Abstract

While learning in an unknown Markov Decision Process (MDP), an agent should trade off exploration to discover new information about the MDP, and exploitation of the current knowledge to maximize the reward. Although the agent will eventually learn a good or optimal policy, there is no guarantee on the quality of the intermediate policies. This lack of control is undesired in real-world applications where a minimum requirement is that the executed policies are guaranteed to perform at least as well as an existing baseline. In this paper, we introduce the notion of conservative exploration for average reward and finite horizon problems. We present two optimistic algorithms that guarantee (w.h.p.) that the conservative constraint is never violated during learning. We derive regret bounds showing that being conservative does not hinder the learning ability of these algorithms.

1 Introduction

While Reinforcement Learning (RL) has achieved tremendous successes in simulated domains, its use in real system is still rare. A major obstacle is the lack of guarantees on the learning process, that makes difficult its application in domains where hard constraints (e.g., on safety or performance) are present. Examples of such domains are digital marketing, healthcare, finance, and robotics. For a vast number of domains, it is common to have a known and reliable baseline policy that is potentially suboptimal but satisfactory. Therefore, for applications of RL algorithms, it is important that are guaranteed to perform at least as well as the existing baseline.

In the offline setting, this problem has been studied under the name of safety w.r.t. a baseline (Bottou et al., 2013; Thomas et al., 2015a,b; Swaminathan and Joachims, 2015; Petrik et al., 2016; Laroche et al., 2019; Simão and Spaan, 2019). Given a set of trajectories collected with the baseline policy, these approaches aim to learn a policy –without knowing or interacting with the MDP– that is guaranteed (e.g., w.h.p.) to perform at least as good as the baseline. This requires that the set of trajectories is sufficiently reach in order to allow to perform counterfactual reasoning with it. This often implies strong requirements on the ability of exploration of the baseline policy. These approaches can be extended to a semi-batch settings where phases of offline learning are alternated with the executing of the improved policy. This is the idea behind conservative policy iteration (e.g., Kakade and Langford, 2002; Pirotta et al., 2013b) where the goal is to guarantee a monotonic policy improvement in order to overcome the policy oscillation phenomena Bertsekas (2011). These approaches has been successively extended to function approximation preserving theoretical guarantees (e.g., Pirotta et al., 2013a; Achiam et al., 2017). A related problem studied in RL is the one of safety, where the algorithm is forced to satisfy a set of constraints, potentially not directly connected with the performance of a policy (e.g., Altman, 1999; Berkenkamp et al., 2017; Chow et al., 2018). In the online setting, which is the focus of this paper, the learning agent needs to trade-off exploration and exploitation while interacting with the MDP. Opposite to offline learning, the agent has direct control over exploration. Exploration means that the agent is willing to give up rewards for policies improving his knowledge of the environment. Therefore, there is no guarantee on the performance of policies generated by the algorithm, especially in the initial phase where the uncertainty about the MDP is maximal and the algorithm has to explore multiple options (almost randomly). To increase the application of exploration algorithm, it is thus important that the policies selected by the algorithm are (cumulatively) guaranteed to perform as well as the baseline by making exploration more conservative. This setting has been studied in multi-

Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).
Conservative Exploration in Reinforcement Learning

We consider a Markov Decision Process (Puterman, 1994, Sec. 8.3) \( M = (S, A, p, r) \) with state space \( S \) and action space \( A \). Every state-action pair \((s, a)\) is characterized by a reward distribution with mean \( r(s, a) \) and support in \([0, r_{\text{max}}]\), and a transition distribution \( p(\cdot | s, a) \) over next states. We denote by \( S = |S| \) and \( A = |A| \) the number of states and action \( A \) stationary Markov randomized policy \( \pi : S \to P(A) \) maps states to distributions over actions. The set of stationary randomized (resp. deterministic) policies is denoted by \( \Pi_{\text{SR}} \) (resp. \( \Pi_{\text{SD}} \)). Any policy \( \pi \in \Pi_{\text{SR}} \) has an associated long-term average reward (or gain) and a bias function defined as

\[
g^\pi(s) := \lim_{T \to +\infty} \mathbb{E}^s_\tau \left[ \frac{1}{T} \sum_{t=1}^{T} r(s_t, a_t) \right]
\]

and

\[
h^\pi(s) := C \cdot \lim_{T \to +\infty} \mathbb{E}^s_\tau \left[ \sum_{t=1}^{T} \left( r(s_t, a_t) - g^\pi(s_t) \right) \right],
\]

where \( \mathbb{E}^s_\tau \) denotes the expectation over trajectories generated starting from \( s_1 = s \) with \( a_t \sim \pi(s_t) \). The bias \( h^\pi(s) \) measures the expected total difference between the reward and the stationary reward in Cesaro-limit (denoted by \( C\)-lim). We denote by \( sp(h^\pi) := \max_s h^\pi(s) - \min_s h^\pi(s) \) the span (or range) of the bias function.

**Assumption 1.** The MDP \( M \) is ergodic.

In ergodic MDPs, any policy \( \pi \in \Pi_{\text{SR}} \) has constant gain, i.e., \( g^\pi(s) = g^\pi \) for all \( s \in S \). There exists a policy \( \pi^* \in \arg \max_\pi g^\pi \) for which \( (g^\pi, h^\pi) \) satisfy the optimality equations,

\[
h^\pi(s) + g^\pi = Lh^\pi(s) := \max_{a \in A} \{ r(s, a) + p(\cdot | s, a)^T h^\pi \},
\]

where \( L \) is the optimal Bellman operator. We use \( D = \max_{s \neq s'} \min_{\pi \in \Pi_{\text{SD}}} \mathbb{E}[r_\pi(s'|s)] \) to denote the diameter of \( M \), where \( r_\pi(s'|s) \) is the hitting time of \( s' \) starting from \( s \). We introduce the “worst-case” diameter

\[
\Upsilon = \max_{s \neq s'} \max_{\pi \in \Pi_{\text{SD}}} \mathbb{E}[r_\pi(s'|s)],
\]

which defines the worst-case time it takes for any policy \( \pi \) to move from any state \( s \) to \( s' \). Asm.\( \Upsilon \) guarantees that \( D \leq \Upsilon < \infty \).

**Exploration in RL.** Let \( M^* \) be the true unknown MDP. We consider the learning problem where \( S, A \) and \( r_{\text{max}} \) are known, while rewards \( r \) and transition probabilities \( p \) are unknown and need to be estimated online. We evaluate the performance of a learning algorithm \( A \) after \( T \) time steps by its cumulative regret

\[
R(A, T) = T g^* - \sum_{t=1}^{T} r_t(s_t, a_t).
\]

The exploration-exploitation dilemma is a well-known problem in RL and (nearly optimal) solutions have been proposed in the literature both base on optimism-in-the-face-of-uncertainty (OFU, e.g., Jaksch et al., 2010; Bartlett and Tewari, 2009; Fruit et al., 2018a) and Thompson sampling (TS, e.g., Gopalan and Mannor, 2015; Osband and Roy, 2018). Refer to Lazarevic et al. (2019) for more details.
3 Conservative Exploration in RL

In conservative exploration, a learning agent is expected to perform as well as the optimal policy over time (i.e., regret minimization) under the constraint that at no point in time its performance is significantly worse than a known baseline policy $\pi_b \in \mathcal{P}^{SR}$. This problem has been studied in the bandit literature [Wei et al., 2016; Kazerouni et al., 2017], where the conservative constraint compares the cumulative expected reward obtained by the actions $a_1, a_2, \ldots, a_t$ selected by the algorithm to the one of the baseline action $a_b$,

$$\forall t > 0, \quad \sum_{i=1}^{t} r(a_i) \geq (1 - \alpha) t r(a_b), \quad (3)$$

where $r(a)$ is the expected reward of action $a$. At any time $t$, conservative exploration algorithms first query a standard regret minimization algorithm (e.g., UCB) and decide whether to play the proposed action $\tilde{a}_t$ or the baseline $a_b$ based on the accumulated budget (i.e., past rewards) and whether the estimated performance of $\tilde{a}_t$ is sufficient to guarantee that the conservative constraint is satisfied at $t + 1$ after $\tilde{a}_t$ is executed. While (3) effectively formalizes the objective of constraining an algorithm to never perform much worse than the baseline, in RL it is less obvious how to define such constraint. In the following we review three possible directions, we point out their limitations, and we finally propose a conservative condition for RL for which we derive an algorithm in the next section.

Gain-based condition. Instead of actions, RL exploration algorithms (e.g., UCRL2), first select a policy and then execute the corresponding actions. As a result, a direct way to obtain a conservative condition is to translate the reward of each action in (3) to the gain associated to the policies selected over time, i.e.,

$$\forall t > 0, \quad \sum_{i=1}^{t} g^{\pi_i} \geq (1 - \alpha) t g^{\pi_b}. \quad (4)$$

The main drawback of this formulation is that the gain $g^{\pi_i}$ is the expected asymptotic average reward of a policy and it may be very far from the actual reward accumulated while executing $\pi_i$ in the specific state $s_i$ achieved at time $i$. The same reasoning applies to the baseline policy, whose cumulative reward up to time $t$ may significantly differ from $t$ times its gain. As a result, an algorithm that is conservative in the sense of (4) may still perform quite poorly in practice depending on $t$, the initial state, and the actual trajectories observed over time.

Reward-based condition. In order to address the concerns about the gain-based condition, we could define the stronger condition

$$\forall t > 0, \quad \sum_{i=1}^{t} r_i \geq (1 - \alpha) \sum_{i=1}^{t} r_i^b, \quad (5)$$

where $r_i$ is the sequence of rewards obtained while executing the algorithm and $r_i^b$ is the reward obtained by the baseline. While this condition may be desirable in principle (the learning algorithm never performs worse than baseline), it is impossible to achieve. In fact, even if the optimal policy $\pi^*$ is executed for all $t$ steps, the condition may still be violated because of an unlucky realization of transitions and rewards. If we wanted to accounting for the effect of randomness, we would need to introduce an additional slack of order $\mathcal{O}(\sqrt{t})$ (i.e., the cumulative deviation due to the randomness in the environment), which would make the condition looser and looser over time.

Condition in expectation. The previous remarks could be solved by taking the expectation of both sides

$$\forall t > 0, \quad \mathbb{E}_A \left[ \sum_{i=1}^{t} r_i(s_i, a_i) \mid s_1 = s \right] \geq (1 - \alpha) \mathbb{E}_A \left[ \sum_{i=1}^{t} r_i(s_i, a_i) \mid s_1 = s, \pi_b \right], \quad (6)$$

where $\mathbb{E}_A$ denotes the expectation w.r.t. the trajectory of states and actions generated by the learning algorithm $\mathcal{A}$, while the RHS is simply the expected reward obtain by running the baseline for $t$ steps. Condition (6) effectively captures the nature of the RL problem w.r.t. the bandit case. In fact, after $t$ steps, the actions followed by the learning algorithm may lead to a state that is possibly very different from the one we would have reached by playing only the baseline policy from the beginning. This deviation in the state dynamics needs to be taken into account when deciding if an exploratory policy is safe to play in the future. In the bandit case, selecting the baseline action contributes to build a conservative budget that can be spent to play explorative actions later on (i.e., by selecting $a_b$, the LHS of (6) in increased by $r(a_b)$, while only a fraction $1 - \alpha$ is added to the RHS, thus increasing the margin that may allow playing alternative actions later). In the RL case, selecting policy $\pi_b$ at time $t$ may not immediately contribute to increasing the conservative budget. In fact, the state $s_i$ where $\pi_b$ is applied may significantly differ from the state that $\pi_b$ would have achieved had we selected it from the beginning. As a result, a conservative RL algorithm should be extra-cautious when selecting policies different from $\pi_b$ since their execution may lead to unfavorable states, where it is difficult to recover good performance, even when selecting the baseline policy.
While this may seem a reasonable requirement, unfortunately it is impossible to build an empirical estimate of \( \epsilon \) that a conservative exploration algorithm could use to guide the choice of policies to execute. In fact, the LHS averages the performance of the algorithm over multiple executions, while in practice we have only access to a single realization of the algorithm’s process. This prevents from constructing accurate estimates of such expectation directly from the data observed up to time \( t \). A possible approach would be to construct an estimate of the MDP and use it to replay the algorithm itself for \( t \) steps. Beside prohibitive computational complexity, the resulting estimate of the expected cumulative reward of \( A \) would suffer from an error that increases with \( t \), thus making it a poor proxy for \( \epsilon \).

**Condition with conditional expectation.** Let \( t \) be a generic time and \( \mu_t = (\pi_1, \pi_2, \ldots, \pi_t) \), the non-stationary policy executed up to \( t \). We require the algorithm to satisfy the following conditional conservative condition

\[
\forall t > 0, \quad \mathbb{E} \left[ \sum_{i=1}^{t} r_i(s_i, a_i)|s_1 = s, \mu_t \right] \\
\geq (1 - \alpha) \mathbb{E} \left[ \sum_{i=1}^{t} r_i(s_i, a_i)|s_1 = s, \pi_b \right].
\]

(7)

where the expectations are taken w.r.t. the trajectories generated by a fixed non-stationary policy \( \mu_t \) (i.e., we ignore how rewards affect \( \mu_t \)). Notice that this condition is now stochastic, as \( \mu_t \) itself is a random variable and thus we require to satisfy (7) with high probability. This formulation can be seen as relying on a pseudo-performance evaluation of the algorithm instead of the actual expectation as in (6) and it is similar to (5), which takes the expected performance of each of the (random) actions, thus ignoring their correlation with the rewards. This formulation has several advantages w.r.t. the conditions proposed above: 1) it considers the sum of rewards rather than the gain as (4), thus capturing the dynamical nature of RL, 2) it contains expected values, so as to avoid penalizing the algorithm by unlucky noisy realizations as (3), 3) as shown in the next section, it can be verified using the samples observed by the algorithm unlike (6).

\[ 1 \text{More precisely, let } \hat{\mu}_t \text{ be an estimate of } M^* \text{ and } \epsilon_t \text{ be the largest error in estimating its dynamics at time } t. \text{ Estimating the expected cumulative reward by running (an infinite number of) simulations of } A \text{ in } \hat{\mu}_t \text{ would suffer from an error scaling as } t \epsilon_t. \text{ For any regret minimization algorithm, } \epsilon_t \text{ cannot decrease linearly with } t \text{ and thus the estimation of } \mathbb{E}_{\hat{A}} \text{ would have an error increasing with } t. \]

\[ 2 \text{ We use pseudo-performance to stress the link the pseudo-regret formulations used in bandit (e.g., \cite{auer2002}) to the finite-horizon case. We conclude the section, by reformulating (7) in the finite-horizon case. In this setting, the learning agent interacts with the environment in episodes of fixed length } H. \text{ Let } \pi \text{ be the initial state, } \pi_t \text{ be the policy proposed at episode } j \text{ and let } t = (k-1)H + 1 \text{ be beginning of the } k \text{-th episode. Then } \mu_t \text{ is a sequence of policies } \pi_j, \text{ each executed for } H \text{ steps. In this case, condition (7) can be conveniently written as}
\]

\[
(1 - \alpha)kV^*(\pi) \leq \mathbb{E} \left[ \sum_{i=1}^{t} r_i(s_i, a_i)|s_1 = s, \mu_t \right]
\]

\[
= \sum_{j=1}^{k} \mathbb{E} \left[ \sum_{i=1}^{H} r_i(s_i, a_i)|s_1 = s, \pi_j \right] = \sum_{j=1}^{k} V_1^{\pi_j}(\pi)
\]

(8)

where \( V_1^{\pi_j}(\pi) \) is the \( H \) step value function of \( \pi \) at the first stage. In this formulation, the conservative condition has a direct interpretation, as it directly mimics the bandit case (3). In fact, the performance of the algorithm up to episode \( k \) is simply measured by the sum of the value functions of the policies executed over time (each for \( H \) steps) and it is compared to the value function of the baseline itself. Note that this definition is compatible with the regret: \( R_T(\mathcal{A}, K) = \sum_{k=1}^{K} V^*(\pi) - V^*(\pi) \). Indeed, the regret defined in expectation w.r.t. the stochasticity of the model but not w.r.t. the algorithm, there is no expectation w.r.t. the possible sequence of policies generated by \( \mathcal{A} \).

4 Conservative UCRL

In this section, we introduce conservative upper-confidence bound for reinforcement learning (CUCRL2), an efficient algorithm for exploration-exploitation in average reward that both minimize the regret and satisfy condition (7).

CUCRL2 builds on UCRL2 in order to perform efficient conservative exploration. At each episode \( k \), CUCRL2 builds a bounded parameter MDP \( \mathcal{M}_k = \{ M = (S, \mathcal{A}, r, p), r(s, a) \in B^b_k(s, a), p(\cdot|s, a) \in B_p^k(s, a) \} \), where \( B^b_k(s, a) \subseteq [0, r_{\max}] \) and \( B^b_k(s, a) \subseteq \Delta_S \) are high-probability confidence intervals on the rewards and transition probabilities such that \( M^* \in \mathcal{M} \) w.h.p. and \( \Delta_S \) is the \( S \)-dimensional simplex. This confidence intervals can be built using Hoeffding or empirical Bernstein inequalities by using the samples available at episode \( k \) (e.g., \cite{Jaksch2010, Fruit2018}). CUCRL2 computes an optimistic policy \( \pi_k \) in the same way as UCRL2: \( (\hat{M}_k, \pi_k) \in \arg \max_{M \in \mathcal{M}_k, \pi \in \Pi^D} \{ \mathbb{E}\pi(M) \} \). This problem can be solved using EVI (see Fig. 3 in appendix) on the optimistic Bellman operator \( \mathcal{L}_G^k \) of \( \mathcal{M}_k \) (Jaksch et al. 2010). Then, it needs to decide whether policy

\[ 3 \mathcal{L}_G^k v(s) = \max_{\pi \in \Pi^D} \{ \mathbb{E}_{\pi} v(s, a) | r \} + \max_{p \in B^b_k(s, a)} \mathbb{E} \pi v \].
The baseline policy, we need to consider the specific RL structure to obtain a condition that can be verified during the execution of the algorithm itself. In order to simplify the derivation, we rely on the following assumption.

**Assumption 2.** The gain and bias function \( (g^*, h^*) \) of the baseline policy are known.

As explained in [Kazerouni et al., 2017], this is a reasonable assumption since the baseline policy is assumed to be the policy currently executed by the company and for which historical data are available. We will mention how to relax this assumption in Sec. 4.1.

We follow two main steps in deriving a checkable condition. 1) We need to estimate the cumulative reward obtained by each of the policies played by the learning algorithm directly from the samples observed so far. We do this by relating the cumulative reward to the gain and bias of each policy and then building their estimates. 2) It is necessary to evaluate whether the policy proposed by UCRL2 is safe to play w.r.t. the conservative condition, before actually executing it. While this is simple in batch, as each action is executed for only one step. In RL, policies cannot be switched at each step and need to be played for a whole episode. Nonetheless, the length of a UCRL2 episode is not known in advance and this requires predicting for how long the explorative policy could be executed in order to check its performance.

**Step 1: Estimating the conditional conservative condition from data.** In order to evaluate (7) from data, one may be tempted to first replace the sum of rewards obtained by each policy \( \pi_j \) in \( \mu_\pi \) on the lhs side by its gain \( g^\pi_j \), similar to the gain-based condition in (4). Indeed, under Asm. 1 any stationary policy \( \pi \) receives asymptotically an expected reward \( g^\pi \) at each step. Unfortunately, in our case \( \mathbb{E} \left[ \sum_{i=1}^t r_i | \mu_j \right] \neq \sum_{j=1}^t T_j g^\pi_j \). In fact, when evaluating a policy for a finite number of steps, we need to account for the time required to reach the steady regime (i.e., mixing time) and, as such, the influence of the state at which the policy is started. The notion of reward collected during the transient regime is captured by the bias function \( h^\pi \). In particular, for any stationary (unichain) policy \( \pi \in \Pi^{SR} \) with gain \( g^\pi \) and gain function \( h^\pi \) executed for \( t \) steps, we have that:

\[
\mathbb{E} \left[ \sum_{i=1}^t r_i | s_1 = s, \pi \right] = t \ g^\pi + h^\pi(s) - P^\pi_t(\cdot | s)^\top h^\pi. \tag{9}
\]

Puterman [1994] Sec. 8.2.1 refers to the gain as “stationary” reward while to the bias as “transient” reward.
As a result, we have the bounds
\[
t^\pi - sp(h^\pi) \leq \mathbb{E}\left[ \sum_{i=1}^{t} r_i | s_1 = s, \pi \right] \leq t^\pi + sp(h^\pi).
\]
Leveraging prior knowledge of the gain and bias of the baseline, we can use the second inequality to directly upper bound the baseline performance as
\[
\mathbb{E}\left[ \sum_{i=1}^{t} r_i | s_1 = s, \pi_b \right] \leq sp(h^{\pi_b}) + t^\pi_b. \tag{10}
\]
On the other hand, for a generic policy \(\pi\), the gain and bias cannot be directly computed since \(M^*\) is unknown. To estimate the cumulative reward of the algorithm we resort to the estimate of the true MDP build by UCRL2 to construct a pessimistic estimate of the cumulative reward for any policy \(\pi_j\) (i.e., to perform counterfactual reasoning).

Given a policy \(\pi\) and the bounded-parameter MDP \(\mathcal{M}_k\), we are interested in finding \(g^\pi\) such that:
\[
g^\pi := \min_{M \in \mathcal{M}_k} \{g^\pi(M)\}.
\]
Define the Bellman operator \(L_k^\pi\) associated to \(\mathcal{M}_k\) as:
\[
L_k^\pi v(s) := \min_{r \in B_k^\pi(s,a)} \min_{p \in B_k^\pi(s,a)} \{p^rv\}. \tag{11}
\]
Then, there exists \((g^\pi, h^\pi)\) such that, \(\forall s \in \mathcal{S}\),
\[
g^\pi e + h^\pi = L_k^\pi h^\pi \quad \text{where} \quad e = (1, \ldots, 1) \quad \text{see Lem. A.1 in App. A}.
\]
Similarly to what is done by UCRL2, we can use EVI with \(L_k^\pi\) to build an \(\epsilon_k\)-approximate solution of the Bellman equations. Let \((g_n, v_n) = \text{EVI}(L_k^\pi, \epsilon_k)\), then \(g_n - \epsilon_k \leq g^\pi \leq g^\pi(M^*)\). The values computed by the pessimistic policy evaluation can be then used to bound the cumulative reward of any stationary policy.

**Lemma 1.** Consider a bounded parameter MDP \(\mathcal{M}\) such that \(M^* \in \mathcal{M}\) w.h.p., a policy \(\pi\) and let \((g_n, v_n) = \text{EVI}(L^\pi, \epsilon)\). Then, under Assm. \([\square]\) for any state \(s \in \mathcal{S}\):
\[
\mathbb{E}\left[ \sum_{i=1}^{t} r_i | s_1 = s, \pi \right] \geq t(g_n - \epsilon) - sp(v_n).
\]

**Step 2: Test safety of optimistic policy.** Let \(t_k\) be the time when episode \(k\) starts. Policies \(\pi_1, \ldots, \pi_{k-1}\) have been executed until \(t_k - 1\) and UCRL2 computed an optimistic policy \(\pi_k\). In order to guarantee that \([\square]\) is verified the algorithm needs to anticipate how well \(\pi_k\) may perform if executed for the next episode. For any policy \(\{\pi_j\}_{j < k} \cup \{\pi_k\}\), we first compute \((g_j^\pi, h_j^\pi) = \text{EVI}(\pi_j^\pi, \epsilon_j)\). If \(\pi_j = \pi^b\) (i.e., the baseline was executed at episode \(j\)), we let \((g_j^\pi, h_j^\pi) = (g^\pi_n, h^\pi_n)\) and \(\epsilon_j = 0\). Then
\[
\mathbb{E}\left[ \sum_{i=1}^{t} r_i | s_1 = s, \mu_t \right] = \sum_{j=1}^{k} \sum_{y \in \mathcal{S}} \mathbb{P}(s_{t_j} = y | s_1 = s, \mu_t) \cdot \mathbb{E}\left[ \sum_{i=1}^{T_j} r_i | y, \pi_j \right] \tag{12}
\]
\[
\geq \sum_{j=1}^{k} T_j (g_j^\pi - \epsilon_j) - sp(h_j^\pi) \tag{13}
\]
where \(t_j \) is time at which episode \(j\) started, \(\mathbb{P}(s_{t_j} = y | s_1 = s, \mu_t)\) is the probability of reaching state \(s\) after \(t_j\) steps starting from state \(s\) following policy \(\mu_t\). The inequality follows from Lem. [\square]. By lower bounding the LHS of \([\square]\) by \([\square]\) and upper bounding the RHS by \([\square]\), the conservative condition becomes:
\[
\sum_{j=1}^{k-1} \left( T_j (g_j^\pi - \epsilon_j - g^\pi) - sp(h_j^\pi) \right) - sp(h^\pi_n) + T_k (g_k^\pi - \epsilon_k - (1 - \alpha)g^\pi) \geq 0 \tag{14}
\]
Note that the algorithm should check this condition at the beginning of episode \(k\) in order to understand if the policy \(\pi_k\) is safe or if it should resort to playing policy \(\pi^b\). In many OFU algorithms, including UCRL2, the length of episode \(k\) (i.e., \(T_k\)) is not known at the beginning of the episode. As a consequence, condition \([\square]\) is not directly computable. To overcome this limitation, we consider the dynamic episode condition introduced by [Ouyang et al., 2017]. This stopping condition provides an upper-bound on the length of each episode as \(T_k \leq T_{k-1} + 1\), without affecting the regret bound of UCRL2 (up to constants). This condition can be used to further lower-bound the last term in \([\square]\) by
\[
T_k (g_k^\pi - \epsilon_k - (1 - \alpha)g^\pi) \geq (T_{k-1} + 1)(g_k^\pi - \epsilon_k - (1 - \alpha)g^\pi) \cdot \mathbb{1}_{(1-\alpha)g^\pi \geq g_k^\pi - \epsilon_k}. \tag{15}
\]
Plugging this lower bound into \([\square]\) gives the final conservative condition
\[
\sum_{j=1}^{k-1} \left( T_j (g_j^\pi - \epsilon_j - g^\pi) - sp(h_j^\pi) \right) - sp(h^\pi_n) + (T_{k-1} + 1)(g_k^\pi - \epsilon_k - (1 - \alpha)g^\pi) \cdot \mathbb{1}_{(1-\alpha)g^\pi \geq g_k^\pi - \epsilon_k} \geq 0,
\]
tested by CUCRL2 at the beginning of each episode. Unknown \((g^\pi_n, h^\pi_n)\). If the gain and bias of the baseline are unknown, we can use EVI on \(L_j^\pi, \nu_k\) (Eq. [\square] with max instead of min) to compute an optimistic
estimate of the cumulative reward of the baseline up to time \( t_l + T_{l-1} + 1 \). While this account for the RHS of Eq. (2), we simply define \((g_l^t, h_l^t) = \text{EVI}(L_{t}^{\pi_b}, e_l)\) for every episode \( l \in \mathcal{A}_{T-1} \) to compute a lower bound to the cumulative reward obtained by the algorithm by playing the baseline in episode before \( k \). Clearly, this approach is very pessimistic and it may be possible to design better strategies for this case.

The finite-horizon case. We conclude this section with a remark on the finite horizon case. This case is much simpler and resemble the bandit setting. We can directly build a lower bound \( g_{l,1} \) to the value function \( V_{l}^{\pi_b} \) by using the model estimate and its uncertainty at episode \( l \). This estimate can be computed via extended backward induction—see \citep{Aazar et al. 2017 Alg. 2}—simply subtracting the exploration bonus, see Lem. 10 in App. C. The same approach can be used to construct and optimistic and pessimistic estimate of \( V_{l}^{\pi_b} \) when it is unknown. This values can be directly plug in \( \mathcal{S} \) to define a checkable condition for the algorithm.

4.2 Regret Guarantees

We start providing an upper-bound to the regret of CUCRL2 showing the dependence on UCRL2 and on the baseline \( \pi_b \). Since the set \( \Lambda_k \) is updated at the end of the episode, we denote by \( \Lambda_T = \Lambda_{k_T} \cup \{ k_T \} \cdot 1 \) (Eq. (15)) the set containing all the episodes where CUCRL2 played an optimistic policy. The set \( \Lambda_T \) is its complement.

**Lemma 2.** Under Asm. 1 and 3, for any \( T \) and any conservative level \( \alpha \), there exists a numerical constant \( \beta > 0 \) such that the regret of CUCRL2 is upper-bounded as

\[
R(\text{CUCRL2}, T) \leq \beta \cdot \left( R_{\text{UCRL2}}(T|\Lambda_T) + (g^* - g^{\pi_b}) \sum_{l \in \Lambda_T} T_l + sp(h^{\pi_b}) \sqrt{SAT \ln(T/\delta)} \right),
\]

and the conservative condition (7) is met at every step \( t = 1, \ldots, T \) with probability at least \( 1 - \frac{2T}{\beta} \).

The regret of UCRL2 over an horizon \( T \) conditioned on the fact that the UCRL2 policy is executed only at episodes \( k \in \Lambda_T \). During the other episodes, the internal statistics of UCRL2 are updated using the samples collected by the baseline policy \( \pi_b \). This does not pose any major technical challenge and, as shown in App. B, the UCRL2 regret can be bounded as follows.

**Lemma 3** \citep{Jaksch et al. 2010}. Let \( L_T = \ln \left( \frac{T}{\delta} \right) \), for any \( T \), there exists a numerical constant \( \beta > 0 \) such that, with probability at least \( 1 - \frac{2T}{\beta} \),

\[
R_{\text{UCRL2}}(T|\Lambda_T) \leq \beta DS\sqrt{ATL_T} + \beta DS^2 A L_T
\]

The second term in Lem. 2 represents the regret incurred by the algorithm when playing the baseline policy \( \pi_b \). The following lemma shows that the total time spent executing conservative actions is sublinear in time (see Lem. 8 in App. D for details).

**Lemma 4.** For any \( T > 0 \) and any conservative level \( \alpha \), with probability at least \( 1 - \frac{2T}{\beta} \), the total number of play of conservative actions is bounded by:

\[
\sum_{l \in \Lambda_T} T_l \leq 2\sqrt{SAT \ln(T)} + \frac{112SALT}{(\alpha g^{\pi_b})^2} (1 + S(D + \Upsilon)^2) + \frac{16\sqrt{TLT}}{\alpha g^{\pi_b}} \left[ (D + \Upsilon)\sqrt{SA} + r_{\max} + \sqrt{SA} s p(h^{\pi_b}) \right]
\]

where \( L_T = \ln \left( \frac{5SAT}{\delta} \right) \) and \( \Upsilon < \infty \) as in Eq. 7.

**Proof.** Let \( \tau \) be the last episode played conservatively: \( \tau = \sup \{ k > 0 : k \in \Lambda_T \} \). This means that at the beginning of episode \( \tau \) the conservative condition was not verified. By rearranging the terms in Eq. (15) and using simple bounds, we can write that:

\[
\Delta_t + 4k_T \left( sp(g^{\pi_b}) + \Upsilon + (1 - \alpha)r_{\max} \right) \geq \alpha \sum_{l=1}^{\tau-1} T_l g^{\pi_b}
\]

where \( \Delta_{\tau} := \sum_{l \in \Lambda_{\tau}} T_l(\tilde{g}_l - \hat{g}_l) \) and \((\tilde{g}_l, \hat{g}_l)\) are optimistic and pessimistic gain of policy \( \pi_t = \pi_{\tau} \).

Note that both satisfies the Bellman equation: \( (\hat{g}_l + \tilde{h}_l) = L_{l}^{\dagger} \hat{h}_l \) (see footnote 3) and \( g_l + h_l = L_{l}^{\dagger} h_l \) (see Eq. 14). At this point, the important terms in upper-bounding \( \Delta_{\tau} \) are similar to the one analysed in UCRL2. In particular, we have a term depending on the confidence intervals \( \Delta_{\alpha l}^{\tau} := 2\beta^l(s_l, a_l) + \beta^l_{\pi}(s_l, a_l) \text{sp}((\hat{h}_l) + \text{sp}(\tilde{h}_l^{\pi})) \) and one depending on the transitions \( \Delta_{\pi}^{\tau} := \sigma^l(s_l, a_l)\left( (\tilde{h}_l + \hat{h}_l) - (\hat{h}_l(s_{l+1}) - \tilde{h}_l^{\pi}(s_{l+1})) \right) \).

Let \( X = \sum_{l=1}^{T} T_l \). By using the definition of the confidence intervals, it is easy to show that

\[
\sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \Delta_{\alpha l}^{\tau} \lesssim \sqrt{SA} X + (D + \Upsilon)\sqrt{SA} X.
\]

Define the \( \sigma \)-algebra based on past history at \( t \): \( F_t = \sigma(s_1, a_1, r_1, s_2, a_2, r_2, \ldots, s_{l+1}) \). The sequence \( \{ \Delta_{\alpha l}^{\tau}, F_t \}_{t=1}^{T} \) is an MDS. Thus, using Azuma inequality we have that

\[
\sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \Delta_{\alpha l}^{\tau} \lesssim (D + \Upsilon)\sqrt{T}.
\]

Putting everything together we have a quadratic form in \( X \) and solving it we can write that \( \alpha g^{\pi_b} X \lesssim b_T + S^2 A L_T (D + \Upsilon)^2 / (\alpha g^{\pi_b}) \) where \( b_T = O(D + T)\sqrt{SA} \) (see App. B). The result follows noticing that \( \sum_{l \in \Lambda_{\tau}} T_l \leq \sum_{t=1}^{T} T_l + T_{\tau} \).
Combining the results of Lem. 2 and Lem. 3 into Lem. 2 leads to an overall regret of order $\tilde{O}(\sqrt{T})$, which matches the regret of UCRL2. This shows that CUCRL2 is able to satisfy the conservative condition without compromising the learning performance. Nonetheless, the bound in Lem. 4 shows how conservative exploration is more challenging in RL compared to the bandit setting. While the dependency on the conservative level $\alpha$ is the same, the number of steps the baseline policy is executed can be as large as $\tilde{O}(\sqrt{T})$ instead of constant as in CUCB (Wu et al., 2016). Furthermore, Lem. 2 relies on an ergodicity assumption instead of the milder communicating assumption needed by UCRL2 to satisfy Lem. 3. Asm. 1 translates into the bound through the “worst-case” diameter $\Upsilon$, which in general is much larger than the diameter $D$. This dependency is due to the need of computing a lower bound to the reward accumulated by policies $\pi_j$ in the past (see Lem. 1). In fact, UCRL2 only needs to compute upper bounds on the gain and the value function returned by EVI by applying the optimistic Bellman operator $L^+ \pi$ has span bounded by the diameter $D$. This is no longer the case for computing pessimistic estimates of the value of a policy. Whether Asm. 1 and the worst-case diameter $\Upsilon$ are the unavoidable price to pay for conservative exploration in infinite horizon RL remains as an open question.

The finite-horizon case. App. C shows how to modify UCB-VI (Azar et al., 2017) to satisfy the conservative condition in Eq. 8. In this setting, it is possible to show (see Prop. 1 in appendix) that the number of conservative episodes is simply logarithmic in $T = KH$. Formally, $|A^c_T| = O(H^2S^2\log(T/\delta)/(\alpha r_b (\Delta \alpha + \alpha r_b))$ where $0 < r_b \leq r(s,a)$, for all $(s,a)$, and $\Delta \alpha = \min\{V^\pi_1(s) - V^\pi_0(s)\}$ is the optimality gap. This problem dependent terms resemble the one in the bandit analysis. The regret of conservative UCB-VI is thus dominated by the UCB-VI term $\tilde{O}(H\sqrt{SAT})$.

5 Experiments
In this section, we report results in the inventory control problem to illustrate the performance of CUCRL2 compared to unconstrained UCRL2 and how it varies with the conservative level. See App. D for additional experiments for both average reward and finite horizon. In order to have a better estimate of the budget, we re-evaluate past policies at each episode. We start considering the stochastic inventory control problem (Puterman, 1994, Sec. 3.2.1) with capacity $M = 6$ and uniform demand. At the beginning of a month $t$, the manager has to decide the number of items to order in order to satisfy the random demand, taking into account the cost of ordering and maintenance of the inventory (see App. D). Since the optimal policy is a threshold policy, as baseline we consider a $(\sigma, \Sigma)$ policy (Puterman, 1994, Sec. 3.2.1) with target stock $\Sigma = 4$ and capacity threshold $\sigma = 4$. Note that $g^* = 0.603$ and $(g^\pi, sp(h^\pi)) = (0.565, 0.651)$. We use this domain to perform an ablation study w.r.t. the conservative level $\alpha$. We have taken $T = 70000$ and the results are averaged over 100 realizations.

Fig. 2(left) shows that the regret of CUCRL2 grows at the same speed as the one of the baseline policy $\pi_b$ at the beginning (the conservative phase), because during this phase CUCRL2 is constrained to follow $\pi_b$ to make sure that constraint (7) is satisfied. Clearly, the duration of this conservative phase is proportional to the conservative level $\alpha$. As soon as CUCRL2 has built margin, it starts interleaving exploratory (optimistic) policies with the baseline. After this phase, CUCRL2 has learn enough about the system and has a sufficient margin to behave as UCRL2. As expected, Fig. 2(left) confirms that the convergence to the UCRL2 behavior happens more quickly for larger values of $\alpha$, i.e., when the conservative condition is relaxed and CUCRL2 can explore more freely. On the other hand, UCRL2 converges faster since it is agnostic to the safety constraint and may explore very poor policies in the initial phase. To better understand this condition, Fig. 2(right) shows the percentage of time the constraint was violated in the first 15000 steps (about 20% of the overall time $T$). CUCRL2 always satisfies the constraint for all values of $\alpha$ while UCRL2 fails a significant number of times, especially when the conditions is tight (small values of $\alpha$).

6 Conclusion
We presented algorithms for conservative exploration for both finite horizon and average reward problems with $O(\sqrt{T})$ regret. We have shown that the non-episodic nature of average reward problems makes the definition of the conservative condition much harder than in finite horizon problems. In both cases, we used a model-based approach to perform counterfactual reasoning required by the conservative condition. Recent papers have focused on model-free exploration in tabular settings or linear function approximation (Jin et al., 2018; Yang and Wang, 2019; Jin et al., 2019), thus a question is if it is possible for model-free algorithms to be conservative and still achieve $O(\sqrt{T})$ regret.
References

Joshua Achian, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In ICML, volume 70 of Proceedings of Machine Learning Research, pages 22–31. PMLR, 2017.

Eitan Altman. Constrained Markov decision processes, volume 7. CRC Press, 1999.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. Machine Learning, 47(2-3):235–256, 2002.

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In ICML, volume 70 of Proceedings of Machine Learning Research, pages 263–272. PMLR, 2017.

Peter L. Bartlett and Ambuj Tewari. REGAL: A regularization based algorithm for reinforcement learning in weakly communicating MDPs. In UAI, pages 35–42. AUAI Press, 2009.

Felix Berkenkamp, Matteo Turchetta, Angela P. Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. In NIPS, pages 908–918, 2017.

Dimitri P Bertsekas. Dynamic programming and optimal control. Vol II. Athena scientific Belmont, MA, 1995.

Dimitri P Bertsekas. Approximate policy iteration: A survey and some new methods. Journal of Control Theory and Applications, 9(3):310–335, 2011.

L. Bottou, J. Peters, J. Quinonero-Candela, D. Charles, D. Chickering, E. Portugaly, D. Ray, P. Simard, and E. Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. Journal of Machine Learning Research, 14:3207–3260, 2013.

Yinlam Chow, Ofir Nachum, Edgar A. Duenez-Guzman, and Mohammad Ghavamzadeh. A Lyapunov-based approach to safe reinforcement learning. In NeurIPS, pages 8103–8112, 2018.

Ronan Fruit, Matteo Pirotta, and Alessandro Lazaric. Near optimal exploration-exploitation in non-communicating markov decision processes. In NIPS, 2018a.

Ronan Fruit, Matteo Pirotta, Alessandro Lazaric, and Ronald Ortner. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In ICML, Proceedings of Machine Learning Research. PMLR, 2018b.

Aditya Gopalan and Shie Mannor. Thompson sampling for learning parameterized markov decision processes. In COLT, volume 40 of JMLR Workshop and Conference Proceedings, pages 861–898. JMLR.org, 2015.

Thomas Jakob, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. Journal of Machine Learning Research, 11:1563–1600, 2010.

Chi Jin, Zeyuan Allen-Zhu, Sébastien Bubeck, and Michael I. Jordan. Is q-learning provably efficient? CoRR, abs/1807.03765, 2018.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I. Jordan. Provably efficient reinforcement learning with linear function approximation. CoRR, abs/1907.05388, 2019.

Sham M. Kakade and John Langford. Approximately optimal approximate reinforcement learning. In ICML, pages 267–274. Morgan Kaufmann, 2002.

Sumeet Katariya, Branislav Kveton, Zheng Wen, and Vanski K. Potluru. Conservative exploration using interleaving. In AISTATS, volume 89 of Proceedings of Machine Learning Research, pages 954–963. PMLR, 2019.

Abbas Kazerouni, Mohammad Ghavamzadeh, Yasin Abbasi, and Benjamin Van Roy. Conservative contextual linear bandits. In NIPS, pages 3910–3919, 2017.

Romain Laroche, Paul Trichelair, and Remi Tachet des Combes. Safe policy improvement with baseline bootstrapping. In ICML, volume 97 of Proceedings of Machine Learning Research, pages 3652–3661. PMLR, 2019.

Alessandro Lazaric, Matteo Pirotta, and Ronan Fruit. Regret minimization in infinite-horizon finite markov decision processes. Tutorial at ALT’19, 2019. URL https://rlgammazero.github.io/.

Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In Proceedings of the Sixteenth ACM Conference on Economics and Computation, pages 565–582. ACM, 2015.

Minou Catharina Anselma Olde Keizer. Condition-based maintenance for complex systems: Coordinating maintenance and logistics planning for the process industries. PhD thesis, University of Groningen, 2016.

Ian Osband and Benjamin Van Roy. Posterior sampling for reinforcement learning without episodes. CoRR, abs/1608.02731, 2016.

Yi Ouyang, Mukul Gagrani, Ashutosh Nayyar, and Rahul Jain. Learning unknown markov decision processes: A thompson sampling approach. In NIPS, pages 1333–1342, 2017.
M. Petrik, M. Ghavamzadeh, and Y. Chow. Safe policy improvement by minimizing robust baseline regret. In Advances in Neural Information Processing Systems, pages 2298–2306, 2016.

Matteo Pirotta, Marcello Restelli, and Luca Bascetta. Adaptive step-size for policy gradient methods. In NIPS, pages 1394–1402, 2013a.

Matteo Pirotta, Marcello Restelli, Alessio Pecorino, and Daniele Calandriello. Safe policy iteration. In ICML (3), volume 28 of JMLR Workshop and Conference Proceedings, pages 307–315. JMLR.org, 2013b.

Martin L. Puterman. Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons, Inc., New York, NY, USA, 1994. ISBN 0471619779.

Thiago D. Simão and Matthijs T. J. Spaan. Safe policy improvement with baseline bootstrapping in factored environments. In AAI, pages 4967–4974. AAAI Press, 2019.

A. Swaminathan and T. Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In Proceedings of The 32nd International Conference on Machine Learning, 2015.

P. Thomas, G. Theocharous, and M. Ghavamzadeh. High confidence off-policy evaluation. In Proceedings of the Twenty-Ninth Conference on Artificial Intelligence, 2015a.

P. Thomas, G. Theocharous, and M. Ghavamzadeh. High confidence policy improvement. In Proceedings of the Thirty-Second International Conference on Machine Learning, pages 2380–2388, 2015b.

Yifan Wu, Roshan Shariff, Tor Lattimore, and Csaba Szepesvári. Conservative bandits. In ICML, volume 48 of JMLR Workshop and Conference Proceedings, pages 1254–1262. JMLR.org, 2016.

Lin F. Yang and Mengdi Wang. Reinforcement leaning in feature space: Matrix bandit, kernels, and regret bound. CoRR, abs/1905.10389, 2019.

Andrea Zanette and Emma Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In ICML, volume 97 of Proceedings of Machine Learning Research, pages 7304–7312. PMLR, 2019.
Input: Operator $\mathcal{L} : \mathbb{R}^S \to \mathbb{R}^S$ and accuracy $\epsilon > 0$
Set $v_0 = 0$, $v_1 = \mathcal{L}v_0$, $n = 0$

1. While $sp(v_{n+1} - v_n) > \epsilon$ do
   (a) $n = n + 1$
   (b) $v_{n+1} = \mathcal{L}v_n$
2. Return: $g_n = \frac{1}{2}(\text{max}_s \{v_{n+1}(s) - v_n(s)\} + \text{min}_s \{v_{n+1}(s) - v_n(s)\})$ and $v_n$

Figure 3: EVI.

A Policy Evaluation with Uncertainties

Consider a bounded parameter MDP $\mathcal{M}$ defined by a compact set $B_r(s,a) \subseteq [0,r_{\text{max}}]$ and $B_p(s,a) \in \Delta_S$:

$$\mathcal{M} = \{M = (S,A,r,p), r(s,a) \in B_r(s,a), p(\cdot|s,a) \in B_p(s,a), \forall (s,a) \in S \times A\}$$ (16)

In this paper, we consider confidence sets $B_r$ and $B_p$ that are polytopes. We are interested in building a pessimistic (robust) estimate of the performance of a policy $\pi \in \Pi^{SD}$ in $\mathcal{M}$. This robust optimization problem can be written as:

$$g^\pi := \inf_{M \in \mathcal{M}} \{g^\pi(M)\}$$ (17)

where $g^\pi(M)$ is the gain of policy $\pi$ in the MDP $M$. Lemma 5 shows that there exists a solution to this problem that can be computed using EVI when the set $\mathcal{M}$ contains an ergodic MDP.

We recall that any bounded parameter MDP admits an equivalent representation as an extended MDP [Jaksch et al., 2010] with identical state space $S$ but compact action space. For a deterministic policy $\pi \in \Pi^{SD}$, the extended (pessimistic) Bellman operator $\mathcal{L}_\pi$ is defined as:

$$\forall v \in \mathbb{R}^S, \forall s \in S, \quad \mathcal{L}_\pi v(s) := \min_{r \in B_r(s,\pi(s))} r + \min_{p \in B_p(s,\pi(s))} \{p^Tv\}$$ (18)

Lemma 5. Let $\mathcal{M}$ be a bounded-parameter MDP defined as in Eq. [16] such that exists an ergodic MDP $M \in \mathcal{M}$ w.h.p. Consider a policy $\pi \in \Pi^{SD}$, then:

1. There exists a tuple $(\bar{g}, \bar{h}) \in \mathbb{R} \times \mathbb{R}^S$ such that:
   $$\forall s \in S, \quad \bar{g} + \bar{h}(s) = \mathcal{L}_\pi \bar{h}(s)$$

   where $\mathcal{L}_\pi$ is the Bellman operator of the extended MDP $\mathcal{M}^+$ associated to $\mathcal{M}$ (see Eq. [18]).

2. In addition, we have the following inequalities on the pair $(\bar{g}, \bar{h})$:
   $$\bar{g} \leq g^\pi(M) \quad \text{and} \quad sp(\bar{h}) \leq \max_{\pi \in \Pi^{SD}(M)} \max_{s \neq s'} \mathbb{E}_M^\pi(\tau(s')|s) := \Upsilon < +\infty$$

   where $\mathbb{E}_M^\pi$ is the expectation of using policy $\pi$ in the MDP $M$ and $\tau(s')$ is the minimal number of steps to reach state $s'$.

Proof. Point 1. We show that this policy evaluation problem is equivalent to a planning problem in an extended MDP $\mathcal{M}^-$ with negative reward. Consider the extended MDP $\mathcal{M}^- = (S,A^-,p^-,r^-)$ such that $A^- = \{\pi(s)\} \times B_r(s,\pi(s)) \times B_p(s,\pi(s))$. For any state $s \in S$ and action $a^- = (\pi(s), r(s,\pi(s)), p(\cdot|s,\pi(s))) \in A_s$,

$$r^-(s,a^-) = -r(s,\pi(s))$$

$$p^-(\cdot|s,a^-) = p(\cdot|s,\pi(s))$$

Denote by $\mathcal{L}^-$ the optimal Bellman operator of $\mathcal{M}^-$. Since $B_r(s,\pi(s))$ and $B_p(s,\pi(s))$ are polytopes, $\mathcal{L}^-$ can be interpreted as an optimal Bellman operator with finite number of actions. A sufficient condition for the
existence of a solution of the optimality equations is that the MDP is weakly communicating \cite[Chap. 8-9]{Puterman1994}. Note that $\mathcal{M}^-$ contains the model defined by $P^\pi$, i.e., the Markov chain induced by $\pi$ in $\mathcal{M}^-$. Since $P^\pi$ is ergodic, $\mathcal{M}^-$ is at least communicating and thus $\mathcal{L}^-$ converges to a solution of the optimality equations.

Extended value iteration \cite{Jaksch2010} on $\mathcal{L}^-$ converges toward a gain and bias $(g^-, h^-)$ such that:

\[
g^- + h^-(s) = L^- h^-(s) = \max_{a \in \mathcal{A}_s} \{r^-(s, a) + p^- (\cdot|s, a)^T h^- \}
\]

By rearranging, we have that:

\[
-g^- + (-h^-)(s) = \min_{p \in B_p(s, \pi(s))} p^T (-h^-)
\]

Thus follows that $\tilde{g} = -g^-$ and $\tilde{h} = -h^-$. This shows the relationship between maximizing over policies in the extended MDP $\mathcal{M}^-$ and minimizing over the set of models induced by $\pi$.

**Point 2.** Let’s begin by bounding the span of the bias $\tilde{h}$. Thanks to Theorem 4 of \cite{Bartlett2009}, we have that the span of $\tilde{h}$ is upper-bounded by the diameter of the extended MDP $\mathcal{M}^-$, i.e:

\[
sp(\tilde{h}) \leq \max_{s \neq s'} \inf_{\pi^- \in \Pi^D(\mathcal{M}^-)} \mathbb{E}_{\pi^-} (\tau(s')|s)
\]

where $\mathbb{E}_{\pi^-}$ is the expectation of using policy $\pi^-$ in the extended MDP $\mathcal{M}^-$ and $\tau(s')$ is the hitting time of state $s'$. But let’s define the policy $\pi^*$ in the extended MDP $\mathcal{M}^-$ such that for a state $s$, it chooses the action:

\[
\pi^*(s) = (\pi(s), r^*(s, \pi(s)), p^*(\cdot|s, \pi(s)))
\]

with $r^*$ and $p^*$ the true parameter of the MDP $\mathcal{M}$, this is possible because w.h.p the MDP $\mathcal{M}^\pi \in \mathcal{M}^-$ with $\mathcal{M}^\pi$ the Markov chain induced by using policy $\pi$ in the MDP $\mathcal{M}$. Thus for any pair of states $(s, s')$:

\[
\mathbb{E}_{\pi^*} (\tau(s')|s) = \mathbb{E}_{\pi^*} (\tau(s')|s)
\]

with $\mathbb{E}_{\pi^*}$ the expectation of using policy $\pi$ in the MDP $\mathcal{M}$. Therefore:

\[
sp(\tilde{h}) \leq \max_{s \neq s'} \inf_{\pi^- \in \Pi^D(\mathcal{M}^-)} \mathbb{E}_{\pi^-} (\tau(s')|s)
\]

\[
\leq \mathbb{E}_{\pi^*} (\tau(s')|s) \leq \Upsilon := \max_{\pi \in \Pi^D(\mathcal{M})} \max_{s \neq s'} \mathbb{E}_{\pi} (\tau(s')|s)
\]

And $\Upsilon < +\infty$ because $\mathcal{M}$ is assumed to be ergodic.

Let’s show that the gain $\tilde{g}$ is a lower bound on the gain of the policy $\pi$ in the MDP $\mathcal{M}$. Indeed, because the operator $\mathcal{L}^-$ converges toward solution of the optimality equations for negative rewards, we have that, see \cite[Th. 8.4.1]{Puterman1994}:

\[
g^- \geq -g^\pi(M)
\]

because reversing the sign of the rewards in the MDP $\mathcal{M}$ changes the sign of the gain of a policy. Thus, $\tilde{g} \leq g^\pi(M)$.

As a consequence, we can use EVI on $\mathcal{L}_\pi$ to compute a solution for problem 17. EVI generates a sequence of vectors $(v_i)$ such that $v_{i+1} = \mathcal{L}_\pi v_i$ and $v_0 = 0$. If the algorithm is stopped when $sp(v_{n+1} - v_n) \leq \epsilon$ we have \cite[Sec. 8.3.1]{Puterman1994} that:

\[
|g_n - \tilde{g}| \leq \epsilon/2 \quad \text{and} \quad ||\mathcal{L}_\pi v_n - v_n - g_n||_\infty \leq \epsilon
\]

where $\epsilon = (1, \ldots, 1)$ and $g_n = \frac{1}{2}(\max_s \{v_{n+1}(s) - v_n(s)\} + \min_s \{v_{n+1}(s) - v_n(s)\})$. The following lemma shows how we can use the value produced by EVI to lower bound the expected sum of rewards under a policy $\pi$.

\footnote{We abuse of language since $\mathcal{M}^-$ is not formally a set. We should formally refer to the bounded parameter MDP associated to $\mathcal{M}^-$, i.e., built considering $B_p(s, \pi(s))$ and $B_r(s, \pi(s))$. Note that $p(\cdot|s, \pi(s)) \in B_p(s, \pi(s))$ w.h.p.}
Lemma 6. Let \((g_n, v_n)\) the values computed by EVI using \(L_\pi\) and an accuracy \(\epsilon\). Then, the cumulative reward collected by policy \(\pi\) in \(M\) after \(T\) steps can be lower bounded by:

\[
\forall y \in S, \quad \mathbb{E}_M \left[ \sum_{i=1}^{t} r_i | s_1 = y, \pi \right] \geq t(g_n - \epsilon) - sp(v_n)
\]

In addition,

\[
sp(v_n) \leq T
\]

Proof. Using the inequalities in \([19]\) we can write that:

\[
v_n(s) + g_n \leq \mathcal{L}_\pi v_n(s) = \min_{r \in B_r(s, \pi(s))} r + \min_{p \in B_p(s, \pi(s))} \{ p^T v \} + \epsilon
\]

\[
\leq r(s, \pi(s)) + p(|s, \pi(s)) v_n + \epsilon
\]

since \(r(s, \pi(s)) \in B_r(s, \pi(s))\) and \(p(|s, \pi(s)) \in B_p(s, \pi(s))\) w.h.p. By iterating this inequality, we get that for all \(t > 0\) and state \(s\):

\[
v_n(s) + tg_n \leq (t - 1)\epsilon + p^t(-s, \pi(s))^T v_n + \mathbb{E} \left[ \sum_{i=1}^{t} r_i(s_i, \pi(s_i)) | s_1 = s \right]
\]

The statement follows by noticing that

\[
sp(v_n) = \max_s v_n(s) - \min_s v_n(s) \geq p^t(-y, \pi(y))^T v_n - v_n(y) \geq \min_s v_n(s)
\]

The last statement is a direct consequence of the argument developed in section 4.3.1 of \([\text{Jaksch et al.}, 2010]\). This reasoning relies on the fact that the initial vector used in EVI is a zero span vector.

B Regret Bound for CUCRL

Lemma 7. The regret of CUCRL2 can be upper-bounded for some \(\beta > 0\), with probability at least \(1 - \frac{24}{\delta^2}\), by:

\[
R(\text{CUCRL}2, T) \leq \beta \cdot \left( R(\text{UCRL}2, T|A_T) + (g^* - g^{\pi_T}) \sum_{k \in \Lambda_T^F} T_k + \max\{r_{max}, sp(h^{\pi_T})\} \sqrt{SAT \ln(T/\delta)} \right)
\]

Proof. Recall that \(k_t = \sup \{ k > 0 : t > t_k \}\) is the episode at time \(t\) and that the regret is defined as \(R(\text{CUCRL}2, T) = \sum_{t=1}^{T} (g^* - r_t(s_t, a_t))\).

Since the baseline policy \(\pi_b\) may be stochastic, as a first step we replace the observed reward by its expectation. As done in \([\text{Fruit et al.}, 2018b]\) we use Azuma’s inequality that gives, with probability at least \(1 - \frac{\delta}{5}\):

\[
\forall T \geq 1, \quad - \sum_{t=1}^{T} r_t \leq - \sum_{t=1}^{T} \max_{a \in A} \sum_{s_t \in S} \pi_k(s_t, a) r(s_t, a) + 2r_{max} \sqrt{T \ln \left( \frac{5T}{\delta} \right)} \tag{20}
\]

We denote by \(\Lambda_T = \Lambda_{k_T} \cup \{ k_T \} \cdot 1_{(E \leq \sqrt{T} \sqrt{T})}\) the set of episodes where the algorithm played an UCRL policy. Note that we cannot directly consider \(\Lambda_{k_T}\) since the set is updated at the end of the episode and the last episode may not have ended at \(T\). Similarly we denote by \(\Lambda_T^F = \Lambda_{k_T}^F \cup \{ k_T \} \cdot 1_{(E \leq \sqrt{T} \sqrt{T})}\). Then, the regret of CUCRL2 can be
decomposed as follow:

\[
R(\text{CUCRL2}, T) = \sum_{t=1}^{T} \left( g^* - \sum_{s,a} \pi^c_{k_t}(s, a)r(s, a) \right) + 2r_{\max} \sqrt{T \ln \left( \frac{5T}{\delta} \right)}
\]

\[
= 2r_{\max} \sqrt{T \ln \left( \frac{5T}{\delta} \right)} + \sum_{k=1}^{k_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} (g^* - \pi^c_{k}(s, a_1))
\]

\[
+ \sum_{k=1}^{k_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} (g^* - \sum_{s,a} \pi^b_{k}(s, a)r(s, a))
\]

\[
:= R(\text{CUCRL2}, T|\Lambda_T)
\]

Moreover, note that the UCRL2 policy is deterministic so that \(\sum_{s,a} \pi^c_{k}(s, a)r(s, a) = r(s, a_1)\) when \(k_t \in \Lambda_T\). The second term, denoted \(R(\text{CUCRL2}, T|\Lambda_{k_T})\), is the regret suffered by UCRL2 over \(\sum_{k \in \Lambda_{k_T}} T_k\) steps. The only difference with the original analysis (Jaksch et al., 2010) is that the confidence intervals used by UCRL are updated when using the baseline policy, however it does not affect the regret of UCRL because it only means the confidence intervals used shrinks faster for some state-action pairs. We will analyze this term in Lem. 9. To decompose \(\Delta^c_k\) we can use the Bellman equations (\(g^a e = L^a h^a \sim h^a\)):

\[
\sum_{k \in \Lambda_T} \Delta^c_k = \sum_{k \in \Lambda_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} \pi_b(s, a)p(\cdot|s, a)^T h^b - h^b(s_t)
\]

\[
= \sum_{k \in \Lambda_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} \sum_{s,a} \pi_b(s, a)p(\cdot|s, a)^T h^b - h^b(s_{t+1}) + \sum_{k \in \Lambda_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} (h^b(s_{t+1}) - h^b(s_t))
\]

But, \(\Delta^c_k\) can be bounded using a telescopic sum argument and the number of episodes:

\[
\sum_{k \in \Lambda_T} \Delta^c_k = \sum_{k \in \Lambda_T} h^a(s_{t_{k+1}}) - h^a(s_{t_k}) \leq |\Lambda_T|\cdot sp(h^a)
\]

Then it is easy to see that \((\Delta^c_k)_{k,t}\) is a Martingale Difference Sequence with respect to the filtration \((\mathcal{F}_t)_{t \in \mathbb{N}}\) which is generated by all the randomness in the environment and in the algorithm up until time \(t\): \(|\Delta^c_{k,t}| \leq 2\|h^a\|_\infty \leq 2sp(h^a)\) and \(\mathbb{E}[\Delta^c_k | \mathcal{F}_t] = 0\). Thus with probability \(1 - \frac{2\delta}{\delta}\):

\[
\sum_{k \in \Lambda_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} \Delta^c_k \leq 4sp(h^a) \sqrt{T \ln \left( \frac{5T}{\delta} \right)}
\]

Therefore putting all the above together, we have that with probability at least \(1 - \frac{2\delta}{\delta}\):

\[
R(\text{CUCRL2}, T) \leq 2r_{\max} \sqrt{T \ln \left( \frac{5T}{\delta} \right)} + R(\text{CUCRL2}, T|\Lambda_T) + (g^* - g^a) \sum_{k=1}^{k_T} \sum_{t_{k+1} = t_k}^{t_{k+1}} (g^* - \pi^c_{k}(s, a_1))
\]

\[
+ sp(h^a) \left( |\Lambda_T| + 4 \sqrt{T \ln \left( \frac{5T}{\delta} \right)} \right)
\]

As shown in (Ouyang et al., 2017 Lem. 1), \(k_T \leq \sqrt{2SAT \ln(T)}\) thus we can simply write that \(|\Lambda_T| \leq \sqrt{2SAT \ln(T)}\). □

In the next lemma, we bound the total number of steps where CUCRL2 used the baseline policy.
Lemma 8. For any, \( \delta > 0 \), the total length of episodes where the baseline policy is played by CUCRL after \( T \) steps is upper-bounded with probability \( 1 - 2\delta/5 \) by:

\[
\sum_{t \in \Lambda^T_T} T_t \leq 2\sqrt{SAT \ln(T)} + \frac{16T\delta}{\alpha g_{\tau_0}} \left[(D + \Upsilon)\sqrt{SAT} + r_{\max} + \sqrt{SA\rho(h_{\tau_0})}\right] + \frac{112SA\delta}{(\alpha g_{\tau_0})^2} (1 + S(D + \Upsilon)^2)
\]

with \( T_T^\delta := \ln(\frac{5SAT}{\delta}) \) a logarithmic term in \( T \).

Proof. Let \( \tau \) be the last episode played conservatively: \( \tau = \sup\{k > 0 : k \in \Lambda^T_k\} \). At the beginning of episode \( \tau \) the conservative condition is not verified that is to say:

\[
\sum_{t \in \Lambda^T_{T-1}} T_t (g_{\pi} - g_t^- + \varepsilon_t) + sp(h_{\tau}) \left(|\Lambda^T_{T-1}| + (1 - \alpha)\right) + \sum_{t \in \Lambda^T_{T-1} \cup \{\tau\}} sp(h_t^-) + \sum_{t \in \Lambda^T_{T-1} \cup \{\tau\}} sp(h_t^-) \leq \sum_{t \in \Lambda^T_{T-1} \cup \{\tau\}} T_t g_{\pi}^T
\]

(22)

Let’s proceeding by analysing each term on the RHS of Eq. (22). First, we have that \( |\Lambda^T_{T-1}| \leq k_T \leq \sqrt{SAT \ln(T)} \), thus:

\[
sp(h_{\pi}) \left(|\Lambda^T_{T-1}| + (1 - \alpha)\right) \leq \left(\sqrt{2SAT \ln(T)} + 1\right) sp(h_{\pi})
\]

(23)

On the other hand, thanks to Lem. [6] we have:

\[
\sum_{t \in \Lambda^T_{T-1} \cup \{\tau\}} sp(h_t^-) \leq \left(|\Lambda^T_{T-1}| + 1\right) \Upsilon \leq 2\sqrt{2SAT \ln(T)} \Upsilon
\]

(24)

Before analysing \( \Delta^T_1 \), let’s bound the contribution of episode \( \tau \):

\[
(T_{T-1} + 1) \left((1 - \alpha)g_{\pi} - g_{\tau}^- + \varepsilon_{\tau}\right) \mathbb{1}_{\{(1 - \alpha)g_{\pi} \geq g_{\tau}^- + \varepsilon_{\tau}\}} \leq (1 - \alpha)g_{\pi} k_T \leq (1 - \alpha) r_{\max} \sqrt{2SAT \ln(T)}
\]

(25)

where we used the fact that for all episode \( k \), we have \( T_k \leq k \). Indeed the dynamic episode condition is such that for an episode \( k \), \( T_k \leq T_{k-1} + 1 \) thus by iterating this inequality, \( T_k \leq T_0 + k = k \). At this point using equations (22) and 25 we have:

\[
\Delta^T_1 + \left(\sqrt{2SAT \ln(T)} + 1\right) sp(h_{\pi}) + 2\sqrt{2SAT \ln(T)} \Upsilon + (1 - \alpha) r_{\max} \sqrt{2SAT \ln(T)} \geq \alpha \sum_{t \in \Lambda^T_{T-1}} T_t g_{\pi}^T
\]

Let’s finish by analysing \( \Delta^T_1 \). Let’s define the event, \( \Gamma = \left\{ \exists T > 0, \exists k \geq 1, s.t. T \not\in M_k \right\} \), by definition of \( B^T_T \) and \( B^T_{t+} \), \( \mathbb{P}(\Gamma) \leq \delta/5 \), see [Lazaric et al., 2019] App. B.2 for a complete proof. We have that on the event \( \Gamma^c \), for any \( l \in \Lambda^T_{T-1} \), \( (g_{\tau}^0, h_{\tau}^0) = \text{EVI}(L_T^\kappa, \varepsilon_l) \) is such that \( |\bar{g}_{\tau}^0 - g_{\tau}^-| \leq \varepsilon_l \) (see App. [A]) where \( \bar{g}_{\tau}^0 \) is the true gain:

\[
\bar{g}_{\tau}^0 + \bar{h}_{\tau}^0 = L^\kappa_1 h_{\tau}^0.
\]

Thus, since \( \varepsilon_l \leq r_{\max} / \sqrt{T} \):

\[
\Delta^T_1 = \sum_{t \in \Lambda^T_{T-1}} T_t (g_{\pi} - g_t^- + \varepsilon_t) \leq 2 \sum_{t \in \Lambda^T_{T-1}} T_t \varepsilon_t + \sum_{t \in \Lambda^T_{T-1}} T_t (g_{\pi} - \bar{g}_{\tau}^0)
\]

\[
\leq 4 r_{\max} \sqrt{T} + \sum_{t \in \Lambda^T_{T-1}} T_t (\bar{g}_t - \bar{g}_{\tau}^0)
\]

where \( \bar{g}_t \) is the optimistic gain at episode \( t \) (see [Lazaric et al., 2019]) thus the last inequality comes from \( g_{\pi} \leq \bar{g}^0 \leq \bar{g}_t \) for every episode \( t \). We can also define the optimistic bias at episode \( t, \bar{h}_t \), the pair \( (\bar{g}_t, \bar{h}_t) \) is such that:

\[
\forall s \in S, \quad \bar{g}_t + \bar{h}_t(s) := L^\kappa_1 h_{\tau} := \max_a \left\{ \max_{r \in B^T_{t+}(s,a)} r + \max_{p \in B^T_{t+}(s,a)} pL^\kappa h_{\tau} \right\}
\]
Recall that \( \bar{\pi}_l \in \Pi^{SD} \) is the optimistic policy at episode \( l \) and when \( l \notin \Lambda_{T-1} \), \( \pi_l = \bar{\pi}_l \). Then, by using Bellman equations:

\[
\sum_{l \in \Lambda_{T-1}} T_l (\bar{g}_l - \hat{g}^{\pi}_l) = \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} (\bar{g}_l - \hat{g}^{\pi}_l) = \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \left( \sum_{l \in \Lambda_{T-1}} \left( \sum_{r, a} r \max_{p \in \mathcal{B}_l^2(s_t, a_t)} \left( \hat{h}_l(s_t) - \bar{h}_l(s_t) - \hat{h}^{\pi}_l(s_t) + \max_{p \in \mathcal{B}_l^2(s_t, a_t)} \hat{h}^{\pi}_l(s_t) \right) \right) \right)
\]

At this point, we need to explicitly define the concentration inequality used to construct the confidence sets \( \mathcal{B}_l^1 \) and \( \mathcal{B}_l^p \). For every \((s, a) \in \mathcal{S} \times \mathcal{A}\), we define \( \beta^\ell_l(s, a) \) such that:

\[
\forall l \geq 1, \quad \mathcal{B}_l^1(s, a) \subset [\bar{\gamma}_l(s, a) - \beta^\ell_l(s, a), \bar{\gamma}_l(s, a) + \beta^\ell_l(s, a)]
\]

where \( \bar{\gamma}_l(s, a) \) is the empirical average of the reward received when visiting the state-action pairs \((s, a)\) at the beginning of episode \( l \). For every \((s, a) \in \mathcal{S} \times \mathcal{A}\), we define \( \beta^p_l(s, a) \) as:

\[
\mathcal{B}_l^p(s, a) = \{ p \in \Delta_{\mathcal{S}} : \| p(\cdot | s, a) - \hat{p}_l(\cdot | s, a) \|_1 \leq \beta^p_l(s, a) \}
\]

with \( \hat{p}_l \) is the empirical average of the observed transitions. Choosing those \( \beta^\ell_l \) and \( \beta^p_l \) is done thanks to concentration inequalities such that event \( \Gamma^c \) holds with high enough probability. In the following, we use:

\[
\forall s, a \quad \beta^\ell_l(s, a) = \sqrt{\frac{7SAL^5}{2 \max\{1, N_l(s, a)\}}} \quad \text{and} \quad \beta^p_l(s, a) = S \sqrt{\frac{14AL^5}{\max\{1, N_l(s, a)\}}}
\]

where \( L^5 = \ln \left( \frac{SAL^5}{2 \max\{1, N_l(s, a)\}} \right) \). For other choices of \( \beta^\ell_l \) and \( \beta^p_l \), refer to \cite{lazaric2019}. Similarly to what done in \cite{lakshmi2010} (Sec. 4.3.1 and 4.3.2), by using Hölder’s inequality and recentering the bias functions, we write:

\[
\sum_{l \in \Lambda_{T-1}} T_l (\bar{g}_l - \hat{g}^{\pi}_l) \leq |\Lambda_{T-1}| (\mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^\ell_l(s_t, a_t) + \mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^p_l(s_t, a_t)) +\]

\[
\sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \left( \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} 2 \beta^\ell_l(s_t, a_t) + \beta^p_l(s_t, a_t)(\mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^\ell_l(s_t, a_t) + \mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^p_l(s_t, a_t)) \right)
\]

\[
\leq \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} p^* (\gamma_t)(\beta^\ell_l(s_t, a_t) + \beta^p_l(s_t, a_t)) + \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} p^* (\gamma_t)(\beta^\ell_l(s_t, a_t) + \beta^p_l(s_t, a_t))
\]

\[
\leq \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} p^* (\gamma_t)(\beta^\ell_l(s_t, a_t) + \beta^p_l(s_t, a_t))
\]

To finish, the proof of this lemma, we need to bound the term (a) and (b). In the following, we use the fact that \( \mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^\ell_l(s_t, a_t) + \mathcal{S} \sum_{l \in \Lambda_{T-1}} \beta^p_l(s_t, a_t) \leq D + T \) (see Lem. 5) and again that \( |\Lambda_{T-1}| \leq k_T \leq \sqrt{2S\bar{F} \ln(T)} \). Let’s begin with (a), by definition of the radius of the confidence sets, we have:

\[
\sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \beta^\ell_l(s_t, a_t) = \sqrt{\frac{7SAL^5}{2}} \sum_{l \in \Lambda_{T-1}} \sum_{t=t_l}^{t_{l+1}-1} \sqrt{\frac{1}{\max\{1, N_l(s_t, a_t)\}}} \leq \sqrt{\frac{7SAL^5}{2}} \sum_{l=1}^{\tau-1} T_l
\]
and,

\[
\sum_{l \in \Lambda_{t-1}} \sum_{i=1}^{t_{l+1}-1} \beta^b_{\pi}(s_i, a_i) \leq S \sqrt{14L_T^\delta A} \sum_{i=1}^{\tau-1} T_i
\]

The second term (b) is easy to bound because it is a Martingale Difference Sequence with respect to the filtration generated by all the randomness in the algorithm and the environment before the current step. For any time \(t\), the \(\sigma\)-algebra generated by the history up to time \(t\) included is \(\mathcal{F}_t = \sigma(s_1, a_1, r_1, \ldots, s_t, a_t, r_t, s_{t+1})\). Define \(X_t = \mathbb{I}(\{k_t \in \Lambda_T\}(p(\cdot|s_t, \pi_{k_t}(s_t))^\top u_{k_t} - u_{k_t}(s_{t+1})) + u_{k_t} - b_t^{\pi_t} \right)\). Since \(\pi_{k_t}\) is \(\mathcal{F}_t\) measurable, \(E[X_t|\mathcal{F}_{t-1}] = 0\) and \(|X_t| \leq 2(D + \Upsilon)\). Therefore using Azuma-Hoeffding inequality, we have, with probability \(1 - \frac{\delta}{T}\) that:

\[
(b) \leq 2(D + \Upsilon)\sqrt{2TL_T^\delta}
\]

Algorithmically, it is possible to evaluate the gain of the policies played in the past episodes at the beginning of the current episode. While this will provide a better estimate for the conservative condition, it will break the MDS structure in (b) since \(b_t^{\pi_t}\) will not be measurable w.r.t. \(\mathcal{F}_t\) since it is computed with samples collected after episode \(l\). Thus putting the bound for (a) and (b) together, we have:

\[
\sum_{l \in \Lambda_{t-1}} T_l (\tilde{g}_l - g_l) \leq (D + \Upsilon) \sqrt{2SAT \ln(T)} + \sqrt{14SAL_T^\delta} \sum_{l=1}^{\tau-1} T_l + (D + \Upsilon) S \sqrt{14L_T^\delta A} \sum_{l=1}^{\tau-1} T_l
\]

That is to say,

\[
\begin{aligned}
&4r_{\text{max}} \sqrt{T} + (D + \Upsilon) \sqrt{2SAT \ln(T)} + 2(D + \Upsilon)\sqrt{2TL_T^\delta} \\
+ &\left(\sqrt{2SAT \ln(T)} + 1\right) sp(h^{\pi_T}) + \sqrt{2SAT \ln(T)} \Upsilon + (1 - \alpha) r_{\text{max}} \sqrt{2SAT \ln(T)} \right) := b_T
\end{aligned}
\]

\[
+ \sqrt{14SAL_T^\delta} \sum_{l=1}^{\tau-1} T_l + (D + \Upsilon) S \sqrt{14L_T^\delta A} \sum_{l=1}^{\tau-1} T_l \geq \alpha \sum_{l=1}^{\tau-1} T_l g^{\pi_T}
\]

Rearranging the terms and calling \(X = \sum_{l=1}^{\tau-1} T_l\), we have:

\[
\alpha g^{\pi_T} X \leq b_T + \left(\sqrt{14SAL_T^\delta} + (D + \Upsilon) S \sqrt{14L_T^\delta A}\right) \sqrt{X}
\]

We have a quadratic equation and thus:

\[
\sum_{l=1}^{\tau-1} T_l \leq \frac{2b_T}{\alpha g^{\pi_T}} + \frac{56SAL_T^\delta}{(\alpha g^{\pi_T})^2} (2 + 2S(D + \Upsilon)^2)
\]

Therefore, as \(\tau\) is the last episode where CUCRL2 played the policy \(\pi_b\), we have \(\sum_{l \in \Lambda_T} T_l = \sum_{l \in \Lambda_T} T_l\). Also, because of the condition on the length of an episode \(T_k \leq k\) for every \(k\), therefore:

\[
\sum_{l \in \Lambda_T} T_l = \sum_{l \in \Lambda_T} T_l \leq \sum_{l=1}^{\tau-1} T_l + T_\tau \leq k_T + \frac{2b_T}{\alpha g^{\pi_T}} + \frac{56SAL_T^\delta}{(\alpha g^{\pi_T})^2} (2 + 2S(D + \Upsilon)^2)
\]

The following lemma states the regret of the UCRL2 algorithm conditioned on running only the episodes in the set \(\Lambda_T\).
Lemma 9. For any $\delta > 0$, we have that after $T$, the regret of UCRL2 is upper bounded with probability at least $1 - \delta/5$ by:

$$R(UCRL2, T|\Lambda_T) \leq \beta DS \sqrt{AT \ln \left(\frac{5T}{\delta}\right)} + \beta DS^2 A \ln \left(\frac{5T}{\delta}\right)$$

with $\beta$ a numerical constant.

Proof. The same type of bound has been shown in numerous work before [Jaksch et al. (2010); Lazaric et al. (2019)], however the proof presented in those works can not be readily applied to our setting. Indeed, when the algorithm chooses to play the baseline policy for an episode, then the confidence sets used in CUCRL2 are updated for the state-action pairs encountered during this episode. However, in the classic proof for the UCRL2 algorithm the confidence sets are the same between the end of one episode and the beginning of the next one are the same. This may not be the case for CUCRL2.

Fortunately, when using the baseline policy during an episode, the confidence sets for every state-action pairs are either the same as the previous episode or are becoming tighter around the true parameters of the MDP $M^*$. Thus, proving Lemma 9 is similar to the proof presented in [Lazaric et al. (2019)], the only difference resides in bounding the sum. $\sum_{k \in \Lambda_T} \sum_{t=k}^{T} 1/\sqrt{N_k^+ (s_t, a_t)}$, which is bounded by the square root of the total number of samples in the proof of [Lazaric et al. (2019)] whereas in the case CUCRL2 it is bounded by the square root of the total number of samples gathered while exploring the set of policies plus the number of samples collected while playing the baseline policies. Therefore, at the end of the day both quantities are bounded by a constant times the square root of $T$.

A doubt someone could have is on controlling the term

$$\sum_{k=1}^{k_T} \sum_{t=k}^{T} 1/\sqrt{N_k^+ (s_t, a_t)} \leq \sqrt{\sum_{k=1}^{k_T} \sum_{t=k}^{T} N_k^+ (s_t, a_t)}.$$

For any time $t$, the $\sigma$-algebra generated by the history up to time $t$ included is $\mathcal{F}_t = \sigma(s_1, a_1, r_1, \ldots, s_t, a_t, r_t, s_{t+1})$. Define $X_t = 1_{(k_t \in \Lambda_T)} (p(\cdot|s_t, \pi_{k_t}(s_t))^T u_k - u_k(s_{t+1}))$. Since $\pi_{k_t}$ is $\mathcal{F}_t$ measurable, $E[X_t|\mathcal{F}_{t-1}] = 0$ and $|X_t| \leq 2D$. Then $(X_t, \mathcal{F}_t)$ is an MDS and nothing change compared to the analysis of UCRL2.

Finally, plugging Lemmas 8 and 9 into Lem. 7 we have that there exists a numerical constant $C_1$ such that with probability $1 - \delta$:

$$R(UCRL2, T) \leq C_1 \left( DS \sqrt{AT L^+_T} + \frac{(g^* - g^u)}{\sqrt{SAT \ln(T)}} + \frac{\sqrt{T} SAL_T^2}{\alpha q_{T}} \max\{sp(h^u), D + \gamma\} \right)$$

$$+ \frac{S^2 AL_T^2}{(\alpha q_{T})^2} (D + \gamma)^2 + \max\{r_{\text{max}}, sp(h^u)\} \sqrt{SAT \ln(T/\delta)}$$

Conservative Exploration in Reinforcement Learning
C Conservative Exploration in Finite Horizon Markov Decision Processes

In this section, we show how the conservative setting can be applied to finite horizon MDPs. Let’s consider a finite-horizon MDP (Puterman, 1994, Chp. 4) $M = (S, A, p, r, H)$ with state space $S$ and action space $A$. Every state-action pair is characterized by a reward distribution with mean $r(s, a)$ and support in $[0, 1]$ and a transition distribution $p(\cdot | s, a)$ over next state. We denote by $S = |S|$ and $A = |A|$ the number of states and actions, and by $H$ the horizon of an episode. A Markov randomized decision rule $d: S \to P(A)$ maps states to distributions over actions. A policy $\pi$ is a sequence of decision rules, i.e., $\pi = (d_1, d_2, \ldots, d_H)$. We denote by $\Pi^{MR}$ (resp. $\Pi^{MD}$) the set of Markov randomized (resp. deterministic) policies. The value of a policy $\pi \in \Pi^{MR}$ is measured trough the value function

$$V_\pi^t(s) = \mathbb{E}^\pi \left[ \sum_{l=t}^H r_l(s_l, a_l) \mid s_t = s \right]$$

where the expectation is defined w.r.t. the model and policy (i.e., $a_l \sim d_l(s_l)$). This function gives the expected total reward that one could get by following policy $\pi$ starting in state $s$, at time $t$. There exists an optimal policy $\pi^* \in \Pi^{MD}$ (Puterman, 1994, Sec. 4.4) for which $V_{\pi^*}^t = V_{\pi^*}^\pi$ satisfies the optimality equations:

$$\forall t \in [H], \forall s \in S, \quad V_{\pi^*}^t(s) = \max_{a \in A} \{ r_t(s, a) + p(\cdot | s, a)^T V_{\pi^*}^{t+1} \} := L_t V_{\pi^*}^t$$

(26)

where $V_{\pi^*}^t(s) = 0$ for any state $s \in S$. The value function can be computed using backward induction (e.g., Puterman, 1994, Bertsekas, 1995) when the reward and transitions are known. Given a policy $\pi \in \Pi^{MD}$, the associated value function satisfies the evaluation equations $V_{\pi^*}^t(s) := L_t^\pi V_{\pi^*}^{t+1}(s) = r(s, d_t(s)) + p(\cdot | s, d_t(s))^T V_{\pi^*}^{t+1}$. The optimal policy is thus defined as $\pi^* = \arg\max_{\pi \in \Pi^{MD}} \{ L_t^\pi V_{\pi^*}^t \}, \forall t \in [H]$.

In the following we assume that the learning agent known $S$, $A$ and $r_{max}$, while the reward and dynamics are unknown and need to be estimated online. Given a finite number of episode $K$, we evaluate the performance of a learning algorithm $\mathfrak{A}$ by its cumulative regret

$$R(\mathfrak{A}, K) = \sum_{k=1}^K V_{\pi_k^*}^t(s_{k,1}) - V_{\pi_k}^t(s_{k,1})$$

where $\pi_k$ is the policy executed by the algorithm at episode $k$.

Conservative Condition Designing a conservative condition, in this setting is much easier than in the average reward case as evaluating a policy can be done through the value function which gives an estimation of the expected reward over an episode. Thus, we can use this evaluation of a policy to use in place of rewards in the bandits condition. Formally, denote by $\pi_b \in \Pi^{MR}$ the baseline policy and assume that $V_{\pi_b}^t$ is known. In general, this assumption is not restrictive since the baseline performance can be estimated from historical data. Given a conservative level $\alpha \in (0, 1)$, we define the conservative condition as:

$$\forall k \in [K], \quad \sum_{l=1}^k V_{\pi_l}^t(s_{l,1}) \geq (1 - \alpha) \sum_{l=1}^k V_{\pi_b}^t(s_{l,1}) \quad \text{w.h.p}$$

(27)

where $\pi_l$ is the policy executed by the algorithm at episode $l$ and $s_{l,1}$ is the starting state of episode $l$ before policy $\pi_l$ is chosen. The initial state can be chosen arbitrarily but should be revealed at the beginning of each episode. Note that this condition is random due the choice of the policies ($\pi_l$) and also because of the starting states thus the condition is required to hold with high probability.

Note that Eq. 27 requires to evaluated the performance of policy $\pi_l$ on the true (unknown) MDP. In order derive a practical condition, we need to construct an estimate of $V_{\pi_l}^t$. In order to be conservative, we are interesting in deriving a lower bound on the value function of a generic policy $\pi$ which can be used in Eq. 27.

Pessimistic value function estimate. We recall that OFU algorithms (e.g., UCB-VI and EULER) builds uncertainties around the rewards and dynamics that are used to perform an optimistic planning. Formally, denote by $\hat{p}_k(\cdot | s, a)$ and $\hat{r}_k(s, a)$ the empirical transitions and rewards at episode $k$. Then, with high probability

$$|r(s, a) - \hat{r}_k(s, a)| \leq \beta^R_k(s, a) \quad \text{and} \quad |p(\cdot | s, a) - \hat{p}_k(\cdot | s, a)|^T v| \leq \beta^P_k(s, a)$$

(26)
for all \((s,a) \in S \times A \) and \(v \in [0,H]^S\). This uncertainties are used to compute an exploration bonus \(b_k(s,a) = \beta_k^v(s,a) + \beta_k^p(s,a)\) that can be used to compute an optimistic estimate of the optimal value function. Formally, at episode \(k\), optimistic backward induction (e.g., [Azar et al., 2017, Alg. 2]) computes an estimate value function \(\hat{v}_{k,h}\) such that \(\hat{v}_{k,h} = V^*_k\) for any state \(s\). The same approach can be used to compute a pessimistic estimate of the optimal value function by subtracting the exploration bonus to the reward (e.g., [Zanette and Brunskill, 2019]).

The only difference in the conservative setting is that we are interesting to compute a pessimistic estimate for a policy different from the optimal one. We thus define the pessimistic evaluation equations for any episode \(k\), step \(h\), state \(s\) and policy \(\pi \in MR\) as:

\[
V^\pi_{k,h}(s) := \frac{L^\pi_{k,h}}{V^\pi_{k,h+1}} = \sum_a \pi_{k,h}(s,a) \left( \hat{r}_{k}(s,a) - b_{k}(s,a) + \hat{p}_{k}(\cdot|s,a)^T V^\pi_{k+1,h} \right) \tag{28}
\]

with \(V^\pi_{k,H+1}(s) = 0\) for all states \(s \in S\). This value function is pessimistic (see Lem. [10]) and can be computed using backward induction with \(L^\pi_{k,h}\).

**Lemma 10.** Let \(\pi = (d_1, \ldots, d_H) \in MR\) and \((V^\pi_{k,h})_{h \in [H]}\) be the value function given by backward induction using Eq. (28) then with high probability:

\[
\forall (h, s) \in [H] \times S, \quad V^\pi_{h}(s) \geq V^\pi_{k,h}(s)
\]

**Proof.** On the event that the concentration inequalities holds, let \(\hat{r}_{k}(s,a)\) be the empirical reward at episode \(k\) and \(\hat{p}_{k}(\cdot|s,a)\) the empirical distribution over the next state from \((s,a)\) at episode \(k\). We proceed with a backward induction. At time \(H\) the statement is true. For \(h < H\) :

\[
V^\pi_{k,h}(s) - V^\pi_{h}(s) = \sum_a d_{h}(s,a) \left( \hat{r}_{k}(s,a) - b_{k}(s,a) + \hat{p}_{k}(\cdot|s,a)^T V^\pi_{k+1,h} \right) - \frac{L^\pi_{k,h}}{V^\pi_{h+1}(s)}
\]

\[
= \sum_a d_{h}(s,a) \left( \hat{r}_{k}(s,a) - r(s,a) - \beta_k^v(s,a) \right) \leq 0
\]

\[
+ \sum_a d_{h}(s,a) \left( \hat{p}_{k}(\cdot|s,a)^T V^\pi_{k+1,h} - p(\cdot|s,a)^T V^\pi_{h+1} - \beta_k^p(s,a) \right)
\]

\[
\leq \sum_a d_{h}(s,a) \left( \hat{p}_{k}(\cdot|s,a)^T V^\pi_{k+1,h} - p(\cdot|s,a)^T V^\pi_{h+1} - \beta_k^p(s,a) \right)
\]

\[
\leq \sum_a d_{h}(s,a) \left( \hat{p}_{k}(\cdot|s,a) - p(\cdot|s,a) \right) V^\pi_{h+1} - \beta_k^p(s,a) \leq 0
\]

where the first inequality is true because of the confidence intervals on the reward function and the penultimate inequality is true because of the backward induction hypothesis.

Thanks to this result, we can formulate a condition that the algorithm can check, at the beginning of episode \(k\) to decide if a policy is safe to play or not:

\[
\sum_{l \in S_{k-1} \cup \{k\} \cap \{1\}} V^\pi_l(s_{l,1}) + \sum_{l \in S_{k-1} \cap \{1\}} V^\pi_{l,1}(s_{l,1}) \geq (1 - \alpha) \sum_{l=1}^{k} V^\pi_{l,1}(s_{l,1}) \tag{29}
\]

where \(S_{k-1}\) is the set of episodes where the algorithm previously played non-conservatively, \(S_{k-1} = [k-1] \setminus S_{k-1}\) is the set of episodes played conservatively and \((\pi_l)\) is the policies that the OFU algorithm (e.g., UCB-VI) would execute without the conservative constraint.

Alg. [3] shows the generic structure of any conservative exploration algorithm for MDPs. First, it computes an optimistic policy by leveraging on an OFU algorithm and the collected history. Then it checks the conservative condition. When Eq. (29) is verified it plays the optimistic policy otherwise it plays conservatively by executing policy \(\pi_b\). This allows to build some budget for playing exploratory actions in the future.
Under this event, we have that for all episode $k$ and finally, \( R_k \) holds and the Martingale Difference Sequence concentration inequalities also hold.

Let’s define the high probability event, \( \mathcal{E} \), that is such that in this event, all the concentration inequalities holds with probability at least \( 1 - \delta \):

\[
\mathcal{E} := \bigcap_{(s,a) \in S \times A} \bigcap_{k \in [K]} \left\{ \| p(\cdot|s,a) - \hat{p}_k(\cdot|s,a) \|_1 \leq \sqrt{\frac{2S \ln (3KSA/\delta)}{\max\{1,N_k(s,a)\}}} \right\} \cap \left\{ \left| \hat{r}_k(s,a) - r(s,a) \right| \leq 2r_{\max} \sqrt{\frac{\ln (3KSA/\delta)}{\max\{1,N_k(s,a)\}}} \right\}
\]

and finally, \( \mathcal{E} := \mathcal{E}_1 \cap \mathcal{E}_2 \), then \( \mathcal{E} \) holds with probability at least \( 1 - \delta \). Indeed,

\[
P(\mathcal{E}^c) \leq \sum_{i=1}^{HK} \frac{\delta}{3HK} + \sum_{s,a} \sum_k \frac{2\delta}{3KSA} \leq \delta
\]

Under this event, we have that for all episode \( k \in S_K \):

\[
\overline{r}_{k,1}(s_{k,1}) - \underline{r}_{k,1}(s_{k,1}) \leq \sum_{h=1}^{H} \varepsilon_{k,h} + 5\beta_k^p(s_{k,h}, a_k^0(s_{k,h})) + 2\beta_k^p(s_{k,h}, d_k^0(s_{k,h}))
\]

Input: Policy \( \pi_b, \delta \in (0,1), r_{\max}, S, A, \alpha' \in (0,1), K \)

Initialization: Set \( H = 0, S_0 = \emptyset \) and \( S_0^0 = \emptyset \)

For episodes \( k = 1, 2, \ldots \) do

1. Compute optimistic policy \( \pi_k \) using any OFU algorithm on history \( H \).
2. Compute pessimistic estimate \( \underline{v}_k \) as in Eq. \( 28 \).
3. if Equation \( (29) \) not verified: then
   (a) \( \pi_k = \pi_b, S_{k+1}^c = S_k^c \cup \{k\} \) and \( S_{k+1} = S_k \)
   else:
   (a) \( S_{k+1} = S_k \cup \{k\} \) and \( S_{k+1}^c = S_k^c \).
4. for \( h = 1, \ldots, H \) do
   (a) Execute \( a_{k,h} = \pi_k(s_{k,h}) \), obtain reward \( r_{k,h} \), and observe \( s_{k,h} \).
   (b) if \( \pi_k \neq \pi_b \) then: add \((s_{k,h}, a_{k,h}, r_{k,h}, s_{k,h+1})\) to \( H \).

Figure 4: CUCB-VI algorithm.

Regret Guarantees We analyse Alg. \( 4 \) with UCB-VI. Before to introduce the upper-bound to the regret of CUCB-VI we introduce the following assumption on the baseline policy.

Assumption 3. The baseline policy \( \pi_b \in \Pi^{MR} \) is such that \( r_b := \min_s \{ V^*_1(s) \} > 0 \).

We can now state the main result:

Proposition 1. For \( \delta > 0 \), the regret of conservative UCB-VI (CUCB-VI) is upper-bounded with probability at least \( 1 - \delta \) by:

\[
R(\text{CUCB-VI}, K) \leq 2H \sqrt{SAHKL_K + 8HSA^2L_K^2 + H\sqrt{2K}L_K + 2HS\sqrt{AHKL_K}} + \frac{1}{4\max_b(\Delta_b) + \alpha_b} \left( 16H^3L_K + (200H^5S^2 + 128H^5SA)L_K^2 \right),
\]

where \( L_K = \max\{\ln (3KSA/\delta) , 1\} \) and \( \Delta_b = \min_{s \in S} \{ V^*_1(s) - V^*_{\pi_b}(s) \} \).

Proof. Let’s define the high probability event, \( \mathcal{E} \), that is such that in this event, all the concentration inequalities holds and the Martingale Difference Sequence concentration inequality also holds:
where \((\varepsilon_{k,h})_{k \in S, h \in [H]}\) is a martingale difference sequence with respect to the filtration \((\mathcal{F}_{k,h})_{k \in S, h \in [H]}\) that is generated by all the randomness before step \(h\) of episode \(k\). Indeed, for an episode \(k\), let \(\pi_k = (d^k_1, \ldots, d^k_H)\), decomposing \(\pi_k\) into successive decision rules.

\[
\tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1}) \leq \Delta_{k,1} \leq 2\beta^k_{\pi_k}(s_{k,1}) + \hat{\beta}_k \langle . | s_{k,1}, d^k_1(s_{k,1}) \rangle^T (\tau_{k,2} - \tau_{k,2}^{\pi_k} + 2\beta^k_{\pi_k}(s_{k,1}, d^k_1(s_{k,1}))
\]

Thus by defining, \(B_{k,h} := 3\beta^k_{\pi_k}(s_{k,h}, d^k_h(s_{k,h})) + 2\beta^k_{\pi_k}(s_{k,h}, d^k_h(s_{k,h}))\), we have:

\[
\tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1}) \leq B_{k,1} + (\hat{\beta}_k \langle . | s_{k,1}, d^k_1(s_{k,1}) \rangle - p(\langle . | s_{k,1}, d^k_1(s_{k,1}) \rangle \tau_{k,2} - \tau_{k,2}^{\pi_k} + 2\beta^k_{\pi_k}(s_{k,1}, d^k_1(s_{k,1}))) + B_{k,1}
\]

But let’s define \(\varepsilon_{k,h} := p(\langle . | s_{k,h}, d^k_h(s_{k,h}) \rangle \tau_{k,h} - \tau_{k,h}^{\pi_k} - (\tau_{k,h}(s_{k,h+1}) - \tau_{k,h}^{\pi_k}(s_{k,h+1}))\) then \((\varepsilon_{k,h})_{k \in [K], h \in [H]}\) is a Martingale Difference Sequence with respect to the filtration \(\mathcal{F}_{k,h}\) which is generated by all the randomness in the environment and the algorithm before step \(h\) of episode \(k\). Then, by recursion, we have:

\[
\tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1}) \leq \sum_{h=1}^{H} B_{k,h} + \varepsilon_{k,h} + 2\beta^k_{\pi_k}(s_{k,h}, d^k_h(s_{k,h}))
\]

The regret of algorithm CUCB-VI can be decomposed as:

\[
R(CUCB-VI, K) = \sum_{k \in S_K} V_1^*(s_{k,1}) - V_1^{\pi_k}(s_{k,1}) + \sum_{k \in S_K} V_1^*(s_{k,1}) - V_1^{\pi_k}(s_{k,1})
\]

\[
\leq |S_K| \Delta_b + R(UCB-VI, |S_K|)
\]

where \(\Delta_b = \max_{s \in S} V_1^*(s) - V_1^{\pi_k}(s)\). Therefore bounding the regret amounts to bound the number of episodes played conservatively. To do so, let’s consider, \(\tau\) the last episode played conservatively, then before the beginning of episode \(\tau\), the condition \(29\) is not verified and thus:

\[
\alpha \sum_{k=1}^{\tau} \sum_{k \in S_{\tau-1} \cup \{s\}} \left| V_1^{\pi_k}(s_{k,1}) - \sum_{k \in S_{\tau-1}} V_1^{\pi_k}(s_{k,1}) - \frac{\tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1})}{\tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1})}\right| = \Delta_{b,1}
\]

Thus, let’s finish this analysis by bounding \(\Delta_{b,1} = V_1^{\pi_k}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1})\) for all \(k \in S_K\). But:

\[
\Delta_{b,1} = V_1^{\pi_k}(s_{k,1}) - V_1^*(s_{k,1}) + V_1^*(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1}) \leq -\Delta_b + \tau_{k,1}(s_{k,1}) - \tau_{k,1}^{\pi_k}(s_{k,1})
\]

where \(\Delta_b := \min_s V_1^*(s) - V_1^{\pi_k}(s)\). Now, we need to bound the sum over all the non-conservative episodes of the difference between the optimistic and pessimistic value function. That is to say:

\[
\sum_{l \in S_{\tau-1}} \sum_{h=1}^{H} \beta^k_{\pi_k}(s_{k,h}, d^k_h(s_{k,h})) = \sum_{l \in S_{\tau-1}} \sum_{h=1}^{H} 2H r_{\max} \sqrt{\frac{2 \ln (3KS\Delta/\delta)}{\max(1, N_k(s_{k,h}, d^k_h(s_{k,h})))}}
\]

\[
\leq 2H r_{\max} H^2 \sqrt{2SAH|S_{\tau-1}|(1 + \ln(|S_{\tau-1}|H)) \ln (3KS\Delta/\delta)}
\]

Also:

\[
\sum_{l \in S_{\tau-1}} \sum_{h=1}^{H} \beta^k_{\pi_k}(s_{k,h}, d^k_h(s_{k,h})) = \sum_{l \in S_{\tau-1}} \sum_{h=1}^{H} \sqrt{\frac{2S \ln (3KS\Delta/\delta)}{\max(1, N_k(s_{k,h}, d^k_h(s_{k,h})))}}
\]

\[
\leq H^2 S \sqrt{2AH|S_{\tau-1}|(1 + \ln(|S_{\tau-1}|H)) \ln (3KS\Delta/\delta)}
\]

and, under the event \(E\), \(\sum_{l \in S_{\tau-1}} \sum_{h=1}^{H} \varepsilon_{k,h} \leq 2H^{3/2} \sqrt{2|S_{\tau-1}| \ln (3KH/\delta)}\). On the other hand, for the episode \(\tau\), we can only bound the difference in value function by \(H\). Finally, we have that \(\tau = 1 + |S_{\tau-1}| + |S_{\tau-1}|\) and
thus if we assume that \( r_b := \min_s V^\pi_b(s) > 0 \):

\[
\alpha r_b(|S_{\tau-1}^c|+1) \leq \frac{1}{4(\Delta_b + \alpha r_b)} \left( 16H^3 \ln \left( \frac{3KH}{\delta} \right) + \left( 200H^5S^2A + 12S^2_{\max}H^5SA \right) \times \right.
\]

\[
\left. \times (1 + \ln(HK)) \ln \left( \frac{3KSA}{\delta} \right) \right)
\]

Thus, the function on the RHS in bounded and using lemma 8 of Kazerouni et al. (2017), we have:

\[
\alpha r_b(|S_{\tau-1}^c|+1) \leq \frac{1}{4(\Delta_b + \alpha r_b)} \left( 16H^3 \ln \left( \frac{3KH}{\delta} \right) + \left( 200H^5S^2A + 12S^2_{\max}H^5SA \right) \times \right.
\]

\[
\left. \times (1 + \ln(HK)) \ln \left( \frac{3KSA}{\delta} \right) \right)
\]

But by definition, \(|S_{\tau-1}^c|+1 = |S_K^c|\). Hence the result.

**Experiments**

Finally, we end this presentation of conservativeness in finite horizon MDPs with some experiments. We consider a classic 3 \( \times \) 4 gridworld problem with one goal state, a starting state and one trap state, we set \( H = 10 \), and the reward of any action in all the state to \(-2\), the reward in the goal state to 10 and the reward of falling in the trapping state to \(-20\). We normalize the rewards to be in [0, 1]. The baseline policy is describing a path around the pit, see Fig. 5. On the two position adjacent to the goal the baseline policy is stochastic with a probability of reaching the goal of 1/2 for the position on the right of the goal and below the goal, respectively. On the last line the probability of going up or right is also uniform. Figure 6 shows the impact of the conservative constraint on the regret of UCB-VI for a conservative coefficient \( \alpha = 0.05 \). Fig. 6 also shows the constraint as a function of the time for UCB-VI and CUCB-VI that is to say: \( \sum_{t=1}^{t} V^\pi_t(s_0) - (1-\alpha) V^\pi_b(s_0) \) as a function of episode \( t \) with \( s_0 \) the starting state of the gridworld. In the first 10% episodes (i.e until episode 300) the condition was violated by UCB-VI 83% of the time.

![Figure 5: Illustration of the baseline policy. S is the starting state, X is the pit and G is the goal state.](image)

![Figure 6: Regret and Conservative Condition for the gridworld problem](image)
D Experiments

For average reward problems we consider “simplified” Bernstein confidence intervals given by:

$$\beta_k^k(s, a) = \sigma_r(s, a) \sqrt{\frac{\ln(\delta)}{N_k^*(s, a)}} + r_{\max} \sqrt{\frac{\ln(\delta)}{N_k^*(s, a)}}$$

and $$\beta_k^k(s, a, s') = \sigma_p(s, a, s') \sqrt{\frac{\ln(\delta)}{N_k^*(s, a)}} + \frac{\ln(\delta)}{N_k^*(s, a)}$$

where $$N_k^*(s, a) = \max\{1, N_k(s, a)\}$$, $$\sigma_r(s, a)$$ is the empirical standard deviation and $$\sigma_p(s, a, s') = \sqrt{\hat{p}(s'|s, a)(1 - \hat{p}(s'|s, a))}$$.  

D.1 Single-Product Stochastic Inventory Control

Maintaining inventories is necessary for any company dealing with physical products. We consider the case of single product without backlogging. The state space is the amount of products in the inventory, $$S = \{0, \ldots, M\}$$ where $$M$$ is the maximum capacity. Given the state $$s_t$$ at the beginning of the month, the manager (agent) has to decide the amount of units $$a_t$$ to order. We define $$D_t$$ to be the random demand of month $$t$$ and we assume a time-homogeneous probability distribution for the demand. The inventory at time $$t + 1$$ is given by

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

The action space is $$A_s = \{0, \ldots, M - s\}$$. As in [Puterman 1994], we assume a fixed cost $$K > 0$$ for placing orders and a variable cost $$c(a)$$ that increases with the quantity ordered: $$O(a) = \begin{cases} K + c(a) & a > 0 \\ 0 & \text{otherwise} \end{cases}$$. The cost of maintaining an inventory of $$s$$ items is defined by the nondecreasing function $$h(s)$$. If the inventory is available to meet a demand $$j$$, the agent receives a revenue of $$f(j)$$. The reward is thus defined as $$r(s_t, a_t, s_{t+1}) = -O(a_t) - h(s_t + a_t) + f(s_t + a_t - s_{t+1})$$. In the experiments, we use $$K = 4$$, $$c(x) = 2x$$, $$h(x) = x$$ and $$f(x) = 8x$$.

In all the experiments, we normalize rewards such that the support is in $$[0, 1]$$ and we use noise proportional to the reward mean: $$r_t(s, a) = (1 + c\eta_t)r(s, a)$$ where $$\eta_t \sim N(0, 1)$$ (we set $$c = 0.1$$).

D.2 Cost-Based Maintenance

The system is composed by $$N$$ components in an active redundant, parallel setting, which are subject to economic and stochastic dependence through load sharing. Each component $$j \in [N]$$ is described by its operational level $$x_j = \{0, \ldots, L\}$$. The level $$L$$ denotes that the component has failed. The deterioration process is modelled using a Poisson process. If all components have failed, the system is shut down and a penalty cost $$p$$ is paid. The replacement of a failed component cost $$c_c$$, while the same operation on an active component cost $$c_p$$ (usually $$c_c \geq c_p$$). There is also a fixed cost for maintenance $$c_m$$. At each time step, it is possible to replace simultaneously multiple components. Please refer to [Olde Keizer 2016] for a complete description of dynamics and rewards.

We terminate the analysis of CUCRL2 with a more challenging test. We consider the condition-based maintenance problem (CBM, [Olde Keizer 2016]) a multi-component system subject to structural, economic and
stochastic dependences. We report a complete description of the problem in App. D. The resulting MDP has $S = 121$ states and $A = 4$ actions. The maintenance policy is often implemented as a threshold policy based on the deterioration level. Such a threshold policy is not necessarily optimal for a system with economic dependence and redundancy. We simulate this scenario by considering a strong (almost optimal) threshold policy for CBM without economic dependence as baseline. We make it stochastic by selecting with probability 0.3 a random action. As a result we have that the optimal gain $g^* = 0.89$ while the baseline gain is $g^{\pi_b} = 0.82$. Fig. 7 shows the cumulative regret for UCRL2 and CUCRL2 with $\alpha = 0.001$. UCRL2 explores faster than CUCRL2 but violates the conservative condition 53% of times in the initial phase (up to $t = 140000$), incurring in multiple complete system failures. On the other hand, CUCRL2 never violates the conservative condition.