A Planning Algorithm for Predictive State Representations

Masoumeh T. Izadi and **Doina Precup**
School of Computer Science
McGill University
Montreal, Canada

Abstract

We address the problem of optimally controlling stochastic environments that are partially observable. The standard method for tackling such problems is to define and solve a Partially Observable Markov Decision Process (POMDP). However, it is well known that exactly solving POMDPs is very costly computationally. Recently, Littman, Sutton and Singh (2002) have proposed an alternative representation of partially observable environments, called predictive state representations (PSRs). PSRs are grounded in the sequence of actions and observations of the agent, and hence relate the state representation directly to the agent's experience. In this paper, we present a policy iteration algorithm for finding policies using PSRs. In preliminary experiments, our algorithm produced good solutions.

1 Predictive State Representation

We assume that we are given a system consisting of a discrete, finite set of n states S, a discrete finite set of actions A, and a discrete finite set of observations $\mathcal{O}$. The interaction with the system takes place at discrete time intervals. The initial state of the system s_0 is drawn from an initial probability distribution over states I. On every time step t, an action a_t is chosen according to some policy. Then the underlying state changes to s_{t+1} and a next observation o_{t+1} is generated. The system is Markovian, in the sense that for every action, the transition to the next state is generated according to a probability distribution described by an $(n \times n)$ transition matrix T^a. Similarly, for a given observation o and action a, the next observation is generated according to an $(n \times n)$ diagonal observation matrix O^{ao}, where O^{ao}_{ii} is the probability of observation o when action a is selected and state i is reached. Since we are interested in optimal control, rather than prediction, we also assume that there exists a set of reward vectors R^a for each action a, where R^a_i is the reward for taking action a in underlying state i.

PSRs are based on the notion of tests. A *test* is an ordered sequence of action-observation pairs $q = a_1 o_1 ... a_k o_k$. The *prediction* for test q is the probability of the sequence of observations $o_1 ... o_k$ being generated, given the sequence of actions $a_1 ... a_k$. The prediction for a test q given prior history h, denoted $p(q|h)$, is the probability of seeing the sequence of observations of q after seeing history h and taking the sequence of actions specified by q. For any set of tests Q, its prediction vector is:

$$p(Q|h) = [p(q_1|h), \ldots p(q_{|Q|}|h)]$$

A set of tests Q is a PSR if its prediction vector forms a sufficient statistic for the dynamical system, i.e., if all tests can be predicted based on $p(Q|h)$. Of particular interest is the case of linear PSRs, in which there exists a projection vector m_q for any test q such that

$$p(q|h) = p(Q|h)^T m_q$$

Littman et al. also define an *outcome* function u mapping tests into n-dimensional vectors defined recursively by: $u(\epsilon) = e_n$ and $u(aoq) = (T^a O^{a,o} u(q)^T)^T$, where ϵ represents a null test and e_n is the $(1 \times n)$ vector of all 1s. Each component $u_i(q)$ indicates the probability of the test q when its sequence of actions is applied from state s_i. A set of tests $Q = \{q_i | i = 1, 2, ..k\}$ is called linearly independent if the outcome vectors of its tests $u(q_1), u(q_2), ...u(q_k)$ are linearly independent. Using this definition, such a set Q can be found by a simple search algorithm in polynomial time, given the POMDP model of the environment. Littman, Sutton and Singh (2002) showed that the outcome vectors of the tests in Q can be linearly combined to produce the outcome vector for any test.

2 Policy evaluation using PSRs

We assume that we are given a policy $\pi : \mathcal{H} \rightarrow \mathcal{A}$ and that the initial start state of the system, s_0, is drawn according to the staring probability distribution I. If we consider a given horizon t, only a finite number of tests of length t are possible when starting from I. Let Ψ_t be this set of possible tests.

The value of a memoryless policy π with respect to a given start state distribution I is the expected return over all possible tests that can occur when the starting state is drawn form I and then behavior is generated according to policy π:

$$V^\pi = \sum_{q \in \Psi_t} V(q|\pi) = \sum_{q \in \Psi_t} P(q|I, \pi) R(q|I, \pi),$$

where $R(q|I, \pi)$ is the expected return for test q given that the initial state is drawn from I and policy π is followed.

Let U be the $(|S| \times |\Psi_t|)$ matrix formed by concatenating the outcome vectors for all tests in Ψ_t, and U^+ be its pseudoinverse. The columns of U define the probability of each test in Ψ_t when applied from each underlying state. Consequently, IU represents the probability of the tests in Ψ_t when starting in I.

For each action-observation combination ao, we can define a $(|\Psi_t| \times |\Psi_t|)$ projection matrix:

$$M^{ao} = (U^+ T^a O^{ao} U)^T$$

and a projection vector:

$$m^{ao} = (U^+ T^a O^{ao} e_n^T)^T.$$

Considering the probability distribution over tests that π generates and the expected return of a test q given I and π, we have:

$$\begin{aligned} V(q|I,\pi) &= \sum_{i=1}^{t} \frac{IU \prod_{j=1}^{i} (M^{a_j o_j})^T U^+}{IU m^T_{a_1 o_1 \ldots a_i o_i}} \\ &* \quad R^{a_i} P(a_i|\pi, a_1 o_1 \ldots a_{i-1} o_{i-1}) \end{aligned}$$

This evaluation method requires a large amount of precomputation, but it can be useful if a small horizon suffices to get a good policy.

3 Policy iteration for PSRs

Similar to POMDPs in PSRs we can define action-value functions on the level of tests. The action-value function for taking action a after test q can be computed as:

$$Q_i^\pi(q,a) = R^a u(q) + \sum_{o \in O} \sum_{q' \in \Psi_{i-1}} p(q'|qao) V(q'|\pi, u(qao))$$

Then, the policy of the agent can be improved by choosing an action greedily with respect to this action-value function:

$$\pi^*(t) = \arg\max_a Q_i^\pi(a,t)$$

The agent must, in other words, select the best policy tree rooted at each decision point and each time step. Therefore the total running time of the algorithm is $O(|S|^3(|A||\mathcal{O}|)^h)$ and the complexity of the algorithm is only single-exponential in the horizon time.

4 Experimental results

We experimented with a standard gridworld navigation task used in the POMDP literature (Cassandra, 1994; Parr& Russell, 1996). The environment is a (4×4) grid. The agent has four actions, N, S, E, and W, which change its location deterministically to one of the four neighboring states. There is one goal state, in the lower right corner, which generates a distinct observation and a reward of +1. All the other states are perceptually aliased and generate no reward. The initial probability distribution is uniform over all states except the goal. Taking any action in the goal state moves the agent uniformly randomly to one of the other states.

For this problem we have 6 possible combinations of action-observation (ao) pairs: 1-Nnothing,2-Wnothing,3-Enothing,4-Snothing,5-Egoal,6-Sgoal. The set of core tests

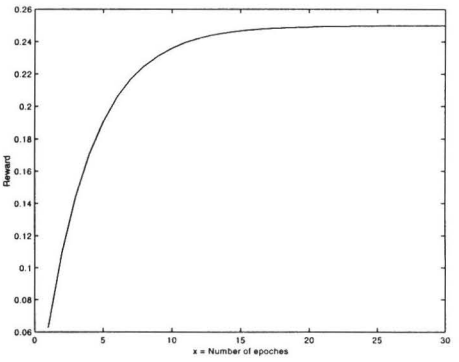

Figure 1: Policy quality vs. number of iterations for the 4x4 grid world

Q found consists of 16 linearly independent tests: 4, 5, 6, 15, 36, 46, 436, 446, 4436, 3336, 1336, 44336, 43336, 31336. We tried this problem for finite horizon case with a discount factor $\gamma = 0.8$. Figure 1 indicates the performance of the algorithm on this problem.

5 Conclusion and future work

The key difficulty for our planning algorithm is that the number of possible tests grows exponentially with the horizon length. Hence, the algorithm cannot be used for large problems or for the infinite horizon case. However, this is not worse than other existing exact solution methods for solving POMDPs. Our hope is that good approximations of the optimal solution can be found efficiently.

References

[Cassandra *et al.* , 1994] A. R. Cassandra, L. P. Kaelbling, and M. L. Littman. Acting Optimally in Partially Observable Stochastic Domains. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1994.

[Littman, 1996] . L. Littman. Algorithms for Sequential Decision Making. PhD thesis, Brown University, Providence, RI, March 1996.

[Littman *et al.* , 2002] M. L. Littman, R. S Sutton, and S. Singh. Predictive representations of state. *Advances in Neural Information Processing Systems 14. (Proceedings of the 2001 conference)*. MIT Press.

[Parr and Russell, 1995] R. Parr, S. Russell. Approximating Optimal Policies for Partially Observable Stochastic Domains. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (IJCAI-95).