Mind the Gap: Safely Bridging Offline and Online Reinforcement Learning

Wanqiao Xu
University of Michigan
wanqiaox@umich.edu

Kan Xu
University of Pennsylvania
kanxu@sas.upenn.edu

Hamsa Bastani
University of Pennsylvania
hamsab@wharton.upenn.edu

Osbert Bastani
University of Pennsylvania
obastani@seas.upenn.edu

Abstract

A key challenge to deploying reinforcement learning in practice is exploring safely. We propose a natural safety property—*uniformly* outperforming a conservative policy (adaptively estimated from all data observed thus far), up to a per-episode exploration budget. This property formalizes the idea that we should spread out exploration to avoid taking actions significantly worse than the ones that are currently known to be good. We then design an algorithm that uses a UCB reinforcement learning policy for exploration, but overrides it as needed to ensure safety with high probability. To ensure exploration across the entire state space, it adaptively determines when to explore (at different points of time across different episodes) in a way that allows "stitching" sub-episodes together to obtain a *meta-episode* that is equivalent to using UCB for the entire episode. Then, we establish reasonable assumptions about the underlying MDP under which our algorithm is guaranteed to achieve sublinear regret while ensuring safety; under these assumptions, the cost of imposing safety is only a constant factor.

1 Introduction

Reinforcement learning is a promising approach to learn policies for sequential decision-making to enable data-driven decision-making. For instance, it can be used to help manage health conditions such as sepsis [1] and chronic illnesses [2], which require the clinician to make sequences of decisions regarding treatment. Other applications include adaptively sequencing educational material for students [3] or learning inventory control policies with uncertain demand [4, 5].

The core challenge in reinforcement learning is how to balance the exploration-exploitation tradeoff—i.e., how to balance taking exploratory actions (to estimate the transitions and rewards of the underlying system) and exploiting the knowledge acquired thus far (to make good decisions). However, in high-stakes settings, exploration can be costly or even unethical—for instance, taking exploratory actions on patients or students can lead to adverse outcomes that could have been avoided.

As a consequence, there has been great interest in safe reinforcement learning [6], where the algorithm explores in a way that satisfies some kind of safety property. For instance, these constraints can take the form of restricting the system to a safe region of the state space [7]. However, many desirable safety properties cannot be formulated this way. For instance, we may want to ensure that a patient does not suffer avoidable adverse outcomes, but some adverse outcomes may be unavoidable even under the optimal policy. Instead, we consider a safety property based on the following intuition:

Preprint. Under review.

With high probability, the algorithm should never take actions significantly worse than the ones known to be good based on the knowledge accumulated so far.

For instance, if the algorithm has discovered that a treatment achieves good outcomes for the current patient, then it is obligated to use either that treatment or an alternative that is only slightly worse.

To formalize this property, we need to devise notions of "actions known to be good" and "knowledge accumulated so far". We take "knowledge accumulated so far" to simply be all observations that have been gathered so far. Formalizing "actions known to be good" is more challenging. For instance, we could consider estimating the value of all actions based on the current dataset, and take a "good action" to be one with high value. However, this definition does not account for estimation error. *Offline* (or *batch*) reinforcement learning algorithms [8, 9] are designed to provide conservative estimates of the values of different actions based on historical data [10, 11]. Building on these ideas, we formalize a "good action" to be one for which the conservative lower bound on its value is high. We refer to the policy that always takes the action with the best conservative value estimate as the *baseline policy*. Then, our safety property is that, with high probability, the algorithm never takes an action that is significantly worse than using the current baseline policy.

This safety property is challenging to satisfy for two reasons. First, it requires the learning algorithm to *uniformly* outperform the current baseline (up to a given per-episode tolerance η) with high probability. In particular, the algorithm cannot outperform the baseline for some time and then subsequently "spend" this improvement on exploratory actions; instead, we want to "spread out" our potentially harmful exploration across episodes. Second, the baseline is always improving as more data is collected, so the exploration becomes successively more constrained.

Our goal is to establish assumptions under which we can provably learn (i.e., achieve sublinear regret) while satisfying safety. We consider the classic setting of a Markov decision process (MDP) with finite state and action spaces; in this setting, we can straightforwardly convert existing algorithms for computing optimistic value estimates [12, 13] into ones for computing conservative value estimates. When unconstrained exploration is allowed, algorithms such as upper confidence bound (UCB) [12, 13] and posterior sampling [14] can learn the MDP parameters (i.e., the transitions and rewards) while achieving strong regret guarantees. We aim to adapt these algorithms to satisfy safety without significantly hurting learning.

In general, we need to make assumptions about the underlying system to guarantee safety; otherwise, taking any exploratory action could lead to a safety violation. First, we assume that the MDP is ergodic—i.e., every state is reached with some probability during an episode. This assumption is standard [15]; it is necessary since we need to be able to safely transition to each state using the baseline policy to ensure that we can learn the MDP parameters at every state. This assumption is reasonable in practice as long as there is sufficient randomness in the MDP transitions. Second, we assume that any *single* exploratory action cannot violate the safety constraint. This assumption is also necessary; otherwise, we would be unable to take any exploratory action since doing so would risk violating safety. Instead, the safety constraint can only be violated by sequences of suboptimal actions. Intuitively, this assumption is reasonable in settings where the time steps are sufficiently small, since delaying an action by a small amount should not significantly affect outcomes. For instance, short medication delays for ICU patients often do not affect readmissions/mortality outcomes [16].

Then, to satisfy the safety constraint, our algorithm uses a *shielding* strategy [7, 17], where it uses a classical UCB policy if it does not risk exhausting its *exploration budget* (i.e., the allowable deficit in performance compared to the baseline policy); otherwise, it switches to using the baseline policy, which is guaranteed to satisfy the safety constraint.

However, naïvely, this strategy can lead to suboptimal exploration (e.g., it only explores state-action pairs at the very beginning of an episode, and then exhausts its exploration budget), potentially failing to learn and causing linear regret. In contrast, our algorithm does not always explore starting at the beginning of an episode. Instead, it adaptively decides at which point to start exploring via UCB (using the baseline policy before this point), to enable construction of *meta-episodes* that "stitch" together consecutive exploration steps to obtain a sequence of state-action pairs that are effectively sampled from the same distribution as a full episode using UCB; note that we cannot use UCB for a full episode since it likely violates the safety constraint. Thus, this strategy explores in a way that is identical to UCB, thereby ensuring sufficient learning for strong regret guarantees.

We prove that our algorithm not only ensures safety, but also enjoys regret guarantees similar to those of existing reinforcement learning algorithms for finite-state MDPs. In other words, for MDPs that satisfy our assumptions, the cost of our safety constraint is only a constant factor.

Related literature. There has been a great deal of recent interest in safe reinforcement learning [6], although it has largely focused on guaranteeing safety rather than proving regret bounds (that guarantee convergence to an optimal policy). Furthermore, most of these approaches focus on safety constraints in the form of safe regions, where the goal is to stay inside the safe region. Such constraints are common in robotics, but less so for other applications of reinforcement learning such as healthcare [1], education [3], and operations research [4, 5].

The most closely related work to ours is on *conservative* learning [15, 18], where the goal is to maintain cumulative regret less than that of a baseline policy up to some exploration budget. There are two key differences. First, their baseline policy is provided exogenously by the user, and is static; in contrast, our baseline is updated as new information becomes available. Thus, our safety constraint grows stronger over time. Second, their guarantee is only valid *on average* up to the current time step; thus, it suffices to follow the baseline policy for enough time (accumulating similar average performance) and then "spend" this improvement by exploring arbitrarily (i.e., using UCB for an entire episode). For instance, they may perform safely on many patients and excessive explore on a few subsequent patients, leading to particularly adverse outcomes for these patients. In contrast, we require safety *uniformly* on every time step of every episode, thereby spreading out exploration across the time horizon and ensuring that exploration does not significantly harm any single individual. Importantly, spreading exploration across different episodes can lead to biased exploration, which is not an issue for [15]; we address this issue by constructing meta-episodes that mimic UCB episodes.

2 Problem Formulation

Preliminaries. Consider a Markov decision process (MDP) M, with finite states $s \in S$, finite actions $a \in A$, transitions $P(s' \mid s, a) \in \mathbb{I} = [0, 1]$, rewards $R(s, a) \in \mathbb{I}$, and time horizon $H \in \mathbb{N}$. We consider policies $a = \pi_t(s, z)$ with internal state $z \in Z$, along with internal state transitions $z' = \sigma_t(s, z, a)$. Our safety property (described below) is a constraint on the reward accrued by our policy across multiple steps in the MDP; thus, our policy maintains an internal state to track this information and ensure that we satisfy the safety constraint.

Given (π, σ), a *rollout* is a random sequence $\alpha = ((s_1, z_1, a_1, r_1), ..., (s_H, z_H, a_H, r_H))$ such that $a_t = \pi_t(s_t, z_t)$, $r_t = R(s_t, a_t)$, $s_{t+1} \sim P(\cdot \mid s_t, a_t)$, and $z_{t+1} = \sigma_t(s_t, z_t, t)$; we assume s_1 is deterministic and z_1 is given. We denote the distribution over rollouts by $\alpha \sim D_{\pi,\sigma}(\cdot)$. We define the Q and value functions by initializing $Q_H^{(\pi)}(s, z, a) = 0$, $V_H^{(\pi)}(s, z) = 0$, and recursively defining

$$Q_t^{(\pi,\sigma)}(s, z, a) = R(s, a) + \sum_{s' \in S} P(s' \mid s, a) \cdot V_t^{(\pi,\sigma)}(s, \sigma_t(s, z, a)),$$

$$V_t^{(\pi,\sigma)}(s, z) = Q_t^{(\pi,\sigma)}(s, z, \pi_t(s)).$$

Regret. We let $\pi_t^*(s)$ denote the (deterministic) optimal policy, and $Q_t^*(s, a)$ and $V_t^*(s)$ its Q and value functions, respectively. We consider the episodic reinforcement learning setting, where P and R are initially unknown, and over a sequence of $k \in [N] = \{1, ..., N\}$ episodes, we choose a policy (π^k, σ^k) along with an initial internal state $z_{k,1}$ based on the observations so far, and observe a rollout $\alpha_k \sim D_{\pi^k,\sigma^k}(\cdot)$. Our goal is to choose (π^k, σ^k) and $z_{k,1}$ to minimize the *cumulative regret*

$$\rho = \mathbb{E}\left[\sum_{k=1}^{N} V_1^*(s_1) - V_1^{(\pi^k,\sigma^k)}(s_1, z_{k,1})\right],$$

where the expectation is taken over the random sequence of rollouts $\alpha_1, ..., \alpha_H$.

Safety property. Intuitively, our safety property says that we do not take any sequences of actions across an episode that achieve significantly worse rewards compared to the current baseline policy $\bar{\pi}^k$ (for simplicity, we assume that this policy is only updated at the end of an episode). Thus, it ensures that our algorithm does not take unsafe sequences of actions that could have been avoided by $\bar{\pi}^k$.

The baseline policy $\bar{\pi}^k$ is constructed using all data collected before the current episode. It should be constructed based on lower bounds on the value function that account for estimation error. The

strength of the safety property depends on $\bar{\pi}^k$; thus, these bounds should be as tight as possible to achieve the strongest safety property. We build on a UCB strategy called UCBVI [13], a state-of-the-art algorithm that achieves minimax regret guarantees. This algorithm constructs policies based on values that are optimistic compared to the true values; its minimax guarantees stem from the fact that its confidence intervals around its value estimates are very tight. We modify UCBVI to instead construct policies based on conservative values. We describe our approach in detail in Section 3.

Now, given $\eta, \delta \in \mathbb{R}_{>0}$, our safety property is to ensure that with probability at least $1-\delta$ (over the randomness of the rollouts $\alpha_1, ..., \alpha_N$), for every $k \in [N]$ and $t \in [H]$, we have

$$z_t^* \coloneqq \sum_{\tau=1}^{t} \max\left\{V_\tau^{(\bar{\pi}^k)}(s_{k,\tau}) - Q_\tau^{(\bar{\pi}^k)}(s_{k,\tau}, a_{k,\tau}), 0\right\} \leq \eta. \tag{1}$$

We call z_t^* the *reward deficit*, since it is the deficit in reward compared to $\bar{\pi}^k$, and η the *exploration budget*, since it bounds how much exploration we can do. To understand (1), consider the alternative

$$V_t^{(\bar{\pi}^k)}(s_{k,1}) - V^{(\pi^k,\sigma^k)}(s_{k,1}, z_{k,1}) = \mathbb{E}\left[\sum_{t=1}^{H} V_t^{(\bar{\pi}^k)}(s_{k,t}) - Q_t^{(\bar{\pi}^k)}(s_{k,t}, a_{k,t})\right] \leq \eta, \tag{2}$$

where the equality follows by a telescoping sum argument (see, e.g., Lemma 2.1 in [19]). Intuitively, (2) says that our cumulative expected reward must be within η of that of $\bar{\pi}^k$ across the entire episode. In contrast, (1) is significantly stronger, since taking the maximum ensures that we cannot compensate for performing worse than $\bar{\pi}^k$ in one part of a episode by performing better than $\bar{\pi}^k$ elsewhere.

Note that our algorithm can always use $\bar{\pi}^k$, which satisfies (1); the challenge is how to take exploratory actions in a way that minimizes regret while maintaining safety.

Assumptions. Ensuring safety is impossible without additional assumptions, since otherwise any exploration the agent undertakes could lead to a safety violation. We make two key assumptions. The first one is standard, saying that the MDP is ergodic:

Assumption 2.1. We have $\Upsilon = \max_{s' \neq s} \max_{\pi \in \Pi} \mathbb{E}[T^\pi(s', s)] < \infty$, where Π is the set of all deterministic policies $a_t = \pi_t(s_t)$.

Here, Υ is the worst-case diameter of the MDP—i.e., the worst-case time it takes for any policy π to reach any state s from any state s'. This assumption says that every state is visited by any policy π; for instance, if there is a state not visited by one of our baseline policies $\bar{\pi}^k$, then we would not be able to explore that state, potentially leading to linear regret. Our second assumption says that any *single* step of exploration in the MDP does not violate our safety property:

Assumption 2.2. For any $\pi \in \Pi$, $s \in S$ and $a \in A$, we have $V_t^{(\pi)}(s) - Q_t^{(\pi)}(s, a) \leq \eta/2$.

That is, using an arbitrary action a in state s and then switching to π (i.e., $Q_t^{(\pi)}(s, a)$), is not much worse than using π (i.e., $V_t^{(\pi)}(s)$). Note that at the very least, assuming $V_t^{(\bar{\pi}^k)}(s) - Q_t^{(\bar{\pi}^k)}(s, a) \leq \eta$ is necessary; otherwise, any exploratory action could potentially violate safety. Our algorithm only requires this assumption for the baseline policies $\bar{\pi}^k$ that occur during learning. The stricter $\eta/2$ tolerance enables us to continue to take exploratory steps if we have only accrued error $\leq \eta/2$ so far. In particular, if the tolerance were η, then if we take a single step such that $V_t^{(\bar{\pi}^k)}(s) - Q_t^{(\bar{\pi}^k)}(s, a) > 0$, then at each subsequent step $t' > t$, we cannot take an exploratory action, since we run the risk that $(V_t^{(\bar{\pi}^k)}(s) - Q_t^{(\bar{\pi}^k)}(s, a)) + (V_{t'}^{(\bar{\pi}^k)}(s) - Q_{t'}^{(\bar{\pi}^k)}(s, a)) > \eta$, which would violate safety.

3 Algorithm

The key challenge is how to take exploratory actions to minimize regret while ensuring that our safety property holds. We build on the upper confidence bound value iteration (UCBVI) algorithm by [12], which obtains near-optimal regret guarantees for finite-horizon MDPs. Like other UCB algorithms, it relies on *optimism* to minimize regret—i.e., it takes actions that optimize the cumulative reward under optimistic assumptions about its estimates of the MDP parameters. A natural strategy is to use the internal state to keep track of the reward deficit accrued so far; then, we can use the UCBVI policy from the beginning of each episode until we exhaust our exploration budget, after which we switch to the baseline policy.

However, the challenge employing this strategy is that the UCBVI regret guarantees depend crucially on using the UCBVI policy for the entire horizon, or at least for extended periods of time. The reason is that selectively using UCBVI at the beginning of each episode biases the portions of the state space where UCBVI is used; for instance, if there are some states that are only reached late in the episode, then we may never use UCBVI in these states, causing us to underexplore and accrue high regret.

To avoid this issue, our algorithm uses the UCBVI policy in portions of each episode in a sequence of episodes, such that we can "stitch" these portions together to form a single *meta-episode* that is mathematically equivalent to using the UCBVI policy for an entire episode. The cost is that we may require multiple episodes to obtain a single UCBVI episode, which would slow down exploration and increase regret. However, we can show that the number of episodes in a meta-episode is not too large with high probability, so the strategy actually achieves similar regret as UCBVI.

Overall algorithm. Our algorithm is summarized in Algorithm 1. Here, m indexes a single meta-episode, and n indexes an episode of m. To be precise, we use *meta-episode* to an iteration m of the outer loop of Algorithm 1, and *episode* to refer to an iteration (m, n) of the inner loop; we alternatively index episodes by k when referring to the sequence of all episodes. Then, we use *rollout* to refer to the sequence $\alpha_{m,n}$ of observations (s, a, r, s') during an episode, and a *meta-rollout* to refer to the rollout $\hat{\alpha}_m$ consisting of a subset of the observations in $\{\alpha_{m,1}, ..., \alpha_{m,N_m}\}$, where N_m is the total number of episodes in meta-episode m. In particular, $\hat{\alpha}_m$ consists of observations (s, a, r, s') where the UCB policy $\hat{\pi}$ was used; our algorithm uses $\hat{\pi}$ in a way that ensures that $\hat{\alpha}_m$ is equivalent to a single rollout sampled from the MDP while exclusively using $\hat{\pi}$.

At a high level, at the beginning of each episode k, our algorithm constructs the baseline policy $\bar{\pi}$ using the current rollouts $A = \{\alpha_1, ..., \alpha_{k-1}\}$. Furthermore, at the beginning of each meta-episode m, our algorithm constructs the UCBVI policy $\hat{\pi}$ using the current meta-rollouts $\hat{A} = \{\hat{\alpha}_1, ..., \hat{\alpha}_{m-1}\}$. Then, it obtains a sequence of rollouts using $\tilde{\pi}$, which combines the current $\bar{\pi}$ and $\hat{\pi}$ in a way that ensures safety. It does so in a way that it can "stitch" together portions of the rollouts using $\hat{\pi}$ into a single rollout $\hat{\alpha}_m$ whose distribution equals the distribution over rollouts induced by using $\hat{\pi}$. In other words, $\hat{\alpha}_m$ is equivalent to using $\hat{\pi}$ for a single episode. Thus, each meta-rollout of our algorithm corresponds to a single UCBVI episode. As long as the number of episodes per meta-episode is not too large, we obtain similar regret as UCBVI. We describe our algorithm in more detail below.

Safety. First, we describe how our algorithm ensures safety. Our strategy is to use internal state to keep track of reward deficit. In particular, suppose we have $\hat{V}_t^{(\bar{\pi})}$ satisfying $\hat{V}_t^{(\bar{\pi})}(s) \geq V_t^{(\bar{\pi})}(s)$ and $\bar{Q}_t^{(\bar{\pi})}$ satisfying $\bar{Q}_t^{(\bar{\pi})}(s, a) \leq Q_t^{(\bar{\pi})}(s, a)$ with high probability; then, we use internal state $z_1 = 0$ and

$$\sigma_t(s, z, a) = z + \max\{\hat{V}_t^{(\bar{\pi})}(s) - \bar{Q}_t^{(\bar{\pi})}(s, a), 0\} = z + \hat{V}_t^{(\bar{\pi})}(s) - \bar{Q}_t^{(\bar{\pi})}(s, a),$$

where the second equality follows since we always have $\hat{V}_t^{(\bar{\pi})}(s) \geq \bar{Q}_t^{(\bar{\pi})}(s, a)$. In particular, we have $z_t \geq z_t^*$ with high probability. Then, our algorithm switches to using $\bar{\pi}$ as soon as $z_t > \eta/2$ (i.e., $z_{t-1} \leq \eta/2$)—i.e., it uses the *shield policy*

$$\tilde{\pi}_t(s, z) = \begin{cases} \hat{\pi}_t(s) & \text{if } z_t \leq \eta/2 \\ \bar{\pi}_t(s) & \text{otherwise,} \end{cases}$$

where $\hat{\pi}$ is the current UCBVI policy. As a consequence, we have

$$z_t^* \leq z_t \leq z_{t-1} + \eta/2 \leq \eta,$$

where the second inequality follows by Assumption 2.2. Since using $\bar{\pi}$ does not increase the reward deficit, we have $z_H^* \leq \eta$, so (1) holds—i.e., using $\tilde{\pi}$ ensures safety with high probability.

Meta-episodes. As defined, $\tilde{\pi}$ implements the naïve strategy of using $\hat{\pi}$ at the beginning of each episode, and switching to $\bar{\pi}$ if it can no longer ensure safety. However, as discussed above, this strategy may explore in a biased way, causing it to accrue high regret. Instead, we modify $\tilde{\pi}$ to construct a single UCBVI episode (called a *meta-episode*) across multiple actual episodes, which ensures exploration equivalent to UCBVI. We denote such a meta-episode by $m \in [M]$ and an episode in meta-epsiode m by $n \in [N_m]$ (i.e., there are N_m episodes in m, so we have $N = \sum_{m=1}^{M} N_m$ total episodes); we index our episodes by (m, n) instead of k.

At a high level, in the first episode of a meta-episode m (i.e., $n = 1$), we use $\hat{\pi}$ from the beginning. If $\tilde{\pi}$ uses $\hat{\pi}$ for the entire episode, then this single episode is equivalent to a UCBVI episode, so we are

Algorithm 1 Safe offline-to-online UCBVI.

procedure SAFEUCBVI(M, N, δ)
  Initialize rollout history $A \leftarrow \varnothing$ and meta-rollout history $\hat{A} \leftarrow \varnothing$
  for $m \in \mathbb{N}$ **do**
    Compute $\hat{\pi}$ using $\hat{A}$
    Initialize target state $s' \leftarrow s_1$
    for $n \in \mathbb{N}$ **do**
      Compute $\bar{\pi}$ using A, along with $\hat{V}^{(\bar{\pi})}$ and $\bar{Q}^{(\bar{\pi})}$
      Obtain a rollout $\alpha_{m,n}$ using $z_1 = (s', 0)$, σ as in (4), and $\tilde{\pi}$ as in (5), and add it to A
      Update s' to be the next target state, or break if done (and terminate if $|A| \geq N$)
    end for
    Construct $\hat{\alpha}_m$ from $\alpha_{m,1}, ..., \alpha_{m,N_m}$ and add it to $\hat{A}$
  end for
end procedure

done. Otherwise, we switch to using $\bar{\pi}$ at some step t (i.e., at state $s_{m,1,t}$). Then, in the subsequent episode, we initially use $\bar{\pi}$ until some step t' such that $s_{m,2,t'} = s_{m,1,t}$; at this point, we switch to $\hat{\pi}$ until we have exhausted our exploration budget. If we do not encounter $s_{m,1,t}$, then we try again in the next episode; since the MDP is ergodic, we are guaranteed to find $s_{m,1,t}$ after a few tries with high probability. We continue this process until we have used $\hat{\pi}$ for H steps (i.e., a full UCB episode).

Formally, we augment the internal state of our policy with the target state s form which we want to continue using $\hat{\pi}^m$ (or s_1 for the initial episode), so $z = (s', \zeta) \in S \times \mathbb{R}$. In particular, we let

$$z_{m,n,1} = \begin{cases} (s_1, 0) & \text{if } n = 1 \\ (s'_{m,n}, 0) & \text{otherwise,} \end{cases} \tag{3}$$

where $s'_{m,n}$ is the target state for episode n—i.e., the state $s_{m,n',t}$ at which we switched to $\bar{\pi}$ for some $n' < n$, such that we did not encounter $s_{m,n',t}$ in episodes $n' < n'' < n$. Next, we have

$$\sigma_t\Big(s, (s', \zeta), a\Big) = \begin{cases} (s', 0) & \text{if } s' \neq \varnothing \text{ and } s' \neq s \\ \Big(\varnothing, \zeta + \hat{V}_t^{(\bar{\pi})}(s) - \bar{Q}_t^{(\bar{\pi})}(s, a)\Big) & \text{otherwise.} \end{cases} \tag{4}$$

In other words, the internal state remains $z = (s', 0)$ until encountering the target state s'; at this point, it becomes $(\varnothing, 0)$ and starts accruing reward deficit as before. Finally, we have shield policy

$$\tilde{\pi}_t\Big(s, (s', \zeta)\Big) = \begin{cases} \hat{\pi}_t(s) & \text{if } s' = \varnothing \text{ and } \zeta \leq \eta/2 \\ \bar{\pi}_t(s) & \text{otherwise.} \end{cases} \tag{5}$$

In other words, we use the UCBVI policy $\hat{\pi}$ if we have reached the target state s' and do not risk exceeding our exploration budget; otherwise, we use the backup policy $\bar{\pi}$.

Finally, a meta-episode terminates once we have used $\hat{\pi}$ at least H times across the rollouts $\alpha_{m,1}, ..., \alpha_{m,n}$; in this case, we have $n = N_m$ episodes in meta-episode m. Then, our algorithm constructs the corresponding *meta-rollout* $\hat{\alpha}_m$ by concatenating the portions of $\alpha_{m,1}, ..., \alpha_{m,n}$ that use $\hat{\pi}$. Note that in the very last episode $\alpha_{m,n}$, we may continue using $\hat{\pi}$ even after we have obtained the necessary H steps using $\hat{\pi}$; we ignore the extra steps so $\hat{\alpha}_m$ is exactly H steps long.

Policy construction. Finally, we describe how our algorithm constructs the quantities $\bar{Q}^{(\bar{\pi})}$, $\hat{V}_t^{(\bar{\pi})}(s)$, $\bar{\pi}$, and $\hat{\pi}$. The constructions are based on the UCBVI algorithm; in particular, note that on step m, $\hat{A}$ is equivalent to a set of $m - 1$ UCBVI rollouts, so we can use it to construct a UCBVI policy $\hat{\pi}$ for the mth episode.[1] In particular, we construct $\hat{\pi}$ by estimating the transitions and rewards based on the data collected so far (i.e., the tuples (s, a, r, s') collected on steps using the UCBVI policy, so $a = \hat{\pi}(s)$, $r = R(s, a)$, and $s' \sim P(\cdot \mid s, a)$), to obtain

$$\hat{P}(s' \mid s, a) = \frac{|\{(s, a, s', \cdot, \cdot) \in A\}|}{N(s, a)} \qquad \text{and} \qquad \hat{R}(s, a) = \frac{\sum_{(s,a,\cdot,r)\in A} r}{N(s, a)}$$

[1] By only using meta-episodes to construct $\hat{\pi}$, we ensure that the meta-episodes exactly mimic the execution of UCBVI; in practice, we can use the entire dataset A to construct $\hat{\pi}$.

where $N(s,a) = |\{(s,a,\cdot,\cdot) \in A\}|$ is the number of observations of state-action pair (s,a) in the data collected so far. Then, we use value iteration to solve the Bellman equations

$$\hat{Q}_t^*(s,a) = \hat{R}'(s,a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s,a) \cdot \hat{V}_{t+1}^*(s') \quad \text{and} \quad \hat{V}_t^*(s) = \max_{a \in A} \hat{Q}_t^*(s,a),$$

where $\hat{R}'(s,a) = \hat{R}(s,a) + b(s,a;N(s,a))$, where $b(s,a;N) = 4H\sqrt{SL/\max\{1,N\}}$ is a bonus term, and where $L = \log(5SAH\sum_{m=1}^{M} N_m/\delta)$. Finally, we take $\hat{\pi}_t(s) = \arg\max_{a \in A} \hat{Q}^*(s,a)$.

We construct $\bar{Q}^{(\bar{\pi})}$ and $\hat{V}^{(\bar{\pi})}$ similarly. First, we construct $\bar{Q}^{(\bar{\pi})}$ by using the above strategy except we subtract the bonus—i.e., letting $\bar{R}'(s,a) = \hat{R}(s,a) - b(s,a;N(s,a))$, we have

$$\bar{Q}_t^*(s,a) = \bar{R}'(s,a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s,a) \cdot \bar{V}_{t+1}^*(s') \quad \text{and} \quad \bar{V}_t^*(s) = \max_{a \in A} \bar{Q}_t^*(s,a).$$

Then, we take $\bar{\pi}_t(s) = \arg\max_{a \in A} \bar{Q}_t^*(s,a)$. Finally, we construct $\hat{V}^{(\bar{\pi})}$ by adding the bonus, but using value iteration for policy evaluation instead of policy optimization—i.e.,

$$\hat{Q}_t^{(\bar{\pi})}(s,a) = \hat{R}'(s,a) + \gamma \cdot \sum_{s' \in S} P(s' \mid s,a) \cdot \hat{V}_{t+1}^{(\bar{\pi})}(s') \quad \text{and} \quad \hat{V}_t^{(\bar{\pi})}(s) = \hat{Q}_t^{(\bar{\pi})}(s, \bar{\pi}(s)).$$

4 Theoretical Guarantees

All our results in this section are conditioned on a high-probability event $\mathcal{E}$ which says that (i) our confidence sets around the estimated transitions $\hat{P}$ and rewards $\hat{R}$ hold, and (ii) a condition saying that we find the target state s' in a reasonable number of episodes (see Lemma 4.7). This event holds with probability at least $1-\delta$; we give details in Appendix A.1.

Safety guarantee. First, we prove that our algorithm satisfies our safety constraint.

Theorem 4.1. *On event $\mathcal{E}$, Algorithm 1 satisfies (1) for all $k \in [N]$*

Proof. First, we show that $z_t \le \eta$ for all $t \in [H]$. To this end, consider following cases at step t: (i) if $\tilde{\pi}$ uses $\hat{\pi}$, then $z_t \le \eta/2$, (ii) if $\tilde{\pi}$ switches to $\bar{\pi}$ on step t, then $z_t \le z_{t-1} + \eta/2 \le \eta$, and (iii) otherwise, $z_t = z_{t-1}$ remains the same, so the claim follows by induction. As a consequence, it suffices to show that $z_t \ge z_t^*$ on event $\mathcal{E}$. To this end, the following lemma says that the high probability upper and lower bounds $\hat{V}^{(\bar{\pi})}$ and $Q_t^{(\bar{\pi})}(s,a)$ used to construct z_t are correct.

Lemma 4.2. *On event $\mathcal{E}$, for all $s \in S$, $a \in A$, $k \in [N]$, and $t \in [H]$, we have (i) $\bar{Q}_{k,t}^{(\pi)}(s,a) \le Q_t^{(\pi)}(s,a)$, and (ii) $\hat{V}_{k,t}^{(\pi)}(s) \ge V_t^{(\pi)}(s)$.*

This result is based on standard arguments; we give a proof in Appendix A.2. Now, by Lemma 4.2,

$$z_t \ge \sum_{\tau=1}^{t} \max\left\{ V_\tau^{(\bar{\pi}^k)}(s_{k,\tau}) - Q_\tau^{(\bar{\pi}^k)}(s_{k,\tau}, a_{k,\tau}), 0 \right\} = z_t^*$$

on event $\mathcal{E}$, so the claim holds. □

Regret bound. Next, we prove that Algorithm 1 has sublinear regret.

Theorem 4.3. *On event $\mathcal{E}$, the cumulative expected regret of Algorithm 1 is*

$$\rho \le 20HL\sqrt{SAHN} + 250H^2S^2AL^2 + \frac{960H^3S}{\eta}\sqrt{2ALHN},$$

where $L = \log(5SAH^2MN)$, and where the expectation is taken over the randomness during all of the rollouts taken by the algorithm. Furthermore, letting $T = H\sum_{m=1}^{M} N_m = HN$ be the total number of time-steps by the end of meta-episode M, the regret satisfies $\rho = \tilde{O}(H^3\sqrt{SAT})$.

Proof. The main idea is to bound the regret by the regret of the meta-rollouts (which correspond to UCBVI rollouts), plus the regret of the shield policy $\tilde{\pi}$ on the remaining steps—i.e., $\rho = \hat{\rho} + \bar{\rho}$, where

$$\hat{\rho} = \mathbb{E}\left[\sum_{m=1}^{M} V_1^*(s_1) - V_1^{(\hat{\pi}^m)}(s_1)\right] \qquad \bar{\rho} = \mathbb{E}\left[\sum_{m=1}^{M}\sum_{n=1}^{N_m}(V_1^*(s_1) - V_1^{(\bar{\pi}^{m,n})}(s_1))\mathbb{1}((n,t) \notin \hat{\alpha}_m)\right],$$

where we have used $(n,t) \notin \hat{\alpha}_m$ to denote that the tth step $(s_{m,n,t}, a_{m,n,t})$ of episode n is not included in meta-rollout $\hat{\alpha}_m$. By equivalence to UCBVI, $\hat{\rho}$ is bounded by the UCBVI regret:

Lemma 4.4. *On event $\mathcal{E}$, we have $\hat{\rho} \le 20HL\sqrt{SAHN} + 250H^2S^2AL^2$.*

The proof is based directly on the UCBVI regret analysis; for completeness, we give a proof in Appendix A.3. Thus, we focus on bounding $\bar{\rho}$. First, we have the straightforward bound,

$$\bar{\rho} \le \mathbb{E}\left[\sum_{m=1}^{M} H(N_m - 1)\right], \tag{6}$$

which follows since the maximum regret during a single episode is H (since the rewards are bounded by 1), and since we can also omit the steps for which $(n,t) \in \hat{\alpha}_m$, which which there are exactly H.

As a consequence, the key challenge in bounding $\bar{\rho}$ is proving that the number of episodes N_m in a meta-episode becomes small—in particular, once $N_m = 1$, then the entire (single) rollout $\alpha_{m,1}$ is part of the meta-rollout $\hat{\alpha}_m$, so the second term in the regret is zero.

To prove that N_m becomes small, we note that for any episode, one of the following conditions must hold: (i) the exploration budget is exhausted—i.e., $z_H^* \ge \eta/2$, (ii) the algorithm explores using $\hat{\pi}$ for at least $H/4$ time steps, or (iii) the episode does not reach the target state s' in the first $3H/4$ time steps; in particular, if (iii) does not hold, then either the episode uses $\hat{\pi}$ for the final $H/4$ steps of that episode (so (ii) holds) or the exploration budget is exhausted (so (i) holds). We let N_m^1, N_m^2, N_m^3 denote the number of episodes that satisfy the three respective cases in meta-episode m; note that either $N_m = 1$ (i.e., always use the UCBVI policy) or $N_m = N_m^1 + N_m^2 + N_m^3$.

We bound the three possibilities separately. Our first lemma shows that the number of episodes N_m^1 in case (i) is bounded by the UCBVI regret (i.e., the regret of the meta-episode), which is sublinear.

Lemma 4.5. *On event $\mathcal{E}$, we have*

$$N_m^1 \le \frac{2}{\eta}\sum_{t=1}^{H} \hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})).$$

where $L = \log(5SAH\sum_{m=1}^{M} N_m/\delta)$, and $N_m(s,a)$ is the total number of observations of the state-action pair (s,a) prior to meta-episode m.

Proof. First, we sum the condition $z_{m,n,H}^* \ge \eta/2$ over episode $n \in [N_m]$, which gives

$$N_m^1 \cdot \frac{\eta}{2} \le \sum_{n=1}^{N_m} z_{m,n,H}^* = \sum_{n=1}^{N_m}\sum_{t=1}^{H} \max\left\{V_t^{(\bar{\pi}^{m,n})}(s_{m,n,t}) - Q_t^{(\bar{\pi}^{m,n})}(s_{m,n,t}, \hat{\pi}(s_{m,n,t})), 0\right\}.$$

Now, note that $V_t^{(\bar{\pi}^{m,n})}(s_{m,n,t}) \ne Q_t^{(\bar{\pi}^{m,n})}(s_{m,n,t}, a_{m,n,t})$ only when $a_{m,n,t} \ne \bar{\pi}^{m,n}(s_{m,n,t})$—i.e., when $(s_{m,n,t}, a_{m,n,t})$ is part of the meta-rollout $\hat{\alpha}_m$. Thus, we can restrict the sum to steps in $\hat{\alpha}_m$:

$$\begin{aligned}
N_m^1 \cdot \frac{\eta}{2} &\le \sum_{t=1}^{H} \max\left\{V_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}) - Q_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})), 0\right\} \\
&\le \sum_{t=1}^{H} \hat{V}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \\
&\le \sum_{t=1}^{H} \hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})).
\end{aligned}$$

Here, the second line follows since by Lemma 4.2, $\hat{V}^{(\bar{\pi})}$ is an upper bound and $\bar{Q}^{(\bar{\pi})}$ is a lower bound on event $\mathcal{E}$, and since $\hat{V}_t^{(\bar{\pi})}(s) \ge \bar{V}_t^{(\bar{\pi})}(s) \ge \bar{Q}_t^{(\bar{\pi})}(s,a)$ for any a since $\bar{\pi}$ by definition of $\bar{\pi}$. Finally, the third line follows since $\hat{\pi}$ is optimistic. □

The left-hand side of the bound is essentially (but not exactly) the UCBVI regret, and we can bound it using the same strategy. In particular, we have the following result:

Lemma 4.6. *On event $\mathcal{E}$, we have*

$$\sum_{m=1}^{M}\sum_{t=1}^{H}\hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t},\hat{\pi}(\hat{s}_{m,t})) \leq 12H^2S\sqrt{ALHN}.$$

The proof is based on the same strategy as UCBVI, so we defer it to Appendix A.4. Note that we have summed over meta-episodes $m \in [M]$; later, we use Lemma 4.6 to directly bound $\sum_{m=1}^{M} N_m^1$.

Next, N_m^2 is straightforward to bound—in particular, note that we can use $\hat{\pi}$ for $H/4$ time steps in at most four episodes, since at the end of the fourth episode we would have a complete UCBVI episode (which has length H). Thus, we have $N_m^2 \leq 4$. Next, we use the following result to bound N_m^3.

Lemma 4.7. *On event $\mathcal{E}$, for any state $s \in S$, a rollout using $\tilde{\pi}$ will reach state s within $3H/4$ time steps after at most $6\log(1/\delta)$ episodes.*

This result follows applying Markov's inequality in conjunction with Assumption 2.1, which says the MDP M is ergodic; thus, it visits s with high probability early in the rollout. We give a proof in Appendix A.5. Finally, we have the following overall bound:

Lemma 4.8. *On event $\mathcal{E}$, we have $N_m \leq \max\{30N_m^1\log(1/\delta), 1\}$.*

Proof. If $N_m = 1$, then the bound trivially holds. Otherwise, note that we must have $N_m^1 \geq 1$, since if $N_m \neq 1$ then we must have exhausted the exploration budget during the first episode $n = 1$. Next, by Lemma 4.7, we have $N_m^3 \leq (N_m^1 + N_m^2)\max\{6\log(1/\delta) - 1, 0\}$—i.e., it is bounded by the number of "successful episodes $N_m^1 + N_m^2$ times the maximum number of tries $\max\{6\log(1/\delta) - 1, 0\}$ before finding a successful episode. Together with the fact that $N_m^2 \leq 4$, we have

$$\begin{aligned} N_m = N_m^1 + N_m^2 + N_m^3 &\leq N_m^1 + 4 + (N_m^1 + 4)\max\{6\log(1/\delta) - 1, 0\} \\ &\leq 5N_m^1 + 5N_m^1\max\{6\log(1/\delta) - 1, 0\} \\ &\leq 30N_m^1\log(1/\delta), \end{aligned}$$

where on the second line, we have used the fact that we are considering the case $N_m^1 \geq 1$, which implies $N_m^1 + 4 \leq 5N_m^1$. The claim follows. □

In summary, we have

$$\bar{\rho} \leq 30H\log(1/\delta)\mathbb{E}\left[\sum_{m=1}^{M} N_m^1\right] \leq \frac{960H^3S}{\eta}\sqrt{2ALHN}, \tag{7}$$

where the first inequality follows by combining (6) with Lemma 4.8 (which implies $N_m - 1 \leq 30\log(1/\delta)N_m^1$), and the second inequality follows from Lemmas 4.5 & 4.6. Finally, Theorem 4.3 follows immediately by combining (7) and Lemma 4.4. □

5 Conclusion

We propose a novel reinforcement learning algorithm that ensures close performance compared to our current knowledge uniformly across every step of every episode with high probability. We derive assumptions on the MDP under which both safety and sublinear regret can be achieved. Our results show that the price of uniform safety on learning is negligible—i.e., a constant, T-independent factor. One limitation of our approach is that we need to make several assumptions about the MDP; a key direction for future work is relaxing these assumptions. Furthermore, we make specific assumptions about the safety property; extending our techniques to additional properties is another direction for future work. Our work has ethical considerations insofar as we are proposing a way to improve safety of reinforcement learning in practice; before deploying our approach (or any machine learning algorithm) in any domain, it is critical to ensure the algorithm does not harm individuals.

References

[1] Matthieu Komorowski, Leo A Celi, Omar Badawi, Anthony C Gordon, and A Aldo Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature medicine*, 24(11):1716–1720, 2018.

[2] Mo Zhou, Yonatan Mintz, Yoshimi Fukuoka, Ken Goldberg, Elena Flowers, Philip Kaminsky, Alejandro Castillejo, and Anil Aswani. Personalizing mobile fitness apps using reinforcement learning. In *CEUR workshop proceedings*, volume 2068. NIH Public Access, 2018.

[3] Travis Mandel, Yun-En Liu, Sergey Levine, Emma Brunskill, and Zoran Popovic. Offline policy evaluation across representations with applications to educational games. In *AAMAS*, pages 1077–1084, 2014.

[4] Ilaria Giannoccaro and Pierpaolo Pontrandolfo. Inventory management in supply chains: a reinforcement learning approach. *International Journal of Production Economics*, 78(2):153–161, 2002.

[5] Philipp W Keller, Shie Mannor, and Doina Precup. Automatic basis function construction for approximate dynamic programming and reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 449–456, 2006.

[6] Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

[7] Shuo Li and Osbert Bastani. Robust model predictive shielding for safe reinforcement learning with stochastic dynamics. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7166–7172. IEEE, 2020.

[8] Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556, 2005.

[9] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

[10] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *arXiv preprint arXiv:2005.13239*, 2020.

[11] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

[12] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(4), 2010.

[13] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pages 263–272. PMLR, 2017.

[14] Ian Osband and Benjamin Van Roy. Why is posterior sampling better than optimism for reinforcement learning? In *International Conference on Machine Learning*, pages 2701–2710. PMLR, 2017.

[15] Evrard Garcelon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Matteo Pirotta. Conservative exploration in reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 1431–1441. PMLR, 2020.

[16] Lesley Meng, Krzysztof Laudanski, Ann Huffenberger, and Christian Terwiesch. The impact of medication delays on patient health in the icu: Estimating marginal effects under endogenous delays. *Available at SSRN 3590744*, 2020.

[17] Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[18] Yifan Wu, Roshan Shariff, Tor Lattimore, and Csaba Szepesvári. Conservative bandits. In *International Conference on Machine Learning*, pages 1254–1262. PMLR, 2016.

[19] Osbert Bastani, Yewen Pu, and Armando Solar-Lezama. Verifiable reinforcement learning via policy extraction. *arXiv preprint arXiv:1805.08328*, 2018.

A Proofs for Section 4

A.1 High probability event

We first introduce the high probability event $\mathcal{E}$ under which the concentration inequalities described in the policy construction and in UCBVI-CH hold. Let $\mathcal{E}_{1,\delta}$ be the high probability event under which the UCBVI-CH regret analysis holds. This event $\mathcal{E}_{1,\delta}$ is defined in the equation on the bottom of page 16 in the appendices of [13]. The proof that $\mathcal{E}_{1,\delta}$ holds with probability at least $1-\delta$ is proved in the subsequent Lemma 1. We then define

$$c_p(n) \coloneqq 2\sqrt{\frac{SL}{\max\{1,n\}}},$$
$$c_r(n) \coloneqq 2\sqrt{\frac{L}{\max\{1,n\}}}.$$

Let $\mathcal{P}$ denote the set of all probability distributions on the states S, then construct the confidence sets for every $k = 1, \dots, N$ and $(s,a) \in S \times A$

$$B_p^k(s,a) \coloneqq \left\{\tilde{R}(\cdot \mid s,a) : |\tilde{R}(s,a) - R(s,a)| \le c_r(N_k(s,a))\right\},$$
$$B_r^k(s,a) \coloneqq \left\{\tilde{P}(\cdot \mid s,a) \in \mathcal{P} : \left\|\tilde{P}(\cdot \mid s,a) - P(\cdot \mid s,a)\right\|_1 \le c_r(N_k(s,a))\right\}.$$

Next, we define the random event $\mathcal{E}_{2,\delta}$

$$\mathcal{E}_{2,\delta} \coloneqq \bigcap_{(s,a)\in S\times A} \bigcap_{k\in[N]} \left\{\hat{P}_k(\cdot \mid s,a) \in B_p^k(s,a)\right\} \bigcap \left\{\hat{R}_k(s,a) \in B_r^k(s,a)\right\}.$$

where $N_k(s,a)$ is the number of observations of state-action pair (s,a) up to episode k. Finally, letting $\mathcal{E} \coloneqq \mathcal{E}_{1,\delta} \cap \mathcal{E}_{2,\delta}$, we conclude that $\mathcal{E}$ holds with probability at least $1-2\delta$. Indeed,

$$\Pr(\mathcal{E}_{2,\delta}^c) \le \sum_{s,a}\sum_{k} \frac{2\delta}{5NSA} \le \delta$$

and the claim follows from a union bound.

A.2 Proof of Lemma 4.2

Proof. First, we prove claim (i). We show by induction that $\bar{Q}_{k,t}^{(\pi)}$ is indeed a lower bound on $Q_t^{(\pi)}$, the real Q functions. Define the sets

$$\bar{\Omega}_{k,t}^{(\pi)} = \{\bar{Q}_{i,j}^{(\pi)} \le Q_j^{(\pi)}, \forall (i,j), i \in [N], j \in [H], i < k \vee (i = k \wedge j > t)\}$$

We want to show that the set of events $\{\bar{\Omega}_{k,t}^{(\pi)}\}_{k\in[K],t\in[H]}$ hold under the event $\mathcal{E}$.

We proceed by induction. For $t = H$, by definition, $\bar{Q}_{k,H}^{(\pi)} = Q_H^{(\pi)}$, so $\bar{Q}_{k,t}^{(\pi)} \le Q_t^{(\pi)}$ holds. Now, assuming $\bar{Q}_{k,t+1}^{(\pi)} \le Q_{t+1}^{(\pi)}$ holds, we want to show that $\bar{Q}_{k,t}^{(\pi)} \le Q_t^{(\pi)}$ also holds. To this end, note that

$$\begin{aligned}
Q_t^{(\pi)}(s,a) - \bar{Q}_{k,t}^{(\pi)}(s,a) &= b_k(s,a) + P(\cdot \mid s,a)V_{t+1}^{(\pi)} - \hat{P}(\cdot \mid s,a)\bar{V}_{k,t+1}^{(\pi)} + R(s,a) - \hat{R}(s,a) \\
&= b_k(s,\pi(s)) + \left(P(\cdot \mid s,a) - \hat{P}(\cdot \mid s,a)\right)V_{t+1}^{(\pi)}(s) \\
&\quad + \hat{P}(\cdot \mid s,a)\left(V_{t+1}^{(\pi)} - \bar{V}_{k,t+1}^{(\pi)}\right)(s) + R(s,a) - \hat{R}(s,a) \\
&= b_k(s,\pi(s)) + \left(P(\cdot \mid s,a) - \hat{P}(\cdot \mid s,a)\right)V_{t+1}^{(\pi)}(s) \\
&\quad + \hat{P}\left(\cdot \mid s,a)(Q_{t+1}^{(\pi)} - \bar{Q}_{k,t+1}^{(\pi)}\right)(s,\pi(s)) + R(s,a) - \hat{R}(s,a) \\
&\ge b_k(s,\pi(s)) + \left(P(\cdot \mid s,a) - \hat{P}(\cdot \mid s,a)\right)V_{t+1}^{(\pi)}(s) + R(s,a) - \hat{R}(s,a)
\end{aligned}$$

where we use the induction hypothesis in the last inequality. The event $\mathcal{E}$, by Hölder's inequality, implies that

$$\begin{aligned}
&\left|\left(P(\cdot \mid s,a) - \hat{P}(\cdot \mid s,a)\right) V_{t+1}^{(\pi)}(s) + R(s,a) - \hat{R}(s,a)\right| \\
&\leq \|P(\cdot \mid s,a) - \hat{P}(\cdot \mid s,a)\|_1 \|V_{t+1}^{(\pi)}\|_\infty + 2\sqrt{L/\max\{1, N_k(s,\pi(s))\}} \\
&\leq 2H\sqrt{SL/\max\{1, N_k(s,\pi(s))\}} + 2\sqrt{L/\max\{1, N_k(s,\pi(s))\}} \\
&\leq b_k(s,\pi(s))
\end{aligned}$$

as claimed. Next, we prove claim (ii), again by backwards induction. Define the sets

$$\hat{\Omega}_{k,t}^{(\pi)} = \{\hat{V}_{i,j}^{(\pi)} \geq V_j^{(\pi)}, \forall (i,j), i \in [N], j \in [H], i < k \vee (i = k \wedge j > t)\}.$$

We want to show that the set of events $\{\hat{\Omega}_{k,t}^{(\pi)}\}_{k\in[K],t\in[H]}$ hold under the event $\mathcal{E}$. Again we proceed by induction. By definition, $\hat{V}_{k,H}^{(\pi)} = V_H^{(\pi)}$. Assuming $\hat{V}_{k,t+1}^{(\pi)} \geq V_{t+1}^{(\pi)}$ holds, we want to show that $\hat{V}_{k,t}^{(\pi)} \geq V_t^{(\pi)}$ also holds. To this end, note that

$$\begin{aligned}
\hat{V}_{k,t}^{(\pi)}(s) - V_t^{(\pi)}(s) &= b_k(s,\pi(s)) + \hat{P}_k^{(\pi)}\hat{V}_{k,t+1}^{(\pi)}(s) - P^{(\pi)}V_{t+1}^{(\pi)}(s) + \hat{R}(s,\pi(s)) - R(s,\pi(s)) \\
&= b_k(s,\pi(s)) + (\hat{P}_k^{(\pi)} - P^{(\pi)})V_{t+1}^{(\pi)}(s) + \hat{P}_k^{(\pi)}(\hat{V}_{k,t+1}^{(\pi)} - V_{t+1}^{(\pi)})(s) \\
&\quad + \hat{R}(s,\pi(s)) - R(s,\pi(s)) \\
&\geq b_k(s,\pi(s)) + (\hat{P}_k^{(\pi)} - P^{(\pi)})V_{t+1}^{(\pi)}(s) + \hat{R}(s,\pi(s)) - R(s,\pi(s))
\end{aligned}$$

where we use the induction hypothesis in the last inequality. The event $\mathcal{E}$, by Hölder's inequality, implies that

$$\begin{aligned}
&|(\hat{P}_k^{(\pi)} - P^{(\pi)})V_{t+1}^{(\pi)}(s) + \hat{R}(s,\pi(s)) - R(s,\pi(s))| \\
&\leq ||\hat{P}_k^{(\pi)} - P^{(\pi)}||_1 ||V_{t+1}^{(\pi)}||_\infty + 2\sqrt{L/\max\{1, N_k(s,\pi(s))\}} \\
&\leq 2H\sqrt{SL/\max\{1, N_k(s,\pi(s))\}} + 2\sqrt{L/\max\{1, N_k(s,\pi(s))\}} \\
&\leq b_k(s,\pi(s))
\end{aligned}$$

as claimed. □

A.3 Proof of Lemma 4.4

Proof. Note that

$$\begin{aligned}
\hat{\rho} &\coloneqq \mathbb{E}\left[\sum_{m=1}^{M} V_1^*(s_1) - V_1^{(\tilde{\pi}^m)}(s_1)\right] \\
&= \sum_{m=1}^{M} \sum_{(n,t)\in\hat{\alpha}_m} \mathbb{E}\left[V_t^*(s_{m,n,t}) - Q_t^*(s_{m,n,t}, \hat{\pi}(s_{m,n,t}))\right] \\
&= \sum_{m=1}^{M} \sum_{t=1}^{H} \mathbb{E}\left[V_t^*(\hat{s}_{m,t}) - Q_t^*(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t}))\right]
\end{aligned}$$

where the last equality follows after relabeling the steps in the UCBVI pseudo-episodes. We then apply the same argument in the proof of Theorem 1 in [13]. Note that the pigeon-hole principle only works if we take the total time steps in the theorem to be the total time steps of the whole M meta-episodes. Therefore, the desired bound holds with probability at least $1 - \delta$. □

A.4 Proof of Lemma 4.6

Proof. By Bellman equations,

$$\begin{aligned}
&\hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \\
&= \left(\hat{R}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) + \left[P^{(\hat{\pi})}\right]^{\mathsf{T}} \hat{V}_{t+1}^{(\hat{\pi})}(\hat{s}_{m,t+1}) + b_m(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t}))\right) \\
&\quad - \left(\hat{R}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) + \left[P^{(\hat{\pi})}\right]^{\mathsf{T}} \bar{V}_{t+1}^{(\bar{\pi})}(\hat{s}_{m,t+1}) - b_m(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t}))\right) \\
&\leq \max_{q \in B_p^m} (q - p^*)^{\mathsf{T}} \hat{V}_{t+1}^{(\hat{\pi})}(\hat{s}_{m,t+1}) - \min_{q \in B_p^m} (q - p^*)^{\mathsf{T}} \bar{V}_{t+1}^{(\bar{\pi})}(\hat{s}_{m,t+1}) + 2b_m(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \\
&\quad + {p^*}^{\mathsf{T}} \left(\hat{V}_{t+1}^{(\hat{\pi})} - \bar{V}_{t+1}^{(\bar{\pi})}\right)(\hat{s}_{m,t+1}).
\end{aligned}$$

We define $(a)_{m,t} \coloneqq \max_{q \in B_p^m} (q - p^*)^{\mathsf{T}} \hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \min_{q \in B_p^m} (q - p^*)^{\mathsf{T}} \bar{V}_t^{(\bar{\pi})}(\hat{s}_{m,t})$ and $(b)_{m,t} \coloneqq 2b_m(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t}))$. For each t, note that

$$\begin{aligned}
\hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \bar{V}_t^{(\bar{\pi})}(\hat{s}_{m,t}) &= \hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \max_{a \in A} \bar{Q}_t^{(\bar{\pi})}(\hat{s}_{m,t}, a) \\
&\leq \hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})).
\end{aligned}$$

Thus,

$$\hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \leq (a)_{m,t+1} + (b)_{m,t} + {p^*}^{\mathsf{T}} \left(\hat{V}_{t+1}^{(\hat{\pi})} - \bar{Q}_{t+1}^{(\bar{\pi})}(\cdot, \hat{\pi})\right)(\hat{s}_{m,t+1}).$$

Continuing this argument, and noticing that by construction $\hat{V}_H^{(\hat{\pi})}(\hat{s}_{m,H}) = \bar{Q}_H^{(\bar{\pi})}(\hat{s}_{m,H}, \hat{\pi}(\hat{s}_{m,H})) = 0$, we have by induction that

$$\hat{V}_t^{(\hat{\pi})}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \leq \sum_{\ell=0}^{H-t} ((a)_{m,t+\ell+1} + (b)_{m,t+\ell})$$

Summing over the whole UCBVI episode, under the event $\mathcal{E}$, we have

$$\begin{aligned}
&\sum_{t=1}^{H} \hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \\
&\leq \sum_{t=1}^{H} \sum_{\ell=0}^{H-t} ((a)_{m,t+\ell+1} + (b)_{m,t+\ell}) \\
&= \sum_{\ell=1}^{H} \sum_{t=\ell}^{H} ((a)_{m,t+1} + (b)_{m,t}) \\
&\leq \sum_{\ell=1}^{H} \sum_{t=1}^{H} \left(4\sqrt{\frac{SL}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}} + 8H\sqrt{\frac{SL}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}}\right) \\
&= H \sum_{t=1}^{H} \left(4\sqrt{\frac{SL}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}} + 8H\sqrt{\frac{SL}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}}\right) \\
&= 4H\sqrt{SL}\,(1 + 2H) \sum_{t=1}^{H} \sqrt{\frac{1}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}}
\end{aligned}$$

where $L = \log(5SAH\sum_{m=1}^{M} N_m/\delta)$. Then, summing over the M meta-episodes, we have

$$\begin{aligned}
&\sum_{m=1}^{M}\sum_{t=1}^{H} \hat{V}_t^{(\hat{\pi}^m)}(\hat{s}_{m,t}) - \bar{Q}_t^{(\bar{\pi}^{m,n_t})}(\hat{s}_{m,t}, \hat{\pi}(\hat{s}_{m,t})) \\
&\leq 4H\sqrt{SL}(1+2H)\sum_{m=1}^{M}\sum_{t=1}^{H}\sqrt{\frac{1}{\max\{1, N_m(\hat{s}_{m,t}, \hat{a}_{m,t})\}}} \\
&\leq 4H\sqrt{SL}(1+2H)\sum_{s,a}\sum_{n=1}^{N_M(s,a)}\sqrt{\frac{1}{n}} \\
&\leq 12H^2S\sqrt{ALHN}.
\end{aligned}$$

□

A.5 Proof of Lemma 4.7

Proof. Under our assumption that the MDP is ergodic, let

$$\Gamma = \max_{s=s'}\max_{\pi}\mathbb{E}[T^{\pi}(s, s')] \leq \frac{H}{2}$$

be the worst-case diameter. Then given any initial state s', target state s and shield policy $\tilde{\pi}$, the expected exit time

$$\mathbb{E}[T^{\tilde{\pi}}(s', s)] \leq \Gamma.$$

By Markov's inequality,

$$\Pr(T^{\tilde{\pi}}(s', s) \geq \alpha H) \leq \frac{1}{2\alpha}.$$

Therefore, with probability at least $1 - (\frac{1}{2\alpha})^N$, during N episodes, there exists one where the MDP will reach s from s' using $\tilde{\pi}$ within αH steps. Letting $N = \log(\frac{1}{\delta})/\log(2\alpha)$ and $\alpha = 3/4$ completes the proof. □

B Experiments

In this section, we present experiments in a single-product stochastic inventory control problem to compare the performance and safety violations of our algorithm to those of UCBVI. We consider a similar inventory control problem as [15], but instead of an infinite horizon, we adopt a finite horizon. At the beginning of each month t, the manager notes the current inventory of a single product. They then decide the number of items to order from a supplier before observing the random demand. They have to account for the tradeoff between the costs of keeping inventory and lost sales or penalties resulting from being unable to satisfy customer demand. The objective is to maximize profit during the entire decision-making process.

The state space is the number of items in the inventory, $S = \{0, \ldots, M\}$, where $M = 5$ is the maximum capacity. The action space is $A_s = \{0, \ldots, M - s\}$ for each state $s \in S$. Given inventory state s_t at the beginning of month t, the number of items a_t to order is determined by the manager. We assume that a time-homogeneous uniform distribution D_t generates the random demand of each month t, and that the horizon is $H = 20$. The inventory at the beginning of month $t + 1$ is given by

$$s_{t+1} = \max\{0, s_t + a_t - D_t\}.$$

Next, we define the associated cost functions. We assume a fixed cost $K = 2$ for placing orders and a variable cost $c(u) = 2u$ that increases with the quantity ordered:

$$O(u) = \begin{cases} K + c(u) & \text{if } u > 0 \\ 0 & \text{if } u = 0. \end{cases}$$

The cost of maintaining an inventory of u units for a month is represented by the nondecreasing function $h(u) = u$. If the demand is j units and sufficient inventory is available to meet the

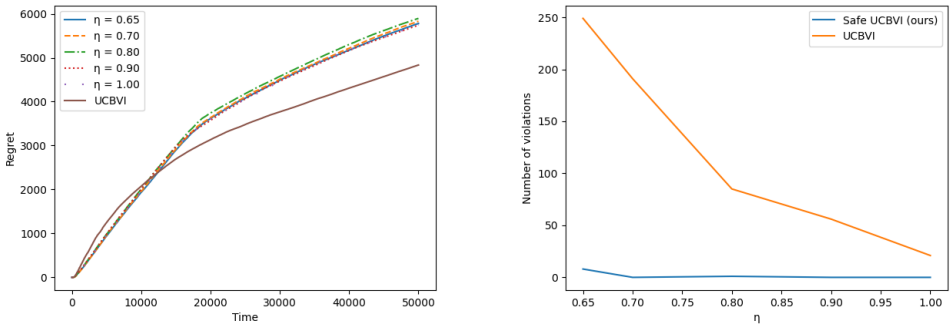

Figure 1: Results on the inventory control task. Left: Regret of our approach (for various exploration budgets η) vs. UCBVI. Right: Safety violations of our approach vs. UCBVI as a function of η.

demand, the manager receives a revenue of $f(j)$. Finally, the reward is defined as $r(s_t, a_t, s_{t+1}) = -O(a_t) - h(s_t + a_t) + f(s_t + a_t - s_{t+1})$, where we take $f(u) = 8u$ in our experiments. We normalize the rewards so that they are supported in $[0, 1]$. We use an offline dataset of 1500 randomly generated past episodes to warm-start the algorithms, and then compute the regret and number of safety violations corresponding to various safety budgets η. We use $N = 50000$ total episodes.

Figure 1 (left) shows that the regret of our algorithm starts out linearly increasing, since in the beginning the meta algorithm is forced to switch to the baseline policy a lot in one meta-episode to guarantee safety. As historical data accrues, our algorithm uses the UCBVI policy more frequently since its reward deficit decreases. At some point, our algorithm starts to converge at a similar rate as UCBVI. Note that UCBVI converges faster since it ignores the safety constraint and can explore arbitrarily, even if some actions result in poor values. Figure 1 (right) shows the number of times the safety constraint is violated. Our algorithm almost always satisfies the constraint for all shown values of η, whereas UCBVI fails to do so in a significant number of episodes, especially when η is small.