Jianxun Zhou
zhou-jx23@mails.tsinghua.edu.cn

EXPERIMENT 1: POLICY GRADIENT

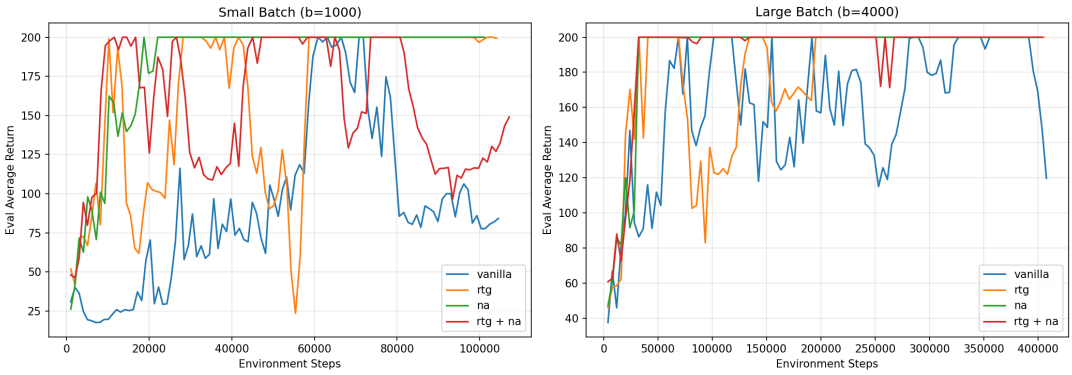


Figure 1: **Eval Return** vs. **Env Steps**

1. Without **advantage normalization**, the **reward-to-go** value estimator performs better than the **trajectory-centric** one.

2. **advantage normalization** helps the policy to achieve the same performance in **fewer environment step**.

3. In fact, the training with a **larger batch size** achieves the same performance in **more environment steps** but **fewer training iteration steps**.

4. The command line configuration i use is totally the same as the one given in homework document.

EXPERIMENT 2: NEURAL NETWORK BASELINE

BASELINE LOSS

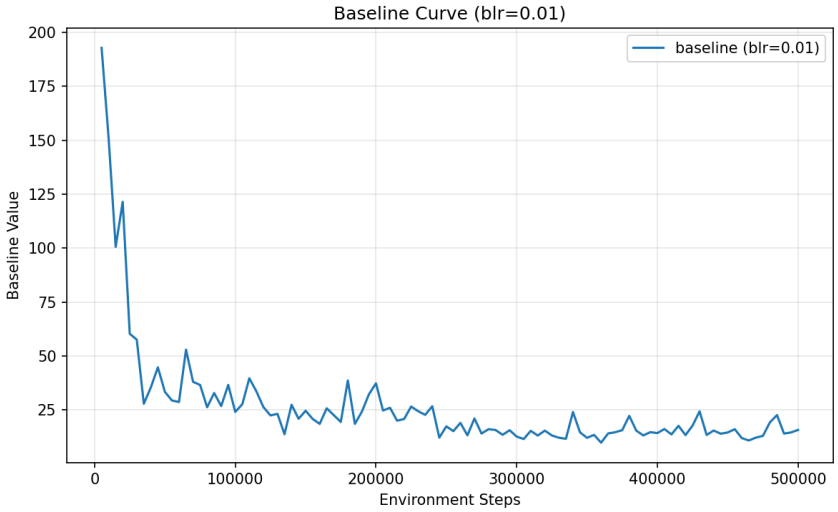


Figure 2: **Baseline Loss** vs. **Env Steps** with baseline learning rate=0.01

RETURN COMPARISON

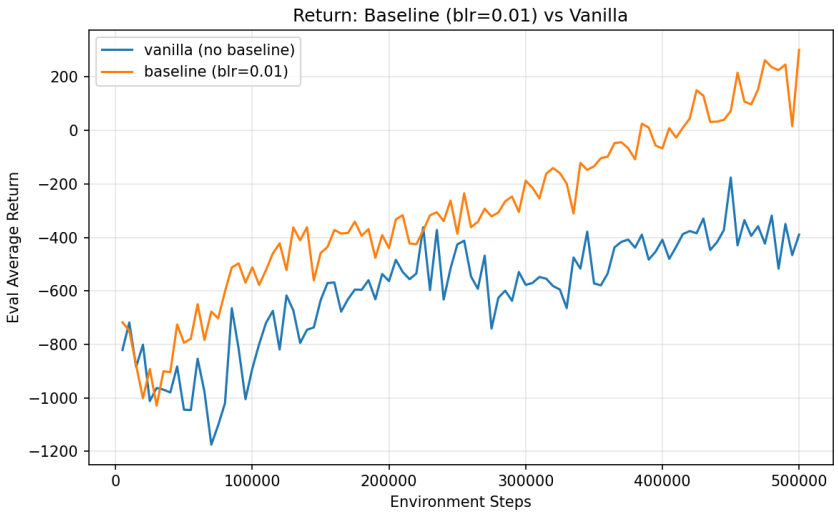
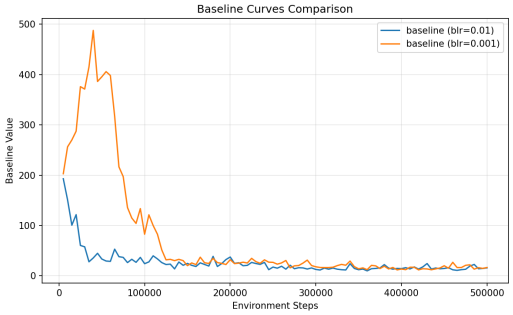
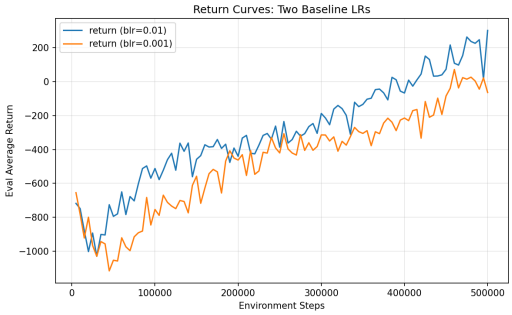


Figure 3: **Average Return** vs. **Env Steps** with baseline learning rate=0.01

BASELINE LEARNING RATE COMPARISON



(a) Baseline Loss with different baseline learning rate



(b) Baseline Return with different baseline learning rate

Compared to **baseline learning rate = 0.01**, training with **baseline learning rate 0.001** causes slower **baseline loss convergence** and a slower increase in **average return**.

EXPERIMENT 3: GENERALIZED ADVANTAGE ESTIMATION

RETURN CURVES

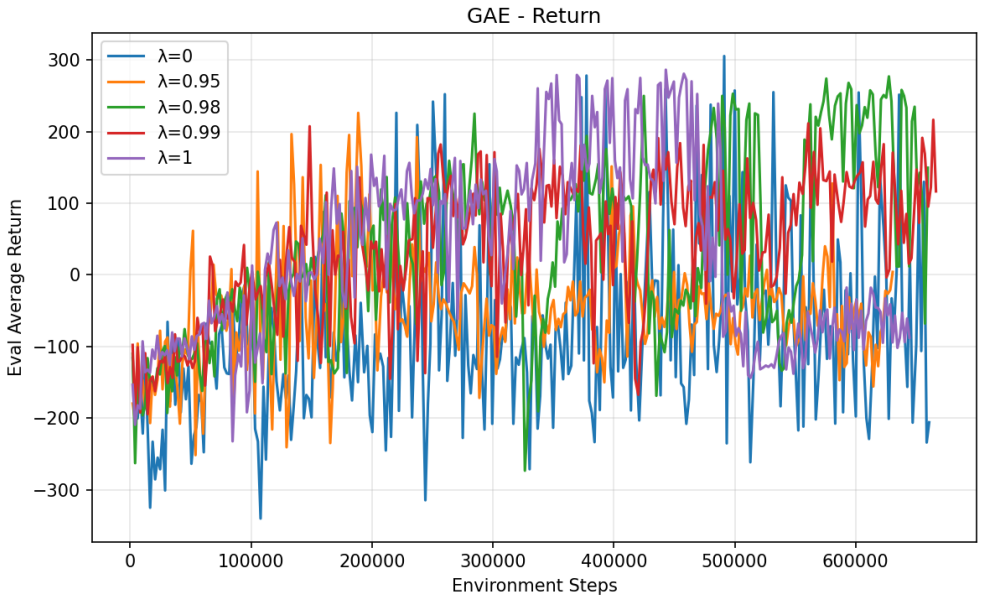


Figure 5: **Average Return** vs. **Env Steps** with different GAE lambda

While increasing λ generally improves policy performance, an excessively large λ leads to instabilities in the later stages of training.

ANALYSIS LAMBDA

$\lambda = 0$ correspond to $\hat{A}(s_t, a_t) = r(s_t, a_t) + \gamma \hat{V}(s_{t+1}) - \hat{A}(s_t)$, which means we totally use prediction from critic as the estimation of state value. At the beginning of training, the critic's predictions are **highly biased**, which leads to **suboptimal** policy performance. However, because the critic's predictions have **low variance**, the resulting training curves exhibit fewer oscillations.

$\lambda = 1$ correspond to $\hat{A}(s_t, a_t) = \sum_{t'=t}^{T} \gamma^{t'-t} r(s_{t'}, a_{t'}) - \hat{V}(s_t)$, which means we totally use the following steps' rewards from environment to estimate the advantage of a_t. This causes high variance since the estimate is based on the sum of actual rewards, it is unbiased but suffers from high variance due to the stochasticity of the environment.

In the figure 5, the blue curve($\lambda = 0$)'s *eval_return* struggles near -100 because of huge bias caused by estimating the value totally depend on critic. And other four curves obviously perform

better than the blue one. However, due to the high variance from stochastic step reward from environment, sometimes the return falls down violently.

EXPERIMENT 4: HYPERPARAMETERS AND SAMPLE EFFICIENCY

4.1 HYPERPARAMETERS

Parameter	**Value**
iteration	100
batch size	500
learning rate	0.01
discount	0.99
GAE lambda	0.98
reward to go	true
use baseline	true
baseline learning rate	0.01
baseline gradient steps	5
n layers	2
layer size	64

Table 1: Hyperparameters

4.2 RETURN CURVES

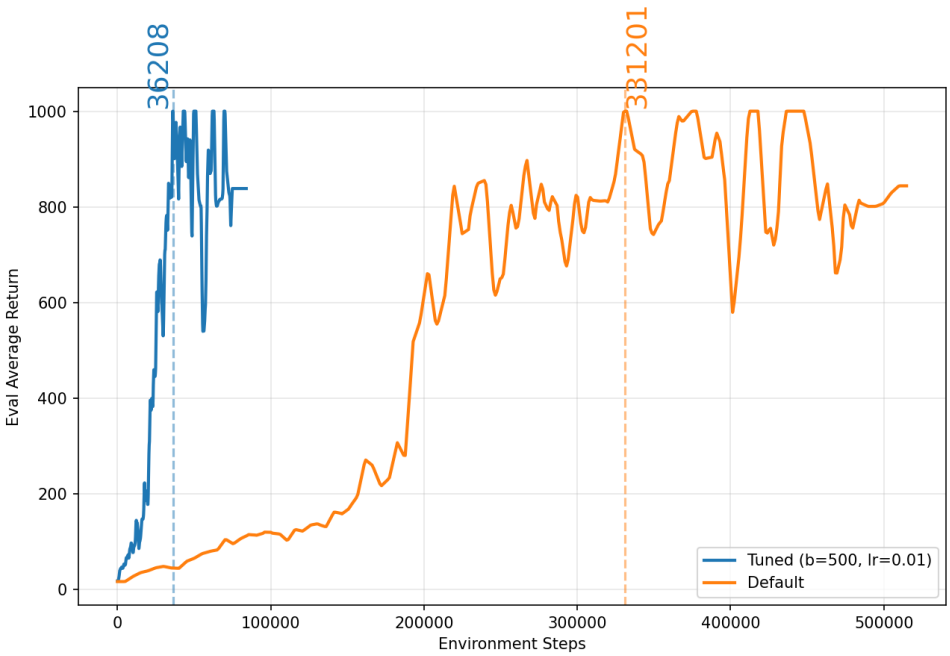


Figure 6: Eval Average Return-Env Steps(default VS. self-define)

ANALYSIS

APPLY POLICY GRADIENTS

a.

$$\begin{aligned}
\nabla_\theta J(\theta) &= \mathbb{E}_{\tau\sim p_\theta(\tau)}[\nabla_\theta \log P_\theta(\tau) r(\tau)] \\
&= \sum_{k=1}^{\infty} \theta^{k-1}(1-\theta)\cdot\left(\frac{k}{\theta}-\frac{1}{1-\theta}\right)\cdot k \\
&= (1-\theta)\sum_{k=1}^{\infty} k^2\theta^{k-1} - \theta\sum_{k=1}^{\infty} k\theta^{k-1} \\
&= (1-\theta)\cdot\frac{1+\theta}{(1-\theta)^3} - \theta\cdot\frac{1}{(1-\theta)^2} \\
&= \frac{1}{(1-\theta)^2}
\end{aligned}$$

b. Calculate $J(\theta) = \mathbb{E}_{\pi_\theta}[R(\tau)]$ recursively.

$$J(\theta) = \theta \cdot (1 + J(\theta)) + (1 - \theta) \cdot 0$$
$$\Rightarrow J(\theta) = \frac{\theta}{1 - \theta}$$

Differentiate with respect to θ:

$$\nabla_\theta J(\theta) = \frac{1}{(1 - \theta)^2}$$

This matches the result from the direct calculation.

POLICY GRADIENT VARIANCE

$$\begin{aligned}
\mathbb{E}[(\nabla_\theta J(\theta))^2] &= \sum_{k=1}^{\infty} \theta^{k-1}(1-\theta) \cdot \left[\left(\frac{k}{\theta} - \frac{1}{1-\theta} \right) k \right]^2 \\
&= \sum_{k=1}^{\infty} \left(\frac{k^4}{\theta^2} + \frac{k^2}{(1-\theta)^2} - \frac{2k^3}{\theta(1-\theta)} \right) \theta^{k-1}(1-\theta) \\
&= \frac{1-\theta}{\theta^2} \sum_{k=1}^{\infty} k^4 \theta^{k-1} + \frac{1}{1-\theta} \sum_{k=1}^{\infty} k^2 \theta^{k-1} - \frac{2}{\theta} \sum_{k=1}^{\infty} k^3 \theta^{k-1} \\
&= \dots \\
&= \frac{1 + 9\theta + 4\theta^2}{\theta(1-\theta)^4}
\end{aligned}$$

Note: The intermediate steps involve calculating sums of the form $\sum k^n x^k$. After simplification (omitted for brevity), we arrive at the variance term.

From equation $Var(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$:

$$\begin{aligned}
Var[\nabla_\theta J(\theta)] &= \mathbb{E}[(\nabla_\theta J(\theta))^2] - (\mathbb{E}[\nabla_\theta J(\theta)])^2 \\
&= \frac{1 + 9\theta + 4\theta^2}{\theta(1-\theta)^4} - \left(\frac{1}{(1-\theta)^2} \right)^2 \\
&= \frac{1 + 9\theta + 4\theta^2 - \theta}{\theta(1-\theta)^4} \\
&= \frac{4\theta^2 + 8\theta + 1}{\theta(1-\theta)^4}
\end{aligned}$$

APPLY REWARD-TO-GO

a. For a trajectory with horizon k, the reward-to-go of step t is $(k-t)$ if $t \leq k$ and 0 if $t = k$. Let G_{τ_k} denote the policy gradient of trajectory with horizon k.

$$\begin{aligned} G_{\tau_k} &= \sum_{t=0}^{k-1} \frac{1}{\theta}(k-t) + 0 \\ &= \sum_{t=0}^{k-1} \frac{k}{\theta} - \sum_{t=0}^{k-1} \frac{t}{\theta} \\ &= \frac{k(k-1)}{2\theta} \end{aligned}$$

Expectation of G_{τ_k}:

$$\begin{aligned} \nabla_\theta J(\theta) &= \sum_{k=1}^{\infty} G_{\tau_k} \cdot \theta^{k-1}(1-\theta) \\ &= \frac{1-\theta}{2} \sum_{k=1}^{\infty} k(k-1)\theta^{k-2} \\ &= \frac{1-\theta}{2} \frac{d^2}{d\theta^2} \sum_{k=1}^{\infty} \theta^k \\ &= \frac{1-\theta}{2} \frac{d^2}{d\theta^2} \frac{\theta}{1-\theta} \\ &= \frac{1-\theta}{2} \frac{2(1-\theta)}{(1-\theta)^4} \\ &= \frac{1}{(1-\theta)^2} \end{aligned}$$

This form matches the one without reward-to-go, confirming that reward-to-go estimator is unbiased.

b. With reward-to-go:

$$\begin{aligned} \mathbb{E}[(\nabla_\theta J(\theta))^2] &= \sum_{k=1}^{\infty} \theta^{k-1}(1-\theta) \cdot \frac{k^2(k-1)^2}{4\theta^2} \\ &= \dots \\ &= \frac{1+4\theta+\theta^2}{\theta(1-\theta)^4} \end{aligned}$$

Variance:

$$\begin{aligned} Var(\nabla_\theta J(\theta)) &= \frac{1+4\theta+\theta^2}{\theta(1-\theta)^4} - (\frac{1}{(1-\theta)^2})^2 \\ &= \frac{\theta^2+3\theta+1}{\theta(1-\theta)^4} \end{aligned}$$

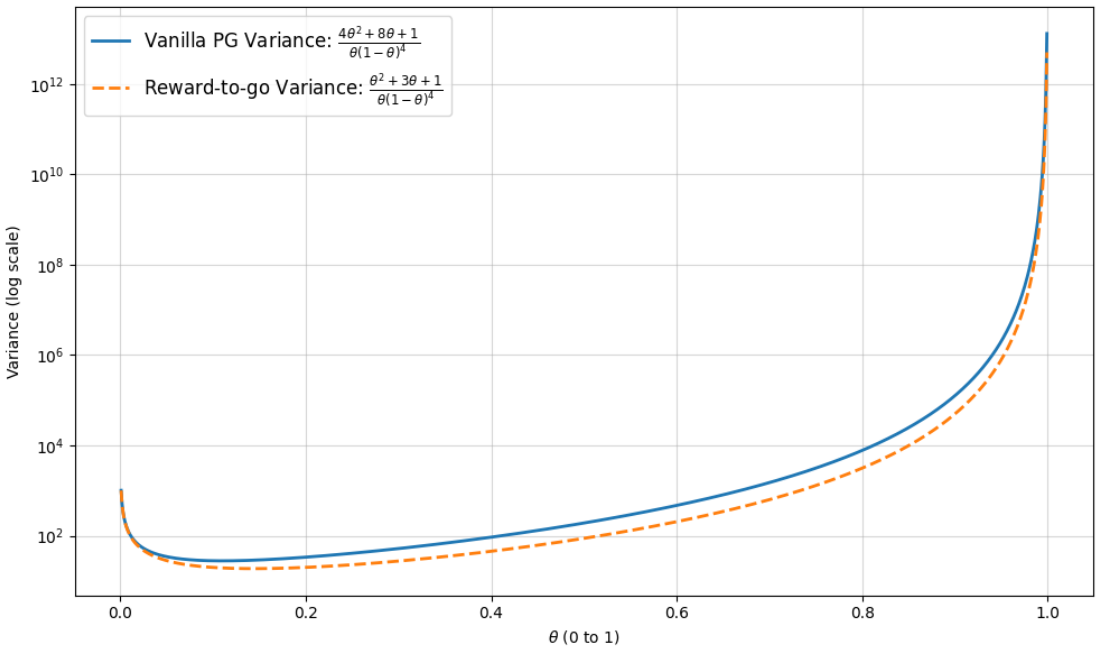


Figure 7: **Variance Comparison(reward-to-go VS. default)**

IMPORTANCE SAMPLING

a. Only the trajectory ending at s_H contributes to the expectation of Policy Gradient.

$$\begin{aligned} J(\theta) &= \mathbb{E}_{\tau\sim p_{\theta'(\tau)}}[\frac{p_\theta(\tau)}{p_{\theta'}(\tau)}r(\tau)] \\ &= \theta'^{H-1}\cdot(\frac{\theta}{\theta'})^{H-1}\cdot 1 \\ &= \theta^{H-1} \end{aligned}$$

Differentiate with respect to θ:

$$\nabla_\theta J(\theta) = (H-1)\theta^{H-2}$$

b.

$$\begin{aligned} \mathbb{E}[(\nabla_\theta J(\theta))^2] &= \theta'^{H-1}\cdot[(\frac{\theta}{\theta'})^{H-1}\cdot\frac{H-1}{\theta}\cdot 1]^2 \\ &= \frac{(H-1)^2\cdot\theta^{2H-4}}{\theta'^{H-1}} \end{aligned}$$

Variance:

$$\begin{aligned} Var[\nabla_\theta J(\theta)] &= \mathbb{E}[(\nabla_\theta J(\theta))^2] - \mathbb{E}^2[(\nabla_\theta J(\theta))] \\ &= (H-1)^2\theta^{2H-4}(\frac{1}{\theta'^{H-1}} - 1) \end{aligned}$$

As H increases, the variance grows exponentially.