Generalizing and Categorizing Skills in Reinforcement Learning Agents
Using Partial Policy Homomorphisms

Srividhya Rajendran, Manfred Huber
Department of Computer Science and Engineering
The University of Texas at Arlington
P.O. Box 19015, Arlington, Texas 76019
srividhya.rajendran@mavs.uta.edu, huber@cse.uta.edu

Introduction

A life long learning agent has to continuously learn, adapt, and perform tasks. However, making real time decisions without control knowledge becomes easily intractable.

Biological systems encounter similar problems and still learn to perform complex tasks in a dynamic environment. Developmental theories (Lakoff 1987; Mandler 1992) suggest that they do this by learning to abstract information and to form reusable skills and concepts, allowing them to apply this knowledge to more complex tasks.

Artificial Intelligence (www.aaai.org). All rights reserved. Copyright © 2009, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Policy Homomorphism Framework

Figure 1 shows the overall learning framework. Here the learning component learns skills for task instances using Q-learning. The policy abstraction component uses these to form generalized skills and representational concepts as general policies and corresponding mapping functions using the policy homomorphism framework. These generalized skills are added to the agent’s skill set and subsequently used to learn novel tasks or to perform similar tasks in new situations. When the agent invokes a generalized policy in an applicable state (as identified by the mapping function), it executes it until it terminates.

Figure 2 shows a grid world example with two policies and the resulting general policy. To derive this, the agent uses the policy homomorphism concept to identify similar tasks and to construct the general policy. Here, a policy \( \pi \) in a finite SMDP with state space \( S \) and absorbing states \( S_A \) is defined by \( < \pi, S, S_I, S_T, S_G > \) where \( \pi(s, a) : S \times A \rightarrow [0,1] \) maps states to action probabilities, \( S_I \subseteq S \) is the state set for \( \pi \) and \( S_T \subseteq S - S_A \), \( S_G \subseteq S_T \) are the initiation, terminal, and goal state sets, respectively, with \( S_A \subseteq S_T \).

Definition 1: A Policy Homomorphism \( f : \pi \rightarrow \pi' \) is a surjection from base policy \( \pi \) to an abstract policy \( \pi' \). \( f \) is defined by surjections \( (h : S_{\pi} \rightarrow S_{\pi'}, g_{\pi} : A_{\pi} \rightarrow A'_{\pi'}) \) and functions \( g_{\pi} = A_{\pi'} \times s \rightarrow [0,1] \) over the state and action sets for \( \pi \), \( S_{\pi} \subseteq S \) and \( A_{\pi} = \{ a \in A \mid \exists s \in S_{\pi} : \pi(s, a) > 0 \} \), such that the following holds: i) For each state-action pair \( (s, a) \):

\[ \pi'(h(s), g_{\pi}(a)) = g_{\pi}(s, a) \]

ii) For each state pair \( (s_i, s_j) \):

\[ \sum_{s_i, a} \pi'(h(s), g_{\pi}(a))T(h(s), b, h(s)) = \sum_{s_i, a} \pi(s, a)T(s, a, s_j) \]

where \( T \) and \( T' \) are the transition probabilities in the base and in the abstract policy, respectively.
A complete policy homomorphism requires that every state in a given policy be mapped onto a particular state in the abstract policy. As a result it does not allow abstraction of policies that are only partially homomorphic.

**Definition 2: A Partial Policy Homomorphism**

\( f : \pi_p \rightarrow \pi'_p \) is a surjection from a partial base policy \( \pi_p \) to an abstract policy \( \pi'_p \), where \( f \) is defined by a tuple of surjections, \((h : S_p \rightarrow S'_p, g : A_p \rightarrow A'_p)\) and functions \( s_h = A_p \times S_p \rightarrow [0,1] \) over the state and action sets of \( \pi_p \), \( S'_p \subseteq S_p \) and \( A'_p \subseteq A_p \) such that: i) For all \((s, a)\): \( \pi_p(h(s), g(a)) = g_f(h(s), a) \). ii) For each state pair \((s_i, s_j)\) with \( s_i \in S_p, h(s_j) \in S'_p \):

\[
\sum_{a \in A'_p} \pi'(h(s_i), b) T'(h(s_i), b, h(s_j)) = \sum_{a \in A_p} \pi(s_i, a) T(s_i, a, s_j)
\]

where \( T \) and \( T' \) are the transition probabilities for the base policy and the abstract partial policy, respectively. \( \Sigma \)

Both complete and partial policy homomorphisms can be applied to derive abstract policies. However, to ensure that the general policy captures the objective of the underlying policies, this work limits the application to goal based policies by extending the definitions.

**Definition 3: A Goal Based Policy Homomorphism**

\( f : \pi_G \rightarrow \pi'_G \) is a policy homomorphism (complete or partial) that fulfills: i) All goal states of the base policy are present as goal states in the abstract policy. ii) For all states \( s' \) in the homomorphism: \( S' \cap (S - S' S_p) = \emptyset \). iii) All non-goal states in \( \pi'_G \) are either terminal states or have a non-zero probability to lead to a goal under \( \pi'_G \). \( \Sigma \)

**Experiments and Results**

The approach is demonstrated using a set of grid world navigation tasks with deterministic forward and turn actions and states represented by the agent’s location and orientation and the locations of the four nearest doors. First the set of grid worlds shown in Figure 3(a) is used to learn a set of five basic policies to reach particular doorways.

Once the agent has learned the basic policies, it uses a greedy algorithm to extract a general policy for “Reach Door” tasks using policy homomorphisms. Starting from the goal state and adding one state at a time, this algorithm builds a set of state and action mappings that classify the states of each policy in terms of the corresponding state in the general policy and that determine the corresponding action. The algorithm continues until the addition of states no longer increases the expected utility of the policy.

To evaluate the utility of the abstracted policy it is used to learn new tasks in the novel grid world in Figure 3(b). Figure 4 shows the resulting learning curves with (grey) and without (black) the general “Reach Door” policy. Each curve represents the mean reward per trial averaged over 30 learning runs with confidence intervals indicating one standard deviation. These curves show significantly improved learning times with the generalized policy.

**Conclusion and Future Work**

This paper proposes a new approach for skill and concept abstraction using policy homomorphisms. The experiments demonstrate that reuse of the general policy and the corresponding mapping functions reduces the learning time for new tasks. Further, it allows the agent to compress the state space by abstracting task specific information. While this paper presents this approach in deterministic domains, we are currently extending it to stochastic domains.

**References**


