NEAR-OPTIMAL PAC LEARNING FOR LOW-RANK MDPS

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-rank Markov decision processes (MDPs) provide a general framework for reinforcement learning in large or continuous state spaces with unknown representations. However, whether near-optimal sample complexity can be achieved using only a maximum-likelihood estimation (MLE) oracle and polynomial additional computation remains unclear. In this work, we resolve this question for probably approximately correct (PAC) learning with known per-step rewards in $[0, 1]$. We propose an uncertainty marking algorithm for both time-homogeneous and time-inhomogeneous MDPs. Its exploration bonus controls the model error over a whole trajectory using the probability of at least one uncertainty mark. The algorithm uses MLE as its only optimization oracle, with bonus estimation and optimistic planning implemented in polynomial time using fitted-model evaluations and samples. We show that, with probability at least $1 - \delta$, our algorithm returns an ε-optimal policy using $\widetilde{O}(H^3 dA \log |\Theta|/\varepsilon^2)$ episodes in the homogeneous setting and $\widetilde{O}(H^4 dA \log |\Theta|/\varepsilon^2)$ episodes in the inhomogeneous setting, where H is the horizon, d is the rank, A is the number of actions, and Θ is the common finite realizable kernel class with $|\Theta| \geq 2$. Moreover, we establish lower bounds that match these rates up to logarithmic factors, showing that our algorithm achieves near-optimal PAC sample complexity in both settings. Finally, we validate our algorithm through experiments on homogeneous block MDPs.

1 INTRODUCTION

Low-rank Markov decision processes (MDPs) provide a general framework for reinforcement learning (RL) in complex environments with large or continuous state spaces (Agarwal et al., 2020). They factor the transition kernel into two unknown low-dimensional feature maps, allowing nonlinear representations while retaining a linear structure for theoretical analysis. Under suitable regularity conditions, many continuous-state transition kernels also admit approximate low-rank factorizations (Huang et al., 2026). These properties make low-rank MDPs a useful model for studying representation learning and exploration. Despite substantial progress, their optimal sample complexity and the computation needed to attain it remain unclear.

There are two main difficulties. First, sharp lower bounds are needed to characterize the statistical difficulty of learning low-rank MDPs. Existing lower bounds (Cheng et al., 2023) do not determine the joint dependence on the horizon, rank, number of actions, and model-class complexity. Establishing this dependence requires hard instances that encode a sufficiently rich model class while keeping the rank independent of the horizon. Such bounds must also distinguish the cost of learning one shared transition kernel from that of learning stage-dependent kernels.

Second, an algorithm must attain this statistical limit while using weak computational oracles. Both feature maps are unknown, so exploration and planning must use representations learned from adaptively collected data. Existing approaches based on maximum-likelihood estimation (MLE) have polynomial sample-complexity guarantees, but retain higher-order dependence on the rank and number of actions (Agarwal et al., 2020; Uehara et al., 2022; Cheng et al., 2023; Huang et al., 2026). MOPS achieves linear dependence on their product through optimistic posterior sampling and model planning (Agarwal & Zhang, 2022). It remains unclear whether ordinary MLE suffices for near-optimal sample complexity with efficient bonus construction and planning.

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.

We take MLE as the only optimization oracle, together with evaluation and sampling access to fitted models. This leads to the following central question:

Can we achieve near-optimal sample complexity for low-rank MDPs using only an MLE oracle and polynomial additional computation?

We answer this question affirmatively for probably approximately correct (PAC) learning with known rewards. We consider both time-homogeneous MDPs, where stages share one transition kernel, and time-inhomogeneous MDPs, where kernels may vary across stages. Our contributions are summarized as follows.

- **Algorithm.** We propose an *uncertainty marking* algorithm for both settings, using MLE as its only optimization oracle. We construct an exploration bonus that controls the model error over a whole trajectory and design polynomial-time subroutines for bonus estimation and optimistic planning using fitted-model evaluations and samples. Thus all computation beyond the MLE oracle calls is polynomial in the problem parameters.
- **Upper bounds.** With probability at least $1-\delta$, our algorithm returns an ϵ-optimal policy using $\widetilde{O}(H^3 dA \log|\Theta|/\epsilon^2)$ episodes in the homogeneous setting and $\widetilde{O}(H^4 dA \log|\Theta|/\epsilon^2)$ episodes in the inhomogeneous setting, for known per-step rewards in $[0,1]$. Here H is the horizon, d is the rank, A is the number of actions, and $|\Theta| \geq 2$ is the common kernel-class size. To our knowledge, these are the tightest known PAC bounds for this setting. In particular, both bounds improve the corresponding MOPS guarantee by a factor of H, up to logarithmic factors.
- **Lower bounds.** We establish separate lower bounds for the homogeneous and inhomogeneous settings that match our upper bounds up to logarithmic factors in the parameter ranges of Theorems 9 and 10. These bounds allow unrestricted computation and keep the rank independent of the horizon. The homogeneous lower bound also matches the confidence dependence up to logarithmic factors. Together, these lower bounds show that our algorithm achieves near-optimal PAC sample complexity in both settings using MLE and fitted-model access.

The main difficulty is to accumulate model error across changing learned representations. A function feasible for our radius under the fitted model remains feasible under the true model after a constant-factor budget increase. This permits an analysis in the fixed true feature space. The binary variable gives a trajectory error bound, avoiding the horizon loss from summing one-step total variation bounds. These arguments yield the dimension and horizon dependence in our guarantees.

1.1 Related work

Statistical efficiency with stronger oracles. OLIVE and witness-rank elimination use global optimism subject to empirical error constraints (Jiang et al., 2017; Sun et al., 2019). GOLF uses squared Bellman-error constraints; its coverability analysis gives a linear dependence on the coverability coefficient under Bellman completeness (Jin et al., 2021; Xie et al., 2023). Bilinear classes provide another general framework (Du et al., 2021), and E2D reduces exploration to online estimation and decision-estimation coefficient (DEC) optimization (Foster et al., 2021). MOPS obtains linear dA dependence through optimistically tilted posterior sampling and model planning (Agarwal & Zhang, 2022). These optimization requirements differ from ordinary MLE. Our contribution combines near-optimal low-rank PAC rates with an implementation using MLE and fitted-model access.

Model-based representation learning. FLAMBE establishes polynomial sample complexity for learning low-rank transition models using MLE and model sampling (Agarwal et al., 2020). REP-UCB improves the statistical guarantee through optimistic exploration with learned features (Uehara et al., 2022). RAFFLE improves reward-free exploration (Cheng et al., 2023), while Opt-AC studies actor–critic updates through supervised-learning reductions (Huang et al., 2026). Their objectives and access assumptions differ: REP-UCB uses a discounted objective, RAFFLE supports rewards revealed after exploration, and Opt-AC uses additional sampling and coverage conditions. Table 1 compares the episodic PAC guarantees after accounting for reward normalization and class complexity.

Table 1: PAC episode complexity for known per-step rewards in $[0, 1]$ and common kernel class $|\Theta| \geq 2$. "No" allows stage-dependent kernels. PP, PE, and OPS denote model planning, policy evaluation, and optimistic posterior sampling; "—" allows unrestricted computation. Appendix F gives conversions and MOPS confidence amplification. GOLF appears in Table 3. Lower-bound ranges are in Theorems 9 and 10. Tildes hide logarithms, including confidence factors.

Algorithm / result	Time-homogeneous	Strongest oracle	Sample complexity
OLIVE	No	Constrained opt.	$\widetilde{O}(H^6d^2A\log\lvert\Theta\rvert/\epsilon^2)$
Witness-rank elimination	No	Constrained opt.	$\widetilde{O}(H^6d^2A\log\lvert\Theta\rvert/\epsilon^2)$
MOPS	Yes	OPS + PP	$\widetilde{O}(H^4dA\log\lvert\Theta\rvert/\epsilon^2)$
MOPS	No	OPS + PP	$\widetilde{O}(H^5dA\log\lvert\Theta\rvert/\epsilon^2)$
RFOLIVE	No	Constrained opt.	$\widetilde{O}(H^9d^3A\log\lvert\Theta\rvert/\epsilon^2)$
RAFFLE	No	PP + MLE	$\widetilde{O}(H^5d^2A(d^2+A)\log\lvert\Theta\rvert/\epsilon^2)$
Opt-AC	No	PE + MLE	$\widetilde{O}(H^5d^3A^2\log\lvert\Theta\rvert/\epsilon^2)$
Ours	Yes	MLE	$\widetilde{O}(H^3dA\log\lvert\Theta\rvert/\epsilon^2)$
Ours	No	MLE	$\widetilde{O}(H^4dA\log\lvert\Theta\rvert/\epsilon^2)$
Lower bounds (ours)	Yes	—	$\widetilde{\Omega}(H^3dA\log\lvert\Theta\rvert/\epsilon^2)$
Lower bounds (ours)	No	—	$\widetilde{\Omega}(H^4dA\log\lvert\Theta\rvert/\epsilon^2)$

Model-free representation learning. MOFFLE learns state-action features using minimax objectives under latent-state reachability (Modi et al., 2024). BRIEE gives a model-free approach for block MDPs without reachability (Zhang et al., 2022). RFOLIVE establishes reward-free sample efficiency for general low-rank MDPs, without an efficient implementation (Chen et al., 2022). VoX removes reachability for oracle-efficient model-free exploration through barycentric spanners (Mhammedi et al., 2024). These methods require only a class for the state-action features. Our method assumes classes for both factors, which enable density evaluation and sampling from fitted transitions.

Density ratios and coverability. Density ratios and coverability are also related to our techniques. Huang et al. (2023) use next-state density features to estimate occupancy distributions and construct exploratory data. Coverability gives a broader condition for online exploration (Xie et al., 2023). GLOW uses density-ratio and value-function realizability (Amortila et al., 2024a); CODEX optimizes coverage objectives through policy optimization (Amortila et al., 2024b). Our radius constrains next-state test functions by their historical fitted expectations. Its analysis combines a binary-testing formula (Elkayam & Feder, 2016), online ℓ_1 sensitivity (Woodruff & Yasuda, 2022), and a trajectory Hellinger inequality (Foster et al., 2024).

Lower bounds and computational assumptions. Episodic tabular lower bounds identify the cost of stage-dependent transitions (Domingues et al., 2021), and Cheng et al. (2023) establish a low-rank reward-free lower bound. Our constructions additionally capture model-class entropy while keeping rank independent of H. Oracle choice also matters: Rohatgi & Foster (2025) give cryptographic evidence that a regression oracle sufficient for block MDPs is insufficient for general low-rank MDPs. Our computational guarantee retains global MLE and fitted-model access. Related results address constant regret under gap and occupancy assumptions (Sturm & Tschiatschek, 2025), and adversarial losses (Liu et al., 2024). Table 1 compares our sample complexity and oracle requirements with prior work. In appendix, we have more detailed comparisons (Tables 3 and 4).

2 Preliminaries and problem formulation

Notation. For a positive integer n, write $[n] = \{1, \ldots, n\}$. Let $\|x\|$ be the Euclidean norm and $\|x\|_M = (x^\top Mx)^{1/2}$ for a positive definite matrix M. Write $\mathbf{1}\{E\}$ for the indicator of E, $[x]_+ = \max\{x, 0\}$, and $\log_+ x = \max\{0, \log x\}$. For simplicity of notation, we write sums over sample spaces; in continuous spaces, these denote integrals with respect to the relevant reference measure. This convention does not restrict the state space. For probability laws P, Q on the same sample

space, let p, q be their densities with respect to a common dominating measure. With x ranging over that space, define $D_{\mathrm{TV}}(P,Q) = \frac{1}{2}\sum_x |p(x) - q(x)|$, $D^2_{\mathrm{H}}(P,Q) = \sum_x \big(\sqrt{p(x)} - \sqrt{q(x)}\big)^2$, and $D_{\mathrm{KL}}(P\|Q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$. Thus $D^2_{\mathrm{H}}(P,Q) \in [0,2]$ and $D_{\mathrm{TV}}(P,Q)^2 \le D^2_{\mathrm{H}}(P,Q)$. The notation $\widetilde{O}$ hides logarithms of $H, d, A, 1/\epsilon, 1/\delta$ and the displayed class-complexity budget.

Episodic MDPs. We consider an episodic MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P^\star, r)$ with state space $\mathcal{S}$, finite action space $\mathcal{A}$ of size A, horizon $H \ge 2$, and fixed initial state s_1. At step h, the learner takes action a_h, receives the known reward $r_h(s_h, a_h) \in [0,1]$, and observes $s_{h+1} \sim P^\star_h(\cdot \mid s_h, a_h)$. An episode contains all H transitions, including s_{H+1}. The sequence $P^\star = (P^\star_1, \dots, P^\star_H)$ is fixed across episodes. We note that $P^\star_h(s' \mid s, a)$ denotes a probability mass in a discrete state space and a density in a continuous state space.

For a kernel sequence P and policy π, let $\mathbb{E}_{P,\pi}$ denote expectation under their trajectory law and $V^\pi_{P,r} = \mathbb{E}_{P,\pi}\sum_{h=1}^H r_h(s_h, a_h)$. Write $[Pf](s,a) = \sum_{s' \in \mathcal{S}} P(s' \mid s, a) f(s') = \mathbb{E}_{S \sim P(\cdot|s,a)} f(S)$. For a Markov policy, define $V^\pi_{h,P,r}(s)$ and $Q^\pi_{h,P,r}(s,a)$ as its expected remaining returns from state s and state-action pair (s,a) at step h. With $V^\pi_{H+1,P,r} = 0$, they satisfy

$$Q^\pi_{h,P,r}(s,a) = r_h(s,a) + [P_h V^\pi_{h+1,P,r}](s,a), \qquad V^\pi_{h,P,r}(s) = \mathbb{E}_{a \sim \pi_h(\cdot|s)} Q^\pi_{h,P,r}(s,a).$$

Let $V^\star = \sup_\pi V^\pi_{P^\star,r}$. For accuracy $\epsilon > 0$ and failure probability $\delta \in (0,1)$, the learner is (ϵ, δ)-PAC with sample complexity N if it collects N episodes and its output $\bar{\pi}$ satisfies

$$\mathbb{P}\left(V^\star - V^{\bar{\pi}}_{P^\star,r} \le \epsilon\right) \ge 1 - \delta.$$

We adopt the standard low-rank MDP model and realizability assumption of Agarwal et al. (2020). A broad class of continuous-state transition kernels also admits approximate low-rank factorizations under suitable regularity conditions (Huang et al., 2026).

Definition 1 (Low-rank MDP)**.** For an integer $d \ge 1$, an MDP has rank at most d if, for any $h \in [H]$, there are unknown feature maps $\phi^\star_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ and $\mu^\star_h : \mathcal{S} \to \mathbb{R}^d$ such that $P^\star_h(s' \mid s, a) = \phi^\star_h(s,a)^\top \mu^\star_h(s')$, where $\|\phi^\star_h(s,a)\| \le 1$, and $\|\sum_{s' \in \mathcal{S}} \mu^\star_h(s') f(s')\| \le \sqrt{d} \quad \forall f : \mathcal{S} \to [0,1]$.

Assumption 2 (Realizability)**.** The learner is given a finite class Θ of proper transition kernels, each with a factorization satisfying Definition 1. The true kernel $P^\star_h$ belongs to Θ for every $h \in [H]$.

In the *time-homogeneous* setting, $P^\star_h = P^\star$ for all h; write $\phi^\star, \mu^\star$ for the shared factors. In the *time-inhomogeneous* setting, the kernels and factors may differ across stages, and the sequence class is Θ^H. Both settings allow stage-dependent rewards.

MLE Oracle. An approximate MLE oracle returns a candidate within total negative log-likelihood error $\tau = \log(|\Theta|/\delta)$. For a fitted kernel, the learner can evaluate its density and both factors and can sample its next state at any state-action pair. These samples are separate from environment interactions. Our computation bounds count oracle calls and real arithmetic.

3 Algorithm and Sample Complexity

Our proposed Algorithm 1 alternates between model estimation, optimistic planning, and η-greedily collecting B episodes. The uniform-exploration probability $\eta = 1/(2H)$. At a high level, the algorithm runs for K iterations, indexed by $k = 1, \dots, K$, with B complete episodes per iteration. Thus it uses KB real episodes and KBH environment transitions. Episode i has observations $(s_{i,h}, a_{i,h}, s_{i,h+1})$ at stage h; iteration k contains episodes $i = (k-1)B + 1, \dots, kB$. We denote a next-state random variable by S and a realization by s' or $s_{i,h+1}$.

Model estimation At the start of iteration k, we already collected $(k-1)B$ episodes. In the homogeneous setting, let $\widehat{P}^{(k)}_h = \widehat{P}^{(k)}$, and the MLE oracle returns $\widehat{P}^{(k)} \in \Theta$ satisfying

$$\sum_{i=1}^{(k-1)B} \sum_{h=1}^{H} \log \widehat{P}^{(k)}(s_{i,h+1} \mid s_{i,h}, a_{i,h}) \ge \max_{P \in \Theta} \sum_{i=1}^{(k-1)B} \sum_{h=1}^{H} \log P(s_{i,h+1} \mid s_{i,h}, a_{i,h}) - \tau. \qquad (1)$$

In the inhomogeneous setting, the oracle returns $\widehat{P}_h^{(k)} \in \Theta$ for each $h \in [H]$, satisfying

$$\sum_{i=1}^{(k-1)B} \log \widehat{P}_h^{(k)}(s_{i,h+1} \mid s_{i,h}, a_{i,h}) \geq \max_{P \in \Theta} \sum_{i=1}^{(k-1)B} \log P(s_{i,h+1} \mid s_{i,h}, a_{i,h}) - \tau. \tag{2}$$

Both objectives use total log-likelihoods, with the fixed tolerance $\tau = \log(|\Theta|/\delta)$. At $k = 1$, initialize any model from Θ. Note that for each h, $\widehat{P}_h^{(k)}(s' \mid s, a) = \widehat{\phi}_h^{(k)}(s,a)^\top \widehat{\mu}_h^{(k)}(s')$.

Measuring model uncertainty. For $h < H$, we define the following multiset of historical inputs:

$$\mathcal{I}_{k,h} = \begin{cases} \{(s_{i,j}, a_{i,j}) : i \leq (k-1)B,\ j < H\}, & \text{time-homogeneous,} \\ \{(s_{i,h}, a_{i,h}) : i \leq (k-1)B\}, & \text{time-inhomogeneous.} \end{cases} \tag{3}$$

For a confidence budget $L_{\mathrm{rad}} > 0$, define radius $u_{k,h}(s,a)$ as follows.

$$u_{k,h}(s,a) = \sup_{0 \leq f \leq 2} \left\{ [\widehat{P}_h^{(k)} f](s,a) : \sum_{(s_{i,j}, a_{i,j}) \in \mathcal{I}_{k,h}} [\widehat{P}_h^{(k)} f](s_{i,j}, a_{i,j}) \leq L_{\mathrm{rad}} \right\}. \tag{4}$$

The radius is large when some next-state function has small historical expectations but a large expectation at (s,a). As shown in Section 4.1, it bounds the next-step model error. Homogeneous models share one radius across h. To make the algorithm computable, we provide a subroutine EstimateRadius (Algorithm 2) to estimate Equation (4). We show that, for parameters $\zeta \in (0,1]$ and $\delta_0 \in (0,1)$, at each fixed (k,h,s,a), let

$$\widehat{u}_{k,h}(s,a) = \textsc{EstimateRadius}(\widehat{P}_h^{(k)}, \mathcal{I}_{k,h}, L_{\mathrm{rad}}, s, a; \zeta, \delta_0). \tag{5}$$

Then, Proposition 25 proves that $\widehat{u}_{k,h}(s,a) \in [0,2]$ and with probability $1-\delta_0$, we have $u_{k,h}(s,a) \leq \widehat{u}_{k,h}(s,a) \leq \min\{2, 4u_{k,h}(s,a) + \zeta\}$. We denote $\widehat{u}_h = (\widehat{u}_{k,h}(s_h,a))_{a \in \mathcal{A}}$ the returned vector, and set $\widehat{u}_{k,H}(s,a) = 0$.

Constructing the exploration bonus using a mark. At each visit to s_h with $h < H$, draw fresh independent radius estimates $\widehat{u}_h$. The augmented state is $\tilde{s}_h = (s_h, \widehat{u}_h, z_h)$, with $\widehat{u}_H = 0$. The binary variable z_h records whether a mark has occurred before stage h, with $z_1 = 0$. The policy first observes $\tilde{s}_h$ and selects action a_h. If $z_h = 0$, toss a coin with success probability $\min\{1, \widehat{u}_{k,h}(s_h,a_h)\}$ and set z_{h+1} to its outcome; otherwise set $z_{h+1} = 1$. A mark is the change from zero to one, so each episode has at most one mark. Original transitions and rewards continue after marking.

Throughout, we augment the policy to receive $\tilde{s}_h$. With a little abuse of notation, the value notation and $\widehat{P}^{(k)}$ extends to these policies and $\tilde{s}_h$. Under $\widehat{P}^{(k)}$ and policy π, define the marking probability

$$C_k(\pi) = \mathbb{P}_{\widehat{P}^{(k)},\pi}(z_{H+1} = 1) = \mathbb{E}_{\widehat{P}^{(k)},\pi}\left[\sum_{h=1}^{H-1} \mathbf{1}\{z_h = 0\} \min\{1, \widehat{u}_{k,h}(s_h,a_h)\} \right] \in [0,1]. \tag{6}$$

Set $\rho_1 = 2$ and $\rho_k = \min\{2, 20AH\log(12H|\Theta|/\delta)/[(k-1)B]\}$ for $k \geq 2$. The quantity ρ_k bounds the initial-state model error on the MLE confidence event. Let $\Delta_{\mathrm{MC}} = 28(H-1)A\delta_0$ account for radius-estimation error, and let $\alpha > 0$ be a regularization parameter. With the parameter choices in Appendix A.1, define the optimistic objective

$$J_k(\pi) = V^\pi_{\widehat{P}^{(k)},r} + H\sqrt{18C_k(\pi) + 14\rho_k + \Delta_{\mathrm{MC}} + \alpha}. \tag{7}$$

This bonus design is a key ingredient in achieving our near-optimal sample complexity bounds. Because counting only the first mark keeps $C_k(\pi)$ in $[0,1]$, even when uncertainty is large at several stages. After the first mark, later uncertainty adds no further marking reward. This allows the bonus to control the model error over the whole trajectory, as shown in Section 4.1. The construction serves a similar purpose to the truncated exploration value in Cheng et al. (2023). It also permits efficient planning: for any fixed $\gamma \geq 0$, the objective $V^\pi_{\widehat{P}^{(k)},r} + \gamma C_k(\pi)$ is an additive reward objective that pays γ at the first mark. Appendix E.5 uses these additive problems to maximize (7) approximately.

Planning and collection. The subroutine $\textsc{Plan}(J_k; \epsilon/100, \delta/(4K))$ receives the fitted kernels, known reward, radius rules, and budgets defining J_k. Conditional on these inputs, it returns an executable randomized policy π_k satisfying $J_k(\pi_k) \geq \sup_\pi J_k(\pi) - \epsilon/100$ with probability at least $1 - \delta/(4K)$. Finally, we use π_k to do η-greedy exploration to collect B episodes. The resulting behavior policy is β_k, i.e. $\beta_k(a|s) = (1-\eta)\pi_k(a|s) + \eta/A$. Finally, the output $\bar{\pi}$ samples k uniformly from $[K]$ and executes π_k without added uniform exploration. The pseudocode is in Algorithm 1. All subroutines can be found in Section E.

Algorithm 1 Uncertainty marking for low-rank MDPs

Require: Model class Θ, $H, d, A, \epsilon, \delta$; iteration count K, episodes per iteration B, and other parameters from Appendix A.1.

Initialize the transition data $\mathcal{D} = \varnothing$.
for $k = 1, \ldots, K$ **do**
  Model estimation: obtain $\widehat{P}^{(k)}$ by (1) or $\{\widehat{P}_h^{(k)}\}_{h=1}^H$ by (2).
  Exploration bonus: form the histories (3) and fix the radius rules.
  Policy update: $\pi_k \leftarrow \textsc{Plan}(J_k; \epsilon/100, \delta/(4K))$.
  for $i = (k-1)B + 1, \ldots, kB$ **do**
    Initialize π_k for a new episode; set $s_{i,1} = s_1$ and $z_1 = 0$.
    for $h = 1, \ldots, H$ **do**
      If $h < H$, obtain $\widehat{u}_{k,h}$ by calling (5) at $(s_{i,h}, a)$ for every $a \in \mathcal{A}$; otherwise set all radius estimates to zero.
      Query π_k with $(s_{i,h}, \widehat{u}_{k,h}, z_h)$; execute its action with probability $1 - \eta$, or a uniform action otherwise, obtaining $a_{i,h}$.
      Set $z_{h+1} = 1$ if $z_h = 1$; otherwise set it to one with probability $\min\{1, \widehat{u}_{k,h}(s_{i,h}, a_{i,h})\}$, or zero.
      Observe $s_{i,h+1}$ and add $(h, s_{i,h}, a_{i,h}, s_{i,h+1})$ to $\mathcal{D}$.
    end for
  end for
end for
Return the uniform mixture $\bar{\pi} = \mathrm{Unif}(\pi_1, \ldots, \pi_K)$, without the added uniform exploration.

4 Theoretical guarantees

Use the common confidence budget $\ell = 10 \log(12H|\Theta|/\delta)$. Set $\Xi = \ell$ in the homogeneous case and $\Xi = H\ell$ in the inhomogeneous case. We use $B = \lceil A\ell \rceil$ and $B = \lceil 2AH\ell \rceil$, respectively.

Theorem 3 (Time-homogeneous upper bound)**.** *Under the homogeneous model and access conditions of Section 2, for $0 < \epsilon \leq 1$ and $0 < \delta < 1$, Algorithm 1 with the homogeneous configuration returns an ϵ-optimal policy with probability at least $1 - \delta$ and sample complexity*

$$N = KB = \widetilde{O}\left(\frac{H^3 dA}{\epsilon^2} \log \frac{2|\Theta|}{\delta} \right). \tag{8}$$

Theorem 4 (Time-inhomogeneous upper bound)**.** *Under the inhomogeneous model and access conditions of Section 2, for $0 < \epsilon \leq 1$ and $0 < \delta < 1$, Algorithm 1 with the inhomogeneous configuration returns an ϵ-optimal policy with probability at least $1 - \delta$ and sample complexity*

$$N = KB = \widetilde{O}\left(\frac{H^4 dA}{\epsilon^2} \log \frac{H|\Theta|}{\delta} \right). \tag{9}$$

Theorem 4 improves the sample complexity bounds of Opt-AC (Huang et al., 2026) and RAFFLE (Cheng et al., 2023) by factors of Hd^2A and $Hd(d^2 + A)$, respectively, up to logarithmic factors under the normalization in Table 1. This comparison concerns known-reward learning; RAFFLE also provides a reward-free guarantee. Compared with MOPS (Agarwal & Zhang, 2022), both upper bounds save a factor of H, while using MLE and fitted-model access. The additional factor of H in the inhomogeneous bound comes from estimating H kernels in Θ, with sequence-class entropy $\log |\Theta^H| = H \log |\Theta|$, instead of one shared kernel. Exploration and planning are the same in both settings. Combined with the lower bounds in Theorems 9 and 10, these results establish **near-optimal** PAC sample complexity in $H, d, A, 1/\epsilon$, and $\log |\Theta|$ for both settings.

Computational complexity Under the access conditions of Section 2 and the parameter choices of Appendix A.1, Algorithm 1 uses $\widetilde{O}(H^3 d/\epsilon^2)$ MLE oracle calls in the homogeneous setting and $\widetilde{O}(H^4 d/\epsilon^2)$ in the inhomogeneous setting. In both settings, the remaining computation uses $\mathrm{poly}(d, A, H, 1/\epsilon, \log(|\Theta|/\delta))$ fitted-model sampling and evaluation calls and real arithmetic operations. The computational bounds are given in Appendix E.6.

4.1 Proof overview

We explain how the following three lemmas and one corollary imply Theorems 3 and 4. We use the parameters of Appendix A.1 throughout.

MLE error Let $\ell_{\mathrm{emp}} = 3\log(12H|\Theta|/\delta)$ be the empirical-risk budget.

Lemma 5 (Good event for MLE). *There is an event $\mathcal{E}_{\mathrm{MLE}}$ of probability at least $1-\delta/6$ on which, at every iteration k, the homogeneous fit satisfies*

$$\sum_{i=1}^{(k-1)B}\sum_{h=1}^{H} D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_{i,h}, a_{i,h}), \widehat{P}^{(k)}(\cdot \mid s_{i,h}, a_{i,h})\big) \le \ell_{\mathrm{emp}}. \tag{10}$$

In the inhomogeneous case, the same bound holds for each h with $P_h^\star, \widehat{P}_h^{(k)}$ and the sum over h omitted. The event also satisfies the conditional-error bounds in Appendices A.2 and B.1.

Lemma 5 controls the error of each fitted model on the previously collected state-action pairs, simultaneously over all iterations. Its event is used in the trajectory comparison and optimism guarantees below. The proof is in Appendices A.2 and B.1.

Trajectory error and optimism For fixed k and policy π, let $P^\pi, \widehat{P}^\pi$ be its trajectory laws through the last reward under the true and fitted models, including all internal random variables.

Lemma 6 (Trajectory comparison). *On the event $\mathcal{E}_{\mathrm{MLE}}$ of Lemma 5, for every k and policy π using the implemented marking rule,*

$$D_{\mathrm{H}}^2(P^\pi, \widehat{P}^\pi) \le 18C_k(\pi) + 14\rho_k + \Delta_{\mathrm{MC}}. \tag{11}$$

The marking probability also satisfies

$$C_k(\pi) \le 8\mathbb{E}_{P^\star,\pi}\sum_{h<H} u_{k,h}(s_h,a_h) + 4HA\delta_0 + 14\mathbb{E}_{P^\star,\pi}\sum_{h\le H} D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h,a_h), \widehat{P}_h^{(k)}(\cdot \mid s_h,a_h)\big).$$

The first bound shows that the marking probability controls the model error over a whole trajectory. The second bounds this probability by the expected radii and model errors along the policy's trajectory under $P^\star$. The proof is in Appendix A.4 for both settings.

Corollary 7 (Optimism and policy error). *On $\mathcal{E}_{\mathrm{MLE}}$, $J_k(\pi) \ge V^\pi_{P^\star,r}$ for every policy π. If $J_k(\pi_k) \ge \sup_\pi J_k(\pi) - \epsilon/100$, then*

$$V^\star - V^{\pi_k}_{P^\star,r} \le 2H\sqrt{18C_k(\pi_k) + 14\rho_k + \Delta_{\mathrm{MC}} + \alpha} + \epsilon/100.$$

Corollary 7 converts the trajectory bound into optimism of J_k and bounds the value gap of its approximate maximizer by the exploration bonus and planning error. The proof is in Appendix A.5.

Cumulative uncertainty in the true feature space Recall that $\Xi = \ell$ in the homogeneous case and $\Xi = H\ell$ in the inhomogeneous case, where $\ell = 10\log(12H|\Theta|/\delta)$.

Lemma 8 (Sublinear summation). *With probability at least $1-5\delta/12$, Algorithm 1 satisfies*

$$B\sum_{k=1}^{K}\mathbb{E}_{P^\star,\pi_k}\left[\sum_{h<H} u_{k,h}(s_h,a_h) + \sum_{h\le H} D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h,a_h), \widehat{P}_h^{(k)}(\cdot \mid s_h,a_h)\big)\right] = \widetilde{O}(dAH\Xi). \tag{12}$$

Moreover, $B\sum_{k=1}^{K}\rho_k = \widetilde{O}(AH\Xi)$.

Lemma 8 bounds the total expected uncertainty under the policies generated during training. Its right-hand side grows only logarithmically with K, so the average uncertainty decreases as more episodes are collected. The proof is in Appendices A.7 and B.3.

Combining the bounds Lemmas 6 and 8 give $B\sum_{k=1}^{K} C_k(\pi_k) = \widetilde{O}(dAH\Xi) + O(KBAH\delta_0)$. Averaging Corollary 7 over the output mixture $\bar{\pi}$ and using the prescribed numerical parameters then gives

$$V^\star - V^{\bar{\pi}}_{P^\star,r} \le \widetilde{O}\left(H\sqrt{\frac{dAH\Xi}{KB}}\right) + 0.04\epsilon.$$

Thus $N = KB = \widetilde{O}(H^3 dA\Xi/\epsilon^2)$ episodes suffice. Substituting $\Xi = \ell$ and $\Xi = H\ell$ yields the homogeneous and inhomogeneous bounds, respectively. The failure probabilities, including those of planning, sum to at most δ. Appendices A.8 and B.4 give the parameter substitution and complete the proofs of Theorems 3 and 4.

5 Lower bounds

Let N count real episodes, as in the PAC definition; Algorithm 1 uses $N = KB$. Both lower bounds are in the worst case and allow arbitrary computation and adaptive data collection, with known rewards in $[0, 1]$. They hold even when the next-state feature map $\mu^\star$, shared by all candidates in each construction, is revealed.

Theorem 9 (Time-homogeneous lower bound)**.** *Let $d \ge 4$, $A \ge 2$, $0 < \epsilon \le 1$, and $0 < \delta \le 1/16$. For sufficiently large H, there exists a finite class Θ of time-homogeneous kernels with rank at most d such that every algorithm that is (ϵ, δ)-PAC on the class requires sample complexity at least*

$$N = \Omega\left(\frac{H^3 dA}{\epsilon^2}\left[\log|\Theta| + \log\frac{1}{\delta}\right]\right). \tag{13}$$

This matches the model-class and confidence terms in Theorem 3 up to logarithms. Appendix C.1 allows horizons sufficiently long relative to $\log d$ and arbitrarily large class entropies.

The proof of Theorem 9 uses two main ingredients. First, we construct a time-homogeneous MDP in which one action repeatedly draws a new state from a fixed distribution, while the remaining actions lead to absorbing states. A policy knowing the model can wait for a state where an action is more likely to reach the high-reward absorbing state. The learner, however, receives at most one informative transition per episode. This allows the learning difficulty to grow with H.

Second, we construct many candidate kernels that share these distributions but have different probabilities of reaching the absorbing states. An information-theoretic argument shows that finding a near-optimal policy requires $\Omega(H^3 dA\log|\Theta|/\epsilon^2)$ episodes, even when the learner chooses its actions adaptively based on the observed states. Comparing alternative kernels within the same class also gives the $\Omega(H^3 dA\log(1/\delta)/\epsilon^2)$ term. The detailed construction and proof are in Appendix C.

Theorem 10 (Time-inhomogeneous lower bound)**.** *Let $d \ge 5$, $A \ge 2$, $0 < \epsilon \le 1$, and $0 < \delta \le 1/16$. For sufficiently large H, there exists a finite class Θ of kernels with rank at most d such that every algorithm that is (ϵ, δ)-PAC on Θ^H requires sample complexity at least*

$$N = \Omega\left(\frac{H^4 dA\log|\Theta|}{(1+\log d)^2\epsilon^2} + \frac{H^3 dA\log(1/\delta)}{(1+\log d)\epsilon^2}\right). \tag{14}$$

Theorem 10 matches the upper bound in Theorem 4 up to logarithmic factors, showing that our uncertainty marking algorithm is near-optimal in the inhomogeneous setting. The proof is in Appendix D.

6 Experiments

In this section, we study whether the uncertainty marking bonus helps a neural implementation explore homogeneous low-rank MDPs with continuous states. These experiments serve as a sanity check of the implementation and its ability to find optimal or near-optimal policies in challenging exploration environments. They are not intended to establish a general performance advantage over standard reinforcement learning algorithms.

Setup. The state space is $\mathbb{R}^{64}$, the horizon is $H = 48$, and the rank is at most 26. Reaching the goal requires 24 consecutive correct actions; any wrong action enters an absorbing safe block. Each transition produces a fresh observation from a ball associated with the destination block. One correct action is sampled independently and uniformly at each progress block and then fixed across episodes; block labels and correct actions are unknown. The optimal value is 24, whereas entering the safe block immediately gives 9.4.

Methods and evaluation. We learn both factors of a 26-component Gaussian mixture transition model by neural maximum likelihood and use numerical integration for bonus computation and planning. We compare against the same model without the bonus, PPO, Double DQN, DQN with a horizon prior, and Bootstrapped DQN with a random prior. All methods receive the same state observations and use four paired seeds with equal interaction budgets. We set $B = 128A$ and $K = 128$, and extend every method to $K = 256$ at $A = 8$. We evaluate the mixture of policies used in the last 1,024 training episodes; Appendix G gives implementation details, independent final evaluations, and the full-history output used in the theory.

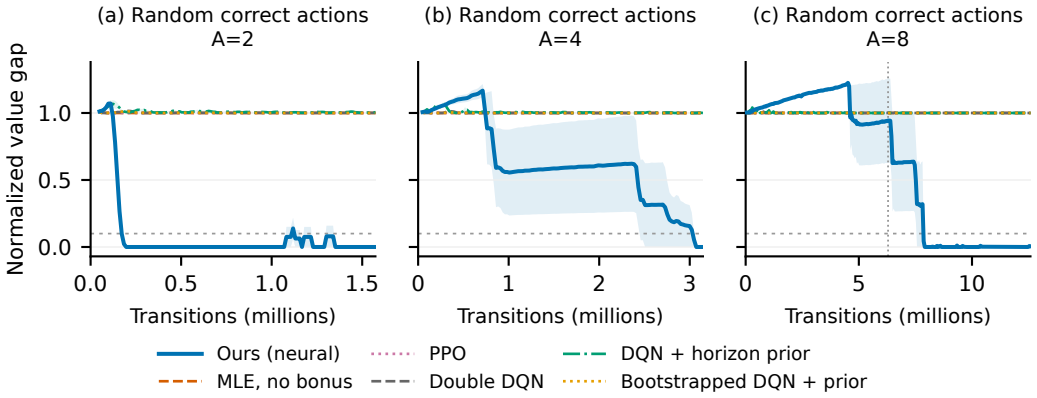


Figure 1: Learning curves against real training transitions. The normalized value gap is $(24 - V)/14.6$: zero is optimal and one is the safe value. Lines show four-seed means; bands show one standard error. Each checkpoint evaluates the preceding 1,024-episode policy mixture. The horizontal dotted line is the target 0.1; the vertical line in (c) marks the original $K = 128$ budget. Several baseline curves overlap near one.

Results. In Figure 1(a,b), our method achieves final evaluated return 24 on all four seeds for $A = 2, 4$, while all five baselines remain near the safe value. The no-bonus comparison supports the benefit of exploration with the same model and reward access. In panel (c), doubling the $A = 8$ budget improves our final target success from $1/4$ to $4/4$ seeds, with return 23.93 ± 0.07; all baselines remain at $0/4$. These extensions reuse the original seeds. Gaps above one occur when a policy advances farther but fails before the goal, delaying the safe reward. Appendix G.4 reports runtime and samples to the target. The comparison matches interactions, not computation; model methods also query the known reward. These results support exploration on this block family, but do not verify the PAC rates under the approximate neural implementation.

7 Conclusion

In this paper, we studied PAC learning in time-homogeneous and time-inhomogeneous low-rank MDPs with known rewards. We proposed an uncertainty marking algorithm that uses MLE as its only optimization oracle and implements bonus estimation and planning in polynomial time using fitted-model evaluations and samples. Our upper and lower bounds match up to logarithmic factors, establishing near-optimal sample complexity in both settings. Experiments on homogeneous continuous-state block MDPs support the benefit of uncertainty marking for exploration. Future directions include extending the guarantees to misspecified model classes, reducing fitted-model simulation costs, and determining the sharp confidence dependence in the inhomogeneous setting.

AI USE STATEMENT

In this work, we used generative AI tools to assist with proofs and algorithm design and to write code for the experiments. We did not use generative AI tools for research ideation, problem formulation, interpretation of results, or the development of proof strategies. Additionally, we used generative AI tools for language editing and improving the presentation. We have manually reviewed the AI-assisted proofs and checked their correctness. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Our assumptions are stated in Section 2, and complete proofs are provided in the appendix. Appendix G describes the experimental environment, algorithm implementations, hyperparameters, random seeds, and evaluation protocol. The accompanying code includes the experimental implementations, configurations, saved results, and instructions for running the experiments and reproducing the figures and tables.

REFERENCES

Alekh Agarwal and Tong Zhang. Model-based RL with optimistic posterior sampling: Structural conditions and sample complexity. In *Advances in Neural Information Processing Systems*, volume 35, pp. 35284–35297, 2022. URL https://arxiv.org/abs/2206.07659.

Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. FLAMBE: Structural complexity and representation learning of low rank MDPs. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2006.10814.

Philip Amortila, Dylan J. Foster, Nan Jiang, Ayush Sekhari, and Tengyang Xie. Harnessing density ratios for online reinforcement learning. In *International Conference on Learning Representations*, 2024a. URL https://arxiv.org/abs/2401.09681.

Philip Amortila, Dylan J. Foster, and Akshay Krishnamurthy. Scalable online exploration via coverability. In *International Conference on Machine Learning*, pp. 1392–1455, 2024b. URL https://proceedings.mlr.press/v235/amortila24a.html.

Jinglin Chen, Aditya Modi, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. On the statistical efficiency of reward-free exploration in non-linear RL. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2206.10770.

Yuan Cheng, Ruiquan Huang, Jing Yang, and Yingbin Liang. Improved sample complexity for reward-free reinforcement learning under low-rank MDPs. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2303.10859.

Omar Darwiche Domingues, Pierre Ménard, Emilie Kaufmann, and Michal Valko. Episodic reinforcement learning in finite MDPs: Minimax lower bounds revisited. In *International Conference on Algorithmic Learning Theory*, pp. 578–598, 2021. URL https://proceedings.mlr.press/v132/domingues21a.html.

Simon S. Du, Sham M. Kakade, Jason D. Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in RL. In *International Conference on Machine Learning*, pp. 2826–2836, 2021. URL https://proceedings.mlr.press/v139/du21a.html.

Nir Elkayam and Meir Feder. Variational formulas for the power of the binary hypothesis testing problem with applications. arXiv:1601.06810, 2016. URL https://arxiv.org/abs/1601.06810.

Dylan J. Foster, Sham M. Kakade, Jian Qian, and Alexander Rakhlin. The statistical complexity of interactive decision making. arXiv:2112.13487, 2021. URL https://arxiv.org/abs/2112.13487. Version 3, revised July 2023.

Dylan J. Foster, Yanjun Han, Jian Qian, and Alexander Rakhlin. Online estimation via offline estimation: An information-theoretic framework. arXiv:2404.10122, 2024. URL https://arxiv.org/abs/2404.10122.

Minbo Gao, Tianle Xie, Simon S. Du, and Lin F. Yang. A provably efficient algorithm for linear Markov decision process with low switching cost. arXiv:2101.00494, 2021. URL https://arxiv.org/abs/2101.00494.

Audrey Huang, Jinglin Chen, and Nan Jiang. Reinforcement learning in low-rank MDPs with density features. In *International Conference on Machine Learning*, pp. 13710–13752, 2023. URL https://proceedings.mlr.press/v202/huang23a.html.

Ruiquan Huang, Donghao Li, Yingbin Liang, and Jing Yang. Breaking the computational barrier: Provably efficient actor–critic for low-rank MDPs. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2605.01242.

Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E. Schapire. Contextual decision processes with low Bellman rank are PAC-learnable. In *International Conference on Machine Learning*, pp. 1704–1713, 2017. URL https://proceedings.mlr.press/v70/jiang17c.html.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I. Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, 2020. URL https://arxiv.org/abs/1907.05388.

Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman Eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. In *Advances in Neural Information Processing Systems*, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/6f5e4e86a87220e5d361ad82f1ebc335-Abstract.html.

Romain Laroche and Rémi Tachet des Combes. On the occupancy measure of non-Markovian policies in continuous MDPs. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 18548–18562, 2023. URL https://proceedings.mlr.press/v202/laroche23a.html.

Haolin Liu, Zakaria Mhammedi, Chen-Yu Wei, and Julian Zimmert. Beating adversarial low-rank MDPs with unknown transition and bandit feedback. In *Advances in Neural Information Processing Systems*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/f2ae937c93d45def2b0eb246831c8685-Abstract-Conference.html.

Zakaria Mhammedi, Adam Block, Dylan J. Foster, and Alexander Rakhlin. Efficient model-free exploration in low-rank MDPs. arXiv:2307.03997v2, 2024. URL https://arxiv.org/abs/2307.03997v2. Revised version of the NeurIPS 2023 paper.

Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank MDPs. *Journal of Machine Learning Research*, 25(6):1–76, 2024. URL https://jmlr.org/papers/v25/22-0687.html.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in neural information processing systems*, 31, 2018.

Dhruv Rohatgi and Dylan J. Foster. Necessary and sufficient oracles: Toward a computational taxonomy for reinforcement learning. In *Conference on Learning Theory*, pp. 4838–4936, 2025. URL https://proceedings.mlr.press/v291/rohatgi25b.html.

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

Alexander Sturm and Sebastian Tschiatschek. On constant regret for low-rank MDPs. In *Conference on Uncertainty in Artificial Intelligence*, pp. 4044–4079, 2025. URL https://proceedings.mlr.press/v286/sturm25a.html.

Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based RL in contextual decision processes: PAC bounds and exponential improvements over model-free approaches. In *Conference on Learning Theory*, pp. 2898–2933, 2019. URL https://proceedings.mlr.press/v99/sun19a.html.

Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline RL in low-rank MDPs. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2110.04652.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

David P. Woodruff and Taisuke Yasuda. Online Lewis weight sampling. arXiv:2207.08268, 2022. URL https://arxiv.org/abs/2207.08268.

Tengyang Xie, Dylan J. Foster, Yu Bai, Nan Jiang, and Sham M. Kakade. The role of coverage in online reinforcement learning. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.04157.

Xuezhou Zhang, Yuda Song, Masatoshi Uehara, Mengdi Wang, Alekh Agarwal, and Wen Sun. Efficient reinforcement learning in block MDPs: A model-free representation learning approach. In *International Conference on Machine Learning*, pp. 26517–26547, 2022. URL https://proceedings.mlr.press/v162/zhang22aa.html.

A Proof of the time-homogeneous upper bound

We first specify the algorithm parameters for both settings, then prove Theorem 3. In its proof, stage indices on shared kernels, radii, and histories are suppressed: write $\widehat{P}^{(k)}$, u_k, and $\mathcal{I}_k$. The geometry and trajectory arguments also apply in the stagewise proof.

A.1 Algorithm parameters

We use the usual measurability convention: all functions are measurable, and all transition densities and next-state factors are taken with respect to a common reference measure on $\mathcal{S}$. The MLE tolerance $\tau = \log(|\Theta|/\delta)$ applies to the pooled loss or each stagewise loss. A joint MLE over Θ^H with total tolerance τ also suffices, since the nonnegative stagewise loss gaps sum to at most τ. The fitted models and radius rules remain fixed within each iteration, and the historical multisets retain repeated visits.

We use the same confidence and radius budgets in both settings:

$$\ell_{\mathrm{emp}} = 3\log(12H|\Theta|/\delta), \qquad \ell = 10\log(12H|\Theta|/\delta), \qquad L_{\mathrm{rad}} = 240AH\log(H|\Theta|/\delta).$$

Since $H|\Theta|/\delta \geq 2$, we have $\log(12H|\Theta|/\delta) \leq 5\log(H|\Theta|/\delta)$. Together with $AH \geq 2$, this gives

$$4AH\ell + 4\ell_{\mathrm{emp}} = (40AH + 12)\log(12H|\Theta|/\delta) \leq 230AH\log(H|\Theta|/\delta) \leq L_{\mathrm{rad}}.$$

The quantities ℓ_{emp} and ℓ bound empirical and conditional model errors, respectively. The common confidence budget includes a union bound over stages; its extra $\log H$ in the homogeneous case is absorbed by $\widetilde{O}$. Set $\eta = 1/(2H)$. At iteration k, there are $(k-1)B$ historical episodes. Use $\rho_1 = 2$ and $\rho_k = \min\{2, 2AH\ell/[(k-1)B]\}$ for $k > 1$ in both settings. The remaining differences are listed in Table 2.

Table 2: Parameters that differ between the two settings. The histories are defined in (3); confidence, radius, and root budgets are common.

Quantity	Time-homogeneous	Time-inhomogeneous
Total budget Ξ	ℓ	$H\ell$
Episodes per iteration B	$\lceil A\ell \rceil$	$\lceil 2AH\ell \rceil$

Let e_{plan} denote the planning objective-error tolerance. The common stopping and numerical parameters are

$$L_0 = \log\left(e + \frac{2dAH\Xi}{\epsilon\delta}\right), \qquad K = \left\lceil \frac{10^8 H^3 d L_0^2}{\epsilon^2} \right\rceil,$$
$$\zeta = \frac{\epsilon^2}{2 \cdot 10^6 H^3}, \qquad \delta_0 = \zeta/A, \qquad \alpha = \frac{\epsilon^2}{10^4 H^2},$$
$$\Delta_{\mathrm{MC}} = 28(H-1)A\delta_0, \qquad e_{\mathrm{plan}} = \epsilon/100.$$

Allocate $\delta/(4K)$ to each PLAN call and divide this budget among its internal planning and probability tests using the deterministic call limits in Appendix E. All K calls then fail with probability at most $\delta/4$. For empty data, choose any candidate in the supplied class. The budget δ_0 is used in the conditional radius-estimation guarantee, rather than a union bound over all states or visits. The rest of this appendix treats the homogeneous case; Appendix B treats the stagewise case.

Given the fitted model and historical data at iteration k, let $\mathcal{R}_{k,h,s}$ be the joint distribution of the independent outputs of ESTIMATERADIUS at (s,a) for all actions:

$$\mathcal{R}_{k,h,s}(E) = \mathbb{P}\big((\widehat{u}_{k,h}(s,a))_{a\in\mathcal{A}} \in E\big), \qquad E \subseteq [0,2]^A \text{ measurable.} \tag{15}$$

At stages H and $H+1$, this distribution is concentrated on the zero vector. Each visit draws fresh estimates from this fixed conditional distribution.

Policies may retain within-episode history and internal randomness; their memory is initialized afresh each episode. Planning, data collection, and deployment use the same policy initialization, fitted model, radius rules, and marking rule for iteration k, with the same radius vector used for action selection and the marking coin. During collection, the policy history records the executed action, including uniform-exploration actions. Policy values and marking probabilities average over all transition, action, radius-estimation, and marking randomness.

A.2 PROOF OF LEMMA 5: HOMOGENEOUS MLE ERROR

Lemma 11 (Restatement of Lemma 5)**.** *There is an event $\mathcal{E}_{\mathrm{MLE}}$ of probability at least $1-\delta/6$ on which, at every iteration k, the homogeneous fit satisfies*

$$\sum_{i=1}^{(k-1)B}\sum_{h=1}^{H} D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_{i,h},a_{i,h}), \widehat{P}^{(k)}(\cdot \mid s_{i,h},a_{i,h})\big) \le \ell_{\mathrm{emp}}. \tag{16}$$

In the inhomogeneous case, the same bound holds for each h with $P_h^\star, \widehat{P}_h^{(k)}$ and the sum over h omitted. The event also satisfies the conditional-error bounds in Appendices A.2 and B.1.

Proof of Lemma 5, homogeneous case. We follow the likelihood-ratio proof of Proposition D.1 in Huang et al. (2026). Order the real transitions as (s_j,a_j,S_j), where S_j is the next state, and let $\mathcal{F}_j$ contain the history through (s_j,a_j), before S_j is drawn. Fix $P \in \Theta$. Note that

$$\begin{aligned} &D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_j,a_j), P(\cdot \mid s_j,a_j)\big) \\ &\quad = 2\left(1 - \mathbb{E}\left[\sqrt{\frac{P(S_j \mid s_j,a_j)}{P^\star(S_j \mid s_j,a_j)}}\,\middle|\,\mathcal{F}_j\right]\right) \\ &\quad \le -2\log \mathbb{E}\left[\sqrt{\frac{P(S_j \mid s_j,a_j)}{P^\star(S_j \mid s_j,a_j)}}\,\middle|\,\mathcal{F}_j\right], \end{aligned}$$

where the inequality follows from $1 - x \le -\log x$. Thus the sum of Hellinger errors is bounded by the sum of negative log affinities.

Suppose first that the conditional affinities are positive. Repeated conditioning gives

$$\mathbb{E}\exp\left\{\frac{1}{2}\sum_{j\le t}\log\frac{P(S_j \mid s_j,a_j)}{P^\star(S_j \mid s_j,a_j)} - \sum_{j\le t}\log\mathbb{E}\left[\sqrt{\frac{P(S_j \mid s_j,a_j)}{P^\star(S_j \mid s_j,a_j)}}\,\middle|\,\mathcal{F}_j\right]\right\}$$
$$= \mathbb{E}\prod_{j\le t}\frac{\sqrt{P(S_j \mid s_j,a_j)/P^\star(S_j \mid s_j,a_j)}}{\mathbb{E}[\sqrt{P(S_j \mid s_j,a_j)/P^\star(S_j \mid s_j,a_j)} \mid \mathcal{F}_j]} = 1.$$

Each normalized factor has conditional mean one, so the product is a nonnegative martingale. If an affinity vanishes, set the product to zero from that time onward. The resulting process is a nonnegative supermartingale in general. Moreover, the corresponding candidate has zero likelihood for that observation almost surely, so it cannot be selected by the approximate MLE on a prefix containing that observation. These statements use conditional expectation with respect to the full history and allow adaptive inputs.

Apply Ville's inequality to this process and take a union bound over $P \in \Theta$. With probability at least $1 - \delta/(12H)$, for every prefix and every candidate with positive likelihood on that prefix,

$$-2\sum_{j\le t}\log\mathbb{E}\left[\sqrt{\frac{P(S_j \mid s_j,a_j)}{P^\star(S_j \mid s_j,a_j)}}\,\middle|\,\mathcal{F}_j\right]$$
$$\le \sum_{j\le t}\log\frac{P^\star(S_j \mid s_j,a_j)}{P(S_j \mid s_j,a_j)} + 2\log(12H|\Theta|/\delta).$$

Ville's inequality gives the bound simultaneously over prefixes; a fixed-prefix bound followed by a union over $k \in [K]$ would introduce an additional $\log K$.

At the prefix containing the first $(k-1)B$ episodes, substitute $P = \widehat{P}^{(k)}$. By realizability and the MLE objective (1), the log-likelihood ratio sum is at most $\tau = \log(|\Theta|/\delta)$. Since $\tau \le \log(12H|\Theta|/\delta)$, the empirical error is at most $\tau + 2\log(12H|\Theta|/\delta) \le 3\log(12H|\Theta|/\delta)$. This proves (16) with

$$\ell_{\mathrm{emp}} = 3\log(12H|\Theta|/\delta), \qquad \ell = 2\ell_{\mathrm{emp}} + 4\log(12H|\Theta|/\delta) = 10\log(12H|\Theta|/\delta). \tag{17}$$

Conditional-error bound. We also need to bound the model error averaged over the state and action at each historical visit. We first record the bounded-observation inequality used here and later in the cumulative-error proof. For a filtration $(\mathcal{G}_j)_{j\ge 0}$ and $c > 0$, let $W_j \in [0,c]$ be $\mathcal{G}_j$-measurable. Convexity gives

$$\mathbb{E}[e^{W_j/c} \mid \mathcal{G}_{j-1}] \le \exp((e-1)\mathbb{E}[W_j \mid \mathcal{G}_{j-1}]/c)\,,$$
$$\mathbb{E}[e^{-W_j/c} \mid \mathcal{G}_{j-1}] \le \exp\bigl(-(1-e^{-1})\mathbb{E}[W_j \mid \mathcal{G}_{j-1}]/c\bigr)\,.$$

Normalize each exponential increment by its displayed upper bound and apply Ville's inequality. For $\delta' \in (0,1)$, since $e - 1 \le 2$ and $(1-e^{-1})^{-1} \le 2$, separately with probability at least $1-\delta'$ and uniformly over t,

$$\sum_{j\le t} W_j \le 2\sum_{j\le t}\mathbb{E}[W_j \mid \mathcal{G}_{j-1}] + c\log(1/\delta'),$$
$$\sum_{j\le t}\mathbb{E}[W_j \mid \mathcal{G}_{j-1}] \le 2\sum_{j\le t} W_j + 2c\log(1/\delta'). \tag{18}$$

We use separate failure budgets whenever both directions are required.

Let $\mathcal{F}^-_{i,h}$ be the history immediately before generating the stage-h state. For $h \ge 2$, it includes the actual preceding pair $(s_{i,h-1}, a_{i,h-1})$ and the preceding flag coin, but not its physical transition. At the root it contains the previous episodes and the model and policy distribution chosen for episode i. Write $\beta_{i,h}(a \mid s)$ for the behavior action probability given this history and state s, averaging over fresh radius estimates and remaining action randomization. Every such probability is at least $1/(2AH)$. The state-action density at $h \ge 2$ is therefore

$$P^\star(s \mid s_{i,h-1}, a_{i,h-1})\beta_{i,h}(a \mid s),$$

and the initial state is s_1. The conditioning times form a chronological filtration, including intervening transitions and planning randomness.

For a fixed $P \in \Theta$, apply the second inequality in (18) to $D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_{i,h}, a_{i,h}), P(\cdot \mid s_{i,h}, a_{i,h})\big) \in [0, 2]$. A union bound over candidates with failure budget $\delta/(12H)$, followed by selection of $\widehat{P}^{(k)}$, gives

$$\sum_{i=1}^{(k-1)B} \Bigg[\sum_a \beta_{i,1}(a \mid s_1) D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_1, a), \widehat{P}^{(k)}(\cdot \mid s_1, a)\big) + \sum_{h=2}^{H} \sum_{s \in \mathcal{S}} P^\star(s \mid s_{i,h-1}, a_{i,h-1}) \sum_a \beta_{i,h}(a \mid s) D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big) \Bigg] \le \ell. \tag{19}$$

These sums hold the fitted kernel fixed as a function: they do not condition retrospectively on that kernel. Uniformity over fixed candidates before fitting justifies this step, without assuming independence between $\widehat{P}^{(k)}$ and its fitting data.

Define $\mathcal{E}_{\mathrm{MLE}}$ as the intersection of the two uniform events above. A union bound gives $\mathbb{P}(\mathcal{E}_{\mathrm{MLE}}) \ge 1 - \delta/(6H) \ge 1 - \delta/6$. Both error bounds therefore hold simultaneously at every iteration. □

A.3 Proof of Lemma 12: homogeneous radii

Define $u_k^\star$ by replacing $\widehat{P}^{(k)}$ with $P^\star$ and L_{rad} with $2L_{\mathrm{rad}} + 4\ell_{\mathrm{emp}}$ in (4), using the pooled history $\mathcal{I}_k$.

Lemma 12 (One-step-back bound and uncertainty transfer)**.** *On the event $\mathcal{E}_{\mathrm{MLE}}$ of Lemma 5, for every k, (s, a), and $0 \le f \le 2$,*

$$[\widehat{P}^{(k)} f](s, a) \le 2[P^\star f](s, a) + 4D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big), \tag{20}$$

$$\mathbb{E}_{S \sim \widehat{P}^{(k)}(\cdot \mid s, a)}\Big[\max_{a'} D_{\mathrm{H}}^2\big(P^\star(\cdot \mid S, a'), \widehat{P}^{(k)}(\cdot \mid S, a')\big)\Big] \le u_k(s, a), \tag{21}$$

and

$$u_k(s, a) \le 2u_k^\star(s, a) + 4D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big). \tag{22}$$

Also $\max_a D_{\mathrm{H}}^2(P^\star(\cdot \mid s_1, a), \widehat{P}^{(k)}(\cdot \mid s_1, a)) \le \rho_k$ *for every k.*

Proof of Lemma 12, homogeneous case. We condition on $\mathcal{E}_{\mathrm{MLE}}$. As in the one-step-back bounds of Cheng et al. (2023); Huang et al. (2026), we condition on the preceding state-action pair and compare the selected action with uniform exploration. Here we control a nonnegative test function through its historical expectations.

Step 1: comparison of expectations. For nonnegative densities $p, \hat{p}$, expanding $\hat{p} = (\sqrt{p} + (\sqrt{\hat{p}} - \sqrt{p}))^2$ yields $\hat{p} \le 2p + 2(\sqrt{p} - \sqrt{\hat{p}})^2$. Thus for $0 \le g \le 2$,

$$\begin{aligned} [\widehat{P}^{(k)} g](s, a) &\le 2[P^\star g](s, a) + 4D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big), \\ [P^\star g](s, a) &\le 2[\widehat{P}^{(k)} g](s, a) + 4D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big). \end{aligned} \tag{23}$$

Step 2: next-stage and initial-state errors. For $g_k(s) = \max_a D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s, a), \widehat{P}^{(k)}(\cdot \mid s, a)\big)$, the action floor gives

$$[P^\star g_k](s_{i,h}, a_{i,h}) \le 2AH \sum_{s' \in \mathcal{S}} P^\star(s' \mid s_{i,h}, a_{i,h}) \sum_{a'} \beta_{i,h+1}(a' \mid s') \cdot D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s', a'), \widehat{P}^{(k)}(\cdot \mid s', a')\big).$$

Summing over historical predecessors and using (10) and (19),

$$\sum_{(s_{i,j}, a_{i,j}) \in \mathcal{I}_k} [\widehat{P}^{(k)} g_k](s_{i,j}, a_{i,j}) \le 4AH\ell + 4\ell_{\mathrm{emp}} \le L_{\mathrm{rad}}.$$

This proves (21). At the root, each historical episode contributes conditional risk at least $(2AH)^{-1}\max_a D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_1,a),\widehat{P}^{(k)}(\cdot \mid s_1,a)\big)$. Therefore $\max_a D_{\mathrm{H}}^2\big(P^\star(\cdot \mid s_1,a),\widehat{P}^{(k)}(\cdot \mid s_1,a)\big) \le \rho_k$.

Step 3: transfer to the true model. For every g feasible under the fitted model, the reverse inequality in (23) gives $\sum_{(s_{i,j},a_{i,j})\in\mathcal{I}_k}[P^\star g](s_{i,j},a_{i,j}) \le 2L_{\mathrm{rad}} + 4\ell_{\mathrm{emp}} = L'$. Applying the forward inequality at the queried input and then taking the supremum proves (22). The same historical multiset is used on both sides. Since $AH \ge 2$, $\ell_{\mathrm{emp}} = 3\ell/10$, and $L_{\mathrm{rad}} \le 24AH\ell$,

$$L' = 2L_{\mathrm{rad}} + 4\ell_{\mathrm{emp}} \le 50AH\ell. \tag{24}$$

□

Scalar radius formula. To obtain the scalar formula used by EstimateRadius, write $p = \widehat{P}^{(k)}(\cdot \mid s,a)$ and $p_\Sigma = \sum_{(s_{i,j},a_{i,j})\in\mathcal{I}_k}\widehat{P}^{(k)}(\cdot \mid s_{i,j},a_{i,j})$. The Lagrangian for the single constraint is

$$tL_{\mathrm{rad}} + \sum_{s'\in\mathcal{S}}\big(p(s') - tp_\Sigma(s')\big)g(s').$$

Maximizing over $0 \le g \le 2$ gives $tL_{\mathrm{rad}} + 2\sum_{s'\in\mathcal{S}}[p(s') - tp_\Sigma(s')]_+$. There is no duality gap: one can work in L_∞ for the finite measure with density $p + p_\Sigma$, where the box is weak-star compact, and use strict feasibility of $g = 0$ since $L_{\mathrm{rad}} > 0$. Equivalently, this is the binary-testing variational formula (Elkayam & Feder, 2016). The value at $t = 0$ is two, whereas $t > 2/L_{\mathrm{rad}}$ has objective greater than two. This restricts the minimization to the compact interval $[0, 2/L_{\mathrm{rad}}]$. Where $p(s') = 0$, the positive part is zero, so the expectation representation is valid even with unequal supports.

A.4 Proof of Lemma 6: both settings

Lemma 13 (Restatement of Lemma 6)**.** *On the event $\mathcal{E}_{\mathrm{MLE}}$ of Lemma 5, for every k and policy π using the implemented marking rule,*

$$D_{\mathrm{H}}^2(P^\pi,\widehat{P}^\pi) \le 18C_k(\pi) + 14\rho_k + \Delta_{\mathrm{MC}}. \tag{25}$$

The marking probability also satisfies

$$C_k(\pi) \le 8\mathbb{E}_{P^\star,\pi}\sum_{h<H}u_{k,h}(s_h,a_h) + 4HA\delta_0 + 14\mathbb{E}_{P^\star,\pi}\sum_{h\le H}D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h,a_h),\widehat{P}_h^{(k)}(\cdot \mid s_h,a_h)\big). \tag{26}$$

Proof of Lemma 6. We use the following established inequality: if P, Q are sequence laws with conditional kernels P_h, Q_h given the full preceding history $\mathcal{H}_{h-1}$, then

$$D_{\mathrm{H}}^2(P,Q) \le 7\mathbb{E}_P\sum_h D_{\mathrm{H}}^2(P_h(\cdot \mid \mathcal{H}_{h-1}),Q_h(\cdot \mid \mathcal{H}_{h-1})). \tag{27}$$

This is Lemma D.2 of Foster et al. (2024); its constant is unchanged by using our normalization of squared Hellinger distance.

Step 1: control error before the first mark. We prove the lemma for both settings, using $P_h^\star$, $\widehat{P}_h^{(k)}$, and $u_{k,h}$. In the homogeneous case these are $P^\star$, $\widehat{P}^{(k)}$, and u_k for $h < H$; in the inhomogeneous case, use the MLE event of Lemma 18. Fix k and condition on the training data and fitted models. First use the ideal marking probability $\min\{1, u_{k,h}(s,a)\}$, and write $C_k^{\mathrm{ideal}}(\pi)$ for its fitted-model probability of a mark. Until Step 3, the flags and trajectory laws use these ideal coins. Let z_h be the flag immediately before the stage-h coin. For $2 \le h \le H$,

$$\begin{aligned}
&\mathbb{E}_{\widehat{P}^\pi}[\mathbf{1}\{z_h = 0\}D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h,a_h),\widehat{P}_h^{(k)}(\cdot \mid s_h,a_h)\big)] \\
&\quad\le \mathbb{E}_{\widehat{P}^\pi}[\mathbf{1}\{z_{h-1} = 0\}(1 - \min\{1, u_{k,h-1}(s_{h-1},a_{h-1})\})u_{k,h-1}(s_{h-1},a_{h-1})] \\
&\quad\le \mathbb{E}_{\widehat{P}^\pi}[\mathbf{1}\{z_{h-1} = 0\}\min\{1, u_{k,h-1}(s_{h-1},a_{h-1})\}].
\end{aligned} \tag{28}$$

To prove the first inequality, condition on the trajectory history through action a_{h-1}, before drawing its marking coin or the next state. The fitted transition draws s_h from $\widehat{P}_{h-1}^{(k)}(\cdot \mid s_{h-1},a_{h-1})$,

independently of that coin conditional on this history. The probability that $z_h = 0$ is therefore $\mathbf{1}\{z_{h-1} = 0\}(1 - \min\{1, u_{k,h-1}(s_{h-1}, a_{h-1})\})$. The error at the selected a_h is bounded by the maximum over actions, even if a_h depends on the flag, the full history, and the fresh radius estimates. Thus the tower property gives

$$\begin{aligned}
&\mathbb{E}_{\widehat{P}^\pi}\Big[\mathbf{1}\{z_h = 0\} D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h, a_h), \widehat{P}_h^{(k)}(\cdot \mid s_h, a_h)\big)\Big] \\
&\quad \le \mathbb{E}_{\widehat{P}^\pi}\Big[\mathbf{1}\{z_{h-1} = 0\}\big(1 - \min\{1, u_{k,h-1}(s_{h-1}, a_{h-1})\}\big) \\
&\qquad \cdot \mathbb{E}_{S \sim \widehat{P}_{h-1}^{(k)}(\cdot \mid s_{h-1}, a_{h-1})}\Big[\max_{a'} D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid S, a'), \widehat{P}_h^{(k)}(\cdot \mid S, a')\big)\Big]\Big].
\end{aligned}$$

The one-step-back bound in Lemma 12, or its inhomogeneous version in Appendix B.2, applied at stage $h - 1$, bounds the inner expectation by $u_{k,h-1}(s_{h-1}, a_{h-1})$. This proves the first inequality in (28). For the second, $(1 - \min\{1, u\})u \le \min\{1, u\}$ for every $u \ge 0$: if $u \le 1$, the left side is $(1 - u)u \le u$, and if $u \ge 1$, it is zero. The last expectation in (28) equals $\mathbb{P}_{\widehat{P}^\pi}(z_{h-1} = 0, z_h = 1)$, the probability of a first mark at stage $h - 1$.

Step 2: compare trajectory laws. We next show that

$$D_{\mathrm{H}}^2(P^\pi, \widehat{P}^\pi) \le 18 C_k^{\mathrm{ideal}}(\pi) + 14\rho_k. \tag{29}$$

Fix the iteration, the radius estimation subroutine, and a policy. For an episode mixture, include the sampled policy index along with all internal variables in the trajectory. It is sufficient to retain the first $H - 1$ physical transitions and all reward-generating actions. The final physical transition is irrelevant to this value comparison, although it remains in the fitting data. For ideal flags, let R^π use fitted transitions before the first mark and true transitions from the marked transition onward. Apply (27) with R^π first and P^π second. Their conditional transition kernels differ only when the current flag coin does not mark an unmarked trajectory. On that event the prefix under R^π agrees with the fitted prefix. Thus the chain sum is

$$\sum_{h<H} \mathbb{E}_{\widehat{P}^\pi}[\mathbf{1}\{z_h = 0\}(1 - \min\{1, u_{k,h}(s_h, a_h)\}) D_{\mathrm{H}}^2\big(P_h^\star(\cdot \mid s_h, a_h), \widehat{P}_h^{(k)}(\cdot \mid s_h, a_h)\big)].$$

The root term is at most ρ_k. For $h \ge 2$, discard the factor $1 - \min\{1, u_{k,h}(s_h, a_h)\}$ and apply (28). The first-mark events are disjoint, so the sum of their probabilities is at most $C_k^{\mathrm{ideal}}(\pi)$. Hence $D_{\mathrm{H}}^2(R^\pi, P^\pi) \le 7(\rho_k + C_k^{\mathrm{ideal}}(\pi))$.

The first-mark prefix has the same law under R^π and $\widehat{P}^\pi$. Conditional on each such prefix, the subsequent squared Hellinger distance is at most two. The no-mark trajectories coincide. Consequently $D_{\mathrm{H}}^2(R^\pi, \widehat{P}^\pi) \le 2C_k^{\mathrm{ideal}}(\pi)$. Applying $(x + y)^2 \le 2x^2 + 2y^2$ to the Hellinger metric proves (29), including for policies that use the flag.

Step 3: account for radius estimation. We now use the implemented marking probability $\min\{1, \widehat{u}_{k,h}(s, a)\}$. All flags and trajectory laws below use this rule. For $h < H$, put $u_a = u_{k,h}(s, a)$ and let $\widehat{u}_a$ be the corresponding subroutine output. The relative-radius guarantee gives, at every fixed state,

$$\begin{aligned}
\mathbb{E}_{\widehat{u}} \max_a [\min\{1, u_a\} - \min\{1, \widehat{u}_a\}]_+ &\le A\delta_0, \\
\mathbb{E}_{\widehat{u}} \max_a [\min\{1, \widehat{u}_a\} - 4u_a]_+ &\le \zeta + A\delta_0.
\end{aligned}$$

The pointwise bound $[(1 - v)u - v]_+ \le 2[\min\{1, u\} - v]_+$ for $u \in [0, 2]$ and $v \in [0, 1]$, applied with $v = \min\{1, \widehat{u}_a\}$, means that the preceding chain argument incurs at most $2A\delta_0$ per predecessor before multiplication by seven. After the triangle step the additional term is at most $28(H - 1)A\delta_0$. This proves (25). Conditioning before the fresh vector $\widehat{u}_h$ is essential: the action may depend on the whole vector, but its deficit is bounded by the maximum over the finite action set.

Step 4: bound the marking probability. Let $C_P(\pi)$ denote the probability of an implemented mark under physical kernel P. For a trajectory function $F \in [0, 1]$, the density comparison gives $\mathbb{E}_{\widehat{P}} F \le 2\mathbb{E}_P F + 2D_{\mathrm{H}}^2(P, \widehat{P})$. Use the marking event for F and (27) with the true trajectory law first. Under the true law,

$$C_P(\pi) \le \mathbb{E}_{P,\pi} \sum_{h<H} \min\{1, \widehat{u}_{k,h}(s_h, a_h)\} \le 4\mathbb{E}_{P,\pi} \sum_{h<H} u_{k,h}(s_h, a_h) + H(\zeta + A\delta_0).$$

Combining these inequalities and using $\zeta = A\delta_0$ proves (26). All comparisons use the same implemented policy and radius estimation subroutine under each physical model.

□

A.5 Proof of Corollary 7: optimism

Corollary 14 (Restatement of Corollary 7)**.** *On $\mathcal{E}_{\mathrm{MLE}}$, $J_k(\pi) \geq V^{\pi}_{P^\star,r}$ for every policy π. If $J_k(\pi_k) \geq \sup_\pi J_k(\pi) - \epsilon/100$, then*

$$V^\star - V^{\pi_k}_{P^\star,r} \leq 2H\sqrt{18C_k(\pi_k) + 14\rho_k + \Delta_{\mathrm{MC}} + \alpha} + \epsilon/100.$$

Proof of Corollary 7. The total reward is a trajectory function in $[0, H]$. Thus $D_{\mathrm{TV}}(P^\pi, \widehat{P}^\pi)^2 \leq D^2_{\mathrm{H}}(P^\pi, \widehat{P}^\pi)$ and Lemma 6 imply

$$\left|V^{\pi}_{P^\star,r} - V^{\pi}_{\widehat{P}^{(k)},r}\right| \leq H\sqrt{18C_k(\pi) + 14\rho_k + \Delta_{\mathrm{MC}}}.$$

Since $\alpha > 0$, the definition of J_k gives $J_k(\pi) \geq V^{\pi}_{P^\star,r}$. Policies can ignore all auxiliary variables, so taking the supremum yields $\sup_\pi J_k(\pi) \geq V^\star$. For a policy satisfying the stated planning guarantee,

$$\begin{aligned} V^\star - V^{\pi_k}_{P^\star,r} &\leq J_k(\pi_k) - V^{\pi_k}_{P^\star,r} + \epsilon/100 \\ &\leq 2H\sqrt{18C_k(\pi_k) + 14\rho_k + \Delta_{\mathrm{MC}} + \alpha} + \epsilon/100. \end{aligned}$$

This proves the corollary in both settings, including policies that use the flag and radius estimates. □

A.6 A bound for batched linear predictions

Lemma 15 (Batched linear predictions)**.** *Suppose each batch contains at most m vectors $x_{k,t} \in \mathbb{R}^d$ of norm at most one. For each query, let $v_{k,t}$ be the supremum of $x_{k,t}^\top w$ over vectors with $\|w\| \leq R_w$, nonnegative historical predictions whose sum is at most V, and current prediction in $[0, U]$. Then, for every $\lambda > 0$,*

$$\sum_{k=1}^{K}\sum_{t} v_{k,t} \leq \frac{d}{\log 2}(V + \lambda R_w + mU)\log\left(1 + \frac{Km}{\lambda}\right). \tag{30}$$

Proof of Lemma 15. **Step 1: bound each prediction.** Let B_2 be the Euclidean unit ball in $\mathbb{R}^d$, and set $Z_k = \lambda B_2 + \sum_{i<k,t}[-x_{i,t}, x_{i,t}]$, a centrally symmetric convex body with nonempty interior. Its support function is

$$h_{Z_k}(w) = \lambda\|w\| + \sum_{i<k,t} |x_{i,t}^\top w|.$$

For every feasible w, nonnegativity of historical predictions gives $h_{Z_k}(w) \leq \lambda R_w + V$. Let $\sigma_{k,t} = \inf\{r \geq 0 : x_{k,t} \in rZ_k\}$ be the gauge of $x_{k,t}$. The support-function inequality implies $v_{k,t} \leq (V + \lambda R_w)\sigma_{k,t}$.

Step 2: bound the volume increase. For a centrally symmetric convex body Z and $x \neq 0$, project Z onto the hyperplane orthogonal to x. The lengths of fibers in direction x form an even concave function on this projection; hence the central fiber is longest. Its length is $2\|x\|/\operatorname{gauge}_Z(x)$. Integration over the projection yields

$$\operatorname{vol}(Z) \leq \frac{2\|x\|}{\operatorname{gauge}_Z(x)}\operatorname{vol}_{d-1}(\operatorname{proj}_{x^\perp} Z).$$

Adding $[-x, x]$ increases every nonempty fiber length by $2\|x\|$, so its volume increment is at least $\operatorname{gauge}_Z(x)\operatorname{vol}(Z)$. The statement for $x = 0$ is immediate; for $d = 1$ use the usual zero-dimensional volume convention. When adding a batch sequentially, projected volumes can only increase. Evaluating each projection at the iteration-start body therefore gives

$$\operatorname{vol}(Z_{k+1}) \geq \left(1 + \sum_t \sigma_{k,t}\right)\operatorname{vol}(Z_k).$$

Step 3: sum over batches. The batch cost is at most

$$\min\{mU, (V+\lambda R_w)\sum_t \sigma_{k,t}\} \le (V+\lambda R_w+mU)\min\{1, \sum_t \sigma_{k,t}\} \le \frac{V+\lambda R_w+mU}{\log 2}\log(1+\sum_t \sigma_{k,t}).$$

Finally, $Z_{K+1} \subseteq (\lambda + Km)B_2$ and $Z_1 = \lambda B_2$. Summing logarithmic volume increments proves (30). Each current query can use a different feasible coefficient vector; the proof does not assume a common maximizer.

□

A.7 Proof of Lemma 8: homogeneous cumulative uncertainty

Lemma 16 (Restatement of Lemma 8)**.** *With probability at least* $1-5\delta/12$*, Algorithm 1 satisfies*

$$B\sum_{k=1}^{K}\mathbb{E}_{P^\star,\pi_k}\left[\sum_{h<H} u_{k,h}(s_h,a_h) + \sum_{h\le H} D_{\mathrm{H}}^2\big(P_h^\star(\cdot\mid s_h,a_h), \widehat{P}_h^{(k)}(\cdot\mid s_h,a_h)\big)\right] = \widetilde{O}(dAH\Xi). \tag{31}$$

Moreover, $B\sum_{k=1}^{K}\rho_k = \widetilde{O}(AH\Xi)$.

Proof of Lemma 8, homogeneous case. **Step 1: sum predictions in the true feature space.** For $0 \le g \le 2$, define $w_g = \sum_{s'\in\mathcal{S}} \mu^\star(s')g(s')$. Then $[P^\star g](s,a) = \phi^\star(s,a)^\top w_g$ and $\|w_g\| \le 2\sqrt{d}$. Thus both $u_k^\star$ and $f_k(s,a) = \mathbb{E}_{S\sim P^\star(\cdot|s,a)}\max_{a'} D_{\mathrm{H}}^2\big(P^\star(\cdot\mid S,a'), \widehat{P}^{(k)}(\cdot\mid S,a')\big)$ are covered by Lemma 15. The latter has historical budget at most $2AH\ell$. Set

$$m = B(H-1), \qquad \lambda = (2\sqrt{d})^{-1}, \qquad G_{KB} = \log(1+2\sqrt{d}\,KB(H-1)).$$

Since $B = \lceil A\ell\rceil \le 2A\ell$, $m \le 2AH\ell$. On the MLE event, the lemma gives the deterministic bounds

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\sum_{h<H} f_k(s_{i,h},a_{i,h}) \le \frac{8dAH\ell}{\log 2}G_{KB}, \tag{32}$$

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\sum_{h<H} u_k^\star(s_{i,h},a_{i,h}) \le \frac{56dAH\ell}{\log 2}G_{KB}. \tag{33}$$

The first batch is included: its lack of history is paid for by mU. The functions in each batch depend only on the information available at its start.

Step 2: control the errors on observed state-action pairs. The root risks satisfy

$$B\sum_{k=1}^{K}\rho_k \le 2B + 2AH\ell\sum_{k=2}^{K}\frac{1}{k-1} \le 2AH\ell(2+\log K) \le 6AH\ell G_{KB}. \tag{34}$$

For a non-root observation, the expected model error before drawing its state and action is at most $f_k(s_{i,h-1},a_{i,h-1})$. The model is fixed before iteration k, so the first bound in (18) applies to the actual error sequence without a further union over candidates. With failure probability at most $\delta/8$,

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\sum_{h=1}^{H} D_{\mathrm{H}}^2\big(P^\star(\cdot\mid s_{i,h},a_{i,h}), \widehat{P}^{(k)}(\cdot\mid s_{i,h},a_{i,h})\big) \le 2B\sum_{k=1}^{K}\rho_k + \frac{16dAH\ell}{\log 2}G_{KB} + 2\log(8/\delta).$$

Together with the radius transfer and (33), this gives

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\left[\sum_{h<H}u_k(s_{i,h},a_{i,h})+\sum_{h=1}^{H}D_{\mathrm{H}}^2\big(P^\star(\cdot\mid s_{i,h},a_{i,h}),\widehat{P}^{(k)}(\cdot\mid s_{i,h},a_{i,h})\big)\right]\\
&\quad\le\frac{192dAH\ell}{\log 2}G_{KB}+10B\sum_{k=1}^{K}\rho_k+10\log(8/\delta)\\
&\quad\le 338dAH\ell G_{KB}+10\log(8/\delta).
\end{aligned}\tag{35}$$

Step 3: pass to policy expectations. Apply the second bound in (18) to the sum inside one episode, which lies in $[0,4H]$. The model and policy rule are fixed at the episode start; internal policy memory is initialized afresh, and each radius estimate uses fresh samples. With failure probability at most $\delta/8$,

$$\begin{aligned}
&B\sum_{k=1}^{K}\mathbb{E}_{P^\star,\beta_k}\left[\sum_{h<H}u_k(s_h,a_h)+\sum_{h=1}^{H}D_{\mathrm{H}}^2\big(P^\star(\cdot\mid s_h,a_h),\widehat{P}^{(k)}(\cdot\mid s_h,a_h)\big)\right]\\
&\quad\le 2\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\left[\sum_{h<H}u_k(s_{i,h},a_{i,h})+\sum_{h=1}^{H}D_{\mathrm{H}}^2\big(P^\star(\cdot\mid s_{i,h},a_{i,h}),\widehat{P}^{(k)}(\cdot\mid s_{i,h},a_{i,h})\big)\right]\\
&\qquad+8H\log(8/\delta).
\end{aligned}$$

Include the policy's internal variables, flag, and radius estimates in the trajectory. Conditional on this history, the ratio between a base-action probability p and its behavior probability is

$$\frac{p}{(1-\eta)p+\eta/A}\le\frac{1}{1-\eta+\eta/A}.$$

All other conditional kernels, including internal policy initialization and updates, are shared. The trajectory ratio is therefore at most $\kappa=(1-\eta+\eta/A)^{-H}\le 2$, and marginalizing the internal variables preserves the bound. Combining with (35), $\log(8/\delta)\le\ell$, $2AH\ge 2H$, and $G_{KB}\ge\log 3$ gives

$$B\sum_{k=1}^{K}\mathbb{E}_{P^\star,\pi_k}\left[\sum_{h<H}u_k(s_h,a_h)+\sum_{h\le H}D_{\mathrm{H}}^2\big(P^\star(\cdot\mid s_h,a_h),\widehat{P}^{(k)}(\cdot\mid s_h,a_h)\big)\right]\le 2000dAH\ell G_{KB}.\tag{36}$$

This event is proved on the chronological filtration. We do not condition retrospectively on future fitted models or future planning outputs.

The common MLE event fails with probability at most $\delta/6$, and the two additional martingale events each fail with probability at most $\delta/8$. Their intersection therefore has probability at least $1-5\delta/12$, as stated in Lemma 8. □

A.8 Proof of Theorem 3: the training budget

Theorem 17 (Restatement of Theorem 3)**.** *Under the homogeneous model and access conditions of Section 2, for $0<\epsilon\le 1$ and $0<\delta<1$, Algorithm 1 with the homogeneous configuration returns an ϵ-optimal policy with probability at least $1-\delta$ and sample complexity*

$$N=KB=\widetilde{O}\left(\frac{H^3dA}{\epsilon^2}\log\frac{2|\Theta|}{\delta}\right).\tag{37}$$

Proof of Theorem 3. Use (17) and set $\eta=1/(2H)$, $B=\lceil A\ell\rceil$. Define

$$\begin{aligned}
&\zeta=\frac{\epsilon^2}{2\cdot 10^6H^3},\qquad \delta_0=\frac{\epsilon^2}{2\cdot 10^6AH^3},\qquad \alpha=\frac{\epsilon^2}{10^4H^2},\qquad e_{\mathrm{plan}}=\epsilon/100,\\
&L_0=\log\left(e+\frac{2dAH\ell}{\epsilon\delta}\right),\qquad K=\left\lceil\frac{10^8H^3dL_0^2}{\epsilon^2}\right\rceil.
\end{aligned}$$

These quantities are chosen before training and contain no implicit dependence on K. Allocate at most $\delta/4$ to all internal planning and probability-test failures, using the deterministic call limits in Appendix E. The per-query δ_0 is a numerical parameter handled by the distribution of the radius estimates, not an event to union bound over all visits.

Corollary 7, followed by averaging over the K iteration policies (equivalently, the KB episodes), gives

$$V^\star - V^{\bar{\pi}}_{P^\star,r} \le 2H\sqrt{\frac{506000dAH\ell G_{KB}}{KB}} + 0.04\epsilon.$$

Indeed, the root coefficient is 14, and (26) gives at most $18 \cdot 14 = 252$ times the cumulative radius and error in (36). Equations (34) and (36) therefore give the displayed constant; the numerical terms from ζ, δ_0, α and planning contribute less than 0.04ϵ. Substitution of the displayed iteration count gives $G_{KB} \le 40L_0$ and $2AH\ell/B \le 2H$. Also $L_0 \ge 2$. The statistical term is at most 0.64ϵ, so the full error is below ϵ. The two MLE events, two real-risk martingale events, and planning events have total failure probability at most $\delta/6 + \delta/4 + \delta/4 < \delta$. This proves Theorem 3. The initial empty-data iteration may skip MLE, giving at most $K-1$ nonempty calls. Every iteration collects exactly B episodes, so the output weights are $1/K$.

□

B Proof of the time-inhomogeneous upper bound

We prove Theorem 4 for the inhomogeneous configuration of Algorithm 1. The model class and MLE interface are defined in Section 2, and the common budgets $\ell_{\mathrm{emp}}, \ell, L_{\mathrm{rad}}$ are specified in Appendix A.1. The number of episodes per iteration is

$$B = \lceil 2AH\ell \rceil. \tag{38}$$

B.1 Proof of Lemma 5: inhomogeneous MLE error

Lemma 18 (Restatement of Lemma 5)**.** *There is an event $\mathcal{E}_{\mathrm{MLE}}$ of probability at least $1-\delta/6$ on which, at every iteration k, the homogeneous fit satisfies*

$$\sum_{i=1}^{(k-1)B}\sum_{h=1}^{H} D^2_{\mathrm{H}}\big(P^\star(\cdot \mid s_{i,h},a_{i,h}), \widehat{P}^{(k)}(\cdot \mid s_{i,h},a_{i,h})\big) \le \ell_{\mathrm{emp}}. \tag{39}$$

In the inhomogeneous case, the same bound holds for each h with $P^\star_h, \widehat{P}^{(k)}_h$ and the sum over h omitted. The event also satisfies the conditional-error bounds in Appendices A.2 and B.1.

Proof of Lemma 5, inhomogeneous case. For each h, apply the homogeneous likelihood-ratio argument to the stage-h subsequence: bound Hellinger distance by negative log affinity, normalize the square-root likelihood ratio, and substitute $\widehat{P}^{(k)}_h$ using the stagewise MLE objective (2). Apply (18) to obtain the conditional-error bound. With failure probability at most $\delta/(6H)$, simultaneously for every k,

$$\sum_{i=1}^{(k-1)B} D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s_{i,h},a_{i,h}), \widehat{P}^{(k)}_h(\cdot \mid s_{i,h},a_{i,h})\big) \le \ell_{\mathrm{emp}},$$

$$\sum_{i=1}^{(k-1)B}\sum_{s\in\mathcal{S}} P^\star_{h-1}(s \mid s_{i,h-1},a_{i,h-1}) \sum_a \beta_{i,h}(a \mid s) D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s,a), \widehat{P}^{(k)}_h(\cdot \mid s,a)\big) \le \ell, \quad h \ge 2.$$

At $h=1$, replace the sum over the state reached by the preceding transition by evaluation at s_1. The behavior probabilities and conditioning times are those of (19). The events are uniform over candidates before fitting; the sums keep the selected kernels fixed as functions.

Define $\mathcal{E}_{\mathrm{MLE}}$ as the intersection of these events over $h \in [H]$. A union bound gives $\mathbb{P}(\mathcal{E}_{\mathrm{MLE}}) \ge 1-\delta/6$. □

B.2 Inhomogeneous version of Lemma 12: radii

Lemma 19 (Inhomogeneous version of Lemma 12). *Define $u^\star_{k,h}$ by replacing $\widehat{P}^{(k)}_h$ with $P^\star_h$ and L_{rad} with $2L_{\mathrm{rad}} + 4\ell_{\mathrm{emp}}$ in (4), using the stage-h history. On the event $\mathcal{E}_{\mathrm{MLE}}$ of Lemma 18, for every k, $h < H$, (s,a), and $0 \le f \le 2$,*

$$[\widehat{P}^{(k)}_h f](s,a) \le 2[P^\star_h f](s,a) + 4D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s,a), \widehat{P}^{(k)}_h(\cdot \mid s,a)\big), \tag{40}$$

$$\mathbb{E}_{S\sim\widehat{P}^{(k)}_h(\cdot|s,a)}\Big[\max_{a'} D^2_{\mathrm{H}}\big(P^\star_{h+1}(\cdot \mid S,a'), \widehat{P}^{(k)}_{h+1}(\cdot \mid S,a')\big)\Big] \le u_{k,h}(s,a), \tag{41}$$

and

$$u_{k,h}(s,a) \le 2u^\star_{k,h}(s,a) + 4D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s,a), \widehat{P}^{(k)}_h(\cdot \mid s,a)\big). \tag{42}$$

Also $\max_a D^2_{\mathrm{H}}(P^\star_1(\cdot \mid s_1,a), \widehat{P}^{(k)}_1(\cdot \mid s_1,a)) \le \rho_k$ for every k.

Proof of Lemma 19. **Step 1: control the next-stage error.** For $h < H$, define

$$u_{k,h}(s,a) = \sup_{0\le g\le 2}\left\{[\widehat{P}^{(k)}_h g](s,a) : \sum_{i\le(k-1)B}[\widehat{P}^{(k)}_h g](s_{i,h},a_{i,h}) \le L_{\mathrm{rad}}\right\}. \tag{43}$$

The historical density sum uses only stage-h predecessor inputs. The conditional-risk budget at stage $h+1$ controls the next-state error $g_{k,h+1}(s) = \max_a D^2_{\mathrm{H}}\big(P^\star_{h+1}(\cdot \mid s,a), \widehat{P}^{(k)}_{h+1}(\cdot \mid s,a)\big)$. The action floor gives its true historical budget at most $2AH\ell$, and positive-function transfer with $L_{\mathrm{rad}} \ge 4AH\ell + 4\ell_{\mathrm{emp}}$ proves

$$\mathbb{E}_{S\sim\widehat{P}^{(k)}_h(\cdot|s,a)}\Big[\max_{a'} D^2_{\mathrm{H}}\big(P^\star_{h+1}(\cdot \mid S,a'), \widehat{P}^{(k)}_{h+1}(\cdot \mid S,a')\big)\Big] \le u_{k,h}(s,a). \tag{44}$$

The scalar radius dual and relative estimator are unchanged, with the stage-specific kernel and density sum and the common budget.

Step 2: transfer to the true features at stage h. Let $u^\star_{k,h}$ be the corresponding true-kernel radius with budget $L' = 2L_{\mathrm{rad}} + 4\ell_{\mathrm{emp}} \le 50AH\ell$. The same feasible-function argument gives

$$u_{k,h}(s,a) \le 2u^\star_{k,h}(s,a) + 4D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s,a), \widehat{P}^{(k)}_h(\cdot \mid s,a)\big), \qquad (H-1)L' \le 50AH^2\ell.$$

Also $f_{k,h}(s,a) = \mathbb{E}_{S\sim P^\star_h(\cdot|s,a)} \max_{a'} D^2_{\mathrm{H}}\big(P^\star_{h+1}(\cdot \mid S,a'), \widehat{P}^{(k)}_{h+1}(\cdot \mid S,a')\big)$ has historical budget at most $2AH\ell$. Each $f_{k,h}$ and each direction defining $u^\star_{k,h}$ is linear in the fixed true feature map $\phi^\star_h$, with coefficient norm at most $2\sqrt{d}$. Different stages use different true spaces; predictions are accumulated within their own stage before summing over stages.

Step 3: control the initial-state error. At the initial state, each historical episode contributes conditional risk at least $(2AH)^{-1}\max_a D^2_{\mathrm{H}}(P^\star_1(\cdot \mid s_1,a), \widehat{P}^{(k)}_1(\cdot \mid s_1,a))$. The conditional budget ℓ therefore gives the stated bound, with $\rho_1 = 2$ for empty data. The pointwise comparison (40) applies to each stage without any probability event. □

B.3 Proof of Lemma 8: inhomogeneous cumulative uncertainty

Lemma 20 (Restatement of Lemma 8). *With probability at least $1 - 5\delta/12$, Algorithm 1 satisfies*

$$B\sum_{k=1}^{K}\mathbb{E}_{P^\star,\pi_k}\left[\sum_{h<H} u_{k,h}(s_h,a_h) + \sum_{h\le H} D^2_{\mathrm{H}}\big(P^\star_h(\cdot \mid s_h,a_h), \widehat{P}^{(k)}_h(\cdot \mid s_h,a_h)\big)\right] = \widetilde{O}(dAH\Xi). \tag{45}$$

Moreover, $B\sum_{k=1}^{K}\rho_k = \widetilde{O}(AH\Xi)$.

Proof of Lemma 8, inhomogeneous case. **Step 1: sum predictions separately at each stage.** Set $G_{KB} = \log(1 + 2\sqrt{d}\,KB)$. Apply Lemma 15 separately at each stage with $m = B$, since each

episode contributes only one input to a given stage. By (38), $HB \le 4AH^2\ell$. Summing the geometric bounds yields

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\sum_{h<H} f_{k,h}(s_{i,h},a_{i,h}) \le \frac{d}{\log 2}\left(2AH^2\ell + H + 2HB\right) G_{KB} \le \frac{12dAH^2\ell}{\log 2} G_{KB},$$

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\sum_{h<H} u^{\star}_{k,h}(s_{i,h},a_{i,h}) \le \frac{d}{\log 2}\left(50AH^2\ell + H + 2HB\right) G_{KB} \le \frac{60dAH^2\ell}{\log 2} G_{KB}.$$

Step 2: bound the observed errors. Use root risk $\rho_1 = 2$ and $\rho_k = \min\{2, 2AH\ell/[(k-1)B]\}$ thereafter. Its weighted sum is at most $2B + 2AH\ell(1+\log K) \le 8AH^2\ell G_{KB}$. The same martingale argument controls observed errors: the conditional risk at stage $h+1$ is at most $f_{k,h}(s_{i,h},a_{i,h})$. Combining this bound with radius transfer gives

$$\sum_{k=1}^{K}\sum_{i=(k-1)B+1}^{kB}\left[\sum_{h<H} u_{k,h}(s_{i,h},a_{i,h}) + \sum_{h=1}^{H} D^2_{\mathrm{H}}\big(P^{\star}_h(\cdot \mid s_{i,h},a_{i,h}), \widehat{P}^{(k)}_h(\cdot \mid s_{i,h},a_{i,h})\big)\right]$$
$$\le 428dAH^2\ell G_{KB} + 10\log(8/\delta).$$

Step 3: pass to policy expectations. Apply the episode-level martingale to this per-episode sum in $[0, 4H]$ and use the same trajectory likelihood-ratio bound $\kappa \le 2$. Absorbing the logarithmic remainders into $H\ell$ gives

$$B\sum_{k=1}^{K}\mathbb{E}_{P^{\star},\pi_k}\left[\sum_{h<H} u_{k,h}(s_h,a_h) + \sum_{h\le H} D^2_{\mathrm{H}}\big(P^{\star}_h(\cdot \mid s_h,a_h), \widehat{P}^{(k)}_h(\cdot \mid s_h,a_h)\big)\right] \le 2000dAH^2\ell G_{KB}. \tag{46}$$

Together with the root sum above, this proves the inhomogeneous case of Lemma 8. The MLE event has failure probability at most $\delta/6$, and the two real-risk martingale events have total failure probability at most $\delta/4$, giving the stated $5\delta/12$ bound. □

B.4 Proof of Theorem 4: the training budget

Theorem 21 (Restatement of Theorem 4)**.** *Under the inhomogeneous model and access conditions of Section 2, for $0 < \epsilon \le 1$ and $0 < \delta < 1$, Algorithm 1 with the inhomogeneous configuration returns an ϵ-optimal policy with probability at least $1-\delta$ and sample complexity*

$$N = KB = \widetilde{O}\left(\frac{H^4 dA}{\epsilon^2}\log\frac{H|\Theta|}{\delta}\right) \tag{47}$$

Proof of Theorem 4. Lemma 6 and Corollary 7, proved in Appendices A.4 and A.5, apply to the inhomogeneous setting using the MLE event of Lemma 18. The first-mark argument uses (44). Generate fresh radius estimates from the stage-specific distribution $\mathcal{R}_{k,h,s}$ defined in (15) and keep the same numerical budgets ζ, δ_0, α and planning error as in Appendix A.8. The augmented additive-reward model has dimension $2d$ at each stage: its next-state factors use $\widehat{\mu}^{(k)}_h(s')\mathcal{R}_{k,h+1,s'}(d\widehat{u}')$. The LSVI proof permits stage-dependent features and measures, so its simulation order and scalar planning guarantee remain valid.

Combining Corollary 7 with (46) and (26) gives

$$V^{\star} - V^{\bar{\pi}}_{P^{\star},r} \le 2H\sqrt{\frac{506000dAH^2\ell G_{KB}}{KB}} + 0.04\epsilon.$$

An explicit sufficient choice is

$$L_0 = \log\left(e + \frac{2dAH^2\ell}{\epsilon\delta}\right), \qquad K = \left\lceil \frac{10^8 H^3 d L_0^2}{\epsilon^2}\right\rceil.$$

Since $2AH^2\ell/B \le H$ and $G_{KB} \le 40L_0$, the statistical term is below 0.64ϵ. The combined MLE, real-risk, and planning failure budgets remain below δ. This proves Theorem 4.

□

C Proof of the time-homogeneous lower bound

C.1 Parameter range

Theorem 9 uses the following parameter range. Let $d \geq 4$, $A \geq 2$, and $J = 2^t$ be the largest power of two at most $d/2$. Require $H \geq 2t + 2$ and $R = H - t - 1 \geq 128$, and define

$$n = \lfloor R/128 \rfloor, \qquad F = Jn(A - 1).$$

For every integer $m \geq 128 + 32 \log_2 F$, the class has size $|\Theta| = F2^m + F - 1$. The accuracy and confidence ranges are $0 < \epsilon \leq 1$ and $0 < \delta \leq 1/16$. These are the complete parameter conditions for the theorem.

C.2 Proof of Theorem 9

Theorem 22 (Restatement of Theorem 9)**.** *Let $d \geq 4$, $A \geq 2$, $0 < \epsilon \leq 1$, and $0 < \delta \leq 1/16$. For the horizons and class sizes specified in Appendix C.1, there exists a finite class Θ of time-homogeneous kernels with rank at most d such that every algorithm that is (ϵ, δ)-PAC on the class requires*

$$N = \Omega\left(\frac{H^3 dA}{\epsilon^2} \left[\log |\Theta| + \log \frac{1}{\delta} \right] \right) \tag{48}$$

real episodes in the worst case.

Proof of Theorem 9. We construct a finite class, show what an ϵ-optimal policy must identify, and bound the information in the learner's transcript. The entropy and confidence arguments use the same class.

Step 1: construct and normalize the model class. Use the parameters above. There are $J = 2^t \geq 2$ branches, $G = A - 1$ query actions and one continue action. A full binary tree has t action levels before the first live state. At each internal node assign at least one action to each child and make additional actions aliases. Write $\mathbb{F}_2 = \{0, 1\}$ for the binary field, with parity inner products computed modulo two. The last tree action selects branch j and draws (i, x) uniformly from $[n] \times \mathbb{F}_2^m$. At a live state (j, i, x), continue draws a new pair uniformly within branch j. Every query moves immediately to good or bad, both absorbing. Set the signal amplitude to $\Delta = 16\epsilon/R \leq 1/8$.

For target $v^\star = (j^\star, i^\star, g^\star)$ and parity $\theta \in \mathbb{F}_2^m$, the query probability is

$$P(\text{good} \mid j, i, x, g) = \tfrac{1}{2} + \Delta \mathbf{1}\{(j, i, g) = v^\star\}(-1)^{\theta^\top x}. \tag{49}$$

Rewards are $1/2$ in the tree and live states, one in good, and zero in bad, for every action and stage. Querying at the first live state leaves $R = H - t - 1$ subsequent reward steps. The final transition is also observed during training. A query at this last step may supply information even though it no longer changes the episode value; the information argument counts it. There is still at most one query per episode.

Take the right-factor coordinates to be the $J - 2$ non-root internal-node point masses, J uniform branch distributions, and two absorbing point masses. Their supports are disjoint, each distribution occurs as an actual row of the kernel, and every other row is a convex combination of them. Thus the transition rank is exactly $d_0 = 2J$, every left factor has norm at most one, and summing a function in $[0, 1]$ against the right factor gives a vector in $[0, 1]^{d_0}$, of norm at most $\sqrt{d_0}$. These right factors are common to all models, including the alternatives introduced below. Padding their factorization with zero coordinates embeds them in dimension d. If exact rank d is desired, add $d - d_0$ unreachable independent absorbing states with fixed rewards, along with their point-mass factors. This preserves all learning arguments and the normalization.

The initial class has one model for every pair (v, θ) with $v = (j, i, g) \in [J] \times [n] \times [G]$ and $\theta \in \mathbb{F}_2^m$. These are $F2^m$ distinct kernels. This is a proper, realizable finite class, and its description is supplied to the learner. Giving the common right factor as additional information only makes the lower bound stronger.

Step 2: identify a necessary condition for near-optimality. In both lower-bound proofs, write $V^\pi = V^\pi_{P^\star,r}$ for the current true model and $\widehat{\pi}$ for the learner's output. For a fixed, possibly randomized and history-dependent policy π, define $p_v(\pi)$ as its probability of making its query at target triple v. This probability averages over tree decisions, fresh contexts, internal randomization, and episode-mixture components. All transitions before querying are identical in all models, so p_v is a model-independent functional of the policy. Also $\sum_v p_v(\pi) \le 1$, since an episode can contain at most one query. The same definition applies to a fixed realized output of any learning algorithm.

Under the zero-signal reference, the value of every policy is exactly $H/2$. Indeed, conditioning on a query time gives expected future reward $1/2$ per remaining step, while continuing also pays $1/2$. In a parity model, if the query occurs on the ℓth live visit, its additional expected value is

$$(R+1-\ell)\Delta \mathbf{1}\{v = v^\star\}(-1)^{\theta^\top x}.$$

Here $1 \le \ell \le R+1$ for a query within the horizon. Thus any policy's gain is at most $R\Delta p_{v^\star}(\pi)$.

A policy knowing the model first selects the correct branch. It waits during the first $w = \lfloor R/8 \rfloor$ live visits for $i = i^\star$ and $(-1)^{\theta^\top x} = 1$, querying $g^\star$ at the first match; otherwise it keeps continuing. A nonzero parity is positive on half the inputs, and the zero parity is always positive. The probability of a match at each visit is at least $1/(2n)$. Since $w/(2n) \ge 4$ and $R+1-w \ge 7R/8$, this policy gains at least

$$R\Delta\frac{7}{8}(1-e^{-4}) \ge \frac{17}{20}R\Delta.$$

An ϵ-optimal policy therefore satisfies

$$p_{v^\star}(\pi) \ge \frac{17}{20} - \frac{\epsilon}{R\Delta} = \frac{63}{80} > \frac{3}{4}. \tag{50}$$

For a learning algorithm with failure probability at most $\delta \le 1/16$, this implies, under every prior on the parity class,

$$\mathbb{E}p_{v^\star}(\widehat{\pi}) \ge (1-\delta)\frac{3}{4} \ge \frac{45}{64}. \tag{51}$$

This argument does not require a deterministic or Markov output policy.

Step 3: bound the stopped likelihood ratio. Suppose an algorithm uses at most N real episodes. If it stops earlier, pad its transcript with an independent fixed behavior up to N; the output remains unchanged. Let $\mathbb{P}_0$ be the data law when every query outcome is $\mathrm{Bernoulli}(1/2)$; this reference need not lie in the class, since no PAC guarantee will be applied to it. Fix a target $v = (j,i,g)$, and mix the data laws uniformly over all $M_{\mathrm{par}} = 2^m$ parities at this target. For an integer query cap $b_{\mathrm{cap}} \ge 1$, stop each experiment after its b_{cap}th query at v, or at the end of training, whichever comes first. The stopping rule is the same function of the observed transcript under every law. Denote the parity-mixture law by $\overline{\mathbb{P}}_v$, its stopped law by $\overline{\mathbb{P}}_v^{\mathrm{cap}}$, and the stopped reference law by $\mathbb{P}_0^{\mathrm{cap}}$. Expectations carry the same subscripts as their probability laws.

Let L_θ be the likelihood ratio of a component law to the reference, on the stopped transcript up to the current time, and $\bar{L} = M_{\mathrm{par}}^{-1}\sum_\theta L_\theta$. At a query at v, if the outcome is $S \in \{\mathrm{bad}, \mathrm{good}\}$, then

$$L_\theta \;\longleftarrow\; L_\theta(1 + 2\Delta U \chi_\theta(x)), \qquad U = 2\mathbf{1}\{S = \mathrm{good}\} - 1, \quad \chi_\theta(x) = (-1)^{\theta^\top x}.$$

Conditionally on the history, current context, and selected action, U is a uniform sign under $\mathbb{P}_0$. Non-target queries and all continue transitions leave the likelihood ratios unchanged. If N_v denotes the stopped target-query count, then

$$\frac{L_\theta^2}{(1+4\Delta^2)^{N_v}} \text{ is a nonnegative martingale}, \qquad \mathbb{E}_0 L_\theta^2 \le (1+4\Delta^2)^{b_{\mathrm{cap}}}. \tag{52}$$

The expectation bound holds at every bounded transcript time: $N_v \le b_{\mathrm{cap}}$ and the normalized martingale has mean one.

Now condition immediately before a fresh live context is generated. At this point the current branch and all L_θ are fixed. The query decision may depend on the new (i', x) and on private randomness.

For the increment $\Delta\bar{L}$ caused by the subsequent query, or zero if no target query occurs,

$$\mathbb{E}_0[(\Delta\bar{L})^2 \mid \text{history before context}] \le 4\Delta^2\mathbb{E}_{i',x}\left[\mathbf{1}\{i'=i\}\left(\frac{1}{M_{\text{par}}}\sum_\theta L_\theta\chi_\theta(x)\right)^2\right]$$
$$= \frac{4\Delta^2}{nM_{\text{par}}^2}\sum_\theta L_\theta^2. \tag{53}$$

The inequality discards the query-selection indicator from a nonnegative square. The equality uses $\mathbb{E}_x\chi_\theta(x)\chi_{\theta'}(x) = \mathbf{1}\{\theta=\theta'\}$. If the branch is not j, the episode is no longer live, or the experiment has stopped, the increment is zero. Thus the argument allows arbitrary selection of query inputs by waiting; it never treats selected inputs as independent uniform draws.

For probability laws $P \ll Q$, let $\chi^2(P\|Q) = \mathbb{E}_Q[(dP/dQ-1)^2]$. There are at most NH context opportunities. Index them by episode and step, padding missing opportunities with zero increments. The increments of $\bar{L}$ are orthogonal in $L_2(\mathbb{P}_0)$. Combining (52) and (53) proves

$$\chi^2(\overline{\mathbb{P}}_v^{\text{cap}}\|\mathbb{P}_0^{\text{cap}}) = \mathbb{E}_0(\bar{L}-1)^2 \le \frac{4\Delta^2 NH}{n2^m}(1+4\Delta^2)^{b_{\text{cap}}}. \tag{54}$$

Since $n = \lfloor R/128\rfloor \ge R/256$ and $R \ge H/2$, $H/n \le 512$. Choose

$$b_{\text{cap}} = \left\lfloor \frac{m\log 2}{16\Delta^2}\right\rfloor, \qquad N \le Fb_{\text{cap}}/8.$$

Then $\log(1+4\Delta^2) \le 4\Delta^2$ gives

$$\chi^2(\overline{\mathbb{P}}_v^{\text{cap}}\|\mathbb{P}_0^{\text{cap}}) \le 16Fm\log 2\; 2^{-3m/4} \le (m\log 2)\,2^{-23m/32} < 1/4096.$$

For the second inequality, $m \ge 128 + 32\log_2 F$ implies $16F \le 2^{m/32}$. The last inequality holds at $m = 128$ and the expression decreases thereafter. By Cauchy–Schwarz, the stopped-law total variation distance v_{TV} is at most $1/128$.

Step 4: count informative queries. Let N_v^{full} be the target-query count before any artificial truncation. The event $T_v = \{N_v^{\text{full}} \ge b_{\text{cap}}\}$ is measurable on the stopped transcript. So is $p_v(\widehat{\pi})\mathbf{1}\{N_v^{\text{full}} < b_{\text{cap}}\}$: if the cap is hit set this variable to zero; otherwise training and output have completed. The stopped and full experiments agree up to the stopping time. Two applications of total variation comparison give

$$\mathbb{E}_{\overline{\mathbb{P}}_v} p_v(\widehat{\pi}) \le \mathbb{E}_{\overline{\mathbb{P}}_v^{\text{cap}}}[p_v(\widehat{\pi})\mathbf{1}\{N_v^{\text{full}} < b_{\text{cap}}\}] + \overline{\mathbb{P}}_v^{\text{cap}}(T_v)$$
$$\le \mathbb{E}_0 p_v(\widehat{\pi}) + \mathbb{P}_0(T_v) + 2v_{\text{TV}}.$$

Since every output has $\sum_v p_v \le 1$, and every training transcript has $\sum_v N_v^{\text{full}} \le N$, averaging over the $F \ge 2$ targets yields

$$\frac{1}{F}\sum_v \mathbb{E}_{\overline{\mathbb{P}}_v} p_v(\widehat{\pi}) \le \frac{1}{F} + \frac{N}{Fb_{\text{cap}}} + 2v_{\text{TV}} \le \frac{1}{2} + \frac{1}{8} + \frac{1}{64} = \frac{41}{64}.$$

This contradicts (51). Hence

$$N > \frac{Fb_{\text{cap}}}{8} \ge \frac{FmR^2\log 2}{65536\epsilon^2}. \tag{55}$$

The floor inequality uses $m\log 2/(16\Delta^2) \ge 2$, which follows from $m \ge 128$ and $\Delta \le 1/8$.

Step 5: prove the confidence lower bound. Fix one target v_0. The baseline model is already in the parity class: it has a constant positive signal of amplitude Δ at v_0, corresponding to $\theta = 0$. For each $v \ne v_0$, add an alternative that keeps this signal and has a constant positive signal of amplitude 2Δ at v. There are $F-1$ distinct new models. They share the same transition directions and remain proper because $2\Delta \le 1/4$. The resulting class has size

$$|\Theta| = F2^m + F - 1. \tag{56}$$

Baseline PAC success requires $p_{v_0} \geq 3/4$. In the alternative at v, a policy that waits for v obtains gain at least $(17/20)2R\Delta$. Any policy's gain is at most $R\Delta(p_{v_0} + 2p_v) \leq R\Delta(1 + p_v)$. Thus alternative PAC success requires

$$p_v \geq \frac{17}{10} - 1 - \frac{1}{16} > \frac{1}{2}.$$

For $E_v = \{p_v(\widehat{\pi}) \geq 1/2\}$, the baseline probability is at most δ, and the alternative probability is at least $1 - \delta$. These are disjoint success requirements because $\sum_v p_v \leq 1$.

Write $\mathbb{P}_{\text{base}}$ and $\mathbb{P}_v$ for the full training laws under the baseline and alternative at v, and let N_v count all training queries to v. Let $\mathrm{kl}(u\|v) = u\log(u/v) + (1-u)\log((1-u)/(1-v))$ be Bernoulli relative entropy. Only a query at v distinguishes these laws. Its conditional divergence is

$$\mathrm{kl}(1/2\|1/2 + 2\Delta) = -\tfrac{1}{2}\log(1 - 16\Delta^2) \leq \frac{32}{3}\Delta^2,$$

where $16\Delta^2 \leq 1/4$. The KL chain rule for an adaptive transcript and data processing to E_v therefore imply

$$\frac{32}{3}\Delta^2 \mathbb{E}_{\text{base}} N_v \geq \mathrm{kl}(\mathbb{P}_{\text{base}}(E_v), \mathbb{P}_v(E_v)) \geq (1-\delta)\log(1/\delta) - \log 2 \geq \tfrac{1}{2}\log(1/\delta).$$

The middle inequality uses the binary-entropy bound and the baseline failure probability of E_v. Summing over $v \neq v_0$ and using $\sum_v N_v \leq N$ under the same baseline gives

$$N \geq \frac{3(F-1)}{64\Delta^2}\log(1/\delta) = \Omega(F\Delta^{-2}\log(1/\delta)). \tag{57}$$

The enlarged class still contains every parity instance, so both (55) and (57) apply to any algorithm correct on that one class. Their maximum is at least half their sum. Since $J = \Theta(d)$, $n = \Theta(H)$, $G = \Theta(A)$, $R = \Theta(H)$, and $\log(F2^m + F - 1) = \Theta(m)$, Theorem 9 follows.

Fitted-model samples, model-learning outputs, and private computation can be included in the transcript. Conditional on real data and private randomness, they are generated by the same algorithmic rules under every true model and supply no additional environmental information. The lower bound is therefore information-theoretic, and does not invoke a computational hardness assumption. □

C.3 Interpretation of the parameter range

The theorem is stated for the explicit sequence of class sizes (56). Since increasing m by one changes this size by a constant factor, it also describes entropy budgets up to constant factors in the displayed range. It is not a statement that adding duplicate or uninformative candidates makes a fixed task harder. The restrictions $R \geq 128$, $H \geq 2\log_2 J + 2$, and $m \geq 128 + 32\log_2 F$ ensure enough time to wait and enough parity entropy relative to the number of targets. The conclusion covers rank bounds $d \geq 4$ and action sets $A \geq 2$, not all boundary cases.

The extra horizon factor is obtained without adding a clock to the state or using a separate unknown parameter at each stage. A live branch always refreshes from one fixed distribution. Its $\Theta(H)$ labels create distinct query targets while preserving rank. Unlike a one-context construction, an informed policy can wait and still leave $\Theta(H)$ rewards after querying. Unlike observing many independent informative transitions, training still obtains at most one query outcome per episode. This separates the number of free contexts from the number of useful feedback events. It is also why the construction differs from the stage-dependent hard instances of Cheng et al. (2023); Domingues et al. (2021).

D Proof of the time-inhomogeneous lower bound

The shared-kernel lower bound does not by itself account for the sum of stagewise entropies. We give a product class with independent hidden parameters at successive decision stages. The resulting lower bound matches Theorem 4 up to logarithmic factors. We retain the explicit confidence dependence to distinguish this matching rate from a sharper characterization of the confidence term.

D.1 Parameter range

Let $d \geq 5$, $A \geq 2$, and $J = 2^t$ be the largest power of two at most $(d-1)/2$. Set

$$C = t + 1, \qquad L = \left\lfloor \frac{H}{4C} \right\rfloor, \qquad n = \lfloor L/16 \rfloor, \qquad F = Jn(A-1).$$

Require $H \geq 512C$ and an integer $m \geq 256 + 32\log_2(HF)$. We construct a common kernel class Θ of size $|\Theta| = F(2^m + 1)$. The hard instances form a subfamily of Θ^H: only L stages vary, while all other stages use one fixed kernel. This subfamily has size $|\Theta|^L$. For every $0 < \epsilon \leq 1$ and $0 < \delta \leq 1/16$, we prove the more explicit bound

$$N = \Omega\left(\frac{H^2 F}{\epsilon^2}\left[nm + \log(1/\delta)\right]\right) \tag{58}$$

Since $C = \Theta(1 + \log d)$, this implies Theorem 10. These are the complete parameter conditions for the theorem.

D.2 Proof of Theorem 10

Theorem 23 (Restatement of Theorem 10)**.** *Let $d \geq 5$, $A \geq 2$, $0 < \epsilon \leq 1$, and $0 < \delta \leq 1/16$. For the horizons and class sizes specified in Appendix D.1, there exists a finite class Θ of kernels with rank at most d such that every algorithm that is (ϵ, δ)-PAC on Θ^H requires*

$$N = \Omega\left(\frac{H^4 dA \log|\Theta|}{(1+\log d)^2\epsilon^2} + \frac{H^3 dA \log(1/\delta)}{(1+\log d)\epsilon^2}\right) \tag{59}$$

real episodes in the worst case.

Proof of Theorem 10. We use independent hidden parameters at the active stages. We first relate value to prediction at each stage, then sum the information requirements under one product prior.

Step 1: construct a model class of rank at most d. Use a complete binary tree of depth t, with its J leaves replaced by live-state distributions. The tree actions select a branch $j \in [J]$. On the last tree transition, draw a live state (j, i, x) with i uniform on $[n]$ and x uniform on $\mathbb{F}_2^m$. At a live state, one action returns to the root, and the other $G = A - 1$ actions query and enter good or bad, both absorbing. Thus, until the first query, the live stages are exactly

$$h_\ell = \ell C, \qquad \ell = 1, 2, \ldots.$$

An episode has at most one query, including a query on the final observed transition. All rewards are $1/2$ before absorption, one in good, and zero in bad.

Set $\Delta = 32\epsilon/H \leq 1/32$. The zero-signal kernel P^0 makes every query outcome $\mathrm{Bernoulli}(1/2)$. For each $v = (j, i, g) \in [J] \times [n] \times [G]$ and $\theta \in \mathbb{F}_2^m$, define a parity kernel whose query probability is

$$P^{v,\theta}(\text{good} \mid j', i', x, g') = \tfrac{1}{2} + \Delta \mathbf{1}\{(j', i', g') = v\}(-1)^{\theta^\top x}.$$

Fix one target v_0. Also include, for every $v \neq v_0$, a kernel with constant signal Δ at v_0 and constant signal 2Δ at v. These alternatives will be used only for the confidence bound. Let Θ contain P^0, all $F2^m$ parity kernels, and these $F - 1$ alternatives, so $|\Theta| = F(2^m + 1)$. Within Θ^H, restrict attention to sequences that use an arbitrary member of Θ at each h_ℓ, $\ell \leq L$, and use P^0 at all other stages. Every learner that is PAC on Θ^H must also be PAC on this subfamily; it suffices to prove a lower bound on the subfamily.

The common right factors are the $J - 1$ internal-node point masses, including the root, the J uniform live-branch distributions, and the two absorbing point masses. Their supports are disjoint. Every kernel row is a convex combination of these distributions, and every distribution occurs as a row. Consequently, each kernel has rank exactly $2J + 1 \leq d$. The left factors have norm at most one, and summing $0 \leq f \leq 1$ against the right factors gives a vector of norm at most $\sqrt{2J+1}$. Zero padding gives the normalization in Definition 1. The right factor may be disclosed to the learner. Neither L, n, nor m adds a transition direction.

Step 2: relate value to prediction. First put an independent uniform prior on (v_ℓ, θ_ℓ) at each active stage. This prior is supported on the hard subfamily of Θ^H. A policy knowing the complete sequence selects branch j_ℓ in each tree traversal, queries when $i = i_\ell$ and $(-1)^{\theta_\ell^\top x} = 1$, and otherwise returns to the root. Conditional on reaching a live stage, the probability of this event is at least $1/(2n)$. Every active stage leaves at least $3H/4$ subsequent reward steps. Since $L/n \geq 16$,

$$V^\star - H/2 \geq \tfrac{3}{4} H\Delta(1 - e^{-8}) \geq \tfrac{23}{32} H\Delta. \tag{60}$$

For a fixed realized output policy π, let

$$W_{\ell,v,\theta}(\pi) = \mathbb{E}\left[\mathbf{1}\{\text{query at } h_\ell \text{ with target } v\}(-1)^{\theta^\top x}\right].$$

The expectation is over a fresh deployment trajectory up to its query. Before a query, every model has identical transitions. Thus W is a functional of the policy and (ℓ, v, θ), independent of the environment used for its evaluation. The label drawn on entering the live state is uniform, so $|W_{\ell,v,\theta}(\pi)| \leq 1/n$. For a sequence of parity models,

$$V^\pi - H/2 = \Delta \sum_{\ell=1}^{L} (H - h_\ell) W_{\ell,v_\ell,\theta_\ell}(\pi). \tag{61}$$

In particular, every policy has gain at least $-H\Delta$. Combining PAC success with (60), $\epsilon/(H\Delta) = 1/32$, and $\delta \leq 1/16$ shows that under the product prior,

$$\mathbb{E}[V^{\widehat{\pi}} - H/2] \geq H\Delta \left[(1-\delta)\frac{11}{16} - \delta\right] \geq \frac{149}{256} H\Delta > \frac{1}{2} H\Delta. \tag{62}$$

Step 3: bound the gain at a stage with few queries. Suppose the learner uses at most N episodes, padding early stopping if necessary. Write N_ℓ for its number of training queries at active stage h_ℓ. On every transcript, $\sum_\ell N_\ell \leq N$. Set

$$B_{\text{target}} = \left\lfloor \frac{m \log 2}{16\Delta^2} \right\rfloor, \qquad B_{\text{stage}} = \lfloor F B_{\text{target}}/4096 \rfloor, \tag{63}$$

and suppose, for a contradiction, that $N \leq n B_{\text{stage}}/8$. The parameter conditions imply $F B_{\text{target}} \geq 8192$, so both caps are positive and $B_{\text{stage}} \geq F B_{\text{target}}/8192$.

Fix an active stage ℓ and fix all other stage parameters. Truncate training as soon as its B_{stage}th query at stage h_ℓ is observed. On truncation, assign the output policy that never queries; otherwise use the learner's output after N episodes. Let $\widehat{\pi}^{[\ell]}$ denote this experiment's output. Under a reference law $\mathbb{P}_{0,\ell}$, change only stage h_ℓ to P^0. Data and computations at all other stages can be arbitrarily adaptive and are unchanged conditional on the observed history.

Let $N_{\ell,v}$ count target-v queries in the stage-truncated experiment, and let $\mathbb{E}_{\ell,v,\theta}$ denote expectation when stage ℓ has target v and parity θ, with other stage parameters fixed. For a target v, further stop at the B_{target}th query to (h_ℓ, v). Let L_θ be the stopped likelihood ratio for parity θ at that target relative to $\mathbb{P}_{0,\ell}$. Exactly as in (52),

$$\mathbb{E}_{0,\ell} L_\theta^2 \leq (1 + 4\Delta^2)^{B_{\text{target}}}.$$

Denote the further-stopped reference law by $\mathbb{P}_{0,\ell}^{\text{cap}}$ and its uniform parity-mixture counterpart by $\overline{\mathbb{P}}_{\ell,v}^{\text{cap}}$. There is at most one fresh stage-h_ℓ live context per episode. Applying the conditional Parseval argument in (53) at these at most N opportunities gives

$$\chi^2(\overline{\mathbb{P}}_{\ell,v}^{\text{cap}} \| \mathbb{P}_{0,\ell}^{\text{cap}}) \leq \frac{4\Delta^2 T}{n 2^m} (1 + 4\Delta^2)^{B_{\text{target}}} \leq \frac{F m \log 2}{131072}\, 2^{-3m/4}. \tag{64}$$

The bar averages over θ. In particular, the total variation distance a_{TV} of these stopped laws is at most $1/4096$ under the stated condition on m.

We also need a prediction bound, since identifying a target at one stage does not identify the independent parities at the other stages. On the further-stopped experiment, put the policy equal to the never-query policy when the target cap is reached. Denote this output by π^{cap}. For each fixed

stopped transcript and fresh history before the stage-h_ℓ live context, the branch is fixed and the policy's query probability is a function in $[0,1]$ of the new (i,x). Removing this function after taking absolute values and then applying Cauchy–Schwarz and Parseval yields

$$\left|2^{-m}\sum_\theta L_\theta W_{\ell,v,\theta}(\pi^{\mathrm{cap}})\right| \le \frac{1}{n}\mathbb{E}_x\left|2^{-m}\sum_\theta L_\theta(-1)^{\theta^\top x}\right| \le \frac{1}{n2^{m/2}}\left(2^{-m}\sum_\theta L_\theta^2\right)^{1/2}.$$

This bound also averages over the fresh history and the policy's private randomness. The deployment context is independent of the stopped training transcript. Set $a_{\mathrm{pred}} = 2^{-m/2}(1+4\Delta^2)^{B_{\mathrm{target}}/2}$. Taking reference expectations and using Jensen's inequality therefore gives

$$\begin{aligned} 2^{-m}\sum_\theta \mathbb{E}_{\ell,v,\theta}\left[W_{\ell,v,\theta}(\widehat{\pi}^{[\ell]})\mathbf{1}\{N_{\ell,v} < B_{\mathrm{target}}\}\right] &\le \frac{a_{\mathrm{pred}}}{n}, \\ a_{\mathrm{pred}} := 2^{-m/2}(1+4\Delta^2)^{B_{\mathrm{target}}/2} &\le 2^{-3m/8} \le \frac{1}{4096}. \end{aligned} \tag{65}$$

Here the event and count refer to the experiment with the total stage cap B_{stage}; when its cap is reached, its output already has $W = 0$.

The event of reaching the target cap is measurable on the further-stopped transcript. Total variation comparison bounds its probability under the parity mixture by $\mathbb{P}_{0,\ell}(N_{\ell,v} \ge B_{\mathrm{target}}) + a_{\mathrm{TV}}$. Under the common reference experiment with the total stage cap, the sum of these probabilities over v is at most $B_{\mathrm{stage}}/B_{\mathrm{target}}$, since its total stage-query count is at most B_{stage} on every transcript. Combining this fact with (65), the bound $W \le 1/n$, and the uniform target prior proves

$$\mathbb{E}W_{\ell,v_\ell,\theta_\ell}(\widehat{\pi}^{[\ell]}) \le \frac{1}{n}\left(\frac{B_{\mathrm{stage}}}{FB_{\mathrm{target}}} + a_{\mathrm{TV}} + a_{\mathrm{pred}}\right) \le \frac{3}{4096n}. \tag{66}$$

The estimate is uniform in the other stages' fixed parameters, so it also holds after averaging over their prior. This step permits arbitrary sharing of data and computation between stages.

Step 4: sum over the active stages. Couple the original learner and the stage-truncated experiment until its B_{stage}th stage-h_ℓ query. If $N_\ell < B_{\mathrm{stage}}$, their outputs agree. Otherwise the truncated output has zero W, while the original output has $W \le 1/n$. Therefore

$$\mathbb{E}W_{\ell,v_\ell,\theta_\ell}(\widehat{\pi}) \le \frac{3}{4096n} + \frac{1}{n}\mathbb{P}(N_\ell \ge B_{\mathrm{stage}}).$$

This comparison now uses the same full product-prior law for every ℓ. The deterministic query count implies $\sum_\ell \mathbb{P}(N_\ell \ge B_{\mathrm{stage}}) \le N/B_{\mathrm{stage}}$. Multiplying each stage bound by its nonnegative remaining horizon in (61) gives

$$\frac{\mathbb{E}[V^{\widehat{\pi}} - H/2]}{H\Delta} \le \frac{3L}{4096n} + \frac{N}{nB_{\mathrm{stage}}} \le \frac{3}{128} + \frac{1}{8} = \frac{19}{128} < \frac{1}{2},$$

where $L/n \le 32$. This contradicts (62). Hence

$$N > \frac{nB_{\mathrm{stage}}}{8} \ge \frac{nFB_{\mathrm{target}}}{65536} \ge \frac{nFm\log 2}{2^{21}\Delta^2}. \tag{67}$$

The proof uses the caps only as comparison experiments. It does not restrict the learner's allocation across stages, targets, or episodes.

Step 5: prove the confidence lower bound. Use as baseline the sequence with constant signal Δ at v_0 at every active stage. For each $v \ne v_0$, use the sequence with this baseline signal and constant signal 2Δ at v at every active stage. All these sequences are in the same product class used above. Let $p_v(\pi)$ be the probability of querying target v at any active stage. It is model independent and $\sum_v p_v \le 1$.

In the baseline, (60) holds with a constant positive signal, so PAC success requires $p_{v_0} \ge 11/16$. In the alternative at v, the informed policy obtains gain at least $(23/32)2H\Delta$, whereas every policy's gain is at most $H\Delta(p_{v_0} + 2p_v) \le H\Delta(1+p_v)$. PAC success there requires $p_v \ge 13/32$. Consequently the event $E_v = \{p_v(\widehat{\pi}) \ge 3/8\}$ has baseline probability at most δ and alternative probability at least $1-\delta$.

Let N_v be the total training-query count at target v, summed over active stages, and let $\mathbb{E}_{\text{base}}$ denote expectation under the baseline. Only a real query at target v at an active stage distinguishes these two laws. Its conditional divergence is at most $32\Delta^2/3$, as in the shared-kernel confidence argument. The adaptive KL chain rule and data processing therefore imply

$$\frac{32}{3}\Delta^2\mathbb{E}_{\text{base}}N_v \geq \frac{1}{2}\log(1/\delta).$$

Summing over $v \neq v_0$ under this one baseline yields

$$N \geq \frac{3(F-1)}{64\Delta^2}\log(1/\delta). \tag{68}$$

Both lower bounds hold for any learner that is PAC on Θ^H. Their maximum is at least half their sum, proving (58). Finally, $J = \Theta(d)$, $G = \Theta(A)$, $H/(128C) \leq n \leq H/(64C)$, $L = \Theta(H/C)$, and $\log|\Theta| = \Theta(m)$ imply (59). □

D.3 Comparison with the upper bound

The entropy lower bound is $\Omega(H^4 dA \log|\Theta|/(C^2\epsilon^2))$, within a factor $C^2 = O((1+\log d)^2)$ of the entropy term in Theorem 4. This establishes the additional horizon factor in the entropy term even when the right factors are shared and known. The proof needs independent stage parameters and a prediction bound; merely enlarging the stationary class does not give this conclusion.

Corollary 24 (Matching rate up to logarithmic factors)**.** *For the parameters and common class size $|\Theta|$ in Theorem 10, the worst-case PAC episode complexity under the stagewise access assumptions of Theorem 4 is*

$$\widetilde{\Theta}\left(\frac{H^4 dA\log|\Theta|}{\epsilon^2}\right).$$

Here the displayed model-class entropy is retained, while the logarithmic factors specified in Section 2, including logarithms of $1/\delta$, are suppressed.

Proof of Corollary 24. The upper bound holds for every common class Θ, and Theorem 10 supplies a class attaining the lower bound in its stated cardinality range. Apart from the logarithms already suppressed in the upper bound, the ratio of its expression to the entropy lower bound is at most a constant times

$$C^2\left(1+\frac{1+\log(H/\delta)}{\log|\Theta|}\right).$$

Since $C = O(1+\log d)$ and $\log|\Theta| \geq 1$ in the stated range, this is a logarithmic factor. The conclusion holds throughout the stated confidence range. □

The explicit confidence dependence remains different: the lower bound gives $H^3 dA\log(1/\delta)/C$, whereas the current stagewise upper analysis gives $H^4 dA\log(H/\delta)$. This does not prevent the matching rate in Corollary 24, whose notation suppresses confidence logarithms. Determining the sharp coefficient of $\log(1/\delta)$ is a separate refinement.

D.4 The confidence dependence of this construction

The confidence term in (58) is sharp for the restricted hard subfamily, with P^0 known at all inactive stages. We give an upper bound for this subfamily to clarify the limitation of the construction. This guarantee concerns the subfamily, rather than all of Θ^H.

For each active stage, collect n_{fit} separate episodes that continue at all preceding live stages, choose a uniformly random branch on the last tree traversal, and query a uniformly random query action at the designated stage. The resulting (j,i,g,x) is uniform. Data for distinct stages use disjoint episodes and independent randomization. Fit each stage by exact MLE over Θ, discarding other stages' data, and plan optimally in the fitted sequence. There is no restriction on computation in this comparison.

Write $s_Q(j,i,g,x) = Q(\text{good} \mid j,i,x,g) - 1/2$. Any two distinct members of Θ satisfy

$$\mathbb{E}_{j,i,g,x}(s_Q - s_{Q'})^2 \geq \Delta^2/F.$$

For parity kernels this follows from orthogonality or disjoint targets. For the confidence alternatives it follows by cancelling their common baseline signal; an alternative compared with a parity kernel or P^0 also has a squared difference of at least Δ^2/F. Since squared Hellinger distance is at least squared total variation, the affinity of the distributions of one uniform query input and its outcome is at most $1 - \Delta^2/(2F)$. The square-root likelihood-ratio bound and a union over candidates give

$$\mathbb{P}(\widehat{P}_{h_\ell} \neq P^\star_{h_\ell}) \leq |\Theta| \exp\left(-\frac{n_{\mathrm{fit}}\Delta^2}{2F}\right) =: p.$$

These error events are independent across active stages for every fixed true sequence.

If r fitted stages are incorrect, then for every fixed policy

$$|V^\pi_P - V^\pi_{\widehat{P}}| \leq 6H\Delta r/n.$$

Indeed, all transitions before a query agree. At a given incorrect stage, each model's signal has total amplitude at most 3Δ across its targets. After averaging the fresh uniform label and taking the maximum over query actions, their total variation difference is at most $6\Delta/n$, regardless of the selected branch or preceding history. A coupling at the sole query then proves the displayed value bound. Thus planning in the fitted sequence has regret at most $12H\Delta r/n = 384\epsilon r/n$.

Set $r_0 = \lfloor n/384 \rfloor + 1$ and take

$$n_{\mathrm{fit}} = \left\lceil \frac{2F}{\Delta^2}\left[\log|\Theta| + \log\frac{eL}{r_0} + \frac{1}{r_0}\log\frac{1}{\delta}\right]\right\rceil.$$

If fewer than r_0 stages are incorrect, the regret is at most ϵ. Independence and a union over subsets of r_0 incorrect stages give

$$\mathbb{P}(r \geq r_0) \leq \binom{L}{r_0} p^{r_0} \leq (eLp/r_0)^{r_0} \leq \delta.$$

Since $L/n \in [16, 32]$ and $r_0 \geq n/384$, this algorithm uses

$$Ln_{\mathrm{fit}} = O\left(\frac{F}{\Delta^2}\left[n\log|\Theta| + \log(1/\delta)\right]\right)$$

episodes. Together with (58), this characterizes the restricted hard subfamily up to absolute constants. Consequently, a stronger confidence lower bound for the general problem would require another class.

E SUBROUTINES: ALGORITHMS AND GUARANTEES

We give five subroutines for the uncertainty marking algorithm (Algorithm 1). ESTIMATERADIUS estimates $u_{k,h}(s,a)$, and MODELEPISODE simulates the augmented fitted model. LINPLAN solves an additive planning problem, while TESTMARK estimates a policy's marking probability. PLAN uses these routines to return a policy with near-optimal J_k. Each subsection states the inputs, algorithm, and guarantee before proving the result.

Throughout this appendix, fix iteration k and condition on the historical data and fitted kernels $\widehat{P}^{(k)} = (\widehat{P}^{(k)}_1, \ldots, \widehat{P}^{(k)}_H)$. All subroutines share these kernels, the known rewards, the histories $\mathcal{I}_{k,h}$, and the numerical parameters of Appendix A.1; we suppress these shared arguments in internal calls. Every simulated transition is drawn from a fitted kernel, separately from real environment interaction. A subroutine's failure probability is conditional on all inputs and earlier calls. Fresh random samples and the tower property allow these guarantees to be combined in call order.

E.1 ESTIMATERADIUS: RADIUS ESTIMATION

Define the historical density sum $\widehat{p}_{\Sigma,k,h}(s') = \sum_{(s_{i,j},a_{i,j})\in\mathcal{I}_{k,h}} \widehat{P}^{(k)}_h(s' \mid s_{i,j}, a_{i,j})$. The radius has the scalar representation

$$u_{k,h}(s,a) = \min_{0\leq t\leq 2/L_{\mathrm{rad}}}\left\{tL_{\mathrm{rad}} + 2\mathbb{E}_{S\sim\widehat{P}^{(k)}_h(\cdot|s,a)}[1 - t\frac{\widehat{p}_{\Sigma,k,h}(S)}{\widehat{P}^{(k)}_h(S \mid s,a)}]_+\right\}. \tag{69}$$

Algorithm 2 implements the interface (5) using (69). It returns an estimate of the radius in $[0, 2]$; the marking probability is the estimate capped at one.

For the sample count and correction, set

$$n = \left\lceil \frac{1024}{\zeta} \log \frac{1024}{\zeta \delta_0} \right\rceil, \qquad c_n = \frac{8 \log(4(n+1)/\delta_0) + 2}{n}. \tag{70}$$

Let $S_1, \ldots, S_n$ be the sampled next states. The estimator is

$$\widehat{u}_{k,h}(s,a) = \min \left\{ 2, \min_{0 \le t \le 2/L_{\mathrm{rad}}} \left\{ t L_{\mathrm{rad}} + \frac{4}{n} \sum_{j=1}^{n} [1 - t \frac{\widehat{p}_{\Sigma,k,h}(S_j)}{\widehat{P}_h^{(k)}(S_j \mid s,a)}]_+ \right\} + 2c_n \right\}. \tag{71}$$

For a positive ratio $\widehat{p}_{\Sigma,k,h}(S_j)/\widehat{P}_h^{(k)}(S_j \mid s,a)$, the corresponding breakpoint is $\widehat{P}_h^{(k)}(S_j \mid s,a)/\widehat{p}_{\Sigma,k,h}(S_j)$. Ratios with zero numerator contribute a constant and need no breakpoint.

Algorithm 2 EstimateRadius

Require: Fitted kernel $\widehat{P}_h^{(k)}$, history $\mathcal{I}_{k,h}$, budget $L_{\mathrm{rad}} > 0$, query (s,a) with $h < H$, and $\zeta \in (0,1]$, $\delta_0 \in (0,1)$.

Ensure: An estimate $\widehat{u}_{k,h}(s,a) \in [0,2]$.

1: Set n, c_n by (70) and draw $S_1, \ldots, S_n$ independently from $\widehat{P}_h^{(k)}(\cdot \mid s,a)$.
2: Evaluate the fitted density and historical density sum at every S_j.
3: Sort the breakpoints lying in $[0, 2/L_{\mathrm{rad}}]$ and include the interval endpoints.
4: Evaluate the objective in (71) at these points using cumulative sums.
5: Return its minimum with the correction and clipping in (71).

The denominator is positive almost surely under the sampling distribution. The historical density sum is the inner product of the sum of historical fitted features with the fitted next-state factor. Appendix A proves the dual identity, including unequal supports.

Proposition 25 (Radius estimation)**.** *Conditional on the fitted kernel and history, Algorithm 2 returns a value in* $[0,2]$ *satisfying*

$$\mathbb{P}(u_{k,h}(s,a) \le \widehat{u}_{k,h}(s,a) \le \min\{2, 4u_{k,h}(s,a) + \zeta\}) \ge 1 - \delta_0. \tag{72}$$

It uses n *fitted-kernel samples, where* n *is specified in* (70)*.*

Proof of Proposition 25. Fix $h < H$ and a state-action pair. In this subsection suppress the stage index, writing $\widehat{P}^{(k)} = \widehat{P}_h^{(k)}$, $u_k = u_{k,h}$, and $\widehat{p}_\Sigma = \widehat{p}_{\Sigma,k,h}$. Put $T = \widehat{p}_\Sigma(S)/\widehat{P}^{(k)}(S \mid s,a) \ge 0$ and $f_t(T) = (1 - tT)_+$. For an integer sample count $n \ge 1$, draw independent $T_1, \ldots, T_n$ using the fitted kernel and denote empirical averages by $\mathbb{P}_n$. Define

$$c_n = \frac{8 \log(4(n+1)/\delta_0) + 2}{n}.$$

Step 1: uniform concentration over thresholds. With probability at least $1 - \delta_0$, the distribution function $F(r) = \mathbb{P}(T \le r)$ and empirical distribution function $\widehat{F}_n(r) = n^{-1} \sum_{j=1}^{n} \mathbf{1}\{T_j \le r\}$ satisfy, simultaneously for every threshold r,

$$F(r) \le 2\widehat{F}_n(r) + c_n, \qquad \widehat{F}_n(r) \le 2F(r) + c_n. \tag{73}$$

For completeness, represent each sample by the generalized inverse distribution function of an independent uniform variable. At each grid point $p = j/n$, Bernstein's inequality for the empirical Bernoulli mean $\hat{p}$ gives, for any $c > 0$,

$$\mathbb{P}(\hat{p} > 2p + c) \le e^{-3nc/8}, \qquad \mathbb{P}(p > 2\hat{p} + c) \le e^{-nc/2}.$$

For the second bound, when the event is possible, the required downward deviation is $(p+c)/2$ and $(p+c)^2/p \ge 4c$. Use $c = 8 \log(4(n+1)/\delta_0)/n$, union bound over the two tails and $n+1$ grid

points, and extend by monotonicity between grid points at cost at most $2/n$. Generalized inverse sampling also handles atoms; strict thresholds follow by taking left limits.

Step 2: compare the empirical and population objectives. Let $Z \sim \mathrm{Unif}([0,1])$ be independent of T. For $t > 0$,

$$f_t(T) = \mathbb{E}[\mathbf{1}\{T < (1-Z)/t\} \mid T].$$

Averaging (73) over Z gives, uniformly in t,

$$\mathbb{E}f_t \le 2\mathbb{P}_n f_t + c_n, \qquad \mathbb{P}_n f_t \le 2\mathbb{E}f_t + c_n.$$

The same inequalities hold trivially at $t = 0$. Writing $\widehat{u} = \widehat{u}_{k,h}(s,a)$ for the output in (71), we have

$$\widehat{u} = \min\left\{2, \min_{0 \le t \le 2/L_{\mathrm{rad}}} \{tL_{\mathrm{rad}} + 4\mathbb{P}_n f_t\} + 2c_n\right\}. \tag{74}$$

The first uniform inequality proves $u_k(s,a) \le \widehat{u}$. Evaluate the other at an ideal minimizer t_* to obtain

$$\widehat{u} \le t_* L_{\mathrm{rad}} + 8\mathbb{E}f_{t_*} + 6c_n \le 4u_k(s,a) + 6c_n.$$

Step 3: choose the sample count. The sample count in (70) has $6c_n \le \zeta$, proving (72). Indeed, put $a = \log(1024/(\zeta\delta_0)) \ge \log 1024$. Then $n \le 1025a/\zeta$ and $\log(4(n+1)/\delta_0) \le a + \log(5a) \le 2a$. Substitution yields $6c_n \le (96a+12)\zeta/(1024a) < \zeta$.

□

Computation and repeated calls. The empirical minimization requires only endpoints and breakpoints $1/T_j$ in $[0, 2/L_{\mathrm{rad}}]$. Sorting and using cumulative sums costs $O(n \log n)$; density-ratio evaluation costs $O(nd)$ arithmetic after forming the historical feature sum. Call the subroutine independently for all A actions at the same state. Restoring h, the joint law of these outputs is precisely $\mathcal{R}_{k,h,s}$ in (15). Fresh calls at every visit leave this law fixed while returning a new vector of radius estimates. The sample count is $\widetilde{O}(H^3/\epsilon^2)$ per state-action query. The estimates have a constant-factor upper error, not a uniform small additive error; the statistical proof uses precisely this relative guarantee.

E.2 ModelEpisode: fitted-model simulation

Fix iteration k and suppress k in the fitted factors and distributions of the radius estimates. Write $v(\pi) = V^{\pi}_{\widehat{P}^{(k)},r}$, $c(\pi) = C_k(\pi)$, and $\rho = \rho_k$. At stage h, write $z = z_h$ for the incoming flag and $\widehat{u} = \widehat{u}_h$ for the vector of radius estimates. Let $p_{\mathrm{mark}} = \mathbf{1}\{h < H\} \min\{1, \widehat{u}_a\}$. On the augmented state $\tilde{s} = (s, \widehat{u}, z)$ define

$$\psi_h(\tilde{s}, a) = \begin{pmatrix} (1-z)(1-p_{\mathrm{mark}})\widehat{\phi}_h(s,a) \\ [z + (1-z)p_{\mathrm{mark}}]\widehat{\phi}_h(s,a) \end{pmatrix} \in \mathbb{R}^{2d}. \tag{75}$$

Denote the augmented transition kernel by P^+_h and its next-state factor by μ^+_h. Thus $P^+_h(d\tilde{s}' \mid \tilde{s}, a) = \psi_h(\tilde{s}, a)^\top \mu^+_h(d\tilde{s}')$. The measure μ^+_h has two blocks, supported respectively on $z' = 0$ and $z' = 1$, with factors $\widehat{\mu}_h(s')\mathcal{R}_{h+1,s'}(d\widehat{u}')$. Hence the augmented kernel is linear in these known features. Their norm is at most one. For a next-state function in $[0,1]$, averaging over the fresh radius vector and summing over the next physical state gives each block norm at most $\sqrt{d}$, so the combined coefficient norm is at most $\sqrt{2d}$. The root vector of radius estimates is drawn from the known distribution $\mathcal{R}_{1,s_1}$.

For a coefficient $\gamma \ge 0$, the objective $v(\pi) + \gamma c(\pi)$ is the expected sum of original rewards and a one-time payment γ on marking. After normalization, its known expected immediate reward is

$$\bar{r}_{\gamma,h}(\tilde{s}, a) = \frac{r_h(s,a) + \gamma(1-z)\mathbf{1}\{h < H\}\min\{1, \widehat{u}_a\}}{H + \gamma}.$$

Although realized marking payments are random, this expected reward defines the same policy values. The remaining expected reward of every policy is in $[0,1]$ from every augmented state, because the actual total is at most $H + \gamma$. Values must be clipped to $[0,1]$, including at late stages before marking. A bound of $(H - h + 1)/H$ would be invalid for this augmented task.

Inputs and output. MODELEPISODE takes a policy π with its initialization and action rules. It returns the augmented transitions, the original total reward R, and the final marking indicator Y. At a new state, it generates the entire vector $\widehat{u}_h$ before querying the policy and retains that vector for the marking coin. Set $\widehat{u}_H = \widehat{u}_{H+1} = 0$.

Algorithm 3 MODELEPISODE

Require: Policy π and the shared fitted kernels and radius rules.
Ensure: Augmented transitions $(\tilde{s}_h, a_h, \tilde{s}_{h+1})_{h=1}^H$, total reward R, and marking indicator Y.
1: Initialize π at the fixed initial state s_1 and set $z_1 = 0$.
2: **for** $h = 1, \dots, H$ **do**
3: If $h < H$, call ESTIMATERADIUS independently at (s_h, a) for every $a \in \mathcal{A}$; otherwise set $\widehat{u}_h = 0$.
4: Set $\tilde{s}_h = (s_h, \widehat{u}_h, z_h)$ and execute the action a_h returned by π.
5: If $z_h = 1$, set $z_{h+1} = 1$; otherwise draw $z_{h+1} \sim \mathrm{Bernoulli}(\min\{1, \widehat{u}_{k,h}(s_h, a_h)\})$.
6: Draw $S \sim \widehat{P}_h^{(k)}(\cdot \mid s_h, a_h)$ and set $s_{h+1} = S$.
7: **end for**
8: Set $\tilde{s}_{H+1} = (s_{H+1}, 0, z_{H+1})$, $R = \sum_{h=1}^H r_h(s_h, a_h)$, and $Y = z_{H+1}$.
9: Return the augmented transitions, R, and Y.

Lemma 26 (Fitted-model simulation)**.** *Algorithm 3 generates the fitted augmented trajectory law of π. In particular, $\mathbb{E}R = v(\pi)$ and $\mathbb{E}Y = c(\pi)$. Its augmented transition model has dimension $2d$ with features* (75)*. Under the normalized reward $\bar{r}_{\gamma,h}$, every policy's remaining expected return lies in $[0, 1]$ at every augmented state.*

Proof of Lemma 26. The radius outputs have law $\mathcal{R}_{k,h,s_h}$ and are sampled independently at each visit. Conditional on the current augmented state and action, the marking coin and fitted transition have exactly the specified laws. Induction over stages therefore gives the trajectory claim and the identities for R and Y. Splitting the next-state law according to $z' = 0$ and $z' = 1$ gives (75). The two left-factor coefficients are nonnegative and sum to one, so the feature norm is at most one. For each flag value, averaging over the fresh radius vector leaves a $[0, 1]$ function of the next physical state. Summing it against $\widehat{\mu}_h$ gives norm at most $\sqrt{d}$; their combined norm is at most $\sqrt{2d}$. Finally, original rewards total at most H, and the extra payment γ occurs at most once. Dividing by $H + \gamma$ proves the remaining-value bound, including at states that have not yet been marked. □

Computation. One episode makes at most $(H - 1)A$ calls to ESTIMATERADIUS and draws H physical next states. With sample count n per radius query, it uses at most $H + (H - 1)An$ fitted physical-kernel samples, in addition to policy evaluation. It adds no uniform exploration actions. Independent calls initialize the policy independently.

E.3 LINPLAN: ADDITIVE PLANNING

Fix one call LINPLAN$(\gamma, e_L, \delta_{\mathrm{in}})$, where $\gamma \geq 0$, $e_L \in (0, 1]$, and $\delta_{\mathrm{in}} \in (0, 1)$. Use the normalized task of Appendix E.2, dimension $D = 2d$, and J model episodes. Within this call, $(\tilde{s}_h^i, a_h^i, \tilde{s}_{h+1}^i)$ denotes the stage-h transition of model episode i. Set $\psi_h^i = \psi_h(\tilde{s}_h^i, a_h^i)$ and let I_D be the identity matrix. Before episode j maintain

$$\Lambda_h^j = I_D + \sum_{i<j} \psi_h^i (\psi_h^i)^\top.$$

Update all regression parameters at $j = 1$ and whenever some h satisfies $\det \Lambda_h^j > 2 \det \Lambda_h^{j_0}$, where j_0 is the last update time. Let $\beta_{\mathrm{in}} > 0$ be the confidence coefficient specified below and

$\mathrm{clip}_{[0,1]}(x) = \min\{1, \max\{0, x\}\}$. At an update, compute backward with $V^j_{H+1} = 0$:

$$w^j_h = (\Lambda^j_h)^{-1} \sum_{i<j} \psi^i_h V^j_{h+1}(\tilde{s}^i_{h+1}), \tag{76}$$

$$Q^j_h(\tilde{s}, a) = \mathrm{clip}_{[0,1]} \left(\bar{r}_{\gamma,h}(\tilde{s}, a) + \psi_h(\tilde{s}, a)^\top w^j_h + \beta_{\mathrm{in}} \|\psi_h(\tilde{s}, a)\|_{(\Lambda^j_h)^{-1}} \right), \tag{77}$$

$$V^j_h(\tilde{s}) = \max_a Q^j_h(\tilde{s}, a). \tag{78}$$

At an update time j_0, store the deterministic policy $\pi^{j_0} = (\pi^{j_0}_h)_{h=1}^H$ defined by

$$\pi^{j_0}_h(\tilde{s}) = \arg\max_{a \in \mathcal{A}} Q^{j_0}_h(\tilde{s}, a), \qquad \tilde{s} = (s, \widehat{u}, z),$$

with a fixed action tie rule. Storing its regression vectors, Gram matrices, and coefficient γ, together with the fitted model and radius rule, specifies this action map at every augmented state. Between updates execute π^{j_0}. Save every new model transition and update its Gram matrix. Preserve the sampled radius estimates $\widehat{u}$ when recomputing historical targets, since they are part of the stored augmented states. Return the episode mixture of the stored policies, with each weight equal to the number of model episodes that used that policy divided by J.

Here is an explicit parameter choice with conservative constants. With

$$L_{\mathrm{in}} = \log \frac{10^8 DH(H+\gamma)}{e_L \delta_{\mathrm{in}}}, \quad J = \left\lceil \frac{10^{12} D^3 H^2 (H+\gamma)^2 L^4_{\mathrm{in}}}{e^2_L} \right\rceil, \quad \beta_{\mathrm{in}} = 100 D \log \frac{8DHJ}{\delta_{\mathrm{in}}},$$

we obtain the algorithm below. For each stored policy π^{j_0}, let n_{j_0} count the model episodes that execute it. The subroutine returns the episode mixture with weights n_{j_0}/J.

Algorithm 4 LINPLAN

Require: Coefficient $\gamma \geq 0$, objective tolerance $e_L \in (0, 1]$, failure probability $\delta_{\mathrm{in}} \in (0, 1)$, and the shared augmented model.

Ensure: An executable policy π for the additive objective $v(\pi) + \gamma c(\pi)$.

1: Set $D = 2d$, $L_{\mathrm{in}}, J, \beta_{\mathrm{in}}$ as above; initialize empty stagewise transition lists and $\Lambda^1_h = I_D$.
2: **for** $j = 1, \ldots, J$ **do**
3: **if** $j = 1$ or some stage has $\det \Lambda^j_h > 2 \det \Lambda^{j_0}_h$, where j_0 is the last update time **then**
4: Set $V^j_{H+1} = 0$; compute w^j_h, Q^j_h, V^j_h backward using (76) and (77).
5: Store the greedy policy π^j, set $j_0 = j$, and initialize $n_{j_0} = 0$.
6: **end if**
7: Call MODELEPISODE with π^{j_0} and append its augmented transitions to the stagewise lists.
8: For every h, form ψ^j_h and set $\Lambda^{j+1}_h = \Lambda^j_h + \psi^j_h (\psi^j_h)^\top$.
9: Increment n_{j_0} by one.
10: **end for**
11: Return the policy that samples a stored index j_0 with probability n_{j_0}/J once per episode and executes π^{j_0} throughout.

Proposition 27 (Additive planning)**.** *Conditional on its inputs, Algorithm 4 returns a policy π satisfying*

$$v(\pi) + \gamma c(\pi) \geq \sup_{\pi'} [v(\pi') + \gamma c(\pi')] - e_L$$

with probability at least $1 - \delta_{\mathrm{in}}$. It uses J fitted-model episodes and stores at most $1 + DH \log_2(1 + J/D)$ deterministic augmented-state policies. Its regression arithmetic is $\widetilde{O}(AD^3H^2J + D^4H^2 + HJD^2)$, excluding radius estimation and model-interface evaluations.

Proof of Proposition 27. We first prove uniform regression confidence and optimism by the LSVI argument of Jin et al. (2020). We then sum the bonuses using the elliptical potential bound, as in the sublinear-summation analysis of Huang et al. (2026). The features are fixed by the fitted model, and the known reward is kept outside the regression.

Step 1: uniform regression confidence. For a fixed stage h and a function V taking values in $[0,1]$, define its next-state coefficient by

$$\theta_{h,V} = \begin{pmatrix} \sum_{s' \in \mathcal{S}} \widehat{\mu}_h(s')\, \mathbb{E}_{\widehat{u}' \sim \mathcal{R}_{h+1,s'}} V(s', \widehat{u}', 0) \\ \sum_{s' \in \mathcal{S}} \widehat{\mu}_h(s')\, \mathbb{E}_{\widehat{u}' \sim \mathcal{R}_{h+1,s'}} V(s', \widehat{u}', 1) \end{pmatrix}.$$

The two blocks correspond to the next flag values zero and one. Each expectation is over the fresh radius vector at s', and each sum is over the next physical state. Thus $\|\theta_{h,V}\| \le \sqrt{D}$. The centered targets $V(\tilde{s}^i_{h+1}) - P^+_h V(\tilde{s}^i_h, a^i_h)$ are bounded martingale differences. The usual exponential-mixture proof of the self-normalized inequality gives

$$\|\sum_{i<j} \psi^i_h (V(\tilde{s}^i_{h+1}) - P^+_h V(\tilde{s}^i_h, a^i_h))\|_{(\Lambda^j_h)^{-1}} \le \sqrt{D \log(1 + J/D) + 2\log(1/\delta')}$$

with failure probability δ', using $\det \Lambda^j_h \le (1 + J/D)^D$. This is the standard self-normalized step in LSVI (Jin et al., 2020).

To apply it to reused targets, take a uniform net over the possible value functions. Write $\|\cdot\|_F$ for the Frobenius norm. Each has the form

$$\max_a \mathrm{clip}_{[0,1]} \left(\bar{r}_{\gamma,h}(\tilde{s}, a) + \psi_h(\tilde{s}, a)^\top w + \sqrt{\psi_h(\tilde{s}, a)^\top M \psi_h(\tilde{s}, a)} \right),$$

where $\|w\| \le \sqrt{J}$, $M \succeq 0$, and $\|M\|_F \le \sqrt{D}\beta^2_{\mathrm{in}}$. The bound on w follows by collecting the feature rows in Ψ and value targets in $\mathbf{v}$, and writing (76) as $(I + \Psi^\top \Psi)^{-1} \Psi^\top \mathbf{v}$ with $\|\mathbf{v}\| \le \sqrt{J}$. A Euclidean net for w and a Frobenius net for M, together with $|\sqrt{a} - \sqrt{b}| \le \sqrt{|a-b|}$, give a sup-norm ϵ_0 net of logarithmic size at most

$$D \log(1 + 8\sqrt{J}/\epsilon_0) + \frac{D(D+1)}{2} \log(1 + 32\sqrt{D}\beta^2_{\mathrm{in}}/\epsilon^2_0).$$

One can choose centers within the parameter sets; the larger displayed constants allow this. Set $\epsilon_0 = 1/(8J)$ and union bound over H stages, at most J times, and these nets, using failure budget $\delta_{\mathrm{in}}/2$. Replacing a net function by the actual function changes the self-normalized sum by at most $2J\epsilon_0 = 1/4$. The regularization term costs at most $\sqrt{D}$. Substitution of β_{in} then proves, simultaneously for all updates, states, and actions,

$$|\psi^\top_h w^j_h - P^+_h V^j_{h+1}| \le \beta_{\mathrm{in}} \|\psi_h\|_{(\Lambda^j_h)^{-1}}. \tag{79}$$

The fixed known reward is part of the function-class definition; it introduces no additional parameter or entropy. The fitted features are fixed throughout this call, and the true unknown features are not used.

Step 2: optimism and planning error. Let V^*_h be the optimal remaining value in the normalized augmented model. Backward induction using (79) shows $V^j_h \ge V^*_h$ for the normalized task. Clipping at one preserves optimism, since true remaining values are at most one. The Bellman residual of a stored greedy policy is bounded above by twice its stored bonus. If no update is triggered, $\Lambda^j_h \succeq \Lambda^{j_0}_h$ and the determinant ratio is at most two. Every eigenvalue of $(\Lambda^{j_0}_h)^{-1/2} \Lambda^j_h (\Lambda^{j_0}_h)^{-1/2}$ is at least one and their product is at most two. Hence

$$\Lambda^j_h \preceq 2\Lambda^{j_0}_h, \qquad \|\psi\|_{(\Lambda^{j_0}_h)^{-1}} \le \sqrt{2} \|\psi\|_{(\Lambda^j_h)^{-1}}.$$

The same bound is immediate at an update. The elliptical potential identity gives $\sum_{j=1}^J \|\psi^j_h\|^2_{(\Lambda^j_h)^{-1}} \le 2D \log(1 + J/D)$. Thus the sum of twice the stored bonuses is at most

$$4\beta_{\mathrm{in}} H \sqrt{DJ \log(1 + J/D)}.$$

Telescoping Bellman residuals and applying Azuma's inequality also handles both the transition noise and the sampled initial radius estimates. For example, bounding the centered terms per episode

by $2(H+1)$ gives an additional $2(H+1)\sqrt{2J\log(2/\delta_{\text{in}})}$ with failure probability at most $\delta_{\text{in}}/2$. Dividing by J proves normalized average error at most

$$\frac{4\beta_{\text{in}}H\sqrt{D\log(1+J/D)}+2(H+1)\sqrt{2\log(2/\delta_{\text{in}})}}{\sqrt{J}}.$$

The displayed choice of J makes this at most $e_L/(H+\gamma)$: in particular $\log(8DHJ/\delta_{\text{in}}) \le 10L_{\text{in}}$ and $\log(1+J/D) \le 10L_{\text{in}}$. Multiplying by $H+\gamma$ proves the additive-response guarantee. This argument covers random update times through the event uniform in all regression times; reducing the update set does not assume independence of the targets.

Step 3: count updates and arithmetic operations. Each update after the first increases $\sum_h \log_2 \det \Lambda_h$ by more than one. Therefore the number of complete updates is at most

$$U_{\text{in}} \le 1 + DH\log_2(1+J/D).$$

Store the features and radius estimates of old data. One full update costs at most $O(HJAD^2 + HD^3)$, including recomputing next-state values over all actions. Rank-one inverse and determinant updates cost $O(D^2)$ per new sample. Total regression arithmetic is

$$O(U_{\text{in}}HJAD^2 + U_{\text{in}}HD^3 + HJD^2) = \widetilde{O}(AD^3H^2J + D^4H^2 + HJD^2).$$

The returned mixture contains at most U_{in} distinct deterministic policies. This is an application of the determinant switching rule of Gao et al. (2021) to a fixed fitted model.

□

E.4 TESTMARK: MARKING-PROBABILITY COMPARISON

TESTMARK takes a policy π, a threshold $p \in [0,1]$, tolerance $t > 0$, and failure probability $\delta_{\text{test}} \in (0,1)$. It compares $c(\pi)$ with p using independent marking indicators from MODELEPISODE. Set

$$n_{\text{test}} = \left\lceil 64\log(8/\delta_{\text{test}})\left(\frac{p}{t^2}+\frac{1}{t}\right)\right\rceil \tag{80}$$

Let $\hat{c}$ be the fraction of these episodes that end with a mark. The possible outputs are `above`, `below`, and `near`.

Algorithm 5 TESTMARK

Require: Policy π, threshold $p \in [0,1]$, tolerance $t > 0$, and $\delta_{\text{test}} \in (0,1)$.
Ensure: A comparison of $c(\pi)$ with p.
1: Set n_{test} by (80).
2: Make n_{test} independent calls to MODELEPISODE with π; let $\hat{c}$ be the average of their indicators Y.
3: If $\hat{c} > p + t$, return `above`; if $\hat{c} < p - t$, return `below`; otherwise return `near`.

Lemma 28 (Marking-probability test)**.** *With probability at least $1-\delta_{\text{test}}$, the output of Algorithm 5 implies respectively*

$$\begin{aligned} \texttt{above}: &\quad c(\pi) > p, \\ \texttt{below}: &\quad c(\pi) < p, \\ \texttt{near}: &\quad |c(\pi) - p| \le 2t. \end{aligned}$$

The guarantee is conditional on the input policy and all earlier calls. The test uses exactly n_{test} fitted-model episodes.

Proof of Lemma 28. Write $c = c(\pi)$. Lemma 26 makes the indicators independent Bernoulli variables of mean c. For a wrong positive sign, monotonicity in the Bernoulli mean reduces to mean p, and Bernstein bounds the probability by $\exp[-n_{\text{test}}t^2/(2(p+t/3))]$. A wrong negative sign is handled by the lower tail. If $c \ge p + 2t$ but `near` is returned, the lower-tail exponent is at least $n_{\text{test}}t^2/(2(p+2t))$; the function $(c-p-t)^2/(2c)$ increases on this range. The remaining case uses the corresponding upper tail for $c \le p - 2t$. Impossible boundary cases are omitted. The budget in (80) covers all four errors. □

E.5 PLAN: OPTIMISTIC PLANNING

The call $\textsc{Plan}(J_k; e, \delta_{\mathrm{plan}})$ takes the inputs defining J_k, an objective tolerance $e \in (0, 1]$, and a failure probability $\delta_{\mathrm{plan}} \in (0, 1)$. Algorithm 1 uses $e = \epsilon/100$. Set $\chi = 14\rho + \Delta_{\mathrm{MC}} + \alpha$, so $\alpha \le \chi \le 29$, and $f(c) = H\sqrt{18c + \chi}$. Write

$$S(\gamma) = \sup_{\pi'}[v(\pi') + \gamma c(\pi')], \qquad \gamma_- = \frac{9H}{\sqrt{\chi + 18}}, \quad \gamma_+ = \frac{9H}{\sqrt{\chi}}, \quad p(\gamma) = \frac{9H^2}{2\gamma^2} - \frac{\chi}{18}.$$

On this interval p decreases from one to zero, with $f'(p(\gamma)) = \gamma$. For $c, p \in [0, 1]$, if $x = 18c + \chi$ and $x_0 = 18p + \chi$, then

$$f(p) + f'(p)(c - p) - f(c) = \frac{H(\sqrt{x} - \sqrt{x_0})^2}{2\sqrt{x_0}} \le f'(p)|c - p|. \tag{81}$$

Each tangent is an upper bound for the concave function, so for any episode mixture π,

$$\sup_{\pi'}[v(\pi') + f(c(\pi'))] - [v(\pi) + f(c(\pi))] \le S(\gamma) - [v(\pi) + \gamma c(\pi)] + \gamma|c(\pi) - p(\gamma)|. \tag{82}$$

This only uses tangent inequalities, not an exchange of maximization and minimization. The supremum allows history-dependent policies. For each additive objective, finite-horizon dynamic programming attains its supremum with a deterministic Markov policy on the augmented state. Proposition 27 provides the approximate additive responses used below.

Call budgets. For an entirely deterministic failure allocation, upper bound all coefficients by $\Gamma = 900H^2/\epsilon$, put $w_{\min} = e/[16(1 + 47/9)]$, and $t_{\min} = e/(32\Gamma)$. The integer

$$Q_{\max} = 10 + \lceil \log_2 \max\{1, \Gamma/w_{\min}\} \rceil + \lceil \log_2 \max\{1, 1/t_{\min}\} \rceil$$

bounds both the number of additive calls and the number of tests in an iteration. Set $\delta_{\mathrm{call}} = \delta_{\mathrm{plan}}/(2Q_{\max})$ and pass it to every LINPLAN and TESTMARK call. In Algorithm 1, $\delta_{\mathrm{plan}} = \delta/(4K)$, recovering the total $\delta/4$ planning-failure budget.

In our calls $p = p(\gamma)$ and $t = e/(32\gamma)$. Since $p(\gamma) \le 9H^2/(2\gamma^2)$, each test uses $\widetilde{O}(H^2/e^2 + \gamma/e)$ episodes, which is $\widetilde{O}(H^2/\epsilon^2)$ when $e = \epsilon/100$ and $\gamma \le 900H^2/\epsilon$.

For two returned policies π_l, π_u and $\omega \in [0, 1]$, write $\pi_\omega = (1 - \omega)\pi_l + \omega\pi_u$ for the policy that chooses between them once per episode. Its marking probability is $(1 - \omega)c(\pi_l) + \omega c(\pi_u)$. The second loop below adjusts ω after the coefficient interval has been reduced.

Algorithm 6 PLAN

Require: Inputs defining J_k, tolerance $e \in (0, 1]$, and failure probability $\delta_{\mathrm{plan}} \in (0, 1)$.
Ensure: An executable policy π_k with approximate objective value $\sup_\pi J_k(\pi)$.

Set $\chi, \gamma_-, \gamma_+, p(\cdot), Q_{\max}, \delta_{\mathrm{call}}$ as above.
Set $e_L = e/8$, $C_\chi = (\chi + 18)/9$, $w = e/[16(1 + C_\chi)]$.
Set $l = \gamma_-$, $u = \gamma_+$ and compute $\pi_l = \textsc{LinPlan}(l, e_L, \delta_{\mathrm{call}})$, $\pi_u = \textsc{LinPlan}(u, e_L, \delta_{\mathrm{call}})$.
while $u - l > w$ **do**
  Set $\gamma = (l + u)/2$; compute $\pi = \textsc{LinPlan}(\gamma, e_L, \delta_{\mathrm{call}})$.
  Call $\textsc{TestMark}(\pi, p(\gamma), e/(32\gamma), \delta_{\mathrm{call}})$.
  If `near`, return π; if `below`, set $(l, \pi_l) = (\gamma, \pi)$; otherwise set $(u, \pi_u) = (\gamma, \pi)$.
end while
Set $\bar\gamma = (l + u)/2$, $p_0 = p(\bar\gamma)$, $t_0 = e/(32\bar\gamma)$.
Call $\textsc{TestMark}(\pi_l, p_0, t_0, \delta_{\mathrm{call}})$; if `near` or `above`, return π_l.
Call $\textsc{TestMark}(\pi_u, p_0, t_0, \delta_{\mathrm{call}})$; if `near` or `below`, return π_u.
Set $\omega_- = 0, \omega_+ = 1$.
while $\omega_+ - \omega_- > \min\{1, t_0\}$ **do**
  Set $\omega = (\omega_- + \omega_+)/2$ and $\pi_\omega = (1 - \omega)\pi_l + \omega\pi_u$.
  Call $\textsc{TestMark}(\pi_\omega, p_0, t_0, \delta_{\mathrm{call}})$.
  If `near`, return π_ω; if `below`, set $\omega_- = \omega$; otherwise set $\omega_+ = \omega$.
end while
Return $\pi_{(\omega_- + \omega_+)/2}$.

Proposition 29 (Bisection guarantee)**.** *If all response and test guarantees hold, Algorithm 6 has objective error at most e. It makes $O(1+\log_+((\gamma_+-\gamma_-)/w))$ additive planning calls and at most an additional $O(1+\log_+(1/t_0))$ probability tests. Its output retains at most two response mixtures.*

Proof of Proposition 29. **Step 1: maintain a valid coefficient interval.** The maintained facts are $c(\pi_l) \le p(l)$ and $c(\pi_u) \ge p(u)$, initially automatic at $p(l) = 1, p(u) = 0$. A middle `near` answer gives error at most $e_L + e/16$ by (82). The other answers preserve the stated inequalities. No claim that approximate responses are exact dual subgradients is required.

Step 2: bound the error at the final coefficient. At the final reference coefficient, both endpoint policies have additive error at most $e_L + (u-l)$. Indeed, changing a coefficient from γ_i to $\bar\gamma$ changes the difference of two policy objectives by at most $|\gamma_i - \bar\gamma|$, since both marking probabilities lie in $[0,1]$. The same error bound holds for every mixture of the two endpoints.

If the left endpoint test returns `above`, then $p_0 < c(\pi_l) \le p(l)$ and

$$\bar\gamma[p(l) - p(\bar\gamma)] = \frac{9H^2}{2}\frac{(\bar\gamma - l)(\bar\gamma + l)}{l^2\bar\gamma} \le \frac{9H^2}{l^2}(u-l) \le C_\chi(u-l).$$

For the right endpoint returning `below`, the same bound follows with $p(\bar\gamma) - p(u)$. Thus these returns have error at most $e_L + (1+C_\chi)w \le 3e/16$. An endpoint `near` return has error at most $e_L + w + e/16 < e$.

Step 3: find the mixture weight. If neither endpoint is returned, their marking probabilities straddle p_0. The mixture probability is affine in ω with slope in $[0,1]$. Valid tests preserve an interval containing its root. When the interval has width at most t_0, its midpoint has probability error at most t_0, and (82) proves the guarantee. Each loop halves its interval, proving the call bounds. □

Theorem 30 (Guarantee of PLAN)**.** *Conditional on the fitted kernels and historical data, Algorithm 6 returns an executable randomized policy π_k satisfying*

$$J_k(\pi_k) \ge \sup_\pi J_k(\pi) - e$$

with probability at least $1 - \delta_{\mathrm{plan}}$. It makes at most $Q_{\max}$ calls each to LINPLAN *and* TESTMARK*. With $e = \epsilon/100$ and the parameters of Algorithm 1, $Q_{\max} = O(1+\log(H/\epsilon))$, and all fitted-model calls and arithmetic operations are polynomial in the stated parameters.*

Proof of Theorem 30. The coefficient loop halves an interval of width at most Γ until its width is at most $w \ge w_{\min}$. The probability loop halves an interval of width one until its width is at most $\min\{1, t_0\}$, where $t_0 \ge t_{\min}$. Counting the initial responses and endpoint tests gives the displayed bound $Q_{\max}$. Proposition 27 and Lemma 28 apply conditionally to each adaptive call. Their failure probabilities sum to at most $2Q_{\max}\delta_{\mathrm{call}} = \delta_{\mathrm{plan}}$. On their joint success event, Proposition 29 proves the objective guarantee. The bounds on $\Gamma, w_{\min}, t_{\min}$ give the logarithmic call count. The polynomial implementation follows from Algorithms 2–5 and their proved costs. The representation below specifies initialization and actions at every augmented state. □

Internal representation and execution of π_k. PLAN implements the policy returned to Algorithm 1 by storing m_k deterministic Markov policies $\pi_k^{(j)}$, $j \in [m_k]$, and probabilities $w_{k,j} \ge 0$ summing to one. Each stored policy has action maps $\pi_{k,h}^{(j)} : \mathcal{S} \times [0,2]^A \times \{0,1\} \to \mathcal{A}$, $h \in [H]$. Initializing π_k samples j with probability $w_{k,j}$ once per episode. Each subsequent query returns $\pi_{k,h}^{(j)}(s_h, \widehat{u}_h, z_h)$ with this same index. The index is internal policy memory; Algorithm 1 only initializes and queries π_k.

Each additive response is a finite episode mixture of the stored policies constructed in Appendix E.3. For a returned response π_l, let m_l be its number of stored policies, J_l its number of model episodes, and $n_{l,j}$ the number of those episodes that execute its jth stored policy. Thus $\sum_{j=1}^{m_l} n_{l,j} = J_l$. If Algorithm 6 returns this response alone, set $m_k = m_l$ and

$$w_{k,j} = \frac{n_{l,j}}{J_l}, \qquad j \in [m_l].$$

If it returns $(1-\omega)\pi_l + \omega\pi_u$, define $m_u, J_u, n_{u,j}$ in the same way for the second response. List the m_l policies from π_l first and the m_u policies from π_u second. Then $m_k = m_l + m_u$ and

$$w_{k,j} = \begin{cases} (1-\omega)n_{l,j}/J_l, & 1 \le j \le m_l, \\ \omega n_{u,j-m_l}/J_u, & m_l < j \le m_l + m_u. \end{cases}$$

Here j only labels a stored policy in the returned list; the two model-episode budgets may differ. These counts are obtained during planning in the fitted model, before collecting the B real episodes.

The coefficient ω is computed by the second bisection in Algorithm 6. At each trial, fresh fitted-model episodes test whether $(1-\omega)c(\pi_l)+\omega c(\pi_u)$ is above, below, or within the prescribed tolerance of $p_0 = p(\bar{\gamma})$. If the marking probabilities were known exactly and $c(\pi_l) < p_0 < c(\pi_u)$, the exact solution would be

$$\omega = \frac{p_0 - c(\pi_l)}{c(\pi_u) - c(\pi_l)}.$$

Algorithm 6 computes the coefficient using probability tests with the stated stopping rule.

The resulting weights sum to one. Sampling the combined list once per episode has the same trajectory law as the two successive mixture draws. Consequently, for any physical model P,

$$V_{P,r}^{\pi_k} = \sum_{j=1}^{m_k} w_{k,j} V_{P,r}^{\pi_k^{(j)}}, \qquad C_k(\pi_k) = \sum_{j=1}^{m_k} w_{k,j} C_k(\pi_k^{(j)}).$$

The guarantee concerns $J_k(\pi_k)$, with the mixture marking probability inside the square root. The stored action maps and weights remain fixed during the B real episodes. Only the physical state, flag, and fresh radius estimates change within an episode.

E.5.1 Occupancy equivalence to a randomized Markov policy

The policy returned by PLAN is already a randomized policy. Its internal index can create dependence on earlier observations. For a fixed fitted augmented model, a randomized Markov policy can attain the same stagewise state-action distributions. This is the finite-horizon form of occupancy equivalence; see also Laroche & Tachet des Combes (2023) for general state spaces. The equivalence preserves the quantities in J_k, since marking is recorded in the augmented state.

General equivalence. For a policy π, let $\widehat{d}_h^{\pi}(d\tilde{s}, a)$ be its fitted stage-h joint law of the augmented state and action, and let $\widehat{d}_h^{\pi}(d\tilde{s})$ be the state marginal. Since the action set is finite, each action-specific measure is absolutely continuous with respect to this marginal. Define

$$\pi_h^{\mathrm{M}}(a \mid \tilde{s}) = \frac{d\widehat{d}_h^{\pi}(\cdot, a)}{d\widehat{d}_h^{\pi}(\cdot)}(\tilde{s}), \qquad \tilde{s} = (s, \widehat{u}, z). \tag{83}$$

Thus π_h^{M} is the fitted conditional action distribution given the current augmented state. The derivatives are Radon–Nikodym derivatives; they exist without a density with respect to Lebesgue measure. Set any action distribution on the common null set where their values are unspecified.

Proposition 31 (Markov policy with the same planning objective)**.** *For a fixed fitted augmented model, the policy in* (83) *has the same joint law of* $(\tilde{s}_h, a_h)$ *as* π *at every stage. Consequently,*

$$V_{\widehat{P}^{(k)},r}^{\pi^{\mathrm{M}}} = V_{\widehat{P}^{(k)},r}^{\pi}, \qquad C_k(\pi^{\mathrm{M}}) = C_k(\pi), \qquad J_k(\pi^{\mathrm{M}}) = J_k(\pi).$$

This statement does not require equality of the full trajectory laws.

Proof of Proposition 31. The initial augmented-state law is the same. If the two policies have the same stage-h state marginal, multiplying it by (83) gives the original joint state-action law. Averaging the shared augmented transition kernel under this joint law gives the same next-state marginal. Induction proves equality through stage $H+1$. Expected original rewards therefore agree, and $C_k(\pi) = \mathbb{P}(z_{H+1} = 1)$ also agrees. Substitution in the definition of J_k proves the last identity. □

Relation to the implemented planner. If this conversion were computed inside each PLAN call, before collecting its B real episodes, the converted policy would satisfy the same J_k guarantee. The statistical proof would apply to that sequence of policies without requiring equality with the original policies' true-model values. The occupancy equivalence is specific to the fitted model; it does not justify replacing the final policy after training while assuming its true-model value is unchanged.

Proposition 31 is an existence result. The fitted-model interfaces do not directly return occupancy measures. An implementation of the conversion would require an additional computational and approximation-error analysis. Algorithm 6 therefore returns the executable randomized policy specified above, with its internal episode initialization. Its guarantee and computational cost do not rely on a Markov conversion.

E.6 COMBINED COMPUTATIONAL COST

Algorithm 1 makes at most $K = \widetilde{O}(H^3 d/\epsilon^2)$ MLE calls in the homogeneous setting and $HK = \widetilde{O}(H^4 d/\epsilon^2)$ stagewise MLE calls in the inhomogeneous setting.

We first count the general-state implementation, which uses the internal episode mixture. Ignoring logarithms, $n = \widetilde{O}(H^3/\epsilon^2)$, each additive response uses $J = \widetilde{O}(d^3 H^6/\epsilon^4)$ model episodes, and all probability tests in one full plan use $\widetilde{O}(H^2/\epsilon^2)$ episodes. One augmented episode uses $O(H(1 + An))$ fitted physical-kernel samples, including radius estimation. Let N_{SAMP} be the total number of fitted physical-kernel samples. Then

$$N_{\mathrm{SAMP}} = \widetilde{O}\left(KA\left[\frac{d^3H^{10}}{\epsilon^6} + \frac{H^6}{\epsilon^4}\right] + \frac{KBAH^4}{\epsilon^2}\right),$$

Substituting $K = \widetilde{O}(H^3 d/\epsilon^2)$ and $KB = \widetilde{O}(H^3 dA\Xi/\epsilon^2)$ gives

$$N_{\mathrm{SAMP}} = \widetilde{O}\left(\frac{d^4AH^{13}}{\epsilon^8} + \frac{dAH^9}{\epsilon^6} + \frac{dA^2H^7\Xi}{\epsilon^4}\right). \tag{84}$$

Regression arithmetic contributes $\widetilde{O}(KAd^6H^8/\epsilon^4) = \widetilde{O}(d^7AH^{11}/\epsilon^6)$, while radius sorting and inner products cost $O(N_{\mathrm{SAMP}}(d + \log n))$. Historical re-encoding uses $O(K^2BH)$ fitted-feature evaluations. Model-interface evaluation costs must be added according to the supplied representation.

The oracle-call bounds exclude the internal runtime of global MLE and numerical bit complexity. Let $T_{\mathrm{MLE}}(n, \tau)$ be the runtime of an MLE call on n transitions with total-loss tolerance $\tau = \log(|\Theta|/\delta)$; additional subscripts specify the stage or product class. In the homogeneous case it is $\sum_{k=1}^{K} T_{\mathrm{MLE}}((k-1)BH, \tau)$. With stagewise fitting it is $\sum_{k=1}^{K}\sum_{h=1}^{H} T_{\mathrm{MLE},h}((k-1)B, \tau)$; joint product-class MLE instead costs $\sum_{k=1}^{K} T_{\mathrm{MLE,prod}}((k-1)BH, \tau)$. Empty-data fits require no call. Each tolerance concerns total loss: on a nonempty data list of size n, the corresponding average-loss tolerance is τ/n. A constant average-loss tolerance or a stationary-point guarantee does not suffice. Output regression parameters require $\widetilde{O}(Kd^3H^2)$ storage, in addition to fitted models, historical sums, and training data. Each iteration stores one shared fitted kernel or H stagewise kernels. The mixture changes only at iteration boundaries, but its sampled policy index can differ between episodes.

F COMPARISON WITH RELATED WORK

We give the conventions and reductions used in Table 1, followed by comparisons for other learning objectives. For each result, we specify the model class, reward normalization, sample unit, and optimization oracle.

F.1 EPISODIC PAC BOUNDS IN TABLE 1

All rows in Table 1 use policy-value error ϵ with per-step rewards in $[0, 1]$. In both settings, Θ is the common class of candidate kernels. The complete model class is Θ for a shared kernel and Θ^H for stage-dependent kernels. The reward is known for the comparison, although RFOLIVE and

RAFFLE support stronger reward-free guarantees. Logarithmic confidence factors are suppressed. The table assumes $|\Theta| \geq 2$, so $1 + \log|\Theta|$ and $\log|\Theta|$ differ by at most a constant factor; the precise theorems also cover singleton classes.

Oracle convention. The oracle column records the optimization access used by each algorithm. PP denotes planning in a model and PE denotes policy evaluation; MLE remains explicit when either interface is also used. OLIVE, witness-rank elimination, and RFOLIVE optimize over classes subject to empirical error constraints. OPS denotes sampling from an optimistically tilted model posterior. Neither interface is an ordinary MLE objective; no general polynomial reduction to our MLE interface is provided by these works. Our algorithm implements planning using fitted-model evaluations and samples, as proved in Appendix E. Sampling and density access are stated separately from optimization oracles. A dash in a lower-bound row means that the learner's computation is unrestricted.

OLIVE and witness-rank elimination. Let ϵ_Σ denote policy-value error when total reward is at most one. Theorem 1 of Jiang et al. (2017) and Theorem 6 of Sun et al. (2019) give $\widetilde{O}(H^3 d^2 A \log|\Theta^H|/\epsilon_\Sigma^2)$ episodes under unit total reward after specialization to low-rank MDPs. For OLIVE, take the optimal action-value functions of the candidate models as the value class. Bellman rank is at most d. For witness-rank elimination, let V_{h+1}^θ be the optimal remaining value in candidate model θ. Tests $\pm(r_h + V_{h+1}^\theta)$ give Bellman domination with coefficient one, and the Bellman-error matrix has rank at most d by factoring the stage-h state distribution through the preceding true kernel. Both class entropies are $O(\log|\Theta^H|)$. Substituting $\epsilon_\Sigma = \epsilon/H$ and $\log|\Theta^H| = H\log|\Theta|$ gives the two H^6 rows. These algorithms optimize over their surviving classes, beyond planning in one fitted model.

MOPS. Let Θ_{full} be the class of complete models. The finite-dimensional witness specialization of Agarwal & Zhang (2022), Proposition 3 and the posterior-estimation analysis preceding Corollary 1, yields expected policy-mixture gap ϵ_Σ using $O(H^2 dA \log|\Theta_{\mathrm{full}}|/\epsilon_\Sigma^2)$ episodes. The low-rank Bellman factorization above has witness coefficient one. The algorithm samples from a likelihood posterior tilted by model value and assumes model planning; the output mixture uses the sampled models' policies, without the added exploration action. Reward rescaling gives H^4 for a shared class and H^5 for a product of stage classes. To obtain the high-probability rows, run $O(\log(1/\delta))$ independent copies with expected gap at most $\epsilon/8$, evaluate each output to accuracy $\epsilon/8$, and return the best estimate. Markov's inequality and a union bound give gap at most ϵ with probability $1-\delta$. The additional $\widetilde{O}(H^2/\epsilon^2)$ evaluation episodes, including confidence logarithms, are dominated by the displayed rates.

RFOLIVE. Let Φ_h be the stage-h feature class, $\Phi_{\mathrm{prod}} = \prod_h \Phi_h$, and $\mathcal{R}$ the reward class. Write $N_{\mathcal{R}}(\kappa)$ for its covering number at the scale κ specified in Appendix D.3 of Chen et al. (2022). Their Corollary 4 and that full statement give the episode bound

$$\widetilde{O}\left(\frac{A}{\epsilon^2}\left[H^8 d^3 \log|\Phi_{\mathrm{prod}}| + H^5 d^2 \log N_{\mathcal{R}}(\kappa)\right]\right),$$

where, for one known reward, $N_{\mathcal{R}}(\kappa) = 1$. Taking Φ_h to be the left-factor projection of Θ at every stage gives $|\Phi_h| \leq |\Theta|$, and hence $\log|\Phi_{\mathrm{prod}}| \leq H\log|\Theta|$. This yields the $\widetilde{O}(H^9 d^3 A \log|\Theta|/\epsilon^2)$ row. The additional H comes from expressing a full sequence class in terms of per-stage class sizes. RFOLIVE itself requires only the left-factor classes; it need not learn the transition densities. Its elimination and optimization steps are not implemented with a polynomial number of the MLE and fitted-model calls assumed here.

RAFFLE. Let Φ and Ψ be the two factor classes available at each stage, with $\Theta = \Phi \times \Psi$. Theorem 2 of Cheng et al. (2023) gives

$$\widetilde{O}\left(\frac{H^3 d^2 A(d^2 + A)\log(|\Phi||\Psi|)}{\epsilon_\Sigma^2}\right)$$

trajectories when total reward is at most one. For rewards $r_h \in [0,1]$, apply the guarantee to r_h/H with target error $\epsilon_\Sigma = \epsilon/H$. This yields the H^5 dependence in Table 1. The full factor $d^2 + A$ is

retained. RAFFLE uses MLE and assumes access to planning in the learned model; the table lists PP + MLE. It supports rewards revealed after exploration.

Opt-AC. Theorem 5.2 of Huang et al. (2026) gives $\widetilde{O}(H^5 d^3 A^2 \log|\Theta|/\epsilon^2)$ trajectories with per-step bounded rewards. Their Assumption 3.1 defines $\Theta = \Phi \times \Psi$ for the factor classes used at each stage, so no reward or class-size conversion is needed. The table lists PE + MLE: the algorithm learns representations with MLE and uses policy evaluation for its critic. The policy-evaluation reduction uses supervised learning and sampling from a distribution satisfying coverage conditions; these computational conditions are additional to model realizability.

Our bounds. Theorems 3 and 4 retain the exact entropy and confidence terms. In the inhomogeneous row, $\log|\Theta^H| = H \log|\Theta|$ gives the displayed rate. The lower-bound notation uses the same per-kernel cardinality convention. Our implementation uses global approximate MLE, evaluation of both fitted factors and the fitted density, and sampling at arbitrary inputs to the fitted model. The lower bounds apply even when the right factor is known and computation is unrestricted. Their class-size and horizon ranges are stated in Appendices C.1 and D.1; the tildes in the table do not assert a sharp confidence coefficient in the inhomogeneous case.

F.2 Other statistical guarantees

Table 3 records several central results in their original error scales and sample units. Here Θ denotes the candidate class used by the algorithm, as specified below, and the class-dependent rows assume $|\Theta| \geq 2$. In particular, GOLF uses a Bellman-complete value class, rather than a per-kernel transition class. The errors are ϵ_Σ for unit-total-reward policy error, ϵ_{TV} for one-step TV error, ϵ_γ for discounted policy error with discount $\gamma \in (0,1)$, and ϵ_{cov} for cover accuracy. Let C_{cov} be the coverability coefficient, the smallest uniform bound on state-action occupancy density ratios over covering distributions, and write $|\mathcal{S}|$ for the number of states in the tabular rows.

Table 3: Additional sample-complexity guarantees. REP-UCB counts discounted interaction steps; the other rows count episodes, with expectations for tabular lower bounds. GOLF and the low-rank lower bound use unit total reward. FLAMBE and VoX concern system identification and policy covers. Classes and error scales differ across rows and are specified in the text.

Algorithm / result	Time-homogeneous	Strongest oracle	Sample complexity
GOLF	No	Constrained opt.	$\widetilde{O}(H^2 C_{\mathrm{cov}} \log\|\Theta\|/\epsilon_\Sigma^2)$
FLAMBE	No	MLE	$\widetilde{O}(H^{22} d^7 A^9 \log\|\Theta\|/\epsilon_{\mathrm{TV}}^{10})$
REP-UCB	Yes	MLE	$\widetilde{O}(d^4 A^2 \log\|\Theta\|/((1-\gamma)^5 \epsilon_\gamma^2))$
VoX	No	Minimax opt.	$\widetilde{O}(H^6 d^{13} A^2 (d + \log\|\Theta\|)/\epsilon_{\mathrm{cov}}^2)$
Lower bound (Cheng et al., 2023)	No	—	$\Omega(HdA/\epsilon_\Sigma^2)$
Lower bound (Domingues et al., 2021)	Yes	—	$\Omega(H^2 \lvert\mathcal{S}\rvert A \log(1/\delta)/\epsilon^2)$
Lower bound (Domingues et al., 2021)	No	—	$\Omega(H^3 \lvert\mathcal{S}\rvert A \log(1/\delta)/\epsilon^2)$

GOLF and E2D. Theorem 1 and Corollary 2 of Xie et al. (2023) analyze GOLF (Jin et al., 2021) under Bellman completeness. With their full value class $\mathcal{F}$ denoted by Θ in Table 3, the PAC guarantee is $\widetilde{O}(H^2 C_{\mathrm{cov}} \log|\Theta|/\epsilon_\Sigma^2)$. This gives $\widetilde{O}(H^4 C_{\mathrm{cov}} \log|\mathcal{F}|/\epsilon^2)$ with our reward convention. The algorithm solves a global constrained optimistic problem. Transition realizability does not by itself make the candidate optimal value functions Bellman complete, so their class cardinality cannot simply replace the transition-class cardinality in Table 1.

For E2D, let T be the number of decision rounds, $\gamma > 0$ a tuning parameter, dec_γ the DEC maximized over the estimator's possible reference models, and Est_H the cumulative Hellinger estimation error. Theorem 4.1 of Foster et al. (2021) bounds regret by $T\,\mathrm{dec}_\gamma + \gamma\,\mathrm{Est}_H$. Finite-class aggregation controls the latter by a class logarithm. Computing the decision distribution also requires DEC minimax optimization. The bilinear specializations distinguish on-policy estimation from forced exploration; one cannot apply an on-policy rate to arbitrary unknown-feature low-rank MDPs without verifying that condition.

FLAMBE and REP-UCB. Theorem 2 of Agarwal et al. (2020) guarantees

$$\sup_{\pi,h} \mathbb{E}_{P^\star,\pi} D_{\mathrm{TV}}\big(P_h^\star(\cdot \mid s_h, a_h), \widehat{P}_h(\cdot \mid s_h, a_h)\big) \le \epsilon_{\mathrm{TV}}.$$

Its model class is $\Theta = \Phi \times \Upsilon$, where Φ and Υ are the state-action and next-state factor classes. This is a uniform one-step model-error guarantee; its accuracy parameter is not a policy-value error. Theorem 4 of Uehara et al. (2022) gives the discounted PAC row, with their model class $\mathcal{M}$ denoted by Θ in the table and total discounted reward bounded by one. It counts interaction steps, including the discounted roll-in. Both algorithms use MLE and implement planning with known fitted features and model samples, so their optimization-oracle entry is MLE. Replacing $(1-\gamma)^{-1}$ by H in that theorem would not, by itself, give an episodic guarantee in the units of Table 1.

VoX and MOFFLE. Theorem 3.2 in the revised arXiv version of Mhammedi et al. (2024) gives the VoX row, with their feature class Φ denoted by Θ in the table. It constructs a $(1/(8Ad), \epsilon_{\mathrm{cov}})$-policy cover, in the sense of their Definition 2.1, without a reachability assumption. The algorithm uses a minimax representation-learning oracle and policy optimization based on regression. The conference version, SpanRL, imposed reachability; the table uses the revised result. Turning an approximate policy cover into a policy-value guarantee requires an additional downstream learning step and an appropriate cover accuracy, so the table keeps the cover criterion explicit.

MOFFLE studies a related minimax representation-learning approach under latent-state reachability. Let $\eta_{\min}$ be its reachability parameter, $\Phi = \bigcup_h \Phi_h$ the union of stagewise feature classes, and $\mathcal{R}$ a finite reward class. Theorem 4 of Modi et al. (2024), for the oracle solving their Equation (5), gives

$$\widetilde{O}\left(\frac{H^6 d^{11} A^{14} \log(e|\Phi|)}{\eta_{\min}^5} + \frac{H^7 d^3 A^5 \log(e|\Phi||\mathcal{R}|)}{\epsilon^2 \eta_{\min}}\right)$$

episodes. Their other oracle implementations have different guarantees. Table 4 records the reachability requirement separately from the rank d.

Earlier lower bounds. Theorem 1 of Cheng et al. (2023) gives the low-rank reward-free lower bound in Table 3, under its stated horizon, action, and state-size conditions. The unit-total-reward convention is retained. Theorem 7 of Domingues et al. (2021) gives $\Omega(H^3|\mathcal{S}|A\log(1/\delta)/\epsilon^2)$ expected episodes for tabular best-policy identification with stage-dependent kernels, under their Assumption 1 and parameter range. Their Appendix C gives $\Omega(H^2|\mathcal{S}|A\log(1/\delta)/\epsilon^2)$ for shared kernels. Both use per-step bounded rewards. Our lower bounds address a different complexity term: finite representation classes with rank bounded independently of both $|\mathcal{S}|$ and H, and an explicit dependence on $\log|\Theta|$.

F.3 Representation, coverage, and computation

Table 4 summarizes the remaining related-work conclusions. General structural results explain why learning is possible; representation-learning and coverage results identify sufficient inputs to an exploration algorithm; regret and oracle-separation results address different performance or computational questions.

Table 4: Related settings and the conclusions relevant to this paper. The table complements the numerical comparisons in Tables 1 and 3.

Work	Main conclusion	Relation to our assumptions or goal
OLIVE, witness-rank elimination, bilinear classes (Jiang et al., 2017; Sun et al., 2019; Du et al., 2021)	Sample efficiency under Bellman or witness factorizations.	Global constrained optimism; low-rank rates for the first two methods appear in Table 1.
MOPS (Agarwal & Zhang, 2022)	Linear dA dependence for realizable low-rank models.	Optimistic posterior sampling and model planning; confidence amplification is included in Table 1.
E2D (Foster et al., 2021)	Regret bounded by $T\,\mathrm{dec}_\gamma + \gamma\,\mathrm{Est}_H$.	DEC minimax optimization is needed in addition to online estimation.
MOFFLE (Modi et al., 2024); BRIEE (Zhang et al., 2022)	Model-free representation learning with exploration.	MOFFLE uses latent-state reachability; BRIEE assumes block structure. Both use state-action feature classes.
Density features (Huang et al., 2023)	Occupancy estimation and exploration using known or learned next-state features.	A class for the right factor suffices for their representation-learning result; our fitted-model access uses both factors.
GOLF (Jin et al., 2021; Xie et al., 2023); GLOW (Amortila et al., 2024a)	GOLF has linear coverability dependence; GLOW uses density-ratio and value realizability.	GOLF uses Bellman completeness and constrained optimism. GLOW gives no oracle-efficient online implementation.
CODEX (Amortila et al., 2024b)	Coverage objectives admit policy-optimization reductions and exploration guarantees under coverability.	Includes model-based and model-free methods. Our bonus directly controls fitted-model error for a known reward.
Rohatgi & Foster (2025)	Regression-oracle characterizations for block MDPs, with cryptographic evidence against the analogous oracle for low-rank MDPs.	Our computation bound explicitly assumes global MLE and fitted-model evaluation and sampling.
UniSREP-UCB (Sturm & Tschiatschek, 2025)	Constant regret under additional gap, optimal-occupancy, and representation assumptions.	An instance-dependent result with known gap and occupancy parameters; we study worst-case PAC complexity.
Liu et al. (2024)	$\mathrm{poly}(H, d, A)T^{2/3}$ regret with unknown transitions and adversarial losses; bandit feedback requires linear loss structure.	The $T^{2/3}$ bandit algorithms are computationally inefficient; oracle-efficient model-free bounds have $T^{4/5}$ dependence.

G Experimental details and additional results

This appendix specifies the environment, neural implementation of uncertainty marking, baselines, and evaluation protocol for Section 6. We then explain the learning curves, report interaction counts and runtime, and describe additional diagnostics. All results come from completed runs; the action-count and training-budget extensions reuse the original four seeds. We use the notation of Algorithm 1: $k \in [K]$ indexes iterations, each iteration collects B episodes, and $N = KB$ counts real training episodes.

G.1 HOMOGENEOUS CONTINUOUS-STATE ENVIRONMENT

Let the observation dimension be $D = 64$, the chain length be 24, and $H = 48$. Sample a random orthogonal matrix in $\mathbb{R}^{D\times D}$ and multiply its first 26 rows by 3 to obtain centers $c_0, \ldots, c_{25}$. Let $\mathcal{B}_j$ be the Euclidean ball of radius 0.5 centered at c_j. The balls are disjoint since distinct centers have distance $3\sqrt{2}$. Blocks $0, \ldots, 23$ are progress blocks, block 24 is safe, and block 25 is the goal. The initial state is $s_1 = c_0$.

For a state s, let $j(s)$ be its nearest-center index, with a fixed tie rule. Sample one correct action $a_j^\star$ independently and uniformly from $\mathcal{A}$ for each progress block, once when constructing the environment. These actions remain fixed across stages and episodes; they are not resampled at each visit. Define the destination $g(s, a)$ as $j(s) + 1$ for a correct action in blocks $0, \ldots, 22$, as 25 for a correct action in block 23, and as 24 for an incorrect action in any progress block. Safe and goal blocks transition to themselves.

Let $b_j(S) = \mathbf{1}\{S \in \mathcal{B}_j\}/\operatorname{vol}(\mathcal{B}_j)$ be the uniform density on block j. The kernel is

$$P^\star(S \mid s, a) = \sum_{j=0}^{25} \mathbf{1}\{g(s, a) = j\} b_j(S).$$

Thus $\phi^\star(s, a)$ is a coordinate vector and $\mu_j^\star(S) = b_j(S)$. The factorization has dimension 26, satisfies Definition 1, and is independent of the stage. Here each coordinate of $\sum_{S\in\mathcal{S}} \mu^\star(S) f(S)$ lies in $[0, 1]$, so its norm is at most $\sqrt{26}$, using the continuous-state convention in Section 2. Every transition draws a fresh observation, including transitions within the safe and goal blocks. The next-state distribution depends only on the observed (s, a); there is no additional hidden state. The learner receives the continuous observation, not a block index, center, or correct-action table.

The reward is 0 in progress blocks, 0.2 in the safe block, and 1 in the goal block, paid before each transition. An optimal policy advances for 24 steps and receives reward 1 for the remaining 24, so $V^\star = 24$. Immediately choosing a wrong action gives value $0.2(48 - 1) = 9.4$. If an episode advances correctly j times and then fails, its return is $0.2(47 - j)$ for $0 \le j \le 23$. Consequently, reaching deeper blocks can initially decrease return. Uniform independent actions reach the goal with probability A^{-24} per episode. The safe reward therefore makes this an exploration problem with a suboptimal alternative, rather than a task whose return improves with every additional correct action. We retain temporary decreases in all learning curves.

G.2 LEARNED TRANSITION MODEL AND PLANNING

For the main comparison, the fitted model is

$$\widehat{P}^{(k)}(S \mid s, a) = \sum_{j=1}^{26} \widehat{\phi}_j^{(k)}(s, a) \mathcal{N}\Big(S; m_j^{(k)}, \operatorname{diag}((\sigma_j^{(k)})^2)\Big).$$

The vectors $m_j^{(k)}$ and $\sigma_j^{(k)}$ are the mean and coordinate standard deviations of component j. A network with two ReLU hidden layers of width 128 takes the continuous state and one-hot action, an input of dimension $64 + A$, and produces 26 softmax weights. The means and diagonal standard deviations are learned jointly with this network. Standard deviations are constrained to $[0.05, 5]$. The networks use ordinary random initialization; latent labels and true factors are never supplied to fitting or planning.

The first iteration plans with a randomly initialized model. At every later iteration, fitting starts from the previous parameters and precedes planning and collection. Adam minimizes the unweighted negative log likelihood over minibatches drawn uniformly from all previously collected transitions. The learning rate is 0.003, minibatch size is 1,024, and each fit uses 100 gradient updates with gradient norm clipped at 10. Fitting uses all H transitions, while the radius history excludes final-stage inputs, as in Algorithm 1. There is no update after the last collection batch, and no final refit changes the reported output.

Numerical integration and the bonus. At each iteration, draw 16 points from each fitted Gaussian, giving 416 integration points $S_1, \ldots, S_{416}$. Let $\widehat{\mu}_j^{(k)}$ be its fitted density, and define the proposal

density $p^{(k)}_{\mathrm{quad}}(S) = 26^{-1}\sum_{j=1}^{26}\widehat{\mu}^{(k)}_j(S)$. The normalized weights for component j are

$$w^{(k)}_{j,\ell} = \frac{\widehat{\mu}^{(k)}_j(S_\ell)/p^{(k)}_{\mathrm{quad}}(S_\ell)}{\sum_{v=1}^{416}\widehat{\mu}^{(k)}_j(S_v)/p^{(k)}_{\mathrm{quad}}(S_v)}, \qquad \ell \in [416].$$

We also include the initial state as a 417th planning point with zero integration weight. Thus the numerical transition probability to point S_ℓ from an arbitrary (s,a) is $\sum_{j=1}^{26}\widehat{\phi}^{(k)}_j(s,a)w^{(k)}_{j,\ell}$. The historical density is computed by summing these probabilities over all previously collected $(s_{i,h},a_{i,h})$ with $h<H$. Sorting the current-to-historical density ratios then solves the radius optimization on these points. The marking probability is the resulting radius clipped at one, and it is zero at $h=H$. The integration points stay fixed within an iteration. This approximates the radius using numerical integration instead of drawing fresh, certified radius estimates at each visit.

The archived implementation uses a confidence multiplier 0.01, a model-count proxy 16, and nominal confidence level 0.05. In terms of the paper's notation, its numerical radius budget is $0.01(32AH+8)\log(12\cdot 16/0.05)$, and its initial-state correction is $\rho_1 = 2$ and $\rho_k = \min\{2, 0.01\cdot 16AH\log(12\cdot 16/0.05)/[(k-1)B]\}$ for $k\geq 2$. These are the practical settings used in the saved runs, rather than the parameter choices in the upper-bound theorems. The proxy 16 is not the cardinality $|\Theta|$ of the neural class.

Planning and execution. For each slope γ, backward induction solves the additive first-mark objective on the integration points. Slope bisection and an episode-level mixture then optimize the square-root objective, as in Appendix E.5. The implementation uses planning input 0.05, sets the regularization to $(0.05)^2/(10^4H^2)$, omits a certified Monte Carlo correction, and stops when its numerical upper bound minus the achieved objective is at most $0.05/100$. This certificate concerns the numerical model; it does not bound integration or density-estimation error.

The policy is evaluated at each actual continuous observation through $\widehat{\phi}^{(k)}(s,a)$ and the stored continuation coefficients. It does not replace that observation by its nearest integration point. At the start of an episode, it draws its internal policy randomization and sets $z_1 = 0$. At each stage, it chooses an action and then draws the first-mark coin for that action; physical transitions and rewards continue after a mark. The reward depends only on the current state, so it is the same for all actions in each action comparison. The learner is not given the simulator's nearest-center map, although the model methods can query the known reward function at the integration points.

The no-bonus ablation uses the same transition family, fitting schedule, initialization rule, reward interface, and collection exploration, with $\gamma = 0$ in planning. Data subsequently differ because the learned policies differ. Neural likelihood optimization uses a fixed number of local Adam updates, and Gaussian densities do not exactly represent the uniform-ball emissions. Together with finite integration and practical confidence scaling, these differences mean that the experiment is not an implementation of the certified oracles assumed in the theory.

G.3 Baselines and training protocol

We compare PPO (Schulman et al., 2017), Double DQN (Van Hasselt et al., 2016), Double DQN with a remaining-horizon prior, and Bootstrapped DQN with a random prior (Osband et al., 2016; 2018). All use the same continuous observation and have no access to latent labels. Policy and Q networks additionally receive the stage as a one-hot vector, for input dimension $64+48=112$. All networks have two hidden layers of width 128; PPO uses tanh activations and DQN uses ReLU. The networks are trained separately and do not share fitted encoder weights with the transition model. All methods use undiscounted finite-horizon returns.

PPO uses separate actor and value networks, Adam with learning rate 0.001, clipping parameter 0.2, entropy coefficient 0.05, and GAE parameter 1. It performs at most 10 epochs per batch with minibatches of size $1{,}024$, stopping epochs when the approximate KL exceeds 0.03. The value-loss coefficient is 0.5 on one-half the mean squared value error, and the gradient clipping threshold is 0.5.

At each iteration after the first, DQN uses 64 gradient steps, Adam with learning rate 0.001, minibatches of size $1{,}024$, and a target-network update every 16 gradient steps. Standard Double DQN

decreases its collection exploration probability from 1 to 0.05 over 4,096 episodes. The horizon-prior version adds the remaining horizon to learned action values and uses constant exploration probability 0.1. Bootstrapped DQN uses 10 heads, a fixed random prior with scale 3, one head sampled per episode, and fixed Bernoulli-0.5 replay masks for each transition and head; it has no additional uniform exploration. DQN uses the smooth-L_1 loss and gradient clipping at 10. These are implementations of the cited approaches, not oracle versions of the theoretical methods in Table 1.

Baseline hyperparameters were chosen in earlier, smaller continuous-chain validation runs with seeds 620–621, then transferred unchanged. The reported block experiments use seeds 1040–1043, paired across methods. Observation geometry is paired across action counts; the action table is fixed within each environment. We use $B = 128A$ and $K = 128$ for $A = 2, 4, 8$. The corresponding training budgets are 32,768, 65,536, and 131,072 episodes. The $A = 8$ extension uses $K = 256$ and 262,144 episodes for every method. The first 128 checkpoints of each extended run reproduce its archived shorter run, excluding timing.

Table 5: Interaction budgets shared by all six methods. Every condition uses four seeds.

A	B	K	Training episodes	Training transitions	Output policies
2	256	128	32,768	1,572,864	4
4	512	128	65,536	3,145,728	2
8	1,024	128	131,072	6,291,456	1
8	1,024	256	262,144	12,582,912	1

The model methods execute a uniformly random action with probability $1/(2H)$ during collection. They can also query the known reward at integration points; PPO and DQN use rewards observed in real interaction. The bonus/no-bonus comparison therefore holds the reward information fixed. Comparisons to model-free baselines also reflect this difference in information access. Increasing A increases both batch size and data, and changes the model input/output dimensions. These runs do not isolate an exponent of A in sample complexity.

Table 6: Independent final returns on random-action chains, mean ± seed standard error. The first three columns use $K = 128$ iterations and $B = 128A$ episodes per iteration; the last doubles K at $A = 8$. The optimum is 24.

Method	$A = 2$	$A = 4$	$A = 8$	$A = 8$, $K = 256$
Ours (neural)	24.00 ± 0.00	24.00 ± 0.00	10.24 ± 4.59	23.93 ± 0.07
MLE, no bonus	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00
PPO	9.39 ± 0.00	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00
Double DQN	9.39 ± 0.01	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00
DQN + horizon prior	9.32 ± 0.05	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00
Bootstrapped DQN + prior	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00	9.40 ± 0.00

G.4 Output, evaluation, and computational cost

The primary output samples one of the policies used in the last 1,024 training episodes, with probability proportional to its episode count, and follows it for a full episode. It contains four, two, and one batch policies at $A = 2, 4, 8$, respectively. Evaluation removes externally added uniform exploration but retains PPO action sampling, the bootstrap head, and the planner's internal randomization. The output is not selected by its evaluation performance. Each saved output receives 8,192 fresh evaluation episodes, divided equally among its batch policies, with independent random streams. These interactions never enter training.

Checkpoint evaluations use 256 episodes per batch policy. Learning curves average these estimates over the preceding 1,024 training episodes and omit incomplete initial windows. They apply no further smoothing. The final table uses a separate independent evaluation, so its values can differ from the curve endpoints. Shading and reported ± values are one standard error across four training seeds, not across evaluation episodes. Four seeds give limited information about variation over environments and initialization. The 8,192 evaluation episodes improve the accuracy of each policy-value estimate; they are not additional training seeds.

We normalize the value gap as $(24 - V)/14.6$ and set the descriptive target to 0.1, equivalent to value at least 22.54. The normalized gap is not clipped at one. The stable-target episode count is the first complete window meeting the target at that and every later recorded checkpoint, with at least $1{,}024$ further training episodes. This statistic uses later observations; it is not an online stopping rule or a PAC guarantee. Table 7 reports the largest such count across seeds. An entry $> N$ means that not all seeds met this stability criterion within the observed budget N; it is a censored measurement, not a lower bound on the method's sample complexity.

Table 7: Independent final return, number of seeds reaching the final target, all-seed stable-target episode count, and mean training time. The three blocks use random correct actions with $A = 2$, $A = 4$, and $A = 8$; only the last uses the extended $K = 256$ budget. Every block has four paired seeds. Multiply episode counts by 48 for real transitions.

Method	Return ± SE	Target	Stable episodes	Time (s)
Random correct actions, $A = 2$, $KB = 32{,}768$				
Ours (neural)	24.00 ± 0.00	4/4	28,160	124.9
MLE, no bonus	9.40 ± 0.00	0/4	$> 32,768$	81.9
PPO	9.39 ± 0.00	0/4	$> 32,768$	24.6
Double DQN	9.39 ± 0.01	0/4	$> 32,768$	10.4
DQN + horizon prior	9.32 ± 0.05	0/4	$> 32,768$	10.8
Bootstrapped DQN + prior	9.40 ± 0.00	0/4	$> 32,768$	18.3
Random correct actions, $A = 4$, $KB = 65{,}536$				
Ours (neural)	24.00 ± 0.00	4/4	64,000	218.1
MLE, no bonus	9.40 ± 0.00	0/4	$> 65,536$	133.8
PPO	9.40 ± 0.00	0/4	$> 65,536$	45.4
Double DQN	9.40 ± 0.00	0/4	$> 65,536$	10.8
DQN + horizon prior	9.40 ± 0.00	0/4	$> 65,536$	11.1
Bootstrapped DQN + prior	9.40 ± 0.00	0/4	$> 65,536$	18.2
Random correct actions, $A = 8$, $KB = 262{,}144$				
Ours (neural)	23.93 ± 0.07	4/4	164,864	895.8
MLE, no bonus	9.40 ± 0.00	0/4	$> 262,144$	530.7
PPO	9.40 ± 0.00	0/4	$> 262,144$	171.2
Double DQN	9.40 ± 0.00	0/4	$> 262,144$	20.8
DQN + horizon prior	9.40 ± 0.00	0/4	$> 262,144$	22.0
Bootstrapped DQN + prior	9.40 ± 0.00	0/4	$> 262,144$	38.7

Runtime and sample counts. Runs execute serially on an RTX 3090 with an AMD Ryzen 9 9950X CPU, Python 3.10.20, PyTorch 2.11.0 with CUDA 13.0, and NumPy 2.2.6. Neural fitting and vectorized environment rollouts use the GPU; numerical backward induction and bisection use NumPy on the CPU. PyTorch uses one CPU thread, deterministic operations, and no TF32. Training time includes initialization, model or policy updates, planning, interaction, and copying output policies. It excludes warmup, evaluation, environment construction, and final file writes. Reused runs retain their original measured times; these timings are not simultaneous measurements under identical external load. Each model run makes $53{,}376$ extra reward queries at $K = 128$, or $106{,}752$ at $K = 256$, counted separately from real transitions. Real training transitions at $A = 8$, $K = 256$ total $12{,}582{,}912$ per seed. The comparison matches samples, not wall-clock budgets. At the extended $A = 8$ budget, our method averages 895.8 seconds, compared with 530.7 for the no-bonus model, 171.2 for PPO, and 20.8 for Double DQN. These costs include different amounts of model fitting, planning, and policy optimization; baseline failure at the shared interaction budget does not imply failure under equal computation or different hyperparameters.

Reading the learning curves. Figure 1(a,b,c) uses $A = 2, 4, 8$, respectively, with independently sampled correct actions fixed at each progress block. Curves near one correspond to the safe value; several baselines overlap there. All four runs of our method have final evaluated return 24 at $A = 2, 4$. Panel (c) keeps the failing $K = 128$ prefix and continues all methods to $K = 256$; the vertical line separates the two budgets. The horizontal ranges differ because $B = 128A$ and the last panel also doubles K. Table 6 compares the independently evaluated endpoints, including the original $A = 8$ budget.

Figure 2 separates the four runs of our method. The upper row uses the last-1,024-episode output; the lower row includes all earlier policies, weighted by their episode counts. Poor early policies remain in the latter mixture, so its gap can stay positive after the current policies perform well. Later increases in the upper row show temporary losses of performance; we retain every seed and do not select the best checkpoint. The last-window results therefore do not measure the output guarantee of Algorithm 1. Figure 3 changes the horizontal axis to measured training time. Each curve covers only the time interval observed in all four runs of that method; its endpoint is not extended to another method's runtime.

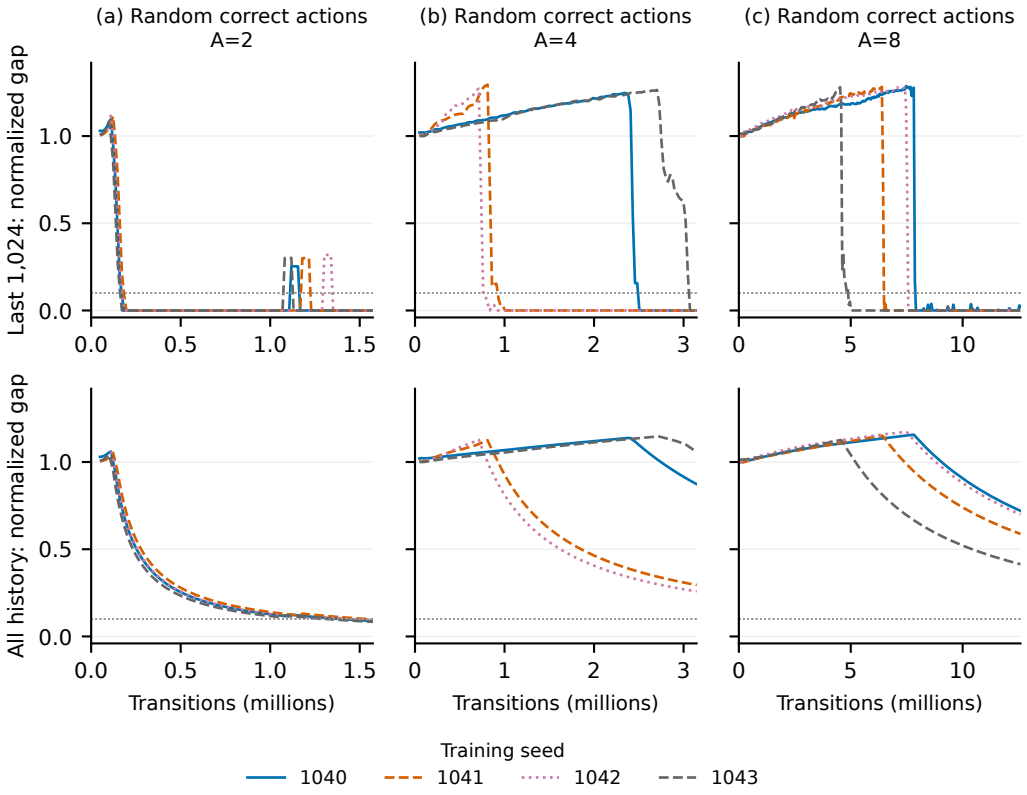


Figure 2: Every seed for the neural bonus method. The upper row uses the fixed last-1,024-episode mixture; the lower row uses the full-history mixture. Temporary increases in the gap remain visible. The two output rules can have substantially different values.

G.5 Additional diagnostics and scope

More fitting versus more online training. The $A = 8$ failure at $K = 128$ motivated two separate interventions on the same four seeds, retaining 26 Gaussian components. Increasing Adam updates from 100 to 400 per fit at the same interaction budget gives final return 14.965 ± 5.218 and $2/4$ target successes. Doubling the online iterations with 100 updates gives 23.932 ± 0.068 and $4/4$, compared with 10.242 ± 4.587 and $1/4$ originally. More iterations change both data and total optimization; they do not isolate a single cause. All five baselines were also extended to $K = 256$ without retuning, including their original exploration schedules. These are adaptive diagnostics on reused seeds, not a new independent test set.

Nonnegative neural right factors. On the $A = 2$ random-action block environment, replacing Gaussian components by normalized nonnegative spline densities gives final value gap 0.976 ± 0.976 with the bonus and $3/4$ target successes. The no-bonus gap is 10.950 ± 3.650. Both factors are learned; each component uses a product of coordinate densities. Random-action training averages 162.4 seconds with the bonus, compared with 124.9 for the Gaussian model. One saved final mixture fails after an earlier successful policy, so these results do not establish that this density family

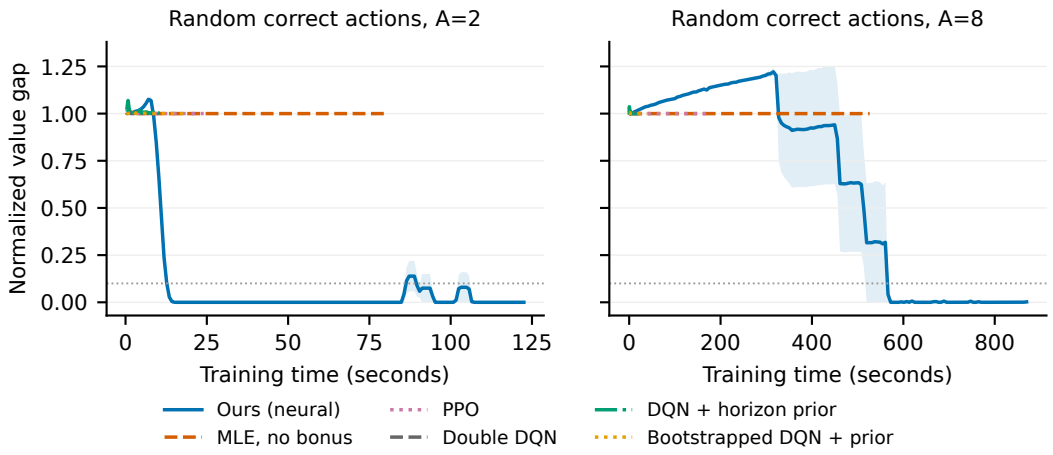


Figure 3: The same checkpoint results against recorded training time. For each method, seed curves are interpolated only over their common observed time interval, with no extrapolation. Bands are one seed standard error. These curves and Table 7 describe measured cost; no equal-time retraining comparison was performed.

improves reliability. The environments, four seeds, 32,768-episode budget, and output rule match the Gaussian comparison.

These comparisons concern the specified homogeneous block family with random correct actions and its practical implementation. No empirical claim is made about all low-rank MDPs, both theoretical settings, or the asymptotic exponents in the PAC bounds.