(More) Efficient Reinforcement Learning via Posterior Sampling

Anonymous Author(s)
Affiliation
Address
email

Abstract

Most provably efficient learning algorithms introduce optimism about poorly-understood states and actions to encourage exploration. We study an alternative approach for efficient exploration, *posterior sampling for reinforcement learning* (PSRL). This algorithm proceeds in repeated episodes of known duration. At the start of each episode, PSRL updates a prior distribution over Markov decision processes and takes one sample from this posterior. PSRL then follows the policy that is optimal for this *sample* during the episode. The algorithm is conceptually simple, computationally efficient and allows an agent to encode prior knowledge in a natural way. We establish an $\tilde{O}(\tau S \sqrt{AT})$ bound on the expected regret, where T is time, τ is the episode length and S and A are the cardinalities of the state and action spaces. This bound is one of the first for an algorithm not based on optimism and close to the state of the art for any reinforcement learning algorithm. We show through simulation that PSRL significantly outperforms existing algorithms with similar regret bounds.

1 Introduction

We consider the classical reinforcement learning problem of an agent interacting with its environment while trying to maximize total reward accumulated over time [1, 2]. The agent's environment is modeled as a Markov decision process (MDP), but the agent is uncertain about the true dynamics of the MDP. As the agent interacts with its environment, it observes the outcomes that result from previous states and actions, and learns about the system dynamics. This leads to a fundamental tradeoff: by exploring poorly-understood states and actions the agent can learn to improve future performance, but it may attain better short-run performance by exploiting its existing knowledge.

Naïve optimization using point estimates for unknown variables overstates an agent's knowledge, which can lead to premature and suboptimal exploitation. To offset this, the majority of provably efficient learning algorithms use a principle known as *optimism in the face of uncertainty* [3] to encourage exploration. In such an algorithm, each state and actions is afforded some optimism bonus such that their value to the agent is modeled to be as high as statistically plausible. The agent will then choose a policy that is optimal under this "optimistic" model of the environment. This incentivizes exploration since poorly-understood states and actions will receive a higher optimism bonus. As the agent resolves its uncertainty, the effect of optimism is reduced and the agent's behavior approaches optimality. Many authors have provided strong theoretical guarantees for optimistic algorithms [4, 5, 6, 7, 8]. In fact, almost all reinforcement learning algorithms with polynomial bounds on sample complexity employ optimism to guide exploration.

We study an alternative approach to efficient exploration, *posterior sampling*, and provide finite time bounds on regret. We model the agent's initial uncertainty over the environment through a prior distribution.[1] At the start of each *episode*, the agent chooses a new policy, which it follows for the duration of the episode. Posterior sampling for reinforcement learning (PSRL) selects this policy through two simple steps. First,

[1] For an MDP, this might explicitly take the form of a prior over transition dynamics and reward distributions.

a single instance of the environment is sampled from the posterior distribution at the start of an episode. Then, PSRL solves for and executes the policy that is optimal under the sampled environment over the episode. PSRL randomly selects policies according to the probability that they are optimal; exploration is guided by the variance of sampled policies as opposed to by optimism.

The idea of posterior sampling goes back to 1933 [9] and has been applied successfully to multi-armed bandits. In that literature, the algorithm is often referred to as *Thompson sampling* or as *probability matching*. Despite its long history, posterior sampling was largely neglected by the multi-armed bandit literature until empirical studies [10, 11] demonstrated that the algorithm could produce state of the art performance. This prompted a surge of interest, and a variety of strong theoretical guarantees are now available [12, 13, 14, 15]. Our results suggest that this method has great potential in reinforcement learning as well.

PSRL was originally introduced in the context of reinforcement learning by [16] under the name "Bayesian Dynamic Programming",[2] where it appeared primarily as a heuristic method. In reference to PSRL and other "Bayesian RL" algorithms, [17] states "little is known about these algorithms from a theoretical perspective, and it is unclear, what (if any) formal guarantees can be made for such approaches." Those Bayesian algorithms for which performance guarantees exist are guided by optimism. BOSS [18] introduces a more complicated version of PSRL that samples many MDPs, instead of just one, and then combines them into an *optimistic* environment to guide exploration. BEB [17] adds an exploration bonus to states and actions according to how infrequently they have been visited. We show here that it is not always necessary to introduce optimism via a complicated construction, and, that the simple algorithm originally proposed in [16] satisfies strong bounds itself.

Our work is motivated by several advantages of posterior sampling relative to optimistic algorithms. First, since PSRL only requires solving for an optimal policy for a single sampled MDP, it is computationally cheap. Many optimistic methods require a simultaneous optimization across a *family* of plausible environments [4, 5, 18] while those that attempt to approximate the Bayes-optimal solution directly are generally computationally intensive [18, 19, 20]. Second, the presence of an explicit prior allows an agent to incorporate known environment structure in a natural way. This is crucial for any practical application, where learning without any prior knowledge would be intractable. Finally, posterior sampling allows us to separate the *algorithm* from the *analysis*. In any optimistic algorithm, performance is greatly influenced by the manner in which optimism is implemented. Past works have designed algorithms, at least in part, to facilitate theoretical analysis. Although our analysis of posterior sampling is closely related to the analysis in [4], this worst-case bound has no impact on the algorithm's actual performance. We show in section 6 that PSRL outperforms the optimistic algorithm UCRL2 [4], a competitor with similar regret bounds.

2 Problem formulation

We consider the problem of learning to optimize a random finite horizon MDP $M = (\mathcal{S}, \mathcal{A}, R^M, P^M, \tau, \rho)$ in repeated finite episodes of interaction. $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $R_a^M(s)$ is a probability distribution over reward realized when selecting action a while in state s whose support is $[0,1]$, $P_a^M(s'|s)$ is the probability of transitioning to state s' if action a is selected while at state s, τ is the time horizon, and ρ the initial state distribution. We define the MDP and all other random variables we will consider with respect to a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We assume that $\mathcal{S}$, $\mathcal{A}$, and τ are deterministic so that the agent need not learn state and action spaces or the time horizon.

A deterministic policy μ is a function that maps each state $s \in \mathcal{S}$ and $i = 1, \ldots, \tau$ to an action $a \in \mathcal{A}$. For each MDP $M = (\mathcal{S}, \mathcal{A}, R^M, P^M, \tau, \rho)$ and policy μ, we define a value function

$$V_{\mu,i}^M(s) := \mathbb{E}_{M,\mu}\left[\sum_{j=i}^{\tau} \overline{R}_{a_j}^M(s_j)\Big| s_1 = s\right],$$

where $\overline{R}_a^M(s)$ denotes the expected reward realized when action a is selected while in state s, and the subscripts of the expectation operator indicate that $a_j = \mu(s_j, j)$, and $s_{j+1} \sim P_{a_j}^M(\cdot|s_j)$ for $j = i, \ldots, \tau$. A policy μ is said to be optimal for MDP M if $V_{\mu,i}^M(s) = \max_{\mu'} V_{\mu',i}^M(s)$ for all $s \in \mathcal{S}$ and $i = 1, \ldots, \tau$. We will associate with each MDP M a policy μ^M that is optimal for M.

[2] We alter terminology to avoid any suggestion that PSRL is a Bayes-optimal solution, or approximates one directly.

The reinforcement learning agent interacts with the MDP over episodes that begin at times $t_k = (k-1)\tau + 1$, $k = 1, 2, \ldots$. At each time t, the agent selects an action a_t, observes a scalar reward r_t, and then transitions to s_{t+1}. If an agent follows a policy μ then when in state s at time t during episode k, it selects an action $a_t = \mu(s, t - t_k)$. Let $H_t = (s_1, a_1, r_1, \ldots, s_{t-1}, a_{t-1}, r_{t-1})$ denote the history of observations made *prior* to time t. A reinforcement learning algorithm is a deterministic sequence $\{\pi_k | k = 1, 2, \ldots\}$ of functions, each mapping H_{t_k} to a probability distribution $\pi_k(H_{t_k})$ over policies. At the start of the kth episode, the algorithm samples a policy μ_k from the distribution $\pi_k(H_{t_k})$. The algorithm then selects actions $a_t = \mu_k(s_t, t - t_k)$ at times t during the kth episode.

We define the regret incurred by a reinforcement learning algorithm π up to time T to be

$$\text{Regret}(T, \pi) := \sum_{k=1}^{\lceil T/\tau \rceil} \Delta_k,$$

where Δ_k denotes regret over the kth episode, defined with respect to the true underlying MDP M^* by:

$$\Delta_k = \sum_{s \in \mathcal{S}} \rho(s) (V^{M^*}_{\mu^*,1}(s) - V^{M^*}_{\mu_k,1}(s)),$$

with $\mu^* = \mu^{M^*}$ and $\mu_k \sim \pi_k(H_{t_k})$. Note that regret is not deterministic since it can depend on the random MDP M^*, the algorithm's internal random sampling and, through the history H_{t_k}, on previous random transitions and random rewards. We will assess and compare algorithm performance in terms of regret and its expectation.

3 Posterior sampling for reinforcement learning

The use of posterior sampling for reinforcement learning (PSRL) was first proposed by [16]. PSRL begins with a prior distribution over MDPs with states $\mathcal{S}$, actions $\mathcal{A}$ and horizon τ. At the start of each kth episode, PSRL computes the posterior distribution conditioned on the history H_{t_k} available at that time. PSRL then samples an MDP M_k from this posterior distribution and follows the policy $\mu_k = \mu^{M_k}$ over episode k. This process is displayed below in an algorithm format.

Algorithm: Posterior Sampling for Reinforcement Learning (PSRL)

```
Data: Prior distribution $f$, t=1
for episodes $k = 1, 2, \ldots$ do
    sample $M_k \sim f(\cdot | H_{t_k})$
    compute $\mu_k = \mu^{M_k}$
    for timesteps $j = 1, \ldots, \tau$ do
        sample and apply $a_t = \mu_k(s_t, j)$
        observe $r_t$ and $s_{t+1}$
        $t = t + 1$
    end
end
```

The following result establishes what, to the best of our knowledge, are the first regret bounds for PSRL.

Theorem 1. *If f is the distribution of M^* then, for all $\delta > 0$, with probability at least $1 - \delta$,*

$$\text{Regret}(T, \pi^{\text{PS}}_\tau) = O\left(\tau S \sqrt{AT \log(SAT/\delta)}\right) \tag{1}$$

and

$$\mathbb{E}\left[\text{Regret}(T, \pi^{\text{PS}}_\tau) \middle| M^*\right] = O\left(\tau S \sqrt{AT \log(SAT)} + \tau \sqrt{\lceil T/\tau \rceil \log(1/\delta)}\right) \tag{2}$$

The bounds have $\tilde{O}(\tau S \sqrt{AT})$ regret, which is close to state of the art. Equation (2) demonstrates that this bound is enjoyed with extremely high confidence over the realized MDP M^*, even if δ is of order $(TSA)^{-S^2 A \tau}$. Further, we note that unlike the similar results of [4, 5] which require δ as an input at runtime, *any* application of PSRL simultaneously satisfies (1) and (2) for all $\delta > 0$.

Theorem 1 applies when the prior provided to the algorithm is consistent with the distribution of the MDP. An immediate corollary accommodates cases where there is a prior misspecification.

Corollary 1. *For all $\delta > 0$, with probability at least $1-\delta$,*

$$\text{Regret}(T, \pi_\tau^{\text{PS}}) = O\left(\tau S \sqrt{AT \log(SAT\gamma/\delta)}\right)$$

and

$$\mathbb{E}\left[\text{Regret}(T, \pi_\tau^{\text{PS}}) \middle| M^*\right] = O\left(\tau S \sqrt{AT \log(SAT)} + \tau \sqrt{\lceil T/\tau \rceil \log(\gamma/\delta)}\right),$$

where $\gamma = \sup_{\mathcal{M}} \mathbb{P}(M^* \in \mathcal{M})/f(\mathcal{M})$ represents the maximum prior misspecification.

4 True versus sampled MDP

A simple observation that is central to our analysis is that, at the start of each kth episode, M^* and M_k are identically distributed. This fact allows us to relate quantities that depend on the true, but unknown, MDP M^*, to those of the sampled MDP M_k, which is fully observed by the agent. We introduce $\sigma(H_{t_k})$ as the σ-algebra generated by the history up to t_k. Readers unfamiliar with measure theory can think of this as "all information known just before the start of period t_k". When we say that a random variable X is $\sigma(H_{t_k})$-measurable, this intuitively means that although X is random, it is deterministically known given the information contained in H_{t_k}.The following lemma is an immediate consequence of this observation [15].

Lemma 2 (Posterior Sampling)**.** *If f is the distribution of M^* then, for any $\sigma(H_{t_k})$-measurable function g,*

$$\mathbb{E}[g(M^*)|H_{t_k}] = \mathbb{E}[g(M_k)|H_{t_k}]. \tag{3}$$

Note that taking the expectation of (3) shows that $\mathbb{E}[g(M^*)] = \mathbb{E}[g(M_k)]$ through the tower property. Another key lemma that will be used several times in our analysis, is the Azuma-Hoeffding inequality, which gives a concentration guarantee for martingales with bounded differences.

Lemma 3 (Azuma-Hoeffding)**.** *If Y_n is a zero-mean martingale with almost surely bounded increments $|Y_i - Y_{i-1}| \leq C$, then for any $\delta > 0$ with probability at least $1-\delta$, $Y_n \leq C\sqrt{2n\log(1/\delta)}$.*

Recall, we have defined $\Delta_k = \sum_{s\in\mathcal{S}} \rho(s)(V_{\mu^*,1}^{M^*}(s) - V_{\mu_k,1}^{M^*}(s))$, to be the regret over period k. A significant hurdle in analyzing this equation is its dependence on the optimal policy μ^*, which we do not observe. For many reinforcement learning algorithms, there is no clean way to relate the unknown optimal policy to the states and actions the agent actually observes. The following result shows we how we can avoid this issue using Lemma 2. First, define

$$\tilde{\Delta}_k = \sum_{s\in\mathcal{S}} \rho(s)(V_{\mu_k,1}^{M_k}(s) - V_{\mu_k,1}^{M^*}(s)) \tag{4}$$

as the difference in expected value of the policy μ_k under the sampled MDP M_k, which is known, and its performance under the true MDP M^*, which is observed through the agent's experience.

Theorem 4 (Regret equivalence)**.**

$$\mathbb{E}\left[\sum_{k=1}^m \Delta_k\right] = \mathbb{E}\left[\sum_{k=1}^m \tilde{\Delta}_k\right] \tag{5}$$

and for any $\delta > 0$ with probability at least $1-\delta$,

$$\sum_{k=1}^m \Delta_k \leq \sum_{k=1}^m \tilde{\Delta}_k + \tau\sqrt{2m\log(1/\delta)} \tag{6}$$

Proof. Note, $\Delta_k - \tilde{\Delta}_k = \sum_{s\in\mathcal{S}} \rho(s)(V_{\mu^*,1}^{M^*}(s) - V_{\mu_k,1}^{M_k}(s)) \in [-\tau, \tau]$. By Lemma 2, $\mathbb{E}[\Delta_k - \tilde{\Delta}_k | H_{t_k}] = 0$. Taking expectations therefore establishes the first claim. The second claim follows by applying Lemma 3 to $\sum_{k=1}^m \Delta_k - \tilde{\Delta}_k$, which is a zero mean martingale with respect to the filtration $\{H_{t_k} : k = 1, .., m\}$. □

This result bounds the agent's regret in epsiode k by the difference between the agent's estimate $V_{\mu_k,1}^{M_k}(s_{t_k})$ of the expected reward in M_k from the policy it chooses, and the expected reward $V_{\mu_k,1}^{M^*}(s_{t_k})$ in M^*. If the agent has a poor estimate of the MDP M^*, we expect it to learn as the performance of following μ_k under M^* differs from its expectation under M_k. As more information is gathered, its performance should improve. In the next section, we formalize these ideas and give a precise bound on the regret of posterior sampling.

5 Analysis

An essential tools in our analysis will be the dynamic programming, or Bellman operator $\mathcal{T}^M_\mu$ which for any MDP $M = (\mathcal{S}, \mathcal{A}, R^M, P^M, \tau, \rho)$, stationary policy $\mu : \mathcal{S} \to \mathcal{A}$ and value function $V : \mathcal{S} \to \mathbb{R}$, is defined by

$$\mathcal{T}^M_\mu V(s) := \overline{R}^M_\mu(s, \mu) + \sum_{s' \in \mathcal{S}} P^M_{\mu(s)}(s'|s)V(s').$$

This operation returns the expected value of state s where we follow the policy μ under the laws of M, for one time step. The following lemma gives a concise form for the dynamic programming paradigm in terms of the Bellman operator.

Lemma 5 (Dynamic programming equation)**.** *For any MDP $M = (\mathcal{S}, \mathcal{A}, R^M, P^M, \tau, \rho)$ and policy $\mu : \mathcal{S} \times \{1, \ldots, \tau\} \to \mathcal{A}$, the value functions V^M_μ satisfy:*

$$V^M_{\mu,i} = \mathcal{T}^M_{\mu(\cdot,i)} V^M_{\mu,i+1} \tag{7}$$

for $i = 1 \ldots \tau$, with $V^M_{\mu,\tau+1} := 0$. In order to streamline our notation we will let $V^*_{\mu,i} := V^{M^*}_{\mu,i}$, $V^k_{\mu,i}(s) := V^{M_k}_{\mu,i}(s)$, $\mathcal{T}^k_\mu := \mathcal{T}^{M_k}_\mu$, $\mathcal{T}^*_\mu := \mathcal{T}^{M^*}_\mu$ and $P^*_\mu(\cdot|s) := P^{M^*}_{\mu(s)}(\cdot|s)$.

5.1 Rewriting regret in terms of Bellman error

We show here that, for $d_{t_k+i} := \sum_{s' \in \mathcal{S}} \{P^*_{\mu_k(\cdot,i)}(s'|s_{t_k+i})(V^*_{\mu_k,i+1} - V^k_{\mu_k,i+1})(s')\} - (V^*_{\mu_k,i+1} - V^k_{\mu_k,i+1})(s_{t_k+i})$

$$\tilde{\Delta}_k = \sum_{i=1}^{\tau} \left[(\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)}) V^k_{\mu_k,i+1}(s_{t_k+i}) \right] + \sum_{i=1}^{\tau} d_{t_k+i} \tag{8}$$

To see why (8) holds, simply apply the Dynamic programming equation inductively:

$$\begin{aligned}
(V^k_{\mu_k,1} - V^*_{\mu_k,1})(s_{t_k+1}) &= (\mathcal{T}^k_{\mu_k(\cdot,1)} V^k_{\mu_k,2} - \mathcal{T}^*_{\mu_k(\cdot,1)} V^*_{\mu_k,2})(s_{t_k+1}) \\
&= (\mathcal{T}^k_{\mu_k(\cdot,1)} - \mathcal{T}^*_{\mu_k(\cdot,1)}) V^k_{\mu_k,2}(s_{t_k+1}) + \sum_{s' \in \mathcal{S}} \{P^*_{\mu_k(\cdot,1)}(s'|s_{t_k+1})(V^*_{\mu_k,2} - V^k_{\mu_k,2})(s')\} \\
&= (\mathcal{T}^k_{\mu_k(\cdot,1)} - \mathcal{T}^*_{\mu_k(\cdot,1)}) V^k_{\mu_k,2}(s_{t_k+1}) + (V^*_{\mu_k,2} - V^k_{\mu_k,2})(s_{t_k+1}) + d_{t_k+1} \\
&= \ldots \\
&= \sum_{i=1}^{\tau} (\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)}) V^k_{\mu_k,i+1}(s_{t_k+i}) + \sum_{i=1}^{\tau} d_{t_k+i}.
\end{aligned}$$

This expresses the regret in terms two factors. The first factor is the one step *Bellman error* $\left[(\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)}) V^k_{\mu_k,i+1}(s_{t_k+i}) \right]$ under the sampled MDP M_k. Crucially, (8) depends only the Bellman error under the observed policy μ_k and the states $s_1, .., s_T$ that are actually visited over the first T periods. As these actions are sampled, we go on to show that the posterior distribution of M_k concentrates around M^* and so this term tends to zero.

The second term captures the randomness in the transitions of the true MDP M^*. In state s_t under policy μ_k, the expected value of $(V^*_{\mu_k,i+1} - V^k_{\mu_k,i+1})(s_{t_k+i})$ is exactly $\sum_{s' \in \mathcal{S}} \{P^*_{\mu_k(\cdot,i)}(s'|s_{t_k+i})(V^*_{\mu_k,i+1} - V^k_{\mu_k,i+1})(s')\}$. Hence, the term $\sum_{t=1}^{T} d_t$ is the sum of martingale differences each bounded by τ, which we will proceed to bound via the Azuma-Hoeffding inequality.

5.2 Introducing confidence sets

The last section reduced the algorithm's regret to its Bellman error plus some additive white noise. We will proceed by arguing that the sampled Bellman operator $\mathcal{T}^k_{\mu_k(\cdot,i)}$ concentrates around the true Bellman operatior $\mathcal{T}^*_{\mu_k(\cdot,i)}$ and that this noise term is subdominant. To do this, we introduce high probability confidence sets similar to those used in [4] and [5]. Let $\hat{P}^t_a(\cdot|s)$ denote the emprical distribution up period t of transitions observed after sampling (s,a), and let $\hat{R}^t_a(s)$ denote the empirical average reward. Finally, define $N_{t_k}(s,a) = \sum_{t=1}^{t_k-1} \mathbb{1}_{\{(s_t,a_t)=(s,a)\}}$ to be the number of times (s,a) was sampled prior to time t_k. Define the confidence set for episode k:

$$\mathcal{M}_k^{\delta_2} := \left\{ M : \left\| \hat{P}^t_a(\cdot|s) - P^M_a(\cdot|s) \right\|_1 \leq \beta_k^{\delta_2}(s,a) \;\&\; |\hat{R}^t_a(s) - R^M_a(s)| \leq \beta_k^{\delta_2}(s,a) \;\; \forall (s,a) \right\}$$

Where $\beta_k^{\delta_2}(s,a) := \sqrt{\frac{14S\log(2SAmt_k/\delta_2)}{\max\{1, N_{t_k}(s,a)\}}}$ is chosen conservatively so that $\mathcal{M}_k^{\delta_2}$ contains both M^* and M_k with high probability. Using that $\tilde{\Delta}_k \leq \tau$ we can decompose regret as follows:

$$\sum_{k=1}^m \tilde{\Delta}_k \leq \sum_{k=1}^m \tilde{\Delta}_k \mathbb{1}_{\{M_k, M^* \in \mathcal{M}_k^{\delta_2}\}} + \tau \sum_{k=1}^m [\mathbb{1}_{\{M_k \notin \mathcal{M}_k^{\delta_2}\}} + \mathbb{1}_{\{M^* \notin \mathcal{M}_k^{\delta_2}\}}] \tag{9}$$

When $\mathcal{M}_k^{\delta_2}$ contains both M_k, and M^*, the triangle inequality provides a bound on $\left\| P^*_a(\cdot|s) - P^k_a(\cdot|s) \right\|_1$ and $|R^*_a(s) - R^k_a(s)|$. Then, using the definition of the Bellman operators $\mathcal{T}^k_{\mu_k(\cdot,i)}$ and $\mathcal{T}^*_{\mu_k(\cdot,i)}$, it follows that in the event $\left\{ M_k, M^* \in \mathcal{M}_k^{\delta_2} \right\}$, for all $V \in \mathbb{R}^S$, stationary policies μ, and states s.

$$(\mathcal{T}^k_\mu - \mathcal{T}^*_\mu)V(s) \leq 2\beta_k^{\delta_2}(s,a)\{1 + \|V\|_\infty\} \tag{10}$$

Now, since $\mathcal{M}_k^{\delta_2}$ is $\sigma(H_{t_k})$-measureable, by Lemma 2, $\mathbb{E}[\mathbb{1}_{\{M_k \notin \mathcal{M}_k^{\delta_2}\}}|H_{t_k}] = \mathbb{E}[\mathbb{1}_{\{M^* \notin \mathcal{M}_k^{\delta_2}\}}|H_{t_k}]$. Therefore, applying the azuma-hoeffding inequality again establishes that with probability at least $1-\delta$, $\sum_{k=1}^m \mathbb{1}_{\{M_k \notin \mathcal{M}_k^{\delta_2}\}} \leq \sum_{k=1}^m \mathbb{1}_{\{M^* \notin \mathcal{M}_k^{\delta_2}\}} + \sqrt{2m\log(1/\delta)}$.

Now, combining (6),(8), and (9) shows that for any $\delta_1 > 0$, with probability at least $1-\delta_1$,

$$\sum_{k=1}^m \Delta_k \leq \sum_{k=1}^m \tilde{\Delta}_k \mathbb{1}_{\{M_k, M^* \in \mathcal{M}_k^{\delta_2}\}} + 2\tau \sum_{k=1}^m \mathbb{1}_{\{M^* \notin \mathcal{M}_k^{\delta_2}\}} + 2\tau\sqrt{2m\log(1/\delta_1)} \tag{11}$$

Lemma 17 of [4] shows[3] that $\mathbb{P}(M^* \notin \mathcal{M}_k^{\delta_2}) \leq \delta_2/m$, which implies $\mathbb{P}\left(\sum_{k=1}^m \mathbb{1}_{\{M^* \notin \mathcal{M}_k^{\delta_2}\}} = 0 | M^* \right) \geq 1 - \delta_2$. The reader should note that the only influence of the prior distribution for M^* is captured through the term δ_1. The regret has only a logarithmic dependence on prior probabilities which shows that these results are robust to prior misspecification. We will now proceed to bound the remaining terms with high probability, conditional on the true MDP M^*. In doing so, we will provide a link between a Bayesian algorithm and associated bounds which hold even in a frequentist sense.

5.3 Regret from Bellman error

We let the set $K_G = \left\{ k \leq m : M_k, M^* \in \mathcal{M}_k^{\delta_2} \right\}$ denote the set of "good episodes" in which confidence sets hold. Recall from Section 5.1 that $\tilde{\Delta}_k = \sum_{i=1}^\tau \left[(\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)}) V^k_{\mu_k, i+1}(s_{t_k+i}) + d_{t_k+i} \right]$. First, we will bound $\sum_{k=1}^m \sum_{i=1}^\tau (\mathbb{1}_{\{k\in K_G\}} d_{t_k+i})$. We know $\mathbb{E}[(\mathbb{1}_{\{k\in K_G\}} d_{t_k+i} | H_{t_k+i}, M^*, M_k] = \mathbb{1}_{\{k\in K_G\}} \mathbb{E}[d_{t_k+i}|H_{t_k+i}, M^*, M_k] = 0$, and $d_{t_k+i}\mathbb{1}_{\{k\in K_G\}} \in [-\tau, \tau]$. Therefore, this is the sum of bounded martingale differences, and by the Azuma-Hoeffding inequality,

$$\mathbb{P}\left(\left| \sum_{k=1}^m \sum_{i=1}^\tau d_{t_k+i} \mathbb{1}_{\{k\in K_G\}} \right| > 2\tau\sqrt{2T\log(1/\delta_2)} \;\middle|\; M^* \right) \leq \delta_2$$

Therefore we have that, conditional on M^*, with probability at least $1-\delta_2$,

$$\sum_{k\in K_G} \tilde{\Delta}_k \leq \min\left\{ \sum_{k\in K_G} \tilde{\Delta}_k,\, T \right\} \leq \min\left\{ \sum_{k\in K_g} \sum_{i=1}^\tau \left[(\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)}) V^k_{\mu_k,i+1}(s_{t_k+i}) \right],\; T \right\} + 2\tau\sqrt{2T\log(1/\delta_2)}.$$

[3] The logarithmic term in our confidence sets are inflated by a factor of m relative to those of [4]. This is equivalent to choosing $\delta_2 = \delta/m$ where δ is the parameter defining their confidence sets.

Equation (10) and the bound $\left\|V^k_{\mu_k,i}\right\|_\infty \le \tau$ shows that for $k \in K_G$, $(\mathcal{T}^k_{\mu_k(\cdot,i)} - \mathcal{T}^*_{\mu_k(\cdot,i)})V^k_{\mu_k,i+1}(s_{t_k+i}) \le \min\{\beta_k^{\delta_2}(s,a), \tau\}$. In the technical appendix we show as per [4],

$$\min\left\{\sum_{k\in K_G}\sum_{i=1}^{\tau}\min\left\{\beta_k^{\delta_2}(s,a),\tau\right\},\ T\right\} = O\left(\tau S\sqrt{AT\log(SAT/\delta_2)}\right) \tag{12}$$

which completes the proof. Equation (2) follows from this analysis by choosing $\delta_2 = 1/T$.

6 Simulation results

We compare performance of PSRL to UCRL2 [4], an optimistic algorithm with similar regret bounds. We use the benchmark example of *RiverSwim* [21], as well as several randomly generated MDPs. We provide results in both the episodic case, where the state is reset every $\tau = 20$ steps, as well as the setting without episodic reset.

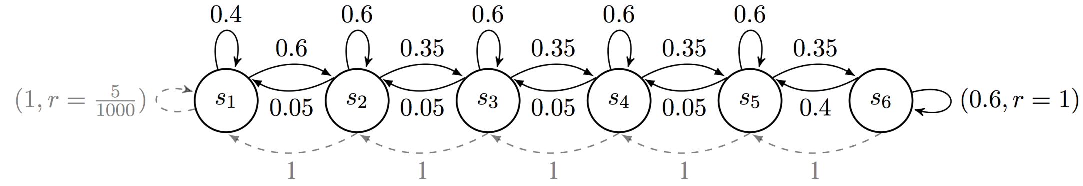


Figure 1: *RiverSwim* - continuous and dotted arrows represent the MDP under the actions “right” and “left”.

RiverSwim consists of six states arranged in a chain as shown in Figure 1. The agent begins at the far left state and at every time step has the choice to swim left or right. Swimming left (with the current) is always successful, but swimming right (against the current) often fails. The agent receives a small reward for reaching the leftmost state, but the optimal policy is to attempt to swim right and receive a much larger reward. This MDP is constructed to require efficient exploration to obtain the optimal policy. For the random MDPs we sampled 10-state, 5-action environments according to the prior.

We express our prior in terms of Dirichlet and normal-gamma distributions over the transitions and rewards respectively.[4] In both environments we perform 20 Monte Carlo simulations and compute the total regret over 10,000 time steps. We implement UCRL2 with $\delta = 0.05$ and optimize the algorithm to take account of finite episodes where appropriate. PSRL outperformed UCRL2 across every environment, as shown in Table 1. In figure 2, we show regret through time across 50 Monte Carlo simulations to 100,000 timesteps in the *RiverSwim* environment, PSRL's outperformance is quite extreme.

Table 1: Total regret in simulation. PSRL outperforms UCRL2 across different environments.

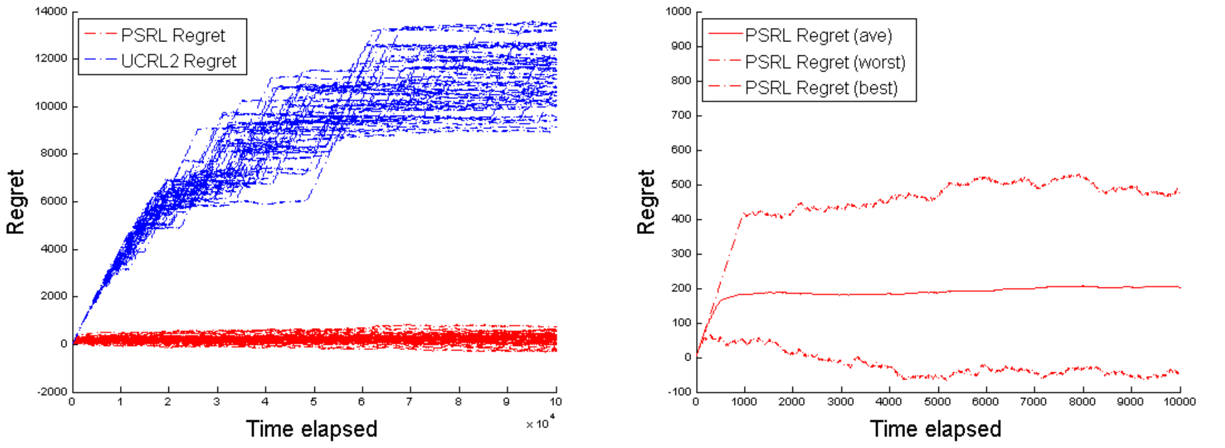
	Random MDP	*Random MDP*	*RiverSwim*	*RiverSwim*
Algorithm	*τ-episodes*	*∞-horizon*	*τ-episodes*	*∞-horizon*
PSRL	1.04×10^4	7.30×10^3	6.88×10^1	1.06×10^2
UCRL2	5.92×10^4	1.13×10^5	1.26×10^3	3.64×10^3

6.1 Learning in MDPs without episodic resets

The majority of practical problems in reinforcement learning can be mapped to repeated episodic interactions for some length τ. Even in cases where there is no actual reset of episodes, one can show that PSRL's regret is bounded against all policies which work over horizon τ or less[6]. Choosing τ large enough from prior knowledge, we can compete with the optimal policy for any given problem. Alternatively, any setting with discount factor α can be learned to arbitrary accuracy for $\tau \propto (1-\alpha)^{-1}$.

Nevertheless, for infinite-horizon and undiscounted problems artificially imposing a τ too small risks overlooking an optimal policy operating over a larger timeframe, whereas a τ too large may learn slowly. One

[4] These priors are conjugate to the multinomial and normal distribution, facilitating efficient posterior updating and sampling. We used the values $\alpha = 1/S, \mu = \sigma^2 = 1$ and pseudocount $n = 1$ for a diffuse uniform prior.



(a) PSRL outperforms UCRL2 by large margins

(b) PSRL learns quickly even with misspecified prior

Figure 2: Simulated regret on the ∞-horizon *RiverSwim* environment.

particularly appealing feature of UCRL2 [4] and REGAL [5] is that they learn an optimal timeframe from experience, and replace the factor τ in (1) by the diameter and span of the underlying MDP respectively. They both accomplish this by allowing the length of their episodes of fixed policy to vary depending upon the states and actions visited. A new episode is begun, and a policy computed, only when the total visits to any state-action pair is doubled, as opposed to fixed τ-length intervals.

We can apply this same rule for episodes to PSRL, allowing the algorithm to learn more quickly initially, while eventually exploring policies over arbitrarily large timeframes. As shown in figure 2 this implementation performs far better than UCRL2, even with a grossly misspecified prior. We see that, while UCRL2's regret grows with the worst case T-dependence of $\sqrt{T}$, PSRL performs much better than these bounds and settles upon the optimal policy after only 500 time steps on average.

At present, our analysis for deterministic τ episodes does not follow through to this infinite horizon extension, since the length episodes of fixed policy is no longer independent observations H_{t_k}. For this reason, we are unable to state an equivalent result to (6) relating optimal rewards to those observed by the agent. However, where the optimal average reward of the sampled MDP M_k is uncorrelated with episode length, we can derive results analogous to [4, 5]: that expected regret is $\tilde{O}(S\sqrt{AT\mathbb{E}[\Psi^2]})$ where Ψ is the span of bias vector of the optimal MDP M^*.

7 Conclusion

We establish *posterior sampling for reinforcement learning* not just as a heuristic, but as a provably efficient learning algorithm. We present regret bounds of $\tilde{O}(\tau S\sqrt{AT})$, which are some of the first for an algorithm not motivated by optimism and close to state of the art for any reinforcement learning algorithm. These bounds hold in expectation irrespective of prior, and with high probability unless the prior misspecification is exponentially large. PSRL is conceptually simple, computationally efficient and can easily incorporate prior knowledge; we demonstrate that PSRL performs well in simulation. Unusually for algorithms with guaranteed finite-time regret bounds, we have separated our *algorithm* from our *analysis*; as such, it may be possible for further analysis to provide even stronger guarantees on regret. We believe there is a strong case for the wider adoption of algorithms based upon posterior sampling in both theory and practice.

Looking forward, we are interested in extending our analysis from the episodic case to infinite horizon learning without episodic reset, perhaps through exponentially growing episodes of posterior sampling. We also wonder whether it is possible to match the theoretical lower bounds for regret dependence on the states actions and time elapsed of $\sqrt{SAT}$ [4] through a posterior sampling algorithm. Finally, since most problems of practical interest occur within extremely large, or even infinite, state and action spaces, we would like to explore variants of PSRL which can exploit some simplifying problem structure in these large spaces, such as linearity or factored MDPs.

References

[1] Apostolos N Burnetas and Michael N Katehakis. Optimal adaptive policies for markov decision processes. *Mathematics of Operations Research*, 22(1):222–255, 1997.

[2] Panqanamala Ramana Kumar and Pravin Varaiya. *Stochastic systems: estimation, identification and adaptive control*. Prentice-Hall, Inc., 1986.

[3] T.L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[4] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *The Journal of Machine Learning Research*, 99:1563–1600, 2010.

[5] Peter L Bartlett and Ambuj Tewari. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 35–42. AUAI Press, 2009.

[6] Ronen I Brafman and Moshe Tennenholtz. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *The Journal of Machine Learning Research*, 3:213–231, 2003.

[7] Sham Machandranath Kakade. *On the sample complexity of reinforcement learning*. PhD thesis, University of London, 2003.

[8] Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49(2-3):209–232, 2002.

[9] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[10] O. Chapelle and L. Li. An empirical evaluation of Thompson sampling. In *Neural Information Processing Systems (NIPS)*, 2011.

[11] S.L. Scott. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

[12] S. Agrawal and N. Goyal. Further optimal regret bounds for Thompson sampling. *arXiv preprint arXiv:1209.3353*, 2012.

[13] S. Agrawal and N. Goyal. Thompson sampling for contextual bandits with linear payoffs. *arXiv preprint arXiv:1209.3352*, 2012.

[14] E. Kauffmann, N. Korda, and R. Munos. Thompson sampling: an asymptotically optimal finite time analysis. In *International Conference on Algorithmic Learning Theory*, 2012.

[15] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *CoRR*, abs/1301.2609, 2013.

[16] Malcolm Strens. A Bayesian framework for reinforcement learning. In *Proceedings of the 17th International Conference on Machine Learning*, pages 943–950, 2000.

[17] J Zico Kolter and Andrew Y Ng. Near-Bayesian exploration in polynomial time. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 513–520. ACM, 2009.

[18] Tao Wang, Daniel Lizotte, Michael Bowling, and Dale Schuurmans. Bayesian sparse sampling for on-line reward optimization. In *Proceedings of the 22nd international conference on Machine learning*, pages 956–963. ACM, 2005.

[19] Arthur Guez, David Silver, and Peter Dayan. Efficient bayes-adaptive reinforcement learning using sample-based search. *arXiv preprint arXiv:1205.3109*, 2012.

[20] John Asmuth and ML Littman. Approaching bayes-optimalilty using monte-carlo tree search. In *Proc. 21st Int. Conf. Automat. Plan. Sched., Freiburg, Germany*, 2011.

[21] Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

A Bounding $\min\left\{\sum_{k\in K_G}\sum_{i=1}^{\tau}\min\left\{\beta_k^{\delta_2}(s,a),\tau\right\},\ T\right\}$

Using the definition of $\beta_k^{\delta_2}(s,a) := \sqrt{\frac{14S\log(2SAmt_k/\delta_2)}{\max\{1,N_{t_k}(s,a)\}}}$ we proceed to bound (12) in a manner similar to [4]:

$$\min\left\{\sum_{k\in K_G}\sum_{i=1}^{\tau}\min\left\{2\tau\sqrt{\frac{14S\log(2SAmt_k/\delta_2)}{\max\{1,N_{t_k}(s,a)\}}},\tau\right\},\ T\right\} = O\left(\tau S\sqrt{AT\log(SAT/\delta_2)}\right).$$

We split the sum into two parts as:

$$\begin{align} \sum_{k\in K_G}\sum_{i=1}^{\tau}\min\left\{2\tau\sqrt{\frac{14S\log(2SAmt_k/\delta_2)}{\max\{1,N_{t_k}(s,a)\}}},\tau\right\} \quad &\leq \quad \tau\sum_{k\in K_G}\sum_{i=1}^{\tau}\left[\mathbf{1}(N_{t_k}(s,a)\leq\tau)\right] \tag{13}\\ &+ \quad 2\tau\sum_{k\in K_G}\sum_{i=1}^{\tau}\left[\mathbf{1}(N_{t_k}(s,a)>\tau)\sqrt{\frac{14S\log(2SAT^2/\delta_2)}{\max\{1,N_{t_k}(s_t,a_t)\}}}\right] \tag{14} \end{align}$$

Now, the consider the event $(s_t,a_t)=(s,a)$ and $(N_{t_k}(s,a)\leq\tau)$. This can happen fewer than 2τ times per state action pair. Therefore, $\tau\sum_{k\in K_G}\sum_{i=1}^{\tau}\mathbf{1}(N_{t_k}(s,a)\leq\tau)\leq 2\tau^2SA$

Now, suppose $N_{t_k}(s,a)>\tau$. Then for any $t\in\{t_k,..,t_{k+1}-1\}$, $N_t(s,a)+1\leq N_{t_k}(s,a)+\tau\leq 2N_{t_k}(s,a)$. Therefore

$$\sum_{k=1}^{m}\sum_{t=t_k}^{t_{k+1}-1}\sqrt{\frac{\mathbf{1}(N_{t_k}(s_t,a_t)>\tau)}{N_{t_k}(s_t,a_t)}}\leq\sum_{k=1}^{m}\sum_{t=t_k}^{t_{k+1}-1}\sqrt{\frac{2}{N_t(s_t,a_t)+1}}=\sqrt{2}\sum_{t=1}^{T}(N_t(s_t,a_t)+1)^{-1/2} \tag{15}$$

Now, we can rewrite this sum as:

$$\sum_{t=1}^{T}(N_t(s_t,a_t)+1)^{-1/2}=\sum_{s,a}\sum_{j=1}^{N_{T+1}(s,a)}j^{-1/2}\leq\sum_{s,a}\int\limits_{x=0}^{N_{T+1}(s,a)}x^{-1/2}dx=2\sum_{s,a}\sqrt{N_{T+1}(s,a)}$$

By the Cauchy-Shwartz inequality, $\sum_{s,a}\sqrt{N_{T+1}(s,a)}\leq\sqrt{SA\sum_{s,a}N_{T+1}(s,a)}=\sqrt{SAT}$. Hence, we have shown that there is a numerical constant c such that

$$\begin{align} \min\left\{\sum_{k\in K_G}\sum_{i=1}^{\tau}\min\left\{2\tau\sqrt{\frac{14S\log(2SAmt_k/\delta_2)}{\max\{1,N_{t_k}(s,a)\}}},\tau\right\},\ T\right\} \quad &\leq \quad \min\left\{2\tau^2SA+c\tau\sqrt{SAT\log(SAT^2)},\ T\right\} \tag{16}\\ &\leq \quad \min\left\{2\tau^2SA,\ T\right\}+c\tau\sqrt{SAT\log(SAT^2)} \tag{17}\\ &\leq \quad \sqrt{2\tau^2SAT}+c\tau\sqrt{SAT\log(SAT^2)}. \tag{18} \end{align}$$

Which completes the proof.