Toward the Autonomous Acquisition of Robot Skill Hierarchies (poster abstract)

George Konidaris
and Andrew Barto
Autonomous Learning Laboratory
Department of Computer Science
University of Massachusetts Amherst
Amherst MA 01002
{gdk, barto}@cs.umass.edu

I. Introduction

The design and coordination of independent specialized skill units (often called action primitives) is fundamental to modern robotics. However, a robot that must act in a complex environment over an extended period of time should do more than just use existing skills: it should learn new skills that increase its capabilities and facilitate later problem solving. Although robots exist that can learn a skill given a reward function and hand-engineered state space, none exist today that display truly autonomous skill acquisition. Reinforcement learning [1] is a natural fit for the robot skill acquisition problem, and with the recent development of the options framework [2] provides a principled approach to hierarchical learning and planning and an appropriate model for a robot skill unit (an *option*) that captures the essential elements required for control.

However, although a great deal of research exists on autonomously acquiring skills and skill hierarchies in reinforcement learning, that work has mostly focused on small, discrete domains. In this paper we outline our work on acquiring skill hierarchies in continuous reinforcement learning domains.

II. Skill Acquisition in Continuous Domains

Central to research in hierarchical reinforcement learning is the development of methods whereby an agent can discover new skills autonomously. Although several methods exist for skill discovery in discrete domains, none are immediately extensible to or have been successfully applied in continuous domains.

We introduce skill chaining, a skill discovery method for reinforcement learning agents in continuous domains. Skill chaining produces chains of lightweight skills leading to a salient event—where salience can be defined simply as an end-of-task reward, or as a more sophisticated heuristic (e.g., an intrinsically interesting event [3]). The goal of each skill in the chain is to reach a state where its successor skill can be executed. This breaks a solution trajectory up into a sequence of modular sub-skills, each of which can be learned separately using its own function approximator, which allows for the construction of complex policies that would be infeasible to represent using a single value function.

Figure 1 shows the performance improvement obtained using skill chaining in PinBall, a dynamic, four dimensional control task. Agents using skill chaining are able to learn better and more consistent policies than those using flat learning, and when the resulting skills are given to "blank slate" agents they find better policies than learned flat policies within a few episodes. Figure 2 shows a sample solution to the PinBall domain with each learned skill shown in a different color.

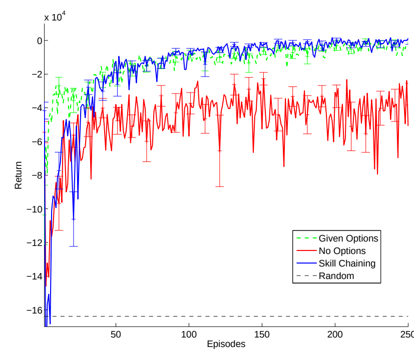

Fig. 1. Performance in the Pinball domain (averaged over 100 runs) for agents with a random policy, agents employing skill chaining, agents with given options, and agents without options.

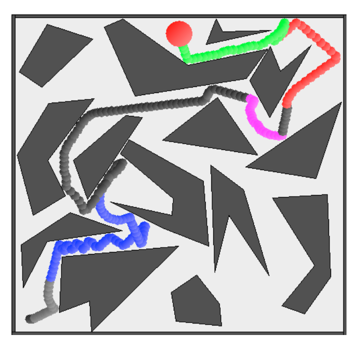

Fig. 2. A good solution to the Pinball domain, showing the acquired skills executed along the sample trajectory in different colors.

The notion of parameterizing controllers so that they allow each other to be executed has been present in robotics for a long time, known as pre-image backchaining or sequential composition [4]. In that work the controllers are given; our work can be thought of as learning controllers that are suitable for sequential composition. This method can also be easily generalized to multiple solution trajectories, where it finds skill trees rather than skill chains.

III. Scaling Up: Abstraction Selection

Although skill chaining provides a method for autonomous skill acquisition in continuous domains, real-time learning is extremely difficult in high-dimensional spaces. Learning often only becomes feasible when the problem is posed in the right space—one in which a small number of relevant (often heavily preprocessed) features are present, and a large number of irrelevant features are not. Because of this, successful robot learning research has made use of carefully designed problem-specific state spaces.

In reinforcement learning, the use of a small relevant state space to solve an initially high-dimensional problem is termed *abstraction*. Abstraction is considered crucial to scaling up, because it can render a high-dimensional problem feasible by reducing it to a small set of variables that express the problem's intrinsic dimensionality. However, learning an abstraction from data is an expensive process that is unlikely to be feasible in the timeframe in which we would like to learn a single skill.

We propose that an autonomous robot should be equipped with a *library* of useful abstractions from which it can select when it decides to learn a new skill. Our work introduces an incremental algorithm that selects an appropriate abstraction for a skill given a set of successful sample trajectories (as could be obtained by, for example, initial exploration. imitation, or a kinematic planner). The algorithm returns a policy fitted to the sample trajectory in the selected space, avoiding the need to start learning from scratch. It has been shown to select an appropriate abstraction on a physically realistically simulated mobile robot [5] and in a high-dimensional synthetic domain [6].

When used in conjunction with skill chaining, abstraction selection results in a chain of skills, each defined using their own abstraction. Thus, we may be able to solve an intrinsically high-dimensional problem by breaking it into sub-skills, each of which can be solved using only a few state variables. Thus, a complex human task such as driving to work, that seems infeasible to learn as a single overall problem, might be broken into a series of small subtasks (unlocking the car, starting the car, navigating to work, parking, walking inside, etc.), each of which is manageable on its own.

Choosing an appropriate representation is critical to the successful application of reinforcement learning to real world problems, but to achieve true autonomy that choice must be made by the robot, not its designer. Abstraction selection shifts the state design problem from one of designing a problem-specific state space to one of designing a library of state spaces sufficient to deal with any skill that a robot might decide to learn. This results in some significant benefits. It shifts the design element out of the robot's control loop, thus removing an obstacle to autonomous learning. We can use knowledge of the class of tasks the robot faces to design and constrain those abstractions. We can also include many abstractions, since selecting one is very much faster than learning a skill, and if necessary selection can be parallelized since each abstraction can be processed completely independently. Finally, we may be able to learn libraries of abstractions over the lifetime of the robot, removing the abstraction design obstacle and allowing us to use a large amount of data over a long period of time.

IV. Building Skill Hierarchies

Our current work addresses how an agent can use the skills it has acquired through repeated experience in a continuous domain to build small, discrete state abstraction models of that domain suitable for use in planning. Given a set of skills learned using skill chaining, an agent can build an abstracted state graph where vertices are state regions where a skill can be executed and edges are skill executions. When using skill chaining, the resulting graph is well formed: it is connected, all option executions change state, the only states that do not have incoming edges are start states, and the only states that do not have outgoing edges are goal states. Furthermore, the resulting structure contains the information required for performing high-level abstract planning to achieve a given goal, and for building even higher level state abstractions if necessary, using existing methods for temporal abstraction in discrete domains.

V. Summary and Future Work

This research has introduced skill chaining, a method for autonomously discovering new skills in continuous reinforcement learning domains; abstraction selection, which aims to scale learning up to high-dimensional spaces by selecting the correct abstraction for use in learning a new skill; and briefly summarized ongoing research on building discrete abstract models of continuous domains given a set of skills acquired using skill chaining. In the future, we aim to validate these models on a real robot (the uBot-5, developed at the Laboratory for Perceptual Robotics at the University of Massachusetts Amherst) to attempt to achieve truly autonomous hierarchical skill acquisition.

References

[1] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[2] R. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, vol. 112, no. 1-2, pp. 181–211, 1999.

[3] S. Singh, A. Barto, and N. Chentanez, "Intrinsically motivated reinforcement learning," in *Proceedings of the 18th Annual Conference on Neural Information Processing Systems*, 2004.

[4] R. Burridge, A. Rizzi, and D. Koditschek, "Sequential composition of dynamically dextrous robot behaviors," *International Journal of Robotics Research*, vol. 18, no. 6, pp. 534–555, 1999.

[5] G. Konidaris and A. Barto, "Sensorimotor abstraction selection for efficient, autonomous robot skill acquisition," in *Proceedings of the 7th IEEE International Conference on Development and Learning*, 2008.

[6] ——, "Efficient skill learning using abstraction selection," in *Proceedings of the Twenty First International Joint Conference on Artificial Intelligence*, July 2009.