Near-Bayesian Exploration in Polynomial Time

J. Zico Kolter KOLTER@CS.STANFORD.EDU
Andrew Y. Ng ANG@CS.STANFORD.EDU
Computer Science Department, Stanford University, CA 94305

Abstract

We consider the exploration/exploitation problem in reinforcement learning (RL). The Bayesian approach to model-based RL offers an elegant solution to this problem, by considering a distribution over possible models and acting to maximize expected reward; unfortunately, the Bayesian solution is intractable for all but very restricted cases. In this paper we present a simple algorithm, and prove that with high probability it is able to perform ϵ-close to the true (intractable) optimal Bayesian policy after some small (polynomial in quantities describing the system) number of time steps. The algorithm and analysis are motivated by the so-called PAC-MDP approach, and extend such results into the setting of Bayesian RL. In this setting, we show that we can achieve lower sample complexity bounds than existing algorithms, while using an exploration strategy that is much greedier than the (extremely cautious) exploration of PAC-MDP algorithms.

1. Introduction

An agent acting in an unknown environment must consider the well-known *exploration/exploitation trade-off*: the trade-off between maximizing rewards based on the current knowledge of the system (exploiting) and acting so as to gain additional information about the system (exploring). The Bayesian approach to model-based reinforcement learning (RL) offers a very elegant solution to this dilemma. Under the Bayesian approach, one maintains a *distribution* over possible models and simply acts to maximize the expected future reward; this objective trades off very naturally between exploration and exploitation. Unfortunately, except for very restricted cases, computing the full Bayesian policy is intractable. This has lead to numerous approximation methods, but, to the best of our knowledge, there has been little work on providing any formal guarantees for such algorithms.

In this paper we present a simple, greedy approximation algorithm, and show that is is able to perform nearly as well as the (intractable) optimal Bayesian policy after executing a "small" (polynomial is quantities describing the system) number of steps. The algorithm and analysis are motivated by the so-called PAC-MDP approach, typified by algorithms such as E^3 and $R_{\max}$, but extend this paradigm to the setting of Bayesian RL. We show that by considering optimality with respect to the optimal Bayesian policy, we can both achieve lower sample complexity than existing algorithms, and use an exploration approach that is far greedier than the (extremely cautious) exploration required by *any* PAC-MDP algorithm. Indeed, our analysis also shows that both our greedy algorithm and the true Bayesian policy are *not* PAC-MDP.

The remainder of the paper is organized as follows. In Section 2 we describe our setting formally and review the Bayesian and PAC-MDP approaches to exploration. In Section 3 we present our greedy approximation algorithm, and then state and discuss the theoretical guarantees for this method. In Section 4 we prove these results, present brief simulation results in Section 5 and conclude in Section 6.

2. Preliminaries

A Markov Decision Process (MDP) is a tuple $\{S, A, P, R, H\}$ where S is a set of states, A is a set of actions, $P : S \times A \times S \to \mathbb{R}_+$ is a state transition probability function, $R : S \times A \to [0, 1]$ is a (bounded) reward function, and H is a time horizon.[1] We consider an agent interacting with an MDP via

Appearing in *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009. Copyright 2009 by the author(s)/owner(s).

[1] We use the finite horizon setting for simplicity, but all results presented here easily extend to the case of infinite horizon discounted rewards.

a single continuous thread of experience, and we assume that the transition probabilities P are unknown to the agent. For simplicity of notation, we will assume that the reward is known, but this does not sacrifice generality, since an MDP with unknown (bounded) rewards and unknown transitions can be represented as an MDP with known rewards and unknown transitions by adding additional states to the system. In this work we will focus on the case of discrete states and actions.

A policy π is a mapping from states to actions. The *value* of a policy for a given state is defined as the sum of rewards over the next H time steps $V_H^\pi(s) = E\left[\sum_{t=1}^H R(s_t, \pi(s)) | s_1 = s, \pi\right]$. The value function can also be written in terms of *Bellman's equation*

$$V_H^\pi(s) = R(s, \pi(s)) + \sum_{s'} P(s'|s,a) V_{H-1}^\pi(s').$$

Finally, when the transitions of the MDP are known, we can find the optimal policy $\pi^\star$ and optimal value function $V^\star$ by solving *Bellman's optimality equation*

$$V_H^\star(s) = \max_a \left\{ R(s,a) + \sum_{s'} P(s'|s,a) V_{H-1}^\star(s') \right\}$$

where $\pi^\star(s)$ is simply the action a that maximizes the right hand side; we can apply classical algorithms such as Value Iteration or Policy Iteration to find a solution to this equation (Putterman, 2005).

2.1. Bayesian Reinforcement Learning

Of course, in the setting we consider, the transitions of the MDP are *not* known, so we must adopt different methods. The Bayesian approach to model-based RL, which has its roots in the topic of Dual Control (Fel'dbaum, 1961; Filatov & Unbehauen, 2004), explicitly represents the uncertainty over MDPs by maintaining a *belief state* b. Since we are concerned with the setting of discrete states and actions, a natural means of representing the belief is to let b consist of a set of Dirichlet distributions that describe our uncertainty over the state transitions

$$b = \{\alpha(s,a,s')\}, \quad P(s'|b,s,a) = \frac{\alpha(s,a,s')}{\alpha_0(s,a)}$$

where $\alpha_0(s,a) = \sum_{s'} \alpha(s,a,s')$. In this setting, the "state" now consists of both the system state *and* the belief state; policies and value functions now depend on both these elements. The value of a policy π for a belief and state is again given by Bellman's equation

$$\begin{aligned} V_H^\pi(b,s) &= R(s,a) + \int_{b',s'} P(b', s'|b,s,a) V_{H-1}^\pi(b',s') \\ &= R(s,a) + \int_{b',s'} P(b'|s',b,s,a) \\ &\qquad\qquad P(s'|b,s,a) V_{H-1}^\pi(b',s') \\ &= R(s,a) + \sum_{s'} P(s'|b,s,a) V_{H-1}^\pi(b',s') \end{aligned}$$

where $a = \pi(s,b)$ and where on the third line the belief b' is equal to b except that we increment $\alpha(s,a,s')$. The simplified form on this third line results from the fact that $P(b'|s',b,s,a)$ is deterministic: if we are in state s, take action a, and end up in state s', then we know how to update b (by incrementing $\alpha(s,a,s')$, as described above). This allows us to remove the integral over b', and since the states are discrete we only need to sum over s'.

We can also use the same logic to derive a Bayesian version of Bellman's optimality equation, which gives us the optimal Bayesian value function and policy

$$V_H^\star(b,s) = \max_a \left\{ R(s,a) + \sum_{s'} P(s'|b,s,a) V_{H-1}^\star(b',s') \right\}$$

and where the optimal Bayesian policy $\pi^\star(b,s)$ is just the action that maximizes the right hand side. Since this Bayesian policy plays a crucial role in the remainder of this paper, it is worth understanding the intuition behind this equation. The optimal Bayesian policy chooses actions based not only on how they will affect the next state of the system, but also based on how they will affect the next *belief state*; and, since a better knowledge of the MDP will typically lead to greater future reward, the Bayesian policy will very naturally trade off between exploring the system to gain more knowledge, and exploiting its current knowledge of the system.

Unfortunately, while the Bayesian approach provides a very elegant solution to the exploration/exploitation problem, it is typically not possible to compute the Bayesian policy exactly. Since the dimension of the belief state grows polynomially in the number of states and actions, computing the Bayesian value function using value iteration or other methods is typically not tractable.[2] This has lead to numerous methods that approximate the Bayesian exploration policy (Dearden et al., 1999; Strens, 2000; Wang et al., 2005; Poupart et al., 2006), typically by computing an approximation to the optimal value function, either by sampling or other methods. However, little is known about these

[2] One exception, where the Bayesian approach is tractable, is the domain of a k-armed bandit (i.e., an MDP with one state and k actions, where the rewards are unknown). In this case, the Bayesian approach leads to the well-known *Gittins indices* (Gittins, 1989). However, the approach does not scale analytically to multi-state MDPs.

algorithms from a theoretical perspective, and it is unclear what (if any) formal guarantees can be made for such approaches.

2.2. PAC-MDP Reinforcement Learning

An alternative approach to exploration in RL is the so-called PAC-MDP approach, exemplified by algorithms such as E^3, $R_{\max}$, and others (Kearns & Singh, 2002; Brafman & Tennenholtz, 2002; Kakade, 2003; Strehl & Littman, 2008a). Such algorithms also address the exploration/exploitation problem, but do so in a different manner. The algorithms are based on the following intuition: if an agent has observed a certain state-action pair enough times, then we can use large deviation inequalities, such as the Hoeffding bound, to ensure that the true dynamics are close to the empirical estimates. On the other hand, if we have *not* observed a state-action pair enough times, then we assume it has very high value; this will drive the agent to try out state-action pairs that we haven't observed enough times, until eventually we have a suitably accurate model of the system — this general technique is known as *optimism in the face of uncertainty.*

Although the precise formulation of the learning guarantees vary from algorithm to algorithm, using these strategies, one can prove theoretical guarantees of the following form, or similar: with high probability, the algorithm performs near optimally for all but a "small" number of time steps — where "small" here means polynomial is various quantities describing the MDP. Slightly more formally, if $\mathcal{A}_t$ denotes the policy followed by the algorithm at time t, then with probability greater than $1-\delta$,

$$V^{\mathcal{A}_t}(s_t) \geq V^\star(s_t) - \epsilon$$

for all but $m = O(\text{poly}(|S|, |A|, H, 1/\epsilon, 1/\delta))$ time steps. This statement does not indicate *when* these suboptimal steps will occur — the algorithm could act near-optimally for a long period of time before returning to sub-optimal behavior for some number of steps — which allows us to avoid issues of mixing times or ergodicity of the MDP; this precise formulation is due to Kakade (2003). Many variations and extensions of these results exist: to the case of metric MDPs (Kakade et al., 2003), factored MDPs (Kearns & Koller, 1999) to continuous linear (in state features) domains (Strehl & Littman, 2008b), to a certain class of switching linear systems (Brunskill et al., 2008), and others. However, the overall intuition behind these approaches is similar: in order to perform well, we want to explore enough that we learn an accurate model of the system.[3] While this results in very powerful guarantees, the algorithms typically require a very large amount of exploration in practice. This contrasts to the Bayesian approach, were we just want to obtain high expected reward over some finite horizon (or alternatively, an infinite discounted horizon). Intuitively, we might then expect that the Bayesian approach could act in a greedier fashion than the PAC-MDP approaches, and we will confirm and formalize this intuition in the next section. Furthermore, many issues that present challenges in the PAC-MDP framework, such as incorporating prior knowledge or dealing with correlated transitions, seemingly can be handled very naturally in the Bayesian framework.

3. A Greedy Approximation Algorithm and Theoretical Results

From the discussion above, it should be apparent that both the Bayesian and PAC-MDP approaches have advantages and drawbacks, and in this section we present an algorithm and analysis that combines elements from both frameworks. In particular, we present a simple greedy algorithm that we show to perform "nearly as well" as the full Bayesian policy, in a sense that we will formalize shortly; this is a PAC-MDP-type result, but we consider optimality with respect to the *Bayesian* policy for a given belief state, rather than the optimal policy for some fixed MDP. As we will show, this alternative definition of optimality allows us to both achieve lower sample complexity than existing PAC-MDP algorithms and use a greedier exploration method.

The algorithm we propose is itself very straightforward and similar to many previously proposed exploration heuristics. We call the algorithm Bayesian Exploration Bonus (BEB), since it chooses actions according to the current mean estimate of the state transitions plus an additional reward bonus for state-action pairs that have been observed relatively little. Specifically, the BEB algorithm, at each time step, chooses actions greedily with respect to the value function

$$\tilde{V}^\star_H(b,s) = \max_a \left\{ R(s,a) + \frac{\beta}{1+\alpha_0(s,a)} + \sum_{s'} P(s'|b,s,a)\tilde{V}^\star_{H-1}(b,s') \right\} \quad (1)$$

where β is a parameter of the algorithm that we will discuss shortly. In other words, the algorithm acts by solving the an MDP using the mean of the cur-

[3] A slightly different approach is taken by Strehl et al. (2006), where they do not build an explicit model of the system. However, the overall idea is the same, only here they want to explore enough until they obtain an accurate estimate of the state-action value function.

rent belief state for the transition probabilities, and an additional exploration bonus of $\beta/(1+\alpha_0(s,a))$ at each state. Note that the belief b is *not* updated in this equation, meaning we can solve the equation using the standard Value Iteration or Policy Iteration algorithms. To simplify the analysis that follows, we also take a common approach and cease updating the belief states after a certain number of observations, which we will describe more fully below. The following theorem gives a performance guarantee for the BEB algorithm for a suitably large value of β.

Theorem 1. *Let $\mathcal{A}_t$ denote the policy followed by the BEB algorithm (with $\beta = 2H^2$) at time t, and let s_t and b_t be the corresponding state and belief. Also suppose we stop updating the belief for a state-action pair when $\alpha_0(s,a) > 4H^3/\epsilon$. Then with probability at least $1-\delta$,*

$$V_H^{\mathcal{A}_t}(b_t, s_t) \geq V_H^\star(b_t, s_t) - \epsilon$$

— i.e., the algorithm is ϵ-close to the optimal Bayesian policy — for all but

$$m = O\left(\frac{|S||A|H^6}{\epsilon^2}\log\frac{|S||A|}{\delta}\right)$$

time steps.

In other words, BEB acts sub-optimality (where optimality is defined in the Bayesian sense), for only a polynomial number of time steps. Like the PAC-MDP results mentioned above, the theorem makes no claims about *when* these sub-optimal steps occur, and thus avoids issues of mixing times, etc.

In terms of the polynomial degree on the various quantities, this bound is tighter than the standard PAC-MDP bounds, which to the best of our knowledge have sample complexity of

$$m = \tilde{O}\left(\frac{|S|^2|A|H^6}{\epsilon^3}\right)$$

time steps.[4] Intuitively, this smaller bound results from the fact that in order to approximate the Bayesian policy, we don't need to learn the "true" dynamics of an MDP, we just need to ensure that the posterior beliefs are sufficiently peaked so that further updates cannot lead to very much additional reward. However, as mentioned above, the two theorems are not directly comparable, since we define optimality with respect to the Bayesian policy for a belief state, whereas the standard PAC-MDP framework defines optimality with respect to the optimal policy for some given MDP. Indeed, one of the chief insights of this work is that by considering the Bayesian definition of optimality, we can achieve these smaller bounds.

To gain further insight into the nature of Bayesian exploration, BEB, and the PAC-MDP approach, we compare our method to a very similar PAC-MDP algorithm known as Model Based Interval Estimation with Exploration Bonus (MBIE-EB) (Strehl & Littman, 2008a). Like BEB, this algorithm at each time step solves an MDP according to the mean estimate of the transitions, plus an exploration bonus. However, MBIE-EB uses an exploration bonus of the form

$$\frac{\beta}{\sqrt{n(s,a)}}$$

where $n(s,a)$ denotes the number of times that the state-action pair s, a has been observed; this contrasts with the BEB algorithm, which has an exploration bonus of

$$\frac{\beta}{1+n(s,a)}$$

where here $n(s,a)$ also includes the "counts" implied by the prior. Since $1/\sqrt{n}$ decays much slower than $1/n$, MBIE-EB consequently explores a great deal more than the BEB algorithm. Furthermore, this is not an artifact of the MBIE-EB algorithm alone: as we formalize in the next theorem, *any* algorithm with an exploration bonus that decays faster than $1/\sqrt{n}$ *cannot* be PAC-MDP.

Theorem 2. *Let $\mathcal{A}_t$ denote the policy followed an algorithm using any (arbitrary complex) exploration bonus that is upper bounded by*

$$\frac{\beta}{n(s,a)^p}$$

for some constant β and $p > 1/2$. Then there exists some MDP M and $\epsilon_0(\beta,p)$, such that with probability greater than $\delta_0 = 0.15$,

$$V_H^{\mathcal{A}_t}(s_t) < V_H^\star(s_t) - \epsilon_0$$

will hold for an unbounded number of time steps.

In other words, the BEB algorithm (and Bayesian exploration itself), are not PAC-MDP, and may in fact *never* find the optimal policy for some given MDP. This result is fairly intuitive: since the Bayesian algorithms are trying to maximize the reward over some finite horizon, there would be no benefit to excessive exploration if it cannot help over this horizon time.

To summarize, by considering optimality with respect to the Bayesian policy, we can obtain algorithms with

[4] The $\tilde{O}(\cdot)$ notation suppresses logarithmic factors. In addition, the model-free algorithm of (Strehl et al., 2006) obtains a bound of $\tilde{O}\left(\frac{|S||A|H^8}{\epsilon^4}\right)$, also larger than our bound.

lower sample complexity and greedier exploration policies than PAC-MDP approaches. Although the resulting algorithms may fail to find the optimal policy for certain MDPs, they are still close to optimal in the Bayesian sense.

4. Proofs of the Main Results

Before presenting the proofs in this section, we want to briefly describe their intuition. Due to space constraints, the proofs of the technical lemmas are deferred to the appendix, available in the full version of the paper (Kolter & Ng, 2009). The key condition that allows us to prove that BEB quickly performs ϵ-close to the Bayesian policy is that at every time step, BEB is *optimistic* with respect to the Bayesian Policy, and this optimism decays to zero given enough samples — that is, BEB acts according to an internal model that always overestimates the values of state-action pairs, but which approaches the true Bayesian value estimate at a rate of $O(1/n(s,a))$. The $O(1/n)$ term itself arises from the L_1 divergence between particular Dirichlet distributions.

Given these results, Theorem 1 follows by adapting standard arguments from previous PAC-MDP results. In particular, we define the "known" state-action pairs to be all those state-action pairs that have been observed more than some number of times and use the above insights to show, similar to the PAC-MDP proofs, that $V_H^{\mathcal{A}_t}(b,s)$ is close to the value of acting according to the optimal Bayesian policy, assuming the probability of leaving the known state-action set is small. Finally, we use the Hoeffding bound to show that this "escape probability" can be large only for a polynomial number of steps.

To prove Theorem 2, we use Slud's inequality (Slud, 1977) to show that any algorithm with exploration rate $O(1/n^p)$ for $p > 1/2$ may *not* be optimistic with respect to the optimal policy for a given MDP. The domain to consider here is a simple two-armed bandit, where one of the arms results in a random Bernoulli payoff, and the other results in a fixed known payoff with slightly lower mean value; we can show that with significant probability, any such exploration bonus algorithm may prefer the suboptimal arm at some point, resulting in a policy that is never near-optimal.

4.1. Proof of Theorem 1

We begin with a series of lemmas used in the proof. The first lemma states that if one has a sufficient number of counts for a Dirichlet distribution, then incrementing one of the counts won't change the probabilities very much. The proof just involves algebraic manipulation.

Lemma 3. *Consider two Dirichlet distributions with parameters $\alpha, \alpha' \in \mathbb{R}^k$. Further, suppose $\alpha'_i = \alpha_i$ for all i, except $\alpha'_j = \alpha_j + 1$. Then*

$$\sum_i \left| P(x_i|\alpha) - P(x_i|\alpha') \right| \leq \frac{2}{1+\alpha_0}.$$

Next we use this lemma to show that if we solve the MDP using the current mean of the belief state, with an additional exploration bonus of $\frac{2H^2}{1+\alpha_0(s,a)}$, this will lead to a value function that is *optimistic* with respect to the Bayesian policy. The proof involves showing that the potential "benefit" of the true Bayesian policy (i.e., how much extra reward we could obtain by updating the beliefs), is upper bounded by the exploration bonus of BEB. The proof is deferred due to space constraints, but since this result is the key to proving Theorem 1, this lemma is one of the key technical results of the paper.

Lemma 4. *Let $\tilde{V}_H^\star(b,s)$ be the value function used by BEB, defined as in (1), with $\beta = 2H^2$; that is, it is the optimal value function for the mean MDP of belief b, plus the additional reward term. Then for all s,*

$$\tilde{V}_H^\star(b,s) \geq V_H^\star(b,s).$$

Our final lemma is a trivial modification of the "Induced Inequality" used by previous PAC-MDP bounds, which extends this inequality to the Bayesian setting. The lemma states that if we execute a policy using two different rewards and belief states R, b and R', b', where $b = b'$ and $R = R'$ on a "known" set of state-action pairs K, then following a policy π will obtain similar rewards for both belief states, provided the probability of escaping from K is small. The proof mirrors that in (Strehl & Littman, 2008a).

Lemma 5. *Let b, R and b', R' be two belief states over transition probabilities and reward functions that are identical on some set of state-action pairs K — i.e., $\alpha_b(s,a,s') = \alpha_{b'}(s,a,s')$ and $R(s,a) = R'(s,a)$ for all $s, a \in K$. Let A_K be the probability that a state-action pair not in K is generated when starting from state s and following a policy π for H steps. Assuming the rewards R' are bounded in $[0, R_{\max}]$ then,*

$$V_H^\pi(R,b,s) \geq V_H^\pi(R',b',s) - HR_{\max}P(A_K)$$

where we now make explicit the dependence of the value function on the reward.

We are now ready to prove Theorem 1.

Proof. (of Theorem 1) Define $\tilde{R}$ as the reward function used by the BEB algorithm (i.e., the reward plus the exploration bonus). Let K be the set of all states that have posterior counts $\alpha_0(s,a) \geq m \equiv 4H^3/\epsilon$. Let R' be a reward function equal to R on K and equal to $\tilde{R}$ elsewhere. Furthermore, let $\tilde{\pi}$ be the policy followed by the BEB at time t— i.e., the greedy policy with respect to the current belief b_t and the reward $\tilde{R}$. Letting A_K be the event that $\tilde{\pi}$ escapes from K when starting in and acting for H steps. Then

$$V_H^{\mathcal{A}_t}(R, b_t, s_t) \geq V_H^{\tilde{\pi}}(R', b_t, s_t) - H^2 P(A_K) \tag{2}$$

by Lemma 5 where we note that we can limit the "exploration bonus" to H (i.e., use a bonus of $\min\{\frac{2H^2}{1+\alpha_0(s,a)}, H\}$), and still maintain optimism, and by noticing that $\mathcal{A}_t$ equals $\tilde{\pi}$ unless A_K occurs.

In addition, note that since R' and $\tilde{R}$ differ by at most $2H^2/m = \epsilon/(2H)$ at each state,

$$|V_H^{\tilde{\pi}}(R', b_t, s_t) - V_H^{\tilde{\pi}}(\tilde{R}, b_t, s_t)| \leq \frac{\epsilon}{2}. \tag{3}$$

Finally, we consider two cases. First, suppose that $P(A_K) > \epsilon/(2H^2)$. By the Hoeffding inequality, with probability $1-\delta$ this will occur no more than

$$O\left(\frac{m|S||A|H^3}{\epsilon}\right) = O\left(\frac{|S||A|H^6}{\epsilon^2}\right)$$

times before all the states become "known". Now suppose $P(A_K) \leq \epsilon/(2H^2)$. Then

$$\begin{aligned}
V_H^{\mathcal{A}_t}(R, b_t, s_t) &\geq V_H^{\tilde{\pi}}(R', b_t, s_t) - H^2 P(A_k) \\
&\geq V_H^{\tilde{\pi}}(R', b_t, s_t) - \frac{\epsilon}{2} \\
&\geq V_H^{\tilde{\pi}}(\tilde{R}, b_t, s_t) - \epsilon \\
&= \tilde{V}_H^{\star}(\tilde{R}, b_t, s_t) - \epsilon \\
&\geq V_H^{\star}(R, b_t, s_t) - \epsilon
\end{aligned}$$

i.e., the policy is ϵ-optimal. In this derivation the first line follows from (2), the second line follows from our assumption that $P(A_K) \leq \epsilon/(2H^2)$, the third line follows from (3), the fourth line follows from the fact that $\tilde{\pi}$ is precisely the optimal policy for $\tilde{R}, b_t$, and the last line follows from Lemma 4. □

4.2. Proof of Theorem 2

We make use of the following inequality, due to Slud (1977), which gives a lower bound on the probability of large deviations from the mean in a binomial distribution.

Lemma 6. *(Slud's inequality) Let $X_1, \ldots, X_n$ be i.i.d. Bernoulli random variables, with mean $\mu \geq 3/4$. Then*

$$P\left(\mu - \frac{1}{n}\sum_{i=1}^{n} X_i > \epsilon\right) \geq 1 - \Phi\left(\frac{\epsilon\sqrt{n}}{\sqrt{\mu(1-\mu)}}\right)$$

where $\Phi(x)$ is the cumulative distribution function of a standard Gaussian random variable

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} dx.$$

Using this lemma, we now prove the theorem.

Proof. (of Theorem 2) As mentioned in the preceding text, the scenario we consider for this proof is a two-armed bandit: action a_1 gives a Bernoulli random reward, with true payoff probability (unknown to the algorithm) of $3/4$; action a_2 gives a fixed (known) payoff of $3/4-\epsilon_0$ (we will define ϵ_0 shortly). Therefore, the optimal policy is to always pick action a_1. Since this is a setting with only one state (and therefore known transition dynamics) but unknown rewards, later in the proof we will transform this domain into one with unknown transitions but known rewards; however, the bandit formulation is more intuitive for the time being.

Since the reward for action a_2 is known, the only "exploratory" action in this case is a_1, and we let n denote the number of times that the agent has chosen action a_1, where at each trial it receives reward $r_i \in \{0, 1\}$. Let $f(n)$ be an exploration bonus for some algorithm attempting to learn this domain, and suppose it is upper bounded by

$$f(n) \leq \frac{\beta}{n^p}$$

for some $p > 1/2$. Then we can lower bound the probability that the algorithm's estimate of the reward, plus its exploration bonus, is pessimistic by more than β/n^p:

$$\begin{aligned}
&P\left(3/4 - \frac{1}{n}\sum_{i=1}^{n} r_i - f(n) \geq \frac{\beta}{n^p}\right) \\
&\geq P\left(3/4 - \frac{1}{n}\sum_{i=1}^{n} r_i \geq \frac{2\beta}{n^p}\right) \\
&\geq 1 - \Phi\left(\frac{8\beta}{\sqrt{3}n^{p-1/2}}\right)
\end{aligned}$$

where the last line follows by applying Slud's inequality. We can easily verify numerically that $1 - \Phi(1) > 0.15$, so for

$$n \geq \left(\frac{8\beta}{\sqrt{3}}\right)^{\frac{2}{2p-1}}$$

we have that with probability greater than $\delta_0 = 0.15$, the algorithm is pessimistic by more than β/n^p. Therefore, after this many steps, if we let

$$\epsilon_0(\beta, p) = \beta / \left(\frac{8\beta}{\sqrt{3}}\right)^{\frac{2}{2p-1}}$$

then with probability at least $\delta_0 = 0.15$, action a_2 will be preferred by the algorithm over action a_1. Once this occurs, the algorithm will never opt to select action a_1 (since a_2 is known, and already has no exploration bonus), so for any $\epsilon \leq \epsilon_0$, the algorithm will be more than ϵ sub-optimal for an infinite number of steps.

Finally, we also note that we can easily transform this domain to an MDP with known rewards but unknown transitions by considering a three state MDP, with transition probabilities and rewards

$$\begin{aligned} P(s_2|s_1, a_1) &= 3/4 \\ P(s_3|s_1, a_1) &= 1/4 \\ P(s_1|s_1, a_2) &= 1 \\ P(s_1|s_{2:3}, a_{1:2}) &= 1 \\ R(s_2, a_{1:2}) &= 1 \\ R(s_3, a_{1:2}) &= 0 \\ R(s_1, a_2) &= 3/4 - \epsilon_0. \end{aligned} \qquad \square$$

5. Simulated Domain

In this section we present empirical results for BEB and other algorithms on a simple chain domain from the Bayesian exploration literature (Strens, 2000; Poupart et al., 2006), shown in Figure 1. We stress that the results here are *not* intended as a rigorous evaluation of the different methods, since the domain is extremely small-scale. Nonetheless, the results illustrate that the characteristics suggested by the theory do manifest themselves in practice, at least in this small-scale setting.

Figure 2 shows the average total reward versus time step for several different algorithms. These results illustrate several points. First, the results show, as suggested by the theory, that BEB can outperform PAC-MDP algorithms (in this case, MBIE-EB), due to it's greedier exploration strategy. Second, the value of β required by Theorem 1 is typically much larger than what is best in practice. This is a common trend for such algorithms, so for both BEB and MBIE-EB we evaluated a wide range of values for β and chose the best for each (the same evaluation strategy was used by the authors of MBIE-EB (Strehl & Littman, 2008a)). Thus, while the constant factors in the theoretical results for both BEB and MBIE-EB are less important from a practical standpoint, the *rates* implied by these results — i.e., the $1/n$ vs. $1/\sqrt{n}$ exploration rates — do result in empirical differences. Finally, for this domain, the possibility that BEB converges to a sub-optimal policy is not a large concern. This is to be expected, as Theorem 2 analyzes a fairly extreme setting, and indeed implies only relatively little sub-optimality, even in the worse case.

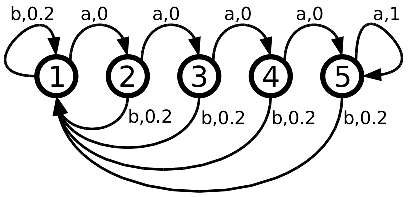

Figure 1. Simple chain domain, consisting of five states and two actions. Arrows indicate transitions and rewards for the actions, but at each time step the agent performs the opposite action as intended with probability 0.2. The agent always starts in state 1, and the horizon is $H = 6$.

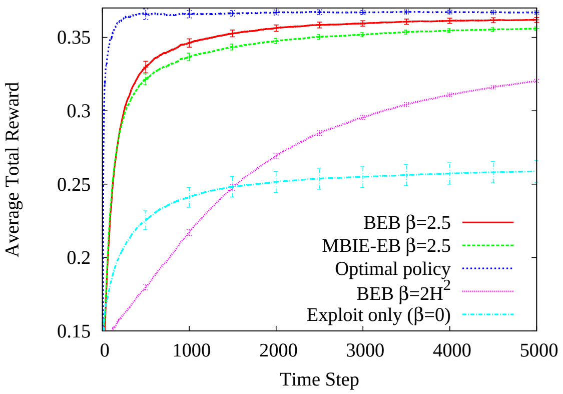

Figure 2. Performance versus time for different algorithms on the chain domain, averaged over 500 trials and shown with 95% confidence intervals.

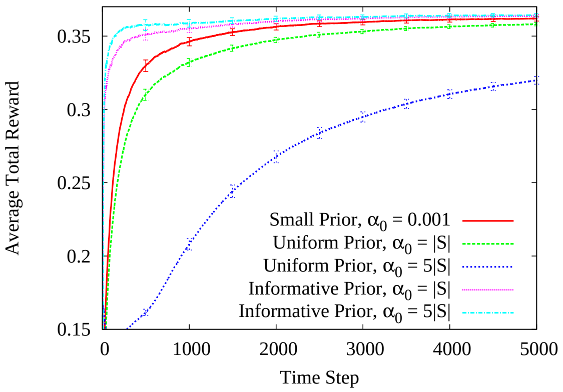

Figure 3. Performance versus time for BEB with different priors on the chain domain, averaged over 500 trials and shown with 95% confidence intervals.

We also evaluated the significance of the prior distribution on BEB. In Figure 3 we show performance for BEB with a very small prior, for uniform priors of varying strength, and for informative priors consisting of the true transition probabilities. As can be seen, BEB is fairly insensitive to either small priors, but can be negatively impacted by a large misspecified prior. These results are quite intuitive, as such as prior will greatly decrease the exploration bonus, while providing a poor model of the environment.

6. Conclusion

In this paper we presented a novel algorithm and mode of analysis that allows an agent acting in an MDP to perform ϵ-close to the (intractable) optimal Bayesian policy after a polynomial number of time steps. We bring PAC-MDP-type results to the setting of Bayesian RL, and we show that by doing so, we can both obtain lower sample complexity bounds, and use exploration techniques that are greedier than those required by any PAC-MDP algorithm.

Looking forward, the same mode of analysis that we use to derive the bounds in this paper (which involves bounding divergences between updates of the belief distributions) can also be applied to more structured domains, such as finite MDPs with correlated transitions or continuous state MDPs with smooth dynamics; it will be interesting to see how the resulting algorithms perform in such domains. An alternative means of analyzing the efficiency of reinforcement learning algorithms is the notion of regret in infinite-horizon settings (Auer & Ortner, 2007), and it remains an open question whether the ideas we present here can be extended to this infinite-horizon case. Finally, very recently Asmuth et al. (2009) have independently developed an algorithm that also combines Bayesian and PAC-MDP approaches. The actual approach is quite different — they use Bayesian sampling to achieve a PAC-MDP algorithm — but it would be very interesting to compare the algorithms.

Acknowledgments

This work was supported by the DARPA Learning Locomotion program under contract number FA8650-05-C-7261. We thank the anonymous reviews and Lihong Li for helpful comments. Zico Kolter is supported by an NSF Graduate Research Fellowship.

References

Asmuth, J., Li, L., Littman, M. L., Nouri, A., & Wingate, D. (2009). A Bayesian sampling approach to exploration in reinforcement learning. *(Preprint)*.

Auer, P., & Ortner, R. (2007). Logarithmic online regret bounds for undiscounted reinforcement learning. *Neural Information Processing Systems*.

Brafman, R. I., & Tennenholtz, M. (2002). R-MAX – a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, *3*, 213–231.

Brunskill, E., Leffler, B. R., Li, L., Littman, M. L., & Roy, N. (2008). CORL: A continuous-state offset-dynamics reinforcement learner. *Proceedings of the International Conference on Uncertainty in Artificial Intelligence*.

Dearden, R., Friedman, N., & Andre, D. (1999). Model based Bayesian exploration. *Proceedings of the International Conference on Uncertainty in Artificial Intelligence*.

Fel'dbaum, A. A. (1961). Dual control theory, parts I–IV. *Automation and Remote Control*, *21* 874–880, *21* 1033–1039, *22* 1–12, *22* 109–121.

Filatov, N., & Unbehauen, H. (2004). *Adaptive dual control: Theory and applications*. Springer.

Gittins, J. C. (1989). *Multiamred bandit allocation indices*. Wiley.

Kakade, S., Kearns, M., & Langford, J. (2003). Exploration in metric state spaces. *Proceedings of the International Conference on Machine Learning*.

Kakade, S. M. (2003). *On the sample complexity of reinforcement learning*. Doctoral dissertation, Gatsby Computational Neuroscience Unit, University College, London.

Kearns, M., & Koller, D. (1999). Efficient reinforcement learning in factored MDPs. *Proceedings of the International Joint Conference on Artificial Intelligence*.

Kearns, M., & Singh, S. (2002). Near-optimal reinforcement learning in polynomial time. *Machine Learning*, *49*.

Kolter, J. Z., & Ng, A. Y. (2009). Near-Bayesian exploration in polynomial time (full version). *Available at http://ai.stanford.edu/˜kolter*.

Poupart, P., Vlassis, N., Hoey, J., & Regan, K. (2006). An analytic solution to discrete Bayesian reinforcement learning. *Proceedings of the International Conference on Machine Learning*.

Putterman, M. L. (2005). *Markov decision processes: Discrete stochastic dynamic programming*. Wiley.

Slud, E. V. (1977). Distribution inequalities for the binomial law. *The Annals of Probability*, *5*, 404–412.

Strehl, A. L., Li, L., Wiewiora, E., Langford, J., & Littman, M. L. (2006). Pac model-free reinforcement learning. *Proceedings of the International Conference on Machine Learning*.

Strehl, A. L., & Littman, M. L. (2008a). An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, *74*, 1209–1331.

Strehl, A. L., & Littman, M. L. (2008b). Online linear regression and its application to model-based reinforcement learning. *Neural Information Processing Systems*.

Strens, M. J. (2000). A Bayesian framework for reinforcement learning. *Proceedings of the International Conference on Machine Learning*.

Wang, T., Lizotte, D., Bowling, M., & Schuurmans, D. (2005). Bayesian sparse sampling for on-line reward optimization. *Proceedings of the International Conference on Machine Learning*.

A. Technical Proofs

A.1. Proof of Lemma 3

Proof. Using definition of the Dirichlet distribution,

$$P(x_i|\alpha) = \int P(x_i|\pi)p(\pi|\alpha)d\pi = \frac{\alpha_i}{\alpha_0}$$

where $\alpha_0 \equiv \sum_i \alpha_i$. By the assumptions on α and α',

$$\begin{aligned}
&\sum_i \big| P(x_i|\alpha) - P(x_i|\alpha') \big| \\
&= \sum_{i \neq j} \left(\frac{\alpha_i}{\alpha_0} - \frac{\alpha_i}{\alpha_0 + 1} \right) + \left(\frac{\alpha_j + 1}{\alpha_0 + 1} - \frac{\alpha_j}{\alpha_0} \right) \\
&= \sum_{i \neq j} \frac{\alpha_i}{\alpha_0 + \alpha_0^2} + \frac{\alpha_0 - \alpha_j}{\alpha_0 + \alpha_0^2} \\
&\leq \sum_i \frac{\alpha_i}{\alpha_0 + \alpha_0^2} + \frac{\alpha_0}{\alpha_0 + \alpha_0^2} = \frac{2}{1 + \alpha_0}.
\end{aligned}$$

□

A.2. Proof of Lemma 4

Proof. Consider some belief and state b and s, and let b_t be the new belief formed by updating b after taking $t \leq H$ steps. Then

$$\begin{aligned}
&\tilde{V}_H^\star(b,s) - V_H^\star(b_t,s) \\
&= \max_a \Bigg\{ R(s,a) + \frac{2H^2}{1 + \alpha_0(s,a)} + \\
&\qquad \sum_{s'} P(s'|b,s,a)\tilde{V}_{H-1}^\star(b,s') \Bigg\} - \\
&\quad \max_a \Bigg\{ R(s,a) + \sum_{s'} P(s'|b_t,s,a)V_{H-1}^\star(b_{t+1},s') \Bigg\} \\
&\geq \min_a \Bigg\{ \frac{2H^2}{1 + \alpha_0(s,a)} + \sum_{s'} P(s'|b,s,a)\tilde{V}_{H-1}^\star(b,s') - \\
&\qquad \sum_{s'} P(s'|b_t,s,a)V_{H-1}^\star(b_{t+1},s') \Bigg\} \\
&\geq \min_a \Bigg\{ \frac{2H^2}{1 + \alpha_0(s,a)} - \\
&\qquad (H-1)\sum_{s'} \big| P(s|b,s,a) - P(s'|b_t,s,a) \big| + \\
&\qquad \sum_{s'} P(s'|b,s,a)\left(\tilde{V}_{H-1}^\star(b,s) - V_{H-1}^\star(b_t,s) \right) \Bigg\} \\
&\geq \min_a \left\{ \sum_{s'} P(s'|b,s,a)\left(\tilde{V}_{H-1}^\star(b,s) - V_{H-1}^\star(b_t,s) \right) \right\} \\
&\geq \min_s \left\{ \tilde{V}_{H-1}^\star(b,s) - V_{H-1}^\star(b_{t+1},s) \right\}.
\end{aligned}$$

The first line just substitutes the definitions of $\tilde{V}_H^\star$ and $V_H^\star$. In the second line we use the fact that

$$\max_x f(x) - \max_x g(x) \geq \min_x \left(f(x) - g(x) \right).$$

In the third line we use the fact that

$$\begin{aligned}
&\sum_x p(x)f(x) - \sum_x q(x)g(x) \\
&\quad \geq \sum_x p(x)(f(x) - g(x)) - \sum_x |p(x) - q(x)|g(x),
\end{aligned}$$

and note that $V_{H-1}^\star(b,s) \leq (H-1)$ for any b and s. In the fourth line we apply Lemma 3 to show that

$$\frac{2H^2}{1 + \alpha_0(s,a)} \geq (H-1)\sum_{s'} \big| P(s|b,s,a) - P(s'|b_t,s,a) \big|$$

which lets us remove these terms. In greater detail, using the triangle inequality, Lemma 3, and the fact that $t \leq H$

$$\begin{aligned}
&\sum_{s'} \big| P(s'|b_t,s,a) - P(s'|b,s,a) \big| \\
&\leq \sum_{i=1}^t \sum_{s'} \big| P(s'|b_i,s,a) - P(s'|b_{i-1},s,a) \big| \\
&\leq \sum_{i=1}^t \frac{2}{1 + \alpha_0(s,a) + i} \leq \frac{2H}{1 + \alpha_0(s,a)}.
\end{aligned}$$

Since s is arbitrary in the above derivation, we have that for any $t \leq H$,

$$\begin{aligned}
&\min_s \left\{ \tilde{V}_H^\star(b,s) - V_H^\star(b_t,s) \right\} \\
&\quad \geq \min_s \left\{ \tilde{V}_{H-1}^\star(b,s) - V_{H-1}^\star(b_{t+1},s) \right\}
\end{aligned}$$

Applying this equation repeatedly proves the desired lemma. □

A.3. Proof of Lemma 5

Proof. Consider a sequence of beliefs, states, actions, and rewards of length t, $p_t = s_1 a_1 r_1, \ldots, s_t, a_t, r_t$. Let $P(p_t)$ be the probability of this sequence under belief b with reward function R when starting in state s, and let $P'(p_t)$ be the probability of the sequence under belief b with reward function R. Let K_t be the set of sequences where all $s_1, \ldots, s_t \in K$ Then

$$\begin{aligned}
&V_H^\pi(R', b', s) - V_H^\pi(R, b, s) \\
&= \sum_{t=1}^{H} \sum_{p_t} \Big(P'(p_t) r_t(p_t) - P(p_t) r_t(p_t) \Big) \\
&= \sum_{t=1}^{H} \Bigg[\sum_{p_t \in K_t} \Big(P'(p_t) r_t(p_t) - P(p_t) r_t(p_t) \Big) + \\
&\qquad \sum_{p_t \notin K_t} \Big(P'(p_t) r_t(p_t) - P(p_t) r_t(p_t) \Big) \Bigg] \\
&= \sum_{t=1}^{H} \sum_{p_t \notin K_t} \Big(P'(p_t) r_t(p_t) - P(p_t) r_t(p_t) \Big) \\
&\leq \sum_{t=1}^{H} \sum_{p_t \notin K_t} P'(p_t) r_t(p_t) \leq H R_{\max} P(A_K)
\end{aligned}$$

where we can eliminate the terms for $p_t \in K_t$, because R, b and R', b' are identical on this set, and where the last line follows since the rewards are bounded in $[0, R_{\max}]$. □