PROVABLY SAFE SIM-TO-REAL TRANSFER

Anonymous authors
Paper under double-blind review

ABSTRACT

We address safe sim-to-real transfer, in which an agent leverages an imperfect simulator and limited real-world interaction while ensuring safety throughout data collection in the real system. This problem arises in applications such as robotics and healthcare: simulators provide cheap data, but sim-to-real mismatch makes direct transfer unreliable, and collecting real-world data to correct this mismatch must itself be safe. Moreover, deployment objectives may vary across tasks, making it costly to collect new data for each reward function. We therefore formulate safe sim-to-real transfer as a reward-free safe reinforcement learning (RL) problem, in which data are collected once and reused to plan for arbitrary reward functions. We develop a computationally efficient algorithm that identifies where the simulator and real dynamics differ, uses certified simulator transitions where they are reliable, and estimates mismatched transitions from safely collected data. With high probability, every policy deployed during learning is feasible, and the collected data support the computation of a feasible and near-optimal policy for any reward function. When the simulator is uninformative, our algorithm recovers online reward-free safe RL while improving the best-known sample complexity by a factor of $\widetilde{\Theta}(H/\xi^2)$, where ξ is the safety margin of a baseline policy. When the simulator is accurate on most transitions, this improvement grows to $\widetilde{\Theta}(H^2|\mathcal{S}||\mathcal{A}|/(\xi^2|\mathcal{B}|))$, where $|\mathcal{B}|$ denotes the size of the sim-to-real mismatch region.

1 INTRODUCTION

Over the last decade, reinforcement learning (RL) has achieved remarkable success in domains ranging from games and robotics to the natural sciences (Mnih et al., 2013; Silver et al., 2016; Ouyang et al., 2022; Lee et al., 2020; Degrave et al., 2022). Despite this progress, deploying RL in real world remains challenging due to two central obstacles: sample complexity and safety. Learning directly on the target system can require many interactions, each of which may be costly and subject to safety requirements, such as collision avoidance in robotics (Haddadin et al., 2009) or compliance with operational constraints in healthcare (Kyrarini et al., 2021).

A common way to reduce real-world interaction is to use a simulator, which can provide cheap samples from an approximate model of the target system. In robotics and control, simulators are often constructed from physical models, system knowledge, or prior calibration. However, the real world is difficult to model perfectly, leading to sim-to-real mismatch (Tan et al., 2018; Peng et al., 2018). Consequently, policies optimized in simulation may be suboptimal in the real world. In safety-critical applications, this mismatch is even more problematic since a policy that appears safe in simulation may violate safety constraints on the real system. We model such safety requirements using constrained Markov decision processes (CMDPs), where the goal is to maximize reward while satisfying constraints. This raises the question:

how can we use an imperfect simulator to learn a near-optimal feasible policy in the real world?

Existing approaches address this question in different ways. A first line of work enlarges the set of environments considered during simulation training. For example, As et al. (2026) study domain randomization for simulated CMDPs and optimize policies under worst-case constraints, while Zhang et al. (2024) study distributionally robust constrained RL. These methods can be viewed as one-shot transfer approaches where the agent learns a policy in simulation and then deploys it on the real system. However, they can be conservative, since the learned policy optimizes worst-case

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.

or distributional performance rather than performance on the specific real system (Ye et al., 2023). To reduce this conservatism, one can collect real-world data to fine tune the policy learned in simulation. This data-collection process must itself ensure constraint satisfaction, a requirement commonly referred to as safe exploration (Koller et al., 2018).

This motivates another line of work, known as meta RL and its safe variants, which trains agents over a distribution of simulated CMDPs. When deployed on a real system assumed to be drawn from the same distribution, these agents can fine tune using relatively few real-world interactions while ensuring safe exploration (Ni & Kamgarpour, 2026; Xu & Zhu, 2026). Among these works, Ni & Kamgarpour (2026) show that learning a near-optimal policy under safe exploration requires only $\tilde{\mathcal{O}}(\varepsilon^{-2}\mathcal{C}(\mathcal{D}))$ real-world samples, with a matching lower bound, where $\mathcal{C}(\mathcal{D})$ measures the complexity of the environment distribution $\mathcal{D}$. This quantity is small when $\mathcal{D}$ concentrates on a low-dimensional family of environments, but can still suffer from the curse of dimensionality when distribution $\mathcal{D}$ spreads broadly across the environment space, such as uniform distribution.

Despite these advances, both one-shot transfer methods and safe meta RL rely on a coverage assumption: the real system must be drawn from, or at least well covered by, the prescribed family or distribution of simulated CMDPs. This assumption can fail when the simulator is biased. For example, a robot may be trained on many simulated terrains, yet encounter a real floor whose wear or dust was not anticipated by the designer. Such discrepancies may be discovered only through real-world interaction and therefore cannot be incorporated into the simulator beforehand. Recent sim-to-real RL methods avoid this coverage assumption by using the simulator to guide real-world data collection, either to correct simulator mismatch (Qu et al., 2025; Wu et al., 2026) or to learn an optimal real-world policy (Wagenmaker et al., 2024). However, these methods address unconstrained settings and may therefore collect unsafe data when applied to safety-critical systems such as autonomous vehicles, surgical robots, or power grids.

To address safety during real-world data collection, a separate line of work studies online safe RL. Prior model-based (Yu et al., 2025; Bura et al., 2022; Liu et al., 2021; Wendl et al., 2026) and model-free (Ni & Kamgarpour, 2025b) methods provide high-probability guarantees for learning a near-optimal policy under safe exploration, but they typically assume a single prespecified reward. In practice, reward functions are often iteratively engineered or vary across deployments (Jin et al., 2020; Ménard et al., 2021); for example, in autonomous driving, different rewards may encode different target locations. Collecting new real-world data for each reward is therefore highly sample inefficient. Reward-free safe RL (Miryoosefi & Jin, 2022; Huang et al., 2023) addresses this issue by exploring without a prespecified reward and subsequently planning for arbitrary reward functions without additional interaction with the real system. However, Miryoosefi & Jin (2022) do not ensure safe exploration during data collection, while Huang et al. (2023) ensure safe exploration but require solving a generally intractable constrained nonconvex optimization problem to compute exploration policies. Moreover, these methods are fully online and do not leverage simulator information.

To answer the question raised above, we address a safe sim-to-real transfer framework that leverages information from an imperfect simulator, focuses on safe real-world exploration, and supports subsequent planning for arbitrary reward functions without additional real-world interaction. We summarize our contributions below:

1. We formulate safe sim-to-real transfer as a reward-free safe RL problem, where the agent uses a simulator and limited real-world interaction under safe exploration guarantees to support accurate reward-free planning. We propose a computationally efficient algorithm that identifies reliable simulator regions, uses certified simulator transitions to reduce real-world interaction, and corrects sim-to-real mismatch with collected real-world data.

2. For our algorithm, we provide high-probability guarantees for safe real-world exploration and accurate reward-free planning; see Theorem 4.1. The sample-complexity bound characterizes the benefit of simulator access through the size of the mismatch region and the separation gap. When the sim-to-real mismatch is large, our framework recovers the fully online setting of Huang et al. (2023) and improves upon its sample-complexity bound. Additional comparisons with prior work are provided in Table 1.

3. We validate our theoretical results in a safety-critical gridworld environment. Our experiments show that the benefit of using the simulator is larger when the sim-to-real mismatch is smaller.

2 PROBLEM SETTING

We first review background on constrained Markov decision processes (CMDPs), then formally state the safe sim-to-real transfer problem and its underlying assumptions.

Notation. Let $\mathbb{N}$ and $\mathbb{R}$ denote the sets of natural and real numbers, respectively. For a set $\mathcal{X}$, let $\Delta(\mathcal{X})$ denote the set of probability measures on $\mathcal{X}$, and let $|\mathcal{X}|$ denote its cardinality. For $p, q \in \Delta(\mathcal{X})$, where $\mathcal{X}$ is finite, their total variation distance is defined as $\|p, q\|_{\mathrm{TV}} \coloneqq \frac{1}{2}\sum_{x\in\mathcal{X}} |p(x) - q(x)|$. For every positive integer m, let $[m] \coloneqq \{1, \dots, m\}$. For $z \in \mathbb{R}$, define its positive part as $[z]_+ \coloneqq \max\{z, 0\}$. For $x > 0$, define $\log_+(x) \coloneqq \max\{\log x, 0\}$. For real numbers x and y, $x \wedge y$ and $x \vee y$ denote $\min\{x, y\}$ and $\max\{x, y\}$, respectively.

2.1 CONSTRAINED MARKOV DECISION PROCESSES

We consider a CMDP defined by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, s_1, \mathcal{F}, c)$, where $\mathcal{S}$ and $\mathcal{A}$ are finite state and action spaces, and H is the horizon. The transition dynamics are given by $P \coloneqq \{P_h\}_{h=1}^H$, where $P_h(s' \mid s, a)$ denotes the probability of transitioning from state s to state s' after taking action a at timestep $h \in [H]$. Without loss of generality, the initial state is denoted by $s_1 \in \mathcal{S}$.[1] In addition, we consider a class of deterministic utility functions[2] $\mathcal{F} \coloneqq \left\{f = \{f_h\}_{h=1}^H \,\middle|\, f_h : \mathcal{S}\times\mathcal{A} \to [0,1],\, \forall h \in [H]\right\}$. Each $f \in \mathcal{F}$ measures the performance of a policy, which we introduce next. Among these utility functions, a constraint function $c \in \mathcal{F}$ encodes the safety requirement.

A Markov policy $\pi = \{\pi_h\}_{h=1}^H$ is a collection of mappings $\pi_h : \mathcal{S} \to \Delta(\mathcal{A})$, and we let Π denote the set of all such policies. Given a utility function $f \in \mathcal{F}$, the agent interacts with $\mathcal{M}$ as follows. Starting from the initial state s_1, at each timestep h, it selects an action $a_h \sim \pi_h(\cdot \mid s_h)$, receives utility $f_h(s_h, a_h)$, and transitions to the next state $s_{h+1} \sim P_h(\cdot \mid s_h, a_h)$. To measure the cumulative utility collected by π under dynamics P, we define the state value function as $V_{f,h}^{P,\pi}(s) = \mathbb{E}_{P,\pi}[\sum_{h'=h}^H f_{h'}(s_{h'}, a_{h'}) \mid s_h = s]$. We call a policy π *feasible* in $\mathcal{M}$ if its constraint value satisfies $V_{c,1}^{P,\pi}(s_1) \geq \ell$, where $\ell \in [0, H]$ is a prescribed safety threshold. The set of feasible policies is defined as $\Pi_{\mathrm{feas}}^P \coloneqq \{\pi \in \Pi \mid V_{c,1}^{P,\pi}(s_1) \geq \ell\}$. And we call π *strictly feasible* if $V_{c,1}^{P,\pi}(s_1) > \ell$.

2.2 SAFE SIM-TO-REAL TRANSFER

In safe sim-to-real transfer, the agent has full access to a simulator during learning, but the resulting policies are ultimately deployed in the real world. We consider multiple deployment tasks whose reward functions may vary over a family $\mathcal{F}$, such as reaching different target locations. In contrast, the constraint function c is fixed as all tasks operate in the same physical system. For example, safety requirements such as collision avoidance in autonomous driving or joint limits in robotic manipulation remain unchanged across tasks. The goal is to exploit cheap simulator access and limited real-world interaction to compute feasible and near-optimal policies for any reward function in $\mathcal{F}$. However, two challenges arise in this safe sim-to-real transfer setting.

First, the simulator is only an approximation of the real world, and this mismatch may cause a policy that is feasible or near-optimal in simulation to become unsafe or suboptimal in the real world. Simulator access alone is therefore insufficient, and the agent must collect real-world data to identify and correct the mismatch. Second, collecting a separate real-world dataset for every reward function in $\mathcal{F}$ would be highly sample inefficient.

These considerations motivate a *hybrid, reward-free safe RL* formulation. In particular, *hybrid* refers to a setting in which the agent leverages both simulator information and real-world interactions (Song et al., 2023; Xie et al., 2021), whereas in online RL the agent learns solely through real-world interactions. The *reward-free* perspective allows the agent to explore the environment without a prespecified reward, and later use the learned information to plan for arbitrary reward functions

[1] As explained by Fiechter (1994), if the initial state is instead drawn from a distribution $s_1 \sim \rho$, one can equivalently introduce an artificial initial state s_0 such that $P_0(\cdot \mid s_0, a) \coloneqq \rho(\cdot)$ for every action a. This augments the state space and horizon each by one, and all bounds carry over with only this constant-size modification.

[2] Although we consider deterministic utility functions for notational simplicity, our results extend to bounded stochastic utilities.

without further real-world interaction (Miryoosefi & Jin, 2022; Huang et al., 2023). Compared with learning for a fixed reward, this imposes a stronger exploration requirement because the collected data must support planning uniformly over all $r \in \mathcal{F}$. Combining these two perspectives, we formulate the learning problem as follows.

Interection setup. During learning, the agent can interact with the real world $\mathcal{M}^{\mathrm{real}} = (\mathcal{S}, \mathcal{A}, H, P^{\mathrm{real}}, s_1, \mathcal{F}, c)$ by rolling out policies. A rollout of policy π generates a trajectory $\tau^{\pi} = \{s_h, a_h, s_{h+1}\}_{h=1}^{H}$, where $a_h \sim \pi_h(\cdot \mid s_h)$ and $s_{h+1} \sim P_h^{\mathrm{real}}(\cdot \mid s_h, a_h)$. The agent also has full knowledge of the simulator $\mathcal{M}^{\mathrm{sim}} = (\mathcal{S}, \mathcal{A}, H, P^{\mathrm{sim}}, s_1, \mathcal{F}, c)$, which differs from $\mathcal{M}^{\mathrm{real}}$ only in its transition dynamics.

Since data collection occurs in the real world, where constraint violations such as collisions in autonomous navigation are unacceptable, every policy π deployed in $\mathcal{M}^{\mathrm{real}}$ must satisfy the safety constraint. We formalize this requirement as safe exploration.

Definition 2.1 (Safe exploration)**.** *An algorithm that deploys a sequence of policies $\{\pi^t\}_{t=1}^{T}$ ensures safe exploration in $\mathcal{M}^{\mathrm{real}}$ if $V_{c,1}^{P^{\mathrm{real}},\pi^t}(s_1) \geq \ell$ for every $t \in [T]$.*

Having described the simulator and real-world interactions, we now define the objective.

Objective 2.2. *Design an algorithm that ensures safe exploration in $\mathcal{M}^{\mathrm{real}}$ during data collection and, for any reward function $r \in \mathcal{F}$, computes without further real-world interaction a policy π satisfying*

$$\text{(Near-optimality)} \quad V_{r,1}^{P^{\mathrm{real}},\pi^\star}(s_1) - V_{r,1}^{P^{\mathrm{real}},\pi}(s_1) \leq \epsilon \quad \textit{and} \quad \text{(Feasibility)} \quad \pi \in \Pi_{\mathrm{feas}}^{P^{\mathrm{real}}}, \quad (1)$$

where $\pi^\star \in \arg\max_{\pi \in \Pi_{\mathrm{feas}}^{P^{\mathrm{real}}}} V_{r,1}^{P^{\mathrm{real}},\pi}(s_1)$.

We say that the algorithm achieves *ϵ-planning accuracy* if both conditions above, namely near-optimality and feasibility, hold for every $r \in \mathcal{F}$. The algorithm's efficiency is measured by the number of state-action pairs sampled from $\mathcal{M}^{\mathrm{real}}$, which we refer to as its sample complexity.

2.3 Assumptions

To achieve the above objective, we make two assumptions.[3] The first is a standard Slater-type condition in safe RL (Bura et al., 2022; Yu et al., 2025; Ni & Kamgarpour, 2025b; Wendl et al., 2026): safe exploration requires a baseline policy that is strictly feasible in $\mathcal{M}^{\mathrm{real}}$.

Assumption 2.3 (Slater's condition)**.** *There exist a known constant $\xi \in (0, H]$ and a known baseline policy $\pi^0 \in \Pi$ such that $V_{c,1}^{P^{\mathrm{real}},\pi^0}(s_1) \geq \ell + \xi$.*

The strictly positive margin ξ serves two purposes. Algorithmically, it provides a safety buffer that allows the agent to explore while maintaining feasibility. Statistically, it ensures that the empirical planning problem remains feasible once the estimation error is sufficiently small: the error cannot cause the estimated constraint value of the baseline policy π^0 below the feasibility threshold. By contrast, if π^0 were exactly feasible ($\xi = 0$), even a small estimation error could make it appear infeasible, leaving the empirical problem without a certifiably feasible solution.

Next, we adopt a separation condition commonly used in hierarchical RL (Chua et al., 2023), meta RL (Chen et al., 2022; Mutti & Tamar, 2024; Brunskill & Li, 2013), and hybrid RL (Qu et al., 2025). It requires each transition mismatch between $\mathcal{M}^{\mathrm{real}}$ and $\mathcal{M}^{\mathrm{sim}}$ to be either negligible or well separated.

Assumption 2.4. *There exist known constants $0 \leq \epsilon_s \leq \sigma_s \leq 1$ such that for every $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$,*

$$\left\| P_h^{\mathrm{real}}(\cdot \mid s, a),\, P_h^{\mathrm{sim}}(\cdot \mid s, a) \right\|_{TV} \in [0, \epsilon_s] \cup [\sigma_s, 1].$$

This condition separates transition mismatches, measured in total variation distance, into two categories: negligible mismatches of at most ϵ_s and significant mismatches of at least σ_s. We refer to $\sigma_s - \epsilon_s$ as the *separation gap*. This gap allows significant mismatches to be distinguished from negligible ones using finitely many real-world samples. Prior knowledge of ϵ_s and σ_s is necessary to exploit the simulator efficiently. Indeed, Cheung & Lyu show that even in multi-armed bandits,

[3] We discuss both assumptions in the experimental setup (see Section 5).

a special case of MDPs with $H = 1$, no hybrid method using an imperfect simulator is guaranteed to outperform a purely online method without prior information about the sim-to-real mismatch.

Under Assumption 2.4, we define the mismatch region as the set of triples whose transition mismatch between the real and simulated environments is at least σ_s.

Definition 2.5 (Mismatch region)**.** *The mismatch region $\mathcal{B}$ is defined as*

$$\mathcal{B} \coloneqq \left\{(h, s, a) : \left\| P_h^{\text{real}}(\cdot \mid s, a),\ P_h^{\text{sim}}(\cdot \mid s, a) \right\|_{TV} \geq \sigma_s \right\}.$$

When the simulator accurately models the real-world dynamics, the mismatch region $\mathcal{B}$ is small, and the simulator is well calibrated on the non-mismatch region, as captured by a small ϵ_s.

3 Algorithm Design

The RF-RL algorithm of Huang et al. (2023) achieves safe exploration and accurate reward-free planning in the fully online setting. However, directly applying RF-RL to our setting has two limitations. First, RF-RL learns solely through interactions with $\mathcal{M}^{\text{real}}$ and therefore cannot exploit simulator information to reduce real-world sample complexity. Second, building on the unconstrained reward-free RL approach of Ménard et al. (2021), RF-RL computes its exploration policy by solving a constrained nonconvex optimization problem, which is generally intractable. Algorithm 1 overcomes these limitations and achieves Objective 2.2.

Algorithm 1, summarized below, addresses these challenges through three components. First, Line 3 uses confidence bounds on empirical real-world transitions to progressively shrink the estimated mismatch region. Second, Line 4 constructs a hybrid model that uses empirical real-world transitions within this region and simulator transitions elsewhere. Together, these components avoid relearning dynamics that are already modeled accurately by the simulator. Third, unlike RF-RL, which maximizes a nonlinear truncated certificate over a pessimistic feasible set and therefore requires solving a nonconvex problem, Line 8 maximizes a *linear* uncertainty certificate over empirically feasible policies through a standard CMDP. This produces a candidate policy that targets uncertainty relevant to reward-free planning. If the corresponding certificate Δ^t indicates that the hybrid model is sufficiently accurate, the algorithm terminates at Line 10. Otherwise, Line 13 deploys an adaptive mixture of the candidate and baseline policies to ensure real-world feasibility and reduce the remaining uncertainty.

Algorithm 1 Safe sim-to-real RL

Require: Baseline policy π^0 with margin ξ, separation parameters ϵ_s and σ_s, stopping tolerance $0 < \tau \leq \xi/4$, confidence $\delta \in (0, 1)$, simulator $\mathcal{M}^{\text{sim}}$, real-world access.

1: Initialize $n_h^0(s, a) \leftarrow 0$ for all $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$, and initialize $\hat{\mathcal{B}}^0 \leftarrow [H] \times \mathcal{S} \times \mathcal{A}$
2: **for** $t = 1, 2, \ldots$ **do**
3: Construct $\hat{\mathcal{B}}^t$ as in Equations (3) and (4) // Estimating the mismatch region
4: Construct $\hat{P}^t$ using Equations (2) and (5) // Constructing the hybrid model
5: **if** $V_{c,1}^{\hat{P}^t, \pi^0}(s_1) < \ell + \xi/2$ **then**
6: $\pi^t \leftarrow \pi^0$
7: **else**
8: $\bar{\pi}^t \in \arg\max_{\pi \in \Pi_{\text{feas}}^{\hat{P}^t}} V_{b,1}^{\hat{P}^t, \pi}(s_1)$.
9: $\Delta^t \leftarrow \min\{H, V_{b,1}^{\hat{P}^t, \bar{\pi}^t}(s_1)\}$
10: **if** $\Delta^t \leq \tau$ **then**
11: **break** the **for** loop
12: **end if**
13: $\pi^t \leftarrow \alpha^t \bar{\pi}^t + (1 - \alpha^t)\pi^0$, where α^t is defined in Equation (9)
14: **end if** } Computing safe exploration policy
15: Roll out π^t in $\mathcal{M}^{\text{real}}$ and observe $\tau^t \coloneqq \{(s_h^t, a_h^t, s_{h+1}^t)\}_{h=1}^H$
16: **end for**
17: **return** $\hat{P}^t$ and $\pi^{\text{out}} \in \arg\max_{\pi \in \Pi} \{V_{r,1}^{\hat{P}^t, \pi}(s_1) : V_{c,1}^{\hat{P}^t, \pi}(s_1) \geq \ell + \tau\}$ for any $r \in \mathcal{F}$

Estimating the mismatch region. We start by initializing the estimated mismatch region as the entire time-state-action space, $\hat{\mathcal{B}}^0 = [H] \times \mathcal{S} \times \mathcal{A}$, as in Line 1 of Algorithm 1. In Line 3, the algorithm iteratively shrinks this region by removing triples estimated to be non-mismatch as follows. At the beginning of each iteration t, given the trajectories $\{\tau^i\}_{i=1}^{t-1}$ collected from $\mathcal{M}^{\text{real}}$ during previous iterations, we define the visitation counts for each $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$ as

$$n_h^t(s,a) \coloneqq \sum_{i=1}^{t-1} \mathbf{1}_{\{(s_h^i, a_h^i) = (s,a)\}}, \quad n_h^t(s,a,s') \coloneqq \sum_{i=1}^{t-1} \mathbf{1}_{\{(s_h^i, a_h^i, s_{h+1}^i) = (s,a,s')\}}.$$

The empirical transition model estimated from real-world data is then defined as

$$\hat{P}_h^{t,\text{real}}(s' \mid s,a) \coloneqq \mathbf{1}_{\{n_h^t(s,a)>0\}} \frac{n_h^t(s,a,s')}{n_h^t(s,a)} + \mathbf{1}_{\{n_h^t(s,a)=0\}} \frac{1}{|\mathcal{S}|}. \tag{2}$$

As shown in Lemma C.1, its statistical uncertainty is quantified by the confidence bound

$$\rho_h^t(s,a) = \min\left\{1, \sqrt{\beta(n_h^t(s,a),\delta)/(2n_h^t(s,a) \vee 1)}\right\},$$

where $\beta(n,\delta) \coloneqq \log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}| \log(8e(n+1))$.

A triple (h,s,a) is removed from the estimated mismatch region $\hat{\mathcal{B}}^{t-1}$ when its empirical real-world transition model $\hat{P}_h^{t,\text{real}}(\cdot \mid s,a)$ is sufficiently close to the simulator transition model $P_h^{\text{sim}}(\cdot \mid s,a)$, after accounting for its confidence bound $\rho_h^t(s,a)$. Specifically, define

$$\mathcal{G}^t \coloneqq \left\{(h,s,a) \in \hat{\mathcal{B}}^{t-1} : \left\|\hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a)\right\|_{\text{TV}} + \rho_h^t(s,a) \le \frac{\epsilon_s + \sigma_s}{2}\right\}. \tag{3}$$

The threshold $(\epsilon_s + \sigma_s)/2$ separates triples in the mismatch region from the remaining triples under Assumption 2.4, allowing them to be distinguished once the confidence bound is sufficiently small. We then update the estimated mismatch region as

$$\hat{\mathcal{B}}^t \coloneqq \hat{\mathcal{B}}^{t-1} \setminus \mathcal{G}^t. \tag{4}$$

By construction, the estimated regions are nonincreasing, namely, $\hat{\mathcal{B}}^t \subseteq \hat{\mathcal{B}}^{t-1}$ for every t. Lemma C.3 further shows that, with high probability, $\mathcal{B} \subseteq \hat{\mathcal{B}}^t$ at every iteration, so no mismatch triple is removed.

Constructing the hybrid model. Given the estimated mismatch region $\hat{\mathcal{B}}^t$, we construct the hybrid model as follows:

$$\hat{P}_h^t(\cdot \mid s,a) \coloneqq \mathbf{1}_{\{(h,s,a) \in \hat{\mathcal{B}}^t\}} \hat{P}_h^{t,\text{real}}(\cdot \mid s,a) + \mathbf{1}_{\{(h,s,a) \notin \hat{\mathcal{B}}^t\}} P_h^{\text{sim}}(\cdot \mid s,a). \tag{5}$$

Thus, the algorithm uses empirical real-world transition estimates within the estimated mismatch region $\hat{\mathcal{B}}^t$ and reuses the simulator transition dynamics elsewhere.

Computing a safe exploration policy. Given the hybrid model $\hat{P}^t$, we seek a safe policy that visits the parts of the estimated mismatch region where real-world transition uncertainty affects reward-free planning. For any policy π and utility function $f \in \mathcal{F}$, this uncertainty is measured by the value-estimation error $e_{f,1}^{t,\pi}(s_1) \coloneqq |V_{f,1}^{P^{\text{real}},\pi}(s_1) - V_{f,1}^{\hat{P}^t,\pi}(s_1)|$. The ideal exploration objective is

$$\max_{f \in \mathcal{F},\, \pi \in \Pi_{\text{feas}}^{P^{\text{real}}}} e_{f,1}^{t,\pi}(s_1). \tag{6}$$

Optimizing this problem would identify a safe policy that collects real-world data in the regions most relevant to reducing the value-estimation error and refining the hybrid model. However, both the objective $e_{f,1}^{t,\pi}(s_1)$ and the feasible set $\Pi_{\text{feas}}^{P^{\text{real}}}$ depend on the unknown real-world model, so Problem (6) cannot be solved directly.

We address this challenge in three steps. First, we construct a computable certificate that upper-bounds $e_{f,1}^{t,\pi}(s_1)$. Second, we maximize such certificate $e_{f,1}^{t,\pi}(s_1)$ over $\Pi_{\text{feas}}^{\hat{P}^t}$, which serves as a surrogate for $\Pi_{\text{feas}}^{P^{\text{real}}}$. The optimizer defines a candidate exploration policy, while the optimal value provides a stopping criterion. Third, if further exploration is needed, we mix the candidate policy with the strictly feasible baseline policy π^0 to guarantee real-world safety.

Step 1: Upper bound on the estimation error $e_{f,1}^{t,\pi}(s_1)$.

Lemma 3.1. *Under Assumption 2.4, with probability at least $1-\delta$, simultaneously for every iteration t, policy $\pi \in \Pi$, and utility function $f \in \mathcal{F}$,*

$$e_{f,1}^{t,\pi}(s_1) \le \min\left\{H,\ V_{b,1}^{\hat{P}^t,\pi}(s_1)\right\}.$$

Here, $V_{b,h}^{\hat{P}^t,\pi}$ denotes the value function associated with bonus function $b^t = \{b_h^t\}_{h=1}^H$, where $b_h^t : \mathcal{S} \times \mathcal{A} \to [0,H]$ is defined by $b_h^t(s,a) \coloneqq H\rho_h^t(s,a)\mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}} + H\epsilon_s\mathbf{1}_{\{(h,s,a)\notin\hat{\mathcal{B}}^t\}}$.

Within $\hat{\mathcal{B}}^t$, the bonus b^t scales with the uncertainty in $\hat{P}_h^{t,\mathrm{real}}$. Outside $\hat{\mathcal{B}}^t$, it scales with ϵ_s, because the sim-to-real mismatch has been certified to be at most ϵ_s. This construction follows from the simulation lemma, which bounds the value-estimation error $e_{f,1}^{t,\pi}(s_1)$ by the expected cumulative one-step bonus b_h^t along trajectories generated by $\hat{P}^t$ and π. This cumulative bonus is exactly $V_{b,1}^{\hat{P}^t,\pi}(s_1)$. The proof is provided in Appendix C.3.

Moreover, $V_{b,1}^{\hat{P}^t,\pi}(s_1)$ is linear in the state-action occupancy measure, making it a tractable surrogate for Problem (6). This contrasts with the clipped or truncated uncertainty values used in prior work (Ménard et al., 2021; Huang et al., 2023).[4]

Step 2: Computing a candidate policy. Line 5 of Algorithm 1 first checks whether $\hat{P}^t$ provides a sufficient safety margin for π^0, ensuring that $\Pi_{\mathrm{feas}}^{\hat{P}^t}$ is nonempty. If this check fails, Line 6 deploys π^0. Otherwise, using the certificate from Lemma 3.1, Line 8 solves the following surrogate for Problem (6):

$$\bar{\pi}^t \in \arg\max_{\pi\in\Pi_{\mathrm{feas}}^{\hat{P}^t}} V_{b,1}^{\hat{P}^t,\pi}(s_1). \tag{7}$$

This is a standard CMDP under the known hybrid model $\hat{P}^t$, with reward b^t and constraint c. It can be solved using existing CMDP solvers, such as constrained policy optimization (Achiam et al., 2017), or exactly through linear programming in the occupancy-measure formulation (Altman, 2021).

The resulting certificate of Problem (7) is

$$\Delta^t \coloneqq \min\left\{H,\ V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)\right\}. \tag{8}$$

Because $\bar{\pi}^t$ maximizes the bonus over $\Pi_{\mathrm{feas}}^{\hat{P}^t}$, Δ^t upper-bounds the uncertainty certificate of every empirically feasible policy. Thus, when $\Delta^t \le \tau$, the hybrid model is sufficiently accurate for reward-free planning under the tightened constraint in Line 17, as formalized by Lemma C.6. The algorithm then terminates.

Step 3: Ensuring safe exploration. When $\Delta^t > \tau$, further exploration is required. Although $\bar{\pi}^t$ is feasible under $\hat{P}^t$, Lemma 3.1 guarantees only that

$$V_{c,1}^{P^{\mathrm{real}},\bar{\pi}^t}(s_1) \ge V_{c,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) - \Delta^t \ge \ell - \Delta^t.$$

Thus, $\bar{\pi}^t$ may violate the safety constraint in $\mathcal{M}^{\mathrm{real}}$ by up to Δ^t. To compensate for this possible violation and ensure safe exploration, Line 13 executes a mixture of $\bar{\pi}^t$ and the strictly feasible baseline policy π^0, defined by

$$\pi^t \coloneqq \alpha^t\bar{\pi}^t + (1-\alpha^t)\pi^0,\ \alpha^t \coloneqq \xi/(\xi + \left(\ell + \Delta^t - V_{c,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)\right)_+). \tag{9}$$

The mixing weight α^t is chosen so that the strict feasibility margin of π^0 offsets the possible real-world constraint violation of $\bar{\pi}^t$. To implement this mixture, at the beginning of each iteration, the algorithm selects $\bar{\pi}^t$ with probability α^t and π^0 otherwise, and then follows the selected policy throughout the iteration.

[4] Huang et al. (2023) design their error certificate $U(\pi)$, based on a truncated bonus value, which is concave function under policy mixing. They maximize this certificate subject to the empirical safety constraint $V_c^{\hat{P},\pi} \ge \ell + U(\pi)$. This defines a sublevel set of a concave function and leads to a nonconvex constrained optimization problem.

4 THEORETICAL GUARANTEES

In this section, we formalize the theoretical guarantees of our algorithm, including the real-world sample somplexity required for safe exploration and accurate planning. The main result is stated below.

Theorem 4.1. *Let Assumptions 2.3 and 2.4 hold with $\epsilon_s \leq \xi/4H^3$. For $\delta \in (0,1)$ and $\epsilon \in \big(0, \min\{1,\ H - 4H^4\epsilon_s/\xi\}\big]$, run Algorithm 1 with $\tau \coloneqq \xi\epsilon/4H + H^3\epsilon_s$. Then, with probability at least $1-\delta$, Algorithm 1 terminates after collecting at most*

$$\widetilde{\mathcal{O}}\Bigg(\min\Bigg\{\frac{H^7|\mathcal{S}|^2|\mathcal{A}|}{\xi^2\epsilon^2}, \underbrace{\frac{H^6|\mathcal{S}||\mathcal{B}|}{\xi^2\epsilon^2}}_{(i)} + \underbrace{\frac{H^5|\mathcal{S}|\big(H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|\big)}{\xi\epsilon}\left(1 + \frac{1}{(\sigma_s - \epsilon_s)H^2}\right)}_{(ii)}\Bigg\}\Bigg).$$

samples from $\mathcal{M}^{\mathrm{real}}$ and guarantees the following:

1. ***Safe exploration.*** *Every executed policy is feasible in $\mathcal{M}^{\mathrm{real}}$.*
2. ***Planning accuracy.*** *For any $r \in \mathcal{F}$, the output policy π^{out} is feasible and $(\epsilon + 4\epsilon_s H^4/\xi)$-optimal for the true CMDP defined by $\max_{\pi \in \Pi^{P^{\mathrm{real}}}_{\mathrm{feas}}} V_{r,1}^{P^{\mathrm{real}},\pi}(s_1)$.*

The full proof of Theorem 4.1 is provided in Appendix C.4. Before presenting a proof sketch, we offer a few remarks on the sample-complexity bound and compare it with prior work.

Theorem 4.1 gives two complementary sample-complexity bounds for Algorithm 1. The first is a fully online bound that does not rely on simulator accuracy. The second captures simulator reuse: Term (i) accounts for learning real-world dynamics on $\mathcal{B}$, while Term (ii) accounts for identifying where simulator transitions can be used, with a cost controlled by the separation gap $\sigma_s - \epsilon_s$. When the mismatch region is small, $|\mathcal{B}| \ll H|\mathcal{S}||\mathcal{A}|$, the separation gap is large, $\sigma_s - \epsilon_s \geq H^{-2}$, our sample complecity bound is dominated by Term (i) and improves on the fully online bound by a factor of order $H|\mathcal{S}||\mathcal{A}|/|\mathcal{B}|$. Since Algorithm 1 reuses simulator transitions outside mismatch region $\mathcal{B}$, where the sim-to-real mismatch is bounded by ϵ_s, it incurs an additional planning error of at most $4\epsilon_s H^4/\xi$.

RF-RL (Huang et al., 2023) provides a fully online sample-complexity bound of $\widetilde{\mathcal{O}}(|\mathcal{A}||\mathcal{S}|^2H^8/(\epsilon^2\xi^4))$ for safe exploration and ϵ-accurate planning.[5] When $\epsilon_s = \sigma_s = 0$ and $\mathcal{B} = [H] \times \mathcal{S} \times \mathcal{A}$, our algorithm recovers this fully online setting as it does not reuse simulator. Even then, our bound improves on RF-RL by a factor of H/ξ^2. Thus, part of the sample-complexity benefit comes from the exploration policy design rather than simulator use.

Proof sketch. The proof of Theorem 4.1 proceeds in three steps. First, Lemma C.3 guarantees that no true mismatch triple in $\mathcal{B}$ is removed, while Lemma C.4 shows that each non-mismatch triple is removed after $\widetilde{\mathcal{O}}(|\mathcal{S}|/(\sigma_s - \epsilon_s)^2)$ visits. Thus, non-mismatch triples incur only a bounded learning cost before their transition estimates are replaced by the simulator, as formalized in Lemma C.5.

Second, Lemma 3.1 shows that the certificate Δ^t, defined in Equation (8), bounds reward and constraint value errors between the hybrid model $\hat{P}^t$ and real model P^{real} of the candidate policy $\bar{\pi}^t$. Together with the Slater margin of the baseline policy π^0, this guarantees safety of the deployed mixture policy π^t in $\mathcal{M}^{\mathrm{real}}$. However, mixing can slow exploration since the candidate policy is executed only with probability α^t, which may be small.

Third, we show that the deployed mixture policy π^t continues to collect sufficient information despite mixing with the baseline policy π^0. The key observation is $\alpha^t \geq \xi/(\xi + V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1))$. Thus, a small probability α^t of executing the candidate policy $\bar{\pi}^t$ can occur only when its uncertainty certificate $V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)$ is large relative to the safety margin ξ. This relationship ensures sufficient exploration progress even when the candidate policy is executed infrequently. We formalize this argument by relating the certificate to real-world visitation counts and separating the residual simulator error, controlled by ϵ_s, which is absorbed into the stopping tolerance τ. The resulting bound also covers iterations in which only the baseline policy π^0 is deployed. We then bound the total number of iterations by counting how often each triple can contribute to the uncertainty certificate. Mismatch triples may

[5] Huang et al. (2023) state their bound in trajectories with cumulative rewards and costs bounded by 1. We rescale to cumulative utilities of order H and multiply by H to count the number of sampled state-action pairs.

contribute throughout learning, producing a learning cost proportional to $|\mathcal{B}|$. Non-mismatch triples contribute only until they are identified and removed, producing an identification cost proportional to $H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|$ and seperation gap $\sigma_s - \epsilon_s$. Counting all triples without using deletion gives the fully online bound. Taking the smaller of these bounds yields the sample complexity in Theorem 4.1. At termination, $\Delta^t \leq \tau$, and Lemma C.6 guarantees real-world safety and reward-free planning accuracy.

5 Experiments

We evaluate our algorithm on a 5×5 gridworld with horizon $H = 12$, adapted from Sutton et al. (1998). The agent aims to reach the goal state as quickly as possible while limiting the expected number of visits to unsafe cells. We choose this setting because its structure allows the assumptions to be directly verified. We compare against two baselines: (i) *reward-free safe RL*, which treats all state-action pairs as mismatch triples and does not exploit simulator information, thereby isolating the benefit of simulator access; and (ii) *unconstrained sim-to-real RL*, which exploits simulator information without enforcing safety, allowing us to quantify the cost of safe exploration. Both baselines are modifications of Algorithm 1 and are described in Appendix B.

Safety versus sample complexity. Table 2 and Figure 1b summarize the performance of the three methods. All methods achieve accurate planning, but differ substantially in safety and sample complexity. The unconstrained sim-to-real baseline is the most sample-efficient, but violates the safety constraint in 50.2% of iterations. At the other extreme, the reward-free safe RL baseline guarantees safe exploration but requires 6.4 times as many samples from $\mathcal{M}^{\text{real}}$ as the unconstrained sim-to-real baseline. Our method maintains safe exploration while using only 1.9 times more samples than the unconstrained baseline. Our algorithm emprically shows the benefit of simulator information to reduce real-world interaction while preserving safety and planning accuracy.

Assumption verification. For Assumption 2.3, the baseline policy π^0 can be obtained in practice through behavior cloning from expert demonstrations (Torabi et al., 2018) or from a known safe controller. Its margin ξ can then be estimated by rolling out π^0 in $\mathcal{M}^{\text{real}}$ and subtracting the safety threshold ℓ from a lower confidence bound on its constraint value. Here, we manually construct π^0 using the gridworld structure. For Assumption 2.4, ϵ_s and σ_s can be specified using domain knowledge or estimated from independent data (Brunskill & Li, 2013). Here, direct computation gives $\epsilon_s = 0$ and $\sigma_s = 0.68$. When direct computation is unavailable, we provide Algorithm 4 to estimate σ_s for a specified tolerance ϵ_s on small modeling errors. Proposition B.1 shows that the estimate is conservative, i.e., $\hat{\sigma}_s \leq \sigma_s$, when all mismatch triples are covered; otherwise, it provides a lower bound on the separation over the detected mismatch region. Appendix B.6.2 analyzes the sample complexity of this procedure and shows that smaller values of σ_s require more samples.

Appendix B.7 studies the effect of misspecifying $\sigma_s + \epsilon_s$, which determines the threshold in Line 3 of Algorithm 1. Underestimation preserves the planning accuracy and safe exploration guarantees of Theorem 4.1 but requires more samples. Overestimation, however, may incorrectly remove true mismatch triples from $\hat{\mathcal{B}}^t$, reducing planning accuracy, as observed in our experiments. Such removals can also invalidate the safe exploration guarantee, although no unsafe exploration was observed experimentally. A conservative estimate of $\sigma_s + \epsilon_s$ is therefore preferable.

Implementation details and additional ablation results on $|\mathcal{B}|$ and σ_s are provided in Appendix B.

6 Conclusion and future work

We proposed a computationally efficient algorithm for safe sim-to-real transfer that uses simulator information to guide safe real-world data collection and correct mismatched transitions. We established high-probability guarantees for safe exploration and reward-free planning. The sample-complexity bound quantifies the benefit of simulator access, particularly when the mismatch region is small and the separation gap is large. Gridworld experiments demonstrate these benefits.

Limitations and future work. This work is limited to tabular CMDPs and gridworld experiments. Future directions include extending the approach to continuous spaces via function approximation, evaluating it on more realistic benchmarks, and establishing lower bounds for safe sim-to-real transfer.

AI USE STATEMENT

In this work, we used generative AI tools to polish writing, provide feedback on experimental design, assist with checking proofs, create or edit code. We did not use generative AI tools to develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proofs, write proofs, propose or refine hypotheses, support qualitative or thematic data analysis, or interpret results. The categories of translating text, cleaning or reformatting datasets, and generating synthetic datasets are not applicable to this work. All AI-assisted content was reviewed by the authors. The proofs were checked by the authors, and AI-assisted code was verified and tested for correctness. We take full responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide all implementation details in Appendix B to facilitate reproducibility. The algorithm is simple to implement, and we will share the code during the review process and release it publicly.

REFERENCES

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

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

Yarden As, Chengrui Ray Qu, Benjamin Unger, Dongho Kang, Max van der Hart, Laixi Shi, Stelian Coros, Adam Wierman, and Andreas Krause. Spidr: A simple approach for zero-shot safety in sim-to-real transfer. *Advances in Neural Information Processing Systems*, 38:90882–90924, 2026.

Emma Brunskill and Lihong Li. Sample complexity of multi-task reinforcement learning. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, pp. 122–131, Arlington, Virginia, USA, 2013. AUAI Press.

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

Xiaoyu Chen, Jiachen Hu, Chi Jin, Lihong Li, and Liwei Wang. Understanding domain randomization for sim-to-real transfer. In *International Conference on Learning Representations*, 2022.

Wang Chi Cheung and Lixing Lyu. Leveraging (Biased) information: Multi-armed bandits with offline data. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 8286–8309. PMLR, 21–27 Jul 2024.

Kurtland Chua, Qi Lei, and Jason Lee. Provable hierarchy-based meta-reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 10918–10967. PMLR, 2023.

Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. *Advances in Neural Information Processing Systems*, 30, 2017.

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022.

Claude-Nicolas Fiechter. Efficient reinforcement learning. In *Proceedings of the seventh annual conference on Computational learning theory*, pp. 88–97, 1994.

Sami Haddadin, Alin Albu-Schäffer, and Gerd Hirzinger. Requirements for safe robots: Measurements, analysis and new insights. *The International Journal of Robotics Research*, 28(11-12): 1507–1527, 2009.

Ruiquan Huang, Jing Yang, and Yingbin Liang. Safe exploration incurs nearly no additional sample complexity for reward-free RL. In *The Eleventh International Conference on Learning Representations*, 2023.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *International conference on machine learning*, pp. 4870–4879. PMLR, 2020.

Anders Jonsson, Emilie Kaufmann, Pierre Ménard, Omar Darwiche Domingues, Edouard Leurent, and Michal Valko. Planning in Markov decision processes with gap-dependent sample complexity. *Advances in Neural Information Processing Systems*, 33:1253–1263, 2020.

Krishna C Kalagarla, Rahul Jain, and Pierluigi Nuzzo. A sample-efficient algorithm for episodic finite-horizon MDP with constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8030–8037, 2021.

Torsten Koller, Felix Berkenkamp, Matteo Turchetta, and Andreas Krause. Learning-based model predictive control for safe exploration. In *2018 IEEE conference on decision and control (CDC)*, pp. 6059–6066. IEEE, 2018.

Maria Kyrarini, Fotios Lygerakis, Akilesh Rajavenkatanarayanan, Christos Sevastopoulos, Harish Ram Nambiappan, Kodur Krishna Chaitanya, Ashwin Ramesh Babu, Joanne Mathew, and Fillia Makedon. A survey of robots in healthcare. *Technologies*, 9(1):8, 2021.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020.

Xin Liu, Bin Li, Pengyi Shi, and Lei Ying. An efficient pessimistic-optimistic algorithm for stochastic linear bandits with general constraints. *Advances in Neural Information Processing Systems*, 34: 24075–24086, 2021.

Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Emilie Kaufmann, Edouard Leurent, and Michal Valko. Fast active learning for pure exploration in reinforcement learning. In *International Conference on Machine Learning*, pp. 7599–7608. PMLR, 2021.

Sobhan Miryoosefi and Chi Jin. A simple reward-free approach to constrained reinforcement learning. In *International conference on machine learning*, pp. 15666–15698. PMLR, 2022.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Mirco Mutti and Aviv Tamar. Test-time regret minimization in meta reinforcement learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 37016–37040. PMLR, 21–27 Jul 2024.

Tingting Ni and Maryam Kamgarpour. A learning-based approach to stochastic optimal control under reach-avoid constraint. In *Proceedings of the 28th ACM International Conference on Hybrid Systems: Computation and Control*, pp. 1–8, 2025a.

Tingting Ni and Maryam Kamgarpour. A safe exploration approach to constrained markov decision processes. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 3592–3600. PMLR, 03–05 May 2025b.

Tingting Ni and Maryam Kamgarpour. Constrained meta reinforcement learning with provable test-time safety. In *Forty-third International Conference on Machine Learning*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 3803–3810. IEEE, 2018.

Chengrui Qu, Laixi Shi, Kishan Panaganti, Pengcheng You, and Adam Wierman. Hybrid transfer reinforcement learning: Provable sample efficiency from shifted-dynamics data. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1054–1062. PMLR, 03–05 May 2025.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Yuda Song, Yifei Zhou, Ayush Sekhari, Drew Bagnell, Akshay Krishnamurthy, and Wen Sun. Hybrid RL: Using both offline and online data can make RL efficient. In *The Eleventh International Conference on Learning Representations*, 2023.

Richard S Sutton, Andrew G Barto, and Andrew Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Jie Tan, Tingnan Zhang, Erwin Coumans, Atil Iscen, Yunfei Bai, Danijar Hafner, Steven Bohez, and Vincent Vanhoucke. Sim-to-real: Learning agile locomotion for quadruped robots. *arXiv preprint arXiv:1804.10332*, 2018.

Faraz Torabi, Garrett Warnell, and Peter Stone. Behavioral cloning from observation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, pp. 4950–4957. AAAI Press, 2018. ISBN 9780999241127.

Andrew Wagenmaker, Kevin Huang, Liyiming Ke, Kevin Jamieson, and Abhishek Gupta. Overcoming the sim-to-real gap: Leveraging simulation to learn to explore for real-world rl. *Advances in Neural Information Processing Systems*, 37:78715–78765, 2024.

Manuel Wendl, Yarden As, Manish Prajapat, Anton Pollak, Stelian Coros, and Andreas Krause. Safe exploration via policy priors. In *The Fourteenth International Conference on Learning Representations*, 2026.

Zheshun Wu, Renjie Zheng, Jinhang Zuo, Zenglin Xu, and Fang Kong. A unified algorithmic framework for hybrid reinforcement learning in tabular mdps with shifted transition dynamics. *arXiv preprint arXiv:2607.25207*, 2026.

Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34:27395–27407, 2021.

Siyuan Xu and Minghui Zhu. Efficient safe meta-reinforcement learning: Provable near-optimality and anytime safety. *Advances in Neural Information Processing Systems*, 38:31362–31406, 2026.

Haotian Ye, Xiaoyu Chen, Liwei Wang, and Simon Shaolei Du. On the power of pre-training for generalization in rl: provable benefits and hardness. In *International Conference on Machine Learning*, pp. 39770–39800. PMLR, 2023.

Kihyun Yu, Duksang Lee, William Overman, and Dabeen Lee. Improved regret bound for safe reinforcement learning via tighter cost pessimism and reward optimism. In *Reinforcement Learning Conference*, 2025.

Zhengfei Zhang, Kishan Panaganti, Laixi Shi, Yanan Sui, Adam Wierman, and Yisong Yue. Distributionally robust constrained reinforcement learning under strong duality. In *Reinforcement Learning Conference*, 2024.

ORGANIZATION OF THE APPENDIX

A Comparison with related work **14**

B Experimental details **15**

B.1 Experiment setup 15

B.2 Baseline algorithms 15

B.3 Safety versus sample complexity 17

B.4 Sample complexity versus the size of the mismatch region $|\mathcal{B}|$ 17

B.5 Sample complexity versus the value of the separation parameter σ_s 19

B.6 Estimating the separation parameter σ_s from real-world data 19

B.6.1 Algorithm for estimating σ_s and its guarantees 19

B.6.2 Sample complexity of estimating σ_s 22

B.7 Effect of misspecifying $\sigma_s + \epsilon_s$ 22

C Proofs **23**

C.1 Concentration events 23

C.2 Analysis of the estimated mismatch region 25

C.3 Proof of Lemma 3.1 27

C.4 Proof of Theorem 4.1 27

D Supporting lemmas **34**

A Comparison with related work

In this section, we compare our setting with the best-known prior work along four directions: (i) *sim-to-real*, whether the algorithm exploits simulator information despite mismatch with the real world or learns entirely from real-world interaction, corresponding to the fully online setting; (ii) *reward-free learning*, whether the returned model supports planning for arbitrary reward functions or only the single reward used during learning; (iii) *constraints*, whether constraints are part of the problem formulation; and (iv) *safe exploration*, whether the algorithm guarantees that constraints are satisfied during the interaction with the real world. The results are summarized in Table 1. Our method is the only one in the comparison that simultaneously handles simulator mismatch, constraints, safe real-world exploration, and reward-free planning for arbitrary rewards.To the best of our knowledge, it is the first provable algorithm with all four properties.

Table 1: Comparison with related work.

Algorithm	Sim-to-real	Reward	Constraints	Safe exploration	Sample complexity
Ménard et al. (2021)	×	any	×	×	$\widetilde{\mathcal{O}}\Big(\frac{H^4\lvert\mathcal{S}\rvert^2\lvert\mathcal{A}\rvert}{\epsilon^2}\Big)$
Miryoosefi & Jin (2022)	×	any	✓†	×	$\widetilde{\mathcal{O}}\Big(\frac{\min\{d,\lvert\mathcal{S}\rvert\}H^5\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert}{\epsilon^2}+\frac{H^4\lvert\mathcal{S}\rvert^2\lvert\mathcal{A}\rvert}{\epsilon}\Big)$
Huang et al. (2023)	×	any	✓	✓	$\widetilde{\mathcal{O}}\Big(\frac{\lvert\mathcal{A}\rvert\lvert\mathcal{S}\rvert^2H^8}{\epsilon^2\xi^4}\Big)$
Yu et al. (2025)	×	fixed	✓	✓	$\widetilde{\mathcal{O}}\Big(\frac{\lvert\mathcal{A}\rvert\lvert\mathcal{S}\rvert^2H^6}{\epsilon^2\xi^2}\Big)$
Ni & Kamgarpour (2025a)	×	fixed	✓	✓	$\widetilde{\mathcal{O}}\Big(\frac{1}{\min\{\epsilon^6,\xi^6\}}\Big)$
Wagenmaker et al. (2024)	✓	fixed	×	×	$\widetilde{\mathcal{O}}\Big(\frac{H^{16}}{\epsilon^8}\Big)$
Qu et al. (2025)	✓	fixed	×	×	$\widetilde{\mathcal{O}}\Big(\min\Big\{\frac{H^4\lvert\mathcal{A}\rvert\lvert\mathcal{S}\rvert}{\epsilon^2},\frac{H^3\lvert\mathcal{B}\rvert}{\epsilon^2}+\frac{H^3\lvert\mathcal{S}\rvert^2\lvert\mathcal{A}\rvert}{\sigma_s^2\sigma_r^2}\Big\}\Big)$
Wu et al. (2026)	✓	fixed	×	×	$\widetilde{\mathcal{O}}\Big(\frac{H^4\lvert\mathcal{A}\rvert\lvert\mathcal{S}\rvert}{\epsilon^2}\Big)$
Our algorithm	✓	any	✓	✓	$\widetilde{\mathcal{O}}\Big(\min\Big\{\frac{H^7\lvert\mathcal{S}\rvert^2\lvert\mathcal{A}\rvert}{\xi^2\epsilon^2},\frac{H^6\lvert\mathcal{S}\rvert\lvert\mathcal{B}\rvert}{\xi^2\epsilon^2}+\frac{H^5\lvert\mathcal{S}\rvert\big(H\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert-\lvert\mathcal{B}\rvert\big)(1+(\sigma_s-\epsilon_s)H^2)}{\xi\epsilon(\sigma_s-\epsilon_s)H^2}\Big\}\Big)$

Remark.

- *Miryoosefi & Jin (2022, Theorem 4.1) require*

$$\widetilde{\mathcal{O}}\Big(\frac{\min\{d,|\mathcal{S}|\}H^4|\mathcal{S}||\mathcal{A}|}{\epsilon^2}+\frac{H^3|\mathcal{S}|^2|\mathcal{A}|}{\epsilon}\Big)$$

  iterations, where d is the dimension of the vector-valued return. To convert this result to the sample-complexity bound used in this work, we multiply the bound by H to account for the H state-action samples collected in each iteration.

  However, their analysis guarantees ϵ-optimality while allowing the returned constraint value vector to lie within distance ϵ of the constraint set, as specified in Miryoosefi & Jin (2022, Definition 2.4). For the scalar constraint considered in this work, the returned policy may therefore violate the constraint by up to ϵ: $V_{c,1}^{P,\pi}(s_1)\geq \ell-\epsilon$. In contrast, Objective 2.2 requires exact feasibility: $V_{c,1}^{P,\pi}(s_1)\geq \ell$.

- *Both Qu et al. (2025); Wu et al. (2026) consider finite-horizon MDPs with stationary transition dynamics and state their sample-complexity bounds in terms of the number of online iterations. For a consistent comparison, Table 1 expresses all bounds in terms of the total number of state-action samples collected from the real environment under non-stationary dynamics. We therefore convert the prior bounds in two steps: we multiply the number of online iterations by H to obtain the number of state-action samples, and we replace SA with HSA to account for a separate transition kernel at each timestep.*

*Moreover, Qu et al. (2025) assume σ_r-reachability: there exists a **known** constant $\sigma_r \in (0, 1]$ such that, for every $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$,*

$$\max_{\pi \in \Pi} d_h^{P^{\text{real}},\pi}(s, a) \geq \sigma_r,$$

where $d_h^{P^{\text{real}},\pi}(s, a)$ is the probability of visiting (s, a) at timestep h under policy π. As shown in Table 1, their sample complexity depends polynomially on $1/\sigma_r$. In contrast, our analysis does not require this reachability assumption and therefore has no dependence on σ_r.

*Qu et al. (2025) also impose a strict separation condition: there exists a **known** constant $\sigma_s \in (0, 1]$ such that, for every $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$,*

$$\left\| P_h^{\text{real}}(\cdot \mid s, a),\ P_h^{\text{sim}}(\cdot \mid s, a) \right\|_{TV} \in \{0\} \cup [\sigma_s, 1].$$

Thus, the simulator transition must either match the real transition exactly or differ from it by at least σ_s. Our Assumption 2.4 relaxes this condition by allowing the discrepancy to lie in

$$[0, \epsilon_s] \cup [\sigma_s, 1].$$

This relaxation allows for a small, nonzero modeling errors and therefore captures a broader class of sim-to-real mismatches.

B Experimental details

Computational resources. All experiments were conducted on a single desktop machine equipped with an AMD Ryzen 9 9950X3D CPU (16 cores) and 128 GB RAM. No GPU was used. All planning subproblems were formulated as linear programs over the occupancy-measure polytope and solved using the HiGHS solver through SciPy. In our largest instance, each linear program contained $H|\mathcal{S}||\mathcal{A}| = 1200$ decision variables.

B.1 Experiment setup

We consider a CMDP with stationary dynamics, $P_h = P$ for all $h \in [H]$, which is a special case of our setting. The agent starts in the bottom-left corner (the star in Figure 1a) and chooses among four actions: *up*, *right*, *down*, and *left*. The intended action is executed with probability 0.85; otherwise, the agent moves uniformly to a neighboring cell. Three central cells form an unsafe wall (the hatched cells in Figure 1a). We define the constraint function as $c(s, a) = \mathbf{1}_{\{s \text{ is safe}\}}$, so the constraint $V_{c,1}^{P^{\text{real}},\pi}(s_1) \geq \ell$ limits the expected number of unsafe visits to $H - \ell$.

The simulator P^{sim} follows these nominal dynamics. The real environment additionally contains three *windy* cells adjacent to the unsafe wall (the red cells in Figure 1a). At a windy cell, the wind reverses the selected action with probability $p_{\text{wind}} = 0.8$; for example, *up* is replaced by *down*. Thus,

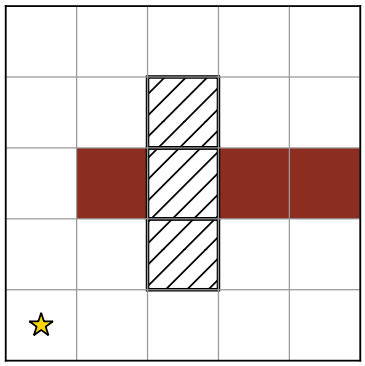
$$P^{\text{real}}(\cdot \mid s, a) = (1 - p_{\text{wind}})P^{\text{sim}}(\cdot \mid s, a) + p_{\text{wind}}P^{\text{sim}}(\cdot \mid s, \bar{a}),$$

where $\bar{a}$ denotes the action opposite to a. The real-world dynamics coincide with the simulator dynamics at all non-windy cells. As detailed in Appendix B.5, the resulting total-variation gap is $0.85p_{\text{wind}}$, giving $\sigma_s = 0.68$. At every other cell, the real and simulator dynamics coincide, so $\epsilon_s = 0$ in Assumption 2.4.

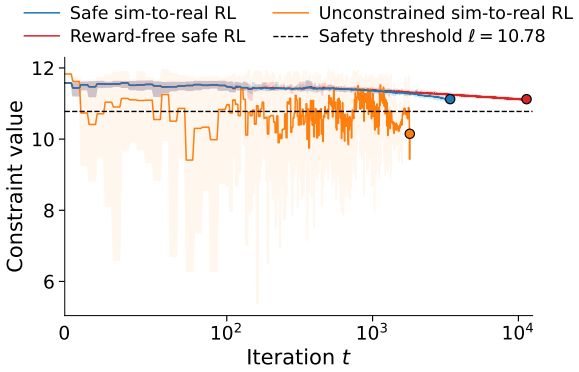
Because the windy cells lie next to the unsafe wall, the mismatch directly affects safety: a policy that expects to move away from the wall under the simulator may instead be pushed toward it in the real environment.

B.2 Baseline algorithms

In this section, we describe two baseline algorithms: a reward-free safe RL algorithm and an unconstrained sim-to-real RL algorithm. Both are modifications of Algorithm 1 and are designed to isolate the benefits of simulator access and safe exploration, respectively.



(a) Gridworld instance: red cells are the mismatch region $\mathcal{B}$, hatched cells are unsafe, and the star marks the initial state s_1.



(b) Constraint values of deployed policies for three algorithms over iterations. Markers denote termination points. Shaded regions indicate the min–max range across 10 seeds.

Reward-free safe RL setting. To adapt Algorithm 1 to the reward-free safe RL setting, we treat all state-action pairs as mismatch triples, i.e., $\mathcal{B} = [H] \times \mathcal{S} \times \mathcal{A}$. In this case, the algorithm does not exploit simulator information and reduces to an online reward-free safe RL algorithm. Correspondingly, we define the empirical model $\hat{P}^t$ and the bonus b^t as

$$\hat{P}_h^t(s' \mid s, a) \coloneqq \hat{P}_h^{t,\mathrm{real}}(s' \mid s, a) \text{ as in Equation (2)}, \quad b_h^t(s, a) \coloneqq H\rho_h^t(s, a). \tag{10}$$

We present reward-free safe RL algorithm below.

Algorithm 2 Reward-free safe RL

Require: A strictly feasible baseline policy π^0 with margin ξ, safety threshold ℓ, stopping tolerance $\tau \le \xi/4$, and access to $\mathcal{M}^{\mathrm{real}}$, confidence level $\delta \in (0, 1)$, access to $\mathcal{M}^{\mathrm{real}}$.

Initialize $n_h^0(s, a) \leftarrow 0$ for all $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$.
for $t = 1, 2, \ldots$ **do**
  Construct $\hat{P}^t$ and b^t from trajectories $\{\tau^i\}_{i=1}^{t-1}$, as defined in Equation (10).
  if $V_{c,1}^{\hat{P}^t,\pi^0}(s_1) < \ell + \xi/2$ **then**
    $\pi^t \leftarrow \pi^0$
  else // Computing safe exploration policy
    $\bar{\pi}^t \in \arg\max_{\pi \in \Pi} \left\{ V_{b,1}^{\hat{P}^t,\pi}(s_1) : V_{c,1}^{\hat{P}^t,\pi}(s_1) \ge \ell \right\}$
    $\Delta^t \leftarrow \min \left\{ H,\ V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) \right\}$
    if $\Delta^t \le \tau/2$ **then**
      break the **for** loop
    end if
    $\pi^t \leftarrow \alpha^t \bar{\pi}^t + (1 - \alpha^t)\pi^0$, where α^t is defined in Equation (9)
  end if
  Roll out π^t in $\mathcal{M}^{\mathrm{real}}$ and observe $\tau^t \coloneqq \{(s_h^t, a_h^t, s_{h+1}^t)\}_{h=1}^H$
end for
return $\hat{P}^t$ and $\pi^{\mathrm{out}} \in \arg\max_{\pi \in \Pi} \left\{ V_{r,1}^{\hat{P}^t,\pi}(s_1) : V_{c,1}^{\hat{P}^t,\pi}(s_1) \ge \ell + \tau/2 \right\}$ for any $r \in \mathcal{F}$

Unconstrained sim-to-real RL setting. To adapt Algorithm 1 to the unconstrained sim-to-real RL setting, we ignore the safety constraint and compute the exploration policy directly by maximizing the bonus value, without enforcing safe exploration. Correspondingly, we choose

$$\pi^t \in \arg\max_{\pi \in \Pi} V_{b,1}^{\hat{P}^t,\pi}(s_1)$$

as the exploration policy. We present the unconstrained sim-to-real RL algorithm below.

Table 2: Performance comparison over 10 seeds. Planning accuracy is defined in (1) and is reported as mean $\pm$ standard deviation over 20 random reward functions per seed. Unsafe iterations denote the fraction of learning iterations in which the constraint is violated.

Algorithm	Samples from $\mathcal{M}^{\text{real}}$	Suboptimality gap	Unsafe iterations
Safe sim-to-real RL	$39,168 \pm 672$	0.003 ± 0.008	0.0%
Reward-free safe RL	$135,456 \pm 545$	0.003 ± 0.007	0.0%
Unconstrained sim-to-real RL	$21,114 \pm 276$	0.003 ± 0.009	50.2%

Algorithm 3 Unconstrained sim-to-real RL

Require: Stopping tolerance τ, simulator information $\mathcal{M}^{\text{sim}}$, and access to $\mathcal{M}^{\text{real}}$, separation parameters ϵ_s and σ_s, confidence level $\delta \in (0,1)$, and (c, ℓ) if constrained planning is requested

Initialize $n_h^0(s,a) \leftarrow 0$ for every triple and $\hat{\mathcal{B}}^0 \leftarrow [H] \times \mathcal{S} \times \mathcal{A}$.
for $t = 1, 2, \ldots$ **do**
  Construct $\hat{\mathcal{B}}^t$ as in Equations (3) and (4) // Estimating the mismatch region
  Construct $\hat{P}^t$ as in Equations (2) and (5) // Constructing the hybrid model
  $\pi^t \in \arg\max_{\pi \in \Pi} V_{b,1}^{\hat{P}^t, \pi}(s_1)$ // Computing exploration policy
  $\Delta^t \leftarrow \min\left\{H,\, V_{b,1}^{\hat{P}^t, \pi^t}(s_1)\right\}$
  if $\Delta^t \le \tau/2$ **then**
    break the **for** loop
  end if
  Roll out π^t in $\mathcal{M}^{\text{real}}$ and observe $\tau^t := \{(s_h^t, a_h^t, s_{h+1}^t)\}_{h=1}^H$
end for
return $\hat{P}^t$ and $\pi^{\text{out}} \in \arg\max_{\pi \in \Pi} \left\{V_{r,1}^{\hat{P}^t, \pi}(s_1) : V_{c,1}^{\hat{P}^t, \pi}(s_1) \ge \ell + \tau/2\right\}$ for any $r \in \mathcal{F}$

B.3 Safety versus sample complexity

Table 2 and Figure 1b summarize the performance of three methods: safe sim-to-real RL (Algorithm 1), reward-free safe RL (Algorithm 2), and unconstrained sim-to-real RL (Algorithm 3). All methods achieve accurate planning, but differ substantially in safety and sample complexity. The unconstrained sim-to-real baseline is the most sample-efficient, but violates the safety constraint in 50.2% of learning iterations. At the other extreme, the reward-free safe RL baseline guarantees safety throughout learning but requires 6.4 times as many samples from $\mathcal{M}^{\text{real}}$ compared with unconstrained sim-to-real baseline. Our method strikes a favorable balance: it maintains safe exploration while using only 1.9 times more samples than the unconstrained baseline. These results show that simulator information can be exploited safely to substantially reduce real-world interaction without sacrificing planning accuracy.

In Figure 2, we visualize the state visitation distributions of all three algorithms at their respective termination iterations. Figure 2 shows that the unconstrained sim-to-real RL and our method concentrate exploration on the windy cells (mismatch regions), where the simulator is inaccurate, while the reward-free safe RL baseline spreads visits almost uniformly across the grid, re-learning dynamics that the simulator already models correctly. Compared with the unconstrained sim-to-real RL, our method also spends less time visiting the unsafe wall: 5.7% of its visits against 10.6%.

B.4 Sample complexity versus the size of the mismatch region $|\mathcal{B}|$

In this section, we study how the size of the mismatch region, $|\mathcal{B}|$, affects the sample complexity of Algorithm 1 in achieving safe exploration and accurate planning.

The size of the mismatch region is controlled by the number of windy cells in the gridworld. Since each windy cell affects all four actions, n windy cells correspond to $|\mathcal{B}| = 4n$ mismatch state-action pairs. We vary the number of windy cells so that $|\mathcal{B}| \in \{4, 8, 12, 24, 40, 64, 84, 100\}$, out of $|\mathcal{S}||\mathcal{A}| = 100$. The default configuration has three windy cells adjacent to the unsafe wall, yielding

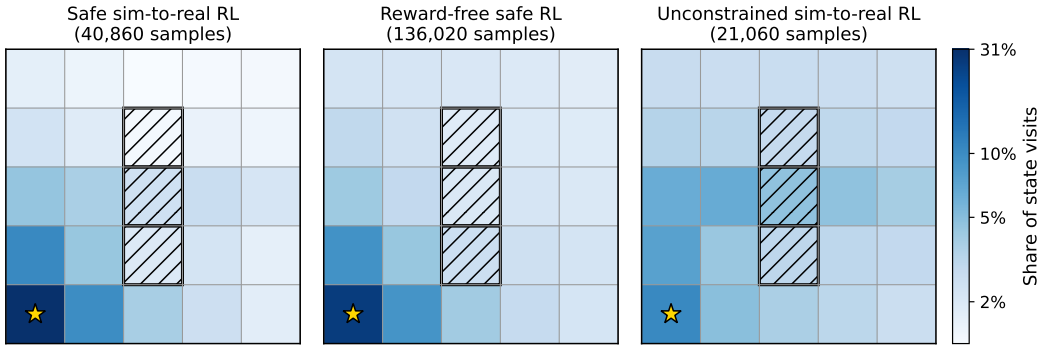


Figure 2: State visitation of the three algorithms for one seed run. Each panel is normalized by the algorithm's total number of samples, reported in the panel title, so the plots show the *share* of visits falling in each cell and compare *where* real-world interaction is spent. The color scale is shared across panels and logarithmic. We use a logarithmic scale because s_1 accounts for 10–30% of visits simply by being visited once per iteration by construction, which would otherwise dominate the rest of the grid on a linear scale.

$|\mathcal{B}| = 12$. Windy cells are selected according to a fixed priority order rather than by uniform sampling. We first rank all regular cells, excluding the three unsafe cells and the initial state, by increasing Manhattan distance to the center of the unsafe wall. Cells at the same distance are ordered randomly using the instance seed. The unsafe cells are appended next, followed by the initial state s_1. The first n cells in this ordering are then designated as windy cells. This construction has two important properties. First, mismatch is introduced closest to the unsafe wall first, so the sparse-mismatch regime is safety-critical rather than benign. Second, the construction is incremental: the windy set for a smaller $|\mathcal{B}|$ is always contained in the windy set for any larger $|\mathcal{B}|$. Thus, the experiment adds mismatch to a fixed layout instead of redrawing unrelated instances, making the growth in Figure 3 attributable to the size of $\mathcal{B}$.

As shown in Figure 3, the number of iterations required by Algorithm 1 to achieve safe exploration and accurate planning grows approximately linearly with $|\mathcal{B}|$. When the mismatch region covers the entire state-action space, Algorithm 1 no longer exploits simulator information, and the total number of iterations coincides with that of the reward-free safe RL baseline.

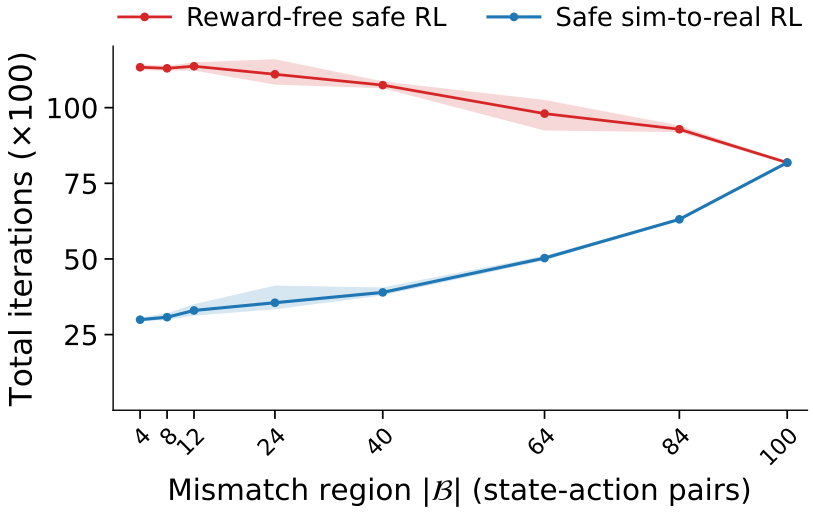


Figure 3: Total iterations versus the size of the mismatch region, $|\mathcal{B}|$, at fixed $\sigma_s = 0.68$ and $\epsilon_s = 0$, where all runs ensure safe exploration and accurate planning. Shaded regions indicate the min–max range across 10 seeds.

B.5 Sample complexity versus the value of the separation parameter σ_s

In this section, we study how different values of the separation parameter σ_s affect the sample complexity of Algorithm 1 in achieving safe exploration and accurate planning.

To vary value of σ_s, we vary the wind strength p_{wind}, defined as the probability that the wind reverses the intended move. On a windy cell, the real transition is

$$P^{\text{real}}(\cdot \mid s, a) = (1 - p_{\text{wind}})P^{\text{sim}}(\cdot \mid s, a) + p_{\text{wind}}P^{\text{sim}}(\cdot \mid s, \bar{a}),$$

where $\bar{a}$ denotes the action opposite to a. Hence, the total-variation gap from the simulator is

$$p_{\text{wind}} \left\| P^{\text{sim}}(\cdot \mid s, \bar{a}), P^{\text{sim}}(\cdot \mid s, a) \right\|_{\text{TV}},$$

which scales linearly with p_{wind}. We vary

$$p_{\text{wind}} \in \{0.15, 0.2, 0.3, 0.4, 0.55, 0.7, 0.85\},$$

which gives

$$\sigma_s \in \{0.1275,\ 0.17,\ 0.255,\ 0.34,\ 0.4675,\ 0.595,\ 0.7225\}.$$

As shown in Figure 4, Algorithm 1 requires more iterations when σ_s is smaller, because non-mismatch state-action pairs become harder to distinguish from mismatch state-action pairs. When the separation is sufficiently small, the total number of iterations approaches that of the reward-free safe RL baseline.

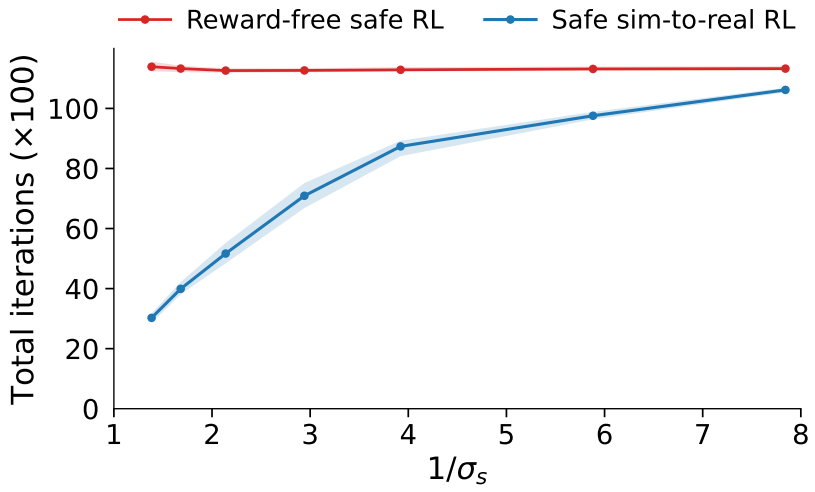


Figure 4: Total iterations versus $1/\sigma_s$ at fixed $|\mathcal{B}| = 12$ and $\epsilon_s = 0$, where all runs ensure safe exploration and accurate planning. Shaded regions indicate the min–max range across 10 seeds.

B.6 Estimating the separation parameter σ_s from real-world data

In our experiments, the simulator matches the real dynamics exactly outside the mismatch region, so $\epsilon_s = 0$ and the separation gap $\sigma_s - \epsilon_s$ reduces to σ_s. We therefore focus on σ_s, which determines how easily mismatch triples can be identified. Algorithm 1 uses σ_s only in the deletion rule $\mathcal{G}^t$ in Line 3, where it sets the threshold $\sigma_s/2$ for estimating $\hat{\mathcal{B}}^t$. Because σ_s is generally unknown, Appendix B.6.1 presents the estimation procedure in Algorithm 4 and its theoretical guarantees in Proposition B.1. Appendix B.6.2 studies the number of real-world samples required for $\hat{\sigma}_s$ to approach σ_s.

B.6.1 Algorithm for estimating σ_s and its guarantees

To estimate σ_s, we roll out the strictly feasible baseline policy π^0 in $\mathcal{M}^{\text{real}}$ for N episodes, which ensures safe exploration by Assumption 2.3. We then use the collected trajectories to estimate the separation parameter σ_s as follows.

Given the trajectories $\{\tau^i\}_{i=1}^N$ collected from $\mathcal{M}^{\text{real}}$ by the policy π^0, we define the visitation counts for each $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$ as

$$n_h(s,a) \coloneqq \sum_{i=1}^{N} \mathbf{1}_{\{(s_h^i, a_h^i)=(s,a)\}}, \quad n_h(s,a,s') \coloneqq \sum_{i=1}^{N} \mathbf{1}_{\{(s_h^i, a_h^i, s_{h+1}^i)=(s,a,s')\}}.$$

The empirical transition model estimated from real-world data is defined as

$$\hat{P}_h^{\text{real}}(s' \mid s,a) \coloneqq \mathbf{1}_{\{n_h(s,a)>0\}} \frac{n_h(s,a,s')}{n_h(s,a)} + \mathbf{1}_{\{n_h(s,a)=0\}} \frac{1}{|\mathcal{S}|}, \tag{11}$$

and its statistical uncertainty is quantified by the confidence bound

$$\rho_h(s,a) = \min\left\{1, \sqrt{\beta(n_h(s,a),\delta)/(2n_h(s,a) \vee 1)}\right\},$$

where $\beta(n,\delta) \coloneqq \log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}| \log(8e(n+1))$.

The quantity we estimate is the minimum *positive* total variation gap over the mismatch region. For every $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$, let

$$g_h(s,a) \coloneqq \left\| P_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}},$$

so that mismatch region $\mathcal{B} = \{(h,s,a) : g_h(s,a) > 0\}$ by Definition 2.5 and $\sigma_s = \min_{(h,s,a)\in\mathcal{B}} g_h(s,a)$ is the constant for which Assumption 2.4 holds. Neither $\mathcal{B}$ nor the gaps g_h are observable, since P^{real} is unknown.

Then, for each $(h, s, a) \in [H] \times \mathcal{S} \times \mathcal{A}$, we compute a lower confidence bound on $g_h(s,a)$,

$$L_h(s,a) \coloneqq \left(\left\| \hat{P}_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} - \rho_h(s,a) \right)_+. \tag{12}$$

Consider

$$\mathcal{C} \coloneqq \big\{(h,s,a) : n_h(s,a) \geq n_{\min} \text{ and } L_h(s,a) > 0\big\}, \tag{13}$$

the set of triples that are visited at least $n_{\min}$ times and whose gap is provably positive; as we show below, every such triple is certified to be a mismatch triple. Unvisited triples have $\rho_h(s,a) = 1$ and hence $L_h(s,a) = 0$, so they never enter $\mathcal{C}$. Finally, we compute the estimated separation parameter as

$$\hat{\sigma}_s \coloneqq \min\Big\{1, \min_{(h,s,a)\in\mathcal{C}} L_h(s,a)\Big\}, \qquad \text{and} \qquad \hat{\sigma}_s \coloneqq 0 \text{ if } \mathcal{C} = \emptyset.$$

The procedure of estimation of σ_s is summarized in Algorithm 4.

Algorithm 4 Safe estimation of the separation parameter σ_s

Require: Strictly feasible baseline policy π^0, simulator information $\mathcal{M}^{\text{sim}}$, number of iterations N from $\mathcal{M}^{\text{real}}$, confidence level δ, coverage requirement $n_{\min} \geq 1$

1: **for** $i = 1, \ldots, N$ **do**
2: Roll out π^0 in $\mathcal{M}^{\text{real}}$ and observe $\tau^i \coloneqq \{(s_h^i, a_h^i, s_{h+1}^i)\}_{h=1}^H$
3: **end for**
4: Compute $\hat{P}^{\text{real}}$ as in Equation (11), L_h as in Equation (12) and $\mathcal{C}$ as in Equation (13).
5: **if** $\mathcal{C} \neq \emptyset$ **then**
6: **return** $\hat{\sigma}_s \leftarrow \min\{1, \min_{(h,s,a)\in\mathcal{C}} L_h(s,a)\}$
7: **else**
8: **return** $\hat{\sigma}_s \leftarrow 0$
9: **end if**

In the following, we characterize the statistical guarantee for $\hat{\sigma}_s$ returned by Algorithm 4.

Proposition B.1. *Let Assumption 2.4 hold with $\epsilon_s = 0$, and let σ_s denote the minimum positive gap over the mismatch region $\mathcal{B}$. Then, with probability at least $1 - \delta$, Algorithm 4 guarantees the following:*

(i) $\mathcal{C} \subseteq \mathcal{B}$.

(ii) $\hat{\sigma}_s \leq \min_{(h,s,a)\in\mathcal{C}} \left\| P_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{TV} = \min_{(h,s,a)\in\mathcal{C}} g_h(s,a)$, *where the minimum over an empty set is interpreted as* $+\infty$.

(iii) If $\emptyset \neq \mathcal{B} \subseteq \mathcal{C}$, *then*

$$\sigma_s - 2 \max_{(h,s,a)\in\mathcal{B}} \rho_h(s,a) \leq \hat{\sigma}_s \leq \sigma_s.$$

Proposition B.1 shows that Algorithm 4 provides a conservative estimate of both the mismatch region and the separation parameter: with high probability, every triple in $\mathcal{C}$ is a true mismatch triple, and $\hat{\sigma}_s$ lower bounds the smallest simulator-real gap over this set. Consequently, if the collected data are sufficiently rich to identify all mismatch triples, i.e., $\mathcal{B} \subseteq \mathcal{C}$, then $\hat{\sigma}_s \leq \sigma_s$.

As shown in Appendix B.6.2, when sufficient data are sampled from $\mathcal{M}^{\text{real}}$, the estimated separation parameter $\hat{\sigma}_s$ approaches the true value σ_s with high probability. In this case, $\mathcal{C} = \mathcal{B}$, and $\hat{\sigma}_s$ is a conservative estimate of σ_s. Furthermore, as shown in Appendix B.7 and implied by Theorem 4.1, such a conservative estimate is sufficient for Algorithm 1 to achieve safe exploration and accurate planning, at the cost of increased sample complexity.

Next, we present proof of Proposition B.1.

Proof of Proposition B.1. By Lemma D.1, applied to each (h,s,a) at level $\delta/(2|\mathcal{S}||\mathcal{A}|H)$, and by a union bound over the $H|\mathcal{S}||\mathcal{A}|$ triples, exactly as in Lemma C.1, with probability at least $1-\delta$,

$$D_{\text{KL}}\Big(\hat{P}_h^{\text{real}}(\cdot \mid s,a), P_h^{\text{real}}(\cdot \mid s,a)\Big) \leq \frac{\beta(n_h(s,a),\delta)}{n_h(s,a)},\ \forall (h,s,a) \in [H]\times\mathcal{S}\times\mathcal{A}.$$

Condition on this event. For every triple with $n_h(s,a) \geq 1$, Pinsker's inequality gives

$$\left\| \hat{P}_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{real}}(\cdot \mid s,a) \right\|_{\text{TV}} \leq \rho_h(s,a).$$

By the triangle inequality,

$$\left| \left\| \hat{P}_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} - g_h(s,a) \right| \leq \rho_h(s,a).$$

Consequently,

$$L_h(s,a) = \left(\left\| \hat{P}_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} - \rho_h(s,a) \right)_+ \leq g_h(s,a).$$

Thus, $L_h(s,a) > 0$ implies $g_h(s,a) > 0$, so every triple in $\mathcal{C}$ belongs to $\mathcal{B}$. This proves (i).

For (ii), if $\mathcal{C} = \emptyset$, Algorithm 4 returns $\hat{\sigma}_s = 0$. If $\mathcal{C} \neq \emptyset$, then

$$\hat{\sigma}_s = \min\left\{1, \min_{(h,s,a)\in\mathcal{C}} L_h(s,a)\right\} \leq \min_{(h,s,a)\in\mathcal{C}} L_h(s,a) \leq \min_{(h,s,a)\in\mathcal{C}} g_h(s,a).$$

For (iii), we assume that $\mathcal{B} \neq \emptyset$. Indeed, if $\mathcal{B} = \emptyset$, then the simulator is exact, σ_s is unconstrained by Assumption 2.4, and part (i) forces $\mathcal{C} = \emptyset$ and $\hat{\sigma}_s = 0$, so the claim is vacuous. Now suppose that $\mathcal{B} \subseteq \mathcal{C}$ and $\mathcal{B} \neq \emptyset$. Since $\mathcal{B} \neq \emptyset$, this implies $\mathcal{C} \neq \emptyset$. Together with (i), it gives $\mathcal{C} = \mathcal{B}$. Hence,

$$\hat{\sigma}_s \leq \min_{(h,s,a)\in\mathcal{C}} g_h(s,a) = \min_{(h,s,a)\in\mathcal{B}} g_h(s,a) = \sigma_s.$$

Furthermore,

$$\left\| \hat{P}_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \geq g_h(s,a) - \rho_h(s,a),$$

so

$$L_h(s,a) \geq (g_h(s,a) - 2\rho_h(s,a))_+ \geq g_h(s,a) - 2\rho_h(s,a).$$

Using $\mathcal{C} = \mathcal{B}$,

$$\begin{aligned}
\min_{(h,s,a)\in\mathcal{C}} L_h(s,a) &\geq \min_{(h,s,a)\in\mathcal{B}} (g_h(s,a) - 2\rho_h(s,a)) \\
&\geq \min_{(h,s,a)\in\mathcal{B}} g_h(s,a) - 2 \max_{(h,s,a)\in\mathcal{B}} \rho_h(s,a) \\
&= \sigma_s - 2 \max_{(h,s,a)\in\mathcal{B}} \rho_h(s,a).
\end{aligned}$$

Since $\sigma_s - 2\max_{(h,s,a)\in\mathcal{B}} \rho_h(s,a) \le \sigma_s \le 1$, clipping at 1 preserves this lower bound. Therefore,

$$\sigma_s - 2 \max_{(h,s,a)\in\mathcal{B}} \rho_h(s,a) \le \hat{\sigma}_s \le \sigma_s.$$

□

B.6.2 Sample complexity of estimating σ_s

In this section, we investigate the sample complexity required to obtain a sufficiently accurate estimate $\hat{\sigma}_s$. Here, sample complexity refers to the number of samples collected by rolling out the safe baseline policy π^0 in $\mathcal{M}^{\text{real}}$.

Using the grid layout in Figure 1a, we vary the *profile* of the wind strength across the mismatch region. We consider three instances: (i) a homogeneous instance with $p_{\text{wind}} = 0.8$; (ii) a homogeneous instance with $p_{\text{wind}} = 0.35$, which has a smaller σ_s and therefore requires more samples to estimate accurately; and (iii) a heterogeneous instance in which p_{wind} takes equispaced values from 0.2 to 0.8. In the heterogeneous instance, windy cells are ordered by increasing distance from the unsafe wall, with wind strength increasing along this order. Consequently, the weakest and hardest-to-certify mismatch lies closest to the unsafe wall, making this instance more challenging. We summarize the three instances in the table below.

Instance	p_{wind}	σ_s
I (homogeneous)	0.8 everywhere	0.68
II (homogeneous)	0.35 everywhere	0.2975
III (heterogeneous)	0.2 to 0.8, equispaced	0.17

Table 3: Three gridworld instances

As shown in Figure 5, $\hat{\sigma}_s$ approaches σ_s from below as the number of samples increases, providing a conservative estimate. The estimated mismatch set $\mathcal{C}$ coincides with the true mismatch region $\mathcal{B}$ after 2.4×10^5, 1.2×10^6, and 6×10^5 samples for Instances I, II, and III, respectively. Instance I is the easiest to estimate because it has the largest σ_s, whereas Instances II and III require more samples because their separation parameters are smaller.

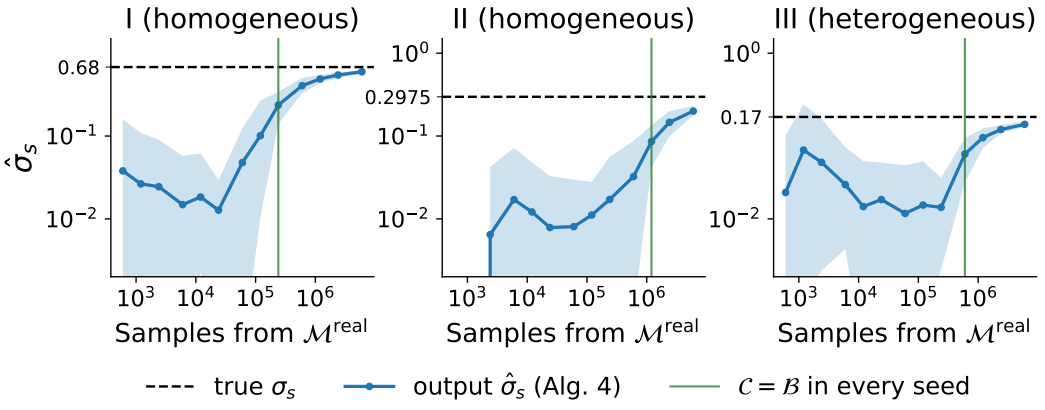


Figure 5: Sample complexity of estimating σ_s in the three instances. The curves show the mean over 20 seeds, with shaded regions indicating the min–max range.

B.7 Effect of misspecifying $\sigma_s + \epsilon_s$

In this section, we investigate the effect of replacing $\sigma_s + \epsilon_s$ with its estimate $\hat{\sigma}_s + \hat{\epsilon}_s$ in Algorithm 1. This quantity determines the threshold used in Line 3 to estimate the mismatch region $\mathcal{B}$.

We consider the same three instances as in Table 3. We vary $\hat{\sigma}_s + \hat{\epsilon}_s$ from 0 to 1, covering both underestimation and overestimation. For each instance, we test 20 values and evaluate sample

complexity, safe exploration, and planning accuracy. Planning accuracy requires the returned policy to be both near-optimal and feasible in $\mathcal{M}^{\mathrm{real}}$.

Table 4 reports the results. No deployed policy violated the safety constraint for any tested value of $\hat{\sigma}_s + \hat{\epsilon}_s$. Underestimating $\sigma_s + \epsilon_s$ preserves planning accuracy but makes the algorithm more conservative, thereby increasing the required real-world samples. In the extreme case $\hat{\sigma}_s + \hat{\epsilon}_s = 0$, the algorithm reduces to online reward-free safe RL. Overestimation has the opposite effect: it can reduce sample complexity but may cause true mismatch triples to be removed from $\hat{\mathcal{B}}^t$, making the resulting model inaccurate for subsequent constrained reward-free planning. False deletions first occur at $\hat{\sigma}_s + \hat{\epsilon}_s \approx 2.5(\sigma_s + \epsilon_s)$ for Instance II and at $\hat{\sigma}_s + \hat{\epsilon}_s \approx 3.2(\sigma_s + \epsilon_s)$ for Instance III.

Overall, conservative underestimation preserves the planning-accuracy and safe-exploration guarantees of Theorem 4.1, but at the cost of higher sample complexity. By contrast, overestimation may remove true mismatch triples from $\hat{\mathcal{B}}^t$, compromising planning accuracy as observed experimentally, and potentially safe exploration. Although no unsafe exploration was observed in our experiments, the theoretical safety guarantee need not hold after such false removals. Therefore, a conservative estimate of $\sigma_s + \epsilon_s$ is preferable.

C Proofs

C.1 Concentration events

Let $d_h^{P^{\mathrm{real}},\pi}(s,a)$ denote the probability of visiting state-action pair (s,a) at timestep h under policy π and transition kernel P^{real}. Let $\mathcal{F}_{t-1}$ denote the σ-field generated by all observations before iteration t, excluding the random draw of the mixture component used in iteration t. For every $(h,s,a) \in [H] \times \mathcal{S} \times \mathcal{A}$ and $t \in \mathbb{N}$, define the cumulative visitation probability

$$\bar{n}_h^t(s,a) := \sum_{i=1}^{t-1} d_h^{P^{\mathrm{real}},\pi^i}(s,a)$$

This quantity represents the cumulative expected number of visits to (s,a) at timestep h over the first $t-1$ episodes of Algorithm 1. When $t = 1$, $\bar{n}_h^1(s,a) = 0$. Since π^t is $\mathcal{F}_{t-1}$-measurable, $d_h^{P^{\mathrm{real}},\pi^t}(s,a)$ is the conditional probability, given $\mathcal{F}_{t-1}$, that (s,a) is visited at time step h in iteration t.

We introduce two favorable events. The event $\mathcal{E}$ ensures that the empirical transition model is close to the true transition model, while $\mathcal{E}^{\mathrm{cnt}}$ guarantees that the observed visitation counts have a uniform lower bound in terms of their cumulative conditional visitation probabilities:

$$\mathcal{E} := \left\{ \forall t \in \mathbb{N},\ \forall h \in [H],\ \forall (s,a) \in \mathcal{S} \times \mathcal{A} :\ D_{\mathrm{KL}}\Big(\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a),\ P_h^{\mathrm{real}}(\cdot \mid s,a)\Big) \le \frac{\beta(n_h^t(s,a),\delta)}{n_h^t(s,a)} \right\},$$

$$\mathcal{E}^{\mathrm{cnt}} := \left\{ \forall t \in \mathbb{N},\ \forall h \in [H],\ \forall (s,a) \in \mathcal{S} \times \mathcal{A} :\ n_h^t(s,a) \ge \tfrac{1}{2}\bar{n}_h^t(s,a) - \beta^{\mathrm{cnt}}(\delta) \right\}.$$

We next show that, for appropriate choices of the confidence functions β and β^{cnt}, the events $\mathcal{E}$ and $\mathcal{E}^{\mathrm{cnt}}$ hold simultaneously with high probability.

Lemma C.1. *By setting*

$$\beta(n,\delta) := \log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}| \log(8e(n+1)),\ \beta^{cnt}(\delta) := \log(2|\mathcal{S}||\mathcal{A}|H/\delta),$$

we have $\Pr(\mathcal{E} \cap \mathcal{E}^{cnt}) \ge 1 - \delta$.

Proof of Lemma C.1. By Lemma D.1, we have $\Pr(\mathcal{E}) \ge 1 - \delta/2$. Similarly, by Lemma D.2, $\Pr(\mathcal{E}^{\mathrm{cnt}}) \ge 1 - \delta/2$. Applying the union bound yields $\Pr(\mathcal{E} \cap \mathcal{E}^{\mathrm{cnt}}) \ge 1 - \delta$. □

Lemma C.2. *On the event* $\mathcal{E}^{\mathrm{cnt}}$*, for every* $(h,s,a) \in [H] \times \mathcal{S} \times \mathcal{A}$ *and every* $t \in \mathbb{N}$*,*

$$\rho_h^t(s,a) \le \min\left\{1, \sqrt{\frac{4\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a) \vee 1}}\right\}.$$

Table 4: Performance of Algorithm 1 with different values of $\hat{\sigma}_s + \hat{\epsilon}_s$. Samples from $\mathcal{M}^{\text{real}}$: total number of samples required from $\mathcal{M}^{\text{real}}$ before termination, reported as mean $\pm$ standard deviation over 20 seeds. Unsafe iterations: share of seeds for which there exists at least one iteration t such that the deployed policy π^t is infeasible during the execution of Algorithm 1. Pairs of $\mathcal{B}$ lost: mean number of mismatch pairs from $\mathcal{B}$ incorrectly removed from $\hat{\mathcal{B}}^t$, averaged over 20 seeds. Runs with false deletion: share of seeds in which at least one pair from $\mathcal{B}$ is lost. Output optimality: $V_r^{\pi^*} - V_r^{\pi}$, averaged over 10 random reward functions per seed, where π is the output optimal policy for reward r, and π^* is the true optimal policy for reward r. Output feasibility: share of output policies that are feasible in $\mathcal{M}^{\text{real}}$. Rows in **bold** indicate settings in which the output policy is infeasible in $\mathcal{M}^{\text{real}}$ for at least one reward and seed.

$\hat{\sigma}_s + \hat{\epsilon}_s$	Samples	Unsafe iterations (share of seeds)	Pairs of $\mathcal{B}$ lost (out of 64)	Runs with false deletion	Optimality $V^* - V^{\pi}$	Output feasible
I (homogeneous), $\sigma_s = 0.6800$ and $\epsilon_s = 0$						
0.000	115,878 ± 3,135	0.0%	0.0	0%	0.0041	100.0%
0.050	115,878 ± 3,135	0.0%	0.0	0%	0.0041	100.0%
0.100	112,266 ± 3,090	0.0%	0.0	0%	0.0043	100.0%
0.150	105,744 ± 1,884	0.0%	0.0	0%	0.0046	100.0%
0.200	104,238 ± 1,232	0.0%	0.0	0%	0.0044	100.0%
0.250	98,163 ± 770	0.0%	0.0	0%	0.0044	100.0%
0.300	91,086 ± 1,227	0.0%	0.0	0%	0.0042	100.0%
0.340	85,677 ± 1,336	0.0%	0.0	0%	0.0041	100.0%
0.400	77,085 ± 1,101	0.0%	0.0	0%	0.0046	100.0%
0.450	71,586 ± 852	0.0%	0.0	0%	0.0044	100.0%
0.500	66,771 ± 730	0.0%	0.0	0%	0.0044	100.0%
0.550	63,795 ± 536	0.0%	0.0	0%	0.0043	100.0%
0.595	62,361 ± 465	0.0%	0.0	0%	0.0043	100.0%
0.620	61,356 ± 501	0.0%	0.0	0%	0.0045	100.0%
0.650	60,636 ± 592	0.0%	0.0	0%	0.0050	100.0%
0.700	59,571 ± 463	0.0%	0.0	0%	0.0045	100.0%
0.750	58,701 ± 419	0.0%	0.0	0%	0.0044	100.0%
0.800	58,074 ± 333	0.0%	0.0	0%	0.0043	100.0%
0.850	57,411 ± 355	0.0%	0.0	0%	0.0043	100.0%
1.000	56,229 ± 304	0.0%	0.0	0%	0.0047	100.0%
II (homogeneous), $\sigma_s = 0.2975$ and $\epsilon_s = 0$						
0.000	118,506 ± 3,659	0.0%	0.0	0%	0.0059	100.0%
0.050	118,506 ± 3,659	0.0%	0.0	0%	0.0059	100.0%
0.100	112,647 ± 5,057	0.0%	0.0	0%	0.0059	100.0%
0.150	105,957 ± 1,332	0.0%	0.0	0%	0.0058	100.0%
0.200	104,793 ± 1,157	0.0%	0.0	0%	0.0061	100.0%
0.250	96,492 ± 1,454	0.0%	0.0	0%	0.0062	100.0%
0.300	88,455 ± 1,358	0.0%	0.0	0%	0.0063	100.0%
0.340	84,690 ± 1,602	0.0%	0.0	0%	0.0062	100.0%
0.400	76,836 ± 2,022	0.0%	0.0	0%	0.0060	100.0%
0.450	71,817 ± 1,605	0.0%	0.0	0%	0.0062	100.0%
0.500	68,349 ± 1,532	0.0%	0.0	0%	0.0060	100.0%
0.550	65,568 ± 1,421	0.0%	0.0	0%	0.0061	100.0%
0.595	63,153 ± 1,014	0.0%	0.0	0%	0.0062	100.0%
0.620	62,028 ± 1,085	0.0%	0.0	0%	0.0064	100.0%
0.650	60,555 ± 973	0.0%	0.0	0%	0.0066	100.0%
0.700	58,773 ± 856	0.0%	0.0	0%	0.0063	100.0%
0.750	58,338 ± 1,170	0.0%	0.9	65%	0.0145	100.0%
0.800	57,732 ± 746	0.0%	3.8	100%	0.0451	100.0%
0.850	**54,774 ± 1,428**	**0.0%**	**11.6**	**100%**	**0.0855**	**99.0%**
1.000	**42,066 ± 1,404**	**0.0%**	**31.7**	**100%**	**0.1079**	**90.0%**
III (heterogeneous), $\sigma_s = 0.1700$ and $\epsilon_s = 0$						
0.000	115,719 ± 4,377	0.0%	0.0	0%	0.0068	100.0%
0.050	115,719 ± 4,377	0.0%	0.0	0%	0.0068	100.0%
0.100	111,855 ± 4,936	0.0%	0.0	0%	0.0068	100.0%
0.150	103,686 ± 2,597	0.0%	0.0	0%	0.0068	100.0%
0.200	102,645 ± 2,264	0.0%	0.0	0%	0.0066	100.0%
0.250	94,635 ± 2,234	0.0%	0.0	0%	0.0072	100.0%
0.300	87,321 ± 1,534	0.0%	0.0	0%	0.0070	100.0%
0.340	83,319 ± 1,626	0.0%	0.0	0%	0.0069	100.0%
0.400	76,380 ± 1,854	0.0%	0.0	0%	0.0075	100.0%
0.450	71,778 ± 1,763	0.0%	0.0	0%	0.0067	100.0%
0.500	67,782 ± 1,751	0.0%	0.0	0%	0.0071	100.0%
0.550	64,563 ± 1,336	0.0%	0.2	20%	0.0074	100.0%
0.595	61,755 ± 1,168	0.0%	0.6	45%	0.0069	100.0%
0.620	60,369 ± 1,155	0.0%	1.4	75%	0.0086	100.0%
0.650	58,719 ± 1,294	0.0%	2.3	100%	0.0080	100.0%
0.700	56,523 ± 1,291	0.0%	4.4	100%	0.0097	100.0%
0.750	53,739 ± 1,303	0.0%	7.2	100%	0.0129	100.0%
0.800	51,099 ± 1,431	0.0%	9.4	100%	0.0160	100.0%
0.850	**48,651 ± 1,549**	**0.0%**	**11.9**	**100%**	**0.0238**	**99.0%**
1.000	**43,410 ± 1,333**	**0.0%**	**17.5**	**100%**	**0.0372**	**95.5%**

Proof of Lemma C.2. If $\bar{n}_h^t(s,a) \le 4\beta\big(\bar{n}_h^t(s,a),\delta\big)$, then, using $\beta\big(\bar{n}_h^t(s,a),\delta\big) \ge 1$ by (i) from Lemma D.5(i), we have

$$\frac{4\beta\big(\bar{n}_h^t(s,a),\delta\big)}{\bar{n}_h^t(s,a) \vee 1} \ge 1.$$

Thus, the right-hand side of the claimed inequality equals 1, and the result follows from $\rho_h^t(s,a) \le 1$.

Now suppose that $\bar{n}_h^t(s,a) > 4\beta\big(\bar{n}_h^t(s,a),\delta\big)$. Since $\beta\big(\bar{n}_h^t(s,a),\delta\big) \ge 1$, we have $\bar{n}_h^t(s,a) > 4$, and therefore

$$\left(\min\left\{1, \sqrt{\frac{4\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a) \vee 1}}\right\}\right)^2 = \frac{4\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a)}.$$

Moreover, since $\beta\big(\bar{n}_h^t(s,a),\delta\big) \ge \beta^{\mathrm{cnt}}(\delta)$, we have

$$\frac{\bar{n}_h^t(s,a)}{4} > \beta^{\mathrm{cnt}}(\delta).$$

On the event $\mathcal{E}^{\mathrm{cnt}}$, it follows that

$$n_h^t(s,a) \ge \frac{\bar{n}_h^t(s,a)}{2} - \beta^{\mathrm{cnt}}(\delta) > \frac{\bar{n}_h^t(s,a)}{2} - \frac{\bar{n}_h^t(s,a)}{4} = \frac{\bar{n}_h^t(s,a)}{4}.$$

Since $z \mapsto \beta(z,\delta)/z$ is nonincreasing by (ii) from Lemma D.5 and $n_h^t(s,a) \ge \bar{n}_h^t(s,a)/4$,

$$\big(\rho_h^t(s,a)\big)^2 \le \frac{\beta(n_h^t(s,a),\delta)}{2n_h^t(s,a)} \le \frac{2\beta(\bar{n}_h^t(s,a)/4,\delta)}{\bar{n}_h^t(s,a)} \le \frac{2\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a)} \le \frac{4\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a)},$$

where the penultimate inequality uses the fact that $\beta(\cdot,\delta)$ is nondecreasing. Taking square roots of the above inequality completes the proof. □

Next, we establish the guarantees of Algorithms 1, under the events $\mathcal{E}$ and $\mathcal{E}^{\mathrm{cnt}}$, which jointly hold with high probability by Lemma C.1.

C.2 Analysis of the estimated mismatch region

This section proves the theoretical guarantees for Algorithm 1 on the estimated mismatch region. Lemma C.3 shows that, with high probability, no mismatch triple is removed from $\hat{\mathcal{B}}^t$ during learning, while Lemma C.4 shows that every non-mismatch triple is removed after being sampled sufficiently often. Finally, Lemma C.5 bounds the cumulative expected visitation to each non-mismatch triple before it is removed from $\hat{\mathcal{B}}^t$, thereby controlling the total cost of identifying non-mismatch triples. We prove that the midpoint test retains all triples in $\mathcal{B}$ and eventually deletes each sufficiently sampled calibrated triple.

Lemma C.3. *On the event $\mathcal{E}$, for every $t \in \mathbb{N}$, the following hold: (i) $\hat{\mathcal{B}}^t \subseteq \hat{\mathcal{B}}^{t-1}$; and (ii) $\mathcal{B} \subseteq \hat{\mathcal{B}}^t$.*

Proof of Lemma C.3. By definition, $\hat{\mathcal{B}}^t = \hat{\mathcal{B}}^{t-1} \setminus \mathcal{G}^t$, which immediately implies $\hat{\mathcal{B}}^t \subseteq \hat{\mathcal{B}}^{t-1}$.

To establish (ii), fix $(h,s,a) \in \mathcal{B}$ and suppose, towards a contradiction, that $(h,s,a) \in \mathcal{G}^t$ for some iteration t. By the definition of $\mathcal{G}^t$, we have

$$\Big\|\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a),\, P_h^{\mathrm{sim}}(\cdot \mid s,a)\Big\|_{\mathrm{TV}} + \rho_h^t(s,a) \le \frac{\sigma_s + \epsilon_s}{2}.$$

Moreover, on the event $\mathcal{E}$, Pinsker's inequality yields

$$\begin{aligned}\Big\|\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a),\, P_h^{\mathrm{real}}(\cdot \mid s,a)\Big\|_{\mathrm{TV}} &\le \min\left\{1, \sqrt{\tfrac{1}{2} D_{\mathrm{KL}}\Big(\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a), P_h^{\mathrm{real}}(\cdot \mid s,a)\Big)}\right\} \\ &\le \rho_h^t(s,a).\end{aligned}$$

Hence, by the triangle inequality,

$$\big\|P_h^{\mathrm{real}}(\cdot \mid s,a),\, P_h^{\mathrm{sim}}(\cdot \mid s,a)\big\|_{\mathrm{TV}}$$

$$
\begin{aligned}
&\leq \left\| P_h^{\text{real}}(\cdot \mid s,a),\, \hat{P}_h^{t,\text{real}}(\cdot \mid s,a) \right\|_{\text{TV}} + \left\| \hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \\
&\leq \rho_h^t(s,a) + \left\| \hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \\
&\leq \frac{\sigma_s + \epsilon_s}{2}.
\end{aligned}
$$

However, since $(h,s,a) \in \mathcal{B}$, Assumption 2.4 implies $\left\| P_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \geq \sigma_s$, which is a contradiction. Therefore, $(h,s,a) \notin \mathcal{G}^t$. Since $\mathcal{B} \subseteq \hat{\mathcal{B}}^0$ by initialization, no element of $\mathcal{B}$ is ever removed from $\hat{\mathcal{B}}^t$, and thus $\mathcal{B} \subseteq \hat{\mathcal{B}}^t$ for all $t \in \mathbb{N}$. □

Lemma C.4. *On the event $\mathcal{E}$, for every $(h,s,a) \notin \mathcal{B}$ and every $t \in \mathbb{N}$, if $n_h^t(s,a) \geq n$, then $(h,s,a) \notin \hat{\mathcal{B}}^t$, where*

$$
n := 1 + \frac{16\left(\log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}|\log(8e)\right)}{(\sigma_s - \epsilon_s)^2} + \frac{16|\mathcal{S}|}{(\sigma_s - \epsilon_s)^2} \log\left(\max\left\{ \frac{16|\mathcal{S}|}{(\sigma_s - \epsilon_s)^2}, e \right\} \right). \tag{14}
$$

Proof of Lemma C.4. Fix $(h,s,a) \notin \mathcal{B}$ and $t \in \mathbb{N}$. Since $(h,s,a) \notin \mathcal{B}$,

$$
\left\| P_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \leq \epsilon_s.
$$

On the event $\mathcal{E}$, Pinsker's inequality and the triangle inequality for total variation distance give

$$
\begin{aligned}
&\left\| \hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \\
&\leq \left\| \hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{real}}(\cdot \mid s,a) \right\|_{\text{TV}} + \left\| P_h^{\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} \\
&\leq \rho_h^t(s,a) + \epsilon_s.
\end{aligned}
$$

By Algorithm 1, the triple belongs to $\mathcal{G}^t$ whenever

$$
\left\| \hat{P}_h^{t,\text{real}}(\cdot \mid s,a),\, P_h^{\text{sim}}(\cdot \mid s,a) \right\|_{\text{TV}} + \rho_h^t(s,a) \leq \frac{\sigma_s + \epsilon_s}{2}.
$$

Therefore, it suffices to ensure

$$
\epsilon_s + 2\rho_h^t(s,a) \leq \frac{\sigma_s + \epsilon_s}{2} \iff \rho_h^t(s,a) \leq \frac{\sigma_s - \epsilon_s}{4}.
$$

Using the definiron of $\rho_h^t(s,a)$, the above condition is equivalent to

$$
n_h^t(s,a) \geq \frac{8\beta(n_h^t(s,a),\delta)}{(\sigma_s - \epsilon_s)^2}.
$$

Applying Lemma D.3 with

$$
\kappa = \frac{(\sigma_s - \epsilon_s)^2}{8}, \qquad b = |\mathcal{S}|, \qquad c = \log\left(\frac{2|\mathcal{S}||\mathcal{A}|H}{\delta}\right) + |\mathcal{S}|\log(8e),
$$

shows that $n_h^t(s,a) \geq n$ implies

$$
n_h^t(s,a) \geq \frac{8\beta(n_h^t(s,a),\delta)}{(\sigma_s - \epsilon_s)^2}.
$$

Hence,

$$
\rho_h^t(s,a) \leq \frac{\sigma_s - \epsilon_s}{4},
$$

and the removal criterion in (3) is satisfied.

If $(h,s,a) \in \hat{\mathcal{B}}^{t-1}$, then $(h,s,a) \in \mathcal{G}^t$, and therefore

$$
(h,s,a) \notin \hat{\mathcal{B}}^t = \hat{\mathcal{B}}^{t-1} \setminus \mathcal{G}^t.
$$

If $(h,s,a) \notin \hat{\mathcal{B}}^{t-1}$, the monotonicity $\hat{\mathcal{B}}^t \subseteq \hat{\mathcal{B}}^{t-1}$ from (i) of Lemma C.3 gives the same conclusion. □

Lemma C.5. *On the event $\mathcal{E} \cap \mathcal{E}^{\mathrm{cnt}}$, for every $(h,s,a) \notin \mathcal{B}$ and every $T \in \mathbb{N}$,*

$$\sum_{t=1}^{T} \mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}} d_h^{P^{\mathrm{real}},\pi^t}(s,a) \le 5n.$$

Proof of Lemma C.5. Fix $(h,s,a) \notin \mathcal{B}$ and $T \in \mathbb{N}$, and let $\mathcal{T} := \{t \in [T] : (h,s,a) \in \hat{\mathcal{B}}^t\}$. If $\mathcal{T} = \emptyset$, the claim holds trivially. Otherwise, Lemma C.3 shows that the sets $\hat{\mathcal{B}}^t$ are nonincreasing in t. Thus, $\mathcal{T}$ is a prefix of $[T]$. Let $t^\star := \max \mathcal{T}$. Since $(h,s,a) \in \hat{\mathcal{B}}^{t^\star}$ and $(h,s,a) \notin \mathcal{B}$, we have $n_h^{t^\star}(s,a) < n$ by Lemma C.4. On the event $\mathcal{E}^{\mathrm{cnt}}$, it follows that

$$\bar{n}_h^{t^\star}(s,a) \le 2\left(n_h^{t^\star}(s,a) + \beta^{\mathrm{cnt}}(\delta)\right) < 2\big(n + \beta^{\mathrm{cnt}}(\delta)\big).$$

Finally, by the definition of the pseudo-count $n_h^t(s,a)$ and the bound $d_h^{P^{\mathrm{real}},\pi^{t^\star}}(s,a) \le 1$,

$$\sum_{t\in\mathcal{T}} d_h^{P^{\mathrm{real}},\pi^t}(s,a) \le \bar{n}_h^{t^\star}(s,a) + d_h^{P^{\mathrm{real}},\pi^{t^\star}}(s,a) < 2\big(n + \beta^{\mathrm{cnt}}(\delta)\big) + 1 \le 5n,$$

where the last inequality uses $\beta^{\mathrm{cnt}}(\delta) \le n$ and $n \ge 1$. This proves the result. □

C.3 Proof of Lemma 3.1

Proof of Lemma 3.1. We condition on the event $\mathcal{E}$, which holds with high probability by Lemma C.1. For every $(h,s,a) \in \hat{\mathcal{B}}^t$, Pinsker's inequality on the event $\mathcal{E}$ gives

$$\left\|\hat{P}_h^t(\cdot \mid s,a),\, P_h^{\mathrm{real}}(\cdot \mid s,a)\right\|_{\mathrm{TV}} = \left\|\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a),\, P_h^{\mathrm{real}}(\cdot \mid s,a)\right\|_{\mathrm{TV}} \le \rho_h^t(s,a).$$

Moreover, Lemma C.3 ensures that $\mathcal{B} \subseteq \hat{\mathcal{B}}^t$. Together with Assumption 2.4, we have $\left\|\hat{P}_h^t(\cdot \mid s,a),\, P_h^{\mathrm{real}}(\cdot \mid s,a)\right\|_{\mathrm{TV}} \le \epsilon_s$ outside $\hat{\mathcal{B}}^t$. Therefore, for every (h,s,a),

$$\left\|\hat{P}_h^t(\cdot \mid s,a),\, P_h^{\mathrm{real}}(\cdot \mid s,a)\right\|_{\mathrm{TV}} \le \rho_h^t(s,a)\mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}} + \epsilon_s \mathbf{1}_{\{(h,s,a)\notin\hat{\mathcal{B}}^t\}}.$$

Applying the simulation lemma (Kalagarla et al., 2021, Lemma 4) and $0 \le V_{f,h+1}^{P^{\mathrm{real}},\pi} \le H - h$, we obtain

$$\begin{aligned} e_{f,1}^{t,\pi}(s_1) &= \left|\sum_{h=1}^{H}\sum_{s,a} d_h^{\hat{P}^t,\pi}(s,a) \sum_{s'\in\mathcal{S}} \left(\hat{P}_h^t(s' \mid s,a) - P_h^{\mathrm{real}}(s' \mid s,a)\right) V_{f,h+1}^{P^{\mathrm{real}},\pi}(s')\right| \\ &\le \sum_{h=1}^{H}\sum_{s,a} d_h^{\hat{P}^t,\pi}(s,a) b_h^t(s,a) = V_{b,1}^{\hat{P}^t,\pi}(s_1). \end{aligned}$$

Furthermore, both value functions lie in $[0,H]$, so $e_{f,1}^{t,\pi}(s_1) \le H$. Combining the two bounds yields

$$e_{f,1}^{t,\pi}(s_1) \le \min\left\{V_{b,1}^{\hat{P}^t,\pi}(s_1), H\right\}.$$

□

C.4 Proof of Theorem 4.1

Before proceeding with the proof of Theorem 4.1, we state the following lemma, which shows that once the estimation error is sufficiently small, solving the empirical CMDP under a slightly tightened safety constraint yields a policy that is feasible and near-optimal for the true CMDP. We then use this lemma to prove Theorem 4.1.

Lemma C.6 (Connection between estimation error and CMDPs)**.** *Suppose Assumption 2.3 holds. At iteration t, assume that there exists a constant $\tau \in (0, \xi/4]$ such that*

$$V_{c,1}^{\hat{P}^t,\pi^0}(s_1) \ge \ell + \frac{\xi}{2} \quad \textit{and} \quad \max_{\pi\in\Pi_{\mathrm{feas}}^{\hat{P}^t}} V_{b,1}^{\hat{P}^t,\pi}(s_1) \le \tau.$$

Then, on the high-probability event of Lemma 3.1, for any reward function $r \in \mathcal{F}$, every optimal solution

$$\pi^{\mathrm{out}} \in \arg\max_{\pi\in\Pi}\Big\{V_{r,1}^{\hat{P}^t,\pi}(s_1) : V_{c,1}^{\hat{P}^t,\pi}(s_1) \geq \ell + \tau\Big\}$$

is feasible for the true CMDP and satisfies

$$V_{r,1}^{P^{\mathrm{real}},\pi^\star}(s_1) - V_{r,1}^{P^{\mathrm{real}},\pi^{\mathrm{out}}}(s_1) \leq \frac{2H\tau}{\xi} + 2\tau,$$

where $\pi^\star \in \arg\max_{\pi\in\Pi_{\mathrm{feas}}^{P^{\mathrm{real}}}} V_{r,1}^{P^{\mathrm{real}},\pi}(s_1)$.

Proof of Lemma C.6. We work on the high-probability event of Lemma 3.1. On this event, for every $\pi \in \Pi$ and $f \in \mathcal{F}$,

$$\Big|V_{f,1}^{\hat{P}^t,\pi}(s_1) - V_{f,1}^{P^{\mathrm{real}},\pi}(s_1)\Big| \leq V_{b,1}^{\hat{P}^t,\pi}(s_1).$$

Since $\max_{\pi\in\Pi_{\mathrm{feas}}^{\hat{P}^t}} V_{b,1}^{\hat{P}^t,\pi}(s_1) \leq \tau$, it follows that

$$\max_{f\in\mathcal{F},\pi\in\Pi_{\mathrm{feas}}^{\hat{P}^t}} \Big|V_{f,1}^{\hat{P}^t,\pi}(s_1) - V_{f,1}^{P^{\mathrm{real}},\pi}(s_1)\Big| \leq \tau. \tag{15}$$

By the assumptions of the lemma, the baseline policy satisfies

$$V_{c,1}^{\hat{P}^t,\pi^0}(s_1) \geq \ell + \frac{\xi}{2}, \qquad V_{c,1}^{P^{\mathrm{real}},\pi^0}(s_1) \geq \ell + \xi. \tag{16}$$

Let $\pi^\star \in \arg\max_{\pi\in\Pi_{\mathrm{feas}}^{P^{\mathrm{real}}}} V_{r,1}^{P^{\mathrm{real}},\pi}(s_1)$. For $\lambda \in [0,1]$, we define $\pi^\lambda = \lambda\pi^0 + (1-\lambda)\pi^\star$. For every $f \in \mathcal{F}$ and $P \in \{\hat{P}^t, P^{\mathrm{real}}\}$, we have

$$V_{f,1}^{P,\pi^\lambda}(s_1) = (1-\lambda)V_{f,1}^{P,\pi^\star}(s_1) + \lambda V_{f,1}^{P,\pi^0}(s_1).$$

Consequently,

$$\max_{f\in\mathcal{F}} \Big|V_{f,1}^{\hat{P}^t,\pi^\lambda}(s_1) - V_{f,1}^{P^{\mathrm{real}},\pi^\lambda}(s_1)\Big| \leq (1-\lambda)V_{b,1}^{\hat{P}^t,\pi^\star}(s_1) + \lambda V_{b,1}^{\hat{P}^t,\pi^0}(s_1) = V_{b,1}^{\hat{P}^t,\pi^\lambda}(s_1).$$

To relate this mixture to a Markov policy, we define

$$d_h^\lambda(s,a) := (1-\lambda)d_h^{\hat{P}^t,\pi^\star}(s,a) + \lambda d_h^{\hat{P}^t,\pi^0}(s,a)$$

and construct $\tilde{\pi} \in \Pi$ by

$$\tilde{\pi}_h(a \mid s) := \frac{d_h^\lambda(s,a)}{\sum_{a'\in\mathcal{A}} d_h^\lambda(s,a')}$$

whenever the denominator is positive, choosing $\tilde{\pi}_h(\cdot \mid s)$ arbitrarily otherwise. Since the occupancy-measure constraints are linear, d^λ is valid under $\hat{P}^t$ and is induced by $\tilde{\pi}$. Thus, for every $u \in \mathcal{F}\cup\{b\}$,

$$V_{u,1}^{\hat{P}^t,\tilde{\pi}}(s_1) = V_{u,1}^{\hat{P}^t,\pi^\lambda}(s_1).$$

Therefore, $\pi^\lambda \in \Pi_{\mathrm{feas}}^{\hat{P}^t}$ impies

$$V_{b,1}^{\hat{P}^t,\pi^\lambda}(s_1) = V_{b,1}^{\hat{P}^t,\tilde{\pi}}(s_1) \leq \tau.$$

Since $\tau \leq \xi/2$, Equation (16) implies $V_{c,1}^{\hat{P}^t,\pi^0}(s_1) \geq \ell + \tau$. Hence,

$$\Lambda := \Big\{\lambda \in [0,1] : V_{c,1}^{\hat{P}^t,\pi^\lambda}(s_1) \geq \ell + \tau\Big\}$$

is nonempty. Define $\lambda_\star := \inf\Lambda$. By continuity of $\lambda \mapsto V_{c,1}^{\hat{P}^t,\pi^\lambda}(s_1)$, the set Λ is closed, so $\lambda_\star \in \Lambda$. Therefore,

$$V_{c,1}^{\hat{P}^t,\pi^{\lambda_\star}}(s_1) \geq \ell + \tau. \tag{17}$$

If $\lambda_\star = 0$, then $\lambda_\star \le 2\tau/\xi$ holds trivially. Otherwise, minimality and continuity imply

$$V_{c,1}^{\hat{P}^t,\pi^{\lambda_\star}}(s_1) = \ell + \tau.$$

Since $\pi^{\lambda_\star} \in \Pi_{\text{feas}}^{\hat{P}^t}$, we have

$$V_{c,1}^{P^{\text{real}},\pi^{\lambda_\star}}(s_1) \le V_{c,1}^{\hat{P}^t,\pi^{\lambda_\star}}(s_1) + \tau = \ell + 2\tau.$$

On the other hand, $\pi^\star \in \Pi_{\text{feas}}^{P^{\text{real}}}$ and Equation (16) give

$$V_{c,1}^{P^{\text{real}},\pi^{\lambda_\star}}(s_1) = (1-\lambda_\star)V_{c,1}^{P^{\text{real}},\pi^\star}(s_1) + \lambda_\star V_{c,1}^{P^{\text{real}},\pi^0}(s_1) \ge (1-\lambda_\star)\ell + \lambda_\star(\ell+\xi) = \ell + \lambda_\star\xi.$$

Combining these inequalities yields

$$\lambda_\star \le \frac{2\tau}{\xi}. \tag{18}$$

We next prove feasibility of π^{out}. Since it is feasible for the tightened empirical CMDP, we have $V_{c,1}^{\hat{P}^t,\pi^{\text{out}}}(s_1) \ge \ell + \tau$. Aplying Equation (15) gives

$$V_{c,1}^{P^{\text{real}},\pi^{\text{out}}}(s_1) \ge V_{c,1}^{\hat{P}^t,\pi^{\text{out}}}(s_1) - \tau \ge \ell.$$

Thus, π^{out} is feasible for the true CMDP.

Finally, Equation (17) and the occupancy-measure construction imply that $\tilde{\pi}$ is feasible for the tightened empirical CMDP. By optimality of π^{out},

$$V_{r,1}^{\hat{P}^t,\pi^{\text{out}}}(s_1) \ge V_{r,1}^{\hat{P}^t,\tilde{\pi}}(s_1) = V_{r,1}^{\hat{P}^t,\pi^{\lambda_\star}}(s_1).$$

Applying Equation (15) to π^{out}, we obtain

$$V_{r,1}^{P^{\text{real}},\pi^{\text{out}}}(s_1) \ge V_{r,1}^{P^{\text{real}},\pi^{\lambda_\star}}(s_1) - 2\tau. \tag{19}$$

Since rewards are nonnegative and bounded by 1,

$$V_{r,1}^{P^{\text{real}},\pi^{\lambda_\star}}(s_1) \ge (1-\lambda_\star)V_{r,1}^{P^{\text{real}},\pi^\star}(s_1)$$

and $V_{r,1}^{P^{\text{real}},\pi^\star}(s_1) \le H$. Therefore, Equation (18) gives

$$V_{r,1}^{P^{\text{real}},\pi^\star}(s_1) - V_{r,1}^{P^{\text{real}},\pi^{\lambda_\star}}(s_1) \le \lambda_\star H \le \frac{2H\tau}{\xi}.$$

Combining this inequality with Equation (19) yields

$$V_{r,1}^{P^{\text{real}},\pi^\star}(s_1) - V_{r,1}^{P^{\text{real}},\pi^{\text{out}}}(s_1) \le \frac{2H\tau}{\xi} + 2\tau.$$

This completes the proof. □

Now, we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. Below, we condition on $\mathcal{E} \cap \mathcal{E}^{\text{cnt}}$, which holds with probability at least $1-\delta$. We first prove the sample-complexity bound and then establish safe exploration and planning accuracy.

For this proof only, define the statistical bonus, which is supported on $\hat{\mathcal{B}}^t$, by

$$b_h^{t,\text{stat}}(s,a) \coloneqq H\rho_h^t(s,a)\mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}},$$

and define the transition kernel

$$P_h^{\dagger,t}(\cdot \mid s,a) \coloneqq \hat{P}_h^{t,\text{real}}(\cdot \mid s,a)\mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}} + P_h^{\text{real}}(\cdot \mid s,a)\mathbf{1}_{\{(h,s,a)\notin\hat{\mathcal{B}}^t\}}.$$

By the triangle inequality and the simulation lemma (Kalagarla et al., 2021, Lemma 4), for every $\pi \in \Pi$,

$$\left|V_{b,1}^{\hat{P}^t,\pi}(s_1) - V_{b^{t,\text{stat}},1}^{P^{\dagger,t},\pi}(s_1)\right| \le \left|V_{b,1}^{\hat{P}^t,\pi}(s_1) - V_{b^{t,\text{stat}},1}^{\hat{P}^t,\pi}(s_1)\right| + \left|V_{b^{t,\text{stat}},1}^{\hat{P}^t,\pi}(s_1) - V_{b^{t,\text{stat}},1}^{P^{\dagger,t},\pi}(s_1)\right|$$

$$\leq H^2\epsilon_s + \left|\sum_{h=1}^{H}\sum_{s,a} d_h^{\hat{P}^t,\pi}(s,a)\sum_{s'\in\mathcal{S}}\left(\hat{P}_h^t(s' \mid s,a) - P_h^{\dagger,t}(s' \mid s,a)\right)V_{b^{t,\mathrm{stat}},h+1}^{P^{\dagger,t},\pi}(s')\right|$$

$$\leq H^2\epsilon_s + H\epsilon_s\sum_{h=1}^{H}(H-h) \leq H^3\epsilon_s. \tag{20}$$

We next lower-bound $V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1)$ for every executed iteration. We consider two cases. First, suppose the condition in Line 5 fails, so the algorithm executes the mixture policy π^t. Since the algorithm has not terminated at iteration $t < T$, the stopping condition gives

$$V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) > \tau.$$

Inequality (20) and the definition of α^t yield

$$V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1) = \alpha^t V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\bar{\pi}^t}(s_1) + (1-\alpha^t)V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^0}(s_1) \geq \alpha^t\left(V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) - H^3\epsilon_s\right)$$

$$\geq \frac{\xi\left(V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) - H^3\epsilon_s\right)}{\xi + V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)} \geq \frac{\xi(\tau - H^3\epsilon_s)}{\xi+\tau}.$$

Using $\tau - H^3\epsilon_s = \xi\epsilon/(4H)$ and $\tau \leq \xi/4$, we obtain

$$V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1) \geq \frac{\xi\epsilon}{5H}. \tag{21}$$

Second, suppose the condition in Line 5 holds. Then

$$V_{c,1}^{\hat{P}^t,\pi^0}(s_1) < \ell + \frac{\xi}{2},$$

and the algorithm sets $\pi^t = \pi^0$. Assumption 2.3 and Lemma 3.1 imply

$$V_{b,1}^{\hat{P}^t,\pi^t}(s_1) = V_{b,1}^{\hat{P}^t,\pi^0}(s_1) \geq \left|V_{c,1}^{P^{\mathrm{real}},\pi^0}(s_1) - V_{c,1}^{\hat{P}^t,\pi^0}(s_1)\right| > \frac{\xi}{2}.$$

Applying Inequality (20) gives

$$V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1) \geq V_{b,1}^{\hat{P}^t,\pi^t}(s_1) - H^3\epsilon_s \geq \frac{\xi}{2} - H^3\epsilon_s \geq \frac{\xi\epsilon}{5H},$$

where the final inequality uses $\epsilon_s \leq \xi/(4H^3)$ and $\epsilon \leq 1$.

Thus, Inequality (21) holds for every executed iteration. Summing over these iterations yields

$$\sum_{t=1}^{T} V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1) \geq \frac{\xi\epsilon T}{5H}. \tag{22}$$

Next, we derive an upper bound on $V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi}(s_1)$. We first show that, for every $(h,s,a) \in [H]\times\mathcal{S}\times\mathcal{A}$ and every $w : \mathcal{S} \to [0,H]$,

$$\sum_{s'\in\mathcal{S}} P_h^{\dagger,t}(s' \mid s,a)w(s') \leq 2H^2(\rho_h^t(s,a))^2\mathbf{1}_{\{(h,s,a)\in\hat{\mathcal{B}}^t\}} + \left(1+\frac{1}{H}\right)\sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a)w(s'). \tag{23}$$

This inequality follows by considering three cases. First, if $(h,s,a) \notin \hat{\mathcal{B}}^t$, then $P_h^{\dagger,t}(\cdot \mid s,a) = P_h^{\mathrm{real}}(\cdot \mid s,a)$, so the inequality holds immediately. Second, if $(h,s,a) \in \hat{\mathcal{B}}^t$ and $\rho_h^t(s,a) = 1$, then the left-hand side is at most $H \leq 2H^2$. Third, suppose that $(h,s,a) \in \hat{\mathcal{B}}^t$ and $\rho_h^t(s,a) < 1$. Applying Lemma D.4 with $q = P_h^{\dagger,t}(\cdot \mid s,a) = \hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a)$, $p = P_h^{\mathrm{real}}(\cdot \mid s,a)$, and $\lambda = 1/H^2$ gives

$$\sum_{s'\in\mathcal{S}} \hat{P}_h^{t,\mathrm{real}}(s' \mid s,a)w(s')$$

$$\leq H^2 D_{\mathrm{KL}}\left(\hat{P}_h^{t,\mathrm{real}}(\cdot \mid s,a), P_h^{\mathrm{real}}(\cdot \mid s,a)\right) + H^2 \log\left(\sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) e^{w(s')/H^2}\right)$$

$$\leq 2H^2\left(\rho_h^t(s,a)\right)^2 + H^2 \log\left(\sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) e^{w(s')/H^2}\right), \tag{24}$$

where the last inequality follows from event $\mathcal{E}$. Meanwhile, the convexity of $z \mapsto e^{\lambda z}$ on $[0,H]$ gives

$$e^{\lambda z} \leq 1 + \frac{z}{H}\left(e^{\lambda H} - 1\right), \forall z \in [0,H].$$

Setting $\lambda = 1/H^2$ and $z = w(s')$, we obtain

$$\sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) e^{w(s')/H^2} \leq 1 + \frac{e^{1/H}-1}{H} \sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) w(s').$$

Using $\log(1+y) \leq y$ for $y \geq 0$, together with $e^y \leq 1 + y + y^2$ for $y \in [0,1]$ and $y = 1/H$, yields

$$\begin{aligned} H^2 \log\left(\sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) e^{w(s')/H^2}\right) &\leq H\left(e^{1/H}-1\right) \sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) w(s') \\ &\leq \left(1 + \frac{1}{H}\right) \sum_{s'\in\mathcal{S}} P_h^{\mathrm{real}}(s' \mid s,a) w(s'). \end{aligned}$$

Substituting this bound into Inequality (24) proves Inequality (23).

Expanding $V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1)$ into its bonus terms, recursively applying Inequality (23) to each term with $w(s) = V_{b^{t,\mathrm{stat}},h+1}^{P^{\dagger,t},\pi^t}(s)/H$, using $(1+1/H)^H \leq e$, and multiplying the resulting inequality through by H, we obtain

$$V_{b^{t,\mathrm{stat}},1}^{P^{\dagger,t},\pi^t}(s_1) \leq \underbrace{e \sum_{h,s,a} d_h^{P^{\mathrm{real}},\pi^t}(s,a)\, b_h^{t,\mathrm{stat}}(s,a)}_{(i)^t} + \underbrace{2eH \sum_{h,s,a} d_h^{P^{\mathrm{real}},\pi^t}(s,a)\left(b_h^{t,\mathrm{stat}}(s,a)\right)^2}_{(ii)^t}.$$

We first bound Term $(i)^t$. For every $(h,s,a) \in \mathcal{B}$, Lemma C.3 ensures that $(h,s,a) \in \hat{\mathcal{B}}^t$, and hence $b_h^{t,\mathrm{stat}}(s,a) = H\rho_h^t(s,a)$. Therefore,

$$\begin{aligned} \sum_{t=1}^{T} d_h^{P^{\mathrm{real}},\pi^t}(s,a) b_h^{t,\mathrm{stat}}(s,a) &= H \sum_{t=1}^{T} d_h^{P^{\mathrm{real}},\pi^t}(s,a) \rho_h^t(s,a) \\ &\overset{(1)}{\leq} H \sum_{t=1}^{T} d_h^{P^{\mathrm{real}},\pi^t}(s,a) \min\left\{1, \sqrt{\frac{4\beta(\bar{n}_h^t(s,a),\delta)}{\bar{n}_h^t(s,a) \vee 1}}\right\} \\ &\overset{(2)}{\leq} 4\sqrt{2}H\sqrt{\beta(\bar{n}_h^{T+1}(s,a),\delta)\bar{n}_h^{T+1}(s,a)} \\ &\overset{(3)}{\leq} 4\sqrt{2}H\sqrt{\beta(T,\delta)\bar{n}_h^{T+1}(s,a)}, \end{aligned}$$

where Step (1) follows from Lemma C.2, Step (2) follows from Lemma D.6, and Step (3) uses the monotonicity of $\beta(\cdot,\delta)$ and $\bar{n}_h^{T+1}(s,a) \leq T$.

Summing over $(h,s,a) \in \mathcal{B}$ and applying the Cauchy–Schwarz inequality gives

$$\begin{aligned} \sum_{(h,s,a)\in\mathcal{B}} \sum_{t=1}^{T} d_h^{P^{\mathrm{real}},\pi^t}(s,a) b_h^{t,\mathrm{stat}}(s,a) &\leq 4\sqrt{2}H\sqrt{\beta(T,\delta)|\mathcal{B}|}\sqrt{\sum_{(h,s,a)\in\mathcal{B}} \bar{n}_h^{T+1}(s,a)} \\ &\leq 4\sqrt{2}H\sqrt{\beta(T,\delta)|\mathcal{B}|HT}, \end{aligned}$$

where the last inequality uses

$$\sum_{(h,s,a)\in\mathcal{B}} \bar{n}_h^{T+1}(s,a) \le \sum_{t=1}^{T}\sum_{h,s,a} d_h^{P^{\text{real}},\pi^t}(s,a) = HT.$$

Next, fix $(h,s,a)\notin\mathcal{B}$ and define

$$\mathcal{T}_{h,s,a} := \{t\in[T] : (h,s,a)\in\hat{\mathcal{B}}^t\}.$$

If $\mathcal{T}_{h,s,a}=\emptyset$, it contributes 0 to $(i)^t$. Otherwise, let $t^\star := \max\mathcal{T}_{h,s,a}$. Since $\mathcal{T}_{h,s,a}$ is a prefix of $[T]$, applying Lemmas C.2 and D.6 up to $t^\star$ yields

$$\begin{aligned}\sum_{t=1}^{T} d_h^{P^{\text{real}},\pi^t}(s,a) b_h^{t,\text{stat}}(s,a) &= H\sum_{t\in\mathcal{T}_{h,s,a}} d_h^{P^{\text{real}},\pi^t}(s,a)\rho_h^t(s,a)\\ &\le 4\sqrt{2}H\sqrt{\beta(\bar{n}_h^{t^\star+1}(s,a),\delta)\bar{n}_h^{t^\star+1}(s,a)}.\end{aligned}$$

By Lemma C.5,

$$\bar{n}_h^{t^\star+1}(s,a) = \sum_{t\in\mathcal{T}_{h,s,a}} d_h^{P^{\text{real}},\pi^t}(s,a) \le 5n,$$

where the last inequality uses $\beta^{\text{cnt}}(\delta)\le n$ and $n\ge 1$. Consequently,

$$\sum_{t=1}^{T} d_h^{P^{\text{real}},\pi^t}(s,a) b_h^{t,\text{stat}}(s,a) \le 4\sqrt{2}H\sqrt{5\beta(5n,\delta)n}.$$

Summing this bound over all non-mismatch triples gives

$$\sum_{(h,s,a)\notin\mathcal{B}}\sum_{t=1}^{T} d_h^{P^{\text{real}},\pi^t}(s,a) b_h^{t,\text{stat}}(s,a) \le 4\sqrt{2}H\big(H|\mathcal{S}||\mathcal{A}|-|\mathcal{B}|\big)\sqrt{5\beta(5n,\delta)n}.$$

Combining the bounds for mismatch and non-mismatch triples, we obtain

$$\sum_{t=1}^{T}(i)^t \le 4\sqrt{2}eH\left(\sqrt{\beta(T,\delta)|\mathcal{B}|HT} + \big(H|\mathcal{S}||\mathcal{A}|-|\mathcal{B}|\big)\sqrt{5\beta(5n,\delta)n}\right). \tag{25}$$

Bounding the second term $\sum_{t=1}^{T}(ii)^t$ is similar as bounding the first term $\sum_{t=1}^{T}(i)^t$, using the second bound of Lemma D.6 instead of the first, we have

$$\sum_{t=1}^{T}(ii)^t \le 16eH^3\big[|\mathcal{B}|\Phi(T) + (H|\mathcal{S}||\mathcal{A}|-|\mathcal{B}|)\Phi(5n)\big], \tag{26}$$

where $\Phi(z)=\beta(z,\delta)\big[1+\log_+(z/4\beta(z,\delta))\big]$.

Combining Inequalities (22), (25) and (26) gives

$$\frac{\xi\epsilon T}{5H} \le 4\sqrt{2}eH\big(\sqrt{\beta(T,\delta)|\mathcal{B}|HT} + N_0\sqrt{5\beta(5n,\delta)n}\big) + 16eH^3\big[|\mathcal{B}|\Phi(T) + N_0\Phi(5n)\big].$$

where $M := H|\mathcal{S}||\mathcal{A}|$ and $N_0 := M-|\mathcal{B}|$. For convenience, define

$$\begin{aligned}\eta &:= \frac{\xi\epsilon}{5H}, \qquad A_T := 4\sqrt{2}eH\sqrt{\beta(T,\delta)|\mathcal{B}|H},\\ R_T &:= 4\sqrt{2}eHN_0\sqrt{5\beta(5n,\delta)n} + 16eH^3\left[|\mathcal{B}|\Phi(T) + N_0\Phi(5n)\right].\end{aligned}$$

The preceding inequality can then be written as

$$\eta T \le A_T\sqrt{T} + R_T. \tag{27}$$

By Young's inequality,

$$A_T\sqrt{T} \le \frac{\eta}{2}T + \frac{A_T^2}{2\eta}.$$

Substituting this bound into Inequality (27) and rearranging gives

$$T \le \frac{A_T^2}{\eta^2} + \frac{2R_T}{\eta}.$$

Since $A_T^2 = 32e^2H^3|\mathcal{B}|\beta(T,\delta)$, we obtain

$$T \le \frac{32e^2H^3|\mathcal{B}|\beta(T,\delta)}{\eta^2} + \frac{8\sqrt{2}eHN_0\sqrt{5\beta(5n,\delta)n}}{\eta} + \frac{32eH^3\left[|\mathcal{B}|\Phi(T) + N_0\Phi(5n)\right]}{\eta}. \tag{28}$$

We also derive an alternative bound that does not rely on the removal of non-mismatch triples. Specifically, applying the same argument to Terms $(i)^t$ and $(ii)^t$ while treating all $M = H|\mathcal{S}||\mathcal{A}|$ triples as mismatch triples yields

$$T \le \frac{32e^2H^3M\beta(T,\delta)}{\eta^2} + \frac{32eH^3M\Phi(T)}{\eta}. \tag{29}$$

It remains to resolve the dependence on T on the right-hand sides of these bounds. Since

$$\beta(T,\delta) = \log\left(\frac{2|\mathcal{S}||\mathcal{A}|H}{\delta}\right) + |\mathcal{S}|\log(8e(T+1)),$$

grows linearly in $\log(e(T+1))$, whereas

$$\Phi(T) = \beta(T,\delta)\left[1 + \log_+\left(\frac{T}{4\beta(T,\delta)}\right)\right]$$

grows at most quadratically in $\log(e(T+1))$. Hence, the right-hand side of each of the bounds in (29) and (28) is at most $C\log^2(e(T+1))$ for some $C \ge 1$ that depends on the problem parameters but not on T. Lemma D.7 therefore implies

$$T \le 64C\log^2(64eC).$$

Thus, the algorithm terminates after finitely many iterations, and the remaining dependence on T contributes only logarithmic factors.In particular,

$$\beta(T,\delta) = \widetilde{\mathcal{O}}(|\mathcal{S}|), \qquad \Phi(T) = \widetilde{\mathcal{O}}(|\mathcal{S}|).$$

By Equation (14), we obtain

$$n = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|}{(\sigma_s - \epsilon_s)^2}\right), \qquad \sqrt{\beta(5n,\delta)n} = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}|}{\sigma_s - \epsilon_s}\right), \qquad \Phi(5n) = \widetilde{\mathcal{O}}(|\mathcal{S}|).$$

Moreover, $\frac{1}{\eta} = \mathcal{O}\left(\frac{H}{\xi\epsilon}\right)$. Substituting these estimates into (28) gives

$$\begin{aligned}
T &\le \widetilde{\mathcal{O}}\left(\frac{H^5|\mathcal{S}||\mathcal{B}|}{\xi^2\epsilon^2} + \frac{H^2|\mathcal{S}|N_0}{\xi\epsilon(\sigma_s - \epsilon_s)} + \frac{H^4|\mathcal{S}|M}{\xi\epsilon}\right) \\
&= \widetilde{\mathcal{O}}\left(\frac{H^4|\mathcal{S}||\mathcal{B}|}{\xi\epsilon}\left(\frac{H}{\xi\epsilon} + 1\right) + \frac{H^2|\mathcal{S}|(H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|)}{\xi\epsilon}\left(\frac{1}{(\sigma_s - \epsilon_s)} + H^2\right)\right) \\
&= \widetilde{\mathcal{O}}\left(\frac{H^5|\mathcal{S}||\mathcal{B}|}{\xi^2\epsilon^2} + \frac{H^4|\mathcal{S}|(H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|)}{\xi\epsilon}\left(\frac{1}{(\sigma_s - \epsilon_s)H^2} + 1\right)\right),
\end{aligned}$$

where first equality uses defintion of M and N_0 and second equality uses $\epsilon \le 1$ and $\xi \le H$.

Since each iteration contains H real-world samples, the total sample complexity satisfies

$$HT \le \widetilde{\mathcal{O}}\left(\frac{H^6|\mathcal{S}||\mathcal{B}|}{\xi^2\epsilon^2} + \frac{H^5|\mathcal{S}|(H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|)}{\xi\epsilon}\left(\frac{1}{(\sigma_s - \epsilon_s)H^2} + 1\right)\right).$$

Similarly, substituting these estimates into (29) gives

$$HT \le \widetilde{\mathcal{O}}\left(\frac{H^7|\mathcal{S}|^2|\mathcal{A}|}{\xi^2\epsilon^2}\right).$$

Both bounds hold for the same algorithm, so we may take their minimum and conclude that the total sample complexity satisfies

$$HT \le \widetilde{\mathcal{O}}\Bigg(\min\Bigg\{\frac{H^7|\mathcal{S}|^2|\mathcal{A}|}{\xi^2\epsilon^2}, \frac{H^6|\mathcal{S}||\mathcal{B}|}{\xi^2\epsilon^2} + \frac{H^5|\mathcal{S}|\big(H|\mathcal{S}||\mathcal{A}| - |\mathcal{B}|\big)}{\xi\epsilon}\left(1 + \frac{1}{(\sigma_s - \epsilon_s)H^2}\right)\Bigg\}\Bigg).$$

Safe exploration. If $V_{c,1}^{\hat{P}^t,\pi^0}(s_1) < \ell + \xi/2$, we execute π^0, which is safe by Assumption 2.3. Otherwise, we execute the mixture policy π^t, which satisfies

$$\begin{aligned}
V_{c,1}^{P^{\mathrm{real}},\pi^t} &= \alpha^t V_{c,1}^{P^{\mathrm{real}},\bar{\pi}^t} + (1-\alpha^t)V_{c,1}^{P^{\mathrm{real}},\pi^0} \\
&\overset{(i)}{\ge} \alpha^t\left(V_{c,1}^{\hat{P}^t,\bar{\pi}^t} - \min\left\{H, V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)\right\}\right) + (1-\alpha^t)(\xi+\ell) \\
&= \xi + \ell - \alpha^t\left(\min\left\{H, V_{b,1}^{\hat{P}^t,\bar{\pi}^t}(s_1)\right\} + \ell - V_{c,1}^{\hat{P}^t,\bar{\pi}^t}(s_1) + \xi\right) \\
&\overset{(ii)}{\ge} \ell.
\end{aligned}$$

Here, step (i) follows from Assumption 2.3 and Lemma 3.1, while step (ii) follows from the definition of α^t. Therefore, safe exploration is guaranteed.

Planning accuracy. Algorithm 1 terminates when $V_{c,1}^{\hat{P}^t,\pi^0}(s_1) \ge \ell + \xi/2$, and ϵ is chosen such that $\tau \le \xi/4$. Therefore, by Lemma C.6, the output policy $\pi^{\mathrm{out}} \in \Pi_{\mathrm{feas}}^{P^{\mathrm{real}}}$ and satisfies

$$V_{r,1}^{P^{\mathrm{real}},\pi^\star}(s_1) - V_{r,1}^{P^{\mathrm{real}},\pi^{\mathrm{out}}}(s_1) \le \left(2 + \frac{2H}{\xi}\right)\tau = \frac{\xi + H}{2H}\epsilon + \left(2 + \frac{2H}{\xi}\right)H^3\epsilon_s \le \epsilon + \frac{4H^4}{\xi}\epsilon_s,$$

where the final inequality uses $\xi \le H$. □

D SUPPORTING LEMMAS

We first state a deviation inequality for empirical distributions of i.i.d. categorical samples, which will be used to control the estimation error of empirical transition models. Let $(X_t)_{t\in\mathbb{N}}$ be i.i.d. samples from a distribution supported on $[m]$ with probability vector $p \in \Delta([m])$. We denote by $\hat{p}_n$ the empirical distribution, i.e.,

$$\hat{p}_{n,k} = \frac{1}{n}\sum_{\ell=1}^{n}\mathbf{1}_{\{X_\ell = k\}},\ k \in [m].$$

Lemma D.1 ((Jonsson et al., 2020, Proposition 1)). *For all $p \in \Delta([m])$ and all $\delta \in (0,1)$,*

$$\Pr\left(\exists n \in \mathbb{N},\ nD_{\mathrm{KL}}(\hat{p}_n, p) > \log(1/\delta) + (m-1)\log\left(e\left(1 + \frac{n}{m-1}\right)\right)\right) \le \delta.$$

Next, we state a deviation inequality for adapted Bernoulli random variables, which will be used to control state-action visitation counts. Let $(\mathcal{F}_t)_{t\in\mathbb{N}}$ be a filtration and let $(X_t)_{t\in\mathbb{N}}$ be Bernoulli random variables such that X_t is $\mathcal{F}_t$-measurable and

$$\Pr(X_t = 1 \mid \mathcal{F}_{t-1}) = P_t,$$

where P_t is $\mathcal{F}_{t-1}$-measurable.

Lemma D.2 ((Dann et al., 2017, Lemma F.4)). *For all $\delta > 0$,*

$$\Pr\left(\exists n \in \mathbb{N} : \sum_{t=1}^{n} X_t < \frac{1}{2}\sum_{t=1}^{n} P_t - \log\frac{1}{\delta}\right) \le \delta.$$

Lemma D.3. *Let $c \geq 0$ and $b \geq 1$, and suppose $f : \mathbb{N} \to \mathbb{R}_+$ satisfies $f(m) \leq c + b + b\log(m+1)$ for all $m \in \mathbb{N}$. Then, for any $\kappa > 0$, define*

$$m^\dagger := 1 + \frac{2c}{\kappa} + \frac{2b}{\kappa} \log\left(\max\left\{\frac{2b}{\kappa}, e\right\}\right).$$

Then $m \geq m^\dagger$ implies $m \geq f(m)/\kappa$.

Proof. Let $t := \max\{2b/\kappa, e\}$. By concavity of the logarithm, for all $z, t > 0$,

$$\log z \leq \log t + \frac{z-t}{t} = \frac{z}{t} + \log t - 1.$$

Applying the above inequality with $z = m+1$ and multiplying by b gives

$$b\log(m+1) \leq \frac{b(m+1)}{t} + b\log t - b.$$

Since $t \geq 2b/\kappa$, we have $b/t \leq \kappa/2$, and hence

$$b\log(m+1) \leq \frac{\kappa(m+1)}{2} + b\log t - b.$$

By definition of $f(m)$, we obtain

$$f(m) \leq c + b + b\log(m+1) \leq c + \frac{\kappa(m+1)}{2} + b\log t.$$

Therefore, $\kappa m \geq f(m)$ holds whenever

$$\kappa m \geq c + \frac{\kappa(m+1)}{2} + b\log t.$$

The last condition is equivalent to

$$m \geq 1 + \frac{2c}{\kappa} + \frac{2b}{\kappa}\log t = 1 + \frac{2c}{\kappa} + \frac{2b}{\kappa}\log\left(\max\left\{\frac{2b}{\kappa}, e\right\}\right) = m^\dagger.$$

Thus $m \geq m^\dagger$ implies $\kappa m \geq f(m)$, equivalently $m \geq f(m)/\kappa$. □

Lemma D.4. *Let $\mathcal{X}$ be a finite set, let $p, q \in \Delta(\mathcal{X})$ satisfy $D_{\mathrm{KL}}(q,p) < \infty$, and let $w : \mathcal{X} \to \mathbb{R}$. Then, for every $\lambda > 0$,*

$$\sum_{x\in\mathcal{X}} q(x)w(x) \leq \frac{1}{\lambda}\left(D_{\mathrm{KL}}(q,p) + \log\left(\sum_{x\in\mathcal{X}} p(x)e^{\lambda w(x)}\right)\right).$$

Proof of Lemma D.4. Let

$$Z := \sum_{x\in\mathcal{X}} p(x)e^{\lambda w(x)} \qquad \text{and} \qquad p_\lambda(x) := \frac{p(x)e^{\lambda w(x)}}{Z}.$$

Since $D_{\mathrm{KL}}(q,p) < \infty$, the support of q is contained in that of p, which coincides with the support of p_λ. Therefore,

$$0 \leq D_{\mathrm{KL}}(q,p_\lambda) = \sum_{x\in\mathcal{X}} q(x)\log\left(\frac{q(x)Z}{p(x)e^{\lambda w(x)}}\right) = D_{\mathrm{KL}}(q,p) - \lambda\sum_{x\in\mathcal{X}} q(x)w(x) + \log Z.$$

Rearranging the above inequality completes the proof. □

Lemma D.5. *Let $\beta(z,\delta) = \log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}|\log(8e(z+1))$ with $\delta \in (0,1)$. Then*

- *(i) $z \mapsto \beta(z,\delta)$ is nondecreasing on $[0,\infty)$ and $\beta(z,\delta) \geq 1$ for every $z \geq 0$;*
- *(ii) $z \mapsto \beta(z,\delta)/z$ is nonincreasing on $(0,\infty)$; equivalently, $z \mapsto z/\beta(z,\delta)$ is nondecreasing on $[0,\infty)$.*

Proof of Lemma D.5. For part (i), $\beta(z,\delta)$ is nondecreasing in z follows from the fact that $\log(8e(z+1))$ is nondecreasing in z. Moreover, since $|\mathcal{S}||\mathcal{A}|H \geq 1$, $\delta < 1$, and $|\mathcal{S}| \geq 1$,

$$\beta(z,\delta) \geq \log 2 + \log(8e) > 1.$$

For part (ii), define $\varphi(z) := \beta(z,\delta)/z$ for $z > 0$. Its derivative is

$$\varphi'(z) = \frac{z|\mathcal{S}| - (z+1)\beta(z,\delta)}{(z+1)z^2} \leq \frac{|\mathcal{S}| - \beta(z,\delta)}{z^2} \leq 0.$$

Hence, $\varphi(z)$ is nonincreasing on $(0,\infty)$. Equivalently, $z \mapsto z/\beta(z,\delta)$ is nondecreasing on $[0,\infty)$, with value 0 at $z = 0$, since $\varphi(z) = \beta(z,\delta)/z \to +\infty$ as $z \downarrow 0$ □

Lemma D.6. *For real $z \geq 0$, define*

$$q(z) = \min\left\{1, \sqrt{\frac{4\beta(z,\delta)}{z \vee 1}}\right\}, \qquad \Phi(z) = \beta(z,\delta)\left[1 + \log_+ \frac{z}{4\beta(z,\delta)}\right],$$

where $\beta(z,\delta) = \log(2|\mathcal{S}||\mathcal{A}|H/\delta) + |\mathcal{S}|\log(8e(z+1))$. Then

(i) the functions $z \mapsto \sqrt{\beta(z,\delta)z}$ and $z \mapsto \Phi(z)$ are nondecreasing on $[0,\infty)$;

(ii) for every $m \in \mathbb{N}$ and all $v_1, \ldots, v_m \in [0,1]$, setting $U_i := \sum_{j<i} v_j$ and $M := \sum_{i=1}^m v_i$,

$$\sum_{i=1}^m v_i\, q(U_i) \leq 4\sqrt{2}\sqrt{\beta(M,\delta)M}, \qquad \sum_{i=1}^m v_i\, q(U_i)^2 \leq 8\Phi(M).$$

Proof of Lemma D.6. Part (i). The map $z \mapsto \sqrt{\beta(z,\delta)z}$ is the product of two nonnegative, nondecreasing functions and is therefore nondecreasing by (i) from Lemma D.5.

To analyze Φ, define $\psi(z) := z/(4\beta(z,\delta))$, which is nondecreasing by (ii) from Lemma D.5. On the region where $\psi(z) > 1$, $\Phi(z) = \beta(z,\delta)\big(1 + \log\psi(z)\big)$, which is also nondecreasing because both factors are nonnegative and nondecreasing. The two expressions agree when $\psi(z) = 1$, so Φ is nondecreasing on $[0,\infty)$.

Part (ii). If $M = 0$, then $v_i = 0$ for every i, and both bounds holds immediately. Assume $M > 0$, and define

$$a := 4\beta(M,\delta) \geq 4, \qquad f(z) := \min\left\{1, \sqrt{\frac{a}{z}}\right\} \text{ for } z > 0, \qquad f(0) := 1.$$

The function f is nonincreasing. We use it as a upper bound for q.

Claim 1. For every i, $q(U_i) \leq f(U_i)$.

If $U_i = 0$, both sides equal 1. If $U_i > 0$, then $U_i \leq M$ and (i) from Lemma D.5 give

$$\frac{4\beta(U_i,\delta)}{U_i \vee 1} \leq \frac{4\beta(M,\delta)}{U_i} = \frac{a}{U_i}.$$

The claim follows from the definitions of q and f.

Claim 2. For every $z \in [U_i, U_i + v_i]$,

$$f(U_i) \leq \sqrt{2}\, f(z).$$

If $U_i < a$, then $f(U_i) = 1$. Since $v_i \leq 1$,

$$z \leq U_i + v_i < a + 1,$$

and hence

$$f(z) \geq \sqrt{\frac{a}{a+1}} \geq \sqrt{\frac{4}{5}} \geq \frac{1}{\sqrt{2}},$$

where we used $a \geq 4$. If $U_i \geq a$, then $z \geq a$, and

$$\frac{f(U_i)}{f(z)} = \sqrt{\frac{z}{U_i}} \leq \sqrt{1 + \frac{v_i}{U_i}} \leq \sqrt{1 + \frac{1}{a}} \leq \sqrt{\frac{5}{4}} \leq \sqrt{2}.$$

Thus, the claim holds in both cases. Taking the square of both sides also gives $f(U_i)^2 \leq 2f(z)^2$.

We now convert the sums into integrals. Since $U_1 = 0$ and $U_{i+1} = U_i + v_i$, the intervals $[U_i, U_i + v_i]$ cover $[0, M]$ consecutively. Claims 1 and 2 imply

$$v_i q(U_i) \leq \int_{U_i}^{U_i+v_i} \sqrt{2}\, f(z)\, \mathrm{d}z, \qquad v_i q(U_i)^2 \leq \int_{U_i}^{U_i+v_i} 2f(z)^2\, \mathrm{d}z.$$

Summing over i yields

$$\sum_{i=1}^{m} v_i q(U_i) \leq \sqrt{2} \int_0^M f(z)\, \mathrm{d}z, \qquad \sum_{i=1}^{m} v_i q(U_i)^2 \leq 2 \int_0^M f(z)^2\, \mathrm{d}z.$$

For the first integral, if $M \leq a$, then $f(z) = 1$ on $[0, M]$, so

$$\int_0^M f(z)\, \mathrm{d}z = M \leq \sqrt{aM}.$$

If $M > a$, then

$$\int_0^M f(z)\, \mathrm{d}z = a + \int_a^M \sqrt{\frac{a}{z}}\, \mathrm{d}z = 2\sqrt{aM} - a \leq 2\sqrt{aM}.$$

Therefore,

$$\sum_{i=1}^{m} v_i q(U_i) \leq 2\sqrt{2aM} = 4\sqrt{2}\sqrt{\beta(M,\delta)M}.$$

For the second integral, if $M \leq a$, then $\int_0^M f(z)^2\, \mathrm{d}z = M \leq a$. If $M > a$, then

$$\int_0^M f(z)^2\, \mathrm{d}z = a + \int_a^M \frac{a}{z}\, \mathrm{d}z = a\left(1 + \log\frac{M}{a}\right).$$

Thus, in both cases,

$$\int_0^M f(z)^2\, \mathrm{d}z \leq a\left(1 + \log_+\frac{M}{a}\right).$$

Using $a = 4\beta(M,\delta)$, we conclude that

$$\sum_{i=1}^{m} v_i q(U_i)^2 \leq 2a\left(1 + \log_+\frac{M}{a}\right) = 8\beta(M,\delta)\left(1 + \log_+\frac{M}{4\beta(M,\delta)}\right) = 8\Phi(M).$$

□

Lemma D.7. *Let $C \geq 1$ and suppose that $T \geq 0$ satisfies $T \leq C\log^2\big(e(T+1)\big)$. Then,*

$$T \leq 64C\log^2(64eC).$$

Proof of Lemma D.7. Define

$$g(z) \coloneqq \frac{z}{\log^2(e(z+1))}, \qquad z \geq 0.$$

We first show that g is increasing. Indeed,

$$g'(z) = \frac{1}{(1+\log(1+z))^3}\left(1 + \log(1+z) - \frac{2z}{1+z}\right).$$

Note that the expression in parentheses is minimized at $z = 1$, where it equals $\log 2 > 0$. Hence, $g'(z) > 0$ for all $z \geq 0$.

Now set

$$x \coloneqq 64eC, \qquad L \coloneqq \log x, \qquad T_0 \coloneqq 64CL^2.$$

We next show that $g(T_0) \geq C$. Since $x \geq 64e$, we have $L^2 \leq x$ and $e \leq x$. Therefore,

$$e(T_0 + 1) = xL^2 + e \leq x^2 + x \leq x^8.$$

Taking logarithms gives

$$\log\big(e(T_0 + 1)\big) \leq 8 \log x = 8L.$$

Consequently,

$$g(T_0) = \frac{T_0}{\log^2(e(T_0 + 1))} \geq \frac{64CL^2}{64L^2} = C.$$

The hypothesis $T \leq C \log^2(e(T + 1))$ gives $g(T) \leq C$. If $T > T_0$, the monotonicity of g would imply $g(T) > g(T_0) \geq C$, which is a contradiction. Therefore, $T \leq T_0 = 64C \log^2(64eC)$. □