Graph Learning based Generation of Abstractions for Reinforcement Learning

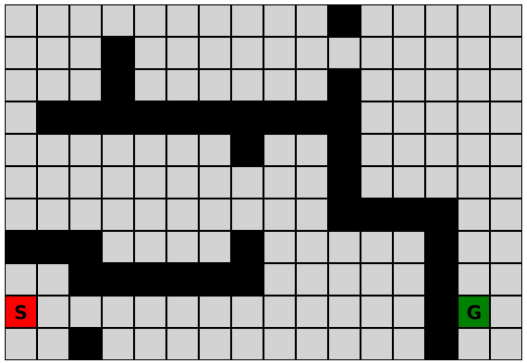
Yuan Xue, Daniel Kudenko, Megha Khosla

L3S Center, Leibniz University Hannover

Motivation

Despite increasing popularity in recent years, reinforcement learning still suffers from high sample complexity and low convergence rate, when facing complex domains with sparse reward.

We give a novel approach for accelerating traditional RL algorithms while requiring little external knowledge.



Potential Based Reward Shaping

Potential Based Reward Shaping can alleviate the problem by giving the agent an extrinsic reward while keeping the optimal policy unchanged.

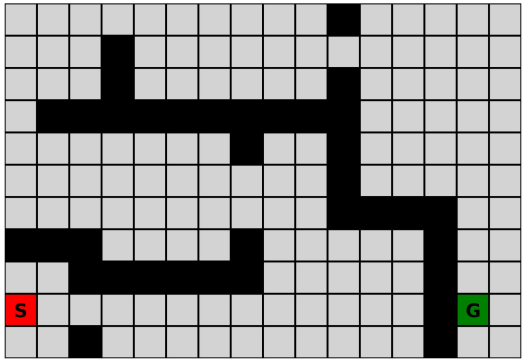
Extrinsic reward is then determined by the difference of potential values between two states.

New goal: to find a proper potential function.

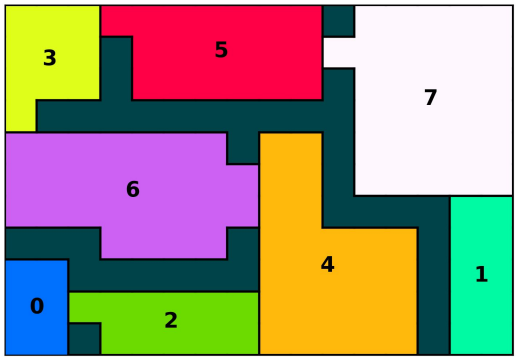
Abstract Markov Decision Process

A potential function can be found by solving an abstract version of task, depicted by AMDP:

$$\mathcal{A} = (S_{\mathcal{A}}, A_{\mathcal{A}}, R_{\mathcal{A}}, P_{\mathcal{A}})$$



Abstraction of states used to be manually labeled by experts, it can be infeasible facing complex problems.



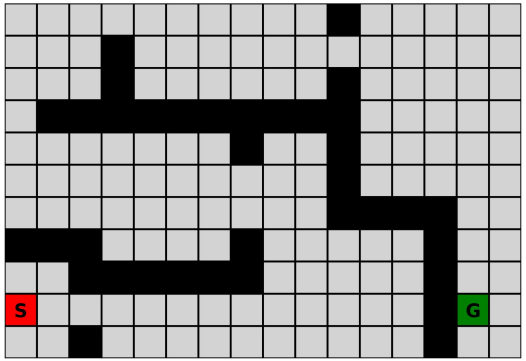
Abstract Markov Decision Process

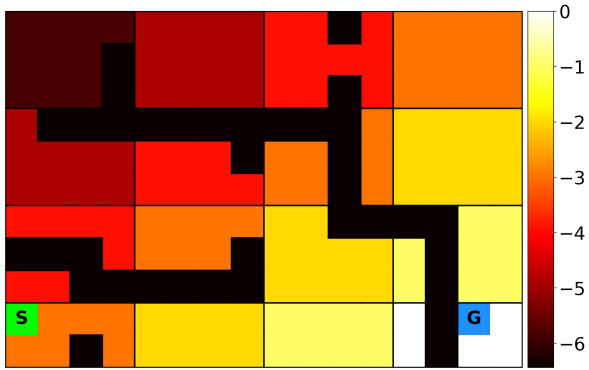
An existing approach is to *uniformly partition* the environment into blocks.

Dynamic programming is used to compute a potential value for each abstract state.

Reward shaping is based on the difference of values between two abstract states

Problem: Low-quality AMDPs cause inaccurate reward shaping, then partially mislead the ground learning process or even destroy it.



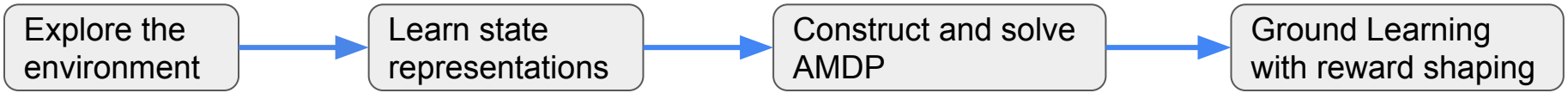


Method

We proposed a novel approach to generate high-quality AMDP automatically, which preserves topological and reward structure of the environment.

Consequently reward shaping derived from AMDP can stably and accurately steer the agent to more rewarding behaviours.

Framework of our approach:



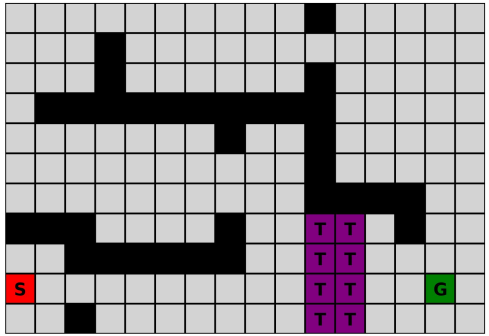
Exploration

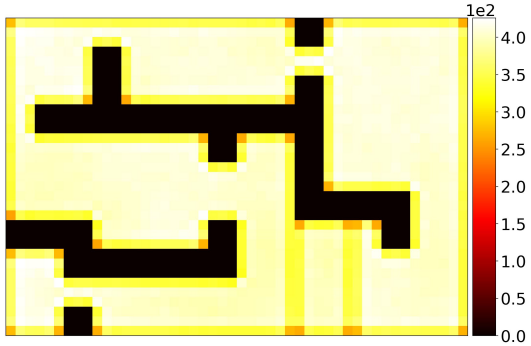
The agent prefers to visit states with lower visit counts.

The agent tries to avoid traps, once it gets into them, it’s hard to come out.

➔ Wide and uniform exploration.
➔ More frequent co-occurrence of states sharing similar topological and reward structure.

The experiences will be stored in form of state sequences.



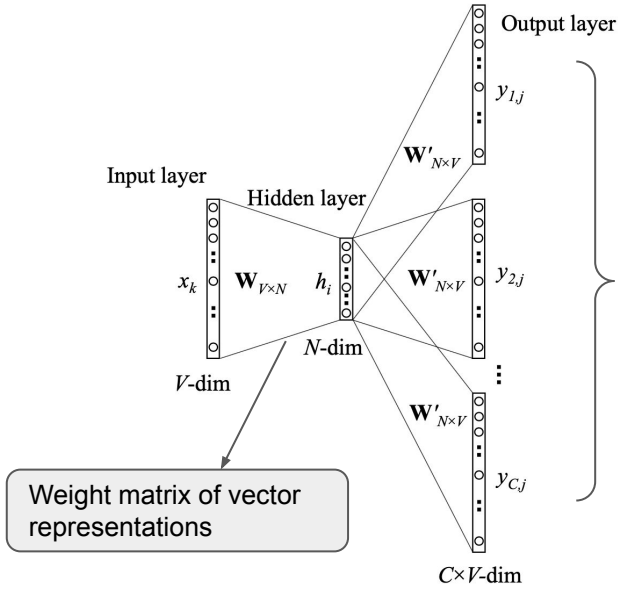


Learn Representations

State transitions in stored experiences induce a graph structure of the ground level MDP.

Inspired by recent graph learning methods, we use a Skip-gram model to generate state representations.

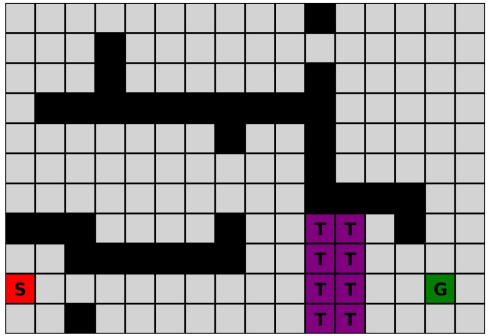
Information of topology and reward structure are encoded into the representations.

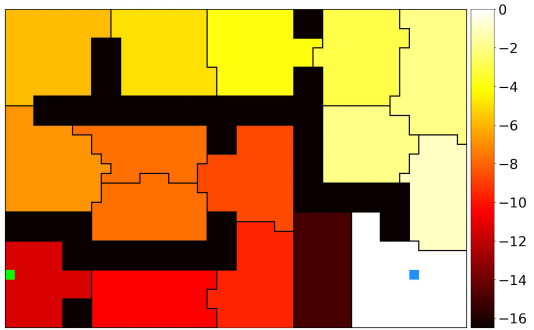


Construct and Solve AMDP

Then representations are clustered into abstract states

The states in each abstract state are topologically close to each other and states of traps are aggregated into a single abstract state.



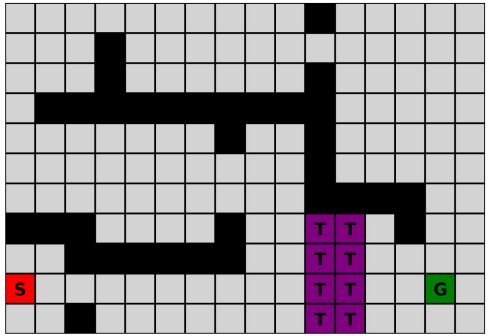


Construct and Solve AMDP

Clustering
+
Stored experiences

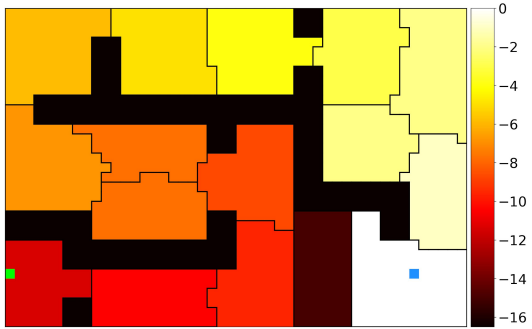
Abstract states
Abstract actions
Abstract rewards
Abstract transitions

AMDP



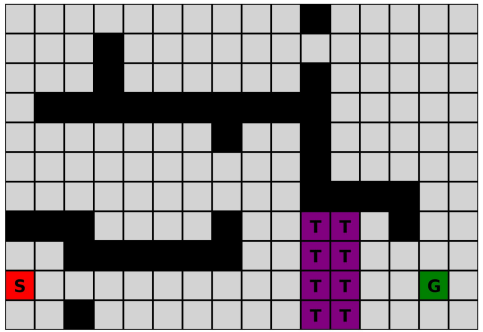
Solve the AMDP by dynamic programming

Abstract states closer to the goal state get higher value except the abstract state of traps which gets much lower value than its neighbours.

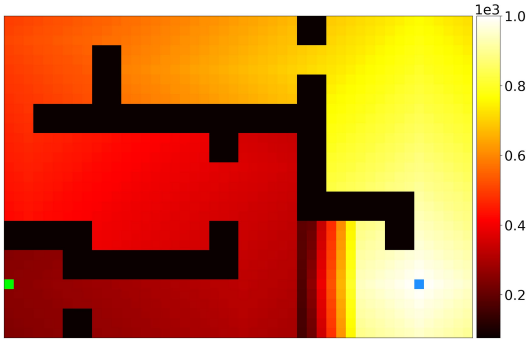


Compare with Uniform Partitioning

Our approach gives the agent correct guidance that agrees with the ground value function.

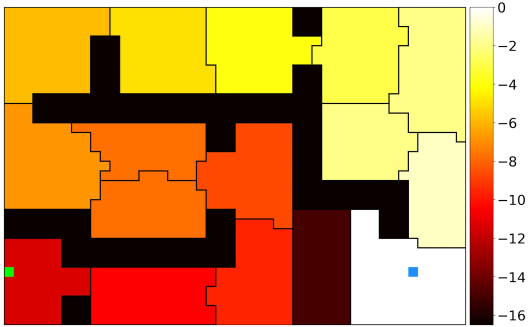


Environment

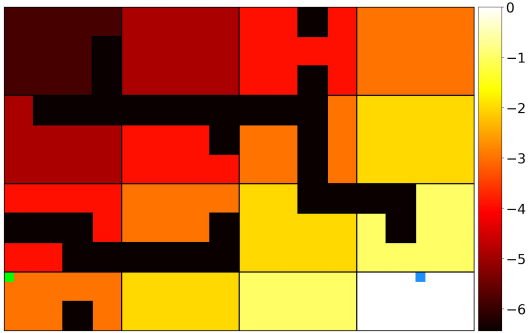


Ground value function

Uniform approach guides the agent straight to the goal, since it ignores topology and reward structure.



Solved AMDP of our approach

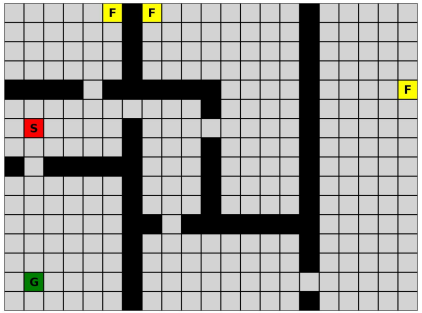


Solved AMDP of Uniform approach

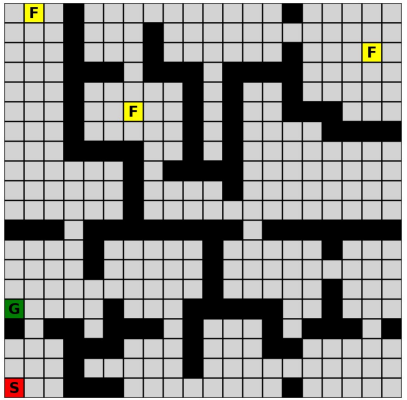
Experiments

Flag Collection Domain:

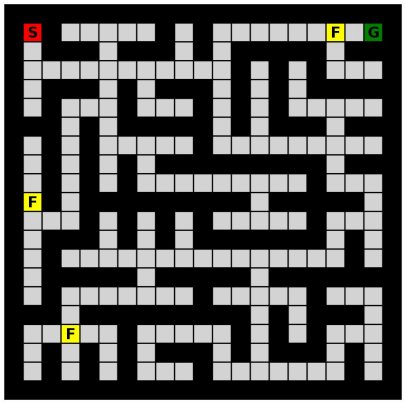
The agent needs to start from the red cell, collect 3 Flags(yellow cells), then reach the green cell.



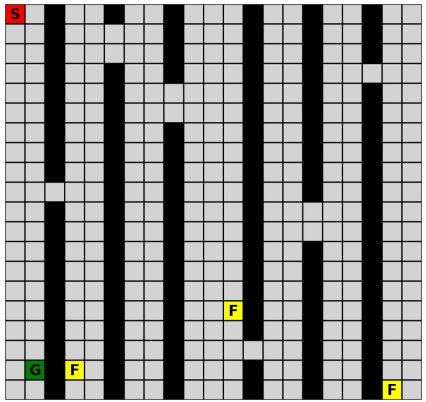
Basic



Low Connectivity



Complex



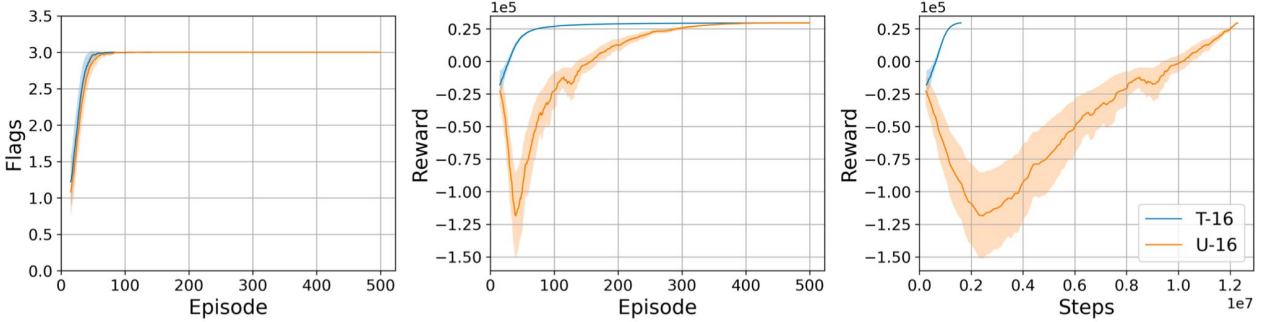
Strips

Results

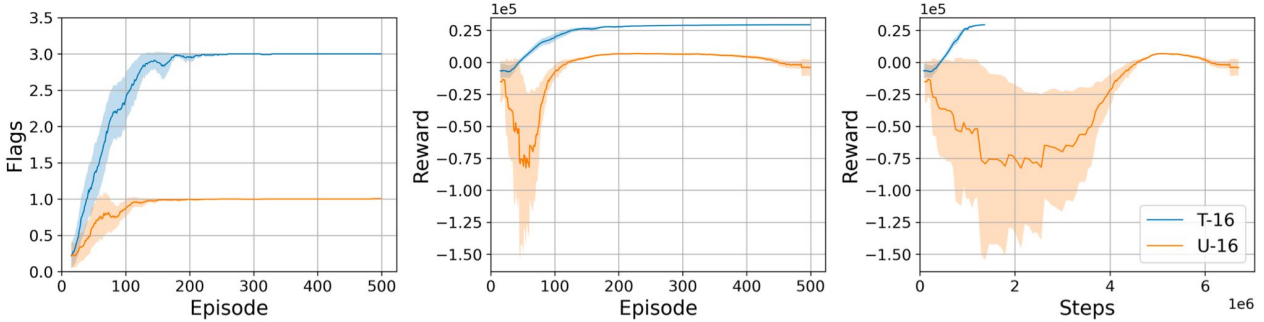
We evaluate both approaches from three angles:

1. Flags against episodes
2. Reward against episodes
3. Reward against total steps

Our approach can stably and successfully converge. It outperforms Uniform approach in terms of **reward performance** and **sample efficiency.**



Basic



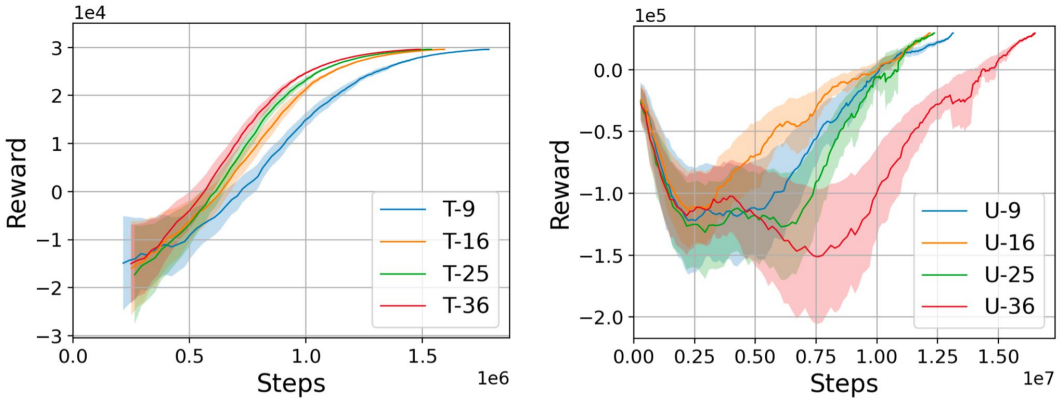
Low Connectivity

Results

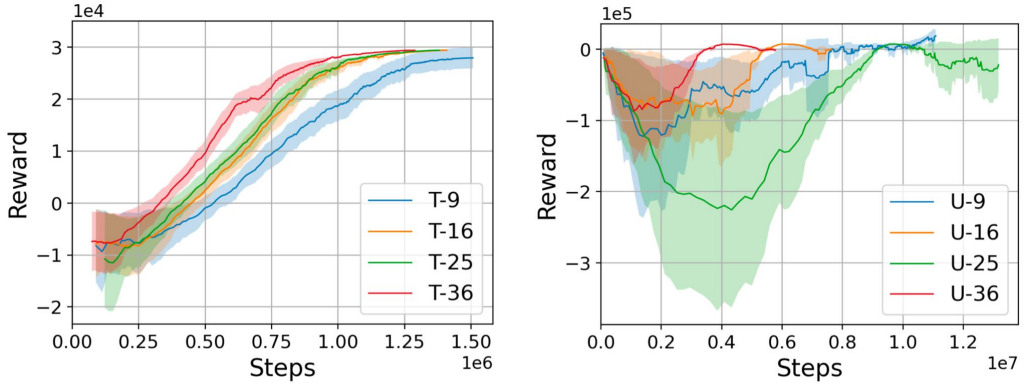
Our approach is less sensitive to different mazes and numbers of abstract states

The performance of fixed number of abstract states is stable among all mazes.

The efficacy of reward shaping from AMDP gets improved as the number of abstract states increases.



Basic

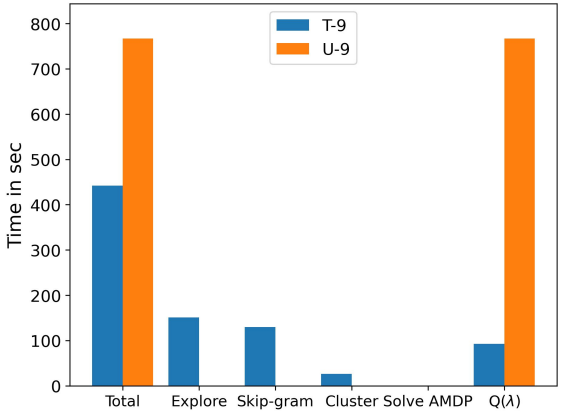


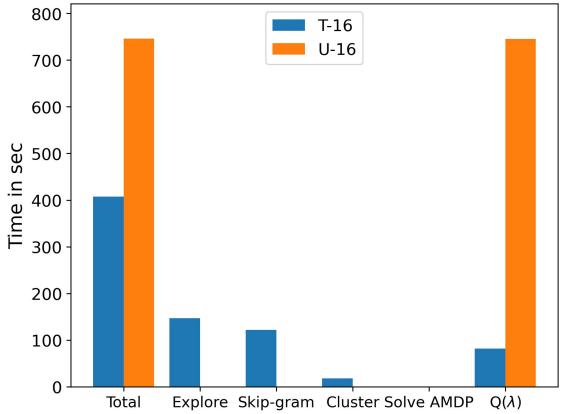
Low Connectivity

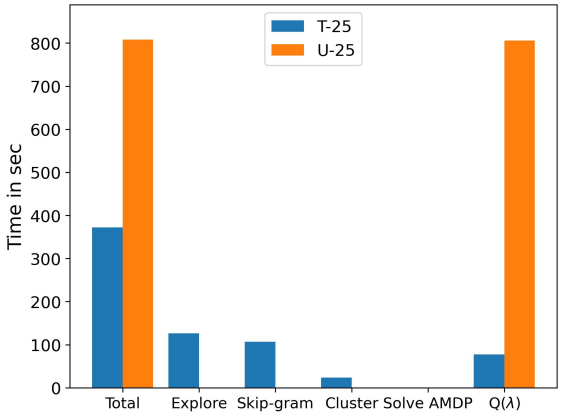
Time Consumption

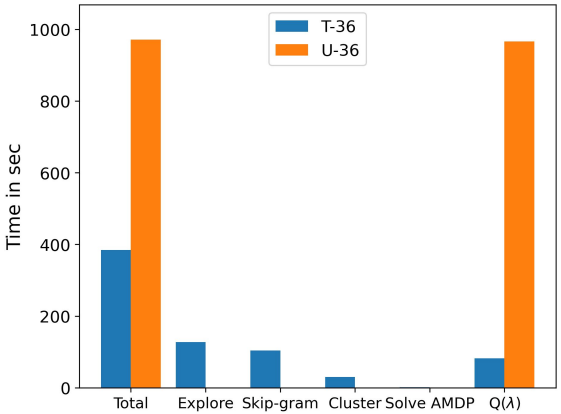
Ground learning time drastically reduced

Total time consumption substantially reduced as well









Conclusion

We proposed a novel approach to generate high-quality AMDP for accelerating RL algorithms.

The generated AMDP preserves topological and reward structure of the environment so that reward shaping can provide accurate guidance to the agent.

Our approach requires little domain knowledge to build AMDP.

We showed impressive performance improvements over the Uniform approach.

Outlook

Non-deterministic abstract transitions

DeepRL models

Thanks!