

1703.08294 Multi-Level Discovery of Deep Options

- **Yunqiu Xu**
 - Other reference:
 - https://github.com/DanielTakeshi/Paper_Notes/blob/master/reinforcement_learning/Multi-Level_Discovery_of_Deep_Options.md
 - <https://danieltakeshi.github.io/2017/11/24/ddo/>
-

1. Introduction

Deep Hierarchical RL
Challenges


Hard to design manually

High-dimensional spaces even harder

Deep RL doesn't scale to deep hierarchy

Need to learn each level separately

Gradient methods for discrete switching?

A hand holding a blue, multi-layered sphere representing a deep hierarchy. The sphere is composed of many small, interconnected nodes, forming a complex, multi-level structure. The hand is positioned at the bottom, supporting the sphere from below. The background is a light gray gradient.

- Challenge:
 - Manually designing options is infeasible in high-dimensional and abstract state spaces
 - Previous work: 2-level, hard to extend to multi-level hierarchies
- Option:
 - Higher level behaviors
 - Smaller fragments
 - Should be executed for several time steps without being hindered by random exploration
- Our work: DDO

- Need a set of demonstration trajectories
- Discover a fixed, parametrized number of options that are most likely to generate these trajectories
- Returning control back to a higher-level meta policy
- Preliminaries: MDP / RL / IL(BC) / Options framework

2. Discovery of Deep Options

- **Assumption: in each visited state, trajectories have preference of actions, these preferences can be represented in a hierarchical structure**

2.1 Imitation Learning for Option Discovery

- Hierarchical BC
 - A generative model to generate trajectory
 - Meta-control signals (form hierarchichy) are unobservable
 - Goal: infer latent variables of the generative model
- Generate trajectory $\xi = (s_0, a_0, \dots, s_T)$:
 - Low level: a set H of options $\langle \pi_h, \psi_h \rangle_{h \in H}$
 - High level:
 - A meta-control policy $\eta(h_t | s_t)$
 - Repeatedly choose an option $h_t \sim \eta(\cdot | s_t)$ given current state, until termination

Initialize $t \leftarrow 0, s_0 \sim p_0, b_0 \leftarrow 1$

for $t \leftarrow 0, \dots, T-1$ **do**

if $b_t = 1$ **then**

 Draw $h_t \sim \eta(\cdot | s_t)$

else // $b_t = 0$

 Set $h_t \leftarrow h_{t-1}$

 Draw $a_t \sim \pi_{h_t}(\cdot | s_t)$

 Draw $s_{t+1} \sim p(\cdot | s_t, a_t)$

 Draw $b_{t+1} \sim \text{Ber}(\psi_{h_t}(s_{t+1}))$

2.2. Expectation-Gradient Algorithm

- Works just like EM is supposed to "work"
- Not only maximize likelihood (which intuitively should lead to matching the expert) but also

sample the latent variables.

2.3 Deeper Hierarchies

We use the following algorithm to iteratively discover a hierarchy of D levels, each level d consisting of k_d options:

```

for  $d = 1, \dots, D - 1$  do
  Initialize a set of options  $\mathcal{H}_d = \{h_{d,1}, \dots, h_{d,k_d}\}$ 
  DDO: train options  $\langle \pi_h, \psi_h \rangle_{h \in \mathcal{H}_d}$  with  $\eta_d$  fixed
  Augment action space  $\mathcal{A} \leftarrow \mathcal{A} \cup \mathcal{H}_d$ 
  Use RL algorithm to train high-level policy

```

3. Experiments

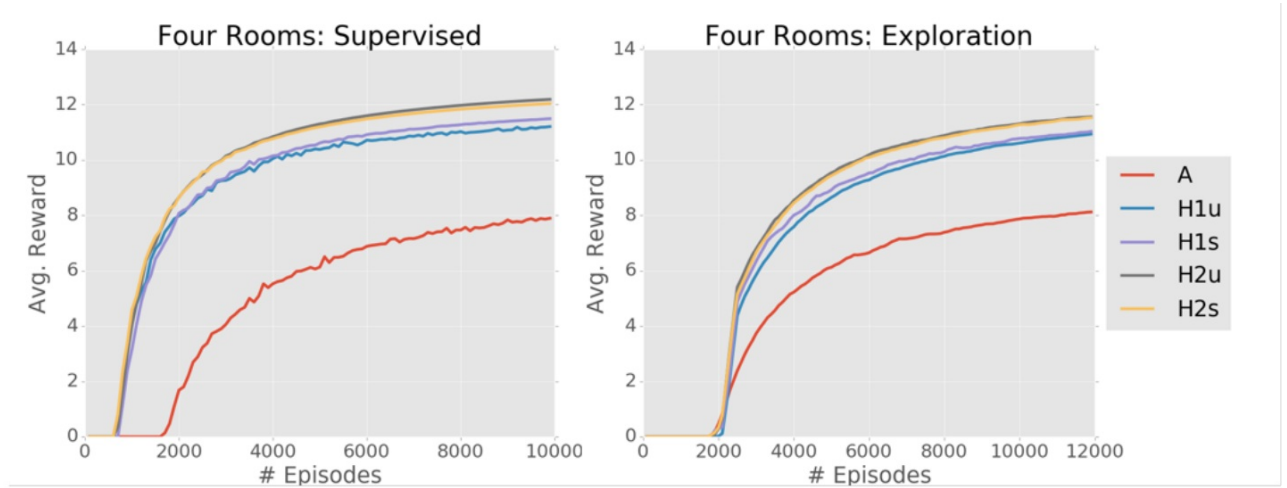


Figure 2: 15-trial mean reward for the Supervised and Exploration problem settings when running DQN with no options (A), low-level options (H1u) and lower- and higher-level options (H2u) augmenting the action space. The options discovered by DDO can accelerate learning, since they benefit from not being interrupted by random exploration.

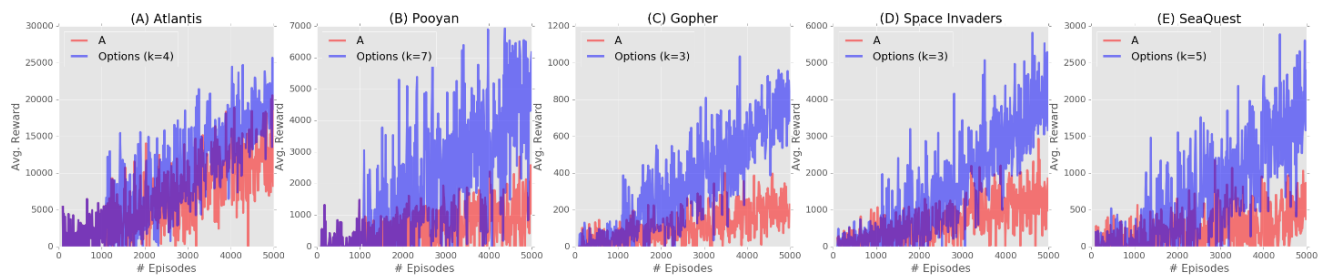


Figure 3: Atari RAM Games: Average reward computed from 50 rollouts when running DQN with atomic actions for 1000 episodes, then generating 100 trajectories from greedy policy, from which DDO discovers options in a 2-level hierarchy. DQN is restarted with action space augmented by these options, which accelerates learning in comparison to running DQN with atomic actions for 5000 episodes. Results suggest significant improvements in 4 out of 5 domains.

4. Conclusion

- DDO can be used recursively to discover multi-level hierarchies
- Limitations:
 - We can replace BC to more robust IL methods
 - Maybe Fig 3 is not convincing
 - Little analysis of how the quality of the supervisor matters
- 原理还需要再了解下