Temporal Abstraction in Reinforcement Learning with Proximity Statistics

Rico Jonschkowski RICO.JONSCHKOWSKI@TU-BERLIN.DE
Technische Universität Berlin, Marchstraße 23, 10587 Berlin, Germany

Marc Toussaint MARC.TOUSSAINT@INFORMATIK.UNI-STUTTGART.DE
Universität Stuttgart, Universitätsstraße 38, 70569 Stuttgart, Germany

Abstract

Automatic discovery of temporal abstractions is a key problem in hierarchical reinforcement learning. We propose a new approach to this problem by learning a time marginalized transition probability—which we call *proximity statistics*—from successful trajectories for related tasks. Viewing the proximity statistics as state values allows the agent to generate greedy policies from them. Making the statistics sparse and combining proximity estimates by *proximity propagation* can substantially accelerate planning compared to value iteration while keeping the size of the statistics manageable.

1. Introduction

Hierarchical reinforcement learning has introduced the idea of temporal abstraction—reasoning at different time scales—to reinforcement learning. While the benefits of temporal abstraction have been demonstrated, the automatic discovery of such abstractions remains an open research question (Bakker and Schmidhuber, 2004).

Our approach to this question is most closely related to the work of Stolle and Precup (2002). Being able to sample successful trajectories for related tasks (rational trajectories), they identify bottlenecks in the state space and use them as subgoals for options. Our work is also based on ideas of transit-node routing, an approach of Bast et al. (2007) to find shortest paths in large deterministic road networks. Using precomputed distances between certain pairs of nodes, they are able to find shortest paths in linear time.

We combine these ideas to generate symmetric temporal abstractions based on trajectories in a stochastic world. Instead of computing distances, we estimate the temporal proximity of states, which can be interpreted as state value. These proximity statistics can be made sparse by not considering all state pairs but focusing on important transit-states. Through proximity propagation these estimates can be combined and used for planning. Our experiments show that this method allows to substantially accelerate planning performance by reusing information from rational trajectories.

2. Temporal Abstraction with Proximity Statistics

The *temporal proximity* $\Omega(s, s')$ is the time marginalized transition probability from s to s', which is equal to the expected discounting from s to s' if both states are visited by the trajectory in this order:

$$\Omega(s, s') = \mathop{\mathrm{E}}_{s_{0:T}} \left\{ \gamma^{d_{s_{0:T}}(s, s')} \mid (s, s') \in s_{0:T} \right\} , \quad (1)$$

where γ is the time discount and $d_{s_{0:T}}(s, s')$ is the number of steps from s to s' on the trajectory $s_{0:T}$.

The *proximity statistics* $\hat{\Omega}(s, s')$ are an estimate of the temporal proximity from a set of rational trajectories $\{s_{0:T}\}$ replacing the expectation by an average. When no trajectory of this set goes from s to s', $\hat{\Omega}(s, s') = 0$.

2.1. Proximity is Goal State Value

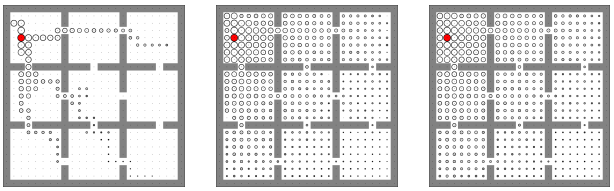
In reinforcement learning the value of state s is

$$V(s) = \mathop{\mathrm{E}}_{s_{0:T}} \left\{ r_0 + \gamma r_1 + \ldots + \gamma^T r_T \mid s_0 = s \right\} . \quad (2)$$

Considering only draining goal state tasks with no intermediate rewards, the state value simplifies to

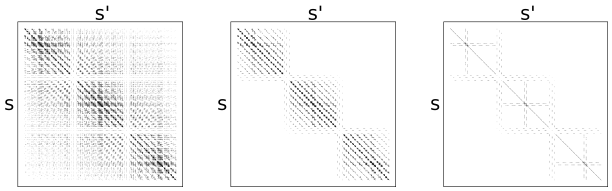
$$V(s) = \mathop{\mathrm{E}}_{s_{0:T}} \left\{ \gamma^T \mid s_0 = s, s_T = s^\star \right\} , \quad (3)$$

which is equal to the temporal proximity $\Omega(s, s^\star)$ from s to the goal state $s^\star$.



(a) $\forall s : \hat{\Omega}_1(s, s^\star)$ (b) $\forall s : \hat{\Omega}_2(s, s^\star)$ (c) $\forall s : \hat{\Omega}_3(s, s^\star)$

Figure 1. Proximity statistics improved with proximity propagation. The proximity statistics $\hat{\Omega}_1(s, s^\star)$ are estimated from 1000 trajectories. Only two propagations are required to give a good estimate of the temporal proximity $\hat{\Omega}_3(s, s^\star)$ from all states s to the goal state $s^\star$ (red dot). This estimate can be viewed as state value and used to generate a greedy policy to reach $s^\star$.



(a) Complete (b) Local (c) Transit

Figure 2. Proximity statistics $\hat{\Omega}(s, s')$ estimated from 1000 trajectories as a table, s denotes the row- and s' the column index, values go from 0 (white) to 1 (black). (a) shows the full proximity statistics. (b) shows the local proximity statistics that are estimated from the trajectories cut at the transit states. (c) shows the (local) transit proximity statistics where only the values from the transit-states (rows in the table) and to the transit-states (columns in the table) are estimated.

Thus, proximity statistics can be viewed as a set of value functions for different goal states and used to generate policies to reach them.

2.2. Proximity Propagation

We usually do not have enough trajectories to estimate the full proximity statistics. Through *proximity propagation*, proximity estimates can be combined:

$$\forall s : \hat{\Omega}_{i+1}(s, s') \leftarrow \max_{s^+} \left[\hat{\Omega}(s, s^+) \hat{\Omega}_i(s^+, s') \right] , \quad (4)$$

which is illustrated in figure 1. The correct temporal proximity can be shown to be a fixed point of proximity propagation.

2.3. Sparse Proximity Statistics

The size of the full proximity statistics scales quadratically with the number of states. However, proximity propagation allows to compute proximity information from sparse proximity statistics.

Thus, the statistics can be made sparse intentionally to save resources for computing and storing them. There are many ways to sparsify the statistics. We explain some ways that are based on bottlenecks in state space (transit-states) here. Transit-states can be found similar to subgoals (Stolle and Precup, 2002). In our grid world example, they correspond to the hallways.

The proximity statistics can be made *local* by cutting the trajectories at the transit-states before estimating the statistics. When the transit-states are hallways, local proximity statistics only include pairs of states in the same room. *Transit sparsification* is a subset of this where the proximity is only estimated to the transit-states and from the transit-states to all other states. Interestingly, this can be viewed as an extension of options defined by subgoals, which correspond to columns in this table, describing a pseudo-value function to reach their subgoal.

3. Experiments

At the workshop, we will present experiments in a stochastic grid world that compare the planning acceleration and memory requirements for different sparse proximity statistics. Transit sparsified proximity statistics allow to decrease the number of planning iterations to about one half compared to plain value iteration while the size of the statistics is still less than the size of the transition function $P(s' \mid s, a)$.

There is a preliminary extension of this abstract including experimental details and proofs (link).

References

Bram Bakker and Jürgen Schmidhuber. Hierarchical reinforcement learning based on subgoal discovery and subpolicy specialization. In *Proceedings of the 8-th Conference on Intelligent Autonomous Systems, IAS-8*, pages 438–445, 2004. 1

Holger Bast, Stefan Funke, Domagoj Matijevic, Peter Sanders, and Dominik Schultes. In transit to constant time shortest-path queries in road networks. In *Workshop on Algorithm Engineering and Experiments*, pages 46–59, 2007. 1

Martin Stolle and Doina Precup. Learning options in reinforcement learning. In Sven Koenig and RobertC. Holte, editors, *Abstraction, Reformulation, and Approximation*, volume 2371 of *Lecture Notes in Computer Science*, pages 212–223. Springer Berlin Heidelberg, 2002. 1, 2.3