TempoRL: Learning When to Act

André Biedenkapp¹, Raghu Rajan¹, Frank Hutter¹² and Marius Lindauer³
¹University of Freiburg | ²Bosch Center for Artificial Intelligence | ³Leibniz University Hannover

In a Nutshell

- We propose a proactive way of doing RL
- We introduce skip-connections into MDPs
  - use of action repetition
  - faster propagation of rewards
- We propose a novel algorithm using skip-connections
  - learn what action to take & when to make a decision
  - condition the when on the what
- We evaluate our approach with in a variety of settings
  - tabular Q-learning on Gridworlds
  - DQN on featurized environments
  - DDPG on featurized environments
  - DQN with image observations on Atari environments

When Do We Need to Act?

An agent has to reach the goal from the start state without falling down the cliff (black squares)

- Classic RL reacts to each observation
- Has to make a decision in each state
- TempoRL anticipates when to make a new decision
- Skips over states that require the same action

Learning when to act reduces the number of required decisions by 80%

Such simplified policies are easier to learn

Experimental Evaluation: Tabular Q-Learning

- Example for tabular q-learning on the gridworld above
- TempoRL
  - learns much faster,
  - leads to better exploration of the space,
  - requires far fewer training steps than vanilla Q-Learning

Deep Q-Learning on Atari

- In the beginning vanilla DDQN outperforms TempoRL
- After learning to make use of action repetition, TempoRL starts to strongly outperform vanilla DDQN
- A prior method using action repetition (DAR) fails to learn proper action repetition in the same timeframe
- More experiments in the paper, incl. TempoRL DDPG, more environments and influence of architectures

TempoRL Allows for

- better exploration
  - Exploring along a longer horizon
- faster learning
  - Learning when to act reduces the policy complexity
  - Policies needing fewer decisions are easier to learn
- better explainability
  - Agent can indicate when new decisions need to be made

Learning to Skip

1. Use standard agent to learn behaviour at a given state $s$
   $Q^a(s_t, a_t) = a_t$
2. Condition skip $j$ on the chosen action $a$
   $Q^{j, i}(s_t, j|a) = j$
3. Repeat action $a$ for $j$ steps
   - Behaviour policy can be learned with vanilla agents
   - The skip Q-function can be learned using n-step updates

Skip MDPs

- Action repetition induces skips
- Information can be propagated faster along skips
- With large skips, multiple smaller skips can be observed and learned from

Code

@AutoML_org