Constant-Time Markov Tracking for Sequential POMDPs

Richard Washington
CRIN-CNRS & INRIA-Lorraine
B.P. 239
54506 Vandoeuvre Cedex
France
washingt@loria.fr

Introduction

Stochastic representations of a problem domain can capture aspects that are otherwise difficult to model, such as errors, alternative outcomes of actions, and uncertainty about the world. Markov models (Bellman 1957) are a popular choice for constructing stochastic representations. The classic MDP, however, does not account for uncertainty in the process state. Often this state is known only indirectly through sensors or tests. To account for the state uncertainty, MDPs have been extended to partially observable MDPs (POMDPs) (Lovejoy 1991; Monahan 1982; Washington 1996). In this model, the underlying process is an MDP, but rather than supplying exact state information, the process produces one of a set of *observations*.

The major drawback of POMDPs is that finding an optimal plan is computationally intractable for realistic problems. We are interested in seeing what limitations may be imposed that allow on-line computation. We have developed the approach of *Markov Tracking* (Washington 1997), which chooses actions that coordinate with a process rather than influence its behavior. It uses the POMDP model to follow the agent's state, and reacts optimally to it. In this paper we discuss the application of Markov Tracking to a subclass of problems called *sequential POMDPs*. Within this class the optimal action is not only computed locally, but in constant time, allowing true on-line performance with large-scale problems.

Markov decision processes

We assume that the underlying process, the *core process*, is described by a finite-state, stationary Markov chain. In MDPs with full observability, actions are chosen by a policy that maps states to actions. The utility of a state is the expected utility of taking the action associated with the state by the policy. The optimal policy maximizes the utility of each state.

However, in a partially observable MDP (POMDP), the progress of the core process is not known, but can only be inferred through a finite set of observations. Since the state is in general not localized to a single state, the knowledge about the process is reflected by a probability distribution over states.

In POMDPs, actions are chosen by a policy that maps state distributions to actions. The utility of a state distribution is the expected utility of taking the action prescribed by the policy for that state distribution. The optimal action is that which maximizes the expected utility for each state distribution.

Sequential POMDPs

A *sequential POMDP* is a restricted POMDP in which the state transitions in the underlying MDP are constrained to the same state or the "following" state.

Given the restriction on the model, updating the state distribution at each moment in time can be computed in $\mathcal{O}(N)$ for N states. In addition, the memory required for the transition and reward matrices is $\mathcal{O}(N \cdot A)$ for N states and A actions; the observation matrix requires $\mathcal{O}(N \cdot M \cdot A)$ memory for M observations. For Markov Tracking, the A terms disappear, reducing the required memory.

Markov Tracking

In Markov Tracking (Washington 1997), the goal is to find the best actions to coordinate with a process represented as a POMDP. This differs from the traditional goal of POMDPs, which is to find the optimal plan to control the process. In the case of coordination, we consider actions that do not directly influence the process. Instead, what is important is that the correct action is taken with respect to the actual process state. If the process is represented by a POMDP, then in general there is uncertainty about the process state, so the choice of action must take into account the possibility that the action is in fact not the best for each possible state, but is rather the best for the set of possible states taken together.

In the POMDP formalism, the lack of influence of actions on the underlying process means that the transition from one state to another is independent of the action, and the observation as well. What remains to distinguish one action from another is the reward function. In fact, an action with a greater immediate reward provides a higher overall plan value. This means that the optimal action, which is the action that gives the highest value, is in fact in this case the action with the greatest immediate reward. In the general POMDP case, this isn't necessarily true because of the difference in transition and observation probabilities.

The way Markov Tracking works is the following. The process transitions probabilistically and provides observations (also probabilistically). Using the POMDP transition function, the current state distribution is updated, and the optimal action is the action that maximizes the immediate reward for the belief state.

Markov Tracking for Sequential POMDPs

As described earlier, sequential POMDPs are a restricted form of POMDPs where state transitions lead to the same or the following state. This class of models can be solved significantly faster if we adopt the "windowing" method that will be described next.

In POMDPs, the state of knowledge of the current state of the process is represented by the belief state, which is a probability distribution over the set of states. Over time, this distribution can have many non-zero but vanishingly small elements, each of which must be taken into account when updating the belief state. However, if we limit the distribution to the k most probable states, by zeroing the rest and renormalizing the distribution, the computational complexity of the general POMDP case reduces from $\mathcal{O}(N^2)$ for N states to $\mathcal{O}(N)$.

For sequential POMDPs, given a distribution of k possible states, there are at most $2k$ states that could have a non-zero probability in the following time step (of which k will be retained). This means that the belief state update can be limited to those states, and in fact that makes the belief state update $\mathcal{O}(1)$ (constant).

Note however that there is some risk in limiting the belief state distribution to a fixed-size window. In particular, it is possible that an observation will be seen that is in fact impossible in any of the k states (because it comes from a state that was very improbable but in fact correct). This results in a null state distribution. We propose the following method to handle this (fortunately rare) occurrence. The transition probabilities indicate the evolution of the belief state given no observations, so the tracking process falls back on these in the presence of "impossible" observations. This allows the model of the process to evolve over time even in this case. In our experience this allows the process to re-orient itself quickly.

Discussion

The Markov Tracking approach can be applied to a number of problem domains. Sequential POMDPs are appropriate for domains that follow a trajectory over time. For example, the computer could have the job of following a spoken text and producing a subtitled text to accompany it. Or the computer could have the job of following a musical score and playing an accompaniment.

We plan to investigate these and other "real" applications, but for initial results, we constructed a number of randomly-generated scenarios to test our ideas. As expected, accuracy declines with an increase in noise and unpredictability, but that in general the tracking method is able to accurately follow the process. In fact, the errors seem to be mostly local errors caused by incorrect observations.

The power of Markov Tracking rests on its ability to take the best action with respect to the uncertainty of the precise state. So if the observations are generally reliable (as in the generated cases), the results are generally quite impressive. The tracking method remains "on course" even in the presence of noise and a restricted belief state. The size of the window can be altered to handle increasing levels of noise, with of course an accompanying increase in the constant factor of computation.

References

Bellman, R. 1957. *Dynamic Programming.* Princeton University Press.

Lovejoy, W. S. 1991. A survey of algorithmic methods for partially observed markov decision processes. *Annals of Operations Research* 28:47–65.

Monahan, G. E. 1982. A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science* 28(1):1–16.

Washington, R. 1996. Incremental Markov-model planning. In *Proceedings of TAI-96, Eighth IEEE International Conference on Tools With Artificial Intelligence.*

Washington, R. 1997. Markov tracking: Optimal, on-line coordination. Draft paper.