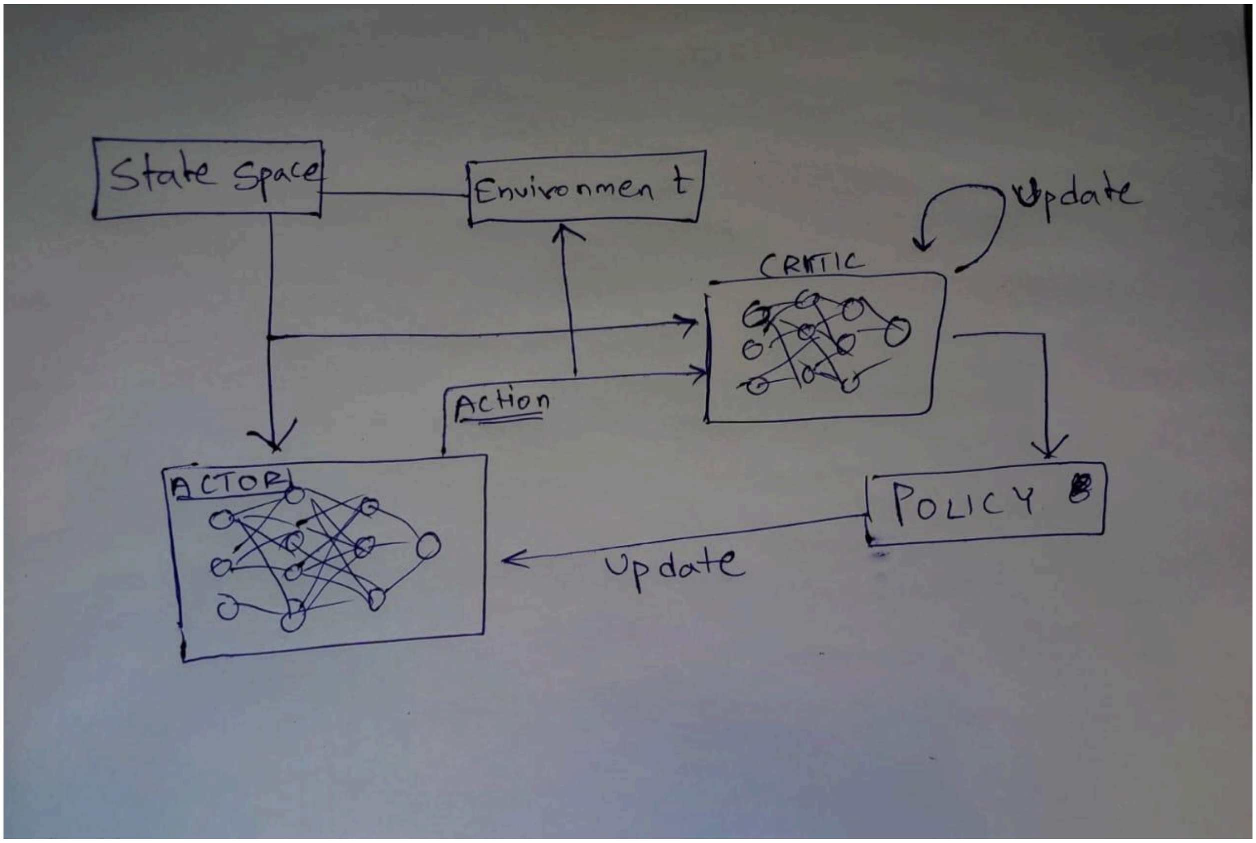
PPO agent - Proximal Policy Optimization

Overview



The Proximal Policy Optimizatino agent is better known for the effectiveness and simplicity

Training Process

- Data Collection:- The agent interacts with environment by applying actions using the current policy given the state space. It uses the reward, new state and its previous actions to update
- Advantage Estimation: Using the collected data, the agent computes the advantages and indicate the quality of action taken.
- Policy Taken: The policy is updated by optimizing a surrogate objective function through gradient descent, **which makes sure the new policy the not far from previous one using clipping**

Feedback Control Loop

- **Plant** is the pendulum system described by its dynamics
- The **controller** is a PPO agent based on observed states
- **Feedback** includes the entire state space which is x1(theta) and x2(omega)
- The **torque**(tau) is then applied which was computed by the agent

Reward Design

```
Reward = k1 * angle_error^2 + k2 * w ^2 + k3 * Action^2;
```

This is the reward function that i was using:-

- Penalty for angular deviation: A significant negative reward is given for deviation from the upright position to encourage alignment along the vertical axis
- Penalty for angular velocity: We want the pendulum to not just go up, but to stay there and stabiliez
- Penalty for action magnitude: This was just a thought following the LQR formulation where we penalize large input values

Tuning

- Tuning was needed while deciding the weights of the penalty by angle_error, omega and action magnitude
- The weight of the angle error was kept higher than omega because initially the system will need to atleast achieve some omega for the system to move.
- The contribution of omega penalty increases as the system reaches the equilibrium(because the angle_error starts approaching zero)

Obervation Space

- Angle(theta) - The angle of the pendulum from vertical down constrained between 0 and 2*pi
- Angular velocity(omega) - The rate of the change of the angle

These observation are selected as these are directly relevant for the aim we need to achieve which gives us both the current state and the dynamics involved.

Action Space

- The action space is defined as a set of discrete torque values rather than continuous values which the agent can apply ranging from -20 to 20
- The discretization allows the agent to explore theaction space at a faster pace, which gives us a trade off between precision and computational feasibility

Simlulation With noise and multiple initial condition

Validation involves testing the trained agent across multiple initial conditions to evaluate its robustness and reliability. Performance metrics focus on the time taken to stabilize the pendulum and the energy (torque) used, ensuring efficiency and effectiveness.

Noise

Noise has been added to our observations like the following. This will also be evident in the final plot

```
Observation=this.State + this.Ts.*[x1_dot;x2_dot]+ 0.001.*[randn(1);randn(1)];
```

Multiple Initial conditions

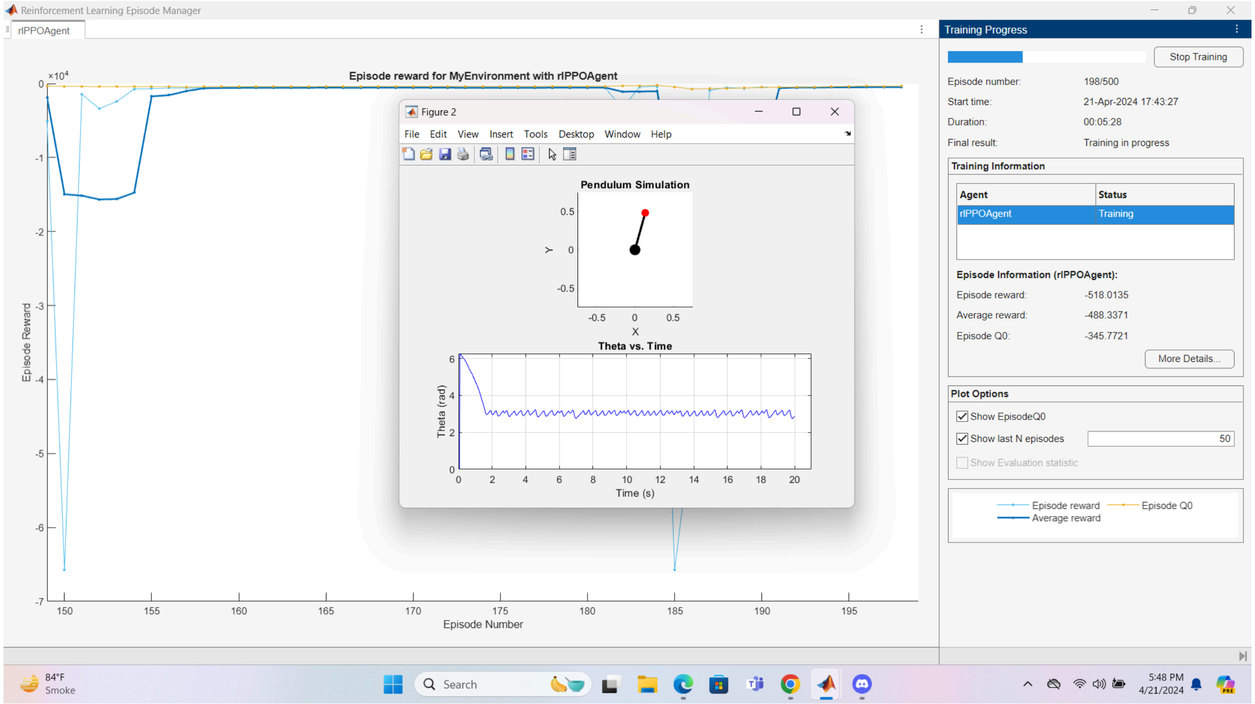
```
T0 = (rand - 0.5)  * 10 * pi / 180;  % random angle value from -5 to 5

Td0 = 0;  % Initial angular velocity

InitialObservation = [T0; Td0];

this.State = InitialObservation;
```

Random initialization of the state space has been by initializing the pendulum in the down position with angle in between -5 to 5 degree from down vertical.

The omega was kept as zero while initializing

Output Plots

- Angle vs. Time: Showcases how quickly and effectively the pendulum reaches and maintains the upright position. The pendulum oscillates with small amplitude around he upright position at Theta = PI. This is due to the noise that has been added to the observations



Episode number: 198/500
Start time: 21-Apr-2024 17:43:27
Duration: 00:05:28
Final result: Training in progress

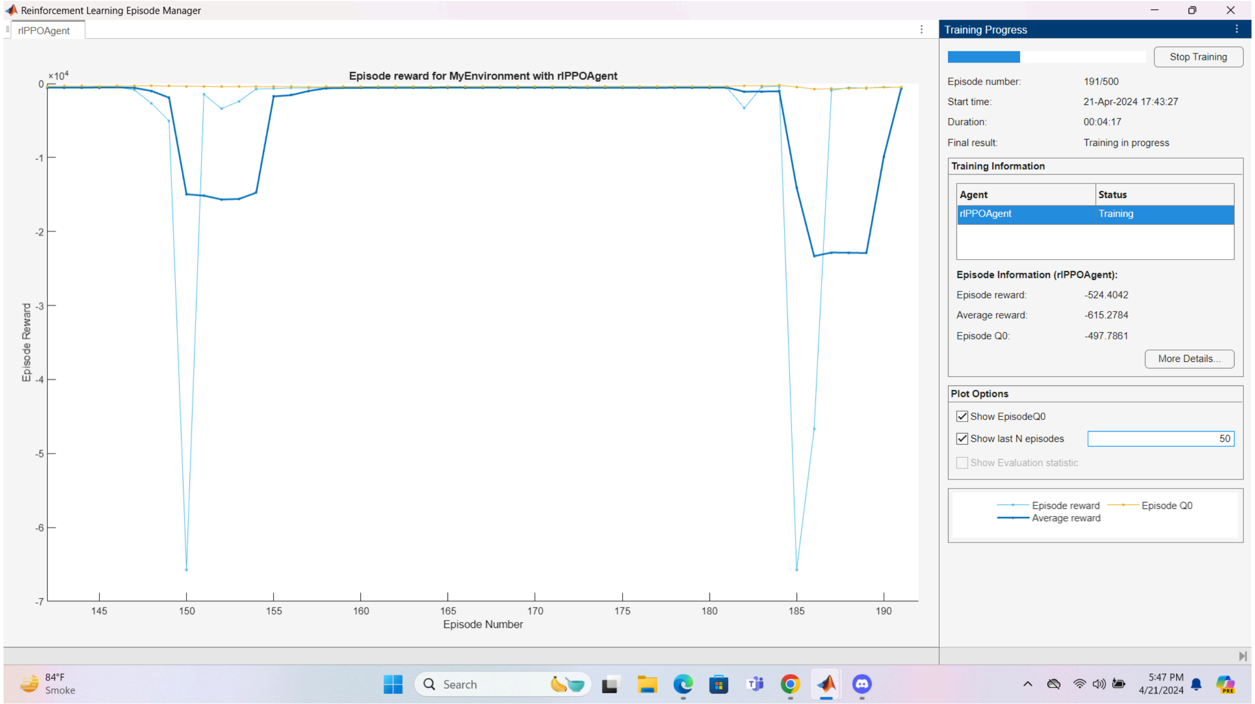
Training Information

Agent	Status
rlPPOAgent	Training

Episode Information (rlPPOAgent):
Episode reward: -518.0135
Average reward: -488.3371
Episode Q0: -345.7721

Plot Options
☑ Show EpisodeQ0
☑ Show last N episodes 50
□ Show Evaluation statistic

- We see peaks like these when agent tries something new and messes up. So it quickly revert back to the original policy



Episode number: 191/500
Start time: 21-Apr-2024 17:43:27
Duration: 00:04:17
Final result: Training in progress

Training Information

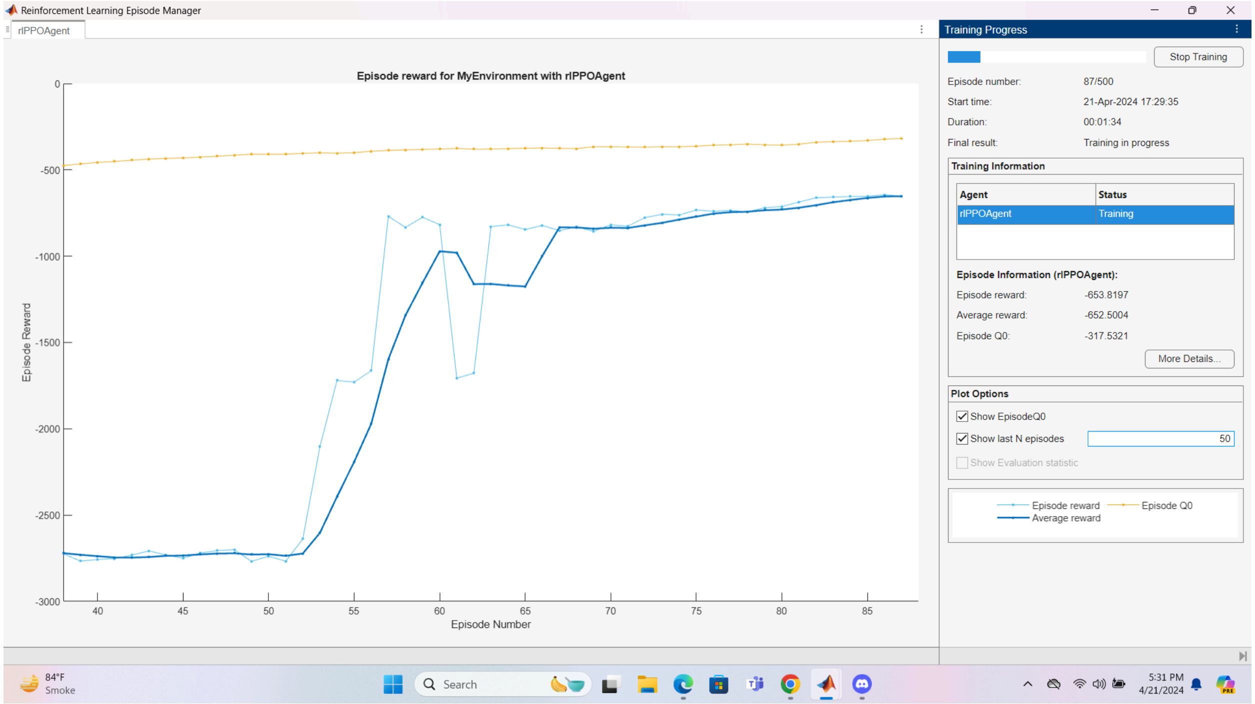
Agent	Status
rlPPOAgent	Training

Episode Information (rlPPOAgent):
Episode reward: -524.4042
Average reward: -615.2784
Episode Q0: -497.7861

More Details...

Plot Options
☑ Show EpisodeQ0
☑ Show last N episodes 50
□ Show Evaluation statistic

Episode reward — Episode Q0 — Average reward



Episode number: 87/500
Start time: 21-Apr-2024 17:29:35
Duration: 00:01:34
Final result: Training in progress

Training Information

Agent	Status
rlPPOAgent	Training

Episode Information (rlPPOAgent):
Episode reward: -653.8197
Average reward: -652.5004
Episode Q0: -317.5321

More Details...

Plot Options
☑ Show EpisodeQ0
☑ Show last N episodes 50
□ Show Evaluation statistic

Episode reward — Episode Q0 — Average reward

The agent start settling down way before the peak we saw earlier. This settling down can be seen the above graph as well

- **Torque vs Time and Angle vs Time**

Here the pendulum settles very quickly but the torque are quickly being changes due to disturbances

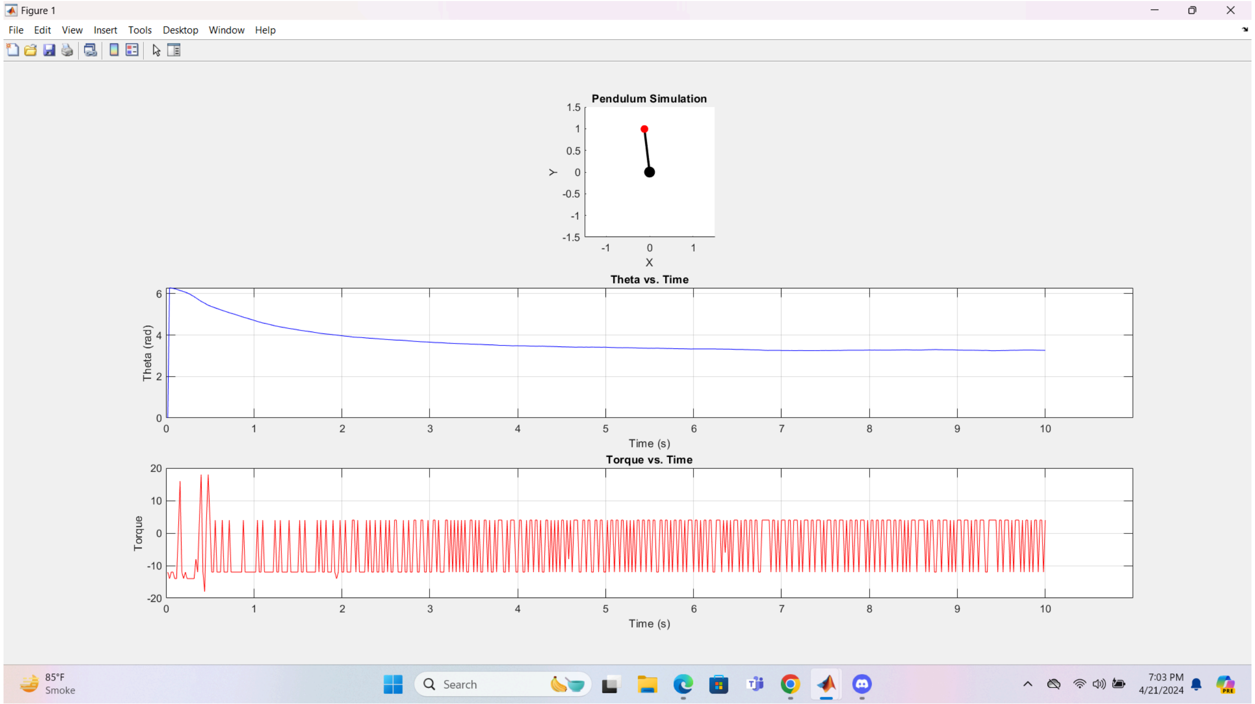


Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
Pendulum Simulation
1.5
1
0.5
0
-0.5
-1
-1.5
Y
-1
0
1
X
Theta vs. Time
Theta (rad)
6
4
2
0
0
1
2
3
4
5
6
7
8
9
10
Time (s)
Torque vs. Time
Torque
20
10
0
-10
-20
0
1
2
3
4
5
6
7
8
9
10
Time (s)
85°F
Smoke
Search
7:03 PM
4/21/2024