CS7011 : Topics in Reinforcement Learning
Summary of Hindsight Experience Replay : Andrychowicz, Wolski, Alex Ray

Sahana Ramnath : EE15B109

Abstract

This paper (Andrychowicz et al., 2017) proposes a novel method called **Hindsight Experience Replay***(HER)* which allows sample-efficient learning from sparse and binary rewards. This technique can be combined with off-policy learning and can be seen as a form of implicit curriculum learning. Usage of HER eliminates the need for complicated reward engineering such as shaping. This can therefore be applied to problems without prior knowledge of the domain. This is especially of great use in domains like robotics where the reward is just a binary signal denoting completion of the task. Experiments have been conducted on the task of manipulating objects with a robotic arm : in particular, pushing,sliding and pick-and-place, each task only giving a binary reward based on the task completion. The agent trained on a simulation was deployed on an actual robot where it successfully completed the task.

1. Idea behind HER

Humans are capable of learning *something* from any task they perform, regardless of whether they achieved the desired goal or not. *HER* facilitates the agent to perform this kind of reasoning. The main idea behind *HER* is to replay each episode with a different goal than the task's goal and learn something from each episode, regardless of whether the task's goal was achieved or not.

To be more explicit, consider an episode with a state sequence $s_1, s_2, .., s_T$ and a task goal $G \neq s_1, ..., s_T$. While this trajectory doesn't tell how to reach G, it tells how to reach s_T. Hence the transitions stored from this episode can be used to learn how to reach G as well as s_T.

2. Background : DDPG and UVFA

Deep Deterministic Policy Gradients(**DDPG**) is a model-free RL algorithm for continuous action spaces. It maintains a neural network to model the policy π(the actor) and the value function Q^π(the critic). Episodes are generated using a behavioural policy which is a noisy version of the actual policy $\pi_b = \pi + \mathcal{N}(0, 1)$.

Universal Function Value Approximators(**UVFA**) is an extension of DQN to the case where there are multiple goals. Every possible goal $g \in \mathcal{G}$ has a corresponding reward function $r_g : SXA \rightarrow$ R. At every step, the agent's input is the current state s_t, *as well as* the current goal g and gets the reward $r_g(s_t, a_t)$. The value function now depends on the goal as well : $Q^\pi(s_t, a_t, g)$.

DDPG is extended with UVFA for Hindsight Experience Replay.

3. The algorithm

Each transition is now stored in the replay buffer along with the goal to be achieved(G or s_T).

Every goal g corresponds to a mapping function $f_g : \mathcal{S} \rightarrow \{0, 1\}$ and the agent's goal is to achieve a state s such that $f_g(s) = 1$. Either a desired state of the system is specified with $\mathcal{S} = \mathcal{G}$ and $f_g(s) = [s = g]$, or only properties of the goal is specified such as reaching a given x-coordinate in the 2D space : $\mathcal{S} = \mathcal{R}^2, \mathcal{G} = \mathcal{R}$ and $f_g((x, y)) = [x = g]$.

A universal policy can be trained by sampling goals and initial states from some distribution, running the agent for some timesteps and giving a reward $r_g(s, a) = -[f_g(s) = 0]$ when the goal is not achieved.

In the simplest algorithm, each trajectory is replayed with the goal s_T of that episode and with the task's final goal G. The goals used for replay gradually shift from those easy to achieve even by random agents to more difficult ones.

Experiments were conducted with *DDPG* and *DDPG+HER* to compare and demonstrate the advantage of *HER*.

4. Comparison to reward shaping

Both the above experiments were conducted with and without additional reward shaping. The experiment with reward shaping yielded poor results mainly because :

- Shaping considers getting *near* the goal itself to be a success, whereas for these tasks, *only* reaching the goal is to be considered a success.
- Shaping penalizes moving in the wrong direction, which may hinder exploration.

5. Choosing goals

One way to choose the additional goals is to choose the final goal of the corresponding episode(strategy *final*). Other ways are strategies *future* : k goals from the same episode and after this transition occurred, *episode* : k goals from the same episode this transition occurred and *random* : k random states observed so far in training.

References

Andrychowicz, Marcin, Wolski, Filip, Ray, Alex, Schneider, Jonas, Fong, Rachel, Welinder, Peter, McGrew, Bob, Tobin, Josh, Abbeel, OpenAI Pieter, and Zaremba, Wojciech. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, pp. 5048–5058, 2017.