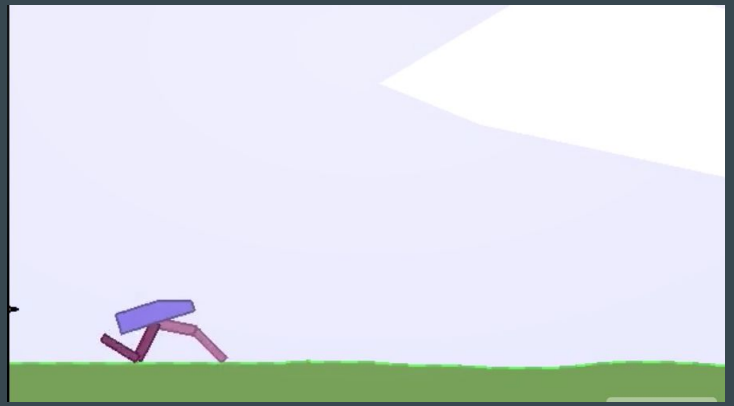
Reinforcement Learning

Project Proposal

Abhishek Gupta 2016004
Yajur Ahuja 2016121

Problem and Motivation

Walking for a robot/agent is a beneficial locomotion as it allows it to commute in an unstructured terrain. We want to use to Reinforcement Learning techniques to teach an agent to learn how to walk in a computer simulated environment. We plan to use the BipedalWalker environment provided by openAI. BipedalWalker is a forward moving agent with 2 legs with 2 joints in each leg. Our task is to teach the agent to learn how to walk by applying appropriate amount of torque on agent's joints so the agent can use the legs to walk through the end of the rough terrain.



BipedalWalker Environment: https://gym.openai.com/envs/BipedalWalker-v2/

Problem Details

Problem Objective:

- For able to get the agent to able to walk through the terrain and reach the far end without falling down on the way.
- Once the model is trained, it should be able to give an average reward per episode very close to +300.

States

The environment has 24 parameters for every state and each of the parameters has a continuous space.

Details: https://github.com/openai/gym/wiki/BipedalWalker-v2

Actions

There are 4 actions in the action space and each of the actions has a continuous space. The actions are torque to either of the knees or either of the hips.

Rewards

High positive reward (+ 300) for reaching the end and a high negative reward(-100) when the agent falls down for each episode.Small negative rewards/costs for applying torques.

RL Algorithms

Deep Deterministic Policy Gradient

(Baseline)

The BipedalWalker environment has a continuous state and action space so DDPG should be a suitable baseline algorithm for our environment.

Source: “Continuous control with deep reinforcement learning”

https://arxiv.org/abs/1509.02971

Twin Delayed DDPG

We propose to implement the Twin Delayed DDPG algorithm and compare the results to that of the baseline model for the Bipedal Environment.

Source: “Error in Actor-Critic Methods”

https://arxiv.org/abs/1802.09477

Existing Work

Other algorithms for problems of this kind

- Evolutionary Strategies: https://openai.com/blog/evolution-strategies/
- Proximal Policy Optimization (PPO): https://arxiv.org/abs/1707.06347
- SAC Algorithm: https://arxiv.org/abs/1801.01290

Proposed deliverables

Deliverable 1	• Implement the baseline model - DDPG
Deliverable 2	• Implement TD3 (2018)
Deliverable 3 (If Time Permits)	• Apply Transfer learning to another Open AI environment. (BipedalWalker Hardcore or Cheetah)