RL Project Template

Andrej Lukic
Department of Computer Science
University of Bath
Bath, BA2 7AY
al2274@bath.ac.uk

1 Problem Definition

The problem chosen for this assignment is the Lunar Lander problem because landing a small rocket on the moon is a bit correlated with my interest in flight controllers used in custom made drones. The environment for testing the algorithm is freely available on the Gymnasium web site (it's an actively maintained fork of the original OpenAI Gym developed by Oleg Klimov.

The Lunar Lander is a classic rocket trajectory optimisation problem (comprehensive environment description is found on the gymnasium web site). In the simulation, the spacecraft has a main engine and two lateral boosters that can be used to control its descent and the orientation of the spacecraft. The spacecraft is subject to the moon's gravitational pull, and the engines have an unlimited amount of fuel. The spacecraft must navigate to the landing spot between two flags at coordinates (0,0) without crashing. Landing outside of the landing pad is possible. The lander starts at the top center of the viewport with a random initial force applied to its center of mass. The environment has 4 discrete actions:

- 0: do nothing
- 1: fire left orientation engine
- 2: fire main engine
- 3: fire right orientation engine

The state is an 8-dimensional vector: the coordinates of the lander in x & y, its linear velocities in x & y, its angle, its angular velocity, and two booleans that represent whether each leg is in contact with the ground or not.

After every step a reward is granted. The total reward of an episode is the sum of the rewards for all the steps within that episode.

For each step, the reward:

- is increased/decreased the closer/further the lander is to the landing pad.
- is increased/decreased the slower/faster the lander is moving.
- is decreased the more the lander is tilted (angle not horizontal).
- is increased by 10 points for each leg that is in contact with the ground.
- is decreased by 0.03 points each frame a side engine is firing.
- is decreased by 0.3 points each frame the main engine is firing.

The episode receive an additional reward of -100 or +100 points for crashing or landing safely respectively. An episode is considered a solution if it scores at least 200 points.

The episode finishes if:

- the lander crashes (the lander body gets in contact with the moon);
- the lander gets outside of the viewport (x coordinate is greater than 1);
- the lander is not awake. From the Box2D docs, a body which is not awake is a body which doesn't move and doesn't collide with any other body:

2 Background

3 Method

4 Results

5 Discussion

6 Future Work

7 Personal Experience

References

Appendices

If you have additional content that you would like to include in the appendices, please do so here. There is no limit to the length of your appendices, but we are not obliged to read them in their entirety while marking. The main body of your report should contain all essential information, and content in the appendices should be clearly referenced where it's needed elsewhere.

Appendix A: Example Appendix 1

Appendix B: Example Appendix 2