A COMPUTATIONALLY EFFICIENT AND RATE-OPTIMAL ALGORITHM FOR ADVERSARIAL LINEAR MDPS WITH (AGGREGATE) BANDIT FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

We study episodic linear Markov decision processes with unknown transitions, adversarial losses, and bandit feedback. The state-of-the-art result for this setting (Liu et al., 2024a) either achieves $\sqrt{K}$ regret with a computationally inefficient algorithm or achieves suboptimal $K^{3/4}$ regret, where K is the number of episodes. Our work gives the first $\sqrt{K}$ regret with a polynomial-time algorithm.

Our method reduces adversarial linear MDPs to adversarial linear bandits whose action set $\mathcal{K}$ consists of policies' expected feature vectors. The central challenge is that $\mathcal{K}$ is unknown and can be estimated only through exploration. We address this challenge by constructing an *approximate separation oracle* for $\mathcal{K}$ that either generates executable policies from queried features, or refines an outer approximation of $\mathcal{K}$. This approach draws inspiration from the reduction of van Erven et al. (2025) for linear contextual bandits and the oracle-efficient linear bandit algorithm of Ito et al. (2019), but additionally deals with the imprecision of the oracle. The resulting algorithm requires only aggregate loss feedback, and to our knowledge, this is the first polynomial-time algorithm for adversarial MDPs with aggregate feedback beyond the tabular case.

1 INTRODUCTION

Achieving rate-optimal regret with polynomial computation is a central challenge in adversarial linear Markov decision processes (MDPs) with bandit feedback and unknown transitions. Starting with the $K^{14/15}$ regret bound of Luo et al. (2021), subsequent works improved the rate to $K^{8/9}$, $K^{6/7}$, and $K^{3/4}$ (Dai et al., 2023; Sherman et al., 2023; Liu et al., 2024a), respectively. Meanwhile, Liu et al. (2024a) obtained the optimal $\sqrt{K}$ rate using a computationally inefficient method. Whether $\sqrt{K}$ regret can be achieved with an efficient (polynomial-time) algorithm has remained open.

Our work closes this gap by giving the first computationally efficient algorithm with $\mathrm{poly}(d, H)\sqrt{K}$ regret, where d is the feature dimension and H is the horizon length. In fact, our work handles the more challenging *aggregate feedback* model: the learner, despite observing the state sequence on the trajectory, only observes the total episode loss rather than the individual step losses. Prior work for the aggregate feedback model focuses either on the tabular case (Cohen et al., 2021; Lancewicki & Mansour, 2025; Ito et al., 2025) or the stochastic-loss setting (Cassel et al., 2024; Chen et al., 2025). To our knowledge, our work gives the first polynomial-time algorithm for aggregate feedback model with adversarial losses beyond the tabular case.

Our approach builds on a reduction from linear MDPs to linear bandits. For each policy π, we stack its expected state-action features across layers as a *policy-feature vector* ϕ^π. Because the expected episode loss is linear in ϕ^π, the MDP can be viewed as a linear bandit whose action set consists of policy-feature vectors. This action set is unknown because ϕ^π depends on the unknown transition kernel. Existing methods based on this reduction (Kong et al., 2024; Liu et al., 2024a) discretize the policy class, estimate the policy-feature of every policy, and apply a linear-bandit algorithm to the resulting finite action set. While this achieves the optimal $\sqrt{K}$ regret rate (Liu et al., 2024a), their computation scales with an exponentially large policy class.

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

Under the same reduction idea, van Erven et al. (2025) develop an efficient reduction from linear contextual bandits to linear bandits. They construct a convex approximation $\widehat{\mathcal{K}}$ of $\mathcal{K}$ from past i.i.d. contexts and construct a linear optimization oracle for $\widehat{\mathcal{K}}$. This oracle allows them to run a linear bandit algorithm (continuous exponential weights) over $\widehat{\mathcal{K}}$ without explicitly representing the set. Their linear contextual bandit setting is a special case of linear MDP when $H = 1$.

Although the approach of van Erven et al. (2025) suggests a promising route for linear MDPs, a direct extension faces a major obstacle: the straightforward estimate $\widehat{\mathcal{K}}$ for linear MDP may not be convex due to the recursive structure of MDPs. Therefore, the log-concave sampling underlying continuous exponential weights no longer applies, and we also lose an exact optimization oracle over $\widehat{\mathcal{K}}$. To address this, we leverage the idea of Ito et al. (2019): we run continuous exponential weights over an outer convex set $\mathcal{O} \supseteq \mathcal{K}$ represented as an intersection of halfspaces (cuts). In Ito et al. (2019), whenever a point queried from $\mathcal{O}$ lies outside $\mathcal{K}$, an exact oracle generates a separating cut to refine $\mathcal{O}$. Unlike in their setting, however, we lack an exact oracle for $\mathcal{K}$. Thus, we construct cuts that exclude the queried point only when doing so preserves $\mathcal{O} \supseteq \mathcal{K}$ with high probability. We remark that while Ito et al. (2019) propose this mechanism to reduce oracle complexity, we use it for a completely different purpose of handling unknown action sets.

When the algorithm does not return a cut, it decomposes the queried point into a distribution over policies whose average policy feature approximates the query point. This will be the policy distribution played by the learner in that episode. Together, the two cases form a *approximate separation procedure*: it either constructs a finite policy mixture approximating the query, or returns a separating cut that refines $\mathcal{O}$. We implement it using a Frank–Wolfe-style procedure. Besides a distribution over policies, our procedure simultaneously generates uncertainty measures, which will be crucial in constructing a bias-correction bonus fed to the learning procedure.

The paper is organized as follows. In Section 2 we describe the problem setting. In Section 3, we give a high-level overview of the main algorithm. In Section 4, we provide a sketch of regret analysis. In Section 5, we focus on the design of the approximate separation procedure. In Section 6, we analyze the computational complexity of the algorithm.

2 PRELIMINARIES

We consider a finite-horizon Markov decision process with horizon H, layer-wise state sets $\mathcal{S}_1, \dots, \mathcal{S}_{H+1}$, a finite nonempty action set $\mathcal{A}$ with $A = |\mathcal{A}|$, and a fixed initial state $s_1 \in \mathcal{S}_1$. For each layer h, the transition kernel $P_h(\cdot \mid s, a)$ is a probability distribution on $\mathcal{S}_{h+1}$. The kernels are unknown to the learner and fixed across episodes. A Markov policy $\pi = (\pi_h)_{h=1}^H$ consists of maps $\pi_h : \mathcal{S}_h \to \Delta(\mathcal{A})$, where $\Delta(\mathcal{A})$ denotes the probability simplex over actions.

The learner interacts with the MDP for K episodes. Before interaction, an oblivious adversary fixes loss functions $\ell_{k,h} : \mathcal{S}_h \times \mathcal{A} \to [0,1]$ for all episodes and layers. At the start of episode k, the learner selects a possibly random Markov policy π_k using past observations. Starting from $S_{k,1} = s_1$, the trajectory is generated by $A_{k,h} \sim \pi_{k,h}(\cdot \mid S_{k,h}), S_{k,h+1} \sim P_h(\cdot \mid S_{k,h}, A_{k,h})$, for every h. We assume the loss feedback observed by the learner consists only of the total episode loss $L_k = \sum_{h=1}^H \ell_{k,h}(S_{k,h}, A_{k,h})$, revealed at the end of the episode. We refer to this observation model as *aggregate bandit feedback*.

For a Markov policy π, define its expected loss in episode k by $V_k^\pi = \mathbb{E}_{P,\pi}\big[\sum_{h=1}^H \ell_{k,h}(S_h, A_h)\big]$. Here $\mathbb{E}_{P,\pi}$ denotes expectation over the trajectory generated by following policy π under transition kernels P. The objective we consider is to minimize regret relative to the best fixed Markov policy,

$$\mathrm{Reg}_K = \mathbb{E}\left[\sum_{k=1}^K V_k^{\pi_k}\right] - \min_{\pi \text{ Markov}} \sum_{k=1}^K V_k^\pi.$$

Linear MDPs. We assume that transitions and losses are linear in known feature maps $\phi_h : \mathcal{S}_h \times \mathcal{A} \to \mathbb{R}^d$. Specifically, there exist unknown vectors $\psi_h(s') \in \mathbb{R}^d$ and $\theta_{k,h} \in \mathbb{R}^d$ such that, for every k, h, s, a, s' with $s \in \mathcal{S}_h$ and $s' \in \mathcal{S}_{h+1}$,

$$P_h(s' \mid s, a) = \phi_h(s,a)^\top \psi_h(s'), \qquad \ell_{k,h}(s,a) = \phi_h(s,a)^\top \theta_{k,h}.$$

The transition coefficients ψ_h are fixed, while the loss coefficients $\theta_{k,h}$ may vary across episodes. We assume $\|\phi_h(s,a)\|_2 \le 1$, $\|\theta_{k,h}\|_2 \le \sqrt{d}$, and $\sum_{s' \in \mathcal{S}_{h+1}} \|\psi_h(s')\|_2 \le \sqrt{d}$ without loss of generality. For every bounded function $f : \mathcal{S}_{h+1} \to \mathbb{R}$, define $w_h(f) = \sum_{s' \in \mathcal{S}_{h+1}} \psi_h(s') f(s')$ and $P_h f(s,a) = \mathbb{E}_{s' \sim P_h(\cdot|s,a)}[f(s')] = \phi_h(s,a)^\top w_h(f)$.

Let $p = dH$ be the expected-feature dimension, obtained by stacking the d-dimensional features across all H layers. Set $\theta_k = (\theta_{k,1}, \dots, \theta_{k,H})$, and let $\phi^\pi = (\phi_1^\pi, \dots, \phi_H^\pi)$ with $\phi_h^\pi = \mathbb{E}_{P,\pi}[\phi_h(S_h, A_h)]$. Then $\|\phi^\pi\|_2 \le \sqrt{H}$, $\|\theta_k\|_2 \le \sqrt{dH}$, and $\mathbb{E}[L_k \mid \pi_k = \pi] = \langle \phi^\pi, \theta_k \rangle = V_k^\pi$. Let $\mathcal{K} = \{\phi^\pi : \pi \text{ is a Markov policy}\}$ be the set of feasible policy-feature vectors. The policy-feature reduction of Liu et al. (2024a) relates regret to linear losses over this set through

$$\mathrm{Reg}_K = \mathbb{E}\left[\sum_{k=1}^K \langle \phi^{\pi_k}, \theta_k \rangle - \min_{u \in \mathcal{K}} \sum_{k=1}^K \langle u, \theta_k \rangle\right] \tag{1}$$

The set $\mathcal{K}$ depends on the unknown transitions. The reduction therefore yields an adversarial linear bandit problem over an unknown feasible set. The algorithm developed in Sections 4 and 5 achieves the following guarantee.

Theorem 2.1 (Main result)**.** *For adversarial linear MDPs with aggregate bandit feedback, Algorithm 1, with the complete implementation in Algorithm 4, achieves* $\mathrm{Reg}_K = \widetilde{O}(d^5 H^{9/2}\sqrt{K})$*. Total computation is polynomial in* K, d, H, A*.*

Notation. For a positive integer m, write $[m] = \{1, \dots, m\}$, let $\mathbf{I}_m$ be the $m \times m$ identity matrix, and denote the Euclidean unit ball by $\mathbb{B}_2^m = \{x \in \mathbb{R}^m : \|x\|_2 \le 1\}$. The indicator $\mathbf{1}\{E\}$ equals one when E holds and zero otherwise. For $\Sigma \succeq 0$, $\|x\|_\Sigma \triangleq \sqrt{x^\top \Sigma x}$. For a nonempty closed convex set $C \subseteq \mathbb{R}^m$, write $\mathrm{Proj}_C(x) = \arg\min_{z \in C} \|z - x\|_2^2$. For scalars $a \le b$, define the clipping operator by $\mathrm{CLIP}_{[a,b]}(t) = \min\{b, \max\{a, t\}\}$. For vectors, $\mathrm{CLIP}_{[a,b]^m}$ applies this operator to each coordinate. For functions, clipping is applied pointwise. Throughout, $\widetilde{O}$ hides logarithmic factors in K, d, H, A.

3 Algorithm Overview

The main algorithm is Algorithm 1. The algorithm maintains an outer approximation $\mathcal{O} \supset \mathcal{K}$ and a cumulative loss estimate $Z_{k-1} \in \mathbb{R}^p$. Since $\mathcal{K}$ may have empty interior, we actually require $\mathcal{O}$ to contain the enlarged set $\mathcal{K}_\zeta = \mathcal{K} + \zeta \mathbb{B}_2^p$, for $\zeta = 1/\sqrt{K}$ to ensure that the linear bandit algorithm is run over a set with non-zero volume. Each episode consists of the following steps:

Step 1. Construct a finite-support exploration distribution. The continuous exponential-weight distribution over $\mathcal{O}$ has density $q_k(x) \propto \exp(-\eta Z_{k-1}^\top x)\mathbf{1}\{x \in \mathcal{O}\}$. Let $\widehat{\mu}_k \in \mathbb{R}^p$ and $\widehat{\Sigma}_k \in \mathbb{R}^{p \times p}$ be estimates of the mean and covariance of q_k. Instead of sampling a point directly from q_k, we construct a uniform distribution over the following $2p$ points:

$$\left\{\widehat{\mu}_k + c_0 \widehat{\Sigma}_k^{1/2} v : \ v \in \{\pm e_1, \pm e_2, \dots, \pm e_p\}\right\} \tag{2}$$

where $e_1, \dots, e_p$ are the standard basis vectors of $\mathbb{R}^p$, $c_0 > 0$ is a universal constant. This uniform distribution has mean $\widehat{\mu}_k$ and covariance $(c_0^2/p)\widehat{\Sigma}_k$. This sampling scheme is similar to that used in SCRiBLe (Abernethy et al., 2009).

We aim for finite support because Step 2 explicitly enumerates and processes every possible support point before the learner samples a policy.

Step 2. Compile each support point or generate a cut. Denote the $2p$ points in (2) as $\{Y_{k,i}\}_{i=1}^{2p}$. For each $i \in [2p]$, ApproxSep returns exactly one of the following:

Case 1. A set of policies $\{\pi_{k,i,j}\}_{j=1}^M$ and an error certificate $\varepsilon_{k,i}$ such that the average policy feature $\frac{1}{M}\sum_{j=1}^M \phi^{\pi_{k,i,j}}$ approximates $Y_{k,i}$, with approximation error quantified by $\varepsilon_{k,i}$.

Case 2. A cut (c, b) such that $c^\top Y_{k,i} < b$ while, with high probability, $c^\top x \ge b$ for all $x \in \mathcal{K}_\zeta$.

Algorithm 1 Continuous exponential weights with policy compilation and adaptive cuts

Require: $c_0 = \frac{1}{16e}$, $\zeta = K^{-1/2}$, $R = \sqrt{H} + \zeta$, $M = K$, and $\eta \in (0, \frac{c_0}{128\sqrt{d^3H^4}}]$.

1: $Z_0 \leftarrow 0$; $\mathcal{O} \leftarrow R\mathbb{B}_2^p$; $\mathcal{D}_{1,h} \leftarrow \varnothing$ for all $h \in [H]$
2: **for** $k = 1, \ldots, K$ **do**
3: **repeat**
4: *restart*$\leftarrow$ FALSE
5: Define $q_k(x) \propto \exp(-\eta Z_{k-1}^\top x)\mathbf{1}\{x \in \mathcal{O}\}$
6: Estimate mean and covariance: $(\widehat{\mu}_k, \widehat{\Sigma}_k) \leftarrow$ MOMENTESTIMATE$(\mathcal{O}, Z_{k-1})$ (Section D.1)
7: Let $\{Y_{k,i}\}_{i=1}^{2p}$ be the set $\left\{\widehat{\mu}_k + c_0\widehat{\Sigma}_k^{1/2}v : \ v \in \{\pm e_1, \pm e_2, \ldots, \pm e_p\}\right\}$.
8: **for** $i = 1, \ldots, 2p$ **do**
9: reply $\leftarrow$ APPROXSEP$(Y_{k,i}, (\mathcal{D}_{k,h})_{h=1}^H, M, \zeta)$ (Algorithm 2)
10: **if** reply is a cut (c, b) **then**
11: $\mathcal{O} \leftarrow \mathcal{O} \cap \{x : c^\top x \geq b\}$
12: *restart*$\leftarrow$ TRUE; **break**
13: **else**
14: $((\pi_{k,i,j})_{j=1}^M, \varepsilon_{k,i}) \leftarrow$ reply
15: **until** *restart*$=$ FALSE
16: Draw $(I_k, J_k) \sim \mathrm{Unif}([2p] \times [M])$ and execute $\pi_k = \pi_{k,I_k,J_k}$.
17: Observe feature vectors along the trajectory $(\phi_{k,h})_{h=1}^H$ and the total loss L_k.
18: $\widehat{\theta}_k \leftarrow \left(\zeta^2\mathbf{I}_p + \frac{c_0^2}{p}\widehat{\Sigma}_k\right)^{-1}(Y_{k,I_k} - \widehat{\mu}_k)L_k$
19: $Z_k \leftarrow Z_{k-1} + \widehat{\theta}_k - B_k$ where $B_k = \frac{10\eta p\varepsilon_k}{c_0}Z_{k-1}$ with $\varepsilon_k = \sqrt{\frac{1}{2p}\sum_{i=1}^{2p}\varepsilon_{k,i}^2}$.
20: $\mathcal{D}_{k+1,h} \leftarrow \mathcal{D}_{k,h} \cup \{(\phi_{k,h}, S_{k,h+1})\}$ for all $h \in [H]$

If the return is Case 1 for all $2p$ points, the algorithm will proceed to Step 3; otherwise, the cut will be used to refine $\mathcal{O}$, and the algorithm will go back to Step 1 to generate new points $\{Y_{k,i}\}_{i=1}^{2p}$ with the refined $\mathcal{O}$. At a high level, APPROXSEP implements an *approximate separation oracle* for $\mathcal{K}_\zeta$. The approximation error $\varepsilon_{k,i}$ is constructed with confidence bounds from past transition data. APPROXSEP returns a cut only when the confidence bounds certify that the queried point is outside $\mathcal{K}_\zeta$. We will also show that: with high probability, the number of cuts produced by APPROXSEP is upper bounded by $\mathrm{poly}(p, K)$ during the learning procedure, so the algorithm will finally proceed to the next step.

Step 3. Sample a policy. If APPROXSEP returns Case 1 for every $i \in [2p]$, the learner draws (I_k, J_k) uniformly from $[2p] \times [M]$ and executes π_{k,I_k,J_k}. Conditional on the completed preparation and I_k, averaging the policy feature over J_k gives $\frac{1}{M}\sum_{j=1}^M \phi^{\pi_{k,I_k,j}}$, which approximates Y_{k,I_k} within ε_{k,I_k} by the guarantee in Case 1. Therefore, the overall algorithm approximates a standard linear bandit algorithm with the sampling scheme (2). After execution, the constructed loss estimator $\widehat{\theta}_k$ and a bias-correction bonus B_k are used to update Z_k.

4 SKETCH OF REGRET ANALYSIS AND BONUS DESIGN

Let $\mu_k = \mathbb{E}_{X\sim q_k}[X]$ and $\Sigma_k = \mathrm{Cov}_{X\sim q_k}(X)$ denote the exact mean and covariance of q_k after preparation of episode k; $\widehat{\mu}_k$ and $\widehat{\Sigma}_k$ are their estimates. Write $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_k]$, where $\mathcal{F}_k$ is the information available after preparation and before drawing (I_k, J_k).

The key step is the APPROXSEP procedure in Step 2 in Section 3. In this section, we provide a skeleton of regret analysis. Then in Section 5, we provide detailed explanation for APPROXSEP.

Recall that in episode k, Algorithm 1 forms $2p$ *target* points (Line 7 of Algorithm 1). For each target $Y_{k,i} \in \mathbb{R}^p$, the APPROXSEP procedure (Line 9 of Algorithm 1) either returns a cut or a list of M policies and a certificate $\varepsilon_{k,i}$. Let $x_{k,i} = \frac{1}{M}\sum_{j=1}^M \phi^{\pi_{k,i,j}}$. Then Lemma 5.2 shows $\|x_{k,i} - Y_{k,i}\|_2 \leq \varepsilon_{k,i}$. Further define $\overline{x}_k = \frac{1}{2p}\sum_{i=1}^{2p} x_{k,i}$. Since $\widehat{\mu}_k = \frac{1}{2p}\sum_{i=1}^{2p} Y_{k,i}$, Jensen's

inequality gives

$$\|\overline{x}_k - \widehat{\mu}_k\|_2 \le \varepsilon_k := \sqrt{\tfrac{1}{2p}\sum_{i=1}^{2p} \varepsilon_{k,i}^2}. \tag{3}$$

Essentially, $\widehat{\mu}_k$ is the mean feature suggested by the underlying linear bandit algorithm, while $\overline{x}_k$ is the true expected feature of the policy executed by the learner. Their mismatch indicates that the underlying linear bandit algorithm is facing *misspecified* or *corrupted* feedback.

Quantifying bias and injecting bias-correction bonus. To handle misspecification, we inject bias-correction bonus B_k into the continuous exponential weight update (Liu et al., 2024b; van Erven et al., 2025). To understand the design of B_k, we perform the following analysis. When the regret comparator has policy feature u, the true regret in episode k is $\langle \overline{x}_k - u, \theta_k\rangle$, while the regret seen by the exponential weight algorithm is $\langle \mu_k - u, \widehat{\theta}_k - B_k\rangle$. Their difference is

$$\underbrace{\langle \overline{x}_k - u, \theta_k\rangle}_{\text{true regret}} - \underbrace{\langle \mu_k - u, \mathbb{E}_k[\widehat{\theta}_k] - B_k\rangle}_{\text{perceived regret}} = \underbrace{\langle \overline{x}_k - \mu_k, \theta_k\rangle}_{\text{feature error}} + \underbrace{\langle \mu_k - u, \theta_k - \mathbb{E}_k[\widehat{\theta}_k]\rangle}_{\text{loss estimator bias}} + \underbrace{\langle \mu_k - u, B_k\rangle}_{\text{correction term}}. \tag{4}$$

The first term (i.e., feature error) comes from the mismatch between the estimated and true policy feature. It is upper bounded by $\sqrt{p}\big(\varepsilon_k + \|\widehat{\mu}_k - \mu_k\|_2\big)$ by (3) with the assumption $\|\theta_k\|_2 \le \sqrt{p}$. The term $\|\widehat{\mu}_k - \mu_k\|_2$ can be made negligible by running MOMENTESTIMATE with high enough accuracy. Therefore, this part is dominated by $\sqrt{p}\varepsilon_k$.

The second term (i.e., loss estimator bias) comes from the bias of $\widehat{\theta}_k$. It can be upper bounded as $\langle \mu_k - u, \theta_k - \mathbb{E}_k[\widehat{\theta}_k]\rangle \le \|\mu_k - u\|_{\Sigma_k^{-1}} \|\theta_k - \mathbb{E}_k[\widehat{\theta}_k]\|_{\Sigma_k}$. Lemma B.2 gives $\|\theta_k - \mathbb{E}_k[\widehat{\theta}_k]\|_{\Sigma_k} \le \frac{5p\varepsilon_k}{2c_0}$, and hence

$$\text{loss estimator bias} = \langle \mu_k - u, \theta_k - \mathbb{E}_k[\widehat{\theta}_k]\rangle \le \tfrac{5p\varepsilon_k}{2c_0}\|\mu_k - u\|_{\Sigma_k^{-1}}. \tag{5}$$

This bias depends on the regret comparator u and can be large if distribution q_k puts low weight on u. The solution to this term has become standard (Lee et al., 2020; Zimmert & Lattimore, 2022; Liu et al., 2024b): The continuous exponential weight $q_k(x) \propto \exp(-\eta Z_{k-1}^\top x)\mathbf{1}\{x \in \mathcal{O}\}$ over $x \in \mathbb{R}^p$ can be viewed as Follow-the-Regularized-Leader with a p-self-concordant barrier (Bubeck & Eldan, 2015). Lemma B.1 therefore gives $\|\mu_k - u\|_{\Sigma_k^{-1}} \le -4\eta\langle \mu_k - u, Z_{k-1}\rangle + 24p$. Combining with (5) and recalling $B_k = \frac{10\eta p\varepsilon_k}{c_0} Z_{k-1}$, we can bound the sum of the last two terms in (4) as

$$\text{loss estimator bias} + \text{correction term} = \langle \mu_k - u, \theta_k - \mathbb{E}_k[\widehat{\theta}_k] + B_k\rangle \le \tfrac{60p^2\varepsilon_k}{c_0}.$$

This explains why we include the bonus term B_k in Line 19 of Algorithm 1. The bonus removes the comparator-dependent term $O(p^2\varepsilon_k\|\mu_k - u\|_{\Sigma_k^{-1}})$ from the bias upper bound and introduces a comparator-independent overhead $O(p^2\varepsilon_k)$. Overall, the discrepancy between the true regret and the perceived regret is of order $O\big(p^2\sum_{k=1}^K \varepsilon_k\big)$.

Bounding the perceived regret. The perceived regret in (4) follows the standard analysis of exponential weights. Even though the decision sets change over time, since they are always shrinking and the true regret comparator always lies in $\mathcal{K} \subseteq \mathcal{O}$ with high probability, the standard FTRL analysis applies (Lemma B.4).

The true regret is then bounded by the sum of the perceived regret and the discrepancy:

Lemma 4.1 (Regret bound)**.** *Algorithm 1 with the optimal choice of η ensures*

$$\mathrm{Reg}_K \le \tilde{\mathcal{O}}\Bigg(\underbrace{d^{3/2}H^{5/2}\sqrt{K}}_{\text{perceived regret}} + \underbrace{d^2H^2\sqrt{K\mathbb{E}\left[\sum_{k=1}^K \varepsilon_k^2\right]}}_{\text{bias + correction}}\Bigg). \tag{6}$$

To obtain $\widetilde{O}(\sqrt{K})$ regret from Eq. (6), it remains to bound $\sum_k \varepsilon_k^2$ by polynomial factors in d, H and logarithmic factors in K. Section 5 establishes this guarantee, and we give the final bound for the "bias + correction" term above in Section 5.3.

5 FROM A TARGET FEATURE VECTOR TO A POLICY LIST

This section builds APPROXSEP, the procedure in Algorithm 1 that performs *approximate separation*:

Input: $y \in \mathbb{R}^p$

Output: exactly one of the following:

- $\left(\{\pi_j\}_{j=1}^M, \varepsilon\right)$ such that $\|y - \frac{1}{M}\sum_{j=1}^M \phi^{\pi_j}\|_2 \le \varepsilon$.
- (c, b) such that $c^\top y < b$ and $c^\top x \ge b$ for all $x \in \mathcal{K}_\zeta$.

Figure 1: Approximate separation — realized by APPROXSEP (Algorithm 2)

To build approximate separation, we leverage the ability to perform *approximate linear optimization*:

Input: $c \in \mathbb{R}^p$

Output: $(\pi, \widehat{\phi}, \Delta)$ such that $\|\widehat{\phi} - \phi^\pi\|_2 \le \Delta$ and $c^\top \widehat{\phi} \le \min_{x\in\mathcal{K}} c^\top x + \|c\|_2\Delta$.

Figure 2: Approximate linear optimization — realized by APPROXLO (Algorithm 3)

In the analysis below, we will see how ε in Fig. 1 depends on Δ in Fig. 2 (Line 9 of Algorithm 2). In implementing approximate linear optimization, the learner has access to previous transition data $(\mathcal{D}_{k,h})_{h=1}^H$ as the source of information for the unknown set $\mathcal{K}$. The error Δ will be related to confidence bound of the transition (see (10)).

Since π is selected based on the same data used to estimate its feature, we establish the confidence bounds underlying Δ uniformly over all possible output policies. We obtain these bounds by restricting the policy class to have controlled complexity and applying a covering argument, allowing the same data to be reused across adaptive linear-optimization queries.

In Section 5.1, we focus on building approximate separation from approximate linear optimization. Then in Section 5.2, we investigate the implementation of approximate linear optimization.

5.1 CONSTRUCTING THE POLICY LIST OR EXCLUDING THE TARGET

This subsection constructs APPROXSEP (Algorithm 2), assuming access to APPROXLO. The algorithm can be viewed as an inexact Frank–Wolfe procedure for the objective $f(x) = \frac{1}{2}\|x - y\|_2^2$ over $\mathcal{K}$. As in standard Frank–Wolfe, the iterate is an average over extreme points. Thus, the procedure also naturally produces a policy list whose average feature approximates the target y.

Algorithm 2 APPROXSEP $(y, (\mathcal{D}_{k,h})_{h=1}^H, M, \zeta)$

Require: Target $y \in \mathbb{R}^p$, transition data $(\mathcal{D}_{k,h})_{h=1}^H$, policy count M, enlargement radius ζ.

1: $\widehat{x}_0 \leftarrow 0$.
2: **for** $j = 1, \dots, M$ **do**
3: $\quad c_j \leftarrow \widehat{x}_{j-1} - y$
4: $\quad (\pi_j, \widehat{\phi}_j, \Delta_j) \leftarrow$ APPROXLO$(c_j; (\mathcal{D}_{k,h})_{h=1}^H)$ (Algorithm 3)
5: $\quad b_j \leftarrow c_j^\top \widehat{\phi}_j - \|c_j\|_2 (\Delta_j + \zeta)$
6: $\quad$ **if** $c_j^\top y < b_j$ **then**
7: $\quad\quad$ **return** cut (c_j, b_j)
8: $\quad \widehat{x}_j \leftarrow \frac{j-1}{j}\widehat{x}_{j-1} + \frac{1}{j}\widehat{\phi}_j$
9: **return** $((\pi_j)_{j=1}^M, \varepsilon)$, where $\varepsilon = \sqrt{4\zeta^2 + \frac{6}{M}\sum_{j=1}^M \Delta_j^2 + \frac{8R^2(1+\log M)}{M}}$

First, we ensure that if APPROXLO performs as desired, then APPROXSEP returns a cut only when $y \notin \mathcal{K}_\zeta$. This is shown in the following lemma:

Lemma 5.1. *Suppose* APPROXLO *(called in Line 4 of Algorithm 2) satisfies the property in Fig. 2 and assume $y \in \mathcal{K}_\zeta$. Then* APPROXSEP *will NOT return a cut.*

Proof. By the guarantee of APPROXLO, we have

$$c_j^\top \widehat{\phi}_j \le \min_{x\in\mathcal{K}} c_j^\top x + \|c_j\|_2\Delta_j \le \big(\min_{x\in\mathcal{K}_\zeta} c_j^\top x + \zeta\|c_j\|_2\big) + \|c_j\|_2\Delta_j \le c_j^\top y + \|c_j\|_2(\Delta_j+\zeta)$$

where we use the assumption $y \in \mathcal{K}_\zeta$ in the last inequality. Thus, the "if" condition in Line 7 of Algorithm 2 does not hold, and the algorithm will not return a cut. □

Next, we ensure that if no cut is returned by APPROXSEP, the approximation error for y is controlled.

Lemma 5.2. *Suppose* APPROXLO *(called in Line 4 of Algorithm 2) satisfies the property in Fig. 2 and suppose* APPROXSEP *does NOT return a cut. Then* $\big\|\frac{1}{M}\sum_{j=1}^M \phi^{\pi_j} - y\big\|_2 \le \varepsilon$ *(defined in Line 9).*

Proof. This proof is related to the analysis for Frank-Wolfe algorithms. For any $j = 1, \dots, M$,

$$\begin{aligned}
\|\widehat{x}_j - y\|^2 &= \left\|\left(1-\frac{1}{j}\right)\widehat{x}_{j-1} + \frac{1}{j}\widehat{\phi}_j - y\right\|^2 && \text{(Line 8)}\\
&= \|\widehat{x}_{j-1}-y\|^2 - \frac{2}{j}\langle \widehat{x}_{j-1}-y, \widehat{x}_{j-1}-\widehat{\phi}_j\rangle + \frac{1}{j^2}\|\widehat{x}_{j-1}-\widehat{\phi}_j\|^2\\
&\le \|\widehat{x}_{j-1}-y\|^2 - \frac{2}{j}\|\widehat{x}_{j-1}-y\|^2 - \frac{2}{j}\langle c_j, y-\widehat{\phi}_j\rangle + \frac{4R^2}{j^2} && (c_j = \widehat{x}_{j-1}-y)\\
&\le \|\widehat{x}_{j-1}-y\|^2 - \frac{2}{j}\|\widehat{x}_{j-1}-y\|^2 + \frac{2}{j}\|c_j\|(\Delta_j+\zeta) + \frac{4R^2}{j^2}\\
& && \text{(the "if" condition in Line 7 of Algorithm 2 does not hold)}\\
&\le \left(1-\frac{1}{j}\right)\|\widehat{x}_{j-1}-y\|^2 + \frac{2\Delta_j^2+2\zeta^2}{j} + \frac{4R^2}{j^2}. && (c_j = \widehat{x}_{j-1}-y \text{ and AM-GM})
\end{aligned}$$

Expanding the recursion using Lemma C.1 yields

$$\|\widehat{x}_M - y\|_2^2 \le 2\zeta^2 + \frac{2}{M}\sum_{j=1}^M \Delta_j^2 + \frac{4R^2(1+\log M)}{M}. \tag{7}$$

Let $x = \frac{1}{M}\sum_{j=1}^M \phi^{\pi_j}$. Since $\widehat{x}_M = \frac{1}{M}\sum_{j=1}^M \widehat{\phi}_j$, by the assumed property of APPROXLO,

$$\|x - \widehat{x}_M\|_2^2 \le \frac{1}{M}\sum_{j=1}^M \left\|\phi^{\pi_j} - \widehat{\phi}_j\right\|_2^2 \le \frac{1}{M}\sum_{j=1}^M \Delta_j^2. \tag{8}$$

Combining (8) and (7) and using $\|x-y\|_2^2 \le 2\|x-\widehat{x}_M\|_2^2 + 2\|\widehat{x}_M - y\|_2^2$ finishes the proof. □

Together, the two lemmas establish the desired dichotomy: feasible targets are never excluded, while any target not excluded is approximated by a policy list with controlled error. The next subsection relates this error to confidence bounds for the unknown transitions.

5.2 PLANNING AND EVALUATING FROM PAST TRANSITIONS

In this subsection, we develop APPROXLO that achieves the property in Fig. 2, with the help of the transition data $(\mathcal{D}_{k,h})_{h=1}^H$ collected before. The algorithm is adapted from the standard Least-Square Value Iteration (LSVI) procedure (Jin et al., 2020; Wang et al., 2020) and is displayed in Algorithm 3. Its guarantee is summarized in the following theorem:

Theorem 5.3 (Certified approximate linear optimization)**.** *Let $\beta = C_\beta d\sqrt{\log(KdHA/\delta)}$, where C_β is a sufficiently large universal constant. Then, with probability at least $1-\delta$, simultaneously for every episode k and every call to Algorithm 3, the returned tuple $(\pi, \widehat{\phi}, \Delta)$ satisfies $\widehat{\phi} \in \sqrt{H}\mathbb{B}_2^p$, $\Delta \in [0, 2\sqrt{H}]$, and*

$$\|\widehat{\phi} - \phi^\pi\|_2 \le \Delta, \qquad c^\top\widehat{\phi} - \|c\|_2\,\Delta \le \min_{u\in\mathcal{K}} c^\top u. \tag{9}$$

Moreover, the returned certificate satisfies

$$\Delta \le 10\sqrt{p}\,\beta^2 H\,\mathbb{E}_{P,\pi}\left[\sum_{h=1}^H \|\phi_h(S_h,A_h)\|_{\Lambda_{k,h}^{-1}}\right] + H\tau\log A. \tag{10}$$

Algorithm 3 APPROXLO $(c, (\mathcal{D}_{k,h})_{h=1}^{H})$

Require: $c \in \mathbb{R}^p$, dataset $(\mathcal{D}_{k,h})_{h=1}^{H}$.
Parameters: $\tau = 1/(H\sqrt{K}\log(eA))$ and β as in Theorem 5.3.
Define: $\Lambda_{k,h} = \mathbf{I}_d + \sum_{(u,s')\in\mathcal{D}_{k,h}} uu^\top$, $\nu_{k,h}(s,a) = \|\phi(s,a)\|_{\Lambda_{k,h}^{-1}}$, and $\bar{c} = c/\|c\|_2$.
Define: $\widehat{P}_{k,h} f(s,a) = \phi_h(s,a)^\top \Lambda_{k,h}^{-1} \sum_{(u,s')\in\mathcal{D}_{k,h}} u f(s')$ for $f : \mathcal{S} \to \mathbb{R}$.

1: Let $V_{H+1}^{\mathrm{pl}}(s) = 0 \in \mathbb{R}$ and $V_{H+1}^{\mathrm{ft}}(s) = 0 \in \mathbb{R}^p$ and $\widehat{U}_{H+1}(s) = 0 \in \mathbb{R}$ for all $s \in \mathcal{S}_{H+1}$.
2: **for** $h = H, H-1, \ldots, 1$ **do**
3: // Task 1: policy optimization: choose the policy π
4: $Q_h^{\mathrm{pl}}(s,a) \leftarrow \mathrm{CLIP}_{[-\sqrt{H},\sqrt{H}]}\big(\phi_h(s,a)^\top \bar{c}_h + \widehat{P}_{k,h} V_{h+1}^{\mathrm{pl}}(s,a) - (\sqrt{H}+1)\beta\nu_{k,h}(s,a)\big)$.
5: $V_h^{\mathrm{pl}}(s) \leftarrow -\tau \log \sum_a \exp(-Q_h^{\mathrm{pl}}(s,a)/\tau)$.
6: $\pi_h(a \mid s) \propto \exp(-Q_h^{\mathrm{pl}}(s,a)/\tau)$.
7: // Task 2: feature evaluation: estimate ϕ^π
8: $r_h^{\mathrm{ft}}(s,a) \leftarrow (0,\ldots,0,\phi_h(s,a),0,\ldots,0) \in \mathbb{R}^p$.
9: $Q_h^{\mathrm{ft}}(s,a) \leftarrow \mathrm{CLIP}_{[-1,1]^p}(r_h^{\mathrm{ft}}(s,a) + \widehat{P}_{k,h} V_{h+1}^{\mathrm{ft}}(s,a))$
10: $V_h^{\mathrm{ft}}(s) \leftarrow \sum_a \pi_h(a \mid s) Q_h^{\mathrm{ft}}(s,a)$
11: // Task 3: uncertainty evaluation: upper bound U_k^π
12: $Q_h^{\mathrm{unc}} \leftarrow \mathrm{CLIP}_{[0,H-h+1]}((1+H\beta)\nu_{k,h} + \widehat{P}_{k,h}\widehat{U}_{h+1})$
13: $\widehat{U}_h(s) \leftarrow \sum_a \pi_h(a \mid s) Q_h^{\mathrm{unc}}(s,a)$
14: $\widehat{\phi} \leftarrow \mathrm{Proj}_{\sqrt{H}\mathbb{B}_2^p}(V_1^{\mathrm{ft}}(s_1))$.
15: $\Delta \leftarrow \min\{5\sqrt{p}\beta\widehat{U}_1(s_1) + H\tau\log A, 2\sqrt{H}\}$
16: **return** $(\pi, \widehat{\phi}, \Delta)$

Below, we overview the elements in Algorithm 3.

Task 1: Policy optimization. We use optimistic LSVI (Jin et al., 2020) to find an approximately optimal policy under the cost function $\phi_h(s,a)^\top c_h$. Different from Jin et al. (2020), we output a Gibbs (softmin) policy (Line 6 of Algorithm 3) instead of a greedy one, and use soft state value (Line 5). The reason of doing this is to make the mapping from c to the output policy feature $\widehat{\phi}$ be Lipschitz, leading to a bounded log covering number for the class of mappings (Appendix C). The clipping for Q_h^{pl} is at $\sqrt{H}$ because $|\langle\phi(s,a),\bar{c}\rangle| \le \|\phi(s,a)\|_2\|\bar{c}\|_2 \le \sqrt{H}$ (Line 4). This implies $|V_h^{\mathrm{pl}}(s)| \le \sqrt{H} + \tau\log A \le \sqrt{H}+1$.

The soft value preserves optimism because $V_h^{\mathrm{pl}}(s) \le \min_a Q_h^{\mathrm{pl}}(s,a)$, giving $V_1^{\mathrm{pl}}(s_1) \le \min_{u\in\mathcal{K}} \bar{c}^\top u$. Thus, to obtain the second guarantee in (9), it suffices to bound $\bar{c}^\top\widehat{\phi} - V_1^{\mathrm{pl}}(s_1)$. This difference can be decomposed into

$$\bar{c}^\top\widehat{\phi} - V_1^{\mathrm{pl}}(s_1) = \underbrace{\bar{c}^\top(\widehat{\phi} - \phi^\pi)}_{\text{feature estimation error}} + \underbrace{\bar{c}^\top\phi^\pi - V_1^{\mathrm{pl}}(s_1)}_{\text{planning error}} \tag{11}$$

The feature-evaluation error is handled in the feature-evaluation step below. The planning error comes from Gibbs smoothing and the transition-prediction errors, which sum to $\bar{c}^\top\phi^\pi - V_1^{\mathrm{pl}}(s_1) \le 2(\sqrt{H}+1)\beta U_k^\pi + H\tau\log A$, where $U_k^\pi = \mathbb{E}_{P,\pi}[\sum_h \nu_{k,h}(S_h,A_h)]$.

Task 2: Feature evaluation. Now fix the Gibbs policy π constructed in the policy-optimization step, and estimate its policy feature ϕ^π. Define vector $r_h^{\mathrm{ft}}(s,a) \in \mathbb{R}^p$ as in Line 8. Then we have

$$\mathbb{E}_{P,\pi}\left[\sum_{h=1}^{H} r_h^{\mathrm{ft}}(S_h,A_h)\right] = \big(\mathbb{E}_{P,\pi}[\phi_1(S_1,A_1)],\ldots,\mathbb{E}_{P,\pi}[\phi_H(S_H,A_H)]\big) = \phi^\pi.$$

Thus estimating ϕ^π is a policy-evaluation problem with p-dimensional *vector rewards*. We estimate ϕ^π by backward policy evaluation, obtaining $\widehat{\phi} \in \sqrt{H}\mathbb{B}_2^p$. A backward induction shows that the

estimation error is controlled by U_k^π, giving $\|\widehat{\phi}-\phi^\pi\|_2 \le \sqrt{p}\,\beta U_k^\pi$. Since $\|\bar{c}\|_2 \le 1$, $\left|\bar{c}^\top\widehat{\phi}-\bar{c}^\top\phi^\pi\right| \le \sqrt{p}\,\beta U_k^\pi$. Adding this estimation error to the policy optimization bound (11) gives

$$\bar{c}^\top\widehat{\phi} - \left[(\sqrt{p}+2\sqrt{H}+2)\beta U_k^\pi + H\tau\log A\right] \le V_1^{\mathrm{pl}}(s_1) \le \min_{u\in\mathcal{K}} \bar{c}^\top u. \tag{12}$$

Thus both inequalities in (9) hold if we choose $\Delta \ge 5\sqrt{p}\beta U_k^\pi + H\tau\log A$. However, the learner cannot compute this threshold because U_k^π is an expectation under the unknown transitions. We next obtain an estimated upper bound of U_k^π, from which we can choose Δ.

Task 3: Uncertainty evaluation. We use the same backward policy evaluation to estimate U_k^π. This can be achieved by viewing $\nu_{k,h}(s,a) \in [0,1]$ as the reward. Since $\|\widehat{U}_{h+1}\|_\infty \le H$, we add the bonus $(1+\beta H)\nu_{k,h}$ to guard against underestimating future uncertainty. A backward induction gives $U_k^\pi \le \widehat{U}_1(s_1) \le 2\beta H U_k^\pi$. Using the computable upper bound $\widehat{U}_1(s_1)$ in place of U_k^π in the feature evaluation step, APPROXLO $(c;(\mathcal{D}_{k,h})_{h=1}^H)$ (Algorithm 3) returns $(\pi,\widehat{\phi},\Delta)$ with $\Delta = \min\{5\sqrt{p}\,\beta\widehat{U}_1(s_1) + H\tau\log A, 2\sqrt{H}\}$, where the clipping to $2\sqrt{H}$ is because $\|\phi-\phi'\|_2 \le 2\sqrt{H}$ for any policy features ϕ,ϕ'.

The above definition of Δ together with $\widehat{U}_1(s_1) \le 2\beta H U_k^\pi$ gives (10).

5.3 FINAL REGRET BOUND

Proof of Theorem 2.1. Now we combine elements to establish the final regret bound. The sum of misspecification in the "bias + correction" term in Lemma 4.1 can be bounded as

$$\begin{aligned}
&\mathbb{E}\left[\sum_{k=1}^K \frac{1}{2p}\sum_{i=1}^{2p}\varepsilon_{k,i}^2\right] \\
&= \mathbb{E}\left[\sum_{k=1}^K\left(4\zeta^2 + \frac{8R^2(1+\log M)}{M} + \frac{6}{M}\frac{1}{2p}\sum_{i=1}^{2p}\sum_{j=1}^{M}\Delta_{k,i,j}^2\right)\right] && \text{(Line 9 of Algorithm 2)} \\
&\le \widetilde{O}\left(H + \mathbb{E}\left[\sum_{k=1}^K \Delta_{k,I_k,J_k}^2\right]\right) && \text{((I_k,J_k) uniformly drawn from $[2p]\times[M]$ in Algorithm 1)} \\
&\le \widetilde{O}\left(H + \mathbb{E}\left[\sum_{k=1}^K p\beta^4 H^3 \sum_{h=1}^H \|\phi_h(S_{k,h},A_{k,h})\|_{\Lambda_{k,h}^{-1}}^2\right]\right) && \text{(Theorem 5.3)} \\
&\le \widetilde{O}\left(d^6H^5\right), && \text{(Elliptical potential lemma: } \mathbb{E}\Big[\textstyle\sum_{k,h}\|\phi_h(S_{k,h},A_{k,h})\|_{\Lambda_{k,h}^{-1}}^2\Big] \le \widetilde{O}(dH))
\end{aligned}$$

Plugging this in Lemma 4.1 gives the desired bound. □

6 COMPUTATION

We bound the computation time of Algorithm 1 by analyzing the costs of MOMENTESTIMATE and APPROXSEP, together with the number of cuts.

The MOMENTESTIMATE procedure and the desired guarantees are same as those in (Ito et al., 2019). We directly use the results in their Section 4.2 to bound the complexity of MOMENTESTIMATE by $\mathrm{poly}(d,H,K,\log(1/\delta))$ calls to the membership oracle of $\mathcal{O}$.

The total number of cuts added in each episode is upper bounded by $\widetilde{O}(p+\sqrt{H}K^{3/2})$ with high probability (see Lemma D.3). To prevent the low probability event to lead to non-termination of the algorithm, we present a version of Algorithm 1 that explicitly counts the number of restarts and executes the fallback plan if the count exceeds a certain threshold (Algorithm 4).

The APPROXLO procedure (Algorithm 3) can be run in $\mathrm{poly}(d,H,K,A)$ time, even though the displayed Algorithm 3 enumerates the updates on all state-action pairs. This is common in prior work for linear MDP (Jin et al. (2020)). For completeness, we explicitly describe the implementation in Section D.3.

AI USE STATEMENT

An OpenAI assistant was used for manuscript editing, lemma development and consistency checks. The authors independently verified the mathematics, references, and final submission.

REFERENCES

Jacob D Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *Conference on Learning Theory*, number 110, 2009.

Sébastien Bubeck and Ronen Eldan. The entropic barrier: a simple and optimal universal self-concordant barrier. *Conference on Learning Theory*, 2015.

Asaf Cassel, Haipeng Luo, Aviv Rosenberg, and Dmitry Sotnikov. Near-optimal regret in linear mdps with aggregate bandit feedback. *International Conference on Machine Learning*, 2024.

Fan Chen, Zeyu Jia, Alexander Rakhlin, and Tengyang Xie. Outcome-based online reinforcement learning: Algorithms and fundamental limits. *Advances in Neural Information Processing Systems*, 38, 2025.

S. Chewi. The entropic barrier is n-self-concordant. In *Geometric Aspects of Functional Analysis*, pp. 209–222. Springer, 2023.

A. Cohen, H. Kaplan, T. Koren, and Y. Mansour. Online Markov decision processes with aggregate bandit feedback. In *Proceedings of the 34th Conference on Learning Theory*, pp. 1301–1329, 2021.

Yan Dai, Haipeng Luo, Chen-Yu Wei, and Julian Zimmert. Refined regret for adversarial mdps with linear function approximation. In *International Conference on Machine Learning*, pp. 6726–6759. PMLR, 2023.

S. Ito, D. Hatano, H. Sumita, K. Takemura, T. Fukunaga, N. Kakimura, and K.-I. Kawarabayashi. Oracle-efficient algorithms for online linear optimization with bandit feedback. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

S. Ito, K. G. Jamieson, H. Luo, A. Maiti, and T. Tsuchiya. Adapting to stochastic and adversarial losses in episodic MDPs with aggregate bandit feedback. In *Advances in Neural Information Processing Systems*, 2025.

C. Jin, Z. Yang, Z. Wang, and M. I. Jordan. Provably efficient reinforcement learning with linear function approximation. In *Proceedings of the 33rd Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 2137–2143, 2020. Full version: arXiv:1907.05388v2.

Fang Kong, Xiangcheng Zhang, Baoxiang Wang, and Shuai Li. Improved regret bounds for linear adversarial mdps via linear optimization. *Transactions on Machine Learning Research*, 2024.

T. Lancewicki and Y. Mansour. Near-optimal regret using policy optimization in online MDPs with aggregate bandit feedback. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

C.-W. Lee, H. Luo, C.-Y. Wei, and M. Zhang. Bias no more: High-probability data-dependent regret bounds for adversarial bandits and MDPs. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

H. Liu, C.-Y. Wei, and J. Zimmert. Towards optimal regret in adversarial linear MDPs with bandit feedback. In *International Conference on Learning Representations*, 2024a.

Haolin Liu, Artin Tajdini, Andrew Wagenmaker, and Chen-Yu Wei. Corruption-robust linear bandits: Minimax optimality and gap-dependent misspecification. *Advances in Neural Information Processing Systems*, 37:24277–24325, 2024b.

L. Lovász and S. Vempala. The geometry of logconcave functions and sampling algorithms. *Random Structures & Algorithms*, 30(3):307–358, 2007.

Haipeng Luo, Chen-Yu Wei, and Chung-Wei Lee. Policy optimization in adversarial mdps: Improved exploration via dilated bonuses. *Advances in Neural Information Processing Systems*, 34:22931–22942, 2021.

A. S. Nemirovski and M. J. Todd. Interior-point methods for optimization. *Acta Numerica*, pp. 191–234, 2008. doi: 10.1017/S0962492906370018.

Uri Sherman, Tomer Koren, and Yishay Mansour. Improved regret for efficient online reinforcement learning with linear function approximation. In *International Conference on Machine Learning*, pp. 31117–31150. PMLR, 2023.

Tim van Erven, Jack Mayo, Julia Olkhovskaya, and Chen-Yu Wei. An improved algorithm for adversarial linear contextual bandits via reduction. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Ruosong Wang, Simon S Du, Lin Yang, and Russ R Salakhutdinov. On reward-free reinforcement learning with linear function approximation. *Advances in neural information processing systems*, 33:17816–17826, 2020.

J. Zimmert and T. Lattimore. Return of the bias: Almost minimax optimal high probability bounds for adversarial linear bandits. In *Proceedings of the 35th Conference on Learning Theory*, 2022.

APPENDIX CONTENTS

A Compete Algorithm with Fallback **13**

B Omitted Proofs in Section 4 **14**

B.1 Properties of continuous exponential weights 14

B.2 Bounding bias and bonus 14

B.3 Overall regret analysis 16

C Omitted Proofs in Section 5 **19**

D Omitted Proofs in Section 6 **22**

D.1 Moment Estimates 22

D.2 Number of cuts 22

D.3 Implementation of Algorithm 3 23

A COMPETE ALGORITHM WITH FALLBACK

Algorithm 1 involves a repeat-until loop that may not stop, though this happens only with negligible probability (Lemma D.3). In order to ensure the algorithm will terminate with probability 1, we present below in Algorithm 4 a more complete version that caps the number of repeats by a finite amount. As shown in Lemma D.3, with high probability, the number of repeats is smaller than the threshold. In the regret analysis (Lemma 4.1), the bad event only contributes to negligible expected regret. In Algorithm 4, we highlight the parts that are different from Algorithm 1.

Algorithm 4 Algorithm 1 with a restart cap and a fallback policy

Require: $c_0 = \frac{1}{16e}$, $\zeta = K^{-1/2}$, $R = \sqrt{H} + \zeta$, $M = K$, $N_{\max} = \widetilde{\Theta}(p + \sqrt{HK^3})$, $\eta \le \frac{c_0}{128\sqrt{d^3H^4}}$.

$Z_0 \leftarrow 0$; $\mathcal{O} \leftarrow R\mathbb{B}_2^p$; $\mathcal{D}_{1,h} \leftarrow \varnothing$ for all $h \in [H]$
for $k = 1, \dots, K$ **do**
  $n \leftarrow 0$
  repeat
    $n \leftarrow n + 1$
    restart$\leftarrow$ FALSE
    Define $q_k(x) \propto \exp(-\eta Z_{k-1}^\top x)\mathbf{1}\{x \in \mathcal{O}\}$
    $(\widehat{\mu}_k, \widehat{\Sigma}_k) \leftarrow$ MOMENTESTIMATE$(\mathcal{O}, Z_{k-1})$
    Let $\{Y_{k,i}\}_{i=1}^{2p}$ be the set $\left\{\widehat{\mu}_k + c_0\widehat{\Sigma}_k^{1/2} v : v \in \{\pm e_1, \pm e_2, \dots, \pm e_p\}\right\}$.
    for $i = 1, \dots, 2p$ **do**
      reply $\leftarrow$ APPROXSEP$(Y_{k,i}, (\mathcal{D}_{k,h})_{h=1}^H, M, \zeta)$
      if reply is a cut (c, b) **then**
        $\mathcal{O} \leftarrow \mathcal{O} \cap \{x : c^\top x \ge b\}$
        restart$\leftarrow$ TRUE; **break**
      else
        $((\pi_{k,i,j})_{j=1}^M, \varepsilon_{k,i}) \leftarrow$ reply
  until *restart*$=$ FALSE or $n = N_{\max}$
  if $n = N_{\max}$ **then**
    Execute the policy π_k with $\pi_{k,h}(\cdot \mid s) = \mathrm{Unif}(\mathcal{A})$ for all h, s.
    $Z_k \leftarrow Z_{k-1}$ and $\mathcal{D}_{k+1,h} \leftarrow \mathcal{D}_{k,h}$ for all $h \in [H]$
  else
    Draw $(I_k, J_k) \sim \mathrm{Unif}([2p] \times [M])$ and execute $\pi_k = \pi_{k,I_k,J_k}$.
    Observe feature vectors along the trajectory $(\phi_{k,h})_{h=1}^H$ and the total loss L_k.
    $\widehat{\theta}_k \leftarrow \left(\zeta^2 \mathbf{I}_p + \frac{c_0^2}{p}\widehat{\Sigma}_k\right)^{-1}(Y_{k,I_k} - \widehat{\mu}_k)L_k$
    $Z_k \leftarrow Z_{k-1} + \widehat{\theta}_k - B_k$, where $B_k = \frac{10\eta p \varepsilon_k}{c_0} Z_{k-1}$ and $\varepsilon_k = \sqrt{\frac{1}{2p}\sum_{i=1}^{2p} \varepsilon_{k,i}^2}$.
    $\mathcal{D}_{k+1,h} \leftarrow \mathcal{D}_{k,h} \cup \{(\phi_{k,h}, S_{k,h+1})\}$ for all $h \in [H]$

B Omitted Proofs in Section 4

B.1 Properties of continuous exponential weights

For a full-dimensional compact convex body $\mathcal{O} \subset \mathbb{R}^p$, define

$$\Phi_{\mathcal{O}}(v) = \log \int_{\mathcal{O}} e^{v^\top x}\, dx, \qquad F_{\mathcal{O}} = \Phi_{\mathcal{O}}^*. \tag{13}$$

Here $F_{\mathcal{O}}(x) = \sup_v \{v^\top x - \Phi_{\mathcal{O}}(v)\}$ is the entropic barrier.

Lemma B.1 (Entropic-barrier toolkit)**.** *Let X_v have density proportional to $e^{v^\top x}\mathbf{1}\{x \in \mathcal{O}\}$. Then*

$$\nabla \Phi_{\mathcal{O}}(v) = \mathbb{E}\left[X_v\right], \qquad \nabla^2 \Phi_{\mathcal{O}}(v) = \mathrm{Cov}(X_v). \tag{14}$$

At every conjugate pair $x = \nabla \Phi_{\mathcal{O}}(v)$,

$$\nabla F_{\mathcal{O}}(x) = v, \qquad \nabla^2 F_{\mathcal{O}}(x) = \big(\nabla^2 \Phi_{\mathcal{O}}(v)\big)^{-1}. \tag{15}$$

Moreover, for every $x, u \in \mathrm{int}(\mathcal{O})$,

$$\|x - u\|_{\nabla^2 F_{\mathcal{O}}(x)} \le 4\,\langle x - u, \nabla F_{\mathcal{O}}(x)\rangle + 24p, \tag{16}$$

$$\|\nabla F_{\mathcal{O}}(x)\|_{(\nabla^2 F_{\mathcal{O}}(x))^{-1}} \le \sqrt{p}. \tag{17}$$

Finally, if $r = \|\delta v\|_{\nabla^2 \Phi_{\mathcal{O}}(v)} < 1$, then

$$\Phi_{\mathcal{O}}(v + \delta v) - \Phi_{\mathcal{O}}(v) - \langle \nabla \Phi_{\mathcal{O}}(v), \delta v\rangle \le -r - \log(1 - r). \tag{18}$$

Proof. Equation (14) follows by differentiation. Inverse-gradient duality gives (15) (Chewi, 2023, Section 2).

By Chewi (2023, Theorem 2), $F_{\mathcal{O}}$ is a p-self-concordant barrier, giving (17). By Zimmert & Lattimore (2022, Lemma 14), some $\gamma' \in [1, 4]$ satisfies

$$\|x - u\|_{\nabla^2 F_{\mathcal{O}}(x)} \le \gamma'\,\langle x - u, \nabla F_{\mathcal{O}}(x)\rangle + 4\gamma' p + 2\sqrt{p}.$$

For $p, \gamma' \ge 1$, $4\gamma' p + 2\sqrt{p} \le 6\gamma' p$. Thus

$$0 \le \|x - u\|_{\nabla^2 F_{\mathcal{O}}(x)} \le \gamma'\big(\langle x - u, \nabla F_{\mathcal{O}}(x)\rangle + 6p\big) \le 4\big(\langle x - u, \nabla F_{\mathcal{O}}(x)\rangle + 6p\big),$$

proving (16).

Since $\Phi_{\mathcal{O}} = F_{\mathcal{O}}^*$ and self-concordance is preserved under convex conjugation (Nemirovski & Todd, 2008, Section 2.1.3), $\Phi_{\mathcal{O}}$ is self-concordant. Its local Taylor bound (Nemirovski & Todd, 2008, Section 2.1.1, Eq. (2.5)) gives (18). □

B.2 Bounding bias and bonus

Lemma B.2 (Loss-estimator bias and local norm)**.** *Assume $\frac{1}{2}\Sigma_k \preceq \widehat{\Sigma}_k \preceq 2\Sigma_k$, and write $\widetilde{\theta}_k = \widehat{\theta}_k - B_k$. Then the loss estimator satisfies*

$$\|\theta_k - \mathbb{E}_k[\widehat{\theta}_k]\|_{\Sigma_k} \le \frac{5p\varepsilon_k}{2c_0}, \qquad \|\widehat{\theta}_k\|_{\Sigma_k}^2 \le \frac{2H^2p^2}{c_0^2}, \qquad \|\widetilde{\theta}_k\|_{\Sigma_k} \le \frac{\sqrt{2}Hp}{c_0} + \frac{10p^{3/2}\varepsilon_k}{c_0}. \tag{19}$$

Proof of Lemma B.2. Define $\alpha = c_0^2/p$. From the estimator definition,

$$\theta_k - \widehat{\theta}_k = (\zeta^2 \mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}\Big((\zeta^2 \mathbf{I}_p + \alpha\widehat{\Sigma}_k)\theta_k - (Y_{k,I_k} - \widehat{\mu}_k)L_k\Big).$$

Our construction gives

$$\frac{1}{2p}\sum_{i=1}^{2p}(Y_{k,i} - \widehat{\mu}_k) = 0, \qquad \frac{1}{2p}\sum_{i=1}^{2p}(Y_{k,i} - \widehat{\mu}_k)(Y_{k,i} - \widehat{\mu}_k)^\top = \alpha\widehat{\Sigma}_k.$$

Define

$$m_k := \frac{1}{2p}\sum_{i=1}^{2p}(Y_{k,i} - \widehat{\mu}_k)(x_{k,i} - Y_{k,i})^\top \theta_k.$$

Using $\mathbb{E}_k[L_k \mid I_k = i] = \langle x_{k,i}, \theta_k \rangle$, we obtain

$$\begin{aligned}\mathbb{E}_k[(Y_{k,I_k} - \widehat{\mu}_k)L_k] &= \frac{1}{2p}\sum_{i=1}^{2p}(Y_{k,i} - \widehat{\mu}_k)x_{k,i}^\top \theta_k \\ &= \alpha\widehat{\Sigma}_k\theta_k + m_k.\end{aligned}$$

Taking $\mathbb{E}_k$ in the first display therefore gives

$$\theta_k - \mathbb{E}_k[\widehat{\theta}_k] = (\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}(\zeta^2\theta_k - m_k). \tag{20}$$

For any $v \in \mathbb{R}^p$, Cauchy–Schwarz for the uniform average over $i \in [2p]$ gives

$$\begin{aligned}|\langle v, m_k\rangle| &= \left|\frac{1}{2p}\sum_{i=1}^{2p} v^\top(Y_{k,i} - \widehat{\mu}_k)(x_{k,i} - Y_{k,i})^\top\theta_k\right| \\ &\le \sqrt{\frac{1}{2p}\sum_{i=1}^{2p} v^\top(Y_{k,i} - \widehat{\mu}_k)(Y_{k,i} - \widehat{\mu}_k)^\top v}\sqrt{\frac{1}{2p}\sum_{i=1}^{2p}\theta_k^\top(x_{k,i} - Y_{k,i})(x_{k,i} - Y_{k,i})^\top\theta_k} \\ &\le \sqrt{\alpha}\|v\|_{\widehat{\Sigma}_k}\sqrt{\frac{p}{2p}\sum_{i=1}^{2p}\|x_{k,i} - Y_{k,i}\|_2^2} && (\|\theta_k\|_2^2 \le dH = p) \\ &= c_0\|v\|_{\widehat{\Sigma}_k}\varepsilon_k. && (\alpha = c_0^2/p)\end{aligned}$$

Taking $v = \widehat{\Sigma}_k^{-1}m_k$, we get $\|m_k\|_{\widehat{\Sigma}_k^{-1}}^2 \le c_0\|m_k\|_{\widehat{\Sigma}_k^{-1}}\varepsilon_k$ and hence

$$\|m_k\|_{\widehat{\Sigma}_k^{-1}} \le c_0\varepsilon_k.$$

Since $\Sigma_k \preceq 2\widehat{\Sigma}_k$, we have

$$\begin{aligned}\left\|(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}m_k\right\|_{\Sigma_k}^2 &\le 2\left\|(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}m_k\right\|_{\widehat{\Sigma}_k}^2 \\ &\le \frac{2}{\alpha^2}\|m_k\|_{\widehat{\Sigma}_k^{-1}}^2 \le \frac{2c_0^2\varepsilon_k^2}{\alpha^2} = \frac{2p^2\varepsilon_k^2}{c_0^2}.\end{aligned}$$

We also have

$$\begin{aligned}\left\|\zeta^2(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}\theta_k\right\|_{\Sigma_k}^2 &\le 2\zeta^4\theta_k^\top(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}\widehat{\Sigma}_k(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}\theta_k \\ &\le \frac{2}{\alpha}\zeta^4\theta_k^\top(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1/2}(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1/2}\theta_k \\ &\le \frac{2\zeta^2}{\alpha}\|\theta_k\|_2^2 = \frac{2\zeta^2p^2}{c_0^2}.\end{aligned}$$

Combining these two bounds in (20) and using $\varepsilon_k \ge 2\zeta$ yields

$$\|\theta_k - \mathbb{E}_k[\widehat{\theta}_k]\|_{\Sigma_k} \le \frac{\sqrt{2}p\varepsilon_k}{c_0} + \frac{\sqrt{2}p\zeta}{c_0} \le \frac{5p\varepsilon_k}{2c_0}.$$

Moreover, $0 \le L_k \le H$ and the leverage identity above imply

$$\begin{aligned}\|\widehat{\theta}_k\|_{\Sigma_k}^2 &\le 2H^2(Y_{k,I_k} - \widehat{\mu}_k)^\top(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}\widehat{\Sigma}_k(\zeta^2\mathbf{I}_p + \alpha\widehat{\Sigma}_k)^{-1}(Y_{k,I_k} - \widehat{\mu}_k) \\ &\le \frac{2H^2}{\alpha}(Y_{k,I_k} - \widehat{\mu}_k)^\top(\alpha\widehat{\Sigma}_k)^{-1}(Y_{k,I_k} - \widehat{\mu}_k) \le \frac{2H^2p^2}{c_0^2}.\end{aligned}$$

This proves the first two bounds in (19).

By (14) and (15),

$$\nabla F_{\mathcal{O}_k}(\mu_k) = -\eta Z_{k-1}, \qquad \nabla^2 F_{\mathcal{O}_k}(\mu_k) = \Sigma_k^{-1},$$

and hence $B_k = -10p\varepsilon_k \nabla F_{\mathcal{O}_k}(\mu_k)/c_0$. Equation (17) gives

$$\|B_k\|_{\Sigma_k} \le \frac{10p^{3/2}\varepsilon_k}{c_0}.$$

The triangle inequality and $(a+b)^2 \le 2a^2 + 2b^2$, together with the bound on $\|\widehat{\theta}_k\|_{\Sigma_k}$ above, prove the last bound in (19). □

Lemma B.3 (Bias–bonus compensation)**.** *Under the assumptions of Lemma B.2, for every $u \in \mathcal{O}_k$, it holds that*

$$\langle \mu_k - u, \theta_k \rangle \le \mathbb{E}_k[\langle \mu_k - u, \widetilde{\theta}_k \rangle] + \frac{60p^2\varepsilon_k}{c_0}. \tag{21}$$

Proof of Lemma B.3. By (14) and (15),

$$\nabla F_{\mathcal{O}_k}(\mu_k) = -\eta Z_{k-1}, \qquad \nabla^2 F_{\mathcal{O}_k}(\mu_k) = \Sigma_k^{-1},$$

and hence $B_k = -10p\varepsilon_k \nabla F_{\mathcal{O}_k}(\mu_k)/c_0$. Local Cauchy–Schwarz, Eq. (19), and Eq. (16) give

$$\begin{aligned}
\langle \mu_k - u, \theta_k - \mathbb{E}_k[\widehat{\theta}_k] \rangle &\le \frac{5p\varepsilon_k}{2c_0} \|\mu_k - u\|_{\nabla^2 F_{\mathcal{O}_k}(\mu_k)} \\
&\le \frac{10p\varepsilon_k}{c_0} \langle \mu_k - u, \nabla F_{\mathcal{O}_k}(\mu_k) \rangle + \frac{60p^2\varepsilon_k}{c_0} \\
&= -\langle \mu_k - u, B_k \rangle + \frac{60p^2\varepsilon_k}{c_0}.
\end{aligned}$$

Since B_k is $\mathcal{F}_k$-measurable,

$$\langle \mu_k - u, \theta_k \rangle \le \mathbb{E}_k[\langle \mu_k - u, \widehat{\theta}_k - B_k \rangle] + \frac{60p^2\varepsilon_k}{c_0},$$

which proves (21). □

B.3 Overall regret analysis

Let $\mathcal{F}_k$ be the information available after preparation of episode k and before the policy draw, and write $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_k]$. Define χ_k to equal one if all Bellman-confidence and moment-estimation events have held through this preparation. Once it becomes zero, it remains zero. Thus χ_k is $\mathcal{F}_k$-measurable and, by Theorem C.2, the moment-estimation guarantee, and a union bound over all calls,

$$\mathbb{P}(\chi_k = 1 \text{ for every } k \in [K]) \ge 1 - \delta. \tag{22}$$

where $\delta = 2K^{-2}$. Whenever $\chi_k = 1$, $\mathcal{K}_\zeta \subseteq \mathcal{O}_k \subseteq R\mathbb{B}_2^p$ and the moment and compiler guarantees hold. Write $\overline{x}_k = \mathbb{E}_k[\phi^{\pi_k}]$ and set the auxiliary quantities to zero when $\chi_k = 0$.

Lemma B.4 (Cumulative estimated loss on nested regions)**.** *Let $\mathcal{O}_1 \supseteq \cdots \supseteq \mathcal{O}_K$ satisfy $\mathcal{K}_\zeta \subseteq \mathcal{O}_k \subseteq R\mathbb{B}_2^p$. Let (μ_k, Σ_k) be the exact mean and covariance of the density proportional to $e^{-\eta Z_{k-1}^\top x}$ on $\mathcal{O}_k$, where $Z_0 = 0$ and $Z_k = Z_{k-1} + \widetilde{\theta}_k$. If $\eta\|\widetilde{\theta}_k\|_{\Sigma_k} \le 1/2$ for every k, then for each $u \in \mathcal{K}$,*

$$\sum_{k=1}^K \langle \mu_k - u, \widetilde{\theta}_k \rangle \le \frac{p \log(R/\zeta)}{\eta} + \eta \sum_{k=1}^K \|\widetilde{\theta}_k\|_{\Sigma_k}^2.$$

Proof. Apply Eq. (18) at $v = -\eta Z_{k-1}$ with increment $-\eta\widetilde{\theta}_k$. Since $r = \eta\|\widetilde{\theta}_k\|_{\Sigma_k} \le 1/2$ and $-r - \log(1-r) \le r^2$,

$$\Phi_{\mathcal{O}_k}(-\eta Z_k) - \Phi_{\mathcal{O}_k}(-\eta Z_{k-1}) \le -\eta\langle \mu_k, \widetilde{\theta}_k \rangle + \eta^2 \|\widetilde{\theta}_k\|_{\Sigma_k}^2.$$

Since $\mathcal{O}_{k+1} \subseteq \mathcal{O}_k$, by Eq. (13) we have $\Phi_{\mathcal{O}_{k+1}} \leq \Phi_{\mathcal{O}_k}$, and thus

$$\begin{aligned}\Phi_{\mathcal{O}_K}(-\eta Z_K) - \Phi_{\mathcal{O}_1}(0) &= \sum_{k=1}^{K}[\Phi_{\mathcal{O}_k}(-\eta Z_k) - \Phi_{\mathcal{O}_k}(-\eta Z_{k-1})] \\ &\quad + \sum_{k=1}^{K-1}[\Phi_{\mathcal{O}_{k+1}}(-\eta Z_k) - \Phi_{\mathcal{O}_k}(-\eta Z_k)] \\ &\leq -\eta \sum_{k=1}^{K}\langle \mu_k, \widetilde{\theta}_k\rangle + \eta^2 \sum_{k=1}^{K} \|\widetilde{\theta}_k\|_{\Sigma_k}^2.\end{aligned}$$

Finally, $\Phi_{\mathcal{O}_1}(0) \leq \log \mathrm{Vol}(R\mathbb{B}_2^p)$ and Jensen on $u + \zeta\mathbb{B}_2^p \subseteq \mathcal{O}_K$ give

$$\int_{\mathcal{O}_K} e^{-\eta Z_K^\top x}\, dx \geq e^{-\eta Z_K^\top u}\, \mathrm{Vol}(\zeta\mathbb{B}_2^p).$$

Using $Z_K = \sum_k \widetilde{\theta}_k$ yields

$$\sum_{k=1}^{K}\langle \mu_k - u, \widetilde{\theta}_k\rangle \leq \frac{p\log(R/\zeta)}{\eta} + \eta \sum_{k=1}^{K} \|\widetilde{\theta}_k\|_{\Sigma_k}^2. \tag{23}$$

□

Proof of Lemma 4.1. Define $\mathcal{E}_K := \mathbb{E}\left[\sum_{k=1}^{K} \chi_k \varepsilon_k^2\right]$. On the valid prefix, Lemma B.3 gives

$$\langle \mu_k - u, \theta_k\rangle \leq \mathbb{E}_k\left[\langle \mu_k - u, \widetilde{\theta}_k\rangle\right] + \frac{60p^2\varepsilon_k}{c_0}. \tag{24}$$

Furthermore,

$$\overline{x}_k - \mu_k = (\widehat{\mu}_k - \mu_k) + \frac{1}{2p}\sum_i (x_{k,i} - Y_{k,i}). \tag{25}$$

Since $p = dH$, the RMS approximation and moment bounds imply

$$\langle \overline{x}_k - u, \theta_k\rangle \leq \mathbb{E}_k\left[\langle \mu_k - u, \widetilde{\theta}_k\rangle\right] + e_k, \qquad e_k := \left(\frac{60p^2}{c_0} + \sqrt{p}\right)\varepsilon_k + 2R\sqrt{p}\,\delta_\mu. \tag{26}$$

Write $r_k = \langle \phi^{\pi_k} - u, \theta_k\rangle$. Since χ_k is $\mathcal{F}_k$-measurable,

$$\mathbb{E}\left[\sum_{k=1}^{K}\chi_k r_k\right] = \mathbb{E}\left[\sum_{k=1}^{K}\chi_k\langle \overline{x}_k - u, \theta_k\rangle\right]. \tag{27}$$

Multiplying Eq. (26) by χ_k and taking expectations gives

$$\mathbb{E}\left[\sum_{k=1}^{K}\chi_k r_k\right] \leq \mathbb{E}\left[\sum_{k=1}^{K}\chi_k\langle \mu_k - u, \widetilde{\theta}_k\rangle\right] + \mathbb{E}\left[\sum_{k=1}^{K}\chi_k e_k\right]. \tag{28}$$

Here we used the tower property, since $\chi_k(\mu_k - u)$ is $\mathcal{F}_k$-measurable.

By Lemma B.2,

$$\left\|\widetilde{\theta}_k\right\|_{\Sigma_k}^2 \leq \frac{4H^2p^2}{c_0^2} + \frac{200p^3\varepsilon_k^2}{c_0^2}.$$

On the valid prefix, $\varepsilon_k \leq 6\sqrt{H}$. Provided $\eta \leq c_0/(128p^{3/2}\sqrt{H})$, Eq. (19) gives

$$\eta\left\|\widetilde{\theta}_k\right\|_{\Sigma_k} \leq \eta\left(\frac{\sqrt{2}Hp}{c_0} + \frac{10p^{3/2}\varepsilon_k}{c_0}\right) \leq \frac{60 + \sqrt{2/d}}{128} \leq \frac{60 + \sqrt{2}}{128} < \frac{1}{2},$$

which verifies the condition of the nested-region potential bound. On every sample path, the indices with $\chi_k = 1$ form a prefix. Applying Lemma B.4 to this realized prefix and then taking expectations yields

$$\mathbb{E}\left[\sum_{k=1}^{K} \chi_k \langle \mu_k - u, \widetilde{\theta}_k \rangle\right] \le \frac{p \log(R/\zeta)}{\eta} + \frac{4\eta H^2 p^2 K}{c_0^2} + \frac{200\eta p^3}{c_0^2}\mathcal{E}_K.$$

Furthermore,

$$\mathbb{E}\left[\sum_{k} \chi_k \varepsilon_k\right] \le \sqrt{K\mathcal{E}_K}.$$

Substitution into Eq. (28), followed by $p = dH$, gives

$$\mathbb{E}\left[\sum_{k=1}^{K} \chi_k r_k\right] \le \widetilde{O}\Big(\frac{dH}{\eta} + \eta d^2 H^4 K + \eta d^3 H^3 \mathcal{E}_K + d^2 H^2 \sqrt{K\mathcal{E}_K} + R\sqrt{p}\,\delta_\mu K\Big).$$

Finally, $r_k \le H$, and by Eq. (22),

$$\begin{aligned}\mathbb{E}\left[\sum_{k=1}^{K} r_k\right] &= \mathbb{E}\left[\sum_{k=1}^{K} \chi_k r_k\right] + \mathbb{E}\left[\sum_{k=1}^{K} (1-\chi_k) r_k\right] \\ &\le \mathbb{E}\left[\sum_{k=1}^{K} \chi_k r_k\right] + HK\delta.\end{aligned}$$

Using $\delta_\mu = 2c_0/\sqrt{K}$, $\delta = 2/(K+1)^2$, $R = \sqrt{H} + K^{-1/2}$, and $p = dH$, we have

$$R\sqrt{p}\,\delta_\mu K + HK\delta = O\Big(R\sqrt{pK} + \frac{H}{K}\Big) = O\Big(H\sqrt{dK}\Big).$$

The unconstrained minimizer of the two $\mathcal{E}_K$-independent η-terms is $(dH^3K)^{-1/2}$. To also ensure the local-norm condition above, choose

$$\eta = \frac{c_0}{128}\min\Big\{(dH^3K)^{-1/2}, d^{-3/2}H^{-2}\Big\}.$$

The second branch is exactly the required stability cap. Moreover, the pathwise bound $\mathcal{E}_K \le 36HK$ gives

$$\eta d^3 H^3 \mathcal{E}_K \le O\Big(d^{3/2} H \mathcal{E}_K\Big) \le O\Big(d^2 H^2 \sqrt{K\mathcal{E}_K}\Big).$$

Thus the $\eta\mathcal{E}_K$ term is absorbed by the $d^2H^2\sqrt{K\mathcal{E}_K}$ term. In either branch, $H\sqrt{dK} = O(\eta d^2 H^4 K)$. If $K \ge d^2 H$, the first branch applies and

$$\frac{dH}{\eta} + \eta d^2 H^4 K = O\Big(d^{3/2} H^{5/2} \sqrt{K}\Big).$$

If $K < d^2 H$, the trivial bound $\mathrm{Reg}_K \le HK$ is no larger than the same expression. Therefore

$$\mathrm{Reg}_K \le \widetilde{O}\Big(d^{3/2} H^{5/2}\sqrt{K} + d^2 H^2 \sqrt{K\mathcal{E}_K}\Big).$$

This proves Lemma 4.1. □

C Omitted Proofs in Section 5

Lemma C.1 (Averaging recursion). *Let $a_0, a_1, \dots, a_M$ and $b_1, \dots, b_M$ be real sequences, and let $c \geq 0$. Suppose that, for every $j \in [M]$,*

$$a_j \leq \left(1 - \frac{1}{j}\right) a_{j-1} + \frac{b_j}{j} + \frac{c}{j^2}.$$

Then, for every $m \in [M]$,

$$a_m \leq \frac{1}{m} \sum_{j=1}^{m} b_j + \frac{c(1 + \log m)}{m}. \tag{29}$$

Proof. Multiplying the recursion by j gives

$$j a_j \leq (j-1) a_{j-1} + b_j + \frac{c}{j}.$$

Summing from $j = 1$ to m telescopes the terms involving a_j and yields

$$m a_m \leq \sum_{j=1}^{m} b_j + c \sum_{j=1}^{m} \frac{1}{j},$$

which proves Eq. (29) because $\sum_{j=1}^{m} j^{-1} \leq 1 + \log m$. □

Theorem C.2 (Uniform Bellman confidence (Lemma D.4 of Jin et al. (2020))). *Fix $\delta \in (0,1)$ and $\epsilon \in (0,1)$. For each $h \in [H]$, let $C \geq 1$ and let $\mathcal{F}$ be a deterministic class of scalar functions from $\mathcal{S}_{h+1}$ to $[-C, C]$. Define the relative covering number*

$$N_{\mathcal{F}}(\epsilon) := \mathcal{N}_\infty(C\epsilon, \mathcal{F}),$$

where $\mathcal{N}_\infty$ denotes the sup-norm covering number. Assume $N_{\mathcal{F}}(\epsilon) < \infty$. There is a universal constant $c_1 > 0$ such that, if

$$\beta \geq c_1 \left(\sqrt{d \log K + \log\left(\frac{H N_{\mathcal{F}}(\epsilon)}{\delta}\right)} + K\epsilon + \sqrt{d} \right), \tag{30}$$

then, with probability at least $1 - \delta$, simultaneously for every $k \in [K]$, $h \in [H]$, $f \in \mathcal{F}_{h+1}$, and $(s,a) \in \mathcal{S}_h \times \mathcal{A}$,

$$|(\widehat{P}_{k,h} - P_h) f(s,a)| \leq \beta C \left\| \phi_h(s,a) \right\|_{\Lambda_{k,h}^{-1}}. \tag{31}$$

Proof of Theorem 5.3. All three tasks in Algorithm 3 are based on least-squares value iteration. The key is to show (31) for all three tasks, each corresponding to a different function class $\mathcal{F}$. Fix a layer h, write $\alpha = (\alpha_1, \alpha_2, \alpha_3)$, and define

$$\begin{aligned}
\nu_h(s,a;\Lambda) &= \sqrt{\phi_h(s,a)^\top \Lambda^{-1} \phi_h(s,a)} \\
q_h(s,a;\Lambda,\vartheta,\alpha) &= \mathrm{CLIP}_{[\alpha_1,\alpha_2]}\big(\phi_h(s,a)^\top \vartheta + \alpha_3 \nu_h(s,a;\Lambda)\big), \\
\pi_h(a \mid s;\Lambda,\widetilde{\vartheta}) &\propto \exp\Big(-q_h(s,a;\Lambda,\widetilde{\vartheta},\alpha^{\mathrm{pl}})/\tau\Big), \\
v_h(s;\Lambda,\vartheta,\widetilde{\vartheta},\alpha) &= \sum_a \pi_h(a \mid s;\Lambda,\widetilde{\vartheta}) q_h(s,a;\Lambda,\vartheta,\alpha),
\end{aligned}$$

where

$$\alpha^{\mathrm{pl}} = (-\sqrt{H}, \sqrt{H}, -(\sqrt{H}+1)\beta), \qquad \alpha^{\mathrm{ft}} = (-1, 1, 0), \qquad \alpha^{\mathrm{unc}} = (0, H-h+1, 1+H\beta).$$

The function classes for the three tasks are respectively

$$s \mapsto -\tau \log \sum_a \exp\big(-q_h(s,a;\Lambda,\vartheta,\alpha^{\mathrm{pl}})/\tau\big) \qquad \text{(planning)}$$

$$s \mapsto v_h(s;\Lambda,\vartheta,\widetilde{\vartheta},\alpha^{\mathrm{ft}}) \qquad \text{(feature evaluation)}$$

$$s \mapsto v_h(s;\Lambda,\vartheta,\widetilde{\vartheta},\alpha^{\mathrm{unc}}). \qquad \text{(uncertainty evaluation)}$$

Denote these classes by $\mathcal{F}_h^{\mathrm{pl}}$, $\mathcal{F}_h^{\mathrm{ft}}$, and $\mathcal{F}_h^{\mathrm{unc}}$, respectively, and set their terminal classes to $\{0\}$, with terminal scale one.

Bounding the covering number of each classes. Clipping and $\tau \log A \le 1$ give the uniform bounds

$$\begin{aligned} f \in \mathcal{F}_h^{\mathrm{pl}} : &\ \|f\|_\infty \le \sqrt{H} + 1, \\ f \in \mathcal{F}_h^{\mathrm{ft}} : &\ \|f\|_\infty \le 1, \\ f \in \mathcal{F}_h^{\mathrm{unc}} : &\ \|f\|_\infty \le H - h + 1. \end{aligned} \tag{32}$$

Write $C_h^{\mathrm{pl}} = \sqrt{H} + 1$, $C_h^{\mathrm{ft}} = 1$, and $C_h^{\mathrm{unc}} = H - h + 1$ for these scales. Therefore,

$$\left\| \Lambda_{k,h}^{-1} \sum_{(u,s') \in \mathcal{D}_{k,h}} u f(s') \right\|_2 \le \sum_{(u,s') \in \mathcal{D}_{k,h}} \|u\|_2 \, |f(s')| \le 3KH.$$

Including the immediate reward/cost gives $\|\vartheta\|_2, \|\widetilde{\vartheta}\|_2 \le 3KH + 1$. Also, $\mathbf{I}_d \preceq \Lambda_{k,h} \preceq (K+1)\mathbf{I}_d$. Thus, each class has at most

$$n = \frac{d(d+1)}{2} + 2d \le 3d^2$$

scalar parameters, and each of them is bounded in absolute value by $\overline{R} = \max\{K + 1, 3KH + 1\}$. Moreover,

$$|\nu_h(s, a; \Lambda) - \nu_h(s, a; \Lambda')| \le \frac{1}{2} \|\Lambda - \Lambda'\|_{\mathrm{F}},$$

so clipping gives, for each fixed α,

$$\|q_h(\cdot; \Lambda, \vartheta, \alpha) - q_h(\cdot; \Lambda', \vartheta', \alpha)\|_\infty \le \|\vartheta - \vartheta'\|_2 + \frac{|\alpha_3|}{2} \|\Lambda - \Lambda'\|_{\mathrm{F}}.$$

The softmax map satisfies

$$\sup_s \|\pi_h(\cdot \mid s; q) - \pi_h(\cdot \mid s; q')\|_1 \le \frac{2}{\tau} \|q - q'\|_\infty.$$

Hence, if C denotes the corresponding scale in (32),

$$\|v_h - v_h'\|_\infty \le \|q_h - q_h'\|_\infty + C \sup_s \|\pi_h(\cdot \mid s) - \pi_h'(\cdot \mid s)\|_1.$$

The planning soft minimum is nonexpansive in sup norm. Since $C \ge 1$, $|\alpha_3|/C \le 1 + H\beta$, and $(\sqrt{H} + 1)\beta \le 2H\beta$, the evaluation is L-Lipschitz with respective to its parameters, with

$$L = \frac{8(1 + H\beta)}{\tau}.$$

A parameter grid of mesh $\epsilon/(nL)$ gives a $C\epsilon$-cover. Therefore,

$$\log \mathcal{N}_\infty\big(C\epsilon, \mathcal{F}\big) = O\left(n \log\left(1 + \frac{\overline{R} n L}{\epsilon} \right) \right). \tag{33}$$

Bounding error. Take

$$\delta = K^{-2}, \qquad \epsilon_{\mathrm{net}} = K^{-2}, \qquad \beta = C_\beta d \sqrt{\log(KdHA)}.$$

Since $\tau^{-1} = H\sqrt{K} \log(eA)$, (33) gives

$$\log N_{\mathcal{F}^t}(\epsilon_{\mathrm{net}}) = O\big(d^2 \log(KdHA)\big).$$

Thus a sufficiently large universal C_β makes (30) hold, and $\beta = \widetilde{O}(d)$. Applying Theorem C.2 directly to the three classes, with failure probability $\delta/3$ each, and taking a union bound yields

$$|(\widehat{P}_{k,h} - P_h) V_{h+1}^{\mathrm{pl}}(s, a)| \le (\sqrt{H} + 1)\beta \nu_{k,h}(s, a), \tag{34}$$

$$|(\widehat{P}_{k,h} - P_h) V_{h+1}^{\mathrm{ft},m}(s, a)| \le \beta \nu_{k,h}(s, a), \qquad m \in [p], \tag{35}$$

$$|(\widehat{P}_{k,h} - P_h) \widehat{U}_{h+1}(s, a)| \le H\beta \nu_{k,h}(s, a). \tag{36}$$

Fix one call and let π be its returned policy. Write

$$U_k^\pi = \mathbb{E}_{P,\pi}\left[\sum_{h=1}^{H} \nu_{k,h}(S_h, A_h)\right].$$

By backward induction (Jin et al., 2020) on the update in Line 4 with confidence bound (34), and that $V_h^{\mathrm{pl}}(s) \le \min_a Q_h^{\mathrm{pl}}(s,a)$, we have $V_h^{\mathrm{pl}} \le \min_\pi V_h^\pi$. Therefore,

$$V_1^{\mathrm{pl}}(s_1) \le \min_{u\in\mathcal{K}} \bar{c}^\top u. \tag{37}$$

Likewise, backward induction on the update in Line 12 with confidence bound (36) gives

$$U_k^\pi \le \widehat{U}_1(s_1). \tag{38}$$

Similarly, backward induction on the update in Line 10 with coordinatewise confidence bounds (35) gives

$$\|\widehat{\phi} - \phi^\pi\|_2 \le \sqrt{p}\,\beta U_k^\pi \le \sqrt{p}\beta\widehat{U}_1(s_1). \tag{39}$$

The feature-estimation error $\|\widehat{\phi} - \phi^\pi\|_2$ is at most $2\sqrt{H}$ because $\widehat{\phi}, \phi^\pi \in \sqrt{H}\mathbb{B}_2^p$. Thus,

$$\|\widehat{\phi} - \phi^\pi\|_2 \le \min\left\{5\sqrt{p}\,\beta\widehat{U}_1(s_1) + H\tau\log A,\ 2\sqrt{H}\right\} = \Delta$$

proving the first bound in (9).

Comparing the planning recursion with the true value of the returned Gibbs policy, each layer contributes at most $2(\sqrt{H}+1)\beta\nu_{k,h}$ from transition error and at most $\tau\log A$ from the soft minimum. Hence

$$\bar{c}^\top\phi^\pi - V_1^{\mathrm{pl}}(s_1) \le 2(\sqrt{H}+1)\beta U_k^\pi + H\tau\log A. \tag{40}$$

The same uncertainty recursion also gives

$$\widehat{U}_1(s_1) \le 2\beta H U_k^\pi. \tag{41}$$

Combining (39), (37), (40) and using $p = dH$ yields

$$\bar{c}^\top\widehat{\phi} - \min_{u\in\mathcal{K}} \bar{c}^\top u \le 5\sqrt{p}\,\beta U_k^\pi + H\tau\log A \le 5\sqrt{p}\,\beta\widehat{U}_1(s_1) + H\tau\log A. \tag{42}$$

The normalized optimization gap $\bar{c}^\top\widehat{\phi} - \min_{u\in\mathcal{K}}\bar{c}^\top u$ is at most $2\sqrt{H}$ because $\widehat{\phi} \in \sqrt{H}\mathbb{B}_2^p$, $\mathcal{K} \subseteq \sqrt{H}\mathbb{B}_2^p$, and $\|\bar{c}\|_2 \le 1$. Thus,

$$\bar{c}^\top\widehat{\phi} - \min_{u\in\mathcal{K}} \bar{c}^\top u \le \min\left\{5\sqrt{p}\,\beta\widehat{U}_1(s_1) + H\tau\log A,\ 2\sqrt{H}\right\} = \Delta$$

proving the second bound in (9).

Finally, (41) gives

$$\Delta \le 10\sqrt{p}\,\beta^2 H U_k^\pi + H\tau\log A,$$

proving the final bound. □

D Omitted Proofs in Section 6

D.1 Moment Estimates

We let MomentEstimate to be a procedure described in Ito et al. (2019), which takes a convex function $f : \mathbb{R}^p \to \mathbb{R}$ and a convex body $\mathcal{P} \subset \mathbb{R}^p$ as input, and output estimators of the mean and covariance of the distribution $q(x) \propto \exp(-f(x))\mathbf{1}\{x \in \mathcal{P}\}$ within polynomial time. In our case $f(x) = \eta x^\top Z_{t-1}$ and $\mathcal{P} = \mathcal{O}$. Its guarantee is the following:

Lemma D.1 (Corollary 1 of Ito et al. (2019))**.** *Let $\mathcal{P} \subset \mathbb{R}^p$ be a convex body with non-zero Lubesgue measure, and let $q(x) \propto \exp(-f(x))\mathbf{1}\{x \in \mathcal{P}\}$ be a log-concave distribution with some convex function $f : \mathcal{P} \to \mathbb{R}$. Let μ and Σ denote the mean and covariance of q. Then there is an algorithm that outputs a vector $\widehat{\mu} \in \mathbb{R}^p$ and a symmetric matrix $\widehat{\Sigma} \in \mathbb{R}^{p\times p}$ satisfying $\frac{1}{2}\Sigma \preceq \widehat{\Sigma} \preceq 2\Sigma$ and $\|\widehat{\mu} - \mu\|_{\mathrm{Cov}(q)^{-1}} \le \varepsilon$ with probability at least $1 - \delta$. The computational time, the number of calls for the membershpi oracle of $\mathcal{P}$, and the value oracle of f are bounded by* $\mathrm{poly}(p, \frac{1}{\varepsilon}, \log\frac{1}{\delta})$.

We let $\varepsilon = \delta_\mu = 2c_0/\sqrt{K}$, which implies the estimation accuracy

$$\|\widehat{\mu}_k - \mu_k\|_{\Sigma_k^{-1}} \le \delta_\mu, \qquad \frac{1}{2}\Sigma_k \preceq \widehat{\Sigma}_k \preceq 2\Sigma_k$$

after calling MomentEstimate in Line 6 of Algorithm 1.

D.2 Number of cuts

Lemma D.2 (Containment of the estimated covariance ellipsoid)**.** *Let q be a full-dimensional log-concave distribution supported on a convex body $\mathcal{O} \subseteq \mathbb{R}^p$, with mean μ and covariance Σ. Suppose $\widehat{\mu}$ and $\widehat{\Sigma}$ satisfy $\frac{1}{2}\Sigma \preceq \widehat{\Sigma} \preceq 2\Sigma$ and $\|\widehat{\mu} - \mu\|_{\Sigma^{-1}} \le \frac{1}{12e}$. Then*

$$\widehat{\mu} + \frac{1}{16e}\widehat{\Sigma}^{1/2}\mathbb{B}_2^p \subseteq \mu + \frac{1}{2e}\Sigma^{1/2}\mathbb{B}_2^p \subseteq \mathcal{O}. \tag{43}$$

Proof. We first show that

$$\mu + \frac{1}{e}\Sigma^{1/2}\mathbb{B}_2^p \subseteq \mathcal{O}. \tag{44}$$

Otherwise, separation gives a nonzero vector a and a point y with $\|y - \mu\|_{\Sigma^{-1}} \le 1/e$ such that $a^\top y < \inf_{x\in\mathcal{O}} a^\top x$. For $X \sim q$, denote the random scalar

$$S = \frac{a^\top(X - \mu)}{\sqrt{a^\top \Sigma a}}$$

and denote the density function of S as q_S. Then S is centered, variance-one, and log-concave. Moreover, its support is contained in $[u_{\min}, \infty)$: Indeed, strict separation and compactness of $\mathcal{O}$ give

$$u_{\min} := \frac{\inf_{x\in\mathcal{O}} a^\top(x - \mu)}{\sqrt{a^\top \Sigma a}} > \frac{a^\top(y - \mu)}{\sqrt{a^\top \Sigma a}} \ge -\frac{1}{e},$$

where the last inequality follows from Cauchy–Schwarz in the Σ-norm. Lemmas 5.4 and 5.5(a) of Lovász & Vempala (2007) give $\mathbb{P}(S < 0) \ge 1/e$ and $q_S(s) \le 1$ for any s. However, the support condition would instead imply

$$\mathbb{P}(S < 0) = \int_{u_{\min}}^{0} q_S(t)\,dt \le |u_{\min}| < \frac{1}{e},$$

a contradiction. This proves Eq. (44).

Now fix $v \in \mathbb{B}_2^p$. The covariance assumption gives $\|\Sigma^{-1/2}\widehat{\Sigma}^{1/2}\|_{\mathrm{op}} \le \sqrt{2}$. Therefore

$$\begin{aligned}\left\|\widehat{\mu} + \frac{1}{16e}\widehat{\Sigma}^{1/2}v - \mu\right\|_{\Sigma^{-1}} &\le \|\widehat{\mu} - \mu\|_{\Sigma^{-1}} + \frac{1}{16e}\left\|\Sigma^{-1/2}\widehat{\Sigma}^{1/2}v\right\|_2 \\ &\le \frac{1}{12e} + \frac{\sqrt{2}}{16e} < \frac{1}{2e}.\end{aligned}$$

The first inclusion in Eq. (43) follows, and the second follows from Eq. (44). □

Lemma D.3 (Number of cuts)**.** *Suppose every invoked moment call, with exact mean μ and covariance Σ, returns estimates satisfying $\frac{1}{2}\Sigma \preceq \widehat{\Sigma} \preceq 2\Sigma$ and $\|\widehat{\mu} - \mu\|_{\Sigma^{-1}} \le 1/(12e)$. Then the number of cuts is upper bounded by $\widetilde{O}(pK + \sqrt{H}K^{5/2})$.*

Proof. By the definition of $\widehat{\theta}_k$, we have $\|\widehat{\theta}_k\|_2 \le \frac{2RH}{\zeta^2}$. Since the update is $Z_k = (1 - \frac{10\eta p \varepsilon_k}{c_0})Z_{k-1} + \widehat{\theta}_k$, we have $\|Z_k\| \le \|Z_{k-1}\| + \|\widehat{\theta}_k\|$. By induction, it holds that $\|Z_k\| \le \frac{2RHK}{\zeta^2}$ for all k.

By Lemma D.2, every target used during preparation belongs to the current region $\mathcal{O}$ and satisfies $\|Y_i - \mu\|_{\Sigma^{-1}} < 1/(2e)$. The local-cut argument of Ito et al. (2019, Lemma 5) shows that each cut removes at least a $\frac{1}{2e}$ fraction of the current mass. Let N_k be the number of new cuts added in episode k, and let $\mathcal{O}_i, \mathcal{O}_f$ be the set $\mathcal{O}$ at the start and the end of episode k. Then we have

$$\begin{aligned}
\mathrm{Vol}(\zeta\mathbb{B}_2^p)e^{-\eta R\|Z_{k-1}\|} &\le \int_{\mathcal{O}_f} e^{-\eta x^\top Z_{k-1}}dx \\
&\le \left(1 - \frac{1}{2e}\right)^{N_k} \int_{\mathcal{O}_i} e^{-\eta x^\top Z_{k-1}}dx \\
&\le \left(1 - \frac{1}{2e}\right)^{N_k} \mathrm{Vol}(R\mathbb{B}_2^p)e^{\eta R\|Z_{k-1}\|}.
\end{aligned}$$

Taking logarithms gives

$$N_k \le O\left(p\log(R/\zeta) + \eta R\|Z_{k-1}\|\right) \le O\left(p\log(R/\zeta) + \frac{\eta R^2 KH}{\zeta^2}\right) \le \widetilde{O}\big(p + \sqrt{H}K^{3/2}\big).$$

Summing over episodes gives the desired bound. □

D.3 Implementation of Algorithm 3

Although Algorithm 3 defines value functions on the entire state space, it can be executed without enumerating states or state–action pairs. Below is an efficient way to implement Algorithm 3. We just use the computation of V_h^{pl} as example; the other two tasks follow the same manner.

For any function $f : \mathcal{S}_{h+1} \to \mathbb{R}$, define:

$$\widehat{w}_h(f) := \Lambda_{k,h}^{-1} \sum_j u_j f(s'_j)^\top \in \mathbb{R}^{d\times m}.$$

Then, for every queried pair (s, a),

$$\widehat{P}_{k,h} f(s,a) = \phi_h(s,a)^\top \widehat{w}_h(f).$$

Below, we show that we only need to keep track of the vector $\widehat{w}_h(V_{h+1}^{\mathrm{pl}})$, or equivalently $\vartheta_h = \bar{c}_h + \widehat{w}_h(V_{h+1}^{\mathrm{pl}})$ during the execution of Algorithm 3. They are computed by backward induction.

Suppose ϑ_h has already been computed. For every state s'_j appearing in $\mathcal{D}_{k,h-1}$, query $\phi_h(s'_j, a)$ for each $a \in \mathcal{A}$ and compute

$$\begin{aligned}
Q_h^{\mathrm{pl}}(s'_j, a) &= \mathrm{CLIP}_{[-\sqrt{H},\sqrt{H}]}\left(\phi_h(s'_j,a)^\top \vartheta_h - (\sqrt{H}+1)\beta \left\|\phi_h(s'_j,a)\right\|_{\Lambda_{k,h}^{-1}}\right), \\
V_h^{\mathrm{pl}}(s'_j) &= -\tau \log \sum_{a\in\mathcal{A}} \exp\Big(-Q_h^{\mathrm{pl}}(s'_j,a)/\tau\Big).
\end{aligned}$$

These are the values needed to form $\widehat{w}_{h-1}(V_h^{\mathrm{pl}})$ and hence

$$\vartheta_{h-1} = \bar{c}_{h-1} + \widehat{w}_{h-1}(V_h^{\mathrm{pl}}).$$

Thus, ϑ_h can be computed in a backward induction manner. This then allows us to evaluate $Q_h^{\mathrm{pl}}(s,a)$ and $V_h^{\mathrm{pl}}(s)$ efficiently for any queried (s,a).