Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence

Multi-Strategy Learning of Robotic Behaviours via Qualitative Reasoning

Timothy Wiley
School of Computer Science and Engineering
The University of New South Wales
Sydney, Australia 2052
timothyw@cse.unsw.edu.au

Introduction

When given a task, an autonomous agent must plan a series of actions to perform in order to complete the goal. In robotics, planners face additional challenges as the domain is typically large (even infinite) continuous, noisy, and non-deterministic. Typically stochastic planning has been used to solve robotic control tasks.

Particular success has been achieved with Model-Based Reinforcement Learning, such as Abbell *et al* (2010) which learnt a control policy for stable helicopter flight. Additionally, techniques such as Behavioral Cloning have been used to direct and speed up learning by providing knowledge from human experts (Michie, Bain, and Hayes-Michie 1990). The downside to such approaches is that the models and planners are highly specialised to a single control task. To change the control task, requires developing an entirely new planner.

The research in my thesis focuses on the problem of specialisation in continuous, noisy and non-deterministic robotic domains. It builds on previous research in the area, specifically using the technique of Multi-Strategy Learning (Sammut and Yik 2010). I am using Qualitative Modelling and Qualitative Reasoning to provide the generality, from which specific, Quantitative controllers can be quickly learnt. The resulting system will be applied to the iRobot Negotiator Robotic Platform (Figure 1), on the domain task of traversing rough terrain types.

Multi-Strategy Learning

Multi-Strategy Learning is motivated by the process humans typically use to complete complex control tasks. For example, if we are learning to drive a car that has a manual gear shift, our driving instructor does not say to us "Here is the steering wheel, the gear leaver, and the pedals. Play with them and figure out how to drive." Instead the instructor will provide to us a sequence of actions to perform, such as "To change gears, depress the clutch as the accelerator is released, followed by a gear change . . . " and so on. However, this sequence is insufficient as the instructor cannot tell us the "feel" of driving, that is, the precise speed and timing with which to perform these actions. We learn the "feel" by trial-and-error, while using the background knowledge to reason about our actions.

Sammut & Yik (2010) proposed a multi-strategy learning technique that combines learning and using background knowledge with trial-and-error learning. Their system architecture is shown in Figure 2. First an Action model of the system is developed, where actions are parameterised. (For example, on a car an action might be drive, with parameter speed). Given a control task, a planner produces a sequence of actions to complete the task. The constraint solver bounds the parameters of the action. Finally a trial-and-error learner determines the optimal value for each parameter.

Sammut & Yik successfully applied their method to learning a bi-pedal gait, showing that given a plan for a gait, the robot could constrain and optimise the parameters within an average of 40 trials. However, they used a simplified planner that does not generalise to other domains. My research fills in the gap by replacing the action model and planner with Qualitative Simulation.

Qualitative Simulation

To produce a plan of actions to perform, fundamentally the planner must be able to reason about the physics of system, in particular the effects of actions. Qualitative Simulation (QSim) (Kuipers 1986) is an effective way to achieve this, and is best illustrated with an example.

Figure 3 models part of the Negotiator robot; two components - the base and the flipper. We have two variables, the

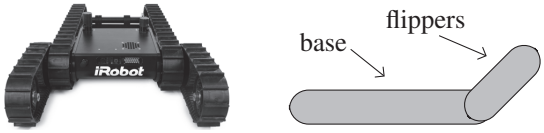

Figure 1: iRobot's Negotiator Platform

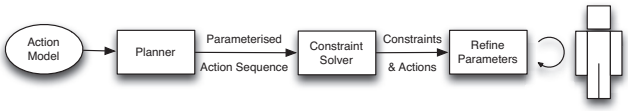

Figure 2: Multi-Strategy Learning Architecture

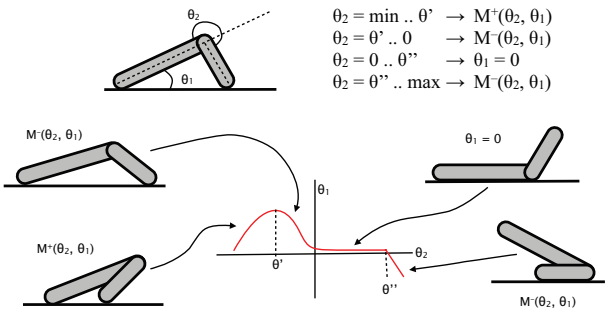

Figure 3: Qualitative Model and States for Negotiator

angle of the base to the ground θ_1, and angle of the flipper to the base θ_2. Each variable is assigned two values, a region (the variable's relation to landmark values) and a magnitude (the variable's rate of change increasing (inc), steady (std), or decreasing (dec)).

Initially let us suppose the robot is lying flat on the ground. The qualitative state of the system is then $\theta_1 = 0/std, \theta_2 = 0/std$. Now, we take an action - decrease θ_2. From Figure 3 we can see the resulting qualitative state is $\theta_1 = 0 \ldots max/inc, \theta_2 = \theta' \ldots 0/dec$. This has occurred since the qualitative model that relates θ_1 and θ_2 is $\mathrm{M}-(\theta_1, \theta_2)$. That is, as θ_1 increases, θ_2 decreases, and vice-versa. Then from this new state, we could reach another state and so on.

Thus, QSim takes as input a given qualitative state plus a qualitative model and determines all possible qualitative states the system could transition to. From this QSim produces a directed graph of connected qualitative states.

My research involves using this graph to develop the parameterised sequence of actions to be passed onto the trial-and-error learner.

- Defining the control problem as finding a way to traverse the QSim graph from an initial qualitative state to a goal.
- Introducing "Action" Variables.

An action variable is one the robotic system can directly set. For example, on Negotiator θ_2 is an action variable, as its value can be directly set, whereas θ_1 is not. Additionally every non-action variable must be related in the qualitative model directly or in-directly to at least one action variable.

To generate the parameterised plan I find a path through the QSim graph, then list in order the action variables that change in each qualitative state. Where a variable has the same magnitude of change over multiple states, these are combined into one action. For example to go from the state $\theta_1 = 0/std, \theta_2 = 0/std$ to the state $\theta_1 = 0/std, \theta_2 = \pi/std$ requires transitioning through at least four qualitative states, but produced an action sequence with only action - set θ_2, as θ_2 is always increasing in each state.

Problems with QSim and Planning

There are three key complications to planning with QSim:

1. Qualitative Models change as the Qualitative State changes.

Figure 4: Climbing a small step (From left to right)

2. Qualitative Simulation is non-deterministic.
3. Incorrect paths can be produced.

As shown in Figure 3, the qualitative model changes with the qualitative state. As additional variables are introduced, (such as the position and velocity of the robot), or obstacles (such as the step in Figure 3) so are more relationships between variables. This makes it difficult to determine which qualitative model to provide to QSim. It also means that in a given qualitative state, multiple models must be tried, greatly increasing the number of qualitative state in the graph, slowing planning. Solving this problem is my current area of work and is incomplete. I expect by the AAAI doctoral consortium I will have largely solved this problem.

The QSim algorithm is non-deterministic. That is, it may be possible to reach two different qualitative states after performing an action. This can make planning problematic since you may not be able to control which option is taken. I have yet to address this problem.

Finally, incorrect paths can be produced as a result of using qualitative variables. Consider the task of climbing the step in Figure 4. If the step is too high, the task cannot be achieved. However QSim cannot determine this as it does not use any quantitative information in its computations. This qualitative-quantitative boundary is an unanswered problem, and it may be possible to solve in the constraint solver.

Future Work

Future work for my thesis includes addressing non-determinism, investigating efficient methods to search the QSim graph (currently I use a iterative-deepening search), further exploring the qualitative-quantitative boundary, and learning the Qualitative Models.

References

Abbeel, P.; Coates, A.; and Ng, A. Y. 2010. Autonomous Helicopter Aerobatics through Apprenticeship Learning. *The International Journal of Robotics Research* 29(13):1608–1639.

Kuipers, B. 1986. Qualitative Simulation. *Artificial Intelligence* 29(3):289–338.

Michie, D.; Bain, M.; and Hayes-Michie, J. 1990. Cognitive Models from Subcognitive Skills. In *Knowledge-Based Systems in Industrial Control*. London: Peter Peregrinus. 71–99.

Sammut, C., and Yik, T. F. 2010. Multistrategy Learning for Robot Behaviours. In Koronacki, J.; Ras, Z.; Wierzchon, S.; and Kacprzyk, J., eds., *Advances in Machine Learning I*. Springer Berlin / Heidelberg. 457–476.