Applying Model-Based Search to Reinforcement Learning

Scott Davies **Andrew Y. Ng** **Andrew Moore**

A common approach to Reinforcement Learning in continuous state spaces involves using a function approximator to learn an approximation to the value function. Unfortunately, even when our agent is given a perfect model of the world, it is often prohibitively computationally expensive to fit a highly accurate value function; this problem is even worse when the agent must repeatedly update its dynamic programming solution online in situations where it is learning a model. Thus, on many occasions we are unable to compute more than a somewhat crude approximation to the true value function, and this can result in poor on-line performance. Here, we investigate the possibility of using online model-based search to alleviate this problem with reinforcement learning using approximate value functions. In particular, we examine the following possible algorithms:

Local Search: Given a value function, agents are typically made to execute a greedy policy using a one-step lookahead search, possibly using its learned model of the world for the lookahead step. We consider the natural extension of performing a a deeper search over limited trajectories from its current position, with the approximated value function being used as the heuristic evaluation function at the end of the search. This search provides the controller with more robust performance in the face of crudely approximated value functions, and (under certain assumptions) is formally equivalent to improving the value function in a certain sense.

Uninformed Global Search: This is a technique borrowed from robot motion planning, and is a breadth-first-search algorithm used for solving shortest-path-to-goal problems in continuous state spaces. From the agent's current position, it uses its model of the world to search for a complete, continuous trajectory to the goal. This is done by first dividing the state space up into a uniform grid. A search procedure similar to the one used by our Local Search algorithm is then used to find paths from one grid element to another, and multiple trajectories entering the same grid element are pruned. In this manner, it builds complete trajectories to the goal that can be trusted to the same extent that its model can be trusted. Failure to reach the goal by attempting to follow these trajectories can be attributed directly to inaccuracies in the agent's model. Using these trajectories can significantly improve the agent's online performance.

Informed Global Search: The uniformed global search procedure outlined above suffers from the fact that its search procedure often requires too much time and memory to be practical. However, we can use use an approximated value function as the heuristic for an A*-style search from grid element to grid element. Even when the approximated value function is relatively crude, this may dramatically reduce the fraction of the state space that is searched.

We investigate the strengths and feasibility of these approaches both in situations in which the agent already has a perfect model of the state space, and in situations in which the agent is learning a model from scratch on-line.