Reinforcement Learning Assignment4 Report

Ananya Hazarika, Kuhu Gupta, Nandika Soni, Meheeka Pandey- Team 1

Answers for section 4–STEP1

1. How many actions are there in the action space?

- **4** actions: do nothing, fire left orientation engine, fire main engine, fire right orientation engine.

2. Is it possible to fire both the main engine and left/right orientation engine at the same time? Justify your answer.

- **Yes**, it is possible to fire both the main engine and the orientation engines at the same time, as the action space is discrete with separate actions for each engine. This allows you to combine actions such as firing the main engine and the left or right orientation engines simultaneously.

3. How many observations are there?

- There are **8** observations in the state vector: the lander's position in x and y, linear velocities in x and y, angle, angular velocity, and two booleans indicating whether each leg is in contact with the ground.

4. How many observations are there in the observation space?

- **8** observations (coordinates, velocities, angle, angular velocity, and leg contact with the ground).

5. What are the sources of randomness for this environment?

- Randomness comes from the initial position and velocity of the lander, as well as the random forces applied to its center of mass at the start of each episode. Additionally, wind effects can be applied when `enable_wind=True`, which introduces further randomness in the lander's movement.

6. Convergence Criteria for DQN Training:

- The convergence criterion is when the agent reaches a **reward of 200 points** or higher, indicating the agent has successfully learned to land and rest on the pad.

Answers for section 4–STEP2

Key Associations

The keys are associated with the respective actions as follows:

- Key `a`: Fire left orientation engine (action 1).
- Key `s`: Fire main engine (action 2).
- Key `d`: Fire right orientation engine (action 3).

Function of `noop`

In line 27, the function of `noop` is to define the default action taken when no key is pressed. Here, it is set to 0, which corresponds to the action "Do nothing."

Function of `callback=total_reward_func`

The `callback` argument is used to process events during gameplay. The `total_reward_func` function is invoked after every action step to calculate and display the total reward accumulated during an episode when the terminated flag is True.

Adjusting Game Speed

To slow down the game, reduce the `fps` (frames per second) value in line 27. For example, change `fps=10` to `fps=5` to make the gameplay slower.

Plots in next page

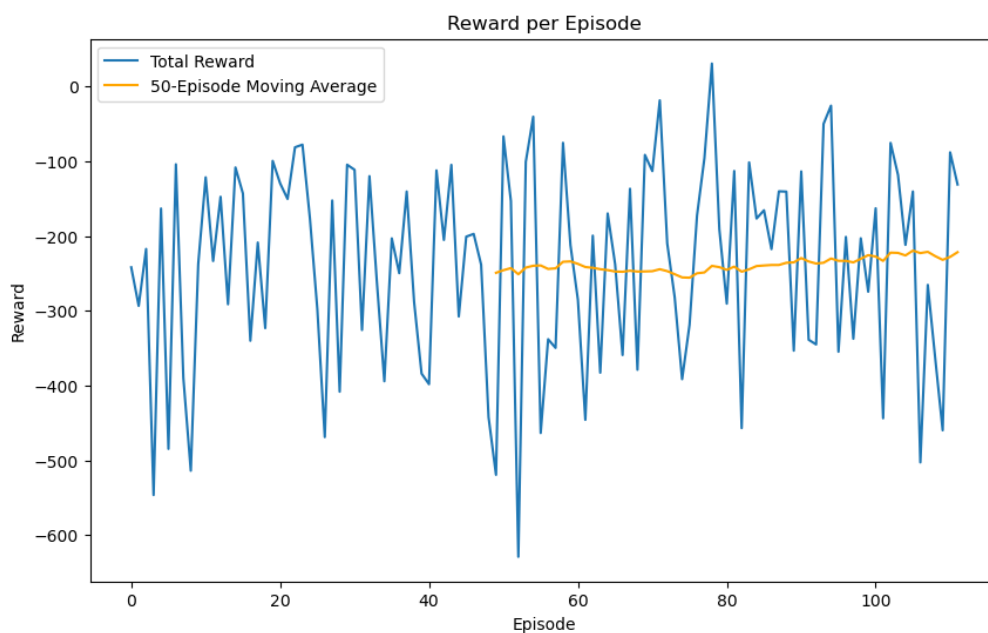# 1 Plot for Offline data use as FALSE, model= lunar_lander_model.keras

Figure 1
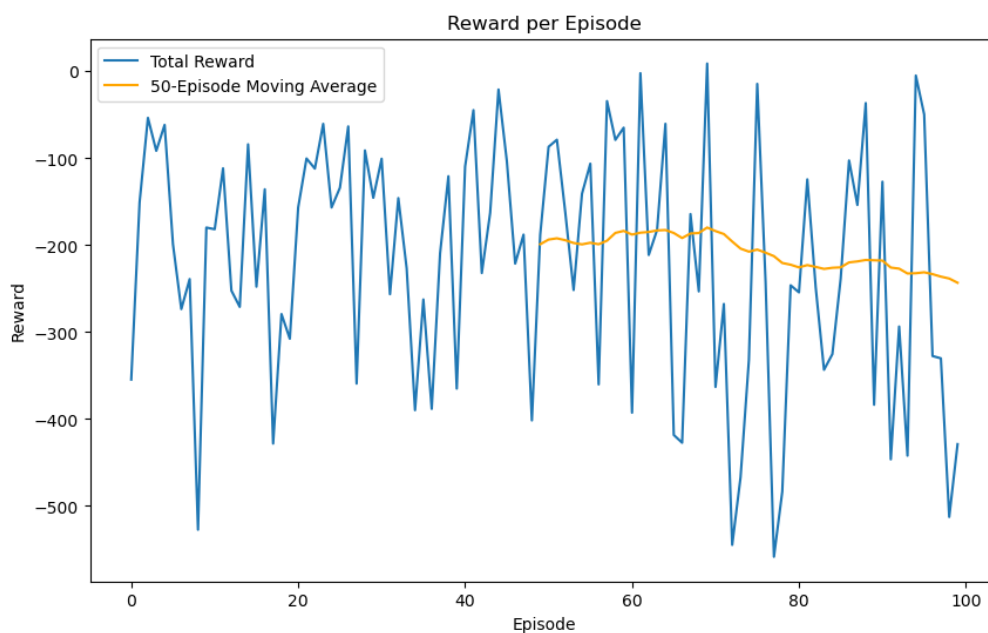
2 Plot for Offline data use as TRUE

Figure 2