}_{i=1}^m)$. Here, $\mathcal{S}$ is the set of states with $S = |\mathcal{S}|$, $\mathcal{A}$ is the set of actions with $A = |\mathcal{A}|$, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$ is the transition probability function, $\rho \in \Delta_{\mathcal{S}}$ is the initial state distribution, $\gamma \in [0, 1)$ is the discount factor and $r : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the reward function. The constraint reward functions are denoted by $\{c_i\}_{i=1}^m$ where $c_i : \mathcal{S} \times \mathcal{A} \to [0, 1]$ and $\{b_i\}_{i=1}^m$ is the set of corresponding non-negative thresholds.

For policy $\pi : \mathcal{S} \to \Delta_{\mathcal{A}}$, the *state-action value function* $Q_\diamond^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as $Q_\diamond^\pi(s, a) := \mathbb{E}[\sum_{h=0}^\infty \gamma^h \diamond (s_h, a_h) | s_0 = s, a_0 = a]$ such that for $h \geq 1$, $s_{h+1} \sim \mathcal{P}(\cdot|s_h, a_h)$, $a_{h+1} \sim \pi(\cdot|s_{h+1})$ and $\diamond : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is a bounded (pseudo) reward function equal to either r or $\{c_i\}_{i=1}^m$ or their linear

combination. The *value function* $V_\diamond^\pi : \mathcal{S} \to \mathbb{R}$ is defined as $V_\diamond^\pi(s) := \mathbb{E}_{a\sim\pi(\cdot|s)}[Q_\diamond^\pi(s,a)]$ and we use $V_\diamond^\pi(\rho) := \mathbb{E}_{s_0\sim\rho}[V_\diamond^\pi(s_0)]$ to denote the expected value function. The *reward state-action value function* and *reward value function* are denoted as Q_r^π and V_r^π. Similarly, for each constraint $i \in [m]$, we denote the *constraint state-action value function* as $Q_{c_i}^\pi$ and the *constraint value function* as $V_{c_i}^\pi$.

The objective is to find a policy π that maximizes the expected reward value function while satisfying the constraints. Formally, we aim to solve the following problem:

Problem 1. $\max_\pi V_r^\pi(\rho) \quad$ *subject to* $\quad V_{c_i}^\pi(\rho) \geq b_i \quad \forall i \in [m]\,.$

Unlike MDPs, the optimal policy π^* of Problem 1 may be stochastic but is stationary [Altman, 2021]. A common approach to solving Problem 1 is to form its Lagrangian and consider the following problem: $\max_\pi \min_{\lambda\in\mathbb{R}_+^m} \mathcal{L}(\pi,\lambda) := V_r^\pi(\rho) + \sum_{i=1}^m \lambda_i \left[V_{c_i}^\pi(\rho) - b_i\right]$, where λ is the Lagrange multiplier (dual variable). Primal-dual (`PD`) methods solve the corresponding max-min problem by iteratively and alternatively updating the policy and the Lagrange multiplier. For example, the policy is updated using natural policy gradient, whereas the Lagrange multiplier is updated using gradient descent [Ding et al., 2020].

For a target error $\varepsilon > 0$, prior work [Jain et al., 2022, Ding et al., 2020, Liu et al., 2025b] prove that after T iterations, it is possible to output a mixture policy $\bar{\pi}_T := \frac{1}{T}\sum_{t=1}^T \pi_t$ which is an ε-optimal solution to Problem 1, i.e.,

$$V_r^{\bar{\pi}_T}(\rho) \geq \max_\pi V_r^\pi(\rho) - \varepsilon \quad \text{and} \quad V_{c_i}^{\bar{\pi}_T}(\rho) \geq b_i - \varepsilon, \quad \forall i \in [m]. \tag{1}$$

However, maintaining the mixture policy requires storing all the policies $\{\pi_t\}_{t=1}^T$, making this memory intensive. Moreover, in order to deploy a mixture policy, a policy is repeatedly chosen uniformly at random from $\{\pi_1, \pi_2, \ldots, \pi_T\}$, and the guarantees for the mixture policy only hold in expectation over this random selection. This implies that constraints are only approximately satisfied in expectation. In many safety-critical settings, such behavior is undesirable because a randomly sampled policy may significantly violate the constraints. These disadvantages significantly limit the practicality of `PD` methods, and motivate the development of principled algorithms that do not rely on mixture policies. To that end, prior work [Mondal and Aggarwal, 2026, Ding et al., 2023, Müller et al., 2024, Ying et al., 2022, Gladin et al., 2023] has developed techniques that ensure *last-iterate* convergence meaning that the resulting guarantees hold for the final stochastic policy. These works employ regularization such as entropy-regularization on the policy and ℓ_2 regularization on the dual variables. In contrast, we propose using the `AL` method for CMDPs, and describe it next.

3 Augmented Lagrangian Policy Optimization Framework

The `AL` method [Bertsekas, 2014, Powell, 1969, Rockafellar, 1976] is a classical approach for constrained optimization problems. It adds a quadratic penalty term to the Lagrangian to discourage constraint violations. In contrast to `PD` methods, the `AL` method considers the following problem:

$$\max_\pi \min_\lambda \mathcal{L}^\beta(\pi,\lambda) := V_r^\pi(\rho) + \frac{\beta}{2}\sum_{i=1}^m \left(-\min\left\{ V_{c_i}^\pi(\rho) - b_i - \frac{\lambda_i}{\beta}, 0 \right\}^2 + \frac{\lambda_i^2}{\beta^2} \right) \tag{2}$$

Here, $0 < \beta < \infty$ is the penalty parameter, and the min operator within the quadratic term corresponds to be the constraint violation for the ith constraint. Intuitively, a larger parameter β increases the weight of the quadratic penalty term which forces the policy to satisfy the constraints, improving feasibility. However, choosing β too large leads to an ill-conditioned optimization problem. As $\beta \to 0$, the `AL` reduces to the standard Lagrangian.

To find a solution to Eq. (2), the inexact `AL` method iteratively and alternatively updates the policy (primal variable) and the Lagrange multiplier (dual variable). Specifically at outer iteration $t \geq 1$, for a target sub-optimality $\epsilon_t > 0$ and dual variable λ_t, the inexact `AL` method seeks an approximate solution π_{t+1} to the `AL` *subproblem*, such that,

$$\mathcal{L}^\beta(\pi_{t+1}, \lambda_t) \geq \max_\pi \mathcal{L}^\beta(\pi, \lambda_t) - \epsilon_t. \tag{3}$$

Once an approximate primal solution is obtained, the dual variable is updated according to: $\forall i \in [m]$

$$\lambda_{t+1}[i] = \lambda_t[i] - \frac{\beta}{2}\left(V_{c_i}^{\pi_{t+1}}(\rho) - [b_i + \xi_i(\pi_{t+1})]\right) \tag{4}$$

where $\xi_i(\pi) \coloneqq \max\left\{V_{c_i}^{\pi}(\rho) - b_i - \frac{\lambda_t[i]}{\beta}, 0\right\}$ acts as a slack variable and prevents overly aggressive updates to the dual variable. For instance, $\xi_i(\pi_{t+1}) > 0 \iff V_{c_i}^{\pi_{t+1}}(\rho) > b_i + \frac{\lambda_t[i]}{\beta}$. As a result, when constraint i is well satisfied, the corresponding dual variable is only perturbed slightly. This avoids the excessive penalization of already-feasible constraints and prevents the dual variables from oscillating. On the other hand, if the constraint is not satisfied, the resulting update is akin to gradient descent w.r.t. λ and enforces feasibility. Compared to `PD` methods that regularize the primal/dual variables, the `AL` method employs a quadratic regularization on the constraint violation. Unlike `PD` methods, the step-size for the dual update in Eq. (4) is directly determined by the penalty β.

Unlike the conventional use of the `AL` method in concave settings, the main difficulty of applying it to CMDPs is that $\mathcal{L}^{\beta}(\pi, \lambda)$ is non-concave w.r.t π [Agarwal et al., 2021], making the `AL` subproblem challenging to solve accurately. Hence, we first assume access to the solver `Oracle-AL`$(\mathcal{L}^{\beta}, \lambda_t, \epsilon_t)$ that returns a policy satisfying Eq. (3) for all $t \geq 1$. This abstraction separates the analysis of the `AL` iterations from the choice of the policy optimization method used to solve the subproblem. Putting everything together, we describe our generic framework in Algorithm 1.

Algorithm 1: Generic `AL` Policy Optimization Method for Problem 1

1: **Input**: π_1 (primal variable), $\lambda_1 = 0$ (dual variable), $\beta > 0$ (penalty parameter), $\epsilon_t = \mathcal{O}\left(\frac{1}{t^2}\right)$ (target sub-optimality for `AL` subproblem), T (number of outer-iterations)
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Form `AL`: $\mathcal{L}^{\beta}(\pi, \lambda_t) \coloneqq V_r^{\pi}(\rho) + \frac{\beta}{2}\sum_{i=1}^{m}\left(-\min\left\{V_{c_i}^{\pi}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\right\}^2 + \frac{\lambda_i^2}{\beta^2}\right)$
4: $\pi_{t+1} = $ `Oracle-AL`$(\mathcal{L}^{\beta}, \lambda_t, \epsilon_t)$
5: $\lambda_{t+1}[i] = \lambda_t[i] - \frac{\beta}{2}\left(V_{c_i}^{\pi_{t+1}}(\rho) - [b_i + \xi_i(\pi_{t+1})]\right)$, $\xi_i(\pi) \coloneqq \max\left\{V_{c_i}^{\pi}(\rho) - b_i - \frac{\lambda_t[i]}{\beta}, 0\right\}$
6: **end for**
7: **Return**: π_{T+1}

Assuming access to `Oracle-AL`$(\mathcal{L}^{\beta}, \lambda, \epsilon)$, we follow a standard analysis for `AL` methods [Liu et al., 2019], and characterize the convergence of Algorithm 1 in Theorem 1.

Theorem 1. *Under Assumption 4, assuming for all $t \geq 1$, there exists an solver* `Oracle-AL`$(\mathcal{L}^{\beta}, \lambda_t, \epsilon_t)$ *which returns a policy π_{t+1} satisfying Eq. (3), for a given $\varepsilon \in (0, 1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \mathcal{O}\left(1/t^2\right)$, $\beta > 0$, and $\lambda_1 = 0$ returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1.*

Proof Sketch: We first recast the CMDP into an equivalent linear program (LP). For this, define $\Pr^{\pi}[s_\tau = s, a_\tau = a | s_0]$ as the probability of visiting state s and taking action a at step τ when starting at the initial state s_0 and the unnormalized *state-action occupancy measure* $\mu^{\pi} \in \mathbb{R}_+^{SA}$ is defined as $\mu^{\pi} \coloneqq \sum_{s_0} \rho(s_0) \sum_{\tau=0}^{\infty} \gamma^{\tau} \Pr^{\pi}[s_\tau = s, a_\tau = a | s_0]$. Note for any μ^{π}, there exists mapping between μ^{π} and π given by: $\pi(a|s) = \mu^{\pi}(s,a) / \sum_{a' \in \mathcal{A}} \mu^{\pi}(s,a')$, $\forall (s,a) \in \mathcal{S} \times \mathcal{A}$, whenever the denominator is positive, with arbitrary choices at states with zero occupancy. Consequently, the objective can be expressed directly in terms of the occupancy measure μ^{π}, i.e., $V_r^{\pi}(\rho) = \langle \mu^{\pi}, r \rangle$ and $V_{c_i}^{\pi}(\rho) = \langle \mu^{\pi}, c_i \rangle$ for each constraint $i \in [m]$. As a result, we can rewrite Problem 1 as a LP in terms of μ^{π} [Altman, 2021] as follows:

Problem 2. $\max_{\mu^{\pi} \in \mathcal{K}} \langle \mu^{\pi}, r \rangle \quad$ *subject to* $\quad \langle \mu^{\pi}, c_i \rangle \geq b_i \quad \forall i \in [m]$,

where $\mathcal{K} \coloneqq \left\{\mu^{\pi} \in \mathbb{R}_+^{SA} \mid \forall s' \in \mathcal{S}, \sum_{a' \in \mathcal{A}} \mu^{\pi}(s', a') = \rho(s') + \gamma \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \mathcal{P}(s'|s,a)\, \mu^{\pi}(s,a)\right\}$ represents the set of all feasible state-action occupancy measures μ^{π} satisfying the Bellman flow constraints. Under this representation, both the objective and the constraints are linear w.r.t μ^{π}. We next convert the inequality constraints in Problem 2 into equalities constraints by introducing non-negative slack variables $z \in \mathbb{R}_+^m$:

Problem 3. $\max_{\mu^{\pi} \in \mathcal{K}, z \in \mathbb{R}_+^m} \langle \mu^{\pi}, r \rangle \quad$ *subject to* $\quad \langle \mu^{\pi}, c_i \rangle - z_i = b_i \quad \forall i \in [m]$

By Lemmas 3 and 4, for a target error $\varepsilon > 0$, obtaining an ε-approximate solution to the LP in Problem 3 in terms of μ^{π} induces a policy π that is an ε-approximate solution in Problem 1. Hence, the LP formulation serves as an analysis device for the policy iterates of Algorithm 1; the algorithm does not explicit construct μ^{π}.

We now consider the AL method used to solve Problem 3. The AL for Problem 3 is defined as:

$$\overline{F}_{\beta,\lambda}(\mu^\pi, z) \coloneqq \langle \mu^\pi, r\rangle + \sum_{i=1}^{m} \lambda_i(\langle \mu^\pi, c_i\rangle - b_i - z_i) - \frac{\beta}{2}\sum_{i=1}^{m}(\langle \mu^\pi, c_i\rangle - b_i - z_i)^2. \tag{5}$$

Note the AL subproblem is concave over (μ^π, z). By the mapping between μ and π, and Lemmas 6 and 7, the policy returned by $\texttt{Oracle-AL}(\mathcal{L}^\beta, \lambda_t, \epsilon_t)$ corresponds to a pair $\left(\mu^{\pi_{t+1}}, \bar{\xi}(\mu^{\pi_{t+1}})\right)$ satisfying

$$\overline{F}_{\beta,\lambda}(\mu^{\pi_{t+1}}, \bar{\xi}(\mu^{\pi_{t+1}})) \geq \max_{\mu^\pi \in \mathcal{K},\, z\in\mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) - \epsilon_t,$$

where $\bar{\xi}_i(\mu^\pi) := \max(\langle \mu^\pi, c_i\rangle - b_i - {}^{\lambda_t[i]}/_{\beta}, 0)$ for each constraint $i \in [m]$. Consequently, Algorithm 1 can be analyzed using standard inexact AL analysis for concave problems with linear constraints. Following Liu et al. [2019], using $\epsilon_t = \mathcal{O}({}^1/_{t^2})$, the inexact AL method results in the following convergence (Lemma 5):

$$V_r^*(\rho) - V_r^{\pi_{T+1}}(\rho) \leq \mathcal{O}\left(\frac{1}{\sqrt{T}}\right) \quad \text{and} \quad b_i - V_{c_i}^{\pi_{T+1}}(\rho) \leq \mathcal{O}\left(\frac{1}{\sqrt{T}}\right) \quad \forall i \in [m].$$

Setting $T = \mathcal{O}({}^1/_{\varepsilon^2})$ ensures both the objective sub-optimality and constraint violation are $\mathcal{O}(\varepsilon)$. Thus, the final policy π_{T+1} is an ε-approximate solution, completing the proof. □

Note the lifting argument used to transfer guarantees from μ^π to π is not tied to this framework and can be extended beyond this setting. Crucially, the AL formulation ensures that the final policy is a valid stochastic policy and does not rely on a mixture of past policies. For a target error $\varepsilon > 0$, the above meta-theorem (proved in Appendix B) shows that the inexact AL method attains global last-iterate $\mathcal{O}({}^1/_{\varepsilon^2})$ convergence if each AL subproblem can be solved to the required accuracy. This result isolates the core requirement of the AL framework, and the remaining challenge is to construct an efficient oracle to solve the AL subproblem. We first consider the tabular setting in the following section.

4 Instantiating the AL Oracle: Tabular Setting

We first derive the AL policy gradient (PG), and subsequently use it to solve the AL subproblem.

4.1 The AL Policy Gradient

We first consider the *direct representation* in the tabular setting, in which the policy is parameterized by its state–action probabilities, i.e., $\pi(\cdot|s) \in \Delta_A$ for all $s \in \mathcal{S}$. In the standard unconstrained RL setting, a recent line of theoretical works [Mei et al., 2020, Lu et al., 2024, Agarwal et al., 2021, Bhandari and Russo, 2021, Lan, 2023, Shi et al., 2023, Zhan et al., 2023] establishes global convergence guarantees for PG methods under various structural assumptions (e.g., smoothness or gradient domination). However, these analyses do not directly apply in this setting due to the quadratic penalty term in the AL. To characterize this challenge precisely, in Proposition 2 we first calculate the PG for the AL subproblem using the chain rule. For fixed $\lambda \in \mathbb{R}^m$ and $s \in \mathcal{S}$:

$$[\nabla\mathcal{L}^\beta(\pi,\lambda)]_s = \frac{d^\pi(s)}{1-\gamma}\sum_{a\in\mathcal{A}} Q^\pi_{R(\pi)}(s,a)\,; R(\pi) := r - \beta\sum_{i=1}^{m} c_i \min\left\{V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\right\}. \tag{6}$$

where $d^\pi(s) \coloneqq (1-\gamma)\sum_{a'\in\mathcal{A}}\mu^\pi(s,a')$ denotes the state-visitation occupancy measure of a policy π. Intuitively, the above equation states the policy aims to maximize a pseudo-reward $R(\pi)$ that depends on its current level of constraint satisfaction. For example, consider the single constraint case ($m = 1$). When the constraints are sufficiently satisfied i.e., $V^\pi_{c_1}(\rho) \geq b_1 + \frac{\lambda}{\beta}$, we have $[R(\pi)](s,a) = r(s,a)$. In this case, the PG solely considers maximizing the reward. On the other hand, when the constraint is (almost) violated, the policy must maximize a combination of the reward and cost, $[R(\pi)](s,a) = r(s,a) + c_1(s,a)\left(b_1 + {}^\lambda/_\beta - V^\pi_{c_1}(\rho)\right)\beta$. Here, the weight on the cost term scales proportionally with the magnitude of the constraint violation $b_1 + {}^\lambda/_\beta - V^\pi_{c_1}(\rho)$ and β.

Unlike the standard PG, the key difference is that the pseudo-reward $R(\pi)$ depends on the current policy. As a result, the pseudo-reward is non-stationary across iterations, and the objective no longer

satisfies the usual structural properties exploited in standard PG analyses. For example, classical tools such as the Bellman equation cannot be directly applied, since they rely on a fixed reward function. Consequently, extending global convergence guarantees of standard PG to the AL subproblem requires arguments that explicitly account for the non-stationary reward.

For convenience, we focus on the following abstract problem, where for a fixed $\lambda \in \mathbb{R}^m$, $\max_\pi G(\pi) \coloneqq \mathcal{L}^\beta(\pi, \lambda)$. We solve the problem using projected Q-ascent (PQA) [Xiao, 2022, Liu et al., 2025a] which closely resembles the projected policy gradient [Agarwal et al., 2021] used to attain global convergence for general objectives with non-stationary rewards [Zhang et al., 2020, Barakat et al., 2024]. For an inner-iteration $k \geq 1$, the PQA update is given as:

$$\forall s \in \mathcal{S} \quad , \quad \pi_{k+1}(\cdot \mid s) = \text{Proj}_{\Delta_A}\Big[\pi_k(\cdot|s) + \eta\, Q^{\pi_k}_{R(\pi_k)}(s, \cdot)\Big]. \tag{7}$$

To state our results, we first introduce and justify the following assumptions.

Assumption 1 (Exploration)**.** *For all $s \in \mathcal{S}$, $\rho(s) \geq \rho_{\min} > 0$.*

Assumption 2 (Bounded Pseudo-Rewards)**.** *For all $\pi \in \Pi$, $R(\pi) \in [0, U]$ for some constant $U \geq 0$.*

Assumption 3 (Smoothness)**.** *The objective $G : \Pi \to \mathbb{R}$ is L-smooth with respect to the Euclidean norm, i.e., $|G(\pi) - G(\pi') - \langle \nabla_\pi G(\pi'), \pi - \pi' \rangle| \leq \frac{L}{2} \|\pi - \pi'\|_2^2$ for any pair of policies $\pi, \pi' \in \Pi$.*

Assumption 1 is standard in PG literature [Asad et al., 2024, Agarwal et al., 2021, Xiao, 2022, Lu et al., 2024, Mei et al., 2020], and helps isolate the optimization aspect from exploration. Assumption 2 ensures that the objective and its gradients remain well-defined and finite. Meanwhile, Assumption 3 is commonly adopted in PG literature [Hazan et al., 2019, Zhang et al., 2020, 2021, Barakat et al., 2023, Ying et al., 2022, Barakat et al., 2024] and enables monotonic improvement of $G(\pi)$ under appropriate step-size choices. By adapting the proof technique in Zhang et al. [2020], we establish the following result in Appendix E.1.

Theorem 2. *Under Assumptions 1 to 3, for $K \geq 2$ iterations, using PQA update in Eq. (7) with $\eta = \rho_{\min}/L$ results in the following convergence: $\max_\pi G(\pi) - G(\pi_K) = \mathcal{O}\left(\frac{L}{\rho_{\min}^3 (1-\gamma)^3} \frac{1}{K}\right)$.*

Thus, PQA achieves an $\mathcal{O}(1/K)$ last-iterate convergence rate, and can be used as an efficient solver for the AL subproblem.

4.2 Putting Everything Together

To start, we consider the following assumption standard in prior work [Ding et al., 2020, Bai et al., 2022, Liu et al., 2025b, Ying et al., 2022, Gladin et al., 2023, Müller et al., 2024].

Assumption 4 (Slater Condition)**.** *$\exists \bar{\pi}$ and $\zeta \in \mathbb{R}^m_+$ such that $\forall i \in [m]$, $\zeta_i \coloneqq V_{c_i}^{\bar{\pi}}(\rho) - b_i > 0$.*

This assumption guarantees the existence of a strictly feasible policy. The constant ζ, refereed to as *Slater constant* measures the margin by which a policy satisfies the constraints and reflects the difficulty of the problem. Under Assumption 4, the optimal dual variable is bounded as $\|\lambda^*\|_1 \leq 1/((1-\gamma)\min_i \zeta_i)$ (Lemma 19). Thus, using this bound, we show in Appendix E.4 that the AL subproblem satisfies Assumptions 2 and 3 uniformly over all outer iterations (Lemmas 16 and 17).

We now instantiate Algorithm 1 with the PQA update for solving the AL subproblem (PQA-ALM). In particular, we combine the results of Theorems 1 and 4 and characterize the convergence of the resulting algorithm in the following corollary. Note, to measure the complexity of solving the CMDP, we count the total number of PG evaluations. For our AL framework, this corresponds to summing all inner-iterations over the T outer iterations. This enables a fair comparison to other methods, such as PD methods who have a single loop.

Corollary 1. *Under Assumptions 1 and 4, for a given $\varepsilon \in (0, 1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \mathcal{O}(1/t^2)$, $\beta > 0$, $\lambda_1 = 0$ and instantiating Oracle-AL using the PQA update in Eq. (7) with $\eta = \frac{\rho_{\min}}{L_t}$ for $K_t = \left\lceil \frac{32\, L_t\, (1 + 1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t} \right\rceil$ iterations returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1 after $\mathcal{O}(1/\varepsilon^6)$ gradient evaluations.*

Comparison to prior work: For general nonconcave-convex problems, without exploiting additional structure, this rate matches the expected complexity proved by Lin et al. [2020]. In the tabular CMDP

setting Ying et al. [2022] establish last-iterate convergence using a dual-based algorithm with a nested loop structure, achieving an $\mathcal{O}(1/\varepsilon^2)$ rate in terms of the total number of gradient evaluations. Gladin et al. [2023] extend their result and attain an $\mathcal{O}(\log(1/\varepsilon))$ last-iterate rate. However, their analysis and algorithm only apply in the tabular setting, limiting their applicability to large-scale problems. Similarily, Ding et al. [2023] use an optimistic primal–dual scheme (OPG), to obtain a $\mathcal{O}(\log(1/\varepsilon))$ last-iterate rate that depends on problem-specific constants that can be unbounded. Moskovitz et al. [2023] propose a related optimistic primal–dual method over state–action occupancy measures (ReLOAD). However, their theoretical guarantees lacks an explicit rate and the method is computationally more expensive than solving the LP directly.

In the alternative finite-horizon setting, Müller et al. [2024] prove last-iterate convergence for regularized NPG PD method (NPG-NRL) with an overall complexity of $\mathcal{O}(1/\varepsilon^{10})$. However their rate also includes unknown polynomial dependence on the number of actions. A closely line of work by Müller et al. [2023] also advocates using AL method in the finite-horizon setting. However, their approach optimizes the AL subproblem directly using dynamic programming. This resulting in substantial computational overhead and prevents the method from extending to practical settings.

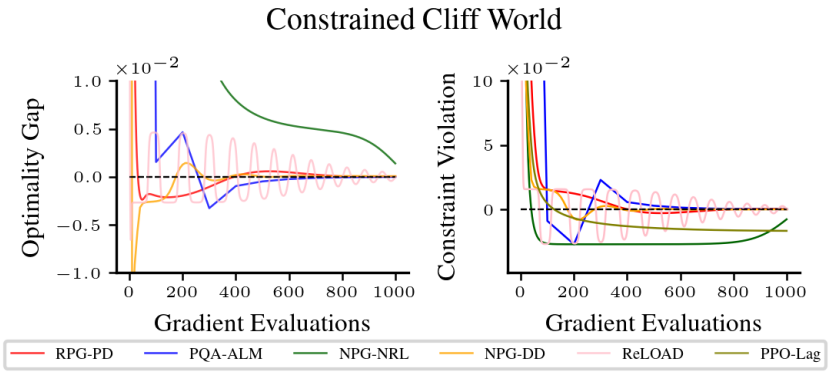

Figure 2: Comparing PQA-ALM against related prior methods. For all methods hyper-parameters were selected via grid search. We report the configuration with the smallest optimality gap among those satisfying $b - V_c^{\pi_{T+1}}(\rho) \leq \varepsilon = 0.001$. We observe that the AL methods, similar to RPG-PD [Ding et al., 2023] and NPG-DD [Ying et al., 2022], converge toward zero optimality gap and satisfy the constraint without noticeable oscillations. ReLOAD also eventually convergences but oscillates rapidly in the beginning. In contrast, NPG-NRL [Müller et al., 2024] is unable to converge to the optimal policy.

When the state-action space is large, it is common to employ function approximation (FA) to generalize across related states. In our framework, extending to this more practical setting requires solving the AL subproblem under FA, and we focus on this in the next section.

5 Handling Function Approximation

We first describe a framework to handle general FA, and then specialize it to the class of log-linear policies for which we derive theoretical convergence guarantees.

In particular, we consider the general parameterized policy class: $\Pi_\theta := \{\pi \mid \exists\, \theta \text{ s.t. } \pi = \pi_\theta\}$, where π_θ is the policy with learnable parameters θ, and the model parameterizing the policy is implicit in the definition. For instance, this model could be chosen to be a neural network. Given Π_θ, the only modification to our framework is that we now solve the AL subproblem within this parameterized class of policies, i.e., $\max_{\pi\in\Pi_\theta} \mathcal{L}^\beta(\pi, \lambda)$.

In order to do so, we note that the PG for the AL sub-problem can be calculated similar to that in Section 4, i.e. if $Q^\pi_{R(\pi)}$ is the state-action value function for the pseudo-reward $R(\pi)$,

$$\nabla_\theta \mathcal{L}^\beta(\pi_\theta) = \sum_{s\in\mathcal{S}} \frac{d^\pi(s)}{1-\gamma} \sum_{a\in\mathcal{A}} Q^\pi_{R(\pi)}(s,a) \frac{d\pi_\theta(a|s)}{d\theta} \; ; R(\pi) = r - \beta \sum_{i=1}^m c_i \min\left\{ V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0 \right\}.$$

Note the pseudo-reward $R(\pi)$ decomposes linearly into the original reward and constraint components. Consequently, its state–action value function $Q^\pi_{R(\pi)}$ admits to the following decomposition:

$Q^\pi_{R(\pi)} = Q^\pi_r - \beta \sum_{i=1}^m Q^\pi_{c_i} \min\Big\{V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\Big\}$. As a result, $Q^\pi_{R(\pi)}$ can be estimated using approximations of Q^π_r, $Q^\pi_{c_i}$, and $V^\pi_{c_i}(\rho)$. Hence, the same estimation tools employed in standard RL algorithms can be used to implement this gradient. This flexibility allows common PG methods, such as `PPO`, `TRPO` [Schulman et al., 2015], or `SPMA` to be used to solve the `AL` subproblem.

Following Section 4, we use policy optimization to solve the `AL` problem. Inspired by Vaswani et al. [2021], Asad et al. [2024], we adopt a surrogate optimization approach for updating the parameterized policy. In particular, if K is the total number of policy optimization iterations, at iteration $k \in [K]$, we first construct an intermediate policy using a generic PG method: $\pi_{k+1/2} = \texttt{PGMethod}(\pi_k, Q^{\pi_k}_{R(\pi_k)})$. We refer to the policy $\pi_{k+1/2}$ as the idealized policy update if there were no restrictions on the policy class. Note that we can use a generic policy optimization method such as `PQA`, `NPG` or `SPMA` to produce this intermediate policy. In the presence of FA, $\pi_{k+1/2}$ may not lie in the set of feasible policies. As a result, we project $\pi_{k+1/2}$ onto the set of feasible policies using the forward KL divergence and solve the following optimization problem: $\min_{\pi\in\Pi_\theta} \sum_{s\in\mathcal{S}} d^{\pi_k}(s)\,\mathrm{KL}(\pi_{k+1/2}(\cdot|s)\|\pi(\cdot|s))$.

We use the forward KL divergence due to its mode-covering behavior which encourages exploration resulting in better practical performance. Note that the subsequent framework does not depend on this choice, and we could use the reverse KL or any other f-divergence. Solving the above optimization problem exactly is typically intractable since π_θ is typically non-convex. Following [Vaswani et al., 2021, Asad et al., 2024, 2025, Tomar et al., 2020, Lavington et al., 2023], we transform this projection problem into an unconstrained optimization and form the following surrogate as a function of θ,

$$\ell_k(\theta) = \sum_{s\in\mathcal{S}} d^{\pi_k}(s)\,\mathrm{KL}(\pi_{k+1/2}(\cdot|s)\|\pi_\theta(\cdot|s)) \;;\; \theta_{k+1} \approx \arg\min_\theta \ell_k(\theta) \;;\; \pi_{k+1} = \pi_{\theta_{k+1}}\,. \tag{8}$$

In practice, this minimization is carried out approximately using a finite number of gradient steps.

Remark. For large problems where $d^{\pi_k}(s)$ cannot be computed exactly, the surrogate $\ell_k(\theta)$ can be efficiently approximated using sampling. In particular, following Vaswani et al. [2021], Asad et al. [2024], we can generate trajectories by rolling out policy π_k. In particular, for each encountered state, we compute $\pi_{k+1/2}(\cdot|s)$ (using Eq. (7)) and the empirical loss $\mathbb{E}_{a\sim\pi_{k+1/2}(\cdot|s)}[-\log\pi_\theta(a|s)]$ on the fly. This loss is obtained by dropping terms independent of θ from surrogate (see Lemma 13 for the derivation), which does not affect its minimization.

This framework supports off-policy updates [Vaswani et al., 2021] similar to `TRPO` and `PPO`, allowing us to reuse the data gathered by policy π_k and perform multiple parameter updates. Next, we specialize this general framework to log-linear policies [Agarwal et al., 2021, Yuan et al., 2022a].

5.1 Log-Linear Policies

Given features $\varphi \in \mathbb{R}^{SA\times d}$ and parameters $\theta \in \mathbb{R}^d$ with $d \ll SA$, a log-linear policy is parameterized as: $\pi_\theta(a|s) = \frac{\exp(\langle\varphi(s,a),\theta\rangle)}{\sum_{a'}\exp(\langle\varphi(s,a'),\theta\rangle)}$. We instantiate the intermediate policy using the `PQA` update, i.e. $\pi_{k+1/2}(\cdot|s) = \mathrm{Proj}_{\Delta_A}[\pi_k(\cdot|s) + \eta\, Q^{\pi_k}_{R(\pi_k)}(s,\cdot)]$. For log-linear policies, the surrogate in Eq. (8) corresponds to a multi-class logistic regression classification with soft labels enabling efficient optimization even in large state–action spaces [Asad et al., 2024]. Minimizing this surrogate carries out the forward-KL projection, yielding the projected `PQA` update (`PPQA`). The corresponding pseudo-code (Algorithm 4) is provided in Appendix E.

Following Agarwal et al. [2021], Asad et al. [2024], in order to control the projection error induced by approximate minimization of $\ell_k(\theta)$, we make the following assumptions:

Assumption 5 (Bias)**.** *For all* $k \geq 1$, $\min_\theta \ell_k(\theta) \leq \epsilon_{\text{bias}}$

Assumption 6 (Optimization Error)**.** *Suppose* θ_{k+1} *is obtained by approximately minimizing* $\ell_k(\theta)$*, then,* $|\ell_k(\theta_{k+1}) - \min_\theta \ell_k(\theta)| \leq \epsilon_{\text{opt}}$.

In Assumption 5, ϵ_{bias} is determined by the expressivity of the given features. For example, in the case when $d = SA$, and the features are "one-hot" vectors, $\epsilon_{\text{bias}} = 0$. Assumption 6 accounts for the optimization error arising from minimizing the surrogate using a finite number of gradient steps. This assumption is common in the analysis of PG methods with function approximation [Asad et al., 2024, Agarwal et al., 2021, Yuan et al., 2022b, Ding et al., 2023]. In particular, if n steps of gradient descent are used to minimize an convex, L-smooth function $\ell_k(\theta)$ (Lemma 13) , we have $\epsilon_{\text{opt}} = \mathcal{O}(1/n)$.

Putting everything together, we instantiate `Oracle-AL` in Algorithm 1 using the `PPQA` update, and refer to the resulting algorithm as `PPQA-ALM`. We characterize the convergence rate of this algorithm in the following corollary.

Corollary 2. *In Algorithm 1, consider instantiating `Oracle-AL` with the `PPQA` update in Algorithm 4 with the appropriate step-size and number of iterations. Under Assumptions 1 and 4 to 6 such that $\epsilon_{\text{bias}} \leq \mathcal{O}(1/\varepsilon^{16})$, for a given $\varepsilon \in (0,1)$, Algorithm 1 with $T = \mathcal{O}(1/\varepsilon^2)$, $\epsilon_t = \mathcal{O}(1/t^2)$, $\beta > 0$, $\lambda_1 = 0$ returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1 after $\mathcal{O}(1/\varepsilon^6)$ gradient evaluations.*

Provided that ϵ_{bias} is sufficiently small, the above corollary proves that `PPQA-ALM` attains last-iterate convergence at an $\mathcal{O}(1/\varepsilon^6)$ rate.

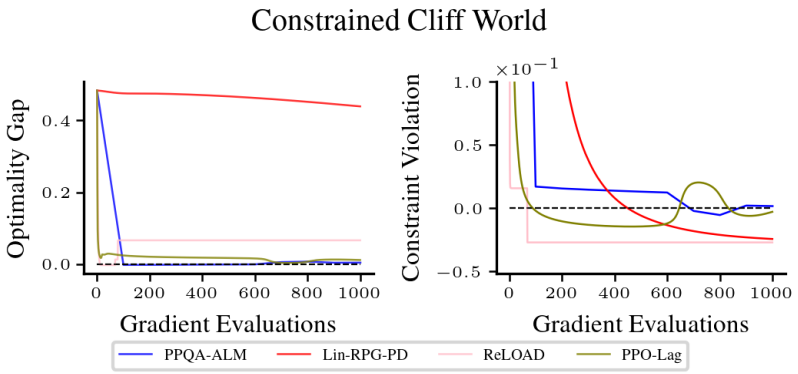

Figure 3: Comparing `PPQA-ALM` to `Lin-RPG-PD`, `ReLoad` and `PPO-Lag` with log-linear policies. `Lin-RPG-PD` has comparable theoretical guarantees in this setting, whereas `ReLOAD` and `PPO-Lag` are included as a competitive empirical benchmark. Following Asad et al. [2024] we use tile-coded features for construct φ with $d = 60 < SA$ (see Appendix H for additional details). We find that `PPQA-ALM` reliably converges to the optimal policy, whereas both `Lin-RPG-PD` and `ReLOAD` perform poorly. While `PPO-Lag` achieves comparable performance, the method lacks any theoretical guarantees.

Comparison to prior work: Ding et al. [2023] analyze a regularized `NPG` primal-dual method for log-linear policies (`Lin-RPG-PD`) and establish a last-iterate $\mathcal{O}(1/\varepsilon^6)$ rate. However, their analysis relies on several restrictive structural assumptions. First, the policy is required to be projected on to a restricted simplex (a non-standard impractical modification). As further noted by Müller et al. [2024], their guarantee only applies to the single constraint case. Second, to handle function approximation, their analysis relies on the compatible function approximation assumption [Agarwal et al., 2021]. Under this assumption, the natural policy gradient direction is obtained by solving a regression problem at every iteration. As a result, a critic is re-estimated after each policy update and is tightly coupled to the policy parameterization. In contrast, our method does not require such structure. The surrogate depends only on value estimates, allowing the use of practical critic estimation methods and supports off-policy updates. More recently, by additionally imposing a Fisher non-degeneracy assumption [Zhan et al., 2023], Montenegro et al. [2024] establish a last-iterate rate of $\tilde{\mathcal{O}}(1/\varepsilon^6)$, which is subsequently improved to $\mathcal{O}(1/\varepsilon^4)$ by Mondal and Aggarwal [2024]. However, this assumption, fails to hold for common policy parameterizations such as log-linear policies [Ni and Kamgarpour, 2023, Proposition C.3]. *In conclusion, our method provides the first last-iterate convergence for CMDPs under log-linear policies without further restrictive assumptions about the policy class.*

6 Conclusion

We developed a policy optimization framework with the `AL` method for CMDPs that provides last-iterate guarantees. In the tabular setting, we prove the framework attains an $\mathcal{O}(1/\varepsilon^6)$ last-iterate rate and under standard assumptions, a comparable rate under linear function approximation. The resulting approach is both principled and practical, demonstrating competitive performance in tabular environments as well as in large-scale continuous-control tasks. The main limitation of our result is the strong dependence on $\rho_{\min}$ in the tabular setting and ϵ_{bias} for log-linear policies. Future directions include weakening these dependencies and incorporating actor–critic or off-policy methods into the theoretical framework.

7 Acknowledgments

The authors would like to thank Ahmet Alacaoglu for providing feedback and ideas for an earlier version of this work. This work is partially supported by the Natural Science and Engineering Research Council of Canada (NSERC) Grant RGPIN-2022-04816 and the Canada CIFAR AI Chairs programs.

8 Bibliography

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pages 22–31. PMLR, 2017.

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

Eitan Altman. *Constrained Markov decision processes*. Routledge, 2021.

Reza Asad, Reza Babanezhad Harikandeh, Issam H Laradji, Nicolas Le Roux, and Sharan Vaswani. Fast convergence of softmax policy mirror ascent. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2024.

Reza Asad, Reza Babanezhad, and Sharan Vaswani. Revisiting actor-critic methods in discrete action off-policy reinforcement learning. *arXiv preprint arXiv:2509.09838*, 2025.

Qinbo Bai, Amrit Singh Bedi, Mridul Agarwal, Alec Koppel, and Vaneet Aggarwal. Achieving zero constraint violation for constrained reinforcement learning via primal-dual approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3682–3689, 2022.

Anas Barakat, Ilyas Fatkhullin, and Niao He. Reinforcement learning with general utilities: Simpler variance reduction and large state-action space. In *International Conference on Machine Learning*, pages 1753–1800. PMLR, 2023.

Anas Barakat, Souradip Chakraborty, Peihong Yu, Pratap Tokekar, and Amrit Singh Bedi. On the global optimality of policy gradient methods in general utility reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2024.

Dimitri P Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

Jalaj Bhandari and Daniel Russo. On the linear convergence of policy gradient methods for finite mdps. In *International Conference on Artificial Intelligence and Statistics*, pages 2386–2394. PMLR, 2021.

Ernesto G Birgin and José Mario Martínez. *Practical augmented Lagrangian methods for constrained optimization*. SIAM, 2014.

Archana Bura, Aria HasanzadeZonuzy, Dileep Kalathil, Srinivas Shakkottai, and Jean-Francois Chamberland. Dope: Doubly optimistic and pessimistic exploration for safe reinforcement learning. *Advances in neural information processing systems*, 35:1047–1059, 2022.

Chiara Buratti, Andrea Conti, Davide Dardari, and Roberto Verdone. An overview on wireless sensor networks technology and evolution. *Sensors*, 9(9):6869–6896, 2009.

Juntao Dai, Jiaming Ji, Long Yang, Qian Zheng, and Gang Pan. Augmented proximal policy optimization for safe reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7288–7295, 2023.

Kangkang Deng, Rui Wang, Zhenyuan Zhu, Junyu Zhang, and Zaiwen Wen. The augmented lagrangian methods: Overview and recent advances. *arXiv preprint arXiv:2510.16827*, 2025.

Dongsheng Ding, Kaiqing Zhang, Tamer Basar, and Mihailo Jovanovic. Natural policy gradient primal-dual method for constrained markov decision processes. *Advances in Neural Information Processing Systems*, 33:8378–8390, 2020.

Dongsheng Ding, Chen-Yu Wei, Kaiqing Zhang, and Alejandro Ribeiro. Last-iterate convergent policy gradient primal-dual methods for constrained mdps. *Advances in Neural Information Processing Systems*, 36:66138–66200, 2023.

Egor Gladin, Maksim Lavrik-Karmazin, Karina Zainullina, Varvara Rudenko, Alexander Gasnikov, and Martin Takac. Algorithm for constrained markov decision process with linear convergence. In *International Conference on Artificial Intelligence and Statistics*, pages 11506–11533. PMLR, 2023.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691. PMLR, 2019.

Arushi Jain, Sharan Vaswani, Reza Babanezhad, Csaba Szepesvari, and Doina Precup. Towards painless policy optimization for constrained mdps. In *Uncertainty in Artificial Intelligence*, pages 895–905. PMLR, 2022.

Jiaming Ji, Borong Zhang, Jiayi Zhou, Xuehai Pan, Weidong Huang, Ruiyang Sun, Yiran Geng, Yifan Zhong, Josef Dai, and Yaodong Yang. Safety gymnasium: A unified safe reinforcement learning benchmark. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=WZmlxIuIGR.

Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024.

David Julian, Mung Chiang, Daniel O'Neill, and Stephen Boyd. Qos and fairness constrained convex optimization of resource allocation for wireless cellular and ad hoc networks. In *Proceedings. twenty-first annual joint conference of the IEEE computer and communications societies*, volume 2, pages 477–486. IEEE, 2002.

Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Navdeep Kumar, Kaixin Wang, Utkarsh Pratiush, Kfir Yehuda Levy, and Shie Mannor. Policy gradient for reinforcement learning with general utilities. In *The Second Tiny Papers Track at ICLR 2024*, 2024.

Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, 198(1):1059–1106, 2023.

Jonathan Wilder Lavington, Sharan Vaswani, Reza Babanezhad Harikandeh, Mark Schmidt, and Nicolas Le Roux. Target-based surrogates for stochastic optimization. In *International Conference on Machine Learning*, pages 18614–18651. PMLR, 2023.

Jingqi Li, David Fridovich-Keil, Somayeh Sojoudi, and Claire J Tomlin. Augmented lagrangian method for instantaneously constrained reinforcement learning problems. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2982–2989. IEEE, 2021.

Jingwang Li and Vincent Lau. Smoothness of the augmented lagrangian dual in convex optimization. *arXiv preprint arXiv:2505.01824*, 2025.

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International conference on machine learning*, pages 6083–6093. PMLR, 2020.

Jiacai Liu, Wenye Li, Dachao Lin, Ke Wei, and Zhihua Zhang. On the convergence of projected policy gradient for any constant step sizes. *Journal of Machine Learning Research*, 26(193):1–35, 2025a.

Xingtu Liu, Lin F Yang, and Sharan Vaswani. Sample complexity bounds for linear constrained mdps with a generative model. *arXiv preprint arXiv:2507.02089*, 2025b.

Ya-Feng Liu, Xin Liu, and Shiqian Ma. On the nonergodic convergence rate of an inexact augmented lagrangian framework for composite convex programming. *Mathematics of Operations Research*, 44(2):632–650, 2019.

Michael Lu, Matin Aghaei, Anant Raj, and Sharan Vaswani. Towards principled, practical policy gradient for bandits and tabular mdps. In *Reinforcement Learning Conference*, 2024.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International conference on machine learning*, pages 6820–6829. PMLR, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

Washim U Mondal and Vaneet Aggarwal. Sample-efficient constrained reinforcement learning with general parameterization. *Advances in Neural Information Processing Systems*, 37:68380–68405, 2024.

Washim Uddin Mondal and Vaneet Aggarwal. Last-iterate convergence of general parameterized policies in constrained MDPs. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=JedrMCZC6l.

Alessandro Montenegro, Marco Mussi, Matteo Papini, and Alberto Maria Metelli. Last-iterate global convergence of policy gradients for constrained reinforcement learning. *Advances in Neural Information Processing Systems*, 37:126363–126416, 2024.

Ted Moskovitz, Brendan O'Donoghue, Vivek Veeriah, Sebastian Flennerhag, Satinder Singh, and Tom Zahavy. Reload: Reinforcement learning with optimistic ascent-descent for last-iterate convergence in constrained mdps. In *International Conference on Machine Learning*, pages 25303–25336. PMLR, 2023.

Adrian Müller, Pragnya Alatur, Giorgia Ramponi, and Niao He. Cancellation-free regret bounds for lagrangian approaches in constrained markov decision processes. *arXiv preprint arXiv:2306.07001*, 2023.

Adrian Müller, Pragnya Alatur, Volkan Cevher, Giorgia Ramponi, and Niao He. Truly no-regret learning in constrained mdps. In *International Conference on Machine Learning*, pages 36605–36653. PMLR, 2024.

Tingting Ni and Maryam Kamgarpour. A safe exploration approach to constrained markov decision processes. *arXiv preprint arXiv:2312.00561*, 2023.

Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepesvari, Satinder Singh, et al. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568*, 2019.

Michael JD Powell. A method for nonlinear constraints in minimization problems. *Optimization*, pages 283–298, 1969.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 7(1):2, 2019.

R Tyrrell Rockafellar. Augmented lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of operations research*, 1(2):97–116, 1976.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

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

Ming Shi, Yingbin Liang, and Ness Shroff. A near-optimal algorithm for safe reinforcement learning under instantaneous hard constraints. In *International Conference on Machine Learning*, pages 31243–31268. PMLR, 2023.

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

Manan Tomar, Lior Shani, Yonathan Efroni, and Mohammad Ghavamzadeh. Mirror descent policy optimization. In *International Conference on Learning Representations*, 2020.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024.

Victor Uc-Cetina, Nicolás Navarro-Guerrero, Anabel Martin-Gonzalez, Cornelius Weber, and Stefan Wermter. Survey on reinforcement learning for language processing. *Artificial Intelligence Review*, 56(2):1543–1575, 2023.

Sharan Vaswani, Olivier Bachem, Simone Totaro, Robert Mü ller, Shivam Garg, Matthieu Geist, Marlos C Machado, Pablo Samuel Castro, and Nicolas Le Roux. A general class of surrogate functions for stable and efficient reinforcement learning. *arXiv preprint arXiv:2108.05828*, 2021.

Sharan Vaswani, Amirreza Kazemi, Reza Babanezhad Harikandeh, and Nicolas Le Roux. Decision-aware actor-critic with function approximation and theoretical guarantees. *Advances in Neural Information Processing Systems*, 36:66451–66498, 2023.

Lin Xiao. On the convergence rates of policy gradient methods. *Journal of Machine Learning Research*, 23(282):1–36, 2022.

Yangyang Xu. Iteration complexity of inexact augmented lagrangian methods for constrained convex programming. *Mathematical Programming*, 185:199–244, 2021.

Donghao Ying, Yuhao Ding, and Javad Lavaei. A dual approach to constrained markov decision processes with entropy regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 1887–1909. PMLR, 2022.

Rui Yuan, Simon S Du, Robert M Gower, Alessandro Lazaric, and Lin Xiao. Linear convergence of natural policy gradient methods with log-linear policies. *arXiv preprint arXiv:2210.01400*, 2022a.

Rui Yuan, Robert M. Gower, and Alessandro Lazaric. A general sample complexity analysis of vanilla policy gradient, 2022b.

Wenhao Zhan, Shicong Cen, Baihe Huang, Yuxin Chen, Jason D Lee, and Yuejie Chi. Policy mirror descent for regularized reinforcement learning: A generalized framework with linear convergence. *SIAM Journal on Optimization*, 33(2):1061–1091, 2023.

Junyu Zhang, Alec Koppel, Amrit Singh Bedi, Csaba Szepesvari, and Mengdi Wang. Variational policy gradient method for reinforcement learning with general utilities. *Advances in Neural Information Processing Systems*, 33:4572–4583, 2020.

Junyu Zhang, Chengzhuo Ni, Csaba Szepesvari, Mengdi Wang, et al. On the convergence and sample efficiency of variance-reduced policy gradient method. *Advances in Neural Information Processing Systems*, 34:2228–2240, 2021.

Supplementary material

A Definitions **16**

B Proofs for the Augmented Lagrangian Policy Gradient Method **16**

B.1 Tabular Setting 17

B.2 Linear Function Approximation Setting 17

C Proof of Technical Tools **18**

C.1 Reduction from Inequality to Equality Constraints 18

C.2 The Augmented Lagrangian Method on the Linear Program View of CMDPs . . . 20

C.3 The Augmented Lagrangian is a General Utility 21

D Proofs and Technical Details for the Augmented Lagrangian Method **23**

D.1 Equality Constraint Problems 23

D.2 Additional Lemmas 28

E Proofs for Projected Q-Ascent on General Utilities **30**

E.1 Tabular Setting 30

E.2 Linear Functional Approximation Setting 33

E.3 Supporting Lemmas 38

E.4 Additional Lemmas for the Augmented Lagrangian General Utility 39

F Large-Scale Experiments on Continuous Control Tasks **43**

F.1 Key Results 43

F.2 Experimental Details 44

F.3 Pseudo-Code for SPMA-ALM 45

F.4 Sensitivity Analysis Of The Final Policy 46

F.5 Effect of the Number of AL Iterations and Penalty β 47

F.6 Effect of the Number of AL Iterations and Dual Step-size 48

G Tabular Experiments **49**

G.1 Environmental Details 49

G.2 Experimental Details & Additional Results 50

H Linear Function Approximation Experiments **51**

I External Results **55**

A Definitions

[Smoothness] A differentiable function f is L-smooth if for all v and w

$$|f(v) - f(w) - \langle \nabla f(w), v - w \rangle| \leq \frac{L}{2} \|v - w\|_2^2.$$

[Lipschitz continuity] A function f is B-Lipschitz if for all v and w

$$|f(v) - f(w)| \leq B \|v - w\|_2.$$

[Convexity] A function f is convex if for all v and w

$$f(v) - f(w) \geq \langle \nabla f(w), v - w \rangle.$$

B Proofs for the Augmented Lagrangian Policy Gradient Method

We first state a meta-theorem describing the sub-optimality guarantees of Algorithm 1, assuming access to an oracle `Oracle-AL` that solves the augmented Lagrangian (`AL`) subproblem.

Theorem 1. *Under Assumption 4, assuming for all $t \geq 1$, there exists an solver `Oracle-AL`$(\mathcal{L}^\beta, \lambda_t, \epsilon_t)$ which returns a policy π_{t+1} satisfying Eq. (3), for a given $\varepsilon \in (0,1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \mathcal{O}(1/t^2)$, $\beta > 0$, and $\lambda_1 = 0$ returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1.*

Proof. To prove this result, we reduce finding an approximate solution to Problem 1 to an equivalent linear program (LP) (see Appendix C.1 for additional details) . Using the standard occupancy measure formulation of CMDPs [Altman, 2021], we have

$$\max_{\substack{\mu^\pi \in \mathcal{K}, \\ z \in \mathbb{R}_+^m}} \langle \mu^\pi, r \rangle \quad \text{s.t.} \quad \langle \mu^\pi, c_i \rangle - z_i = b_i \quad \forall i \in [m] \tag{9}$$

where $z \in \mathbb{R}_+^m$ are non-negative slack variables. and

$$\mathcal{K} := \left\{ \mu^\pi \in \mathbb{R}_+^{SA} \mid \forall s' \in \mathcal{S}, \sum_{a' \in \mathcal{A}} \mu^\pi(s', a') = \gamma \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} \mathcal{P}(s'|s, a)\, \mu^\pi(s, a) + \rho(s') \right\}$$

represents the set of all feasible state-action occupancy measures satisfying the Bellman flow constraints. According to Lemmas 3 and 4 , finding an ε-approximate solution to the LP in Problem 3 in terms of μ^π induces a policy π that is an ε-approximate solution in Problem 1.

Next, we relate the oracle guarantee in Algorithm 1 to the `AL` method applied to Problem 3. For a fixed outer iteration $t \in [T]$ and target $\epsilon_t > 0$, by the oracle guarantee:

$$\mathcal{L}^\beta(\pi_{t+1}, \lambda_t) \geq \max_\pi \mathcal{L}^\beta(\pi, \lambda_t) - \epsilon_t.$$

Using the mapping between μ^π and π, given by $\pi(a|s) = \frac{\mu^\pi(s,a)}{\sum_{a'} \mu^\pi(s,a')}$, the state-action occupancy $\mu^{\pi_{t+1}}$ satisfies:

$$F_{\beta,\lambda_t}(\mu^{\pi_{t+1}}) \geq \max_{\mu^\pi \in \mathcal{K}} F_{\beta,\lambda_t}(\mu^\pi) - \epsilon_t,$$

where

$$F_{\beta,\lambda}(\mu^\pi) := \langle \mu^\pi, r \rangle + \sum_{i=1}^m \frac{\beta}{2} \left[-\left(\min\left\{ \langle \mu^\pi, c_i \rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\} \right)^2 + \frac{\lambda_i^2}{\beta^2} \right].$$

As a result, by Lemmas 5 and 7, Algorithm 1 with $T > 0$, $\epsilon_t = \frac{\sigma}{t^2}$, for a $\sigma > 0$, $\beta > 0$, $\lambda_1 = 0$ returns a policy π_{T+1} such that,

$$V_r^*(\rho) - V_r^{\pi_{T+1}}(\rho) = \langle \mu^* - \mu^{\pi_{T+1}}, r \rangle \leq \frac{2\,B}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T} \right) + \frac{\sigma}{T^2},$$

$$b_i - V_{c_i}^{\pi_{T+1}}(\rho) = b_i - \langle \mu^{\pi_{T+1}}, c_i \rangle \leq \frac{2}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T} \right) \quad \forall i \in [m]$$

where the constants B, C defined in Lemma 5. Hence, for any $\varepsilon \in (0,1)$, setting $T = \mathcal{O}(1/\varepsilon^2)$ implies that

$$V_r^*(\rho) - V_r^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad ; \quad b_i - V_{c_i}^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad \forall i \in [m].$$

□

B.1 Tabular Setting

Lemma 1. *Under Assumptions 1 and 4, for fixed outer-iteration $t \geq 1$, dual variable $\lambda_t \in \mathbb{R}^m$, and penalty parameter $\beta > 0$, let $\mathcal{L}^\beta(\pi, \lambda_t)$ denote the* AL *defined in Eq. (2) at iteration t. Given target error $\epsilon_t > 0$ using the* PQA *update in Eq. (39) with step-size $\eta = \frac{\rho_{\min}}{L_t}$ results in a policy that satisfies Eq. (3) after $K_t = \left\lceil \max\left\{2, \frac{32\, L_t\, (1+1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t}\right\}\right\rceil$ iterations.*

Proof. For a fixed outer-iteration $t \geq 1$,dual variable $\lambda_t \in \mathbb{R}^m$, and penalty parameter $\beta > 0$ by Proposition 1, the AL $\mathcal{L}^\beta(\pi, \lambda_t)$ is a concave general utility and by Lemma 16, $\mathcal{L}^\beta(\pi, \lambda_t)$ is L_t-smooth.

Therefore, under Assumption 1, by Theorem 4 the PQA update in Eq. (39) using a step-size $\eta = \frac{\rho_{\min}}{L_t}$ with the general utility $G(\pi) = \mathcal{L}^\beta(\pi, \lambda_t)$ for K_t inner-iterations results in the following convergence:

$$\max_{\pi \in \Pi} \mathcal{L}^\beta(\pi, \lambda_t) - \mathcal{L}^\beta(\pi_{K_t}, \lambda_t) \leq \frac{32\, L_t\, (1 + 1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, K_t}$$

Hence, using $K_t = \left\lceil \max\left\{2, \frac{32\, L_t\, (1+1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t}\right\}\right\rceil$ iterations results in the policy $\pi_{t+1} = \pi_{K_t}$ satisfying the oracle condition in Eq. (3). □

Corollary 1. *Under Assumptions 1 and 4, for a given $\varepsilon \in (0, 1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \mathcal{O}\left(1/t^2\right)$, $\beta > 0$, $\lambda_1 = 0$ and instantiating* Oracle-AL *using the* PQA *update in Eq. (7) with $\eta = \frac{\rho_{\min}}{L_t}$ for $K_t = \left\lceil \frac{32\, L_t\, (1+1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t}\right\rceil$ iterations returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1 after $\mathcal{O}\left(1/\varepsilon^6\right)$ gradient evaluations.*

Proof. Under Assumption 1 for each outer-iteration $t \geq 1$, by Lemma 1, the PQA update with step-size $\eta_t = \frac{\rho_{\min}}{L_t}$ returns a policy π_{t+1} that satisfies the oracle condition in Eq. (3) after

$$K_t = \left\lceil \max\left\{2, \frac{32\, L_t\, (1 + 1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t}\right\}\right\rceil$$

iterations where L_t is defined in Lemma 16. As a result, by Theorem 1, for a given $\varepsilon \in (0, 1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \frac{\sigma}{t^2}$, for any $\sigma > 0$, $\beta > 0$, $\lambda_1 = 0$ returns policy π_{T+1} that satisfies

$$V_r^*(\rho) - V_r^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad \text{and} \quad b_i - V_{c_i}^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad \forall i \in [m].$$

To characterize the total number of gradient evaluations note under Assumption 4, we have $L_t \leq \bar{L}$ for all $t \geq 1$, where $\bar{L} = \frac{2\gamma A}{(1-\gamma)^3}\, U + \frac{\beta m A}{(1-\gamma)^4}$ and U is defined in Lemma 17. Hence, for all $t \geq 1$,

$$K_t \leq \frac{32\, \bar{L}\, (1 + 1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2\, \epsilon_t} + 2 = \frac{\bar{K}\, t^2}{\sigma} + 2 \quad ; \qquad \bar{K} := \frac{32\, \bar{L}\, (1 + 1/(1-\gamma)\, \rho_{\min})}{(1-\gamma)^2\, \rho_{\min}^2}, \tag{10}$$

Hence, summing over all $t \in [T]$ yields that

$$\sum_{t=1}^{T} K_t = \sum_{t=1}^{T} \mathcal{O}\left(t^2\right) = \mathcal{O}(T^3) = \mathcal{O}(1/\varepsilon^6).$$

□

B.2 Linear Function Approximation Setting

Lemma 2. *Under Assumptions 1 and 2, for a fixed outer-iteration $t \geq 1$, dual variable $\lambda_t \in \mathbb{R}^m$, and penalty parameter $\beta > 0$, let $\mathcal{L}^\beta(\pi, \lambda_t)$ denote the* AL *defined in Eq. (2) at iteration t. Given target $\epsilon_t > 0$ and assuming under Assumptions 5 and 6, $\epsilon_{\text{bias}} + \epsilon_{\text{opt}} \leq \mathcal{O}\left(\frac{1}{K_t^4}\right)$ using the* PPQA *update in Algorithm 4 with step-size $\eta = \frac{\rho_{\min}}{L_t}$ results in a policy that satisfies Eq. (3) after $K_t = \mathcal{O}(1/\epsilon_t)$ iterations.*

Proof. For a fixed outer-iteration $t \geq 1$, dual variable $\lambda_t \in \mathbb{R}^m$, and penalty parameter $\beta > 0$ by Proposition 1, the AL $\mathcal{L}^\beta(\pi, \lambda_t)$ is a concave general utility and by Lemma 16, $\mathcal{L}^\beta(\pi, \lambda_t)$ is L_t-smooth.

Hence, under Assumptions 1, 5 and 6, according to Theorem 5, the PPQA update in Algorithm 4 with $\eta = \frac{\rho_{\min}}{L_t}$, with the general utility $G(\pi) = \mathcal{L}^\beta(\pi, \lambda_t)$ results in the following convergence:

$$\max_\pi \mathcal{L}^\beta(\pi, \lambda_t) - \mathcal{L}^\beta(\pi_{K_t}, \lambda_t) \leq \underbrace{\frac{4\, C_t}{K_t}}_{\text{policy optimization error}} + \underbrace{2\, C_t \sqrt{\frac{\sqrt{2}\, U \sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}}}{C_t\, (1-\gamma)^3\, \rho_{\min}}}}_{\text{projection error}}$$

where $C_t := \frac{16\, L_t}{(1-\gamma)^2 \rho_{\min}^2}(1 + \frac{1}{(1-\gamma)\rho_{\min}})$. Assuming that $\epsilon_{\text{bias}} + \epsilon_{\text{opt}} \leq \mathcal{O}\Big(\frac{1}{K_t^4}\Big)$, we have

$$\max_\pi \mathcal{L}^\beta(\pi, \lambda_t) - \mathcal{L}^\beta(\pi_{K_t}, \lambda_t) \leq \mathcal{O}\left(\frac{1}{K_t}\right).$$

Hence, using $K_t = \mathcal{O}\Big(\frac{1}{\epsilon_t}\Big)$ iterations results a policy $\pi_{t+1} = \pi_{K_t}$ that satisfies the oracle condition in Eq. (3). □

Corollary 2. *In Algorithm 1, consider instantiating* Oracle-AL *with the* PPQA *update in Algorithm 4 with the appropriate step-size and number of iterations. Under Assumptions 1 and 4 to 6 such that $\epsilon_{\text{bias}} \leq \mathcal{O}(1/\varepsilon^{16})$, for a given $\varepsilon \in (0,1)$, Algorithm 1 with $T = \mathcal{O}(1/\varepsilon^2)$, $\epsilon_t = \mathcal{O}(1/t^2)$, $\beta > 0$, $\lambda_1 = 0$ returns a stochastic policy π_{T+1} that is an ε-approximate solution to Problem 1 after $\mathcal{O}(1/\varepsilon^6)$ gradient evaluations.*

Proof. To start, note that under Assumptions 5 and 6 assuming $\epsilon_{\text{bias}} + \epsilon_{\text{opt}} \leq \mathcal{O}\big(\frac{1}{T^8}\big)$ we have,

$$\begin{aligned} \epsilon_{\text{bias}} + \epsilon_{\text{opt}} &\leq \mathcal{O}\left(\frac{1}{T^8}\right) \leq \mathcal{O}\left(\frac{1}{t^8}\right) && \text{(For any } t \leq T\text{)} \\ &\leq \mathcal{O}\big(\epsilon_t^4\big) && \text{(Since } \epsilon_t = \tfrac{\sigma}{t^2} \text{ for any } \sigma > 0\text{)} \end{aligned}$$

Hence, all $t \in [T]$, with $K_t = \mathcal{O}(1/\epsilon_t)$, $\epsilon_{\text{bias}} + \epsilon_{\text{opt}} \leq \mathcal{O}\big(\epsilon_t^4\big) = \mathcal{O}(1/K_t^4)$.

As a result, under Assumption 1, by Lemma 2 using the PPQA update in Algorithm 4 with step-size $\eta = \frac{\rho_{\min}}{L_t}$, results in a policy that satisfies Eq. (3) after $K_t = \mathcal{O}(1/\epsilon_t)$ iterations. Here, Assumption 2 holds by Lemma 17 under Assumption 4, and $L_t \leq \bar{L}$ for all $t \geq 1$ (see the proof of Corollary 1), so $C_t \leq \bar{C} := \frac{16\, \bar{L}}{(1-\gamma)^2 \rho_{\min}^2}\Big(1 + \frac{1}{(1-\gamma)\rho_{\min}}\Big)$ uniformly in t. Hence the constant hidden in $K_t = \mathcal{O}(1/\epsilon_t)$ does not depend on t. Moreover, since $L_t \geq \frac{\beta m A}{(1-\gamma)^4} > 0$, C_t is bounded away from zero uniformly in t, so the condition $\frac{\sqrt{2}\, U}{(1-\gamma)^3\, \rho_{\min}} \sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}} < C_t$ of Theorem 5 holds once $\epsilon_{\text{opt}} + \epsilon_{\text{bias}}$ is sufficiently small.

By Theorem 1, for a target error $\varepsilon \in (0,1)$, and $T = \mathcal{O}(1/\varepsilon^2)$, Algorithm 1 with $\epsilon_t = \frac{\sigma}{t^2}$, for any $\sigma > 0$, $\beta > 0$, $\lambda_1 = 0$ returns policy π_{T+1} which satisfies that

$$V_r^*(\rho) - V_r^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad ; \quad b_i - V_{c_i}^{\pi_{T+1}}(\rho) \leq \mathcal{O}(\varepsilon) \quad \forall i \in [m].$$

Hence, summing over all $t \in [T]$, the total number of gradient evaluations required is

$$\sum_{t=1}^{T} K_t = \sum_{t=1}^{T} \mathcal{O}\big(t^2\big) = \mathcal{O}(T^3) = \mathcal{O}(1/\varepsilon^6).$$

□

C Proof of Technical Tools

C.1 Reduction from Inequality to Equality Constraints

Recall for a given CMDP, the objective is to solve the following problem

Problem 1. $\max_\pi V_r^\pi(\rho) \quad \textit{subject to} \quad V_{c_i}^\pi(\rho) \geq b_i \quad \forall i \in [m]\,.$

Using the standard linear programming (LP) formulation of CMDPs [Altman, 2021], solving Problem 1 with respect to a policy π is equivalent to solving the following LP in terms of the unnormalized state-action occupancy measure μ^π.

Problem 2. $\max_{\mu^\pi \in \mathcal{K}} \langle \mu^\pi, r \rangle \quad \textit{subject to} \quad \langle \mu^\pi, c_i \rangle \geq b_i \quad \forall i \in [m],$

Here, $\mathcal{K}$ represents the set of all feasible unnormalized state-action occupancy measures satisfying the Bellman flow constraints. Under this representation, both the objective and constraints are now linear with respect to the decision variable μ^π. Furthermore, for any $\mu^\pi \in \mathcal{K}$, each feasible state-action occupancy measure $\mu^\pi \in \mathcal{K}$ admits a corresponding policy π where

$$\pi(a|s) = \frac{\mu^\pi(s,a)}{\sum_{a' \in \mathcal{A}} \mu^\pi(s,a')} \quad \forall (s,a) \in \mathcal{S} \times \mathcal{A}. \tag{11}$$

whenever the denominator is positive, with arbitrary choices at states with zero occupancy. To handle inequality constraints, we introduce slack variables and consider the equivalent equality-constrained LP.

Problem 3. $\max_{\mu^\pi \in \mathcal{K}, z \in \mathbb{R}_+^m} \langle \mu^\pi, r \rangle \quad \textit{subject to} \quad \langle \mu^\pi, c_i \rangle - z_i = b_i \quad \forall i \in [m]$

Hence, for a target error $\varepsilon > 0$, we relate obtaining ε-approximate solutions of Problem 3 to Problem 1 with the following lemmas.

Lemma 3. *For a target error $\varepsilon \in (0,1)$, let $(\mu^\pi, z) \in \mathcal{K} \times \mathbb{R}_+^m$ be an ε-approximate solution to Problem 3, i.e.,*

$$\langle \mu^\pi, r \rangle \geq \langle \mu^*, r \rangle - \varepsilon \quad \textit{and} \quad \langle \mu^\pi, c_i \rangle - z_i \geq b_i - \varepsilon \quad \forall i \in [m]$$

then μ^π is an ε-approximate solution to Problem 2, i.e.,

$$\langle \mu^\pi, r \rangle \geq \langle \mu^*, r \rangle - \varepsilon \quad \textit{and} \quad \langle \mu^\pi, c_i \rangle \geq b_i - \varepsilon \quad \forall i \in [m].$$

where μ^* is the optimal state-action occupancy measure of Problem 2.

Proof. For a target error $\varepsilon \in (0,1)$, let (μ^π, z) be an ε-approximate solution to Problem 3, i.e.,

$$\langle \mu^\pi, r \rangle \geq \langle \mu^*, r \rangle - \varepsilon \quad \text{and} \quad \langle \mu^\pi, c_i \rangle - z_i \geq b_i - \varepsilon \quad \forall i \in [m].$$

Since the slack variables $z_i \geq 0$ for each constraint $i \in [m]$

$$\langle \mu^\pi, c_i \rangle \geq z_i + b_i - \varepsilon \geq b_i - \varepsilon.$$

Hence, μ^π is an ε-approximate solution to to Problem 2. □

Lemma 4. *For a target error $\varepsilon \in (0,1)$, let $\mu^\pi \in \mathcal{K}$ be an ε-approximate solution to Problem 2, Then the policy π corresponding to μ^π is an ε-solution to Problem 1, i.e.,*

$$V_r^\pi(\rho) \geq V_r^*(\rho) - \varepsilon \quad \textit{and} \quad V_{c_i}^\pi(\rho) \geq b_i - \varepsilon \quad \forall i \in [m]$$

where π^* is the optimal policy of Problem 1.

Proof. For a target error $\varepsilon \in (0,1)$, let μ^π be an ε-approximate solution to Problem 2. Using the correspondence between μ^π and π in Eq. (11), the policy π corresponding to μ^π satisfies:

$$\langle \mu^\pi, r \rangle = V_r^\pi(\rho) \quad \text{and} \quad \langle \mu^\pi, c_i \rangle = V_{c_i}^\pi(\rho) \quad \forall i \in [m].$$

Hence,

$$V_r^\pi(\rho) \geq V_r^*(\rho) - \varepsilon \quad \text{and} \quad V_{c_i}^\pi(\rho) \geq b_i - \varepsilon \quad \forall i \in [m]\,,$$

and π is an ε-approximate policy to Problem 1. □

As a result, by chaining Lemmas 3 and 4 together, finding an ε-approximation solution to the LP in Problem 3 in terms of μ^π corresponds to a policy π that is an ε-approximation solution in Problem 1.

Next, we present the `AL` method applied to Problem 3.

C.2 The Augmented Lagrangian Method on the Linear Program View of CMDPs

The augmented Lagrangian (AL) for Problem 3 is defined as:

$$\overline{F}_{\beta,\lambda}(\mu^\pi, z) := \langle \mu^\pi, r\rangle + \sum_{i=1}^m \lambda_i(\langle \mu^\pi, c_i\rangle - b_i - z_i) - \frac{\beta}{2}\sum_{i=1}^m (\langle \mu^\pi, c_i\rangle - b_i - z_i)^2.$$

where $\lambda \in \mathbb{R}^m$ is the Lagrange multiplier associated with the equality constraint and $\beta > 0$ is the penalty parameter. The AL method for solving Problem 3 proceeds by alternating between approximately maximizing the AL and updating the dual variables. At outer-iteration $t \in [T]$, given an target $\epsilon_t \in (0,1)$ and Lagrange multiplier $\lambda_t \in \mathbb{R}^m$, the primal update computes a pair $(\mu^{\pi_{t+1}}, z_{t+1})$ satisfying

$$\overline{F}_{\beta,\lambda_t}(\mu^{\pi_{t+1}}, z_{t+1}) \geq \max_{(\mu^\pi, z)\in \mathcal{K}\times\mathbb{R}_+^m} \overline{F}_{\beta,\lambda_t}(\mu^\pi, z) - \epsilon_t. \tag{12}$$

The dual variables are then updated as, for each constraint $i \in [m]$:

$$\lambda_{t+1}[i] = \lambda_t[i] - \frac{\beta}{2}(\langle \mu^{\pi_{t+1}}, c_i\rangle - (b_i + z_{t+1}[i])).$$

The AL method for Problem 3 is summarized below.

Algorithm 2: The AL Method for Problem 3

1: **Input**: μ^{π_1} (primal variable), $\lambda_1 = 0$ (dual variable), $\beta > 0$ (constant constraint penalty), $\epsilon_t = \frac{\sigma}{t^2}$ (subproblem target for any $\sigma > 0$), T (number of iterations)
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Formulate AL:
$\overline{F}_{\beta,\lambda}(\mu^\pi, z) := \langle \mu^\pi, r\rangle + \sum_{i=1}^m \lambda_i(\langle \mu^\pi, c_i\rangle - b_i - z_i) - \frac{\beta}{2}\sum_{i=1}^m (\langle \mu^\pi, c_i\rangle - b_i - z_i)^2$.
4: Find the approximate solution pair $(\mu^{\pi_{t+1}}, z_{t+1}) \in \mathcal{K}\times\mathbb{R}_+^m$ such that

$$\overline{F}_{\beta,\lambda_t}(\mu^{\pi_{t+1}}, z_{t+1}) \geq \max_{(\mu^\pi, z)\in \mathcal{K}\times\mathbb{R}_+^m} \overline{F}_{\beta,\lambda_t}(\mu^\pi, z) - \epsilon_t.$$

5: $\lambda_{t+1}[i] = \lambda_t[i] - \frac{\beta}{2}(\langle \mu^{\pi_{t+1}}, c_i\rangle - (b_i + z_{t+1}[i]))$ for each constraint $i \in [m]$.
6: **end for**
7: Return $\mu^{\pi_{T+1}}$

Lemma 5. *Under Assumption 4, suppose for each iteration $t \geq 1$, there exists an oracle that returns a pair $(\mu^{\pi_{t+1}}, z_{t+1}) \in \mathcal{K}\times\mathcal{Z}$ such that*

$$\overline{F}_{\beta,\lambda_t}(\mu^{\pi_{t+1}}, z_{t+1}) \geq \max_{(\mu^\pi, z)\in \mathcal{K}\times\mathcal{Z}} \overline{F}_{\beta,\lambda_t}(\mu^\pi, z) - \epsilon_t.$$

where $\mathcal{Z} := \{z \in \mathbb{R}_+^m \mid 0 \leq z_i \leq \frac{1}{1-\gamma} b_i\}$. Then Algorithm 2 with $\epsilon_t = \frac{\sigma}{t^2}$, for any $\sigma > 0$, $\beta > 0$, $\lambda_1 = 0$ converges as: for all $T \geq 1$

$$\langle \mu^* - \mu^{\pi_{T+1}}, r\rangle \leq \frac{2\,B}{\sqrt{\beta}}\left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T}\right) + \frac{\sigma}{T^2},$$

$$b_i - \langle \mu^{\pi_{T+1}}, c_i\rangle \leq \frac{2}{\sqrt{\beta}}\left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T}\right) \quad \forall i \in [m]$$

where $B := 2\,M + \sqrt{\sigma\,\frac{\beta\,\pi^2}{3}}$, $M := \frac{1}{(1-\gamma)\min_{i\in[m]}\zeta_i}$ with ζ_i as in Assumption 4, $\kappa := \frac{\beta}{8B^2}$, $\omega := 3$, $C := \frac{4}{\kappa}\left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa+1)}\right) + 4\max\left\{\max_{(\mu^\pi,z)\in\mathcal{K}\times\mathcal{Z}} \bar{F}_{\beta,\lambda_1}(\mu^\pi, z) - \max_{(\mu^\pi,z)\in\mathcal{K}\times\mathcal{Z}} \bar{F}_{\beta,\lambda^*}(\mu^\pi, z), \frac{4}{\kappa}\right\}$.

Proof. To start, we rewrite Problem 3 in standard form in order to match the analysis for the AL analysis for convex, linearly constrained problems presented in Appendix D.

Define the change of variables

$$x := \begin{bmatrix}\mu^\pi \\ z\end{bmatrix} \quad ; \quad f(x) := -\langle \mu^\pi, r\rangle \quad ; \quad \nu = -\lambda$$

$$A := \begin{bmatrix} c_1^\top & -e_1^\top \\ c_2^\top & -e_2^\top \\ \vdots & \vdots \\ c_m^\top & -e_m^\top \end{bmatrix} \quad ; \quad b := \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix} \quad ; \quad \mathcal{X} = \mathcal{K} \times \mathcal{Z}.$$

where $e_i \in \mathbb{R}^m$ is the ith standard basis vector, $\mathcal{Z} := \left\{ z \in \mathbb{R}^m_+ \mid 0 \le z_i \le \bar{z}_i \right\} \subseteq \mathbb{R}^m_+$ is domain of the non-negative slack variables with $\bar{z}_i := \frac{1}{1-\gamma} + b_i$ for each constraint $i \in [m]$. Under this change of variables, Problem 3 coincides with setting described in Appendix D.

We first verify that the iterates and the optimal solution lie in $\mathcal{X}$. By Lemmas 6 and 7, for all $t \ge 1$

$$z_{t+1} = \xi(\mu^{\pi_{t+1}}) \quad ; \quad [z_{t+1}]_i = \max\left\{ \langle \mu^{\pi_{t+1}}, c_i \rangle - b_i - \frac{[\lambda_t]_i}{\beta}, 0 \right\}.$$

Furthermore by Proposition 3, $[\lambda_t]_i \ge 0$, and since $c_i \in [0,1]^{SA}$, $\langle \mu^\pi, c_i \rangle \le \frac{1}{1-\gamma}$ for all $\mu^\pi \in \mathcal{K}$. Hence $0 \le [z_{t+1}]_i \le \frac{1}{1-\gamma} + b_i = \bar{z}_i$, i.e., $z_{t+1} \in \mathcal{Z}$. Similarly, the optimal slack satisfies $0 \le z_i^* = \langle \mu^*, c_i \rangle - b_i \le \bar{z}_i$, so $(\mu^*, z^*) \in \mathcal{X}$.

Using Theorem 3 and the above mapping, if $\epsilon_t = \frac{\sigma}{t^2}$ for any constant $\sigma > 0$, $\beta > 0$ and $\lambda_1 = 0$, Algorithm 2 achieves the following convergence: for $T \ge 1$,

$$\langle \mu^* - \mu^{\pi_{T+1}}, r \rangle \le \frac{2\,B}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T} \right) + \frac{\sigma}{T^2}, \tag{13}$$

$$\left\| [\langle \mu^{\pi_{T+1}}, c_i \rangle - [z_{T+1}]_i - b_i]_{i=1}^m \right\|_2 \le \frac{2}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T} \right) \tag{14}$$

where $B := 2\,M + \sqrt{\sigma\,\beta\,\frac{\pi^2}{3}}$, $\kappa := \frac{\beta}{8B^2}$, $\omega := 3$,
and $C := \frac{4}{\kappa}\left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa+1)}\right) + 4\max\left\{ \max_{(\mu^\pi, z) \in \mathcal{X}} \bar{F}_{\beta,\lambda_1}(\mu^\pi, z) - \max_{(\mu^\pi, z) \in \mathcal{X}} \bar{F}_{\beta,\lambda^*}(\mu^\pi, z), \frac{4}{\kappa} \right\}$.
Finally to relate Eq. (14) to an individual constraint $i \in [m]$,

$$\begin{aligned}
b_i - \langle \mu^{\pi_{T+1}}, c_i \rangle &= -(\langle \mu^{\pi_{T+1}}, c_i \rangle - b_i) \\
&= -(\langle \mu^{\pi_{T+1}}, c_i \rangle - b_i - [z_{T+1}]_i + [z_{T+1}]_i) && \text{(Add/Subtract } [z_{T+1}]_i\text{)} \\
&\le -(\langle \mu^{\pi_{T+1}}, c_i \rangle - b_i - [z_{T+1}]_i) && \text{(Since } [z_{T+1}]_i \ge 0\text{)} \\
&\le |\langle \mu^{\pi_{T+1}}, c_i \rangle - b_i - [z_{T+1}]_i| \\
&\le \max_{i \in [m]} |\langle \mu^{\pi_{T+1}}, c_i \rangle - b_i - [z_{T+1}]_i| \\
&\le \left\| [\langle \mu^{\pi_{T+1}}, c_i \rangle - [z_{T+1}]_i - b_i]_{i=1}^m \right\| && \text{(Since } \|\cdot\|_\infty \le \|\cdot\|\text{)} \\
\implies b_i - \langle \mu^{\pi_{T+1}}, c_i \rangle &\le \frac{2}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{T}} + \frac{\sqrt{2\,\sigma}}{T} \right) \quad \forall i \in [m] && \text{(Using Eq. (14))}
\end{aligned}$$

which completes the proof. □

C.3 The Augmented Lagrangian is a General Utility

Following Bertsekas [2014], we show that AL in Eq. (5) can first be maximized over the slack variables z. This equivalence allows the AL subproblem only in terms of μ^π, implying that it is a concave general utility function. The following two lemmas formalize this reduction and its implications for approximate solutions.

Lemma 6. *For any $\lambda \in \mathbb{R}^m$, $\beta > 0$ and $\mu^\pi \in \mathcal{K}$,*

$$\max_{z \in \mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) = F_{\beta,\lambda}(\mu^\pi) := \langle \mu^\pi, r \rangle + \sum_{i=1}^m \frac{\beta}{2} \left[-\left(\min\left\{ \langle \mu^\pi, c_i \rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\} \right)^2 + \frac{\lambda_i^2}{\beta^2} \right]$$

Furthermore,

$$\overline{F}_{\beta,\lambda}(\mu^\pi, \xi(\mu^\pi)) = F_{\beta,\lambda}(\mu^\pi).$$

where for any constraint $i \in [m]$,

$$\xi_i(\mu^\pi) = \max\left\{\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\right\}.$$

Proof. This proof follows from Bertsekas [2014, Chapter 3] and is added for completeness.

Fix $\lambda \in \mathbb{R}^m$ and $\mu^\pi \in \mathcal{K}$. Since the AL is separable in the slack variables z_i, we may maximize each coordinate independently:

$$\begin{aligned}\max_{z\in\mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) &= \langle \mu^\pi, r\rangle + \sum_{i=1}^m \lambda_i(\langle \mu^\pi, c_i\rangle - b_i - z_i) - \frac{\beta}{2}\sum_{i=1}^m(\langle \mu^\pi, c_i\rangle - b_i - z_i)^2 \\ &= \langle \mu^\pi, r\rangle + \sum_{i=1}^m \max_{z_i\in\mathbb{R}_+}\left\{\lambda_i\left(u_i - z_i\right) - \frac{\beta}{2}(u_i - z_i)^2\right\} \\ &\qquad\qquad \text{(Let } u_i := \langle \mu^\pi, c_i\rangle - b_i)\end{aligned}$$

For a fixed constraint $i \in [m]$, let

$$g_i(z) := \lambda_i\left(u_i - z_i\right) - \frac{\beta}{2}\left(u_i - z_i\right)^2.$$

Since $g_i(z)$ is concave, we can take its derivative and find the global maximum,

$$z_i^* = \arg\max_{z_i\geq 0} g_i(z) = \max\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} \implies g_i(z_i^*) = \lambda_i\left(u_i - z_i^*\right) - \frac{\beta}{2}(u_i - z_i^*)^2$$

Furthermore, for a fixed constraint $i \in [m]$,

$$\begin{aligned} g_i(z_i^*) &= \lambda_i\left(u_i - \max\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\}\right) - \frac{\beta}{2}\left(u_i - \max\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\}\right)^2 \\ &= \lambda_i\left(u_i + \min\left\{-u_i + \frac{\lambda_i}{\beta}, 0\right\}\right) - \frac{\beta}{2}\left(u_i + \min\left\{-u_i + \frac{\lambda_i}{\beta}, 0\right\}\right)^2 \\ &\qquad\qquad (\max\{a,0\} = -\min\{-a,0\}) \\ &= \lambda_i\left(\min\left\{u_i, \frac{\lambda_i}{\beta}\right\}\right) - \frac{\beta}{2}\left(\min\left\{u_i, \frac{\lambda_i}{\beta}\right\}\right)^2 \\ &= \lambda_i\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{\lambda_i}{\beta}\right) - \frac{\beta}{2}\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{\lambda_i}{\beta}\right)^2 \quad \text{(Add/Subtract } \tfrac{\lambda_i}{\beta}) \\ &= -\frac{\beta}{2}\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{\lambda_i}{\beta}\right)^2 + \lambda_i\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{\lambda_i}{\beta}\right) \\ &= \frac{\beta}{2}\left[-\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{\lambda_i}{\beta}\right)^2 + \frac{2\,\lambda_i}{\beta}\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\} + \frac{2\,\lambda_i^2}{\beta^2}\right] \\ &= \frac{\beta}{2}\left[-\left(\min\left\{u_i - \frac{\lambda_i}{\beta}, 0\right\}\right)^2 + \frac{\lambda_i^2}{\beta^2}\right] \qquad \text{(Expanding the square)}\end{aligned}$$

Using the definitions of $g_i(z_i^*)$ and u_i, we have

$$\begin{aligned}&\implies \lambda_i\left(\langle \mu^\pi, c_i\rangle - b_i - z_i^*\right) - \frac{\beta}{2}(\langle \mu^\pi, c_i\rangle - b_i - z_i^*)^2 \\ &\qquad = g_i(z_i^*) = \frac{\beta}{2}\left[-\left(\min\left\{\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\right\}\right)^2 + \frac{\lambda_i^2}{\beta^2}\right] \qquad (15)\end{aligned}$$

Furthermore, since z^* is the optimal slack variable for an arbitrary λ,

$$\max_{z\in\mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) = \overline{F}_{\beta,\lambda}(\mu^\pi, z^*) = \langle \mu^\pi, r\rangle + \sum_{i=1}^m \lambda_i\left(\langle \mu^\pi, c_i\rangle - b_i - z_i^*\right) - \frac{\beta}{2}(\langle \mu^\pi, c_i\rangle - b_i - z_i^*)^2 \qquad (16)$$

$$= \langle \mu^\pi, r \rangle + \sum_{i=1}^{m} \frac{\beta}{2} \left[-\left(\min\left\{ \langle \mu^\pi, c_i \rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\} \right)^2 + \frac{\lambda_i^2}{\beta^2} \right] \qquad \text{(Using Eq. (15))}$$

$$= F_{\beta,\lambda}(\mu^\pi).$$

Finally, defining $\xi(\mu^\pi) \in \mathbb{R}^m$ such that for each $i \in [m]$

$$\xi_i(\mu^\pi) = \max\left\{ \langle \mu^\pi, c_i \rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\},$$

and substituting into $\overline{F}_{\beta,\lambda}(\mu^\pi, \xi(\mu^\pi))$ reproduces the same expression, implying

$$\overline{F}_{\beta,\lambda}(\mu^\pi, \xi(\mu^\pi)) = F_{\beta,\lambda}(\mu^\pi).$$

□

Lemma 7. *For a target $\epsilon \in (0,1)$, any $\lambda \in \mathbb{R}^m$ and $\beta > 0$, suppose there exists $\mu^{\pi'} \in \mathcal{K}$ such that*

$$F_{\beta,\lambda}(\mu^{\pi'}) \geq \max_{\mu^\pi \in \mathcal{K}} F_{\beta,\lambda}(\mu^\pi) - \epsilon.$$

Then the pair $(\mu^{\pi'}, \xi(\mu^{\pi'}))$ satisfies

$$\overline{F}_{\beta,\lambda}(\mu^{\pi'}, \xi(\mu^{\pi'})) \geq \max_{(\mu^\pi, z) \in \mathcal{K} \times \mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) - \epsilon.$$

where $\xi(\mu^{\pi'}) \in \mathbb{R}^m$ such that for each constraint $i \in [m]$, $\xi_i(\mu^{\pi'}) = \max\left\{ \left\langle \mu^{\pi'}, c_i \right\rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\}$

Proof. For a target $\epsilon \in (0,1)$, any $\lambda \in \mathbb{R}^m$ and $\beta > 0$, suppose there exists $\mu^{\pi'} \in \mathcal{K}$ such that

$$F_{\beta,\lambda}(\mu^{\pi'}) \geq \max_{\mu^\pi \in \mathcal{K}} F_{\beta,\lambda}(\mu^\pi) - \epsilon = \max_{(\mu^\pi, z) \in \mathcal{K} \times \mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) - \epsilon \qquad \text{(Using Lemma 6)}$$

Furthermore by Lemma 6,

$$F_{\beta,\lambda}(\mu^{\pi'}) = \overline{F}_{\beta,\lambda}(\mu^{\pi'}, \xi(\mu^{\pi'}))$$

where

$$\xi_i(\mu^{\pi'}) = \max\left\{ \left\langle \mu^{\pi'}, c_i \right\rangle - b_i - \frac{\lambda_i}{\beta}, 0 \right\}.$$

Hence,

$$\overline{F}_{\beta,\lambda}(\mu^{\pi'}, \xi(\mu^{\pi'})) \geq \max_{(\mu^\pi, z) \in \mathcal{K} \times \mathbb{R}^m_+} \overline{F}_{\beta,\lambda}(\mu^\pi, z) - \epsilon.$$

□

Since $\mathcal{Z} \subseteq \mathbb{R}^m_+$ and the slack $\xi(\mu^{\pi_{t+1}})$ lies in $\mathcal{Z}$ (see the proof of Lemma 5), the guarantee of Lemma 7 implies the oracle condition over $\mathcal{K} \times \mathcal{Z}$ required by Lemma 5.

D Proofs and Technical Details for the Augmented Lagrangian Method

D.1 Equality Constraint Problems

To simplify our analysis and align with classical optimization literature, we consider problems in standard form:

Problem 4 (Equality Constraint Problem)**.**

$$\min_{x \in \mathcal{X}} \quad f(x) \quad \text{subject to} \quad Ax = b, \tag{17}$$

where f is a convex function, $\mathcal{X} \subseteq \mathbb{R}^n$ is a closed and bounded set, $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$. Let x^* denote an optimal solution to Problem 4. The augmented Lagrangian (AL) function of Problem 4 is defined as

$$\mathcal{L}^\beta(x, \nu) := f(x) + \langle \nu, Ax - b \rangle + \frac{\beta}{2} \|Ax - b\|^2, \tag{18}$$

where $\nu \in \mathbb{R}^m$ is the Lagrange multiplier associated with the linear constraint, and $\beta > 0$ is the penalty parameter. The AL dual of Problem 4 is

$$\max_{\nu \in \mathbb{R}^m} d(\nu), \quad \text{where} \quad d(\nu) := \min_{x \in \mathcal{X}} \mathcal{L}^\beta(x, \nu). \tag{19}$$

Given an Lagrange multiplier ν_t at iteration $t \in [T]$, the AL method for solving Problem 4 iteratively updates the primal and dual variables via

$$x(\nu_t) = \arg\min_{x \in \mathcal{X}} \mathcal{L}^\beta(x, \nu_t) \quad ; \quad \nu_{t+1} = \nu_t + \frac{\beta}{2}\left(Ax(\nu_t) - b\right). \tag{20}$$

Since computing the primal update can be computationally infeasible, we consider the inexact variant where finding an approximate solution up to a target $\epsilon_t > 0$ is sufficient (shown in Algorithm 3).

Algorithm 3: Inexact Augmented Lagrangian Method for Equality Constraints (Problem 4)

1: **Input**: $x_1 \in \mathcal{X}$ (primal variable) $\nu_1 \in \mathbb{R}^m$ (dual variable), $\beta > 0$ (constraint penalty), $T > 0$ (number of outer-iterations) and the non-negative sequence $\{\epsilon_t\}_{t=1}^T$.
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Formulate augmented Lagrangian $\mathcal{L}^\beta(x, \nu_t) := f(x) + \langle \nu_t, Ax - b\rangle + \frac{\beta}{2}\|Ax - b\|_2^2$
4: Find the approximate solution x_{t+1} of the AL subproblem such that

$$\mathcal{L}^\beta(x_{t+1}, \nu_t) - \min_{x \in \mathcal{X}} \mathcal{L}^\beta(x, \nu_t) \le \epsilon_t$$

5: $\nu_{t+1} = \nu_t + \frac{\beta}{2}\left(Ax_{t+1} - b\right)$
6: **end for**
7: Return x_{T+1}

Before proceeding with the main theorem, we require a couple of technical lemmas. The following notation and relations will be helpful in the proofs: for all $t \ge 1$

$$x(\nu_t) := \arg\min_{x \in \mathcal{X}} \mathcal{L}^\beta(x, \nu_t), \tag{21}$$

$$\nabla d(\nu_t) := Ax(\nu_t) - b$$

$$\bar{d}(\nu_t) := \mathcal{L}^\beta(x_{t+1}, \nu_t) \tag{22}$$

$$\nabla \bar{d}(\nu_t) := Ax_{t+1} - b, \tag{23}$$

As stated in Algorithm 3, at iteration $t \ge 1$, we ensure that the following error condition is satisfied

$$\mathcal{L}^\beta(x_{t+1}, \nu_t) - \mathcal{L}^\beta(x(\nu_t), \nu_t) = \bar{d}(\nu_t) - d(\nu_t) \le \epsilon_t. \tag{24}$$

Lemma 8. *In Algorithm 3, for all iterations $t \ge 1$,*

$$\left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2 \le \frac{2\epsilon_t}{\beta}. \tag{25}$$

Proof.

$$\left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2 = \|Ax(\nu_t) - b - (Ax_{t+1} - b)\|_2^2 = \|Ax(\nu_t) - Ax_{t+1}\|_2^2 \tag{26}$$

Define

$$u^* = Ax(\nu_t) \quad ; \quad \hat{u} = Ax_{t+1} \quad ; \quad h(u) = \langle \nu_t, u - b\rangle + \frac{\beta}{2}\|u - b\|_2^2$$
$$\implies \mathcal{L}^\beta(x, \nu_t) = f(x) + h(Ax)$$

Since h is β-strongly convex in u, we have

$$h(\hat{u}) \ge h(u^*) + \langle \nabla h(u^*), \hat{u} - u^*\rangle + \frac{\beta}{2}\|\hat{u} - u^*\|_2^2.$$

By convexity of f, for any x, y

$$f(y) \;\geq\; f(x) + \langle \nabla f(x),\, y - x \rangle.$$

Furthermore, by the first-order optimality condition at $x(\nu_t)$ gives, for all $x \in \mathcal{X}$,

$$\langle \nabla \mathcal{L}^\beta(x(\nu_t), \nu_t),\, x - x(\nu_t) \rangle \geq 0 \implies \langle \nabla f(x(\nu_t)) + A^T \nabla h(u^*),\, x - x(\nu_t) \rangle \geq 0$$
$$\implies \langle \nabla \mathcal{L}^\beta(x(\nu_t), \nu_t),\, x_{t+1} - x(\nu_t) \rangle \geq 0 \implies \langle \nabla f(x(\nu_t)) + A^T \nabla h(u^*),\, x_{t+1} - x(\nu_t) \rangle \geq 0$$
(Setting $x = x_{t+1}$)

Using the definition of u^* and $\hat{u}$

$$\implies \langle \nabla f(x(\nu_t)),\, x_{t+1} - x(\nu_t) \rangle + \langle \nabla h(u^*),\, \hat{u} - u^* \rangle \geq 0 \tag{27}$$

$$\begin{aligned}
\mathcal{L}^\beta(x_{t+1}, \nu_t) - \mathcal{L}^\beta(x(\nu_t), \nu_t) &= [f(x_{t+1}) - f(x(\nu_t))] + [h(\hat{u}) - h(u^*)] \\
&\geq \langle \nabla f(x(\nu_t)),\, x_{t+1} - x(\nu_t) \rangle + \left[\langle \nabla h(u^*),\, \hat{u} - u^* \rangle + \frac{\beta}{2} \|\hat{u} - u^*\|_2^2 \right] \\
&\qquad \text{(By convexity of } f \text{ and strong-convexity of } h\text{)} \\
&\geq \underbrace{\langle \nabla f(x(\nu_t)),\, x_{t+1} - x(\nu_t) \rangle + \langle \nabla h(u^*),\, \hat{u} - u^* \rangle}_{\geq 0 \text{ from Eq. (27)}} + \frac{\beta}{2} \|\hat{u} - u^*\|_2^2 \\
&\geq \frac{\beta}{2} \|A x_{t+1} - A x(\nu_t)\|_2^2 \qquad \text{(By definition of } u^* \text{ and } \hat{u}\text{)}
\end{aligned}$$

Hence, since $\mathcal{L}^\beta(x_{t+1}, \nu_t) - \mathcal{L}^\beta(x(\nu_t), \nu_t) \leq \epsilon_t$, we conclude

$$\|A x_{t+1} - A x(\nu_t)\|_2^2 \leq \frac{2\epsilon_t}{\beta}.$$

Combining the above inequality with Eq. (26) finishes the proof. □

Lemma 9. *In Algorithm 3, for all $t \geq 1$, the following two inequalities hold:*

$$d(\nu) \leq \bar{d}(\nu') + \left\langle \nabla \bar{d}(\nu'), \nu - \nu' \right\rangle, \; \forall\, \nu, \nu' \tag{28}$$

and

$$d(\nu_{t+1}) \geq \bar{d}(\nu_t) + \frac{\beta}{4} \left\| \nabla \bar{d}(\nu_t) \right\|^2 - 2\,\epsilon_t. \tag{29}$$

Proof. We first show Eq. (28). By the definitions of $d(\nu)$ and $\mathcal{L}^\beta(x, \nu)$, we have

$$d(\nu) = \min_{x \in \mathcal{X}} \left\{ \mathcal{L}^\beta(x, \nu) \right\} \leq \mathcal{L}^\beta(x_{\nu'}, \nu) = \mathcal{L}^\beta(x_{\nu'}, \nu') + \langle \nu - \nu', A x_{\nu'} - b \rangle,$$
(By definition of $\mathcal{L}^\beta(x, \nu)$ and $\mathcal{L}^\beta(x, \nu')$)

where $x_{\nu'}$ is defined such that $\bar{d}(\nu') = \mathcal{L}^\beta(x_{\nu'}, \nu')$. This, together with the definition of $\nabla \bar{d}(\nu') = A x_{\nu'} - b$ yields Eq. (28).

We now prove Eq. (29). According to Lemma 12, $d(\nu)$ is $\frac{1}{\beta}$-smooth,

$$\begin{aligned}
d(\nu_{t+1}) &\geq d(\nu_t) + \langle \nabla d(\nu_t), \nu_{t+1} - \nu_t \rangle - \frac{1}{2\,\beta} \|\nu_{t+1} - \nu_t\|_2^2 \\
&= d(\nu_t) + \langle \nabla d(\nu_t), \frac{\beta}{2} \nabla \bar{d}(\nu_t) \rangle - \frac{\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2 \\
&\qquad \text{(Using the dual update } \nu_{t+1} = \nu_t + \tfrac{\beta}{2} \nabla \bar{d}(\nu_t)\text{)} \\
&= \bar{d}(\nu_t) + \underbrace{d(\nu_t) - \bar{d}(\nu_t)}_{\geq -\epsilon_t \text{ from Eq. (24)}} + \langle \nabla d(\nu_t), \frac{\beta}{2} \nabla \bar{d}(\nu_t) \rangle - \frac{\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2 \\
&\qquad \text{(Add/Subtract } \bar{d}(\nu_t)\text{)}
\end{aligned}$$

$$\geq \bar{d}(\nu_t) - \epsilon_t + \langle \nabla d(\nu_t) + \nabla \bar{d}(\nu_t) - \nabla \bar{d}(\nu_t), \frac{\beta}{2} \nabla \bar{d}(\nu_t) \rangle - \frac{\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2$$

(Add/Subtract $\langle \nabla \bar{d}(\nu_t), \frac{\beta}{2} \nabla \bar{d}(\nu_t) \rangle$)

$$= \bar{d}(\nu_t) - \epsilon_t + \frac{3\,\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2 - \left[\langle -\frac{\beta}{2} \nabla \bar{d}(\nu_t), \nabla d(\nu_t) - \nabla \bar{d}(\nu_t) \rangle \right]$$

$$\geq \bar{d}(\nu_t) - \epsilon_t + \frac{3\,\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2 - \frac{\beta}{8} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2 - \frac{\beta}{2} \left\| \nabla d(\nu_t) - \nabla \bar{d}(\nu_t) \right\|_2^2$$

(Using Young's Inequality, $\langle a, b \rangle \leq \frac{\|a\|_2^2}{2\,\beta} + \frac{\beta\,\|b\|_2^2}{2}$)

$$\implies d(\nu_{t+1}) \geq \bar{d}(\nu_t) - 2\,\epsilon_t + \frac{\beta}{4} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2. \qquad \text{(Using Lemma 8)}$$

□

Lemma 10. *For all $t \geq 1$,*

$$\|\nu_t - \nu^*\| \leq \sqrt{\|\nu_1 - \nu^*\|_2^2 + 2\,\beta \sum_{i=1}^{t-1} \epsilon_i} \tag{30}$$

Proof.

$$\|\nu_{t+1} - \nu^*\|_2^2 = \left\| \nu_t + \frac{\beta}{2} \nabla \bar{d}(\nu_t) - \nu^* \right\|_2^2 = \|\nu_t - \nu^*\|_2^2 + \beta \langle \nabla \bar{d}(\nu_t), \nu_t - \nu^* \rangle + \frac{\beta^2}{4} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2$$

(Using the dual update $\nu_{t+1} = \nu_t + \frac{\beta}{2} \nabla \bar{d}(\nu_t)$)

$$\leq \|\nu_t - \nu^*\|_2^2 + \beta \left[\bar{d}(\nu_t) - d(\nu^*) \right] + \frac{\beta^2}{4} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2$$

(Using Eq. (28) in Lemma 9 with $\nu = \nu^*$ and $\nu' = \nu_t$)

$$= \|\nu_t - \nu^*\|_2^2 + \underbrace{\beta \left[d(\nu_{t+1}) - d(\nu^*) \right]}_{\leq 0 \text{ since } \nu^* := \arg\max_{\nu \in \mathbb{R}^m} d(\nu)} + \beta \left[\bar{d}(\nu_t) - d(\nu_{t+1}) \right] + \frac{\beta^2}{4} \left\| \nabla \bar{d}(\nu_t) \right\|_2^2$$

(Add/Subtract $d(\nu_{t+1})$)

$$\leq \|\nu_t - \nu^*\|_2^2 + \beta \left[\bar{d}(\nu_t) - d(\nu_{t+1}) \right] + \beta [d(\nu_{t+1}) - \bar{d}(\nu_t) + 2\,\epsilon_t]$$

(Using Eq. (29) in Lemma 9 to simplify the last term)

$$\leq \|\nu_t - \nu^*\|_2^2 + 2\,\beta\,\epsilon_t$$

Hence, by recursing, for all $t \geq 1$,

$$\implies \|\nu_t - \nu^*\|_2^2 \leq \|\nu_1 - \nu^*\|_2^2 + 2\,\beta \sum_{i=1}^{t-1} \epsilon_i$$

□

Theorem 3. *If $\epsilon_t = \frac{\sigma}{t^2}$ with any constant $\sigma > 0$ and $\nu_1 = 0$, Algorithm 3 achieves the following convergence: for all $t \geq 1$,*

$$d(\nu^*) - d(\nu_t) \leq \frac{C}{t} \tag{31}$$

$$\|Ax_{t+1} - b\| \leq \frac{2}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{t}} + \frac{\sqrt{2\,\sigma}}{t} \right) \tag{32}$$

$$f(x_{t+1}) - f(x^*) \leq \frac{2(B + \|\nu^*\|)}{\sqrt{\beta}} \left(\frac{\sqrt{C}}{\sqrt{t}} + \frac{\sqrt{2\,\sigma}}{t} \right) + \frac{\sigma}{t^2}, \tag{33}$$

where $B := \|\nu^*\| + \sqrt{\sigma\,\beta\,\frac{\pi^2}{3}}$, $\kappa := \frac{\beta}{8B^2}$, $\omega := 3$, and $C := \frac{4}{\kappa} \left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa + 1)} \right) + 4 \max \left\{ d(\nu^*) - d(\nu_1), \frac{4}{\kappa} \right\}$.

Proof. Using Eq. (29) in Lemma 9,

$$d(\nu_{t+1}) \geq \bar{d}(\nu_t) + \frac{\beta}{4} \left\|\nabla \bar{d}(\nu_t)\right\|_2^2 - 2\epsilon_t$$

Since by definition, $\bar{d}(\nu_t) \geq d(\nu_t)$,

$$\geq d(\nu_t) + \frac{\beta}{4} \left\|\nabla \bar{d}(\nu_t)\right\|_2^2 - 2\epsilon_t$$

Since $\|\nabla d(\nu_t)\|_2^2 = \left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t) + \nabla \bar{d}(\nu_t)\right\|_2^2 \leq 2\left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2 + 2\left\|\nabla \bar{d}(\nu_t)\right\|_2^2$. Hence, $\left\|\nabla \bar{d}(\nu_t)\right\|_2^2 \geq \frac{1}{2}\|\nabla d(\nu_t)\|_2^2 - \left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2$. Hence,

$$\geq d(\nu_t) + \frac{\beta}{8}\|\nabla d(\nu_t)\|_2^2 - \frac{\beta}{4}\left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2 - 2\epsilon_t$$

$$\implies d(\nu^*) - d(\nu_{t+1}) \leq d(\nu^*) - d(\nu_t) - \frac{\beta}{8}\|\nabla d(\nu_t)\|_2^2 + \frac{\beta}{4}\left\|\nabla d(\nu_t) - \nabla \bar{d}(\nu_t)\right\|_2^2 + 2\epsilon_t$$

$$\leq \underbrace{d(\nu^*) - d(\nu_t)}_{:=\delta_t} - \frac{\beta}{8}\|\nabla d(\nu_t)\|_2^2 + 3\epsilon_t \qquad \text{(Using Lemma 8)}$$

$$\implies \delta_{t+1} \leq \delta_t - \frac{\beta}{8}\|\nabla d(\nu_t)\|_2^2 + 3\epsilon_t$$

Furthermore,

$$\|\nu_t - \nu^*\| \leq \sqrt{\|\nu_1 - \nu^*\|_2^2 + 2\,\beta \sum_{i=1}^{t-1} \epsilon_i} \qquad \text{(Using Lemma 10)}$$

$$\leq \left[\|\nu^*\| + \sqrt{2\,\sigma\,\beta \sum_{i=1}^{t-1} \frac{1}{i^2}}\right]$$

(Using that $\nu_1 = 0$, $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ and $\epsilon_t = \frac{\sigma}{t^2}$)

$$\leq \underbrace{\left[\|\nu^*\| + \sqrt{\frac{\sigma\,\beta\,\pi^2}{3}}\right]}_{:=B} \qquad \left(\textstyle\sum_{i=1}^{t-1} \frac{1}{i^2} \leq \sum_{i=1}^{\infty} \frac{1}{i^2} = \frac{\pi^2}{6}\right)$$

$$\implies \|\nu_t - \nu^*\| \leq B. \tag{34}$$

Since $d(\nu)$ is concave in ν,

$$d(\nu^*) - d(\nu_t) \leq \langle \nabla d(\nu_t), \nu^* - \nu_t \rangle \leq \|\nabla d(\nu_t)\|\,\|\nu_t - \nu^*\| \qquad \text{(Cauchy Schwarz)}$$

$$\leq \|\nabla d(\nu_t)\|\,B \qquad \text{(Using Eq. (34))}$$

$$\implies \|\nabla d(\nu_t)\| \geq \frac{d(\nu^*) - d(\nu_t)}{B} = \frac{\delta_t}{B}$$

Combining the above relations,

$$\delta_{t+1} \leq \delta_t - \frac{\beta}{8\,B^2}\,\delta_t^2 + 3\epsilon_t$$

Using Lemma 11, we can now conclude that for $\epsilon_t = \frac{\sigma}{t^2}$, $\kappa = \frac{\beta}{8B^2}$ and $\omega = 3$, if $C := \frac{4}{\kappa}\left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa + 1)}\right) + 4\max\left\{\delta_1, \frac{4}{\kappa}\right\}$, for all $t \geq 1$,

$$\delta_t \leq \frac{C}{t}$$

Now we will use the above result to bound the constraint violation and sub-optimality. Using the above relations, we know that,

$$d(\nu_{t+1}) \geq d(\nu_t) + \frac{\beta}{4}\left\|\nabla \bar{d}(\nu_t)\right\|_2^2 - 2\epsilon_t$$

Since $\nabla \bar{d}(\nu_t) = Ax_{t+1} - b$,

$$\begin{aligned}
\|Ax_{t+1} - b\|^2 &\leq \frac{4}{\beta}\left(d(\nu_{t+1}) - d(\nu_t) + 2\,\epsilon_t\right) \\
&\leq \frac{4}{\beta}\left(d(\nu^*) - d(\nu_t) + 2\,\epsilon_t\right) \\
&= \frac{4}{\beta}\left(\delta_t + 2\,\epsilon_t\right) \\
&\leq \frac{4}{\beta}\left(\frac{C}{t} + \frac{2\,\sigma}{t^2}\right) \\
\implies \|Ax_{t+1} - b\| &\leq \frac{2}{\sqrt{\beta}}\left(\frac{\sqrt{C}}{\sqrt{t}} + \frac{\sqrt{2\,\sigma}}{t}\right)
\end{aligned}$$

We will now bound the sub-optimality.

$$\begin{aligned}
\mathcal{L}^\beta(x_{t+1}, \nu_t) &= \mathcal{L}^\beta(x_{t+1}, \nu_t) - \mathcal{L}^\beta(x(\nu_t), \nu_t) + \mathcal{L}^\beta(x(\nu_t), \nu_t) \leq \mathcal{L}^\beta(x(\nu_t), \nu_t) + \epsilon_t \\
&= d(\nu_t) + \epsilon_t && \text{(By definition of } d(\nu_t)) \\
&\leq d(\nu^*) + \epsilon_t && \text{(Since } \nu^* = \arg\max_\nu d(\nu)) \\
&= f(x^*) + \epsilon_t && \text{(Strong duality)}
\end{aligned}$$

$$\implies f(x_{t+1}) + \langle \nu_t, Ax_{t+1} - b\rangle + \frac{\beta}{2}\|Ax_{t+1} - b\|_2^2 \leq f(x^*) + \epsilon_t$$
(By definition of $\mathcal{L}^\beta(x_{t+1}, \nu_t)$)

$$\begin{aligned}
\implies f(x_{t+1}) &\leq f(x^*) + \|\nu_t\|\,\|Ax_{t+1} - b\| + \epsilon_t && \text{(Cauchy Schwarz)} \\
&\leq f(x^*) + (\|\nu_t - \nu^*\| + \|\nu^*\|)\,\|Ax_{t+1} - b\| + \epsilon_t && \text{(Triangle inequality)} \\
&\leq f(x^*) + (B + \|\nu^*\|)\,\|Ax_{t+1} - b\| + \epsilon_t && \text{(Using Eq. (34))} \\
\implies f(x_{t+1}) - f(x^*) &\leq \frac{2\,(B + \|\nu^*\|)}{\sqrt{\beta}}\left(\frac{\sqrt{C}}{\sqrt{t}} + \frac{\sqrt{2\sigma}}{t}\right) + \frac{\sigma}{t^2} \\
& && \text{(Using the bound on the constraint violation)}
\end{aligned}$$

□

D.2 Additional Lemmas

Lemma 11. *For $t \geq 1$, if $\delta_t \geq 0$, constants $\kappa > 0$, $\omega \geq 0$ and $\epsilon_t = \frac{\sigma}{t^2}$ for $\sigma > 0$, if*

$$\delta_{t+1} \leq \delta_t - \kappa\,\delta_t^2 + \omega\epsilon_t\,,$$

then, for all $t \geq 1$,

$$\delta_t \leq \frac{C}{t} \text{ where } C := \frac{4}{\kappa}\left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa + 1)}\right) + 4\max\left\{\delta_1, \frac{4}{\kappa}\right\}. \tag{35}$$

Proof. We adapt the proof of Liu et al. [2019, Theorem 4.3] and will prove the result inductively. To start, for any $\kappa > 0$, $\omega \geq 0$ define

$$D := \max\left\{\delta_1, \frac{4}{\kappa}\right\} \quad ; \quad K_0 := \omega(\omega\kappa\sigma + 1)\sigma \quad ; \quad z^* := \frac{2}{\kappa}\left(\sqrt{1 + 2\kappa K_0} - 1\right) \geq 0$$

so that by construction

$$\frac{\kappa}{2}(z^*)^2 + 2z^* = 4K_0\,.$$

We first show that for all $t \geq 1$,

$$\delta_{t+1} \leq \delta_t - \kappa\, \delta_t^2 + \omega\epsilon_t \implies \frac{\kappa}{2}\delta_{t+1}^2 + \delta_{t+1} \leq \delta_t + \frac{K_0}{t^2}\,.$$

Since $\kappa > 0$ and $\delta_t, \delta_{t+1} \geq 0$, we have

$$\begin{aligned}
&\delta_{t+1} \leq \delta_t + \omega\epsilon_t \\
\implies &\delta_{t+1}^2 \leq 2\delta_t^2 + 2\omega^2\epsilon_t^2 && \text{(Squaring both sides and using the fact that } (a+b)^2 \leq 2a^2 + 2b^2\text{)} \\
\implies &\frac{\delta_{t+1}^2}{2} - \frac{\omega^2\sigma^2}{t^4} \leq \delta_t^2. && \text{(Rearranging and definition of } \epsilon_t\text{)}
\end{aligned}$$

Using the fact that $\delta_{t+1} \leq \delta_t - \kappa\,\delta_t^2 + \omega\,\frac{\sigma}{t^2}$ we have

$$\begin{aligned}
\delta_{t+1} &\leq \delta_t - \kappa\delta_t^2 + \omega\epsilon_t \\
&\leq \delta_t - \frac{\kappa}{2}\delta_{t+1}^2 + \frac{\kappa\omega^2\sigma^2}{t^4} + \frac{\omega\sigma}{t^2} \\
&\leq \delta_t - \frac{\kappa}{2}\delta_{t+1}^2 + \frac{\omega\sigma(\omega\kappa\sigma + 1)}{t^2}, && \text{(Since } t \geq 1\text{)} \\
&= \delta_t - \frac{\kappa}{2}\delta_{t+1}^2 + \frac{K_0}{t^2}, && (K_0 := \omega\sigma(\omega\kappa\sigma + 1))
\end{aligned}$$

Rearranging the above inequality, we have for all $t \geq 1$

$$\frac{\kappa}{2}\delta_{t+1}^2 + \delta_{t+1} \leq \delta_t + \frac{K_0}{t^2} \tag{36}$$

Furthermore, define $x_t := \delta_t - \frac{z^*}{t}$. We next show that for the shifted sequence,

$$\delta_{t+1} > \frac{z^* + D}{t+1} \implies \frac{\kappa}{2}x_{t+1}^2 + x_{t+1} \leq x_t\,.$$

$$\begin{aligned}
\frac{\kappa}{2}x_{t+1}^2 + x_{t+1} - x_t &\leq \frac{K_0}{t^2} - \frac{\kappa\,\delta_{t+1}\,z^*}{t+1} + \frac{\kappa\,(z^*)^2}{2\,(t+1)^2} + \frac{z^*}{t\,(t+1)} \\
&&\text{(Using Eq. (36) and the definition of } x_t\text{)} \\
&< \frac{1}{(t+1)^2}\left[\frac{(t+1)^2}{t^2}K_0 - \kappa(z^* + D)z^* + \frac{\kappa}{2}(z^*)^2 + \frac{t+1}{t}z^*\right] \\
&&\text{(Assuming } \delta_{t+1} > \tfrac{z^*+D}{t+1}\text{)} \\
&\leq \frac{1}{(t+1)^2}\left[4K_0 - \frac{\kappa}{2}(z^*)^2 - (\kappa D - 2)z^*\right] \\
&&\text{(Since } \tfrac{t+1}{t} \leq 2 \text{ and } \tfrac{(t+1)^2}{t^2} \leq 4\text{)} \\
&\leq \frac{1}{(t+1)^2}\left[4K_0 - \frac{\kappa}{2}(z^*)^2 - 2z^*\right] = 0 && \text{(By construction of } z^*\text{)}
\end{aligned}$$

Using the above relation, we will now show that for all $t \geq 1$

$$x_t \leq \frac{D}{t}\,.$$

For $t = 1$, $x_1 \leq \delta_1 \leq D$. Now for some $t \geq 1$, suppose that $x_t \leq \frac{D}{t}$ and, for contradiction,

$$x_{t+1} > \frac{D}{t+1} > 0$$

By the definition of x_{t+1}, this implies

$$\delta_{t+1} = x_{t+1} + \frac{z^*}{t+1} > \frac{D + z^*}{t+1}.$$

Hence, applying the shifted-sequence inequality established above, we obtain

$$\underbrace{\frac{\kappa}{2}x_{t+1}^2 + x_{t+1}}_{:=f(x_{t+1})} \le x_t \le \frac{D}{t}.$$

Since $x_{t+1} > 0$ and $f(x)$ is increasing on $[0, \infty)$,

$$\begin{aligned}
x_{t+1} &\le \frac{-1 + \sqrt{1 + 2\kappa\frac{D}{t}}}{\kappa} && (f^{-1}\left(\tfrac{D}{t}\right) \text{ is the nonnegative root of } f(x) = \tfrac{D}{t}) \\
&= \frac{2D}{t + \sqrt{t^2 + 2\kappa D t}} && \text{(Rationalize the expression)} \\
\implies x_{t+1} &\le \frac{D}{t+1} && \text{(Since } D \ge \tfrac{4}{\kappa} \text{ and } t \ge 1)
\end{aligned}$$

which is a contradiction. Thus for all $t \ge 1$,

$$\delta_t \le \frac{D + z^*}{t} \tag{37}$$

Furthermore, since $\sqrt{1+a} - 1 \le \sqrt{a}$ for any $a \ge 0$, we have

$$\begin{aligned}
D + z^* &= D + \frac{2}{\kappa}\left(\sqrt{1 + 2\kappa K_0} - 1\right) \\
&\le D + \frac{2}{\kappa}\sqrt{2\kappa K_0} \\
&= D + \frac{2}{\kappa}\sqrt{2\omega\sigma\kappa(\omega\sigma\kappa + 1)} \\
&\le C && \left(C := \tfrac{4}{\kappa}\left(1 + \sqrt{2\,\omega\sigma\kappa(\omega\sigma\kappa+1)}\right) + 4\max\left\{\delta_1, \tfrac{4}{\kappa}\right\}\right)
\end{aligned}$$

which completes the proof. □

Lemma 12 (Theorem 1 in Li and Lau [2025]). *Assuming $\mathcal{X}$ is closed and convex and $f(x)$ is a closed, proper convex function the dual function $d(\nu) = \min_{x\in\mathcal{X}}\{f(x) + \langle \nu, Ax - b\rangle + \frac{\beta}{2}\|Ax - b\|_2^2\}$ with penalty parameter $\beta > 0$ is $\frac{1}{\beta}$-smooth.*

E Proofs for Projected Q-Ascent on General Utilities

In the Reinforcement Learning with General Utilities (RLGU) formulation, we consider the optimization problem

$$\max_\pi G(\pi) := \mathcal{F}(\mu^\pi), \tag{38}$$

where the general utility $\mathcal{F} : \mathcal{K} \to \mathbb{R}$ is a concave function over state-action occupancy measures μ^π. Although $\mathcal{F}(\mu^\pi)$ is concave, optimizing over occupancy measures is impractical when the state or action space is large. Hence, we aim to optimize the induced objective $G(\pi)$ in the policy space. Following prior work [Zhang et al., 2020, Kumar et al., 2024, Barakat et al., 2023, 2024], we will apply policy gradient (PG) methods to optimize $G(\pi)$.

E.1 Tabular Setting

We first consider the tabular setting with direct policy parameterization where $\pi(\cdot|s) \in \Delta_A$ for all $s \in \mathcal{S}$. The policy gradient theorem for general utilities (Theorem 6) yields $[\nabla_\pi G(\pi)](s,a) = \frac{d^\pi(s)\, Q^\pi_{R(\pi)}(s,a)}{1-\gamma}$ where $R(\pi) = \nabla\mathcal{F}(\mu^\pi)$. Therefore, the projected Q-Ascent (PQA) [Xiao, 2022, Liu et al., 2025a] update for the step-size $\eta > 0$ can be written as: for any $k \ge 1$,

$$\pi_{k+1} = \arg\max_\pi \left\{ \sum_{s\in\mathcal{S}} \frac{d^{\pi_k}(s)}{1-\gamma}\left[\left\langle Q^{\pi_k}_{R(\pi_k)}(s,\cdot), \pi(\cdot|s) - \pi_k(\cdot|s)\right\rangle - \frac{1}{2\,\eta}\|\pi(\cdot|s) - \pi_k(\cdot|s)\|_2^2\right]\right\}. \tag{39}$$

Theorem 4. *Under Assumptions 1 to 3, using the* PQA *update in Eq. (39) with* $\eta = \frac{\rho_{\min}}{L}$ *for* $K \geq 2$ *iterations converges as:*

$$\max_{\pi} G(\pi) - G(\pi_K) \leq \frac{32\, L\, (1 + {}^{1}/_{(1-\gamma)\,\rho_{\min}})}{(1-\gamma)^2\, \rho_{\min}^2\, K}\,. \tag{40}$$

Proof. Under Assumption 3, the general utility objective G is L-smooth. i.e., for policies $\pi, \pi' \in \Pi$,

$$|G(\pi) - G(\pi') - \langle \nabla_\pi G(\pi'), \pi - \pi' \rangle| \leq \frac{L}{2} \|\pi - \pi'\|_2^2$$

which further implies the following inequalities:

$$G(\pi) \geq G(\pi') + \langle \nabla_\pi G(\pi'), \pi - \pi' \rangle - \frac{L}{2} \|\pi - \pi'\|_2^2 \tag{41}$$

$$G(\pi') + \langle \nabla_\pi G(\pi'), \pi - \pi' \rangle \geq G(\pi) - \frac{L}{2} \|\pi - \pi'\|_2^2\,. \tag{42}$$

Under Assumption 1, for all $s \in \mathcal{S}$, $\rho(s) \geq \rho_{\min}$. Let $\eta = \frac{\rho_{\min}}{L}$ and under this choice, $\frac{L}{2} = \frac{\rho_{\min}}{2\,\eta}$. Using Eq. (41) with $\pi = \pi_{k+1}$ and $\pi' = \pi_k$, we have that

$$\begin{aligned}
G(\pi_{k+1}) &\geq G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi_{k+1} - \pi_k \rangle - \frac{\rho_{\min}}{2\,\eta} \|\pi_{k+1} - \pi_k\|_2^2 \\
&= G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi_{k+1} - \pi_k \rangle - \frac{\rho_{\min}}{2\,\eta} \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{d^{\pi_k}(s)} \|\pi_{k+1}(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \\
&\geq G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi_{k+1} - \pi_k \rangle - \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{2\,(1-\gamma)\eta} \|\pi_{k+1}(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \\
&\qquad \text{(Under Assumption 1, } \tfrac{1}{d^{\pi_k}(s)} \leq \tfrac{1}{(1-\gamma)\rho_{\min}}\text{)} \\
&= \max_{\pi} \left\{ G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi - \pi_k \rangle - \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{2\,(1-\gamma)\,\eta} \|\pi(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \right\} \\
&\qquad \text{(Using the PQA update in Eq. (39))} \\
&= \max_{\pi} \left\{ G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi - \pi_k \rangle - \frac{L}{2} \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{(1-\gamma)\,\rho_{\min}} \|\pi(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \right\} \\
&\qquad (\eta = \tfrac{\rho_{\min}}{L}) \\
&\geq \max_{\pi} \left\{ G(\pi_k) + \langle \nabla_\pi G(\pi_k), \pi - \pi_k \rangle - \frac{L}{2\,(1-\gamma)\,\rho_{\min}} \|\pi - \pi_k\|_2^2 \right\} \\
&\qquad \text{(Since } d^{\pi_k}(s) \leq 1\text{)} \\
&\geq \max_{\pi} \Big\{ G(\pi) - \underbrace{\left(\frac{L}{2} + \frac{L}{2\,(1-\gamma)\,\rho_{\min}} \right)}_{:= C_1} \|\pi - \pi_k\|_2^2 \Big\} \quad \text{(Using Eq. (42) with } \pi' = \pi_k\text{)}
\end{aligned}$$

$$\implies G(\pi_{k+1}) \geq \max_{\pi} \big\{ G(\pi) - C_1 \|\pi - \pi_k\|_2^2 \big\} \tag{43}$$

Following Zhang et al. [2020], we consider the line segment between the state-action occupancies of π_k and $\pi_G^* := \arg\max_\pi G(\pi)$: for $\alpha_k \in [0, 1]$, define

$$\mu_{\alpha_k} := (1 - \alpha_k)\mu^{\pi_k} + \alpha_k \mu^{\pi_G^*}\,.$$

Since the occupancy set $\mathcal{K}$ is convex, $\mu_{\alpha_k} \in \mathcal{K}$ for any $\alpha_k \in [0, 1]$. Furthermore, using the mapping between π and μ^π, define $\pi_{\alpha_k} \in (\Delta_A)^S$ such that for all $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$\pi_{\alpha_k}(a|s) := \frac{\mu_{\alpha_k}(s, a)}{\sum_{a'} \mu_{\alpha_k}(s, a')}.$$

whenever the denominator is positive, with arbitrary choices at states with zero occupancy. Hence, this defines a valid curve of policies such that $\{\pi_{\alpha_k}\}_{\alpha_k \in [0,1]} \subseteq (\Delta_A)^S$. Using this set of policies with Eq. (43),

$$\implies G(\pi_{k+1}) \geq \max_{\pi} G(\pi) - C_1 \|\pi - \pi_k\|_2^2$$

$$
\begin{aligned}
&\geq \max_{\alpha_k \in [0,1]} G(\pi_{\alpha_k}) - C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2. \\
&= \max_{\alpha_k \in [0,1]} \mathcal{F}(\mu^{\pi_{\alpha_k}}) - C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 \\
&\qquad\qquad \text{(Using the mapping between } \pi \text{ and } \mu^\pi\text{)} \\
&\geq \max_{\alpha_k \in [0,1]} \alpha_k \, \mathcal{F}(\mu^{\pi_G^*}) + (1-\alpha_k)\, \mathcal{F}(\mu^{\pi_k}) - C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 \\
&\qquad\qquad \text{(Since } \mathcal{F} \text{ is concave)} \\
&= \max_{\alpha_k \in [0,1]} \alpha_k \, G(\pi_G^*) + (1-\alpha_k)\, G(\pi_k) - C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 \\
&\qquad\qquad \text{(Using the mapping between } \pi \text{ and } \mu^\pi\text{)}
\end{aligned}
$$

Multiplying both sides by -1 and adding $G(\pi_G^*)$,

$$
\begin{aligned}
\implies G(\pi_G^*) - G(\pi_{k+1}) &\leq G(\pi_G^*) - \max_{\alpha_k \in [0,1]} \left\{ \alpha_k \, G(\pi_G^*) + (1-\alpha_k)\, G(\pi_k) - C_1 \|\pi_{\alpha_k} - \pi_k\|_2^2 \right\} \\
&= G(\pi_G^*) + \min_{\alpha_k \in [0,1]} \left\{ -\alpha_k \, G(\pi_G^*) - (1-\alpha_k)\, G(\pi_k) + C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 \right\} \\
&= \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\, (G(\pi_G^*) - G(\pi_k)) + C_1 \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 \right\} \qquad (44)
\end{aligned}
$$

Additionally, under Assumption 1, by Zhang et al. [2020, Proposition H.1], the mapping $h(\mu^\pi) \to \pi$ is $\frac{2}{\rho_{\min}}$-Lipschitz, where

$$
[h(\mu^\pi)](s,a) = \frac{\mu^\pi(s,a)}{\sum_{a' \in \mathcal{A}} \mu^\pi(s,a')} = \pi(a|s).
$$

Hence,

$$
\begin{aligned}
\left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 &= \left\| h\Big(\alpha \, \mu^{\pi_G^*} + (1-\alpha)\, \mu^{\pi_k}\Big) - h(\mu^{\pi_k}) \right\|_2^2 \\
&\leq \frac{4}{\rho_{\min}^2} \|\alpha_k \, \mu^{\pi_G^*} + (1-\alpha_k)\mu^{\pi_t} - \mu^{\pi_t}\|_1^2 \qquad \text{(Since } \|\cdot\|_2 \leq \|\cdot\|_1\text{)} \\
&= \frac{4}{\rho_{\min}^2} \|\alpha_k (\mu^{\pi_G^*} - \mu^{\pi_t})\|_1^2 = \frac{4\,\alpha_k^2}{\rho_{\min}^2} \|\mu^{\pi_G^*} - \mu^{\pi_k}\|_1^2 \\
\implies \left\|\pi_{\alpha_k} - \pi_k\right\|_2^2 &\leq \frac{16\,\alpha_k^2}{\rho_{\min}^2\,(1-\gamma)^2} \qquad \text{(For } \mu, \mu' \in \mathcal{K}, \|\mu - \mu'\|_1 \leq \tfrac{2}{1-\gamma}\text{)}
\end{aligned}
$$

Using the above inequality with Eq. (44),

$$
\begin{aligned}
\implies G(\pi_G^*) - G(\pi_{k+1}) &\leq \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\, (G(\pi_G^*) - G(\pi_k)) + \frac{16\, C_1\, \alpha_k^2}{\rho_{\min}^2\,(1-\gamma)^2} \right\} \\
&= \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\, (\underbrace{G(\pi^*) - G(\pi_k)}_{:=\delta_k}) + \alpha_k^2\, C_2 \right\} \\
&\qquad\qquad (C_2 := \tfrac{16\, C_1}{\rho_{\min}^2\,(1-\gamma)^2}) \\
\implies \delta_{k+1} &\leq \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\, \delta_k + \alpha_k^2\, C_2 \right\}. \qquad (45)
\end{aligned}
$$

Note that the minimizer of $(1-\alpha_k)\,\delta_k + \alpha_k^2\, C_2$ over $\alpha_k \in [0,1]$ is $\alpha_k^* = \min\left\{1, \frac{\delta_k}{2\,C_2}\right\}$. Using this minimizer, we will prove for all $k \geq 2$,

$$
\alpha_{k+1}^* \leq \left(1 - \frac{\alpha_k^*}{2}\right) \alpha_k^*
$$

For $k = 1$, since $\alpha_1^* \leq 1$, from Eq. (45) we have that

$$
\delta_2 \leq (1-1)\delta_1 + 1^2\, C_2 = C_2,
$$

$$\implies \alpha_2^* = \frac{\delta_2}{2C_2} \leq \frac{1}{2} < 1 \qquad \text{(Dividing by } 2C_2\text{)}$$

Then, for $k \geq 2$, since $\alpha_k^* = \frac{\delta_k}{2C_2} < 1$, using Eq. (45),

$$\delta_{k+1} \leq \left(1 - \frac{\delta_k}{2C_2}\right)\delta_k + \frac{\delta_k^2}{4\,C_2^2}\,C_2 = \left(1 - \frac{\delta_k}{4\,C_2}\right)\delta_k\,,$$

$$\implies \alpha_{k+1}^* \leq \left(1 - \frac{\alpha_k^*}{2}\right)\alpha_k^*. \qquad \text{(Dividing by } 2C_2\text{)}$$

Hence, $\alpha_{k+1}^* \leq \alpha_k^* < 1$ for all $k \geq 2$. Furthermore, using the above inequality,

$$\frac{\alpha_{k+1}^*}{2} \leq \left(1 - \frac{\alpha_k^*}{2}\right)\frac{\alpha_k^*}{2}$$

$$\implies \frac{2}{\alpha_{k+1}^*} \geq \frac{1}{\left(1 - \frac{\alpha_k^*}{2}\right)\frac{\alpha_k^*}{2}} = \frac{2}{\alpha_k^*} + \frac{2}{2 - \alpha_k^*} \qquad \left(\tfrac{1}{x(1-x)} = \tfrac{1}{x} + \tfrac{1}{1-x}\right)$$

$$\geq \frac{2}{\alpha_k^*} + 1\,. \qquad (\alpha_k^* < 1)$$

$$\implies \frac{2}{\alpha_{k+1}^*} \geq \frac{2}{\alpha_k^*} + 1\,. \tag{46}$$

Using Eq. (46) and recursing from $k = 2$ to $K - 1$, we have that

$$\frac{2}{\alpha_K^*} \geq \frac{2}{\alpha_2^*} + (K - 2) \geq K \qquad (\alpha_2^* \leq 1)$$

$$\implies \frac{4\,C_2}{\delta_K} \geq K \qquad \left(\text{For all } k \geq 2, \alpha_k^* = \tfrac{\delta_k}{2C_2}\right)$$

$$\implies \delta_K \leq \frac{4\,C_2}{K}$$

Making the constants explicit, we have

$$\max_\pi G(\pi) - G(\pi_K) \leq \frac{32\,L\,(1 + {}^1\!/_{(1-\gamma)\,\rho_{\min}})}{(1-\gamma)^2\,\rho_{\min}^2\,K}\,.$$

which completes the proof. □

E.2 Linear Functional Approximation Setting

To handle large state/action spaces, we consider linear function approximation using log-linear policies. Given features $\varphi \in \mathbb{R}^{SA \times d}$, the policy is parameterized as

$$\pi_\theta(a|s) = \frac{\exp(\langle \varphi(s,a), \theta \rangle)}{\sum_{a'} \exp(\langle \varphi(s,a'), \theta \rangle)}\,,$$

where $\theta \in \mathbb{R}^d$ are the learnable parameters and $d \ll SA$. Since scaling the features can always be absorbed into a rescaling of θ, without loss of generality, we assume that $\max_{s,a} \|\varphi(s,a)\| = 1$. As a result, we consider the following policy class

$$\Pi_\theta := \{\pi \mid \exists \theta \text{ s.t } \pi = \pi_\theta\}$$

where the model parameterizing the policy is implicit in the definition.

In this setting, to update the policy, we consider the following projected PQA (PPQA) update using the forward KL-divergence. For all inner-iterations $k \geq 1$, using a fixed step-size $\eta > 0$, the PPQA update consists of two steps:

$$\pi_{k+1/2} = \arg\max_\pi \left\{ \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{1-\gamma} \left\langle Q^{\pi_k}_{R(\pi_k)}(s,\cdot), \pi(\cdot|s) - \pi_k(\cdot|s) \right\rangle - \frac{1}{2\,\eta} \sum_{s \in \mathcal{S}} \frac{d^{\pi_k}(s)}{1-\gamma} \|\pi(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \right\}, \tag{47a}$$

$$\pi_{k+1} = \arg\min_{\pi_\theta \in \Pi} \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathrm{KL}(\pi_{k+1/2}(\cdot|s) \| \pi_\theta(\cdot|s))\,. \tag{47b}$$

The intermediate policy $\pi_{k+1/2}$ corresponds to the tabular PQA update (cf. Eq. (39)). However, in the presence of function approximation, $\pi_{k+1/2}$ may not lie in the set of feasible policies class Π_θ. As a result, we project $\pi_{k+1/2}$ onto the set of feasible policies using the forward KL divergence to attain π_{k+1}.

However, solving the projection in Eq. (47b) is generally intractable. To alleviate this issue, following the prior works [Vaswani et al., 2021, Asad et al., 2024, 2025, Tomar et al., 2020, Lavington et al., 2023], we transform this projection into an unconstrained optimization over the parameters θ. To this end, we consider the following surrogate

$$\ell_k(\theta) := \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathrm{KL}\big(\pi_{k+1/2}(\cdot|s) \,\|\, \pi_\theta(\cdot|s)\big) \tag{48}$$

Hence, computing π_θ is equivalent to finding the minimizer of the following problem

$$\theta^*_{k+1} = \arg\min_{\theta \in \mathbb{R}^d} \ell_k(\theta) = \arg\min_{\theta \in \mathbb{R}^d} \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathbb{E}_{a \sim \pi_{k+1/2}(\cdot|s)}[-\log \pi_\theta(a|s)]\,.$$

where the second expression is obtained by expanding the forward KL divergence and dropping terms independent of θ. As a result, for the log-linear policy, minimizing the surrogate corresponds to a multi-class logistic regression classification with soft labels. Furthermore, we consider a practical variant where this minimization is carried out approximately using a finite number of gradient steps where $\theta_{k+1} \approx \arg\min_{\theta \in \mathbb{R}^d} \ell_k(\theta)$ and will set $\pi_{k+1} = \pi_{\theta_{k+1}}$.

To start, in the following Lemma, we prove that $\ell_k(\theta)$ is convex and is 1-smooth.

Lemma 13. *For a fixed inner-iteration $k \in [K]$, for the log-linear parameterization, the surrogate function*

$$\ell_k(\theta) = \sum_{s \in \mathcal{S}} d^{\pi_k}(s) \mathrm{KL}(\pi_{k+1/2}(\cdot|s) \| \pi_\theta(\cdot|s))$$

is convex with respect to θ and 1-smooth where $\pi_{k+1/2}$ is defined in Eq. (47a) and $\theta \in \mathbb{R}^d$.

Proof. Expanding the forward KL term in the surrogate yields that

$$\begin{aligned}
\ell_k(\theta) &= \sum_{s \in \mathcal{S}} d^{\pi_k}(s) \mathrm{KL}(\pi_{k+1/2}(\cdot|s) \| \pi_\theta(\cdot|s)) \\
&= \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathbb{E}_{a \sim \pi_{k+1/2}(\cdot|s)} \left[\log \left(\frac{\pi_{k+1/2}(a|s)}{\pi_\theta(a|s)} \right) \right] \\
&= \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathbb{E}_{a \sim \pi_{k+1/2}(\cdot|s)}[-\log \pi_\theta(a|s)] + Z_k \\
&\qquad\qquad \text{(Let } Z_k := \textstyle\sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathbb{E}_{a \sim \pi_{k+1/2}(\cdot|s)} \big[\log \pi_{k+1/2}(a|s)\big]\text{)} \\
&= \sum_{s \in \mathcal{S}} d^{\pi_k}(s)\, \mathbb{E}_{a \sim \pi_{k+1/2}(\cdot|s)} \left[\log \left(\sum_{a' \in \mathcal{A}} e^{\langle \theta, \varphi(s,a') \rangle} \right) - \langle \theta, \varphi(s,a) \rangle \right] + Z_k \\
&\qquad\qquad (\pi_\theta \text{ is a log-linear policy})
\end{aligned}$$

Since $\theta \mapsto \log \sum_{a'} e^{\langle \theta, \varphi(s,a') \rangle}$ is convex (log-sum-exp of affine functions) and the remaining terms are affine in θ, $\ell_k(\theta)$ is convex in θ. Moreover, by Agarwal et al. [2021, Remark 28], $\log \pi_\theta(a|s)$ is w^2-smooth where $w = \max_{s,a} \|\varphi(s,a)\|_2$. Hence $\ell_k(\theta)$ is 1-smooth. □

As a result, since $\ell_k(\theta)$ is a smooth, convex function, this implies monotone descent under standard gradient-based optimization methods and yields a valid approximate projec-

tion. Putting everything together, we summarize the complete PPQA update in Algorithm 4.

Algorithm 4: PPQA Update

Input: θ_k (parameters for policy π_k), η (policy update step-size), ζ (surrogate update step-size) N (number of surrogate iterations),

Use $\pi_k = \pi_{\theta_k}$ and η to compute $\pi_{k+1/2}$:

$$\pi_{k+1/2} = \arg\max_{\pi} \left\{ \sum_{s\in\mathcal{S}} \frac{d^{\pi_k}(s)}{1-\gamma} \left\langle Q^{\pi_k}_{R(\pi_k)}(s,\cdot), \pi(\cdot|s) - \pi_k(\cdot|s) \right\rangle - \frac{1}{2\,\eta} \sum_{s\in\mathcal{S}} \frac{d^{\pi_k}(s)}{1-\gamma} \|\pi(\cdot|s) - \pi_k(\cdot|s)\|_2^2 \right\}$$

Form surrogate $\ell_k(\theta) = \sum_{s\in\mathcal{S}} d^{\pi_k}(s)\, \mathrm{KL}(\pi_{k+1/2}(\cdot|s) \| \pi_\theta(\cdot|s))$

Initialize inner-loop: $\omega_0 = \theta_k$

for $n = 0, 1, \ldots, N-1$ **do**

  $\omega_{n+1} = \omega_n - \zeta\, \nabla_\omega \ell_k(\omega_n)$

end for

$\theta_{k+1} = \omega_N$

Return θ_{k+1}

To control the projection error induced by approximate minimization of $\ell_k(\theta)$, we recall the following assumptions.

Assumption 5 (Bias)**.** *For all $k \geq 1$, $\min_\theta \ell_k(\theta) \leq \epsilon_{\text{bias}}$*

Assumption 6 (Optimization Error)**.** *Suppose θ_{k+1} is obtained by approximately minimizing $\ell_k(\theta)$, then, $|\ell_k(\theta_{k+1}) - \min_\theta \ell_k(\theta)| \leq \epsilon_{\text{opt}}$.*

Theorem 5. *Suppose Assumptions 1 to 3 holds. Additionally under Assumptions 5 and 6 such that $\frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}} < C$, the PPQA update in Algorithm 4 with $\eta = \frac{\rho_{\min}}{L}$ results in the following convergence: for $K \geq 2$*

$$G(\pi^*) - G(\pi_K) \leq \frac{4\,C}{K} + 2\,C \sqrt{\frac{\sqrt{2}\,U\,\sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}}}{C\,(1-\gamma)^3\,\rho_{\min}}}.$$

where $C := \frac{8\,L}{(1-\gamma)^2 \rho_{\min}^2}(1 + \frac{1}{(1-\gamma)\rho_{\min}})$.

Proof. Under Assumptions 1 and 3, following the initial proof of Theorem 4, except with $\pi_{k+1/2}$, from Eq. (44) we have

$$G(\pi^*_G) - G(\pi_{k+1/2}) \leq \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\,(G(\pi^*_G) - G(\pi_k)) + \alpha_k^2 \underbrace{\frac{8\,L\,(1 + 1/(1-\gamma)\,\rho_{\min})}{\rho_{\min}^2\,(1-\gamma)^2}}_{:=C} \right\}$$

Adding and subtracting $G(\pi_{k+1})$, then rearranging

$$\implies G(\pi^*_G) - G(\pi_{k+1}) \leq \min_{\alpha_k \in [0,1]} \left\{ (1-\alpha_k)\,(G(\pi^*_G) - G(\pi_k)) + \alpha_k^2\, C \right\} + G(\pi_{k+1/2}) - G(\pi_{k+1}). \tag{49}$$

We first bound $G(\pi_{k+1/2}) - G(\pi_{k+1})$:

$$\begin{aligned}
G(\pi_{k+1/2}) - G(\pi_{k+1}) &\leq \frac{1}{1-\gamma}\, \mathbb{E}_{s\sim d^{\pi_{k+1/2}}} \left[\left\langle \pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s), Q^{\pi_{k+1}}_{R(\pi_{k+1})}(s,\cdot) \right\rangle \right] \\
&\qquad \text{(Using Lemma 15 with } \pi' = \pi_{k+1/2} \text{ and } \pi = \pi_{k+1}) \\
&= \frac{1}{1-\gamma} \sum_{s\in\mathcal{S}} d^{\pi_{k+1/2}}(s) \left\langle Q^{\pi_{k+1}}_{R(\pi_{k+1})}(s,\cdot), \pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s) \right\rangle \\
&\leq \frac{1}{1-\gamma} \sum_{s\in\mathcal{S}} d^{\pi_{k+1/2}}(s) \left\| Q^{\pi_{k+1}}_{R(\pi_{k+1})}(s,\cdot) \right\|_\infty \|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\|_1 \\
&\qquad \text{(Holder's inequality)}
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{U}{(1-\gamma)^2} \sum_{s\in\mathcal{S}} d^{\pi_{k+1/2}}(s) \left\|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\right\|_1 \\
&\qquad\qquad \text{(Under Assumption 2, } \left\|Q^{\pi_{k+1}}_{R(\pi_{k+1})}(s,\cdot)\right\|_\infty \leq \tfrac{U}{1-\gamma}\text{)} \\
&= \frac{U}{(1-\gamma)^2} \sum_{s\in\mathcal{S}} \frac{d^{\pi_{k+1/2}}(s)}{\rho(s)}\, \rho(s) \left\|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\right\|_1 \\
&\leq \frac{U}{(1-\gamma)^2} \left\|\frac{d^{\pi_{k+1/2}}(s)}{\rho(s)}\right\|_\infty \sum_{s\in\mathcal{S}} \rho(s) \left\|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\right\|_1 \\
&\leq \frac{U}{(1-\gamma)^2\,\rho_{\min}} \sum_{s\in\mathcal{S}} \rho(s) \left\|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\right\|_1 \\
&\qquad\qquad \text{(Since } d^{\pi_{k+1/2}}(s) \leq 1 \text{ and under Assumption 1, } \rho(s) \geq \rho_{\min}\text{)} \\
&\leq \frac{U}{(1-\gamma)^3\,\rho_{\min}} \sum_{s\in\mathcal{S}} d^{\pi_k}(s) \left\|\pi_{k+1/2}(\cdot|s) - \pi_{k+1}(\cdot|s)\right\|_1 \\
&\qquad\qquad \text{(Since } d^{\pi_k}(s) \geq (1-\gamma)\,\rho(s)\text{)} \\
&\leq \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sum_{s\in\mathcal{S}} d^{\pi_k}(s) \sqrt{\mathrm{KL}(\pi_{k+1/2}(\cdot|s)\|\pi_{k+1}(\cdot|s))} \\
&\qquad\qquad \text{(Pinsker's inequality)} \\
&\leq \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\sum_{s\in\mathcal{S}} d^{\pi_k}(s)\, \mathrm{KL}(\pi_{k+1/2}(\cdot|s)\|\pi_{k+1}(\cdot|s))} \\
&\qquad\qquad \text{(Due to the concavity of } \sqrt{\cdot} \text{ and Jensen's inequality)} \\
&= \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\sum_{s\in\mathcal{S}} d^{\pi_k}(s)\, \mathrm{KL}(\pi_{k+1/2}(\cdot|s)\|\pi_{\theta_{k+1}}(\cdot|s))} \\
&= \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\ell_k(\theta_{t+1})} \qquad \text{(Definition of } \ell_k(\theta)\text{)} \\
&= \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\ell_k(\theta_{t+1}) - \inf_\theta \ell_k(\theta) + \min_\theta \ell_k(\theta)} \\
&\qquad\qquad \text{(Add/Subtract } \min_\theta \ell_k(\theta)\text{)} \\
&\leq \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\ell_k(\theta_{t+1}) - \inf_\theta \ell_k(\theta) + \epsilon_{\text{bias}}} \\
&\qquad\qquad \text{(Under Assumption 5, } \min_\theta \ell_k(\theta) \leq \epsilon_{\text{bias}}\text{)} \\
&\leq \frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}} \\
&\qquad\qquad \text{(Under Assumption 6, } |\ell_k(\theta_{t+1}) - \min_\theta \ell_k(\theta)| \leq \epsilon_{\text{opt}}\text{)}
\end{aligned}
$$

$$
\implies G(\pi_{k+1/2}) - G(\pi_{k+1}) \leq \underbrace{\frac{\sqrt{2}\,U}{(1-\gamma)^3\,\rho_{\min}} \sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}}}_{:=E}\,. \tag{50}
$$

Combing Eq. (49) and Eq. (50), we have that

$$
\begin{aligned}
G(\pi^*_G) - G(\pi_{k+1}) &\leq \min_{\alpha_k\in[0,1]} \left\{ (1-\alpha_k)\, \underbrace{(G(\pi^*_G) - G(\pi_k))}_{:=\delta_k} + \alpha_k^2\, C \right\} + E\,. \\
\implies \delta_{k+1} &\leq \min_{\alpha_k\in[0,1]} \left\{ (1-\alpha_k)\,\delta_k + \alpha_k^2\, C \right\} + E \qquad (51)
\end{aligned}
$$

Note the minimizer of $(1-\alpha_k)\,\delta_k + \alpha_k^2\, C$ over $\alpha_k \in [0,1]$ is $\alpha^*_k = \min\{1, \frac{\delta_k}{2C}\}$. Now under the assumption that $E < C$, we first show that for all $k \geq 2$,

$$
\alpha^*_{k+1} \leq \alpha^*_k - \frac{(\alpha^*_k)^2}{2} + \frac{E}{2C}.
$$

For $k = 1$, we have $\alpha_1^* \leq 1$ and using this fact with Eq. (51),

$$\delta_2 \leq (1-1)\delta_1 + 1^2\, C + E < 2C$$

For $k \geq 2$, we will prove using induction that $\alpha_k^* = \frac{\delta_k}{2C}$.
Base case. For $k = 2$, it follows from the above expression after dividing by $2C$.
Induction hypothesis. Suppose for some $k \geq 2$, $\alpha_k^* = \frac{\delta_k}{2C}$.
Induction step. Stating with Eq. (51) and combining it with the induction hypothesis,

$$\begin{aligned}
\delta_{k+1} &\leq \left(1 - \frac{\delta_k}{2C}\right)\delta_k + \frac{\delta_k^2}{4\,C^2}\,C + E = \underbrace{\delta_k - \frac{\delta_k^2}{4C}}_{:=f(x)} + E \\
&\leq \max_{x\in[0,2C]} \left\{ x - \frac{x^2}{4C} \right\} + E \\
&\qquad \text{(By the induction hypothesis } \alpha_k^* = \tfrac{\delta_k}{2\,C} \leq 1 \implies \delta_k \leq 2C) \\
&\leq 2C - \frac{(2C)^2}{4C} + E \\
&= C + E < 2C \qquad \text{(Assuming } E < C) \\
\implies \alpha_{k+1}^* &= \frac{\delta_{k+1}}{2C} < 1.
\end{aligned}$$

As a result, using the fact that $\alpha_k^* = \frac{\delta_k}{2C}$ with Eq. (51), for all $k \geq 2$

$$\alpha_{k+1}^* \leq \alpha_k^* - \frac{(\alpha_k^*)^2}{2} + \underbrace{\frac{E}{2C}}_{:=D}. \tag{52}$$

Using Eq. (52), we will prove using induction that for all $k \geq 1$,

$$\alpha_k^* \leq \frac{2}{k} + \sqrt{2D}.$$

Base cases. For $k = 1$, we have $\alpha_1^* \leq 1 \implies \alpha_1^* \leq 2 + \sqrt{2D}$. Similarly, for $k = 2$, we have $\alpha_2^* \leq 1$ and $\alpha_2^* \leq 1 + \sqrt{2D}$.
Induction hypothesis. Suppose for some $k \geq 2$, $\alpha_k^* \leq \frac{2}{k} + \sqrt{2\,D}$.
Inductive Step. We consider two cases.
Case (i): Suppose $1 < \frac{2}{k} + \sqrt{2\,D}$. Since $\alpha_k^* \leq 1$ for all $k \geq 1$, we have

$$\alpha_{k+1}^* \leq \alpha_k^* - \frac{(\alpha_k^*)^2}{2} + D \leq \max_{x\in[0,1]} \left\{ x - \frac{x^2}{2} \right\} + D \leq 1 - \frac{1}{2} + D = \frac{1}{2} + D\,.$$

Hence, it suffices to show for $k \geq 2$

$$\frac{1}{2} + D \leq \frac{2}{k+1} + \sqrt{2D}.$$

Define $X := \sqrt{2D}$. As a result, we can rewrite $\frac{1}{2} + D \leq \frac{2}{k+1} + \sqrt{2D}$ in terms of X,

$$\frac{1}{2} + \frac{X^2}{2} \leq \frac{2}{k+1} + X \iff (X-1)^2 \leq \frac{4}{k+1}$$

Since $1 < \frac{2}{k} + X$ and under the assumption that $E < C$, $X = \sqrt{2D} = \sqrt{\frac{E}{C}} < 1$ we have

$$\begin{aligned}
0 \leq 1 - \frac{2}{k} < X < 1 &\implies -\frac{2}{k} < X - 1 < 0 \\
\implies (X-1)^2 < \frac{4}{k^2} &< \frac{4}{k+1}. \qquad \text{(For } k \geq 2,\ k^2 \geq k+1)
\end{aligned}$$

Hence,

$$\alpha_{k+1}^* \leq \frac{1}{2} + D \leq \frac{2}{k+1} + \sqrt{2D}.$$

Case (ii): Suppose $\frac{2}{k} + \sqrt{2\,D} \leq 1$. By the induction hypothesis, $\alpha_k^* \leq \frac{2}{k} + \sqrt{2\,D} \leq 1$. Define the map

$$h(x) := x - \frac{x^2}{2} + D. \tag{53}$$

Since $h'(x) = 1 - x$, $h(x)$ is monotonically increasing on $[0, 1]$. As a result,

$$\begin{aligned}
\alpha_{k+1}^* &\leq h(\alpha_k^*) \leq h\left(\frac{2}{k} + \sqrt{2\,D}\right) && \text{(By Eq. (52)and the induction hypothesis)} \\
&= \frac{2}{k} + \sqrt{2D} - \frac{1}{2}\left(\frac{2}{k} + \sqrt{2D}\right)^2 + D = \frac{2}{k} + \sqrt{2D} - \frac{2}{k^2} - \frac{2\sqrt{2D}}{k} \\
&= \frac{2}{k+1} + \sqrt{2D} - \frac{2}{k^2(k+1)} - \frac{2\sqrt{2D}}{k} \\
&\leq \frac{2}{k+1} + \sqrt{2D}.
\end{aligned}$$

Putting both cases together, we have

$$\alpha_{k+1}^* \leq \frac{2}{k+1} + \sqrt{2D} \implies \alpha_K^* = \frac{\delta_K}{2\,C} \leq \frac{2}{K} + \sqrt{2D}. \qquad \text{(for all } K \geq 2)$$

Making the constants explicit, we have

$$G(\pi^*) - G(\pi_K) \leq \frac{4\,C}{K} + 2\,C\sqrt{\frac{\sqrt{2}\,U\,\sqrt{\epsilon_{\text{opt}} + \epsilon_{\text{bias}}}}{C\,(1-\gamma)^3\,\rho_{\min}}}.$$

where $C = \frac{8\,L}{(1-\gamma)^2\rho_{\min}^2}\left(1 + \frac{1}{(1-\gamma)\rho_{\min}}\right)$ □

E.3 Supporting Lemmas

Lemma 14 (Value Difference Lemma)**.** *For any policies π, π' and an arbitrary pseudo-reward $\diamond \in \mathbb{R}^{SA}$*

$$V_\diamond^{\pi'} - V_\diamond^{\pi} = (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}[\mathcal{T}_\diamond^{\pi'} V_\diamond^{\pi} - V_\diamond^{\pi}], \tag{54}$$

where $\mathcal{T}_\diamond^{\pi'} V^\pi = \diamond^{\pi'} + \gamma\,\mathcal{P}^{\pi'} V^\pi$ is the Bellman operator for policy π' and $\diamond^{\pi'} \in \mathbb{R}^S$ is defined as $\diamond^{\pi'}(s) = \sum_{a\in\mathcal{A}} \diamond(s,a)\,\pi'(a|s)$.

Proof. The proof follows from the standard value difference lemma, except using a bounded pseudo-reward $\diamond$ instead of the reward r, and is added for completeness. The following notation will be useful in the proof.

$$\begin{aligned}
\diamond^\pi \in \mathbb{R}^S \quad \text{s.t.} \quad \diamond^\pi(s) &:= \sum_{a\in\mathcal{A}} \diamond(s,a)\,\pi(a|s) \\
\mathcal{P}^\pi \in \mathbb{R}^{S\times S} \quad \text{s.t.} \quad \mathcal{P}^\pi[s,s'] = \mathbf{Pr}^\pi(s \to s') &:= \sum_{a\in\mathcal{A}} \Pr(s'|s,a)\,\pi(a|s)
\end{aligned}$$

Recall the fact that the value function for the policy π' can be written as $V_\diamond^{\pi'} = (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}\diamond^{\pi'}$. As a result, for any policies π, π'

$$\begin{aligned}
V_\diamond^{\pi'} - V_\diamond^{\pi} &= (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}\,\diamond^{\pi'} - V_\diamond^{\pi} \\
&= (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}\left(\diamond^{\pi'} - (I_S - \gamma\,\mathcal{P}^{\pi'})\,V_\diamond^{\pi}\right) \\
&= (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}\left(\diamond^{\pi'} + \gamma\,\mathcal{P}^{\pi'} V_\diamond^{\pi} - V_\diamond^{\pi}\right) \\
&= (I_S - \gamma\,\mathcal{P}^{\pi'})^{-1}\left(\mathcal{T}_\diamond^{\pi'} V_\diamond^{\pi} - V_\diamond^{\pi}\right) && (\mathcal{T}_\diamond^{\pi'} V^\pi = \diamond^{\pi'} + \gamma\,\mathcal{P}^{\pi'} V^\pi)
\end{aligned}$$

□

Lemma 15 (General Utility Value Difference Lemma)**.** *For any policies π, π', for a concave general utility $G(\pi) = \mathcal{F}(\mu^\pi)$ with pseudo-reward $R(\pi) = \frac{d\mathcal{F}(x)}{dx}\Big|_{x=\mu^\pi}$, it holds that*

$$G(\pi') - G(\pi) \leq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}}\Big[[\mathcal{T}^{\pi'}_{R(\pi)}V^\pi_{R(\pi)}](s) - V^\pi_{R(\pi)}(s)\Big].$$

Proof. For any policies π', π, according to the value difference lemma (Lemma 14) with pseudo-reward $\diamond \in \mathbb{R}^{SA}$

$$\begin{aligned}
V^{\pi'}_\diamond - V^\pi_\diamond &= (I_S - \gamma\mathcal{P}^{\pi'})^{-1}[\mathcal{T}^{\pi'}_\diamond V^\pi_\diamond - V^\pi_\diamond] \qquad (\mathcal{T}^{\pi'}_\diamond V^\pi_\diamond = \diamond^{\pi'} + \gamma\mathcal{P}^{\pi'}V^\pi_\diamond)\\
\implies \rho\,[V^{\pi'}_\diamond - V^\pi_\diamond] &= \rho\,(I_S - \gamma\mathcal{P}_{\pi'})^{-1}[\mathcal{T}^{\pi'}_\diamond V^\pi_\diamond - V^\pi_\diamond]\\
&\qquad\qquad \text{(Multiplying both sides by initial state distribution } \rho)\\
\implies \mathbb{E}_{s\sim\rho}[V^{\pi'}_\diamond(s) - V^\pi_\diamond(s)] &= \frac{(1-\gamma)\,\rho\,(I_S - \gamma\mathcal{P}^{\pi'})^{-1}[\mathcal{T}^{\pi'}_\diamond V^\pi_\diamond - V^\pi_\diamond]}{1-\gamma}\\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}}\Big[[\mathcal{T}^{\pi'}_\diamond V^\pi_\diamond](s) - V^\pi_\diamond(s)\Big]\\
&\qquad\qquad (d^{\pi'} \coloneqq (1-\gamma)\,\rho\,(I_S - \gamma P_{\pi'})^{-1})\\
\implies V^{\pi'}_{R(\pi)}(\rho) - V^\pi_{R(\pi)}(\rho) &= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}}\Big[[\mathcal{T}^{\pi'}_{R(\pi)}V^\pi_{R(\pi)}](s) - V^\pi_{R(\pi)}(s)\Big]\\
&\qquad\qquad (\text{Setting } \diamond = R(\pi) \coloneqq \nabla_\mu\mathcal{F}(\mu^\pi))
\end{aligned}$$

We now use the LP formulation of MDPs to relate the $V^{\pi'}_{R(\pi)}(\rho) - V^\pi_{R(\pi)}(\rho)$ to the general utility objective

$$\begin{aligned}
V^{\pi'}_{R(\pi)}(\rho) - V^\pi_{R(\pi)}(\rho) &= \Big\langle R(\pi), \mu^{\pi'} - \mu^\pi\Big\rangle = \Big\langle \nabla_\mu\mathcal{F}(\mu^\pi), \mu^{\pi'} - \mu^\pi\Big\rangle \quad \text{(By definition of } R(\pi))\\
&\geq \mathcal{F}(\mu^{\pi'}) - \mathcal{F}(\mu^\pi) \qquad \text{(Since } \mathcal{F} \text{ is concave)}\\
&= G(\pi') - G(\pi). \quad \text{(Using the one-to-one relationship between } \pi \text{ and } \mu^\pi)
\end{aligned}$$

Putting everything together,

$$G(\pi') - G(\pi) \leq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}}\Big[[\mathcal{T}^{\pi'}_{R(\pi)}V^\pi_{R(\pi)}](s) - V^\pi_{R(\pi)}(s)\Big].$$

□

E.4 Additional Lemmas for the Augmented Lagrangian General Utility

Proposition 1. *For any dual variable $\lambda \in \mathbb{R}^m$ and penalty parameter $\beta > 0$, the* AL *in Eq. (2) is a concave general utility function:*

$$F_{\beta,\lambda}(\mu^\pi) = \langle\mu^\pi, r\rangle + \frac{\beta}{2}\sum_{i=1}^m\left(-\min\left\{\langle\mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\right\}^2 + \frac{\lambda_i^2}{\beta^2}\right).$$

Proof. Recall for any dual variable $\lambda \in \mathbb{R}^m$ and penalty parameter $\beta > 0$ the AL in Eq. (2) is defined as

$$\begin{aligned}
\mathcal{L}^\beta(\pi,\lambda) &= V^\pi_r(\rho) + \frac{\beta}{2}\sum_{i=1}^m\left(-\min\left\{V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\right\}^2 + \frac{\lambda_i^2}{\beta^2}\right)\\
&= \underbrace{\langle\mu^\pi, r\rangle + \frac{\beta}{2}\sum_{i=1}^m\left(-\min\left\{\langle\mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\right\}^2 + \frac{\lambda_i^2}{\beta^2}\right)}_{:=F_{\beta,\lambda}(\mu^\pi)}\\
&\qquad\qquad \text{(Using the linear programming formulation of CMDPs)}
\end{aligned}$$

Since the function $x \mapsto -\min\{x, 0\}^2$ is concave, $F_{\beta,\lambda}(\mu^\pi)$ is a concave general utility function. □

Proposition 2. *For each outer-iteration $t \in [T]$, the policy gradient of the* AL *in Proposition 1 is given by*

$$\nabla_\theta F_{\beta,\lambda}(\mu^{\pi_\theta}) = \sum_{s\in\mathcal{S}} \frac{d^\pi(s)}{1-\gamma} \sum_{a\in\mathcal{A}} Q^\pi_{R(\pi)}(s,a) \frac{d\pi_\theta(a|s)}{d\theta} \tag{55}$$

where $R(\pi) = r - \beta \sum_{i=1}^m c_i \min\Big\{V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\Big\}$ and $Q^\pi_{R(\pi)}$ is the state-action value function for the general utility pseudo-reward $R(\pi)$.

Proof. By Theorem 6, the policy gradient of a general utility is

$$\nabla_\theta G(\pi) = \nabla_\theta \mathcal{F}(\mu^{\pi_\theta}) = \frac{1}{1-\gamma} \sum_{s\in\mathcal{S}} d^\pi(s) \sum_{a\in\mathcal{A}} Q^\pi_{R(\pi)}(s,a) \frac{d\pi_\theta(a|s)}{d\theta}$$

where $R(\pi) \coloneqq \frac{d\mathcal{F}(x)}{dx}\Big|_{x=\mu^\pi}$ is the general utility pseudo-reward function. Furthermore, according to Proposition 1, the AL in Eq. (2) is equivalent to the following general utility function

$$G(\pi) = \mathcal{F}(\mu^\pi) = \langle \mu^\pi, r\rangle + \frac{\beta}{2} \sum_{i=1}^m \Bigg(\underbrace{-\min\Big\{\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\Big\}^2 + \frac{\lambda_i^2}{\beta^2}}_{:= f_i(\mu^\pi)} \Bigg).$$

where for each $i \in [m]$,

$$f_i(\mu^\pi) = \begin{cases} \frac{\lambda_i^2}{\beta^2} & \text{If } \langle \mu^\pi, c_i\rangle \geq b_i + \frac{\lambda_i}{\beta} \\ -\Big(\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}\Big)^2 + \frac{\lambda_i^2}{\beta^2} & \text{Else if } \langle \mu^\pi, c_i\rangle < b_i + \frac{\lambda_i}{\beta}. \end{cases}$$

$$\implies \frac{d[f_i(\mu^\pi)]}{d\mu^\pi} = \begin{cases} 0 & \text{If } \langle \mu^\pi, c_i\rangle \geq b_i + \frac{\lambda_i}{\beta} \\ -2\, c_i \left(\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}\right) & \text{Else if } \langle \mu^\pi, c_i\rangle < b_i + \frac{\lambda_i}{\beta}. \end{cases}$$

$$= -2c_i \min\Big\{\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\Big\}$$

Putting everything together, we have

$$R(\pi) = \frac{d\mathcal{F}(x)}{dx}\Big|_{x=\mu^\pi} = r - \beta \sum_{i=1}^m c_i \min\Big\{\langle \mu^\pi, c_i\rangle - b_i - \frac{\lambda_i}{\beta}, 0\Big\}.$$

□

Lemma 16. *Under the direct parameterization, for any $\lambda \in \mathbb{R}^m$ and $\beta > 0$, the* AL *defined in Eq. (2) is $L_{\beta,\lambda}$-smooth, where*

$$L_{\beta,\lambda} = \frac{2\,\gamma\, A}{(1-\gamma)^3} \left(1 + \beta \|b\|_1 + \|\lambda\|_1\right) + \frac{\beta\, m\, A}{(1-\gamma)^4}.$$

Proof. Recall that for a fixed $\lambda \in \mathbb{R}^m$ and penalty $\beta > 0$, the AL in Eq. (2) is defined as

$$\mathcal{L}^\beta(\pi, \lambda) \coloneqq V^\pi_r(\rho) + \frac{\beta}{2} \sum_{i=1}^m \left(-\min\Big\{V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}, 0\Big\}^2 + \frac{\lambda_i^2}{\beta^2}\right)$$

To start, for each $i \in [m]$, define

$$g_i(\pi) \coloneqq V^\pi_{c_i}(\rho) - b_i - \frac{\lambda_i}{\beta}.$$

Since $x \mapsto \min\{x, 0\}^2$ is continuously differentiable, the gradient of the AL is

$$\nabla_\pi \mathcal{L}^\beta(\pi, \lambda) = \nabla_\pi V^\pi_r(\rho) - \beta \sum_{i=1}^m \min\{g_i(\pi), 0\} \nabla_\pi V^\pi_{c_i}(\rho).$$

Hence, for any policies π, π'

$$\begin{aligned}
&\|\nabla_\pi \mathcal{L}^\beta(\pi,\lambda) - \nabla_{\pi'}\mathcal{L}^\beta(\pi',\lambda)\| \\
&\le \|\nabla_\pi V_r^\pi(\rho) - \nabla_\pi V_r^{\pi'}(\rho)\| + \beta \sum_{i=1}^m \|\min\{g_i(\pi),0\}\,\nabla_\pi V_{c_i}^\pi(\rho) - \min\{g_i(\pi'),0\}\,\nabla_{\pi'} V_{c_i}^{\pi'}(\rho)\| \\
&\qquad\qquad \text{(Triangle inequality)} \\
&\le \frac{2\,\gamma\,A}{(1-\gamma)^3}\,\|\pi-\pi'\| + \beta \sum_{i=1}^m \|\min\{g_i(\pi),0\}\,\nabla_\pi V_{c_i}^\pi(\rho) - \min\{g_i(\pi'),0\}\,\nabla_\pi V_{c_i}^{\pi'}(\rho)\| \\
&\qquad\qquad \text{(Since } r(s,a)\in[0,1]\text{, by Lemma 18, } V_r^\pi(\rho) \text{ is } \tfrac{2\,\gamma\,A}{(1-\gamma)^3} \text{ smooth)}
\end{aligned}$$

To upper bound the second term, for any $i \in [m]$

$$\begin{aligned}
&\|\min\{g_i(\pi),0\}\nabla V_{c_i}^\pi(\rho) - \min\{g_i(\pi'),0\}\nabla V_{c_i}^{\pi'}(\rho)\| \\
&= \|(\min\{g_i(\pi),0\} - \min\{g_i(\pi'),0\})\nabla V_{c_i}^\pi(\rho) + \min\{g_i(\pi'),0\}(\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho))\| \\
&\qquad\qquad \text{(Add/Subtract } \min\{g_i(\pi'),0\}\nabla V_{c_i}^\pi(\rho)\text{)} \\
&\le |\min\{g_i(\pi),0\} - \min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho)\| + |\min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho)\| \\
&\qquad\qquad \text{(Triangle inequality)} \\
&\le |g_i(\pi) - g_i(\pi')|\|\nabla V_{c_i}^\pi(\rho)\| + |\min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho)\| \\
&\qquad\qquad \text{(Since } x \mapsto \min\{x,0\} \text{ is 1-Lipschitz)} \\
&= \left|V_{c_i}^\pi(\rho) - V_{c_i}^{\pi'}(\rho)\right|\|\nabla V_{c_i}^\pi(\rho)\| + |\min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho)\| \quad \text{(Definition of } g_i(\pi)\text{)} \\
&\le \frac{\sqrt{A}}{(1-\gamma)^2}\left|V_{c_i}^\pi(\rho) - V_{c_i}^{\pi'}(\rho)\right| + |\min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho)\| \\
&\qquad\qquad \text{(Since } c_i(s,a)\in[0,1]\text{, } V_{c_i}^\pi(\rho) \text{ is } \tfrac{\sqrt{A}}{(1-\gamma)^2}\text{-Lipschitz)} \\
&\le \frac{A}{(1-\gamma)^4}\,\|\pi-\pi'\| + |\min\{g_i(\pi'),0\}|\|\nabla V_{c_i}^\pi(\rho) - \nabla V_{c_i}^{\pi'}(\rho)\| \quad \text{(By the mean value theorem)} \\
&\le \frac{A}{(1-\gamma)^4}\,\|\pi-\pi'\| + |\min\{g_i(\pi'),0\}|\,\frac{2\,\gamma\,A}{(1-\gamma)^3}\,\|\pi-\pi'\| \\
&\qquad\qquad \text{(Since } c_i(s,a)\in[0,1]\text{, by Lemma 18, } V_{c_i}^\pi(\rho) \text{ is } \tfrac{2\,\gamma\,A}{(1-\gamma)^3} \text{ smooth)} \\
&\le \left(\frac{A}{(1-\gamma)^4} + \frac{2\,\gamma\,A}{(1-\gamma)^3}\,\max\left\{b_i + \frac{\lambda_i}{\beta}, 0\right\}\right)\|\pi-\pi'\|, \\
&\qquad\qquad (V_{c_i}^{\pi'}(\rho) \ge 0 \implies g_i(\pi') \ge -(b_i + \lambda_i/\beta))
\end{aligned}$$

Putting everything together,

$$\begin{aligned}
&\|\nabla_\pi \mathcal{L}^\beta(\pi,\lambda) - \nabla_\pi \mathcal{L}^\beta(\pi',\lambda)\| \\
&\le \frac{2\,\gamma\,A}{(1-\gamma)^3}\,\|\pi-\pi'\| + \beta\sum_{i=1}^m \left(\frac{A}{(1-\gamma)^4} + \frac{2\,\gamma\,A}{(1-\gamma)^3}\,\max\left\{b_i + \frac{\lambda_i}{\beta}, 0\right\}\right)\|\pi-\pi'\| \\
&= \left(\frac{2\,\gamma\,A}{(1-\gamma)^3}\left(1 + \sum_{i=1}^m \max\{\beta b_i + \lambda_i, 0\}\right) + \frac{\beta\,m\,A}{(1-\gamma)^4}\right)\|\pi-\pi'\| \\
&\le \left(\frac{2\,\gamma\,A}{(1-\gamma)^3}(1 + \beta\|b\|_1 + \|\lambda\|_1) + \frac{\beta\,m\,A}{(1-\gamma)^4}\right)\|\pi-\pi'\|. \\
&\qquad\qquad (b_i \ge 0 \implies \max\{\beta b_i + \lambda_i, 0\} \le \beta b_i + |\lambda_i|)
\end{aligned}$$

which completes the proof □

Proposition 3. *In Algorithm 1 (equivalently, Algorithm 2) with $\lambda_1 = 0$, for all $t \ge 1$, $\lambda_t \in \mathbb{R}_+^m$.*

Proof. We prove $\lambda_t \in \mathbb{R}_+^m$ by induction. The base case holds since $\lambda_1 = 0$. Suppose $\lambda_t \in \mathbb{R}_+^m$ for some $t \ge 1$, and fix $i \in [m]$. Defining $u_i := \langle \mu^{\pi_{t+1}}, c_i\rangle - b_i$, the slack chosen by the algorithm is

$\xi_i(\mu^{\pi_{t+1}}) = \max\{u_i - {}^{[\lambda_t]_i}/_{\beta}, 0\}$, so

$$u_i - \xi_i(\mu^{\pi_{t+1}}) = \min\left\{u_i, \frac{[\lambda_t]_i}{\beta}\right\} \leq \frac{[\lambda_t]_i}{\beta}.$$

Hence, by the dual update,

$$[\lambda_{t+1}]_i = [\lambda_t]_i - \frac{\beta}{2}(u_i - \xi_i(\mu^{\pi_{t+1}})) \geq [\lambda_t]_i - \frac{[\lambda_t]_i}{2} = \frac{[\lambda_t]_i}{2} \geq 0,$$

which completes the induction. □

Lemma 17. *Under Assumption 4, in Algorithm 1, using $\epsilon_t = \frac{\sigma}{t^2}$, $\sigma > 0$, $\beta > 0$ for all outer-iterations $t \in [T]$, the general utility pseudo-reward of the* AL *defined in Proposition 2 satisfies Assumption 2 with $U = 1 + \beta \|b\|_1 + \sqrt{m}\sqrt{M^2 + \frac{\beta\sigma\pi^2}{3}} + M$ where $M := \frac{1}{\min_{i\in[m]} \zeta_i(1-\gamma)}$.*

Proof. For a fixed outer-iteration $t \in [T]$, recall from Proposition 2 the general utility pseudo-reward of the AL in Eq. (2) is

$$R(\pi) = r - \beta \sum_{i=1}^{m} c_i \min\left\{V_{c_i}^{\pi}(\rho) - b_i - \frac{[\lambda_t]_i}{\beta}, 0\right\} = r + \beta \sum_{i=1}^{m} c_i \max\left\{b_i + \frac{[\lambda_t]_i}{\beta} - V_{c_i}^{\pi}(\rho), 0\right\}$$
(Since $-\min(a, 0) = \max(-a, 0)$)

Hence for any $(s, a) \in \mathcal{S} \times \mathcal{A}$,

$$\begin{aligned}
[R(\pi)](s,a) &\leq r(s,a) + \beta \sum_{i=1}^{m} c_i(s,a)\left(b_i + \frac{[\lambda_t]_i}{\beta}\right) && (V_{c_i}^{\pi}(\rho) \geq 0 \text{ and } \lambda_t \in \mathbb{R}_+^m \text{ by Proposition 3}) \\
&\leq 1 + \beta\|b\|_1 + \|\lambda_t\|_1 && (\text{Since } r(s,a), c_i(s,a) \in [0,1]) \\
&\leq 1 + \beta\|b\|_1 + \|\lambda_t - \lambda^*\|_1 + \|\lambda^*\|_1 \\
& && (\text{Add/Subtract } \lambda^* \text{ and using triangle inequality}) \\
&\leq 1 + \beta\|b\|_1 + \sqrt{m}\,\|\lambda_t - \lambda^*\|_2 + \|\lambda^*\|_1 && (\text{Since } \|x\|_1 \leq \sqrt{m}\|x\|_2 \text{ for } x \in \mathbb{R}^m) \\
&\leq 1 + \beta\|b\|_1 + \sqrt{m}\sqrt{\|\lambda_1 - \lambda^*\|_2^2 + 2\,\beta \sum_{i=1}^{t-1} \epsilon_i} + \|\lambda^*\|_1 \\
& && (\text{By following the argument in Lemma 10 where } \epsilon_i = \tfrac{\sigma}{i^2} \text{ for } \sigma > 0) \\
&\leq 1 + \beta\|b\|_1 + \sqrt{m}\sqrt{\|\lambda^*\|_2^2 + \frac{\beta\sigma\pi^2}{3}} + \|\lambda^*\|_1 \\
& && (\lambda_1 = 0 \text{ and } \textstyle\sum_{i=1}^{t-1} \frac{\sigma}{i^2} \leq \sum_{i=1}^{\infty} \frac{\sigma}{i^2} = \frac{\pi^2\sigma}{6}) \\
&\leq 1 + \beta\|b\|_1 + \sqrt{m}\sqrt{M^2 + \frac{\beta\sigma\pi^2}{3}} + M, \\
& && (\text{Under Assumption 4, by Lemma 19 } \|\lambda^*\|_2 \leq \|\lambda^*\|_1 \leq M := \tfrac{1}{\min_{i\in[m]} \zeta_i\,(1-\gamma)})
\end{aligned}$$

□

F Large-Scale Experiments on Continuous Control Tasks

The code to reproduce the experiments is available here.

F.1 Key Results

We evaluate our framework on three MuJCo tasks (each with a single constraint) from Safety-Gymnasium [Ji et al., 2023] and compare against standard on-policy PG methods in the constrained setting: `PPOLag` [Ray et al., 2019] and `CPO` [Achiam et al., 2017].[1]

To instantiate the oracle-solver in Algorithm 1, we use `PPO` [Schulman et al., 2017], and the more recent `SPMA` algorithm [Asad et al., 2024]. We choose `SPMA` since it has been shown to achieve competitive performance on MuJoCo tasks and, unlike `PPQA`, it employs a surrogate that easily generalizes to continuous action spaces. To ensure a fair comparison with baseline methods, we match the total number of PG estimates (i.e., samples used to form the gradients) across all algorithms. Note that the framework itself is agnostic to this choice; other PG algorithms such as `TRPO` [Schulman et al., 2015] could be used interchangeably.

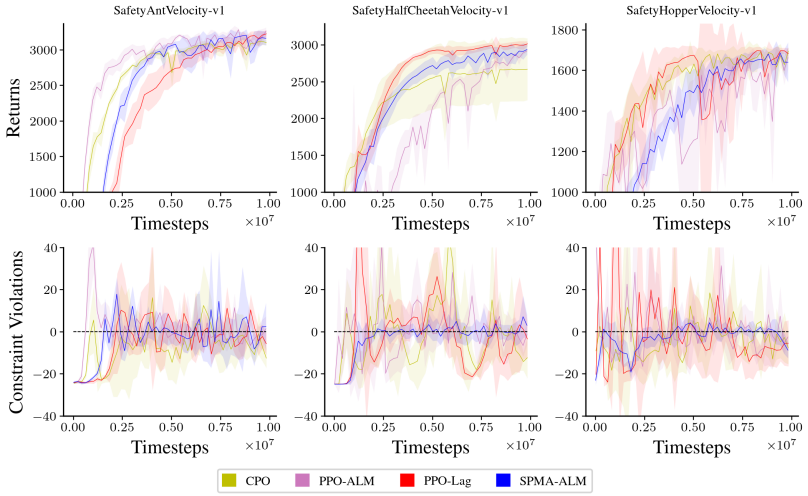

Figure 4: For each experiment, we use 5 seeds and report their 95% confidence intervals. Across all environments, `SPMA-ALM` stays closest to the constraint boundary and exhibits the least oscillatory behavior.

Results. Fig. 4 shows `AL`-based methods achieves comparable performance to the baseline approaches, justifying the practical effectiveness of our framework. During the experiments, we observed that several Lagrangian implementations in `omnisafe` [Ji et al., 2024] rely on heavily tuned heuristics, such as scaling the objective by the multiplier λ, despite the lack of theoretical support for this practice. For consistency and fairness, this modification was removed from all baseline methods. Our `ALM` variants does not require this heuristic, and all results are obtained under a unified formulation of the `AL` framework. Unlike methods such as `PPO-Lag` or `CPO`, our approach is backed by a clear theoretical foundation, yielding a method that is both principled and practical.

For clarity, we list the performance of the final policy from Fig. 4 in Table 2,

[1] We omit `ReLoad` since their implementation for the deep RL setting was not publicly available.

Table 2: Results of the final policy in Fig. 4

	SafetyAntVelocity-v1		SafetyHalfCheetahVelocity-v1		SafetyHopperVelocity-v1	
Algorithm	**Return**	**CV**	**Return**	**CV**	**Return**	**CV**
SPMA-ALM	**3202.29** $\pm$ **79.14**	-2.68 $\pm$ 5.69	2933.20 $\pm$ 69.90	-0.74 $\pm$ 3.34	**1687.37** $\pm$ 7.64	-6.06 $\pm$ 2.27
PPO-ALM	3042.76 $\pm$ 161.36	2.40 $\pm$ 5.45	2933.91 $\pm$ 21.05	-7.06 $\pm$ 1.87	1644.06 $\pm$ 45.45	-9.92 $\pm$ 2.85
PPO-Lag	3223.58 $\pm$ 52.21	-4.96 $\pm$ 2.50	**3010.19** $\pm$ **37.12**	-7.74 $\pm$ 3.30	1675.45 $\pm$ 42.51	-6.56 $\pm$ 9.12
CPO	3011.19 $\pm$ 123.74	-11.26 $\pm$ 5.13	2685.90 $\pm$ 419.20	6.18 $\pm$ 31.42	1682.68 $\pm$ 40.78	3.94 $\pm$ 23.10

F.2 Experimental Details

The Safety MuJoCo benchmark [Ji et al., 2023] augments standard continuous-control tasks from Gymnasium [Towers et al., 2024] with a velocity-based safety signal. Each task is formulated as the following constrained MDP: $\max_\pi V_r^\pi(\rho) \quad \text{s.t} \quad V_c^\pi(\rho) \leq b$. In this setting, the constraint reward is treated as a safety-related quantity that the agent must keep below a specified threshold, with $b = 25$ Descriptions of the individual environments without constraints can be found in Towers et al. [2024].

Hyperparameters are selected via a grid search, and each variant is run with five random seeds. For each configuration, we select the policy with the highest return among those satisfying the relaxed constraint $V_c^\pi(\rho) \leq b + \varepsilon$. Each seed evaluates its final policy over 10 independent episodes, resulting in 50 total rollouts used to estimate $V_c^\pi(\rho)$. For SPMA, to set step-size η for the idealized policy update, we perform a grid search over the following fixed values: $\{0.001, 0.01, 0.1, 0.3, 0.5, 0.7, 0.9\}$. A complete list of hyperparameters used in the experiments are presented in Table 3.

These experiments were conducted on compute cluster entirely on an CPU. using an AMD EPYC 9655 (Zen 5) processors. Each individual run took around ≈ 4 CPU-hours to complete. The total computational budget used for these experiments was approximately 800 CPU-hours.

Table 3: Hyper-parameters for the Safety-Gymnasium experiments.

Hyperparameter	SPMA-ALM	PPO-ALM	PPO-LAG	CPO
Minibatch size	{64, 256}	{64, 256}	{64, 256}	128
Probability ratio clipping	✘	{0.1, 0.2 ,0.3}	{0.1, 0.2, 0.3}	✘
Adam step-size (surrogate)	{3e-3, 3e-2, 3e-1}	{3e-3, 3e-2, 3e-1}	{3e-3, 3e-2, 3e-1}	0.001
Adam step-size (dual)	{1e-4, 1e-3, 1e-2, 1e-1}	{1e-4, 1e-3, 1e-2, 1e-1}	{1e-4, 1e-3, 1e-2, 1e-1}	0.001
Gradient clipping	✘	✘	✓	✓
Number of surrogate loop updates (N)	10	10	10	10
AL Updates	10	10	✘	✘
AL Penalty	{1e-3, 1e-2, 1e-1}	{1e-3, 1e-2, 1e-1}	✘	✘
Steps per Epoch	2,000	2,000	20,000	20,000
Normalize Lagrangian	✘			
Observation Normalization Wrapper	✓			
Action Rescale Wrapper	✓			
Standard Advantage	✓			
Total number of samples	10^7			
GAE λ	0.95			
Discount factor	0.99			

F.3 Pseudo-Code for SPMA-ALM

For simplicity, we present the pseudo-code for SPMA-ALM in the single-constraint setting. We omit implementation details such as the surrogate optimizer, the dual step-size, and the value networks.The corresponding pseudo-code for PPO-ALM is analogous.

Algorithm 5: SPMA-ALM

Input: $\bar{\theta}_1$ (initial paramter of policy) $\lambda_1 = 0$ (dual variable), $\beta > 0$ (constant constraint penalty), T (number of iterations), K (number of AL updates), N (number of surrogate updates) η (policy step-size), $\eta_{\text{surrogate}}$ (surrogate step-size) η_{dual} (dual step-size)
for $t = 1, 2, \ldots, T$ **do**
  Initialize AL sub-problem with $\theta_1 = \bar{\theta}_t$ (equivalent policy $\pi_t = \pi_{\bar{\theta}_t}$)
  for $k = 1, 2, \ldots K$ **do**
    Interact with environment using $\pi_k = \pi_{\theta_k}$ and construct estimate $\hat{V}_c^{\pi_k}(\rho)$, $\hat{A}_r^{\pi_k}$, $\hat{A}_c^{\pi_k}$.
    if $V_c^{\pi_k}(\rho) \geq b_1 + \frac{\lambda_t}{\beta}$ **then**
      $\hat{A}^{\pi_k}_{R(\pi_k)} = \hat{A}_r^{\pi_k}$
    else
      $\hat{A}^{\pi_k}_{R(\pi_k)} = \hat{A}_r^{\pi_k} + \hat{A}_c^{\pi_k}\left(b + \frac{\lambda_t}{\beta} - \hat{V}_c^{\pi_k}(\rho)\right)\beta$
    end if
    Construct SPMA surrogate:

$$\ell_k(\theta) = \mathbb{E}_{s\sim d^{\pi_k}} \mathbb{E}_{a\sim\pi_k(\cdot|s)} \left[\hat{A}^{\pi_k}_{R(\pi_k)}(s,a) + \frac{1}{\xi} \log\left(\frac{\pi_\theta(a|s)}{\pi_k(a|s)}\right)\right]$$

    Initialize: $\omega_0 = \theta_k$
    for $n = 1, 2, \ldots, N$ **do**
      $\omega_{n+1} = \omega_n - \eta_{\text{surrogate}} \nabla_\omega \ell_k(\omega_n)$
    end for
    $\theta_{k+1} = \omega_{N+1}$
  end for
  Recover approximation solution to AL-subproblem: $\bar{\theta}_t + 1 = \theta_{K+1}$; $\pi_{t+1} = \pi_{K+1} = \pi_{\theta_{K+1}}$
  $\lambda_{t+1} = \lambda_t - \eta_{\text{dual}}\left(\hat{V}_c^{\pi_{t+1}}(\rho) - (b_i + \xi(\pi_{t+1}))\right)$ where $\xi(\pi) = \max\left\{\hat{V}_c^{\pi}(\rho) - b - \frac{\lambda_t}{\beta}, 0\right\}$
end for
Return π_{T+1}

F.4 Sensitivity Analysis Of The Final Policy

In Fig. 5, we investigate the sensitivity of how well the final policy satisfies the constraints. We report mean, worst-case, 90th (P90), and 95th (P95) percentiles in Table 4. Overall, `SPMA-ALM` achieves the most consistent constraint satisfaction across environments, with lower mean and tail violations.

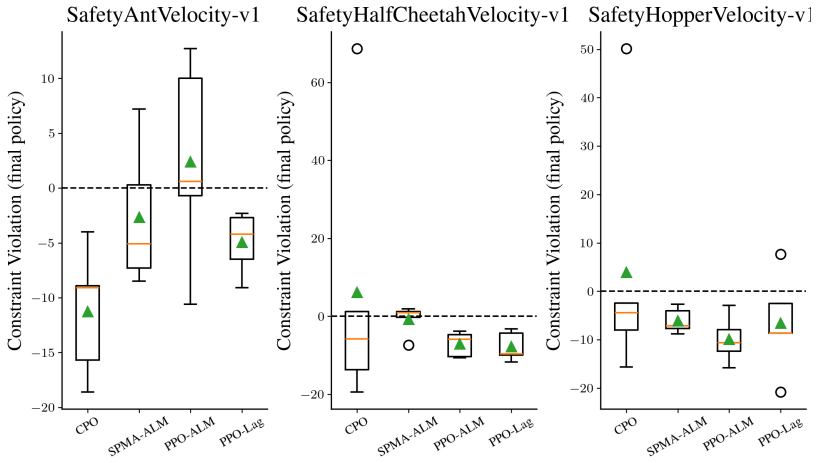

Figure 5: Box plot of constraint violations of the final policy, evaluated over 10 rollouts per seed across 10 seeds. The green triangle denotes the mean, and the orange line indicates the median.

Table 4: Constraint violation statistics over the final policy across in Fig. 4

	SafetyAntVelocity-v1				**SafetyHalfCheetahVelocity-v1**				**SafetyHopperVelocity-v1**			
Algorithm	Mean	Worst	P90	P95	Mean	Worst	P90	P95	Mean	Worst	P90	P95
`CPO`	-11.26	-4.00	-5.96	-4.98	6.18	68.60	41.64	55.12	3.94	50.10	29.10	39.60
`SPMA-ALM`	-2.68	7.20	4.44	5.82	-0.74	1.90	1.62	1.76	-6.06	-2.70	-3.22	-2.96
`PPO-ALM`	2.40	12.70	11.62	12.16	-7.06	-3.80	-4.16	-3.98	-9.92	-2.90	-4.90	-3.90
`PPO-Lag`	-4.96	-2.30	-2.46	-2.38	-7.74	-3.20	-3.64	-3.42	-6.56	7.70	3.62	5.66

F.5 Effect of the Number of AL Iterations and Penalty β

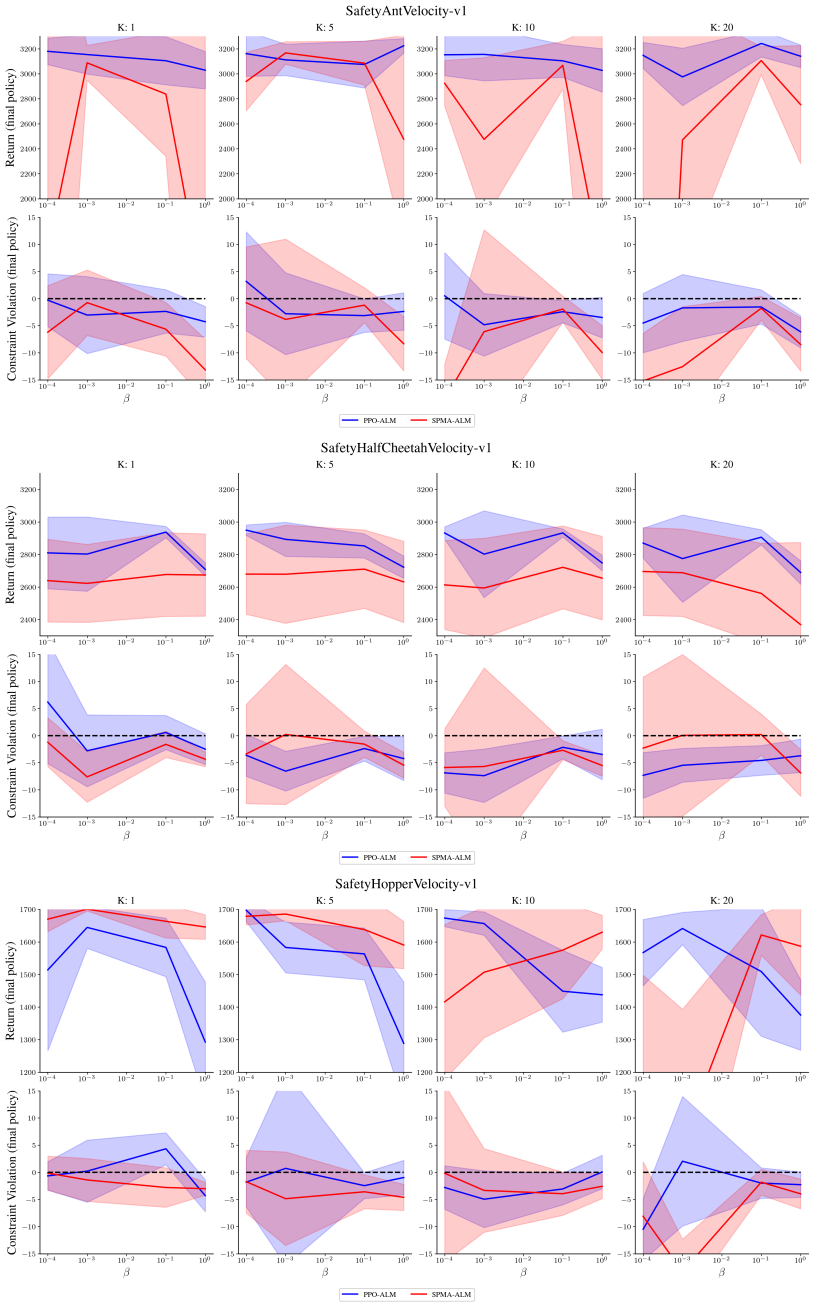

Figure 6: Ablation study on the initial penalty parameter (β) and the number of AL iterations (K) of the final policy. We vary $\beta \in \{0.0001, 0.001, 0.1, 1\}$ and the number of AL iterations $K \in \{1, 5, 10, 20\}$, while keeping all other hyperparameters to their individually best-tuned values. Performance is evaluated for the final policy using 10 rollouts per seed across 10 seeds. The shaded regions indicate 95% confidence intervals. We observe that performance is generally stable for smaller β and exhibits limited sensitivity to K.

F.6 Effect of the Number of AL Iterations and Dual Step-size

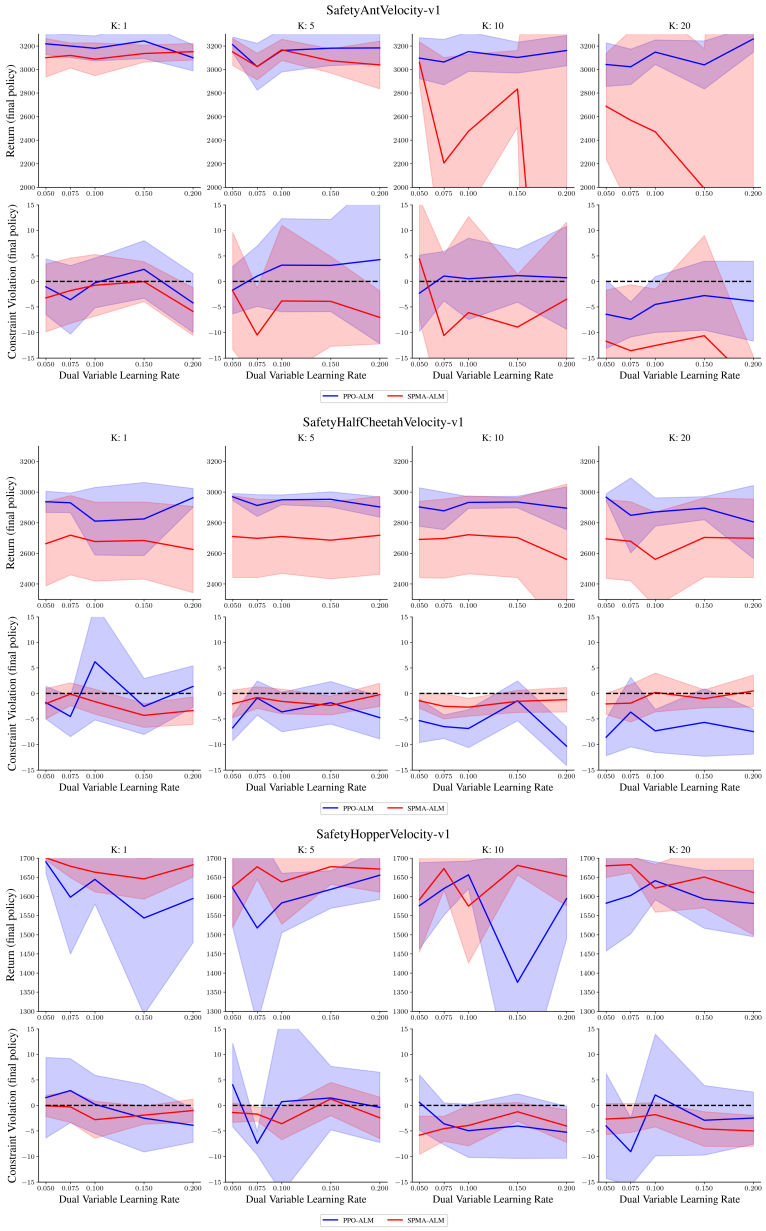

Figure 7: Ablation study on the dual step-size and the number of AL iterations (K) of the final policy. Performance is evaluated for the final policy using 10 rollouts per seed across 10 seeds. We vary the dual variable step-size: $\{0.05, 0.075, 0.1, 0.15, 0.2\}$ and the number of AL iterations $K \in \{1, 5, 10, 20\}$, while keeping all other hyperparameters fixed to their individually best-tuned values. The shaded regions indicate 95% confidence intervals. We find that SPMA-ALM is more sensitive to the dual step-size compared to PPO-ALM.

G Tabular Experiments

The code to reproduce the experiments is available here.

G.1 Environmental Details

In each of the following environments, we set the initial state distribution to be uniform, i.e. for all $s \in \mathcal{S}$, $\rho(s) = \frac{1}{S}$.

Constrained Cliff World: This environment modifies the original Cliff World [Sutton and Barto, 2018, Example 6.6]. The environment consists of 21 states and 4 actions. The objective is for an agent to each the goal state while avoiding a cliff. If the agent falls into the chasm, the agent receives a reward of 0 and a cost of -1. If the agent reaches the goal, the agent receives a reward of $+1$. All other rewards and costs are 0. In this environment $\gamma = 0.9$ and $b = -0.17$.

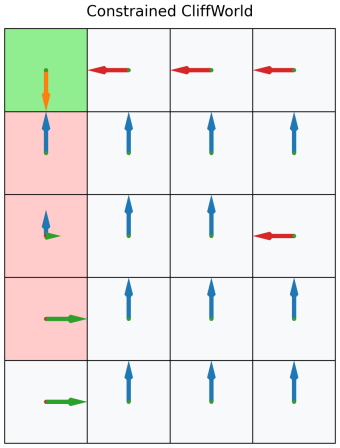

Figure 8: Visualization of the optimal policy for the cliff world environment. A longer arrows indicates a higher the probability of taking a particular action. The red regions represents the cliff and the green region indicates the goal.

Constrained Deep Sea Treasure This environment modifies the original Deep sea treasure [Osband et al., 2019]: The environment consists 25 states and 2 actions. The agent begin from the top-left corner of the grid and descends one row per each time it takes an action. The goal of the agent is to stay left in order to reach the treasure. If the agent transitions to the right, it receives a reward of -0.02. Otherwise if the agent reaches the treasure, it receives a reward of $+1$. Additionally, in the environment there are two landmines. If the agent reaches a landmine, the agent receives a cost of -2. All other costs are 0. In this environment $\gamma = 0.9$ and $b = -0.1$.

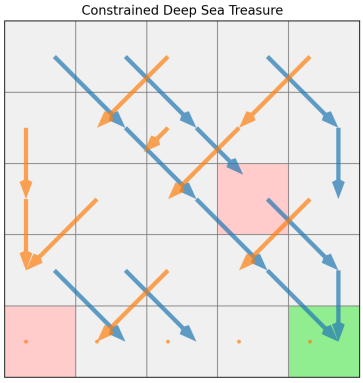

Figure 9: Visualization of the optimal policy for the deep sea treasure environment. A longer arrows indicates a higher the probability of taking a particular action. The red regions represents the landmine and the green region indicates the goal.

G.2 Experimental Details & Additional Results

For all methods, hyperparameters were selected via grid search, and the full configurations are listed in Table 5. We report results using the configuration that achieved the smallest optimality gap subject to the final policy satisfying the constraint violation $b - V_c^{\pi_{T+1}}(\rho) \leq \varepsilon$ where $\varepsilon \in \{0.001, 0.01\}$. Overall, `PQA-ALM` perform well in practice.

Table 5: Grid search ranges for the tabular experiments. n denotes the number of iterates used to update the corresponding surrogate.

Method	**Fixed Params.**	**Search Space**
`PQA-ALM`	$T = 10$, $K = 100$, $\beta = 10$	Primal step-size$\{0.1, 1, 10\}$
`PPO-Lag` [Ray et al., 2019]	$T = 10$, $K = 100$,	Primal step-size: $\{0.01, 0.1, 1, 10, 100\}$ Dual step-size: $\{0.01, 0.1, 1, 10, 100\}$ Clip: $\{0.1, 0.2, 0.3\}$
`RPG-PD` [Ding et al., 2023]	$T = 1000$	Entropy Coefficient $\{10^{-4}, 10^{-2}, 10^{-1}\}$ Primal step-size: $\{0.01, 0.1, 1, 10, 100\}$ Dual step-size: $\{0.01, 0.1, 1, 10, 100\}$
`NPG-DD` [Ying et al., 2022]	$T = 100$, $K = 10$	Entropy Coefficient: $\{0.001, 0.01, 0.1, 1\}$ Dual step-size: $\{0.001, 0.01, 0.1, 1\}$
`ReLOAD` [Moskovitz et al., 2023]	$T = 1000$	Primal step-size $\{0.01, 0.1, 1, 10, 100\}$ Dual step-size $\{0.01, 0.1, 1, 10, 100\}$

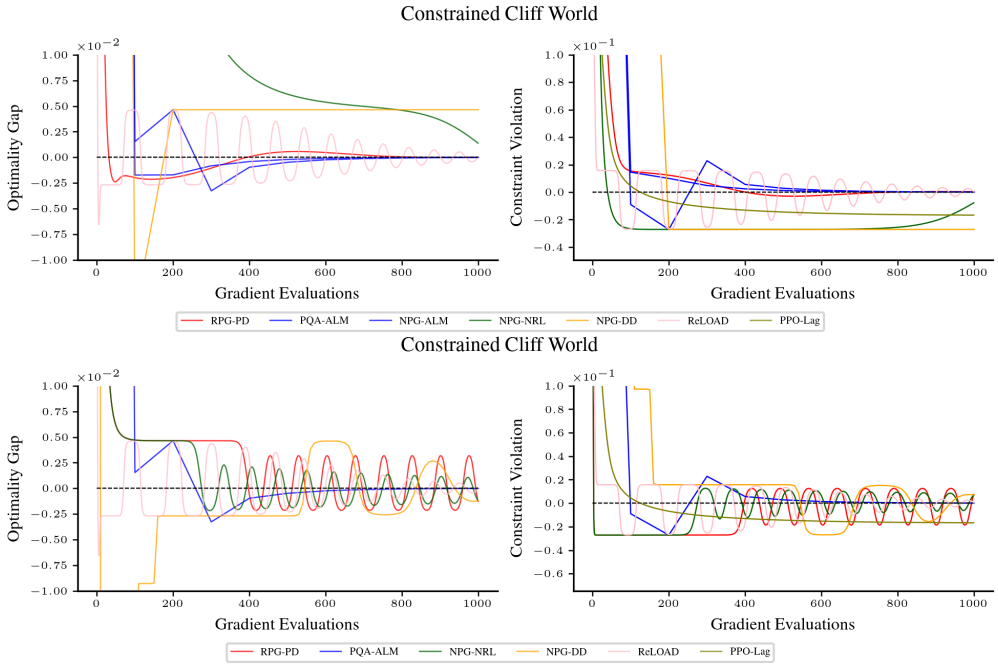

Figure 10: Constrained Cliff World with $\varepsilon = 0.001$ (top) and $\varepsilon = 0.01$ (bottom)

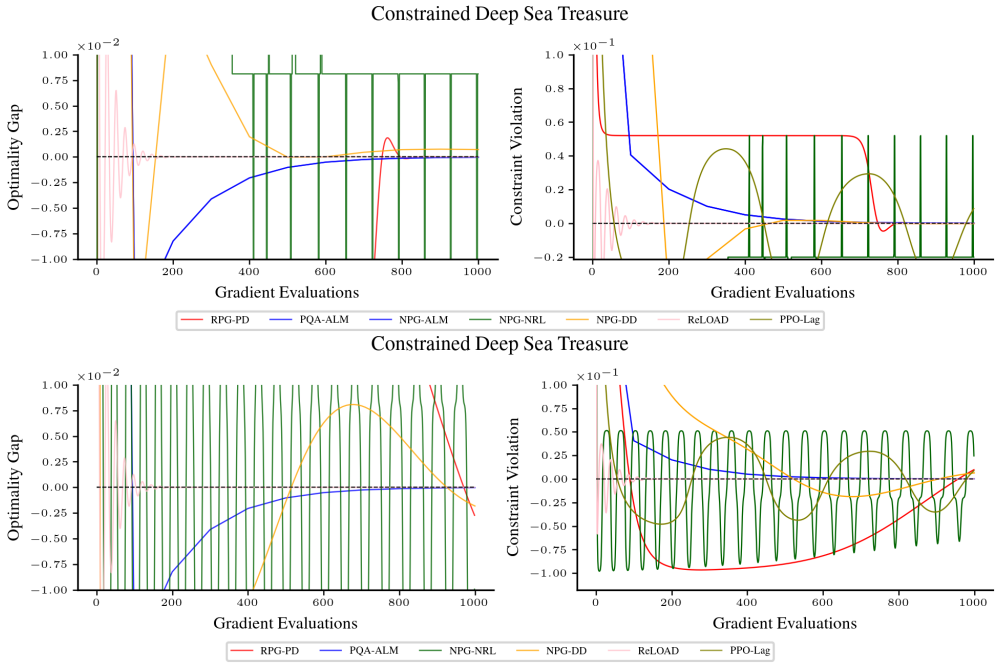

Figure 11: Constrained Deep Sea Treasure with $\varepsilon = 0.001$ (top) and $\varepsilon = 0.01$ (bottom)

H Linear Function Approximation Experiments

To construct the features φ, following Vaswani et al. [2023], Asad et al. [2024] we use tile-code features [Sutton and Barto, 2018]. Tilde-coded features require three parameters to be set: (i) hash table size (feature dimension d), (ii) number of tiles N, and (iii) size of tiles s. Increasing N increases the express ability of the features Increasing s controls how "spread out" the features are. For both environments, we consider the following feature set: $\{(40, 4, 1), (60, 4, 3), (80, 6, 5)\}$

To extend ReLOAD to the linear function approximation setting, we follow the policy-based version of ReLOAD [Moskovitz et al., 2023, Algorithm 6], using estimated Q-functions under the compatible function approximation theorem [Agarwal et al., 2021]. Specifically for the Lin-NPG-PD and ReLOAD, to compute the policy gradient estimate, we solve the following regression problem

$$\min_{w \in \mathbb{R}^d} \mathbb{E}_{(s,a)\sim\mu^\pi}[(\langle \nabla \log \pi_\theta(a|s), w\rangle - Q^\pi_\diamond(s,a))^2$$

with an appropriate pseudo-reward $\diamond$. A complete list of grid-search range is provided in Table 6.

Table 6: Grid search ranges for each method. n denotes the number of iterates used to update the corresponding surrogate.

Method	**Fixed Params.**	**Search Space**
PPQA-ALM	$T = 10$, $K = 100$, $n = 250$, $\beta = 10$	Primal step-size$\{0.0001, 0.001, 0.1, 1.0, 10\}$ Surrogate step-size$\{0.0001, 0.001, 0.1, 1.0, 10\}$
PPO-Lag [Ray et al., 2019]	$T = 1000$, $n = 250$,	Primal step-size: $\{0.01, 0.1, 1, 10, 100\}$ Dual step-size: $\{0.01, 0.1, 1, 10, 100\}$ Surrogate step-size$\{0.0001, 0.001, 0.1, 1.0, 10\}$ Clip: $\{0.1, 0.2, 0.3\}$
Lin-NPG-PD [Ding et al., 2023]	$T = 1000$,	Primal step-size $\{0.0001, 0.001, 0.1, 1, 10\}$; Dual step-size $\{0.0001, 0.001, 0.1, 1, 10\}$ Entropy Coefficient $\{0.001, 0.01, 0.1, 1\}$
ReLOAD [Moskovitz et al., 2023]	$T = 1000$	Primal step-size $\{0.0001, 0.01, 0.1, 1\}$ Dual step-size $\{0.01, 0.1, 1, 10, 100\}$

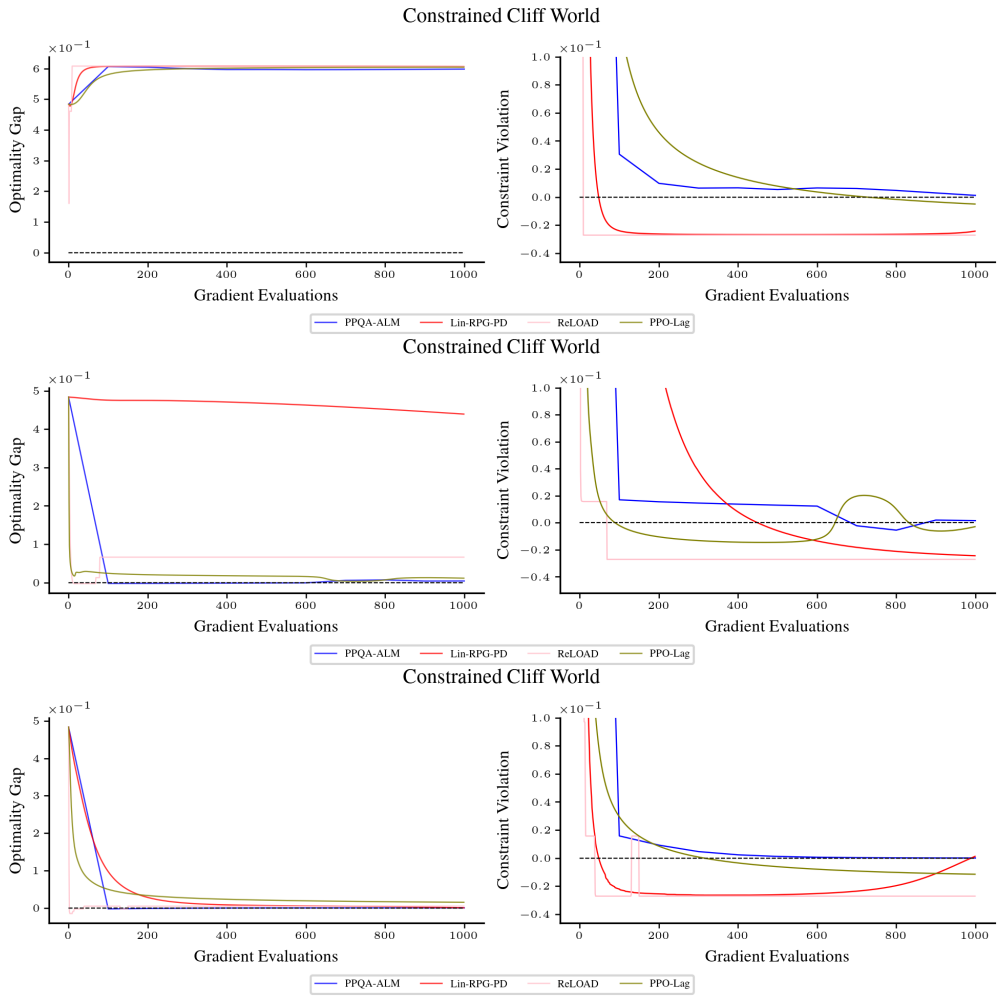

Figure 12: Constrained Cliff world with $\varepsilon = 0.001$ and $d = 40$ (top), $d = 60$ (middle), $d = 80$ (bottom) features

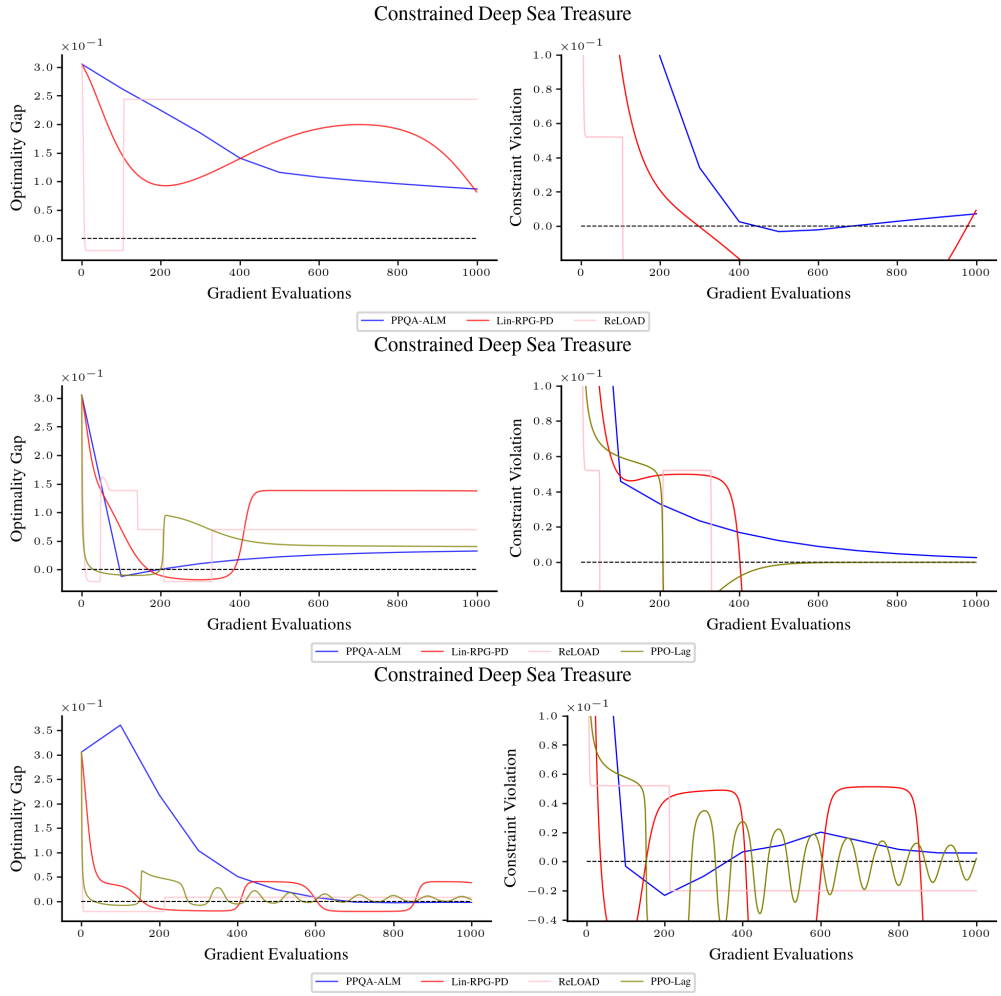

Figure 13: Constrained Deep Sea Treasure with $\varepsilon = 0.001$ and $d = 40$ (top), $d = 60$ (middle), $d = 80$ (bottom) features

I External Results

For completeness, we append external results here.

Theorem 6 (Theorem 1 in Kumar et al. [2024]). *For any given general utility $G : \Theta \to \mathbb{R}$, the policy gradient of G is*

$$\nabla_\theta G(\pi) = \frac{1}{1-\gamma} \sum_{s \in \mathcal{S}} d^\pi(s) \sum_{a \in \mathcal{A}} Q^\pi_{R(\pi)}(s,a) \frac{d\pi(a|s)}{d\theta}$$

where $R(\pi) \coloneqq \frac{d\mathcal{F}(x)}{dx}\Big|_{x=\mu^\pi} \in \mathbb{R}^{SA}$ is the general utility pseudo-reward function.

Lemma 18 (Lemma 54 in Agarwal et al. [2021]). *Under the direct parameterization, for a pseudo-reward $\diamond \in [0,1]^{SA}$, $V^\pi_\diamond(\rho)$ is $\frac{2\gamma A}{(1-\gamma)^3}$-smooth.*

Lemma 19 (Extension of Lemma 4.1 in Jain et al. [2022] to m constraints). *Under Assumption 4, Problem 1 satisfies strong duality, and any optimal dual variable $\lambda^* \in \mathbb{R}^m_+$ satisfies $\sum_{i=1}^m \lambda_i^* \zeta_i \leq \frac{1}{1-\gamma}$. In particular, $\|\lambda^*\|_2 \leq \|\lambda^*\|_1 \leq M \coloneqq \frac{1}{\min_{i\in[m]} \zeta_i (1-\gamma)}$.*

Proof. Strong duality holds since Problem 1 is equivalent to the linear program in Problem 2. To start define

$$D(\lambda) \coloneqq \max_\pi V^\pi_r(\rho) + \sum_{i=1}^m \lambda_i (V^\pi_{c_i}(\rho) - b_i)$$

Furthermore, under Assumption 4, there exist a policy $\bar{\pi}$ and constant $\zeta \in \mathbb{R}^m_+$ such that

$$V^{\bar{\pi}}_{c_i}(\rho) - b_i \geq \zeta_i \quad \forall i \in [m]$$

Using the fact that $V^{\bar{\pi}}_r(\rho) \geq 0$, we have,

$$\sum_{i=1}^m \lambda_i^* \zeta_i \leq D(\lambda^*) = \min_{\lambda \in \mathbb{R}^m_+} D(\lambda) \leq D(0) = \max_\pi V^\pi_r(\rho) \leq \frac{1}{1-\gamma},$$

$$\implies \min_i \zeta_i \|\lambda^*\|_1 \leq \sum_i \lambda_i^* \zeta_i \implies \|\lambda^*\|_2 \leq \|\lambda^*\|_1 \leq \frac{1}{\min_{i\in[m]} \zeta_i \, (1-\gamma)}$$

□

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In Section 3, we provide a general framework for the augmented Lagrangian (AL) method with last-iterate guarantees (Theorem 1). In Section 4, we prove that our framework instantiated with Projected Q-Ascent achieves last-iterate convergence (Theorem 4). In Section 5.1, we prove an analogous last-iterate rate for a variant of Projected Q-Ascent on log-linear policies in Theorem 5. Finally, we evaluate our approach on large-scale continious experiments in Appendix G.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We state and justify our assumptions in Section 4 and Section 5.1. We highlight the main limitation in the conclusion.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proof sketches are provided for the main results. The proofs of all theoretical results are presented in the supplementary material.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The experimental details for continious-control tasks can be found in Appendix F and the details for the tabular experiments can be found in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility.

In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Code to reproduce experiments may be found at https://github.com/sudo-michael/augmented-lagrangian-method-cmdp.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Details of the experiments can be found in Appendix F and Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In Appendix F, we report the 95% confidence intervals for large-scale experiments involving stochasticity.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We outline the compute resources the large-scale experiments in Appendix F. The experiments for the tabular experiments were done on laptop and are omitted.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: As our contribution are mainly theoretical, our work does not involve human subjects nor participants.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

Justification: While CMDPs can model many safety-critical applications, our work is theoretical and does not directly impact a specific direction.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: No datasets or models were released as part of this work.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The deep reinforcement learning framework`omnisafe` [Ji et al., 2024] used to conduct experiments is provided as a reference.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work did not use crowdsourcing nor the involvement of human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work did not use crowdsourcing nor the involvement of human subjects.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [No]

    Justification: An LLM was only used to aid in editing and formatting.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.