Symbolic Performance & Learning in Continuous Environments

Seth O. Rogers

University of Michigan AI Lab
1101 Beal Avenue
Ann Arbor, MI 48109-2110 USA
srogers@eecs.umich.edu

Introduction

We present an approach which enables an agent to learn to achieve goals in continuous environments using a symbolic architecture. Symbolic processing has an advantage over numerical regression techniques because it can interface more easily with other symbolic systems, such as systems for natural language and planning. Our approach is to endow an agent with qualitative "seed" knowledge and allow it to experiment in its environment.

Continuous environments consist of a set of quantitative state variables which may vary over time. The agent represents goals as a user-specified desired value for a variable and a deadline for its achievement. To determine the correct action given the current situation and goals, the agent maps the numbers to symbolic regions, then maps these regions to an action. The learning task of the agent is to develop these mappings.

Performance and Learning

Our system, SPLICE (Symbolic Performance & Learning in Continuous Environments), incorporates a performance module and a learning module. The performance module applies the agent's knowledge to its situation. First the agent maps the numeric variables to hierarchical symbolic regions of varying generality, both for the perceived state and the agent's goals. Then the agent searches the action mappings from specific to general for a match with the symbolic situation and goals. If there is no matching action mapping, the agent uses a qualitative domain model to suggest an action. Finally the agent takes the suggested action.

For learning, the agent waits until the deadline, then evaluates the action's effect to create a new action mapping (a new condition and a new action). Failures may be caused either by overgeneral goal or state conditions. In the first case, the new condition is the most general goal not achieved by the action. In the second case, the new condition is the most general state region different from the situation when the action was first learned. The new action is the linear interpolation of the two closest results straddling the current goal.

Results

Since SPLICE does not explicitly represent domain law equations, it performs about as well in very complex nonlinear domains as simple domains. SPLICE was tested on three successively more detailed and complex automotive simulations. Figure 1 illustrates its performance where the agent is trying to find the right throttle setting for a desired speed. At first, the agent requires several attempts, but over time performance improves. The added detail and complexity of domains 2 and 3 does not slow the learning rate.

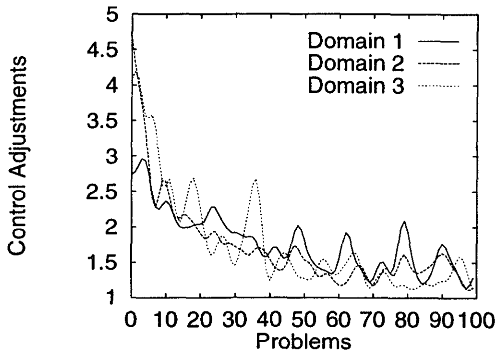

Figure 1: SPLICE performance in three domains.

Conclusion and Future Work

Instead of numerical techniques for adaptive control, SPLICE symbolically represents state variables and searches for mappings, allowing SPLICE to incorporate other symbolic systems, such as a qualitative reasoning module. Our current research has focussed on learning action mappings and left region mapping to a static binary division scheme. We are extending SPLICE in a number of dimensions, including multiple goals and hierarchical control structures. Our final objective is to demonstrate SPLICE in complex realistic environments, such as airplane flight.