Deep RL with MuJoCo

Adrià de Angulo
Daniel Matas
Hariss Mohammad Jabeen
Miquel Quesada
Advisor: JuanJo Nieto

https://github.com/danimatasd/MUJOCO-AIDL
Motivation of the project

- Appliance of RL to solve dynamic/real world problems (robotics, autonomous driving, healthcare...)
- Trial and error approach of the RL process
- Deployment of RL algorithms in dynamic environments generated by a virtual engine (MuJoCo)
Goals

Learn about the basics of RL (states, actions, policies...) and use a physics engine to produce an accurate simulation

Make the robot be able to walk in one direction

Make the robot be able to walk over a small step without falling over

Apply the concepts of MLOps to deploy a replicable repository that can be run by any user
Our proposal: Data
Our proposal: Environment

Half Cheetah

Anymal C

Gym

Custom with Mujoco Engine
Our proposal: Computational Resources

Algorithm 1 PPO, Actor-Critic Style

for iteration=1,2,\ldots \ do
  for actor=1,2,\ldots,N \ do
    Run policy $\pi_{\theta_{\text{old}}}$ in environment for $T$ timesteps
    Compute advantage estimates $\hat{A}_1,\ldots,\hat{A}_T$
  end for
  Optimize surrogate $L$ wrt $\theta$, with $K$ epochs and minibatch size $M \leq NT$
  $\theta_{\text{old}} \leftarrow \theta$
end for
Our proposal: Half Cheetah NN

17 Half Cheetah Environment Inputs

64  128

6 Torque Outputs

1 Expected Return
Our proposal: Anymal C NN

37 Anymal C Environment Inputs

128 128 128

12 Position Outputs

1 Expected Return
Milestones

1) Learn the theoretical base of RL ✓
2) Establish a teamwork setup ✓
3) Get familiar with the engine ✓
4) Generate a “base” code to train a model ✓
5) + Apply code to Half Cheetah
6) + Create a Mujoco Environment function similar to Gym
7) + Apply code to ANYmal C
8) + Hyperparameter Sweep for each code
9) + Final run for each code
10) + Get videos for each run
## Project Plan

### Gantt Diagram

<table>
<thead>
<tr>
<th>Number</th>
<th>Task Description</th>
<th>Responsible</th>
<th>Start Date</th>
<th>Finish Date</th>
<th>Duration</th>
<th>Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Project Kickoff</td>
<td>Mujoco Team</td>
<td>21/11/22</td>
<td>21/11/22</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>1.2</td>
<td>Brainstorming, Project</td>
<td>Mujoco Team</td>
<td>23/11/22</td>
<td>23/11/22</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>1.3</td>
<td>Studying Reinforcement</td>
<td>Mujoco Team</td>
<td>22/12/22</td>
<td>25/01/23</td>
<td>33</td>
<td>100%</td>
</tr>
<tr>
<td>1.4</td>
<td>Studying Mujoco Literature and examples</td>
<td>Mujoco Team</td>
<td>22/12/22</td>
<td>25/01/23</td>
<td>33</td>
<td>100%</td>
</tr>
</tbody>
</table>

### Mockups and Initial Testing

<table>
<thead>
<tr>
<th>Week 1</th>
<th>Week 2</th>
<th>Week 3</th>
<th>Week 4</th>
<th>Week 5</th>
<th>Week 6</th>
<th>Week 7</th>
<th>Week 8</th>
<th>Week 9</th>
<th>Week 10</th>
<th>Week 11</th>
<th>Week 12</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>M</td>
<td>X</td>
<td>L</td>
<td>X</td>
<td>X</td>
<td>V</td>
<td>L</td>
<td>M</td>
<td>X</td>
<td>L</td>
<td>M</td>
</tr>
</tbody>
</table>

### 2. mujoco + RL

<table>
<thead>
<tr>
<th>Task Description</th>
<th>Responsible</th>
<th>Start Date</th>
<th>Finish Date</th>
<th>Duration</th>
<th>Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1. Environment/state</td>
<td>Mujoco Team</td>
<td>23/01/23</td>
<td>01/02/23</td>
<td>6</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.1 Half Cheetah</td>
<td>Mujoco Team</td>
<td>01/02/23</td>
<td>10/03/23</td>
<td>9</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.1.1 Half Cheetah walking</td>
<td>Mujoco Team</td>
<td>01/02/23</td>
<td>10/03/23</td>
<td>9</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.2 ANimal Robot</td>
<td>Mujoco Team</td>
<td>13/02/23</td>
<td>23/03/23</td>
<td>10</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.2.1 ANimal Robot walking</td>
<td>Mujoco Team</td>
<td>13/02/23</td>
<td>15/03/23</td>
<td>2</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.2.2 ANimal Robot get over a platform</td>
<td>Mujoco Team</td>
<td>24/02/23</td>
<td>21/03/23</td>
<td>7</td>
<td>100%</td>
</tr>
<tr>
<td>2.1.3 Documentation</td>
<td>Mujoco Team</td>
<td>10/03/23</td>
<td>22/03/23</td>
<td>12</td>
<td>100%</td>
</tr>
</tbody>
</table>
Results Half-Cheetah
Results Half-Cheetah
Results Half-Cheetah

-304

1000

2000

3000

3908

5006

5734

-506
Results Anybotics Anymal C
Results Anybotics Anymal C
Results Anybotics Anymal C

1300  2222  3347  4286

5213  6134  7647  8325
Results Anybotics Anymal C with a step
Conclusions

- PPO is a powerful algorithm that proved that with a small NN is capable of learning quite fast, solving these particular experiments in less than 24 hours training.

- Hyperparameter tuning is essential to converge on a solution but can take a lot of time.

- Transfer learning is possible when the agent and the environment are the same even when the NN is overfitted for a concrete task.
Next steps to improve results:

Keep on working with hyper parameter tuning.

Training Multiple Actors and parallelizing with GPU.

Add more data to the state: last actions taken, collisions, terrain, etc.

Reward tuning, for example: penalizing energy consumption to optimize movements and make them smoother.

Change the entropy non-linearly or take the value for the covariance matrix from a NN.

Test with a bigger neural network.
Thanks for your attention