RL assignment3 report

520030910342 Jiyu Liu

1. Implementation of the TRPO and PPO algorithms

Note: This section only contains the discussion of coding problems. Please refer to the code folder for the programming part.

(a) Overall, in the range of 1e-6 to 0.001, the greater the constraint δ is, the better the performance is. As we can in Figure 1, all these curves show an upward trend, but those with greater δ value can go faster than others. The first reason of this phenomenon is that a larger δ allows the algorithm to take larger steps in the policy space, which may lead to faster convergence to a better policy. If the current δ is too small, the algorithm may converge too slowly, or get stuck in local optima. The next reason is that a larger δ can result in a higher proportion of exploration, which may lead the agent to the optimal policy more efficiently. However, if δ is too small, the agent may have insufficient exploration and can't find the optimal policy within certain episodes.

(b) As we can see in Figure 2, the agent with a β value of 0 has alomost no reward, the agents with a β value of 10 and 100 have similarly good performances, and the agent with a β value of 1000 has a relatively inferior performance than these two. First, if $\beta = 0$, which means the PPO algorithm has ignored KLD penalty completely, the training process of PPO will be very unstable thus ending with with poor performance. Second, the agents with β values of 10 and 100 have satisfying performance because they provide a reasonable balance between optimizing objective function and ensuring the KL diversity constraint, allowing the agents to effectively updating their policies without taking overly large steps that result in instability. Finally, the agent with a β value of 1000 gets a relatively poor performance due to the excessive constraint. If β is set too large, the KL diversity will dominate the optimizing function, which may lead the algorithm to mainly focus on ensuring the constraints instead of updating the policy, so the algorithm will converge very slowly and become more likely to be stuck in suboptimal policies.

(c) I think the impact of δ in TRPO algorithm and the impact of β in PPO algorithm are similar though they achieve the purpose in different approaches. First of all, they are all designated to constraint KL divergence between new policy and old policy, aiming to avoid instability during training process. However, this purpose is achieved in different ways: δ in TPRO algorithm determines the extent of trust region and limits policy updates to this region, while β in PPO algorithm determines the weight of KL divergence in optimizing function.

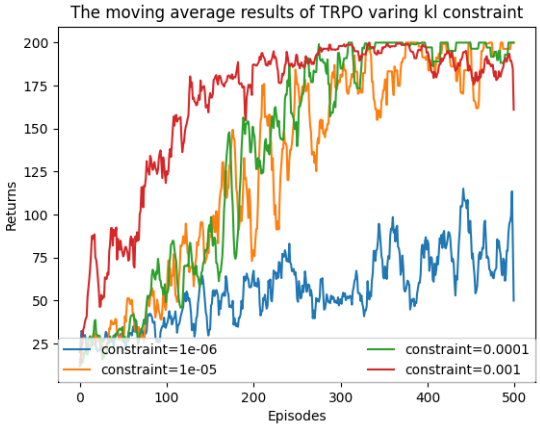

Figure 1: *TRPO Comparison of different ϵ values*

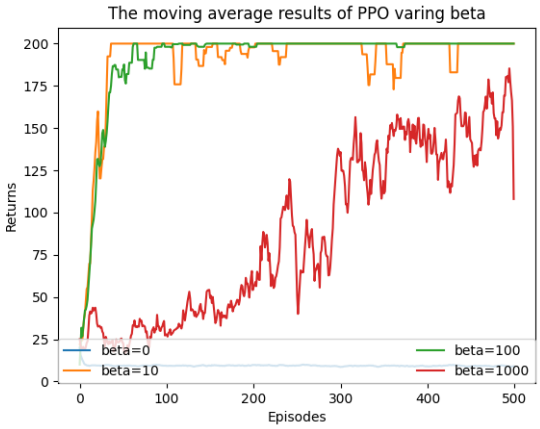

Figure 2: *PPO comparison of different β values*