The Option-Critic Architecture

Pierre-Luc Bacon, Jean Harb, Doina Precup

Reasoning and Learning Lab
McGill University, Montreal, Canada

AAAI 2017

Intelligence:

the ability to generalize and adapt efficiently to new and uncertain situations

- Having good representations is key

"[...] solving a problem simply means representing it so as to make the solution transparent." — Simon, 1969

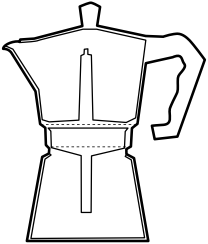
Reinforcement Learning: a general framework for AI

Equipped with a good **state representation**, RL has led to impressive results:

- Tesauro's TD Gammon (1995),
- Watson's Daily-Double Wagering in *Jeopardy!* (2013),
- Human-level video game play in the Atari games (2013),
- AlphaGo (2016)...

The ability to abstract knowledge **temporally over many different time scales** is still missing.

Temporal abstraction



Higher level steps

Choosing the type of coffee maker, type of coffee beans

Medium level steps

Grind the beans, measure the right quantity of water, boil the water

Lower level steps

Wrist and arm movements while adding coffee to the filter, ...

Temporal abstraction in AI

A cornerstone of AI planning since the 1970's:

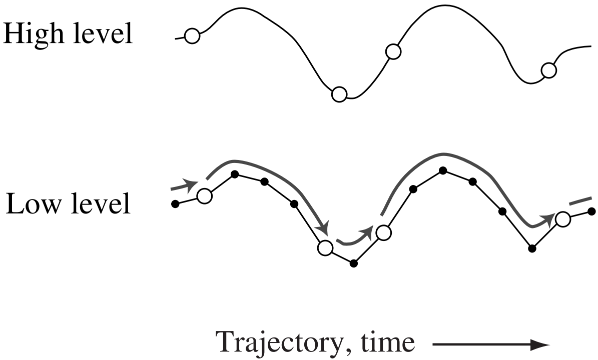
- Fikes et al. (1972), Newell (1972, Kuipers (1979), Korf (1985), Laird (1986), Iba (1989), Drescher (1991) etc.

It has been shown to :

- Generate shorter plans
- Reduce the complexity of choosing actions
- Provide robustness against model misspecification
- Improve exploration by taking shortcuts in the environment

Temporal abstraction in RL

Options (Sutton, Singh, Precup 2000) can represent courses of action at variable time scales:



Options framework

An option ω is a triple:

1. **initiation set**: $\mathcal{I}_\omega$
2. **internal policy**: π_ω
3. **termination condition**: β_ω

Example

Robot navigation: if there is no obstacle in front ($\mathcal{I}_\omega$), go forward (π_ω) until you get too close to another object (β_ω)

We can derive a **policy over options** π_Ω that maximizes the expected discounted sum of rewards:

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \,\middle|\, s_0, \omega_0\right]$$

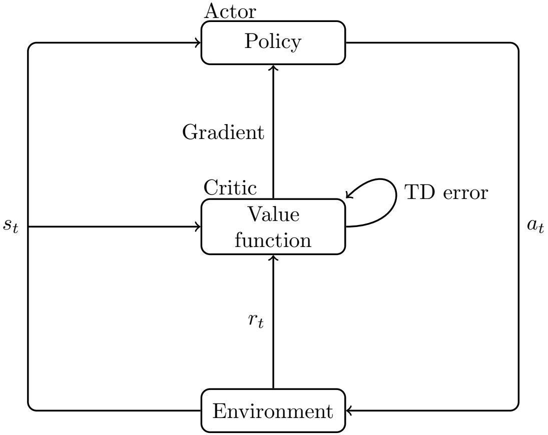
Contribution of this work

The problem of **constructing/discovering** good options has been a challenge for more than 15 years.

Option-critic is a scalable solution to this problem:

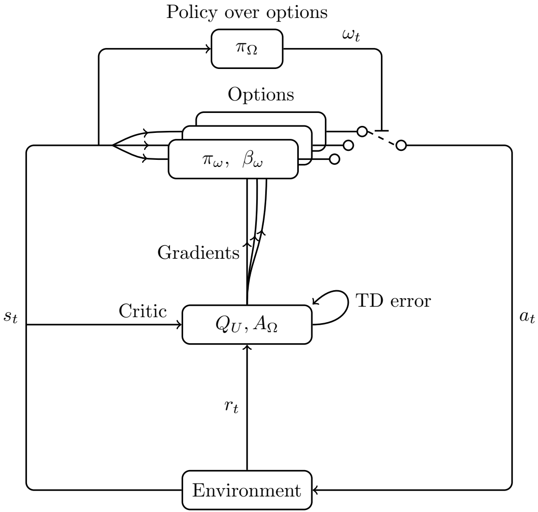
- Online, continual and model-free (but models can be used if desired)
- Requires no a priori domain knowledge, decomposition, or human intervention
- Learns in a single task, at least as fast as other methods which do not use temporal abstraction
- Applies to general continuous state and action spaces

Actor-Critic Architecture (Sutton 1984)



- The policy (actor) is decoupled from its value function.
- The critic provides feedback to improve the *actor*
- Learning is fully online

Option-Critic Architecture



- Parameterize internal policies and termination conditions
- Policy over options is computed by a separate process

Main result: Gradient updates

- The **gradient wrt. the internal policy parameters** θ is given by:

$$\mathbb{E}\left[\frac{\partial \log \pi_{\omega,\theta}(a|s)}{\partial \theta} Q_U(s, \omega, a)\right]$$

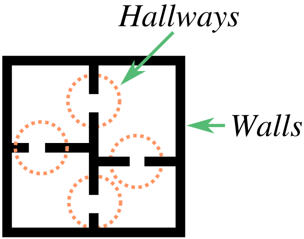
  This has the usual interpretation: **take better primitives more often** inside the option

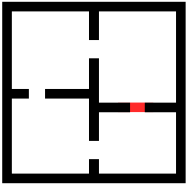
- The **gradient wrt. the termination parameters** ν is given by:

$$\mathbb{E}\left[-\frac{\partial \beta_{\omega,\nu}(s')}{\partial \nu} A_{\pi_\Omega}(s', \omega)\right]$$

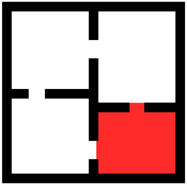
  where $A_{\pi_\Omega} = Q_{\pi_\Omega} - V_{\pi_\Omega}$ is the advantage function This means that we want to **lengthen options that have a large advantage**

Results: Options transfer



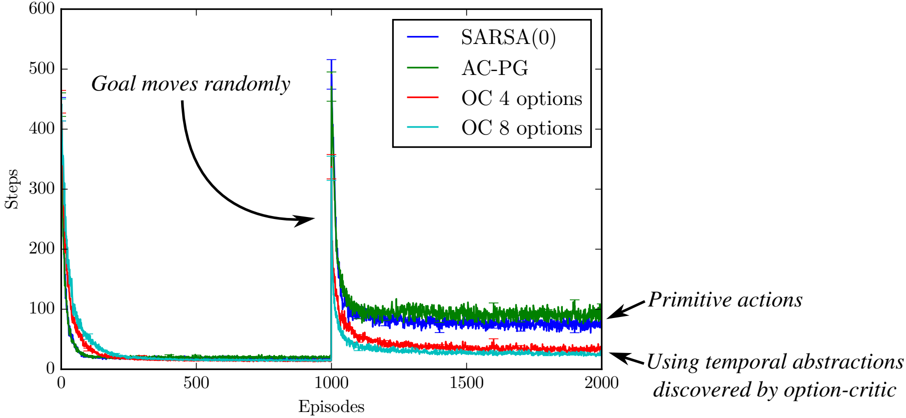


Initial goal



Random goal after 1000 episodes

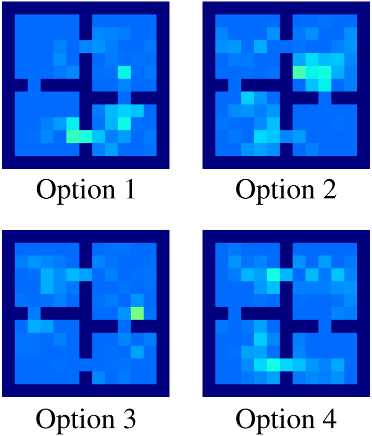
Results: Options transfer



- Learning in the first task no slower than using primitives
- Learning once the goal is moved faster with the options

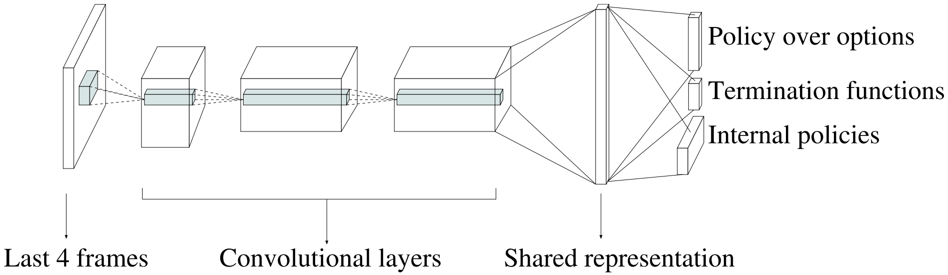
Results: Learned options are intuitive

Probability of terminating in a particular state, for each option:



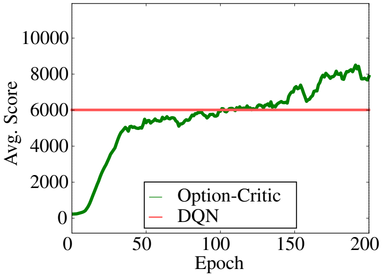
- Terminations are more likely near hallways (although there are no pseudo-rewards provided)

Results: Nonlinear function approximation

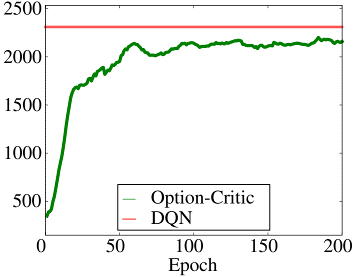


Same architecture as DQN (Mnih & al., 2013) for the 4 first layers but hybridized with options and the policy over them.

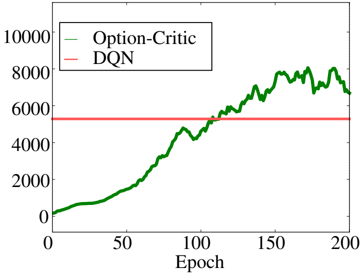
Performance matching or better than DQN



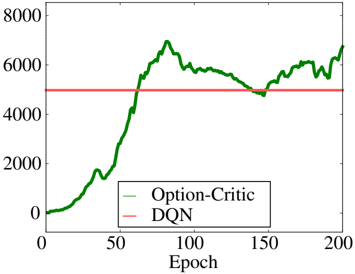
(a) Asterix



(b) Ms. Pacman

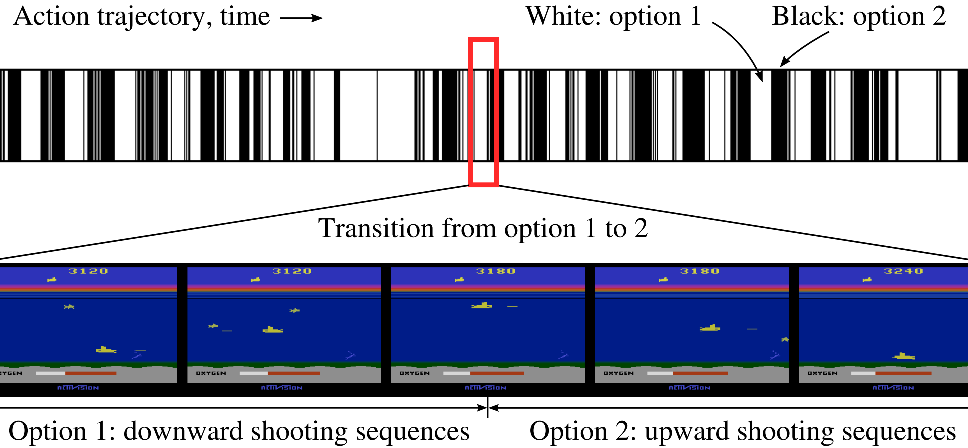


(c) Seaquest



(d) Zaxxon

Interpretable and specialized options in Seaquest



Conclusion

Our results seem to be the first to be:

- fully end-to-end
- within a single task
- at speed comparable or better than using just primitive methods

Using ideas from policy gradient methods, option-critic

- provides continual option construction
- can be used with nonlinear function approximators
- can incorporate regularizers or pseudo-rewards easily

Future work

- Learn initiation sets:
  - Would require a new notion of *stochastic initiation functions*
- More empirical results !

Try our code :
`https://github.com/jeanharb/option_critic`