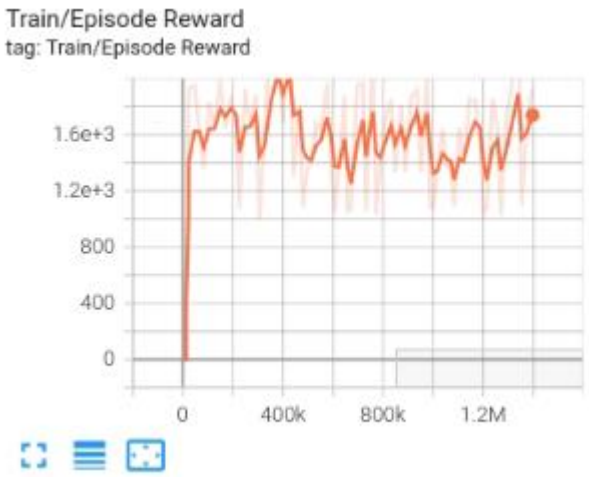
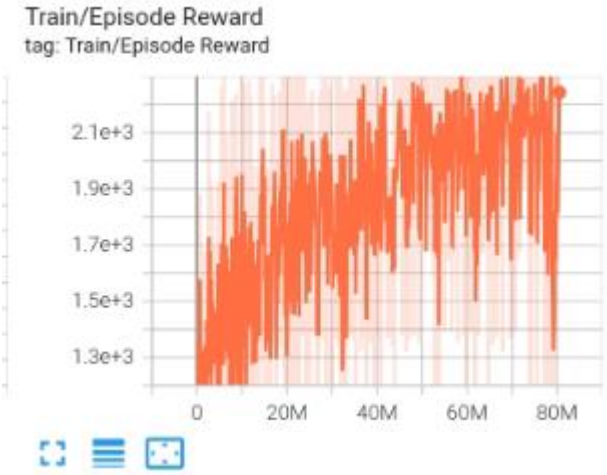
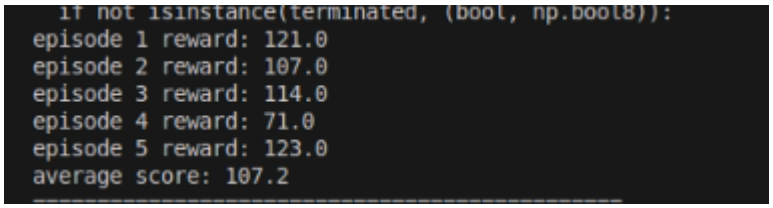
Training curve:





Evaluate

Testing result:



1. PPO is an on-policy or an off-policy algorithm? Why?

Ans: PPO is off-policy because there is a replay buffer which stores the experiences trained by old policy. In this case, PPO can be more stable and do more exploration, most importantly, faster convergence.

2. Explain how PPO ensures that policy updates at each step are not too large to avoid destabilization

Ans: When computing objective function, PPO uses a clip function to ratio which forces the value from 1-epsilon to 1+epsilon to let the new policy be close to the old one.

3. Why is GAE-lambda used to estimate advantages in PPO instead of just one-step advantages? How does it contribute to improving the policy learning process?

Ans: A common approach to estimate advantages is using TD-error or one-step advantage, but one-step advantage can be high variance, making training process noisy and unstable. On the other hand, GAE smooths out these high variance updates by taking in account both the immediate reward and long-term rewards to achieve a better balance between stability and efficiency.

4. Please explain what the lambda parameter represents in GAE-lambda, and how adjusting the

lambda parameter affects the training process and performance of PPO?

Ans: Lambda controls the bias-variance tradeoff. When lambda=0, introduce the one-step advantage, when lambda=1, GAE approximates the sum of future rewards which is Monte-Carlo estimate (computationally expensive). Small lambda will focus more on recent rewards and the immediate value of states. This leads to quicker but noisier updates, which might make the agent more reactive to short-term fluctuations. Large lambda focuses more on the long-term return and less on immediate feedback. This makes the updates smoother and more stable over time, especially in environments with sparse or delayed rewards.