Hindsight Goal Prioritization for Sparse Reward Environments

Final Project

Reinforcement Learning

Flavio Maiorana (2051396)

08/09/2023

SAPIENZA
UNIVERSITÀ DI ROMA

Table of Contents

Problem statement

Robotics environments

- Complex and different goals
- Sparse rewards
- Continuous action space

$\Downarrow$

Problems with exploration and reward shaping

- Goal may be too complex and observation space is big: we may never get reward 1
- Classical off-policy algorithms don't valorize much the failed episodes

$\Downarrow$

Solution: Enhancing the Replay Buffer

Fetch

1 Introduction

- Based on the 7-DoF Fetch Manipulator arm, with a two-fingered parallel gripper
- Tasks: Reach, Push, Slide and Pick-and-Place
- Action: Box(-1.0, 1.0, (4,), float32) $\rightsquigarrow$ Displacement in meters of the EE
- Observation: dictionary with info about the robot's end effector state and goal
  - Observation: ndarray of shape (25,) $\rightsquigarrow$ kinematic info of the block object and EE
  - Desired goal: ndarray of shape (3,) $\rightsquigarrow$ desired position of the EE or the block
  - Achieved goal: ndarray of shape (3,) $\rightsquigarrow$ current position of the EE or the block
- Reward: if we use sparse rewards -1 for every timestep and 0 for reaching the goal
- Termination: episodes have no termination since they have infinite horizon. Thus, they are truncated after T steps (by default 50)

Table of Contents

Architecture

2 DDPG

- Two neural networks in performing actor-critic policy gradient
  - Actor inference: observed state $\rightarrow$ action maximising the action-value function
- Critic inference: state and action $\rightarrow$ $valueoftheaction - valuefunction$

Implementation

2 DDPG

Drawbacks in the Fetch Environment

-

Table of Contents

3 Replay buffer

HER

Intuition

- The intention is to valorize also failed episodes (majority in robotics environments)
- Done by storing episodes multiple times, substituting the desired goal with another from the same episode, treating the episode as if it was successful
- Formally speaking, for each episode ($t = 0...50$) we do the following steps
  - store $(s_t||g, a_t, r_t, s_{t+1}||g)$
  - sample a set of additional achieved goals G from the current episode
  - store $(s_t||g', a_t, r_t, s_{t+1}||g')$ for every $g' \in G$
- Different strategies can be adopted to sample goals
  - Final
  - Future
  - Random

HER

Implementation

Intuition

```
\documentclass{beamer}
```

Enhancements over Vanilla HER

```
\documentclass{beamer}
```

Table of Contents

4 Results

Hindsight Goal Prioritization for Sparse Reward Environments

Thank you for listening!

Any questions?