Bounded Optimal Exploration in MDP

Kenji Kawaguchi
Massachusetts Institute of Technology
Cambridge, MA, 02139
kawaguch@mit.edu

Abstract

Within the framework of probably approximately correct Markov decision processes (PAC-MDP), much theoretical work has focused on methods to attain near optimality after a relatively long period of learning and exploration. However, practical concerns require the attainment of satisfactory behavior within a short period of time. In this paper, we relax the PAC-MDP conditions to reconcile theoretically driven exploration methods and practical needs. We propose simple algorithms for discrete and continuous state spaces, and illustrate the benefits of our proposed relaxation via theoretical analyses and numerical examples. Our algorithms also maintain anytime error bounds and average loss bounds. Our approach accommodates both Bayesian and non-Bayesian methods.

Introduction

The formulation of sequential decision making as a Markov decision process (MDP) has been successfully applied to a number of real-world problems. MDPs provide the ability to design adaptable agents that can operate effectively in uncertain environments. In many situations, the environment we wish to model has unknown aspects, and thus the agent needs to learn an MDP by interacting with the environment. In other words, the agent has to *explore* the unknown aspects of the environment to learn the MDP. A considerable amount of theoretical work on MDPs has focused on efficient exploration, and a number of principled methods have been derived with the aim of learning an MDP to obtain a near-optimal policy. For example, Kearns and Singh (2002) and Strehl and Littman (2008a) considered discrete state spaces, whereas Bernstein and Shimkin (2010) and Pazis and Parr (2013) examined continuous state spaces.

In practice, however, heuristics are still commonly used (Li 2012). The focus of theoretical work (learning a near-optimal policy within a polynomial yet long time) has apparently diverged from practical needs (learning a satisfactory policy within a reasonable time). In this paper, we modify the prevalent theoretical approach to develop theoretically driven methods that come close to practical needs.

Preliminaries

An MDP (Puterman 2004) can be represented as a tuple (S, A, R, P, γ), where S is a set of states, A is a set of actions, P is the transition probability function, R is a reward function, and γ is a discount factor. The value of policy π at state s, $V^\pi(s)$, is the cumulative (discounted) expected reward, which is given by: $V^\pi(s) = E\left[\sum_{i=0}^{\infty} \gamma^i R\left(s_i, \pi(s_i), s_{i+1}\right) \mid s_0 = s, \pi\right]$, where the expectation is over the sequence of states $s_{i+1} \sim P(S|s_i, \pi(s_i))$ for all $i \geq 0$. Using Bellman's equation, the value of the optimal policy or the optimal value, $V^*(s)$, can be written as $V^*(s) = \max_a \sum_{s'} P(s'|s,a))[R(s,a,s') + \gamma V^*(s')]$.

In many situations, the transition function P and/or the reward function R are initially unknown. Under such conditions, we often want a policy of an algorithm at time t, $\mathcal{A}_t$, to yield a value $V^{\mathcal{A}_t}(s_t)$ that is close to the optimal value $V^*(s_t)$ after some exploration. Here, s_t denotes the current state at time t. More precisely, we may want the following: for all $\epsilon > 0$ and for all $\delta = (0, 1)$, $V^{\mathcal{A}_t}(s_t) \geq V^*(s_t) - \epsilon$, with probability at least $1 - \delta$ when $t \geq \tau$, where τ is the exploration time. The algorithm with a policy$\mathcal{A}_t$ is said to be "probably approximately correct" for MDPs (PAC-MDP) (Strehl 2007) if this condition holds with τ being at most polynomial in the relevant quantities of MDPs. The notion of PAC-MDP has a strong theoretical basis and is widely applicable, avoiding the need for additional assumptions, such as reachability in state space (Jaksch, Ortner, and Auer 2010), access to a reset action (Fiechter 1994), and access to a parallel sampling oracle (Kearns and Singh 1999).

However, the PAC-MDP approach often results in an algorithm over-exploring the state space, causing a low reward per unit time for a long period of time. Accordingly, past studies that proposed PAC-MDP algorithms have rarely presented a corresponding experimental result, or have done so by tuning the free parameters, which renders the relevant algorithm no longer PAC-MDP (Strehl, Li, and Littman 2006; Kolter and Ng 2009; Sorg, Singh, and Lewis 2010). This problem was noted in (Kolter and Ng 2009; Brunskill 2012; Kawaguchi and Araya 2013). Furthermore, in many problems, it may not even be possible to guarantee $V^{\mathcal{A}_t}$ close to V^* within the agent's lifetime. Li (2012) noted that, despite the strong theoretical basis of the PAC-MDP approach,

Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16), pp. 1758–1764, 2016

heuristic-based methods remain popular in practice. This would appear to be a result of the above issues. In summary, there seems to be a dissonance between a strong theoretical approach and practical needs.

Bounded Optimal Learning

The practical limitations of the PAC-MDP approach lie in their focus on correctness without accommodating the time constraints that occur naturally in practice. To overcome the limitation, we first define the notion of *reachability in model learning*, and then relax the PAC-MDP objective based on it. For brevity, we focus on the transition model.

Reachability in Model Learning

For each state-action pair (s, a), let $M_{(s,a)}$ be a set of all transition models and $\widehat{P}_t(\cdot|s,a) \in M_{(s,a)}$ be the current model at time t (i.e., $\widehat{P}_t(\cdot|s,a) : S \to [0,\infty)$). Define $S'_{(s,a)}$ to be a set of possible future samples as $S'_{(s,a)} = \{s'|P(s'|s,a) > 0\}$. Let $f_{(s,a)} : M_{(s,a)} \times S'_{(s,a)} \to M_{(s,a)}$ represent the model update rule; $f_{(s,a)}$ maps a model (in $M_{(s,a)}$) and a new sample (in $S'_{(s,a)}$) to a corresponding new model (in $M_{(s,a)}$). We can then write $\mathcal{L} = (M, f)$ to represent a learning method of an algorithm, where $M = \cup_{(s,a)\in(S,A)} M_{(s,a)}$ and $f = \{f_{(s,a)}\}_{(s,a)\in(S,A)}$.

The set of h-reachable models, $\mathcal{M}_{\mathcal{L},t,h,(s,a)}$, is recursively defined as $\mathcal{M}_{\mathcal{L},t,h,(s,a)} = \left\{\widehat{P}' \in M_{(s,a)} | \widehat{P}' = f_{(s,a)}(\widehat{P}, s') \quad \text{for some} \quad \widehat{P} \in \mathcal{M}_{\mathcal{L},t,h-1,(s,a)} \quad \text{and} \quad s' \in S'_{(s,a)}\right\}$ with the boundary condition $\mathcal{M}_{t,0,(s,a)} = \{\widehat{P}_t(\cdot|s,a)\}$.

Intuitively, the set of h-reachable models, $\mathcal{M}_{\mathcal{L},t,h,(s,a)} \subseteq \mathcal{M}_{(s,a)}$, contains the transition models that can be obtained if the agent updates the current model at time t using any combination of h additional samples $s'_1, s'_2, \ldots, s'_h \sim P(S|s,a)$. Note that the set of h-reachable models is defined *separately for each state-action pair*. For example, $\mathcal{M}_{\mathcal{L},t,h,(s_1,a_1)}$ contains only those models that are reachable using the h additional samples drawn from $P(S|s_1,a_1)$.

We define the h-reachable optimal value $V^{d*}_{\mathcal{L},t,h}(s)$ with respect to a distance function d as

$$V^{d*}_{\mathcal{L},t,h}(s) = \max_a \sum_{s'} \widehat{P}^{d*}_{\mathcal{L},t,h}(s'|s,a)[R(s,a,s') + \gamma V^{d*}_{\mathcal{L},t,h}(s')],$$

where

$$\widehat{P}^{d*}_{\mathcal{L},t,h}(\cdot|s,a) = \underset{\widehat{P}\in\mathcal{M}_{\mathcal{L},t,h,(s,a)}}{\arg\min} \; d(\widehat{P}(\cdot|s,a), P(\cdot|s,a)).$$

Intuitively, the h-reachable optimal value, $V^{d*}_{\mathcal{L},t,h}(s)$, is the optimal value estimated with the "best" model in the set of h-reachable models (here, the term "best" is in terms of the distance function $d(\cdot,\cdot)$).

PAC in Reachable MDP

Using the concept of reachability in model learning, we define the notion of "probably approximately correct" in an h-reachable MDP (PAC-RMDP(h)). Let $\mathcal{P}(x_1, x_2, \ldots, x_n)$ be a polynomial in $x_1, x_2, \ldots, x_n$ and |MDP| be the complexity of an MDP (Li 2012).

Definition 1. (PAC-RMDP(h)) An algorithm with a policy $\mathcal{A}_t$ and a learning method $\mathcal{L}$ is PAC-RMDP(h) with respect to a distance function d if for all $\epsilon > 0$ and for all $\delta = (0,1)$,

1) there exists $\tau = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|, h))$ such that for all $t \geq \tau$,

$$V^{\mathcal{A}_t}(s_t) \geq V^{d*}_{\mathcal{L},t,h}(s_t) - \epsilon$$

with probability at least $1-\delta$, *and*

2) there exists $h^*(\epsilon,\delta) = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|))$ such that for all $t \geq 0$,

$$|V^*(s_t) - V^{d*}_{\mathcal{L},t,h^*(\epsilon,\delta)}(s_t)| \leq \epsilon.$$

with probability at least $1-\delta$.

The first condition ensures that the agent efficiently learns the h-reachable models. The second condition guarantees that the learning method $\mathcal{L}$ and the distance function d are not arbitrarily poor.

In the following, we relate PAC-RMDP(h) to PAC-MDP and near-Bayes optimality. The proofs are given in the appendix at the end of this paper.

Proposition 1. (PAC-MDP) If an algorithm is PAC-RMDP($h^*(\epsilon,\delta)$), then it is PAC-MDP, where $h^*(\epsilon,\delta)$ is given in Definition 1.

Proposition 2. (Near-Bayes optimality) Consider model-based Bayesian reinforcement learning (Strens 2000). Let H be a planning horizon in the belief space b. Assume that the Bayesian optimal value function, $V^*_{b,H}$, converges to the H-reachable optimal function such that, for all $\epsilon > 0$, $|V^{d*}_{\mathcal{L},t,H}(s_t) - V^*_{b,H}(s_t,b_t)| \leq \epsilon$ for all but polynomial time steps. Then, a PAC-RMDP(H) algorithm with a policy $\mathcal{A}_t$ obtains an expected cumulative reward $V^{\mathcal{A}_t}(s_t) \geq V^*_{b,H}(s_t,b_t) - 2\epsilon$ for all but polynomial time steps with probability at least $1-\delta$.

Note that $V^{\mathcal{A}_t}(s_t)$ is the *actual* expected cumulative reward with the expectation over the true dynamics P, whereas $V^*_{b,H}(s_t,b_t)$ is the *believed* expected cumulative reward with the expectation over the current belief b_t and its belief evolution. In addition, whereas the PAC-RMDP(H) condition guarantees convergence to an H-reachable optimal value function, Bayesian optimality does *not*[1]. In this sense, Proposition 2 suggests that the theoretical guarantee of PAC-RMDP(H) would be stronger than that of near-Bayes optimality with an H step lookahead.

Summarizing the above, PAC-RMDP($h^*(\epsilon,\delta)$) implies PAC-MDP, and PAC-RMDP(H) is related to near-Bayes optimality. Moreover, as h decreases in the range $(0, h^*)$ or $(0, H)$, the theoretical guarantee of PAC-RMDP(h) becomes

[1] A Bayesian estimation with random samples converges to the true value under certain assumptions. However, for exploration, the selection of actions can cause the Bayesian optimal agent to ignore some state-action pairs, removing the guarantee of the convergence. This effect was well illustrated by Li (2009, Example 9).

Algorithm 1 Discrete PAC-RMDP

Parameter: $h \geq 0$

for time step $t = 1, 2, 3, \ldots$ **do**
Action: Take action based on $\tilde{V}^A(s_t)$ in Equation (1)
Observation: Save the sufficient statistics
Estimate: Update the model $\widehat{P}_{t,0}$

weaker than previous theoretical objectives. This accommodates the practical need to improve the trade-off between the theoretical guarantee (i.e., optimal behavior after a long period of exploration) and practical performance (i.e., satisfactory behavior after a reasonable period of exploration) via the concept of reachability. We discuss the relationship to bounded rationality (Simon 1982) and bounded optimality (Russell and Subramanian 1995) as well as the corresponding notions of regret and average loss in the appendix.

Discrete Domain

To illustrate the proposed concept, we first consider a simple case involving finite state and action spaces with an unknown transition function P. Without loss of generality, we assume that the reward function R is known.

Algorithm

Let $\tilde{V}^{\mathcal{A}}(s)$ be the internal value function used by the algorithm to choose an action. Let $V^{\mathcal{A}}(s)$ be the actual value function according to true dynamics P. To derive the algorithm, we use the principle of optimism in the face of uncertainty, such that $\tilde{V}^{\mathcal{A}}(s) \geq V^{d*}_{\mathcal{L},t,h}(s)$ for all $s \in S$. This can be achieved using the following internal value function:

$$\tilde{V}^{\mathcal{A}}(s) = \max_{a,} \sum_{\tilde{P} \in \mathcal{M}_{\mathcal{L},t,h,(s,a)}} \sum_{s'} \tilde{P}(s'|s,a)[R(s,a,s') + \gamma \tilde{V}^{\mathcal{A}}(s')] \quad (1)$$

The pseudocode is shown in Algorithm 1. In the following, we consider the special case in which we use the sample mean estimator (which determines $\mathcal{L}$). That is, we use $\widehat{P}_t(s'|s,a) = n_t(s,a,s')/n_t(s,a)$, where $n_t(s,a)$ is the number of samples for the state-action pair (s,a), and $n_t(s,a,s')$ is the number of samples for the transition from s to s' given an action a. In this case, the maximum over the model in Equation (1) is achieved when all future h observations are transitions to the state with the best value. Thus, $\tilde{V}^{\mathcal{A}}$ can be computed by $\tilde{V}^{\mathcal{A}}(s) = \max_a \sum_{s' \in S} \frac{n_t(s,a,s')}{n_t(s,a)+h}[R(s,a,s') + \gamma \tilde{V}^{\mathcal{A}}(s')] + \max_{s'} \frac{h}{n_t(s,a)+h}[R(s,a,s') + \gamma \tilde{V}^{\mathcal{A}}(s')]$.

Analysis

We first show that Algorithm 1 is PAC-RMDP(h) for all $h \geq 0$ (Theorem 1), maintains an anytime error bound and average loss bound (Corollary 1 and the following discussion), and is related with previous algorithms (Remarks 1 and 2). We then analyze its *explicit exploration runtime* (Definition 3). We assume that Algorithm 1 is used with the sample mean estimator, which determines $\mathcal{L}$. We fix the distance function as $d(\widehat{P}(\cdot|s,a), P(\cdot|s,a)) = \|\widehat{P}(\cdot|s,a) - P(\cdot|s,a)\|_1$. The proofs are given in the appendix.

Theorem 1. (PAC-RMDP) Let $\mathcal{A}_t$ be a policy of Algorithm 1. Let $z = \max(h, \frac{\ln(2^{|S|}|S||A|/\delta)}{\epsilon(1-\gamma)})$. Then, for all $\epsilon > 0$, for all $\delta = (0,1)$, and for all $h \geq 0$,

1) for all but at most $O\left(\frac{z|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta}\right)$ time steps, $V^{\mathcal{A}_t}(s_t) \geq V^{d*}_{\mathcal{L},t,h}(s_t) - \epsilon$, with probability at least $1-\delta$, *and*
2) there exist $h^*(\epsilon,\delta) = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|))$ such that $|V^*(s_t) - V^{d*}_{\mathcal{L},t,h^*(\epsilon,\delta)}(s_t)| \leq \epsilon$ with probability at least $1-\delta$.

Definition 2. (Anytime error) The anytime error $\epsilon_{t,h} \in \mathbb{R}$ is the smallest value such that $V^{\mathcal{A}_t}(s_t) \geq V^{d*}_{\mathcal{L},t,h}(s_t) - \epsilon_{t,h}$.

Corollary 1. (Anytime error bound) With probability at least $1-\delta$, if $h \leq \frac{\ln(2^{|S|}|S||A|/\delta)}{\epsilon(1-\gamma)}$, $\epsilon_{t,h} = O\left(\sqrt[3]{\frac{|S||A|}{t(1-\gamma)^3} \ln \frac{|S||A|}{\delta} \ln \frac{2^{|S|}|S||A|}{\delta}}\right)$; otherwise, $\epsilon_{t,h} = O\left(\sqrt{\frac{h|S||A|}{t(1-\gamma)^2} \ln \frac{|S||A|}{\delta}}\right)$.

The anytime T-step average loss is equal to $\frac{1}{T}\sum_{t=1}^{T}(1-\gamma^{T+1-t})\epsilon_{t,h}$. Moreover, in this simple problem, we can relate Algorithm 1 to a particular PAC-MDP algorithm and a near-Bayes optimal algorithm.

Remark 1. (Relation to MBIE) Let $m = O(\frac{|S|}{\epsilon^2(1-\gamma)^4} + \frac{1}{\epsilon^2(1-\gamma)^4} \ln \frac{|S||A|}{\epsilon(1-\gamma)\delta})$. Let $h^*(s,a) = \frac{n(s,a)z(s,a)}{1-z(s,a)}$, where $z(s,a) = 2\sqrt{2[\ln(2^{|S|}-2) - \ln(\delta/(2|S||A|m))]/n(s,a)}$. Then, Algorithm 1 with the input parameter $h = h^*(s,a)$ behaves identically to a PAC-MDP algorithm, Model Based Interval Estimation (MBIE) (Strehl and Littman 2008a), the sample complexity of which is $O(\frac{|S||A|}{\epsilon^3(1-\gamma)^6}(|S| + \ln \frac{|S||A|}{\epsilon(1-\gamma)\delta}) \ln \frac{1}{\delta} \ln \frac{1}{\epsilon(1-\gamma)}))$.

Remark 2. (Relation to BOLT) Let $h = H$, where H is a planning horizon in the belief space b. Assume that Algorithm 1 is used with an independent Dirichlet model for each (s,a), which determines $\mathcal{L}$. Then, Algorithm 1 behaves identically to a near-Bayes optimal algorithm, Bayesian Optimistic Local Transitions (BOLT) (Araya-López, Thomas, and Buffet 2012), the sample complexity of which is $O(\frac{H^2|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta})$.

As expected, the sample complexity for PAC-RMDP(h) (Theorem 1) is smaller than that for PAC-MDP (Remark 1) (at least when $h \leq |S|(1-\gamma)^{-3}$), but larger than that for near-Bayes optimality (Remark 2) (at least when $h \geq H$). Note that BOLT is not necessarily PAC-RMDP(h), because misleading priors can violate both conditions in Definition 1.

Further Discussion An important observation is that, when $h \leq \frac{|S|}{\epsilon(1-\gamma)} \ln \frac{|S||A|}{\delta}$, the sample complexity of Algorithm 1 is dominated by the number of samples required to refine the model, rather than the explicit exploration of unknown aspects of the world. Recall that the internal value function $\tilde{V}^{\mathcal{A}}$ is designed to force the agent to explore, whereas the use of the currently estimated value function $V^{d*}_{\mathcal{L},t,0}(s)$ results in exploitation. The difference between $\tilde{V}^{\mathcal{A}}$ and $V^{*}_{\mathcal{L},t,0}(s)$ decreases at a rate of $O(h/n_t(s,a))$, whereas the error between $V^{\mathcal{A}}$ and $V^{d*}_{\mathcal{L},t,0}(s)$ decreases at a rate of $O(1/\sqrt{n_t(s,a)})$. Thus, Algorithm 1 would stop the explicit exploration much sooner (when $\tilde{V}^{\mathcal{A}}$ and $V^{d*}_{\mathcal{L},t,0}(s)$ become

close), and begin exploiting the model, while still refining it, so that $V^{d*}_{\mathcal{L},t,0}(s)$ tends to $V^{\mathcal{A}}$. In contrast, PAC-MDP algorithms are forced to explore until the error between $V^{\mathcal{A}}$ and V^* becomes sufficiently small, where the error decreases at a rate of $O(1/\sqrt{n_t(s,a)})$. This provides some intuition to explain why a PAC-RMDP(h) algorithm with small h may avoid over-exploration, and yet, in some cases, learn the true dynamics to a reasonable degree, as shown in the experimental examples.

In the following, we formalize the above discussion.

Definition 3. (Explicit exploration runtime) The *explicit exploration runtime* is the smallest integer τ such that for all $t \geq \tau$, $|\tilde{V}^{\mathcal{A}_t}(s_t) - V^{d*}_{\mathcal{L},t,0}(s_t)| \leq \epsilon$.

Corollary 2. (Explicit exploration bound) With probability at least $1-\delta$, the explicit exploration runtime of Algorithm 1 is $O(\frac{h|S||A|}{\epsilon(1-\gamma)\Pr[A_K]} \ln \frac{|S||A|}{\delta}) = O(\frac{h|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta})$, where A_K is the escape event defined in the proof of Theorem 1.

If we assume $\Pr[A_K]$ to stay larger than a fixed constant, or to be very small ($\leq \frac{\epsilon(1-\gamma)}{3R_{max}}$) (so that $\Pr[A_K]$ does not appear in Corollary 2 as shown in the corresponding case analysis for Theorem 1), the explicit exploration runtime can be reduced to $O(\frac{h|S||A|}{\epsilon(1-\gamma)} \ln \frac{|S||A|}{\delta})$. Intuitively, this happens when the given MDP does not have low yet not-too low probability and high-consequence transition that is initially unknown. Naturally, such a MDP is difficult to learn, as reflected in Corollary 2.

Experimental Example

We compare the proposed algorithm with MBIE (Strehl and Littman 2008a), variance-based exploration (VBE) (Sorg, Singh, and Lewis 2010), Bayesian Exploration Bonus (BEB) (Kolter and Ng 2009), and BOLT (Araya-López, Thomas, and Buffet 2012). These algorithms were designed to be PAC-MDP or near-Bayes optimal, but have been used with parameter settings that render them neither PAC-MDP nor near-Bayes optimal. In contrast to the experiments in previous research, we present results with ϵ set to several theoretically meaningful values[2] as well as one theoretically non-meaningful value to illustrate its property[3]. Because our algorithm is deterministic with no sampling and no assumptions on the input distribution, we do not compare it with algorithms that use sampling, or rely heavily on knowledge of the input distribution.

[2]MBIE is PAC-MDP with the parameters δ and ϵ. VBE is PAC-MDP in the assumed (prior) input distribution with the parameter δ. BEB and BOLT are near-Bayes optimal algorithms whose parameters β and η are fully specified by their analyses, namely $\beta = 2H^2$ and $\eta = H$. Following Araya-López, Thomas, and Buffet (2012), we set β and η using the ϵ'-approximated horizon $H \approx \lceil \log_\gamma(\epsilon'(1-\gamma)) \rceil = 148$. We use the sample mean estimator for the PAC-MDP and PAC-RMDP(h) algorithms, and an independent Dirichlet model for the near-Bayes optimal algorithms.

[3]We can interpolate their qualitative behaviors with values of ϵ other than those presented here. This is because the principle behind our results is that small values of ϵ causes over-exploration due to the focus on the near-optimality.

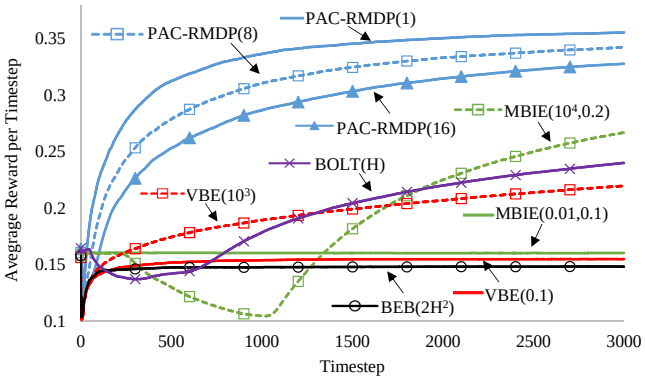


Figure 1: Average total reward per time step for the Chain Problem. The algorithm parameters are shown as PAC-RMDP(h), MBIE(ϵ, δ), VBE(δ), BEB(β), and BOLT(η).

We consider a five-state chain problem (Strens 2000), which is a standard toy problem in the literature. In this problem, the optimal policy is to move toward the state farthest from the initial state, but the reward structure explicitly encourages an exploitation agent, or even an ϵ-greedy agent, to remain in the initial state. We use a discount factor of $\gamma = 0.95$ and a convergence criterion for the value iteration of $\epsilon' = 0.01$.

Figure 1 shows the numerical results in terms of the average reward per time step (average over 1000 runs). As can be seen from this figure, the proposed algorithm worked better. MBIE and VBE work reasonably if we discard the theoretical guarantee. As the maximum reward is $R_{max} = 1$, the upper bound on the value function is $\sum_{i=1}^{\infty} \gamma^i R_{max} = \frac{1}{1-\gamma} R_{max} = 20$. Thus, ϵ-closeness does not yield any useful information when $\epsilon \geq 20$. A similar problem was noted by Kolter and Ng (2009) and Araya-López, Thomas, and Buffet (2012).

In the appendix, we present the results for a problem with low-probability high-consequence transitions, in which PAC-RMDP(8) produced the best result.

Continuous Domain

In this section, we consider the problem of a continuous state space and discrete action space. The transition function is possibly nonlinear, but can be linearly parameterized as: $s^{(i)}_{t+1} = \theta^T_{(i)} \Phi_{(i)}(s_t, a_t) + \zeta^{(i)}_t$, where the state $s_t \in S \subseteq \mathbb{R}^{n_S}$ is represented by n_S state parameters ($s^{(i)} \in \mathbb{R}$ with $i \in \{1, \ldots, n_s\}$), and $a_t \in A$ is the action at time t. We assume that the basis functions $\Phi_{(i)} : S \times A \to \mathbb{R}^{n_i}$ are known, but the weights $\theta \in \mathbb{R}^{n_i}$ are unknown. $\zeta^{(i)}_t \in \mathbb{R}$ is the noise term and given by $\zeta^{(i)}_t \sim \mathcal{N}(0, \sigma^2_{(i)})$. In other words, $P(s^{(i)}_{t+1} | s_t, a_t) = \mathcal{N}(\theta^T_{(i)} \Phi_{(i)}(s_t, a_t), \sigma^2_{(i)})$. For brevity, we focus on unknown transition dynamics, but our method is directly applicable to unknown reward functions if the reward is represented in the above form. This problem is a slightly generalized version of those considered by Abbeel and Ng (2005), Strehl and Littman (2008b), and Li et al. (2011).

Algorithm

We first define the variables used in our algorithm, and then explain how the algorithm works. Let $\hat{\theta}_{(i)}$ be the vector of the model parameters for the i^{th} state component. Let $X_{t,i} \in \mathbb{R}^{t \times n_i}$ consist of t input vectors $\Phi_{(i)}^T(s,a) \in \mathbb{R}^{1 \times n_i}$ at time t. We then denote the eigenvalue decomposition of the input matrix as $X_{t,i}^T X_{t,i} = U_{t,i} D_{t,i}(\lambda_{(1)}, \ldots, \lambda_{(n)}) U_{t,i}^T$, where $D_{t,i}(\lambda_{(1)}, \ldots, \lambda_{(n)}) \in \mathbb{R}^{n_i \times n_i}$ represents a diagonal matrix. For simplicity of notation, we arrange the eigenvectors and eigenvalues such that the diagonal elements of $D_{t,i}(\lambda_{(1)}, \ldots, \lambda_{(n)})$ are $\lambda_{(1)}, \ldots, \lambda_{(j)} \geq 1$ and $\lambda_{(j+1)}, \ldots, \lambda_{(n)} < 1$ for some $0 \leq j \leq n$. We now define the main variables used in our algorithm: $z_{t,i} := (X_{t,i}^T X_{t,i})^{-1}$, $g_{t,i} := U_{t,i} D_{t,i}(\frac{1}{\lambda_{(1)}}, \ldots, \frac{1}{\lambda_{(j)}}, 0, \ldots, 0) U_{t,i}^T$, and $w_{t,i} := U_{t,i} D_{t,i}(0, \ldots, 0, 1_{(j+1)}, \ldots, 1_{(n)}) U_{t,i}^T$. Let $\Delta^{(i)} \geq \sup_{s,a} |(\theta_{(i)} - \hat{\theta}_{(i)})^T \Phi_{(i)}(s,a)|$ be the upper bound on the model error. Define $\varsigma(M) = \sqrt{2 \ln(\pi^2 M^2 n_s h / (6\delta))}$ where M is the number of calls for $\mathbf{I}_h$ (i.e., the number of computing $\tilde{r}$ in Algorithm 2).

With the above variables, we define the h-reachable model interval I_h as

$$I_h(\Phi_{(i)}(s,a), X_{t,i}) / [h(\Delta^{(i)} + \varsigma(M)\sigma_{(i)})] = |\Phi_{(i)}^T(s,a) g_{t,i} \Phi_{(i)}(s,a)| + \|\Phi_{(i)}^T(s,a) z_{t,i}\| \|w_{t,i} \Phi_{(i)}(s,a)\|.$$

The h-reachable model interval is a function that maps a new state-action pair considered in the planning phase, $\Phi_{(i)}(s,a)$, and the agent's experience, $X_{t,i}$, to the upper bound of the error in the model prediction. We define the column vector consisting of n_S h-reachable intervals as $\mathbf{I}_h(s,a,X_t) = [I_h(\Phi_{(1)}(s,a), X_{t,1}), \ldots, I_h(\Phi_{(n_S)}(s,a), X_{t,n_S})]^T$.

We also leverage the continuity of the internal value function $\tilde{V}$ to avoid an expensive computation (to translate the error in the model to the error in value).

Assumption 1. (Continuity) There exists $L \in \mathbb{R}$ such that, for all $s, s' \in S$, $|\tilde{V}^*(s) - \tilde{V}^*(s')| \leq L\|s - s'\|$.

We set the degree of optimism for a state-action pair to be proportional to the uncertainty of the associated model. Using the h-reachable model interval, this can be achieved by simply adding a reward bonus that is proportional to the interval. The pseudocode for this is shown in Algorithm 2.

Analysis

Following previous work (Strehl and Littman 2008b; Li et al. 2011), we assume access to an exact planning algorithm. This assumption would be relaxed by using a planning method that provides an error bound. We assume that Algorithm 2 is used with least-squares estimation, which determines $\mathcal{L}$. We fix the distance function as $d(\widehat{P}(\cdot|s,a), P(\cdot|s,a)) = |E_{s' \sim \widehat{P}(\cdot|s,a)}[s'] - E_{s' \sim P(\cdot|s,a)}[s']|$ (since the unknown aspect is the mean, this choice makes sense). In the following, we use $\bar{n}$ to represent the average value of $\{n_{(1)}, \ldots, n_{(n_S)}\}$. The proofs are given in the appendix.

Lemma 3. (Sample complexity of PAC-MDP) For our problem setting, the PAC-MDP algorithm proposed by Strehl and Littman (2008b) and Li et al. (2011) has sample complexity $\tilde{O}\left(\frac{n_S^2 \bar{n}^2}{\epsilon^5 (1-\gamma)^{10}}\right)$.

Algorithm 2 Linear PAC-RMDP

Parameter: h, δ Optional: $\Delta^{(i)}, L$

Initialize: $\hat{\theta}$, $\Delta^{(i)}$, and L
for time step $t = 1, 2, 3, \ldots\ldots$ **do**
  Action: take an action based on
    $\hat{p}(s'|s,a) \leftarrow \mathcal{N}(\hat{\theta}^T \Phi(s,a), \sigma^2 I)$
    $\tilde{r}(s,a,s') \leftarrow R(s,a,s') + L\|\mathbf{I}_h(s,a,X_{t-1})\|$
  Observation: Save the input-output pair $(s_{t+1}, \Phi_t(s_t, a_t))$
  Estimate: Estimate $\hat{\theta}_{(i)}$, $\Delta^{(i)}$(if not given), and L(if not given)

Theorem 2. (PAC-RMDP) Let $\mathcal{A}_t$ be the policy of Algorithm 2. Let $z = \max(h^2 \ln \frac{m^2 n_s h}{\delta}, \frac{L^2 n_S \bar{n} \ln^2 m}{\epsilon^3} \ln \frac{n_S}{\delta})$. Then, for all $\epsilon > 0$, for all $\delta = (0,1)$, and for all $h \geq 0$,

1) for all but at most $m' = O\left(\frac{z L^2 n_S \bar{n} \ln^2 m}{\epsilon^3 (1-\gamma)^2} \ln^2 \frac{n_S}{\delta}\right)$ time steps (with $m \leq m'$), $V^{\mathcal{A}_t}(s_t) \geq V_{\mathcal{L},t,h}^{d*}(s_t) - \epsilon$, with probability at least $1 - \delta$, *and*
2) there exists $h^*(\epsilon, \delta) = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|))$ such that $|V^*(s_t) - V_{\mathcal{L},t,h^*(\epsilon,\delta)}^{d*}(s_t)| \leq \epsilon$ with probability at least $1 - \delta$.

Corollary 3. (Anytime error bound) With probability at least $1 - \delta$, if $h^2 \ln \frac{m^2 n_s h}{\delta} \leq \frac{L^2 n_S \bar{n} \ln^2 m}{\epsilon^3} \ln \frac{n_S}{\delta}$, $\epsilon_{t,h} = O\left(\sqrt[5]{\frac{L^4 n_S^2 \bar{n}^2 \ln^2 m}{t(1-\gamma)} \ln^3 \frac{n_S}{\delta}}\right)$; otherwise, $\epsilon_{t,h} = O\left(\frac{h^2 L^2 n_S \bar{n} \ln^2 m}{t(1-\gamma)} \ln^2 \frac{n_S}{\delta}\right)$.

The anytime T-step average loss is equal to $\frac{1}{T} \sum_{t=1}^{T} (1 - \gamma^{T+1-t}) \epsilon_{t,h}$.

Corollary 4. (Explicit exploration runtime) With probability at least $1 - \delta$, the explicit exploration runtime of Algorithm 2 is $O\left(\frac{h^2 L^2 n_S \bar{n} \ln m}{\epsilon^2 \Pr[A_k]} \ln^2 \frac{n_S}{\delta} \ln \frac{m^2 n_s h}{\delta}\right) = O\left(\frac{h^2 L^2 n_S \bar{n} \ln m}{\epsilon^3 (1-\gamma)} \ln^2 \frac{n_S}{\delta} \ln \frac{m^2 n_s h}{\delta}\right)$, where A_K is the escape event defined in the proof of Theorem 2.

Experimental Examples

We consider two examples: the mountain car problem (Sutton and Barto 1998), which is a standard toy problem in the literature, and the HIV problem (Ernst et al. 2006), which originates from a real-world problem. For both examples, we compare the proposed algorithm with a directly related PAC-MDP algorithm (Strehl and Littman 2008b; Li et al. 2011). For the PAC-MDP algorithm, we present the results with ϵ set to several theoretically meaningful values and one theoretically non-meaningful value to illustrate its property[4]. We used $\delta = 0.9$ for the PAC-MDP and PAC-RMDP algorithms[5]. The ϵ-greedy algorithm is executed with $\epsilon = 0.1$. In the planning phase, L is estimated as $L \leftarrow \max_{s,s' \in \Omega} |\tilde{V}^{\mathcal{A}}(s) - \tilde{V}^{\mathcal{A}}(s')| / \|s - s'\|$, where Ω is the set of states that are visited in the planning phase (i.e., fitted

[4] See footnote 3 on the consideration of different values of ϵ.

[5] We considered $\delta = [0.5, 0.8, 0.9, 0.95]$, but there was no change in any qualitative behavior of interest in our discussion.

value iteration and a greedy roll-out method). For both problems, more detailed descriptions of the experimental settings are available in the appendix.

Mountain Car In the mountain car problem, the reward is negative everywhere except at the goal. To reach the goal, the agent must first travel far away, and must explore the world to learn this mechanism. Each episode consists of 2000 steps, and we conduct simulations for 100 episodes.

The numerical results are shown in Figure 2. As in the discrete case, we can see that the PAC-RMDP(h) algorithm worked well. The best performance, in terms of the total reward, was achieved by PAC-RMDP(10). Since this problem required a number of consecutive explorations, the random exploration employed by the ϵ-greedy algorithm did not allow the agent to reach the goal. As a result of exploration and the randomness in the environment, the PAC-MDP algorithm reached the goal several times, but kept exploring the environment to ensure near-optimality. From Figure 2, we can see that the PAC-MDP algorithm quickly converges to good behavior if we discard the theoretical guarantee (the difference between the values in the optimal value function had an upper bound of 120, and the total reward had an upper bound of 2000. Hence, $\epsilon > 2000$ does not yield a useful theoretical guarantee).

Simulated HIV Treatment This problem is described by a set of six ordinary differential equations (Ernst et al. 2006). An action corresponds to whether the agent administers two treatments (RTIs and PIs) to patients (thus, there are four actions). Two types of exploration are required: one to learn the effect of using treatments on viruses, and another to learn the effect of not using treatments on immune systems. Learning the former is necessary to reduce the population of viruses, but the latter is required to prevent the overuse of treatments, which weakens the immune system. Each episode consists of 1000 steps (i.e., days), and we conduct simulations for 30 episodes.

As shown in Figure 3, the PAC-MDP algorithm worked reasonably well with $\epsilon = 30^{10}$. However, the best total reward did not exceed 30^{10}, and so the PAC-MDP guarantee with $\epsilon = 30^{10}$ does not seem to be useful. The ϵ-greedy algorithm did not work well, as this example required sequential exploration at certain periods to learn the effects of treatments.

Conclusion

In this paper, we have proposed the PAC-RMDP framework to bridge the gap between theoretical objectives and practical needs. Although the PAC-RMDP(h) algorithms worked well in our experimental examples with small h, it is possible to devise a problem in which the PAC-RMDP algorithm should be used with large h. In extreme cases, the algorithm would reduce to PAC-MDP. Thus, the adjustable theoretical guarantee of PAC-RMDP(h) via the concept of reachability seems to be a reasonable objective.

Whereas the development of algorithms with traditional objectives (PAC-MDP or regret bounds) requires the consideration of confidence intervals, PAC-RMDP(h) concerns

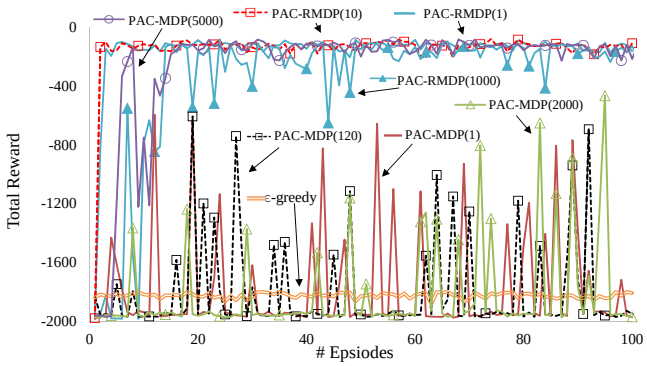


Figure 2: Total reward per episode for the mountain car problem with PAC-RMDP(h) and PAC-MDP(ϵ).

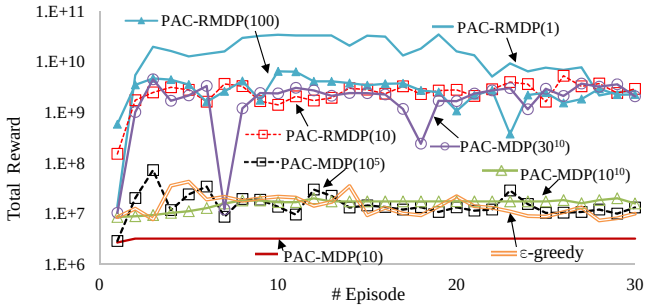


Figure 3: Total reward per episode for the HIV problem with PAC-RMDP(h) and PAC-MDP(ϵ).

a set of h-reachable models. For a flexible model, the derivation of the confidence interval would be a difficult task, but a set of h-reachable models can simply be computed (or approximated) via lookahead using the model update rule. Thus, future work includes the derivation of a PAC-RMDP algorithm with a more flexible and/or structured model.

Acknowledgment

The author would like to thank Prof. Michael Littman, Prof. Leslie Kaelbling and Prof. Tomás Lozano-Pérez for their thoughtful comments and suggestions. We gratefully acknowledge support from NSF grant 1420927, from ONR grant N00014-14-1-0486, and from ARO grant W911NF1410433. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of our sponsors.

References

Abbeel, P., and Ng, A. Y. 2005. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning (ICML)*.

Adams, B.; Banks, H.; Kwon, H.-D.; and Tran, H. T. 2004. Dynamic multidrug therapies for HIV: Optimal and STI control approaches. *Mathematical Biosciences and Engineering* 1(2):223–241.

Araya-López, M.; Thomas, V.; and Buffet, O. 2012. Near-optimal BRL using optimistic local transitions. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *The Journal of Machine Learning Research (JMLR)* 3:397–422.

Bernstein, A., and Shimkin, N. 2010. Adaptive-resolution reinforcement learning with polynomial exploration in deterministic domains. *Machine learning* 81(3):359–397.

Brunskill, E. 2012. Bayes-optimal reinforcement learning for discrete uncertainty domains. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Cook, R. D. 1977. Detection of influential observation in linear regression. *Technometrics* 15–18.

Ernst, D.; Stan, G.-B.; Goncalves, J.; and Wehenkel, L. 2006. Clinical data based optimal STI strategies for HIV: a reinforcement learning approach. In *Proceedings of the 45th IEEE Conference on Decision and Control*.

Fiechter, C.-N. 1994. Efficient reinforcement learning. In *Proceedings of the seventh annual ACM conference on Computational learning theory (COLT)*.

Jaksch, T.; Ortner, R.; and Auer, P. 2010. Near-optimal regret bounds for reinforcement learning. *The Journal of Machine Learning Research (JMLR)* 11:1563–1600.

Kawaguchi, K., and Araya, M. 2013. A greedy approximation of Bayesian reinforcement learning with probably optimistic transition model. In *Proceedings of AAMAS 2013 workshop on adaptive learning agents*, 53–60.

Kearns, M., and Singh, S. 1999. Finite-sample convergence rates for Q-learning and indirect algorithms. In *Proceedings of Advances in neural information processing systems (NIPS)*.

Kearns, M., and Singh, S. 2002. Near-optimal reinforcement learning in polynomial time. *Machine Learning* 49(2-3):209–232.

Kolter, J. Z., and Ng, A. Y. 2009. Near-Bayesian exploration in polynomial time. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*.

Li, L.; Littman, M. L.; Walsh, T. J.; and Strehl, A. L. 2011. Knows what it knows: a framework for self-aware learning. *Machine learning* 82(3):399–443.

Li, L. 2009. *A unifying framework for computational reinforcement learning theory*. Ph.D. Dissertation, Rutgers, The State University of New Jersey.

Li, L. 2012. Sample complexity bounds of exploration. In *Reinforcement Learning*. Springer. 175–204.

Pazis, J., and Parr, R. 2013. PAC Optimal Exploration in Continuous Space Markov Decision Processes. In *Proceedings of the 27th AAAI conference on Artificial Intelligence (AAAI)*.

Puterman, M. L. 2004. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Russell, S. J., and Subramanian, D. 1995. Provably bounded-optimal agents. *Journal of Artificial Intelligence Research (JAIR)* 575–609.

Simon, H. A. 1982. *Models of bounded rationality, volumes 1 and 2*. MIT press.

Sorg, J.; Singh, S.; and Lewis, R. L. 2010. Variance-based rewards for approximate Bayesian reinforcement learning. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI)*.

Strehl, A. L., and Littman, M. L. 2008a. An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences* 74(8):1309–1331.

Strehl, A. L., and Littman, M. L. 2008b. Online linear regression and its application to model-based reinforcement learning. In *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 1417–1424.

Strehl, A. L.; Li, L.; and Littman, M. L. 2006. Incremental model-based learners with formal learning-time guarantees. In *Proceedings of the 22th Conference on Uncertainty in Artificial Intelligence (UAI)*.

Strehl, A. L. 2007. *Probably approximately correct (PAC) exploration in reinforcement learning*. Ph.D. Dissertation, Rutgers University.

Strens, M. 2000. A Bayesian framework for reinforcement learning. In *Proceedings of the 16th International Conference on Machine Learning (ICML)*.

Sutton, R. S., and Barto, A. G. 1998. *Reinforcement learning: An introduction*. MIT press Cambridge.

Weissman, T.; Ordentlich, E.; Seroussi, G.; Verdu, S.; and Weinberger, M. J. 2003. Inequalities for the L1 deviation of the empirical distribution. *Hewlett-Packard Labs, Tech. Rep*.

Zilberstein, S. 2008. Metareasoning and bounded rationality. In *Proceedings of the AAAI workshop on Metareasoning: Thinking about Thinking*.

Appendix A

A1. Proofs of Propositions 1 and 2

In this section, we present the proofs of Propositions 1 and 2.

Proposition 1. (PAC-MDP) PAC-RMDP($h^*(\epsilon, \delta)$) implies PAC-MDP, where $h^*(\epsilon, \delta)$ is given in Definition 1.

Proof. For any PAC-RMDP($h^*(\epsilon, \delta)$) algorithm, Definition 1 implies that $V^{\mathcal{A}}(s_t) \geq V^*_{h^*}(s_t) - \epsilon \geq V^*(s_t) - 2\epsilon$ with probability at least $1 - 2\delta$ for all but polynomial time steps. This satisfies the condition of the PAC-MDP. □

Proposition 2. (Near-Bayes optimality) Consider the model-based Bayesian reinforcement learning (Strens 2000). Let H be a planning horizon in the belief space b. Assume that the Bayesian optimal value function, $V^*_{b,H}$, converges to the H-reachable optimal function such that, for all $\epsilon > 0$, $|V^*_{\mathcal{L},t,H}(s_t) - V^*_{b,H}(s_t, b_t)| \leq \epsilon$ for all but polynomial time steps. Then, a PAC-RMDP(H) algorithm with a policy $\mathcal{A}_t$ obtains an expected cumulative reward $V^{\mathcal{A}_t}(s_t) \geq V^*_{b,H}(s_t, b_t) - 2\epsilon$ for all but polynomial time steps with probability at least $1 - \delta$.

Proof. It directly follows Definition 1 and the assumption. For all but polynomial time steps, with probability at least $1 - \delta$, $V^{\mathcal{A}}(s_t) \geq V^*_{\mathcal{L},t,H}(s_t) - \epsilon \geq V^*_{b,H}(s_t, b_t) - 2\epsilon$. □

A2. Relationship to Bounded Rationality and Bounded Optimality

As the concept of PAC-RMDP considers the inherent limitations of a decision maker, it shares properties with the concepts of bounded rationality (Simon 1982) and bounded optimality (Russell and Subramanian 1995).

Bounded rationality and bounded optimality focus on limitations in the planning phase (e.g., computational resources). In contrast, PAC-RMDP considers limitations in the learning phase (e.g., the agent's lifetime). As in the case of bounded rationality, the performance guarantee of a PAC-RMDP(h) algorithm can be arbitrary, depending on the choice of h. On the contrary, bounded optimality solves the problem of arbitrariness, seemingly at the cost of applicability. It requires a strong notion of optimality, similar to instance optimality; roughly, we must find the *optimal algorithm* given the available computational resources. Automated optimization over the set of algorithms is a difficult task. Zilberstein (2008) claims that bounded optimality is difficult to achieve, resulting in very few successful examples, and is not, in practice, as promising as other bounded rational methods. However, in future research, it would be interesting to compare PAC-RMDP with a possible relaxation of PAC-MDP based on a concept similar to bounded optimality.

A3. Corresponding Notions of Regret and Average Loss

In the definition of PAC-RMDP(h), our focus is on *learning* useful models, enabling us to obtain high rewards in a short period of time. Instead, one may wish to guarantee the worst total reward *in a given time horizon* T. There are several ways to achieve this goal. One solution is to minimize the expected T-step *regret bound* $r^{\mathcal{A}}(T)$, given by

$$r^{\mathcal{A}}(T) \geq V^*(s_0, T) - V^{\mathcal{A}}(s_0, T). \tag{1}$$

In this case, $V^*(s_0, T) = E\left[\sum_{i=0}^{T} \gamma^i R\left(s^*_i, \pi^*(s_i), s^*_{i+1}\right)\right]$, where the sequence of states $s^*_0, s^*_1, \ldots, s^*_T$ with $s^*_0 = s_0$ is generated when the agent follows the optimal policy π^* from s_0, and $V^{\mathcal{A}}(s_0, T) = E\left[\sum_{i=0}^{T} \gamma^i R\left(s_i, \mathcal{A}_i(s_i), s_{i+1}\right)\right]$, where the sequence of states $s_0, s_1, \ldots, s_T$ is generated when the agent follows policy $\mathcal{A}_i$. Since one mistake in the early stages may make it impossible to return to the optimal state sequence s^*_i, all the regret approaches in the literature rely on some reachability assumptions in the state space; for example, Jaksch, Ortner, and Auer (2010) assumed that every state was reachable from every other state within a certain (average) number of steps.

Another approach is to minimize the expected T-step *average loss bound* $r^{\mathcal{A}}(T)$, which obviates the need for any reachability assumptions in the state space:

$$\ell^{\mathcal{A}}(T) \geq \frac{1}{T} \sum_{t=0}^{T} \left[V^*(s_t, T) - V^{\mathcal{A}}(s_t, T)\right], \tag{2}$$

where s_t is the state visited by algorithm $\mathcal{A}$ at time t. The value functions inside the sum are defined as $V^*(s_t, T) = E\left[\sum_{i=0}^{T-t} \gamma^i R\left(s^*_{t+i}, \pi^*(s_{t+i}), s^*_{t+i+1}\right)\right]$ with $s^*_t = s_t$ and $V^{\mathcal{A}}(s_0, T) = E\left[\sum_{i=0}^{T-t} \gamma^i R\left(s^*_{t+i}, \mathcal{A}_t(s_{t+i}), s_{t+i+1}\right)\right]$. By averaging the T-step regrets (i.e., losses) of the T initial states $s_0, s_1, \ldots, s_T$ visited by $\mathcal{A}$, the average loss mitigates the effects of irreversible mistakes in the early stages that may dominate the regret.

The expected h-reachable regret bound $r^{\mathcal{A}}_h(T)$ and average loss bound $\ell^{\mathcal{A}}_h(T)$ are defined as $r^{\mathcal{A}}_h(T) \geq V^*_{\mathcal{L},t,h}(s_0, T) - V^{\mathcal{A}}(s_0, T)$ and $\ell^{\mathcal{A}}_h(T) \geq \frac{1}{T}\sum_{t=1}^{T}\left[V^*_{\mathcal{L},t,h}(s_t, T) - V^{\mathcal{A}}(s_t, T)\right]$. That is, they are the same as the standard expected regret and average loss, respectively, with the exception that the optimal value function V^* has been replaced by the h-reachable optimal value function $V^*_{\mathcal{L},t,h}(s_t)$.

While the definition of PAC-RMDP(h) focuses on exploration, the proposed PAC-RMDP(h) algorithms maintain anytime expected h-reachable average loss bounds and anytime error bounds, and thus the performances of our algorithms are expected to improve with time, rather than after some number of exploration steps.

A4. Proofs of Theoretical Results for Algorithm 1

We first verify the main properties of Algorithm 1 and then analyze a practically relevant property of the algorithm in the subsection of Further Discussion. We assume that Algorithm 1 is used with the sample mean estimator, which determines $\mathcal{L}$.

Main Properties To compare the results with those of past studies, we assume that $R_{max} \leq c$ for some fixed constant c. The effect of this assumption can be seen in the proof of Theorem 1. Algorithm 1 requires no input parameter related to ϵ and δ. This is because the required degree of optimism can be determined independently of the unknown aspect of the world. This means that Theorem 1 holds at any time during an execution for a pair of corresponding ϵ and δ.

Lemma 1. (Optimism) For all $s \in S$ and for all $t, h \geq 0$, the internal value $\tilde{V}^{\mathcal{A}_t}(s)$ used by Algorithm 1 is at least the h-reachable optimal value $V^*_{\mathcal{L},t,h}(s)$; $\tilde{V}^{\mathcal{A}_t}(s) \geq V^*_{\mathcal{L},t,h}(s)$.

Proof. The claim follows directly from the construction of Algorithm 1. It can be verified by induction on each step of the value iteration or the roll-out in a planning algorithm. □

Theorem 1. (PAC-RMDP) Let $\mathcal{A}_t$ be a policy of Algorithm 1. Let $z = \max(h, \frac{\ln(2^{|S|}|S||A|/\delta)}{\epsilon(1-\gamma)})$. Then, for all $\epsilon > 0$, for all $\delta = (0, 1)$, and for all $h \geq 0$,

1) for all but at most $O\left(\frac{z|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta}\right)$ time steps, $V^{\mathcal{A}_t}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \epsilon$, with probability at least $1 - \delta$, *and*
2) there exists $h^*(\epsilon, \delta) = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|))$ such that $|V^*(s_t) - V^*_{\mathcal{L},t,h^*(\epsilon,\delta)}(s_t)| \leq \epsilon$ with probability at least $1 - \delta$.

Proof. Let K be a set of state-action pairs where the agent has at least m samples (this corresponds to *the set of known state-action pairs* described by Kearns and Singh (2002)). With the boundary condition $\overline{V^{\mathcal{A}}}(s, 0) = 0$, define the mixed value function $\overline{V^{\mathcal{A}}}(s, H)$ with a finite horizon $H' = \frac{1}{1-\gamma} \ln \frac{6R_{max}}{\epsilon(1-\gamma)}$ as

$$\overline{V^{\mathcal{A}}}(s, H') = \begin{cases} \sum_{s'} P(s'|s, \mathcal{A}(s))[R(s, \mathcal{A}(s), s') + \gamma \overline{V^{\mathcal{A}}}(s', H' - 1)] & \text{if } (s, \mathcal{A}(s)) \in K \\ \max_{\tilde{P} \in \mathcal{M}_{\mathcal{L},t,h,(s,a)}} \sum_{s'} \tilde{P}(s'|s, \mathcal{A}(s))[R(s, \mathcal{A}(s), s') + \gamma \overline{V^{\mathcal{A}}}(s', H' - 1)] & \text{otherwise} \end{cases}$$

Let A_K be the escape event in which a pair $(s, a) \notin K$ is generated for the first time when starting at state s_t, following policy $\mathcal{A}_t$, and transitioning based on the true dynamics P for H' steps. Then, for all $t, h \geq 0$, with probability at least $1 - \delta/2$,

$$\begin{aligned} V^{\mathcal{A}_t}(s_t) &\geq \overline{V^{\mathcal{A}_t}}(s_t, H') - \frac{R_{max}}{1-\gamma} \Pr(A_k) - \frac{\epsilon}{6} \\ &\geq \tilde{V}^{\mathcal{A}_t}(s_t) - \frac{R_{max}}{1-\gamma} \Pr(A_k) - \frac{\epsilon}{3} - \frac{R_{max}}{1-\gamma} \left(\frac{h}{m} + \sqrt{\frac{2 \ln(2^{|S|+1}|S||A|/\delta)}{m}} \right) \\ &\geq V^*_{\mathcal{L},t,h}(s_t) - \frac{R_{max}}{1-\gamma} \Pr(A_k) - \frac{\epsilon}{3} - \frac{R_{max}}{1-\gamma} \left(\frac{h}{m} + \sqrt{\frac{2 \ln(2^{|S|+1}|S||A|/\delta)}{m}} \right). \end{aligned}$$

The first inequality follows from the fact that $V^{\mathcal{A}_t}(s_t)$ and $\overline{V}^{\mathcal{A}_t}(s_t)$ are only different when event A_K occurs, and their difference is bounded above by $\frac{R_{max}}{1-\gamma}$ (this is the upper bound on the value $\tilde{V}(s_t)$). Furthermore, the finite horizon approximation adds an error of $1/6\epsilon$. A more detailed argument only involves algebraic manipulations that mirror the proofs given by Strehl and Littman (2008a, Lemma 3) and Kearns and Singh (2002, Lemma 2).

The second inequality follows from the fact that $\overline{V}^{\mathcal{A}}$ is different from $\tilde{V}^{\mathcal{A}}$ only for the state-action pairs $(s, a) \in K$, for which $\tilde{V}^{\mathcal{A}_t}(s_t)$ deviates from $\overline{V}^{\mathcal{A}_t}(s_t)$ by at most $\frac{R_{max}}{1-\gamma}(\frac{h}{m} + \sqrt{2 \ln(2^{|S|+1}|S||A|/\delta)/m})$ with probability at least $1 - \delta/2$. This is because $|\tilde{V}^{\mathcal{A}_t}(s_t) - V^{\mathcal{A}_t}_{\mathcal{L},t,0}(s_t)| \leq \frac{R_{max}}{1-\gamma} \frac{h}{m}$ with certainty, and $|V^{\mathcal{A}_t}_{\mathcal{L},t,0}(s_t) - V^{\mathcal{A}_t}(s_t)| \leq \frac{R_{max}}{1-\gamma} \sqrt{2 \ln(2^{|S|+1}|S||A|/\delta)/m}$ with probability at least $1 - \delta/2$ (the later is due to the result of Weissman et al. (2003, Theorem 2.1) and the union bound for state-action pairs).

The third inequality follows from Lemma 1.

Therefore, if $h \leq \sqrt{2m \ln(2^{|S|+1}|S||A|/\delta)}$,

$$V^{\mathcal{A}_t}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \frac{R_{max}}{1-\gamma} \Pr(A_k) - \frac{\epsilon}{3} - \frac{2R_{max}}{1-\gamma} \sqrt{\frac{2 \ln(2^{|S|+1}|S||A|/\delta)}{m}}.$$

If $h > \sqrt{2m \ln(2^{|S|+1}|S||A|/\delta)}$,

$$V^{\mathcal{A}_t}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \frac{R_{max}}{1-\gamma} \Pr(A_k) - \frac{\epsilon}{3} - \frac{2R_{max}}{1-\gamma} \frac{h}{m}.$$

Let us consider the case where $h \leq \sqrt{2m\ln(2^{|S|+1}|S||A|/\delta)}$. We fix $m = \frac{72R_{max}^2 \ln(2^{|S|+1}|S||A|/\delta)}{\epsilon^2(1-\gamma)^2}$ to give $\frac{\epsilon}{3}$ in the last term on the right-hand side. If $\Pr(A_K) \leq \frac{\epsilon(1-\gamma)}{3R_{max}}$ for all t, $V^{\mathcal{A}_t}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \epsilon$ with probability at least $1-\delta/2$. For the case where $\Pr(A_K) > \frac{\epsilon(1-\gamma)}{3R_{max}}$ for some t, we define an independent random event A'_K such that $\Pr(A'_K) = \frac{\epsilon(1-\gamma)}{3R_{max}} < \Pr(A_K)$. According to the Chernoff bound, for all $k \geq 4$, with probability at least $1-\delta/2$, the event A_K will occur at least k times after $\frac{2k}{\Pr(A'_K)} \ln \frac{2}{\delta}$ time steps. Thus, by applying the union bound on $|S|$ and $|A|$, we have a probability of at least $1-\delta/2$ of event A_K occurring at least m times for all state-action pairs after $O\left(\frac{m|S||A|}{\Pr(A'_k)} \ln \frac{|S||A|}{\delta}\right) = O\left(\frac{mR_{max}|S||A|}{\epsilon(1-\gamma)} \ln \frac{|S||A|}{\delta}\right)$ time steps.

Let us carefully consider what this means. Whenever A_K occurs, the sample is used to minimize the error between $V^{\mathcal{A}}$ and $\tilde{V}^{\mathcal{A}}$ by the definition of A_K. Since $\tilde{V}(s) \geq V^*_{\mathcal{L},t,h}(s)$ holds at any time, whenever A_K occurs, the sample is used to reduce the error in $V^{\mathcal{A}_t}(s_t) \geq \tilde{V}^{\mathcal{A}_t}(s_t) - (\text{error}) \geq V^*_{\mathcal{L},t,h}(s_t) - (\text{error})$ (note that if $\tilde{V}(s) \geq V^*_{\mathcal{L},t,h}(s)$ holds randomly, this event must occur concurrently with A_K to reduce the error on the right-hand side). Thus, after this number of time steps, $Pr(A_K)$ goes to zero with probability at least $1-\delta/2$. Hence, from the union bound, the above inequality becomes $V^{\mathcal{A}}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \frac{2}{3}\epsilon$ with probability at least $1-\delta$.

For the case where $h > \sqrt{2m\ln(2^{|S|+1}|S||A|/\delta)}$, we fix $m = \frac{hR_{max}}{6\epsilon(1-\gamma)}$. The rest of the proof follows that for the case of smaller values of h. Therefore, we have proved the first part of the statement.

Finally, we consider the second part of the statement. Let $\widehat{P}_{t,h}(\cdot|s,a)$ be the future model obtained by updating the current model $\widehat{P}(\cdot|s,a)$ with h *random* future samples (h samples drawn from $P(S|s,a)$ for each $(s,a) \in (S,A)$). Using a result given by Weissman et al. (2003, Theorem 2.1), we know that for all $s \in S$, with probability at least $1-\delta$,

$$\max_{s,a} \|\widehat{P}_{t,h}(\cdot|s,a) - P(\cdot|s,a)\|_1 \leq \sqrt{\frac{2\ln(2^{|S|+1}|S||A|/\delta)}{n_{t,min}+h}},$$

where $n_{t,min} = \min_{s,a} n_t(s,a)$. Now, if we use the distance function $d(\widehat{P}(\cdot|s,a), P(\cdot|s,a)) = \|\widehat{P}(\cdot|s,a) - P(\cdot|s,a)\|_1$ to define the h-reachable optimal function,

$$\begin{aligned}
|V^*(s_t) - V^{d*}_{\mathcal{L},t,h^*(\epsilon,\delta)}(s_t)| &\leq \frac{R_{max}}{1-\gamma} \max_{s,a} \|P^{d*}_{\mathcal{L},t,h}(\cdot|s,a) - P(\cdot|s,a)\|_1 \\
&= \frac{R_{max}}{1-\gamma} \max_{s,a} \min_{\widehat{P} \in \mathcal{M}_{\mathcal{L},t,h,(s,a)}} \|\widehat{P}(\cdot|s,a) - P(\cdot|s,a)\|_1 \\
&\leq \frac{R_{max}}{1-\gamma} \sqrt{\frac{2\ln(2^{|S|+1}|S||A|/\delta)}{n_{t,min}+h}},
\end{aligned}$$

The last inequality follows that the models reachable with h *random* samples, $\widehat{P}_{t,h}(\cdot|s,a)$, are contained in a set of h-reachable models and the best h-reachable model, $P^{d*}_{\mathcal{L},t,h}(\cdot|s,a)$, explicitly minimize the norm, resulting in that $P^{d*}_{\mathcal{L},t,h}(\cdot|s,a)$ is at least as good as $\widehat{P}_{t,h}(\cdot|s,a)$ in terms of the norm. The right-hand side of the above inequality becomes less than or equal to ϵ when $h \leftarrow h^*(\epsilon,\delta) = \frac{2R_{max}^2 \ln(2^{|S|+1}|S||A|/\delta)}{\epsilon^2(1-\gamma)^2}$. Thus, we have the second part of the statement. □

Corollary 1. (Anytime error bound) With probability at least $1-\delta$, if $h \leq \frac{\ln(2^{|S|}|S||A|/\delta)}{\epsilon(1-\gamma)}$,

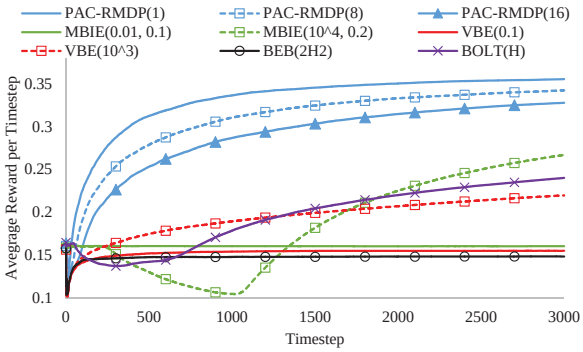
$$\epsilon_{t,h} = O\left(\sqrt[3]{\frac{|S||A|}{t(1-\gamma)^3} \ln \frac{|S||A|}{\delta} \ln \frac{2^{|S|}|S||A|}{\delta}}\right),$$

and otherwise,

$$\epsilon_{t,h} = O\left(\sqrt{\frac{h|S||A|}{t(1-\gamma)^2} \ln \frac{|S||A|}{\delta}}\right).$$

Proof. From Theorem 1, if $t = c\frac{z|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta}$ with c being some fixed constant, $V^{\mathcal{A}}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \epsilon$ with probability at least $1-\delta$. Since this holds for all $t \geq 0$ with corresponding ϵ and δ, it implies that $\epsilon^2 \leq A\frac{z|S||A|}{t(1-\gamma)^2} \ln \frac{|S||A|}{\delta}$ with probability at least $1-\delta$. Substituting $z = \max(h, \frac{\ln(2^{|S|}|S||A|/\delta)}{\epsilon(1-\gamma)})$ yields the statement. □

The anytime T-step average loss is equal to $\frac{1}{T}\sum_{t=1}^{T}(1-\gamma^{T+1-t})\epsilon_{t,h,\delta}$. Since the errors considered in Theorem 1 and Corollary 3 are for an infinite horizon, the factor $(1-\gamma^{T+1-t})$ translates the infinite horizon error to the T-step finite horizon error (this can be seen when we modify the proof of Theorem 1 by replacing $\frac{1}{1-\gamma}$ with $\frac{1-\gamma^{T+1-t}}{1-\gamma}$).

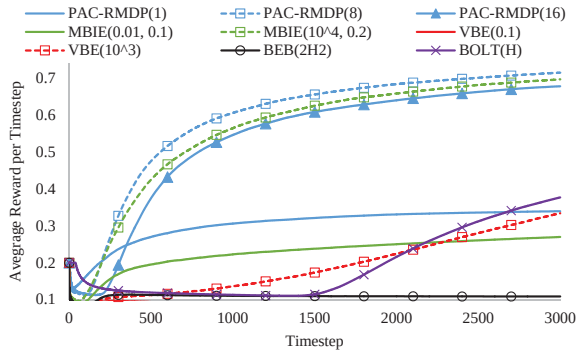


(a) Average of 1000 runs over all time steps

Algorithm	Average	10%	90%
PAC-RMDP(1)	0.357	0.332	0.378
PAC-RMDP(8)	0.343	0.321	0.365
PAC-RMDP(16)	0.328	0.305	0.321
MBIE(0.01, 0.1)	0.160	0.158	0.162
MBIE(20, 0.9)	0.160	0.158	0.162
MBIE(10^4, 0.2)	0.267	0.250	0.285
VBE(0.1)	0.155	0.152	0.158
VBE(0.99)	0.156	0.153	0.158
VBE(10^3)	0.220	0.207	0.232
BEB(2×148^2)	0.148	0.142	0.154
BOLT(148)	0.240	0.221	0.256

(b) Results for 1000 runs at time step 3000

Figure 1: Average total reward per time step for the Chain Problem. The algorithm parameters are shown as PAC-RMDP(h), MBIE(ϵ, δ), VBE(δ), BEB(β), and BOLT(η).



(a) Average for 1000 runs over all time steps

Algorithm	Average	10%	90%
PAC-RMDP(1)	0.339	0.196	0.772
PAC-RMDP(8)	0.715	0.650	0.784
PAC-RMDP(16)	0.678	0.612	0.747
MBIE(0.01, 0.1)	0.270	0.260	0.279
MBIE(20, 0.9)	0.327	0.313	0.340
MBIE(10^4, 0.2)	0.697	0.634	0.752
VBE(0.1)	0.090	0.060	0.122
VBE(0.99)	0.094	0.061	0.126
VBE(10^3)	0.334	0.306	0.360
BEB(2×148^2)	0.108	0.103	0.113
BOLT(148)	0.377	0.314	0.441

(b) Results for 1000 runs at time step 3000

Figure 2: Average total reward per time step for the modified Chain Problem. The algorithm parameters are shown as PAC-RMDP(h), MBIE(ϵ, δ), VBE(δ), BEB(β), and BOLT(η).

Corollary 2. (Explicit exploration runtime) With probability at least $1 - \delta$, the explicit exploration runtime of Algorithm 1 is $O(\frac{h|S||A|}{\epsilon(1-\gamma)\Pr[A_K]} \ln \frac{|S||A|}{\delta}) = O(\frac{h|S||A|}{\epsilon^2(1-\gamma)^2} \ln \frac{|S||A|}{\delta})$, where A_K is the escape event defined in the proof of Theorem 1.

Proof. The proof directly follows that of Theorem 1 with z. Compared to the sample complexity of Algorithm 1, z is replaced by h based on the proof of Theorem 1. □

A5. Additional Experimental Example for Discrete Domain

Figure 1 shows the results in the main paper along with 10% and 90% values.Aside from the proposed algorithm, only BOLT gathered better rewards than a greedy algorithm while maintaining the claimed theoretical guarantee.

In this example, our proposed algorithm worked well and maintained its theoretical guarantee. One might consider the theoretical guarantee of PAC-RMDP, especially PAC-RMDP(1), to be too weak. Two things should be noted. First, the 1-reachable value function is not the value function that can be obtained with just one additional sample, but requires an additional sample for all $|S||A|$ state-action pairs. Second, in contrast to Bayesian optimality, the 1-reachable value function is not the value function *believed* to be obtained with $|S||A|$ additional samples, but is *possibly* reachable in terms of the unknown true world dynamics with the new samples.

However, it is certainly possible to devise a problem such that PAC-RMDP(1) is not guaranteed to conduct sufficient exploration. As an example, we consider a modified version of the five-state chain problem, where the probability of successfully moving away from the initial state is very small (= 0.05), thus requiring more extensive exploration. We modified the transition model as follows: Let a_1 be the optimal action that moves the agent away from the initial state. For $i = \{2, 3, 4, 5\}$, $\Pr(s_i, a_1, s_{\min(i+1,5)}) = 0.99$ and $\Pr(s_i, a_1, s_1) = 0.01$. For $i = 1$, $\Pr(s_i, a_1, s_{i+1)}) = 0.05$ and $\Pr(s_i, a_1, s_1) = 0.95$. For action a_2 and any s_i, $\Pr(s_i, a_2, s_1) = 1$. The numerical results for this example are shown in Figure 2. As expected, the PAC-RMDP(1) algorithm often became stuck in the initial state.

A6. Proofs of Theoretical Results for Algorithm 2

We assume that Algorithm 2 is used with the least square estimation, which determines $\mathcal{L}$. Because the true world dynamics are assumed to have the parametric form $P(s'|s,a) = \mathcal{N}(\theta^T\Phi(s,a), \sigma^2 I)$ with a known σ, their unknown aspect is attributed to the weight vector θ. Therefore, we discuss h-reachability in terms of $\hat{\theta}$ instead of $\widehat{P}$. For each i^{th} component, Let $\hat{\theta}^*_{(i),h,(s,a)}$ be the best h-reachable model parameter corresponding to the best h-reachable models, $\widehat{P}^*_{\mathcal{L},t,h}$ (we drop the index $\mathcal{L}, t$ and d for brevity); using the set $\hat{\theta}^*_{(i),h,(s,a)}$ for every (s,a) pair results in the h-reachable value function $V^{d*}_{\mathcal{L},t,h}$. Note that $\hat{\theta}_{(i)}$ is the current model parameter. In the following, we make a relatively strict assumption to simplify the analysis: when they are not provided as inputs, the estimated values of L and $\Delta^{(i)}$ are correct in that they satisfy Assumption 2 and $\Delta^{(i)} \geq \sup_{s,a}|(\theta_{(i)} - \hat{\theta}_{(i)})^T\Phi_{(i)}(s,a)|$. This assumption can be relaxed by allowing the correctness to be violated with a constant probability. In such a case, we must force the random event to occur concurrently with the escape event, as discussed in the proof of Theorem 1 (the easiest way to do so is to take a union bound over the time steps until convergence). Furthermore, if we can specify the inputs L and $\Delta^{(i)}$, there is no need for this assumption.

Lemma 2. (Correctness of the h-reachable model interval) For the entire execution of Algorithm 2, for all state components $1 \leq i \leq n_s$, for all $t, h \geq 0$, and for all $(s,a) \in (S,A)$, the following inequality holds with probability at least $1 - \delta/2$:

$$\left|[\hat{\theta}_{(i)} - \hat{\theta}^*_{(i),h,(s,a)}]^T\Phi_{(i)}(s,a)\right| \leq I_h(\Phi_{(i)}(s,a), X_t).$$

Proof. Let $s^*_1 \in S'_{(s,a)}$ be the future possible observation from which the current model parameter $\hat{\theta}_{(i)}$ is updated to $\hat{\theta}^*_{(i),1,(s,a)}$. Then,

$$\begin{aligned}\left|[\hat{\theta}^*_{(i),1,(s,a)} - \hat{\theta}_{(i)}]^T\Phi_{(i)}(s,a)\right| &= \left|\Phi^T_{(i)}(s,a)(X_t^TX_t)^{-1}\Phi_{(i)}(s,a)[s^*_1 - \hat{\theta}^{*T}_{(i),1,(s,a)}\Phi_{(i)}(s,a)]\right| \\ &\leq \left|\Phi^T_{(i)}(s,a)D_t(\frac{1}{\lambda_{(1)}},\ldots,\frac{1}{\lambda_{(n)}}){U_t}^T\Phi_{(i)}(s,a)(\Delta^{(i)} + \varsigma(M)\sigma_{(i)})\right|.\end{aligned}$$

The first line follows directly from a result given by Cook (1977, Equation (5)). The second line is due to the following: with probability at least $1 - \frac{1}{2}e^{-\varsigma^2(M)/2}$,

$$\begin{aligned}s^*_1 - \hat{\theta}^{*T}_{(i),1,(s,a)}\Phi_{(i)}(s,a) \leq \theta^T_{(i)}\Phi_{(i)}(s,a) - \hat{\theta}^{*T}_{(i),1,(s,a)}\Phi_{(i)} + \varsigma(M)\sigma_{(i)} &\leq |\theta^T_{(i)}\Phi_{(i)}(s,a) - \hat{\theta}^{*T}_{(i),1,(s,a)}\Phi_{(i)}(s,a)| + \varsigma(M)\sigma_{(i)} \\ &\leq |\theta^T_{(i)}\Phi_{(i)}(s,a) - \hat{\theta}^T_{(i)}\Phi_{(i)}(s,a)| + \varsigma(M)\sigma_{(i)} \\ &\leq \Delta^{(i)} + \varsigma(M)\sigma_{(i)}\end{aligned}$$

where the first inequality follows that $\Pr(s_{t+1} > \theta^T_{(i)}\Phi_{(i)}(s,a) + \varsigma(M)\sigma_{(i)}) < \frac{1}{2}e^{-\varsigma^2(M)/2}$ and the third inequality follows the choice of the distance function d (i.e., the mean prediction with the best h reachable model is at least as good as that of the best $h-1$ model). We then separate the above into two terms with large and small eigenvalues: with probability at least $1 - \frac{1}{2}e^{-\varsigma^2(M)/2}$,

$$\begin{aligned}\left|[\hat{\theta}^*_{(i),1(s,a)} - \hat{\theta}_{(i)}]^T\Phi_{(i)}(s,a)\right| \leq |\Phi^T_{(i)}(s,a)U_tD_t(\frac{1}{\lambda_{(1)}},\ldots,\frac{1}{\lambda_{(j)}},0,\ldots,0){U_t}^T\Phi_{(i)}(s,a)\left(\Delta^{(i)} + \varsigma(M)\sigma_{(i)}\right) \\ + \Phi^T_{(i)}(s,a)U_tD_t(0,\ldots,0,\frac{1}{\lambda_{(j+1)}},\ldots,\frac{1}{\lambda_{(n)}}){U_t}^T\Phi_{(i)}(s,a)(\Delta^{(i)} + \varsigma(M)\sigma_{(i)})|.\end{aligned}$$

With w_t, we can rewrite part of the second term as $UD(0,\ldots,0,\frac{1}{\lambda_{(j+1)}},\ldots,\frac{1}{\lambda_{(n)}})U^T = UD(\frac{1}{\lambda_{(1)}},\ldots,\frac{1}{\lambda_{(n)}})U^Tw_t$. Then, with g_t and z_t, with probability at least $1 - \frac{1}{2}e^{-\varsigma^2(M)/2}$,

$$\left|[\hat{\theta}^*_{(i),1,(s,a)} - \hat{\theta}_{(i)}]^T\Phi_{(i)}(s,a)\right| \leq (\Delta^{(i)} + \varsigma(M)\sigma_{(i)})\left|\Phi^T_{(i)}(s,a)g_t\Phi_{(i)}(s,a) + \Phi^T_{(i)}(s,a)z_tw_t\Phi_{(i)}(s,a)\right|.$$

Thus, by applying the union bound for h, with probability at least $1 - \frac{h}{2}e^{-\varsigma^2(M)/2}$,

$$\begin{aligned}\left|[\hat{\theta}^*_{(i),h,(s,a)} - \hat{\theta}_{(i)}]^T\Phi_{(i)}(s,a)\right| &\leq h\left|[\hat{\theta}^*_{(i),1,(s,a)} - \hat{\theta}_{(i)}]^T\Phi_{(i)}(s,a)\right| \\ &\leq h(\Delta^{(i)} + \varsigma(M)\sigma_{(i)})\left|\Phi^T_{(i)}(s,a)g_t\Phi_{(i)}(s,a) + \Phi^T_{(i)}(s,a)z_tw_t\Phi_{(i)}(s,a)\right| \\ &\leq I_h(\Phi_{(i)}(s,a), X_t).\end{aligned}$$

For n_s components, the above inequality holds with probability at least $1 - \frac{n_sh}{2}e^{-\varsigma^2(M)/2}$ (union bound). For all $M \geq 1$, the above inequality holds with probability at least $1 - \frac{n_sh}{2}\sum_{M=1}^{\infty}e^{-\varsigma^2(M)/2}$ (union bound). Substituting $\varsigma(M) = \sqrt{2\ln(\pi^2M^2n_sh/(6\delta))}$, we obtain the statement. □

In Lemma 3 and Theorem 2, following previous work (Strehl and Littman 2008b; Li et al. 2011), we assume that an exact planning algorithm is accessible. This assumption will be relaxed by using a planning method that provides an error bound. We also assume that $R_{max} \leq c_1$, $\Delta^{(i)} \leq c_2$, and $\|\theta\| \leq c_3$ for some fixed constants c_1, c_2, and c_3. Removing this assumption results in these quantities appearing in the sample complexity, but produces no exponential dependence (thus, the sample complexity

remains polynomial). We assume that $M = O(\text{the number of samples})$, meaning that a planing algorithm calls $\mathbf{I}_h$ every iteration at most for a constant number of times. In the following, we use $\bar{n}$ to represent the average value of $\{n_{(1)}, ..., n_{(n_S)}\}$. Before analyzing the proposed algorithm, we re-derive the sample complexity of an existing PAC-MDP algorithm (Strehl and Littman 2008b; Li et al. 2011) for our problem setting.

Lemma 3. (Sample complexity of PAC-MDP) With an appropriate parameter setting, the PAC-MDP algorithm proposed by Strehl and Littman (2008b) and Li et al. (2011) has the following sample complexity:

$$\tilde{O}\left(\frac{n_S^2 \bar{n}^2}{\epsilon^5 (1-\gamma)^{10}}\right).$$

Proof. The proof follows directly from Theorems 1 and 3 in the previous work of Li et al. (2011). The only difference is that we need to take a union bound of different components $\Phi_{(i)}$ with varying domains, codomains and dimensions $n_{(s)}$. □

Theorem 2. (PAC-RMDP) Let $\mathcal{A}_t$ be a policy of Algorithm 2. Let $z = \max(h^2 \ln \frac{m^2 n_s h}{\delta}, \frac{L^2 n_S \bar{n} \ln^2 m}{\epsilon^3} \ln \frac{n_S}{\delta})$. Then, for all $\epsilon > 0$, for all $\delta = (0, 1)$, and for all $h \geq 0$,

1) for all but at most $m' = O\left(\frac{z L^2 n_S \bar{n} \ln^2 m}{\epsilon^3 (1-\gamma)^2} \ln^2 \frac{n_S}{\delta}\right)$ time steps (with $m \leq m'$), $V^{\mathcal{A}_t}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \epsilon$ with probability at least $1 - \delta$, *and*
2) there exists $h^*(\epsilon, \delta) = O(\mathcal{P}(1/\epsilon, 1/\delta, 1/(1-\gamma), |\text{MDP}|))$ such that $|V^*(s_t) - V^*_{\mathcal{L},t,h^*(\epsilon,\delta)}(s_t)| \leq \epsilon$ with probability at least $1 - \delta$.

Proof. Let $\tilde{V}^A$ be the internal value function used in Algorithm 2. We prove the statement by following the structure of the proof of Theorem 1. Define K, m, A_K, $\overline{V}$, and H in the same manner as in the proof of Theorem 1, and let the vector consisting of n_S estimation error intervals be $\mathbf{ER}(s,a) = (|(\theta_{(1)} - \hat{\theta}_{(1)})^T \Phi_{(1)}(s,a)|, \ldots, |(\theta_{(n_s)} - \hat{\theta}_{(n_s)})^T \Phi_{(n_s)}(s,a)|$. By following the proof of Theorem 1, with probability at least $1 - \delta/2$ (due to Lemma 2),

$$\begin{aligned} V^{\mathcal{A}}(s_t) &\geq \tilde{V}^{\mathcal{A}}(s_t) - \frac{R_{max}}{1-\gamma} Pr(A_k) - \frac{\epsilon}{3} - L\left(\max_{s,a} \|\mathbf{I}_h(s,a,X_{m'})\| + \max_{s,a} \|\mathbf{ER}(s,a)\|\right) \\ &\geq V^*_{\mathcal{L},t,h}(s_t) - \frac{c_1}{1-\gamma} Pr(A_k) - \frac{\epsilon}{3} - L\left(\max_{s,a} \|\mathbf{I}_h(s,a,X_{m'})\| + \max_{s,a} \|\mathbf{ER}(s,a)\|\right). \end{aligned}$$

In the second line, we used the assumption $R_{max} \leq c_1$. In the first line, $\max_{s,a} L\|\mathbf{I}_h(s,a,X_t)\|$ is the difference between $\tilde{V}^{\mathcal{A}}(s_t)$ and $V^*_{\mathcal{L},t,0}(s_t)$, and $\max_{s,a} L\|\mathbf{ER}(s,a)\|$ is the difference between $V^*_{\mathcal{L},t,0}(s_t)$ and $V^{\mathcal{A}}$. The second line follows from the fact that $\tilde{V}^{\mathcal{A}} \geq V^*_{\mathcal{L},t,h}(s_t)$ because of the correctness of I_h shown in Lemma 2 and the assignment of the most optimistic value within the interval $\mathbf{I}_h$ (based on Assumptions 1 and 2). We now impose an upper bound on $\|\mathbf{I}_h(s,a,X_t)\|$ and $\|\mathbf{ER}(s,a)\|$. Following a proof given by Li et al. (2011, Theorem 1) with the assumption $\Delta^{(i)} \leq c_2$ and $\|\theta\| \leq c_3$, with probability at least $1 - \frac{\delta}{4n_S}$,

$$\begin{aligned} \left|(\theta_{(i)} - \hat{\theta}_{(i)})^T \Phi_{(i)}(s,a)\right| &\leq \|\bar{q}\| \Delta_E(\hat{\theta}) + \|\bar{u}\| \leq \frac{2c_3\sqrt{n_{(i)} \ln m}}{m^{1/4}} \left(24 c_2 \ln \frac{8 n_S}{\delta}\right)^{1/4} + \frac{(2c_3\sqrt{\ln m} + 5)\sqrt{n_{(i)}}}{\sqrt{m}} \\ &\leq O\left(\frac{(n_{(i)} \ln m)^{1/2} (\ln(n_S/\delta))^{1/4}}{m^{1/4}}\right), \end{aligned}$$

where $\|\bar{q}\|, \|\bar{u}\|$ and $\Delta_E(\hat{\theta})$ are as defined by Li et al. (2011). Since $\Phi^T_{(i)} z_t (s_{t+1} - \hat{\theta}^T_{t+1} \Phi_{(i)}) = \hat{\theta}_{t+1} - \hat{\theta}_t$, there exist $\hat{\theta}$ and $\hat{\theta}'$ such that $\left\|\Phi^T_{(i)}(s,a) z_t (\Delta^{(i)} + \varsigma(M)\sigma_{(i)})\right\| \leq \|\hat{\theta} - \hat{\theta}'\| \leq \|\hat{\theta}\| + \|\hat{\theta}'\| \leq 2c_3$, where we use the assumption $\|\theta\| \leq c_3$. Then, following the proofs of Lemmas 11, 12, and 13 given by Auer (2002),

$$\begin{aligned} \frac{I_h(\Phi_{(i)}(s,a), X_t)}{h} &\leq (\Delta^{(i)} + \varsigma(M)\sigma_{(i)}) \sum_{j:\lambda_j \geq 1} \frac{\Phi_j^2}{\lambda_j} + \|\hat{\theta} - \hat{\theta}'\| \sqrt{\sum_{j:\lambda_j < 1} \Phi_j^2} \\ &\leq \frac{20(c_2 + \sqrt{2\ln(\pi^2 M^2 n_s h/(6\delta))}\sigma_{(i)}) n \ln(m)}{m} + 2c_3\sqrt{\frac{20 n_{(i)}}{m}} \\ &\leq O\left(\frac{\sqrt{n_{(i)}}}{\sqrt{m}} \ln m \sqrt{\ln(m^2 n_s h/(6\delta))}\right). \end{aligned}$$

If $h \leq O(\frac{m^{1/2} (\ln n_S/\delta)^{1/4}}{(\ln m)^{1/2} (\ln(m^2 n_s h/(6\delta)))^{1/2}})$, with probability at least $1 - n_s \frac{\delta}{4n_s} - \delta/2$,

$$V^{\mathcal{A}}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \frac{c_1 \Pr(A_k)}{1-\gamma} - \frac{\epsilon}{3} - O\left(\frac{L n_S^{1/2} \bar{n}^{1/2} (\ln m)^{1/2} (\ln(n_S/\delta))^{1/4}}{m^{1/4}}\right).$$

If $h > O(\frac{m^{1/2} (\ln n_S/\delta)^{1/4}}{(\ln m)^{1/2} (\ln(m^2 n_s h/(6\delta)))^{1/2}})$, with probability at least $1 - n_s \frac{\delta}{4n_s} - \delta/2$,

$$V^{\mathcal{A}}(s_t) \geq V^*_{\mathcal{L},t,h}(s_t) - \frac{c_1 \Pr(A_k)}{1-\gamma} - \frac{\epsilon}{3} - O\left(\frac{L h n_S^{1/2} \bar{n}^{1/2}}{\sqrt{m}} \ln m \sqrt{\ln(m^2 n_s h/(6\delta))}\right).$$

To have $\epsilon/3$ in the last term, we fix $m = O(\frac{L^4 n_S^2 \bar{n}^2 \ln^4 m}{\epsilon^4} \ln \frac{n_S}{\delta})$ for the former case, and $m = O(\frac{L^2 h^2 n_S \bar{n} \ln^2 m \ln(m^2 n_s h/(6\delta))}{\epsilon^2})$ for the latter case. Then, the rest of the first part of the statement follows from the proof of Theorem 1. That is, we can show that by applying the Chernoff bound, the escape event happens no more than the sample complexity in the statement with probability $1 - \delta/2$ unless the term $\frac{c_1 \Pr(A_k)}{1-\gamma}$ is negligible. Taking union bound on the failure probability, we obtain the sample complexity in the statement with probability at leat $1 - \delta$.

Finally, we consider the second part of the statement, following the proof in Theorem 1. Let $\hat{\theta}_{(i),h,(s,a)}$ be the future model parameter obtained by updating the current model $\hat{\theta}_{(i)}$ with h *random* future samples (h samples drawn from $P(S|s,a)$ for each $(s,a) \in (S,A)$). Based on the first part of the proof, $|(\theta_{(i)} - \hat{\theta}_{(i),h,(s,a)})^T \Phi_{(i)}(s,a)| \leq O\left(\frac{(n_{(i)} \ln(n_{min}+h))^{1/2} (\ln(n_S/\delta))^{1/4}}{(n_{min}+h)^{1/4}}\right)$ with probability at least $1-\delta$. Since $|(\theta_{(i)} - \hat{\theta}^*_{(i),h,(s,a)})^T \Phi_{(i)}(s,a)| \leq |(\theta_{(i)} - \hat{\theta}_{(i),h,(s,a)})^T \Phi_{(i)}(s,a)|$ (this directly follows the definition of $\hat{\theta}^*_{(i),h,(s,a)}$ and the choice of the distance function d), this implies that $h^*(\epsilon,\delta) = O(\frac{L^4 n_S^2 \bar{n}^2 \ln^2 m}{\epsilon^4} \ln \frac{n_S}{\delta})$ is sufficient. □

Corollary 3. (Anytime error bound) With probability at least $1-\delta$, if $h^2 \ln \frac{m^2 n_s h}{\delta} \leq \frac{L^2 n_S \bar{n} \ln^2 m}{\epsilon^3} \ln \frac{n_S}{\delta}$,

$$\epsilon_{t,h} = O\left(\sqrt[5]{\frac{L^4 n_S^2 \bar{n}^2 \ln^2 m}{t(1-\gamma)} \ln^3 \frac{n_S}{\delta}}\right),$$

and otherwise,

$$\epsilon_{t,h} = O\left(\frac{h^2 L^2 n_S \bar{n} \ln^2 m}{t(1-\gamma)} \ln^2 \frac{n_S}{\delta}\right).$$

Proof. The proof follows directly from Theorem 2 and the proof of Corollary 1. □

As in the discrete case, the anytime T-step average loss can be computed by summing the anytime errors as $\frac{1}{T}\sum_{t=1}^{T}(1 - \gamma^{T+1-t})\epsilon_{t,h,\delta}$. In addition, we can derive the explicit exploration runtime.

Corollary 6. (Explicit exploration runtime) With probability at least $1-\delta$, the explicit exploration runtime of Algorithm 2 is

$$O\left(\frac{h^2 L^2 n_S \bar{n} \ln m}{\epsilon^2 \Pr[A_k]} \ln^2 \frac{n_S}{\delta} \ln \frac{m^2 n_s h}{\delta}\right) = O\left(\frac{h^2 L^2 n_S \bar{n} \ln m}{\epsilon^3 (1-\gamma)} \ln^2 \frac{n_S}{\delta} \ln \frac{m^2 n_s h}{\delta}\right),$$

where A_K is the escape event defined in the proof of Theorem 2.

Proof. The proof follows that of Theorem 2. Compared to the sample complexity of Algorithm 2, z is replaced by h based on the proof of Theorem 2. □

A7. Experimental Settings for Continuous Domain

For each problem used in the main paper, we present more detailed descriptions of the experimental settings.

Mountain Car In the mountain car problem, the reward is negative everywhere except at the goal. To reach the goal, the agent must first travel far away, and must explore the world to learn this mechanism. To require a greater degree of exploration, we modify the original problem as follows: The agent obtains a reward equal to -0.9 around the initial position (position = [-0.6, 0.4]), and -1.0 everywhere else but at the goal. At the start of each episode, the agent is always at the bottom of the valley (position = -0.5) with zero velocity. Moreover, a small amount of Gaussian noise with standard deviation 0.001 is added to the velocity. Our model uses 10 grids of residual basis functions over the control signal and velocity as features. For the planning phase, we use a fitted value iteration with a 30×30 grid of residual basis functions. We set $\Delta^{(i)}$ and the corresponding parameter in the PAC-MDP algorithm to be 0.14, because the velocity is bounded in $[-0.07, 0.07]$. Each episode consists of 2000 steps, and we conduct simulations for 100 episodes.

Simulated HIV treatment This problem is described by a set of six ordinary differential equations (Ernst et al. 2006). An action corresponds to whether the agent administers two treatments (RTIs and PIs) to patients (thus, there are four actions). Two types of exploration are required: one to learn the effect of using treatments on viruses, and another to learn the effect of not using treatments on immune systems. Learning the former is necessary to reduce the population of viruses, but the latter is required to prevent the overuse of treatments, which weakens the immune system. We select the initial state to be unhealthy, following Ernst et al. (2006) and Pazis and Parr (2013). As in previous work, we assume that *noise-free* data can be obtained every five days. Unlike past studies, we assume that *noisy* data can be obtained a day after each instance of noise-free data is collected, with the noise term being $\zeta' \sim \mathcal{N}(0, 0.1)$. We add another noise term to represent the model error with $\zeta \sim \mathcal{N}(0, 0.01)$ for each dynamic state. For the model, we use the six states and the multiple of any two of these six states as features (i.e., the number of features is $6 + \binom{6}{2}$). For planning, we use a greedy roll-out method, as described by Adams et al. (2004, Section 5). We set $\Delta^{(i)}$ and the corresponding parameter in the PAC-MDP algorithm to be the average error among all the predictions and observations. Each episode consists of 1000 days, and we conduct simulations for 30 episodes.