Bayes-Optimal Reinforcement Learning for Discrete Uncertainty Domains

(Extended Abstract)

Emma Brunskill
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA
ebrun@cs.cmu.edu

ABSTRACT

An important subclass of reinforcement learning problems are those that exhibit only discrete uncertainty: the agent's environment is known to be sampled from a finite set of possible worlds. In contrast to generic reinforcement learning problems, it is possible to efficiently compute the Bayes-optimal policy for many discrete uncertainty RL domains. We demonstrate empirically that the Bayes-optimal policy can result in substantially and significantly improved performance relative to a state of the art probably approximately correct RL algorithm. Our second contribution is to bound the error of using slightly noisy estimates of the discrete set of possible Markov decision process parameters during learning. We suggest that this is an important and probable situation, given such models will often be constructed from finite sets of noisy, real-world data. We demonstrate good empirical performance on a simulated machine repair problem when using noisy parameter estimates.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Algorithms

Keywords

reinforcement learning, MDPs, POMDPs

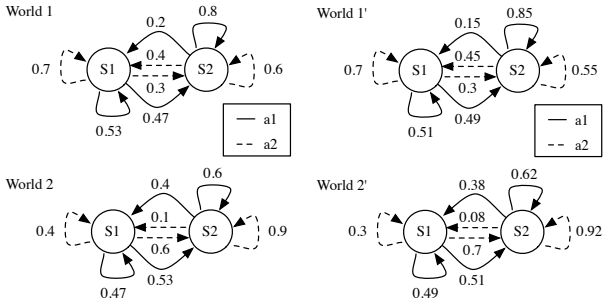
1. INTRODUCTION

Reinforcement learning (RL) is a critical challenge in artificial intelligence, because it seeks to address how an agent can autonomously learn to act well given uncertainty over how the world works. Model-based RL explicitly estimates parameters about the world dynamics and reward. Uncertainty over these parameters is typically allowed to be a continuous distribution. In contrast, there are many scenarios which are commonly represented by discrete uncertainty: the specific world is initially unknown, but there are only a finite set of possible worlds (which we represent by Markov decision processes). Such problems can be represented exactly as a finite-state partially observable MDP, where the discrete hidden state represents the true world (and associated parameters). While a related observation was made in passing by Poupart et al. [7] who noted that a discrete representation could be used to approximate continuous uncertainty, here we argue that many problems naturally exhibit finite uncertainty. For example, in customer relationship management, there may be several different types of customers, and the parameters of such customers can be estimated, but the type of a new customer is unknown.

Due to the finite nature of the uncertainty of these RL problems, we can use existing POMDP solvers to exactly compute a Bayes-optimal (or ϵ-optimal) policy. A Bayes-optimal RL policy is one that maximizes the expected discounted sum of future rewards over the specified time horizon, given an initial distribution of possible MDP model parameters. This is a different objective than Probably Approximately Correct (PAC) RL algorithms (e.g. [5, 2, 9]) which guarantee, with high probability, to select actions whose value is close to the value of the action that would be taken in the optimal policy if the MDP parameters were known, on all but a finite set of time steps. Though elegant, the number of time steps on which the algorithm may be far from optimal is often prohibitively large. To address this, practical instantiations of PAC RL algorithms typically involve a tuning parameter, resulting in good empirical performance, but eliminating theoretical guarantees. If we could solve for the Bayes-optimal policy by treating the problem as a POMDP [4], that would be appealing. However, in generic RL the model parameter values can be drawn from a real-valued set, this results in a continuous-state POMDP which are very challenging to solve, and prior Bayesian RL algorithms typically struggle to scale to large problems, and/or do not provide bounds on the computed policy's performance(e.g. [7, 8]).

However, it may often be possible to efficiently solve for an ϵ-Bayes-optimal policy in finite uncertainty domains. We first demonstrate the benefit of Bayes-optimal RL on an existing domain that naturally exhibits finite uncertainty. In the Wumpus grid world, an agent seeks to kill a wumpus without being first killed by the wumpus or falling into a pit. Our domain is almost identical to that described in [9], except that there are only 8 possible pit locations instead of

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.



(a) Discrete uncertainty RL (b) Erroneous parameters

Figure 1: The agent is placed in one of the two worlds, but it does not know which.

15. There are 3840 possible worlds, each with an associated wumpus location and set of pits; however the agent originally does not know which world it is in. We used the freely available APPL POMDP toolkit[1] to compute an e-Bayes-optimal policy (we set $e = 0.001$).

We compared to our approach to a PAC RL algorithm that computes a policy by adding a reward bonus to state-action pairs. This bonus is based on the variance of the possible hidden model parameters [9]. We focus our comparison to this variance-based bonus approach as the authors' approach outperformed a number of other approaches, including [1, 6].

In the Wumpus problem our Bayes-optimal policy has formal bounds on the performance, and empirically outperformed (mean=0.656, t-test $p < 0.001$) the variance bonus PAC RL approach without formal bounds (mean=0.478, tuning parameter=0.25), highlighting the benefit of Bayes-optimal RL. This, and many other, PAC RL approaches provide a fixed bonus for exploration, independent of the resulting possible benefit of such exploration, or the cost that may be incurred to perform this exploration, in contrast to Bayesian RL approaches.

2. IMPERFECT MODELS

We are interested in discrete uncertainty RL problems that capture real-world domains. In such environments, the possible models will generally be constructed from data. The model parameters estimated from the data will likely have a some error compared to the true generating parameters, due to limited data or local-optima finding fitting methods such as EM. For example, the true state of the world may be that the agent is acting in one of the two MDPs shown in Figure 1(a). However, the parameters of these two MDPs may have been estimated with some error, and the agent may think it is acting in one of the two MDPs shown in Figure 1(b). We can bound the error in the value function resulting from computing the value in a discrete uncertainty RL problem which has parameters that have some error relative to the true parameters:

THEOREM 1. *Let P denote a discrete uncertainty reinforcement learning problem $\langle S, A, R, \gamma, T_1, \ldots, T_M, b_0 \rangle$. In each transition model define $p(s'|s,a,m) = \theta_{sas'm}$. Let $\hat{P}$ be a second discrete uncertainty reinforcement learning problem $\langle S, A, R, \gamma, \hat{T}_1, \ldots, \hat{T}_M, \hat{b}_0 \rangle$. $\hat{\theta}_{sas'm'} = p(s'|s,a,m') = p(s'|s,a,m) + \epsilon_{sas'm}$, where $\sum_{s'} \epsilon_{sas'm} = 1$. $Q(b,s,a)$ is the optimal expected discounted sum of future reward from starting in belief state b and state s, and taking action a in RL problem P. $\hat{Q}(\hat{b},s,a)$ is the same quantity for the RL problem $\hat{P}$ for in belief state $\hat{b}$. Let*

$$\Delta_Q \equiv \max_{b,\hat{b},s,a} |Q(b,s,a) - \hat{Q}(\hat{b},s,a)|.$$

Then

$$\Delta_Q \leq \frac{\gamma V_{max} \max_{b,\hat{b},s,a} \sum_{s'} \left| \sum_i -\epsilon_{sas'i} b(i) + (\theta_{sas'i} + \epsilon_{sas'i})(b(i) - \hat{b}(i)) \right|}{1-\gamma}.$$

Given space limitations we omit the proof. In the worst case the bound provides little limitations. However, the bound is tight when no error is present: if $\epsilon_{sas'i} = 0 \ \forall i$, then $b(i) = \hat{b}(i)$ at all time steps, and $\Delta_Q = 0$, as expected.

We are interested in the empirical performance of a Bayes-optimal algorithm computed for a discrete uncertainty RL problem when the parameters provided are slightly different than the true domain MDPs' parameters. We have conducted preliminary experiments on a machine maintenance problem similar to that in [3]. These initial results suggest that a Bayes-optimal RL approach performs as well or better than the PAC RL variance bonus approach [9].

Our work is the first to examine Bayes-optimal RL in domains which inherently exhibit finite uncertainty. We have demonstrated that it is computationally tractable to compute Bayes-optimal policies in some such domains, and that such policies can perform significantly better than PAC RL approaches. We also provided a bound on the error of using slightly erroneous model parameters, which may be an important and common scenario in real-world situations.

3. ACKNOWLEDGMENTS

We thank Google for support.

4. REFERENCES

[1] J. Asmuth, L. Li, M. L. Littman, A. Nouri, and D. Wingate. A Bayesian sampling approach to exploration in reinforcement learning. In *UAI*, 2009.

[2] R. I. Brafman and M. Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *JMLR*, 3:213–231, 2002.

[3] E. Delage and S. Mannor. *Operations Research.*

[4] M. Duff. *Optimal learning: Computational procedures for Bayes-adaptive Markov decision processes.* PhD thesis, University of Massassachusetts Amherst, 2002.

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

[6] Z. Kolter and A. Y. Ng. Near-Bayesian exploration in polynomial time. In *ICML*, 2009.

[7] P. Poupart, N. Vlassis, J. Hoey, and K. Regan. An analytic solution to discrete Bayesian reinforcement learning. In *ICML*, 2006.

[8] S. Ross, B. Chaib-draa, and J. Pineau. Bayesian reinforcement learning in continuous POMDPs with application to robot navigation. In *ICRA*, 2008.

[9] J. Sorg, S. Singh, and R. L. Lewis. Variance-Based Rewards for Approximate Bayesian Reinforcement Learning. In *UAI*, 2010.

[1] http://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/