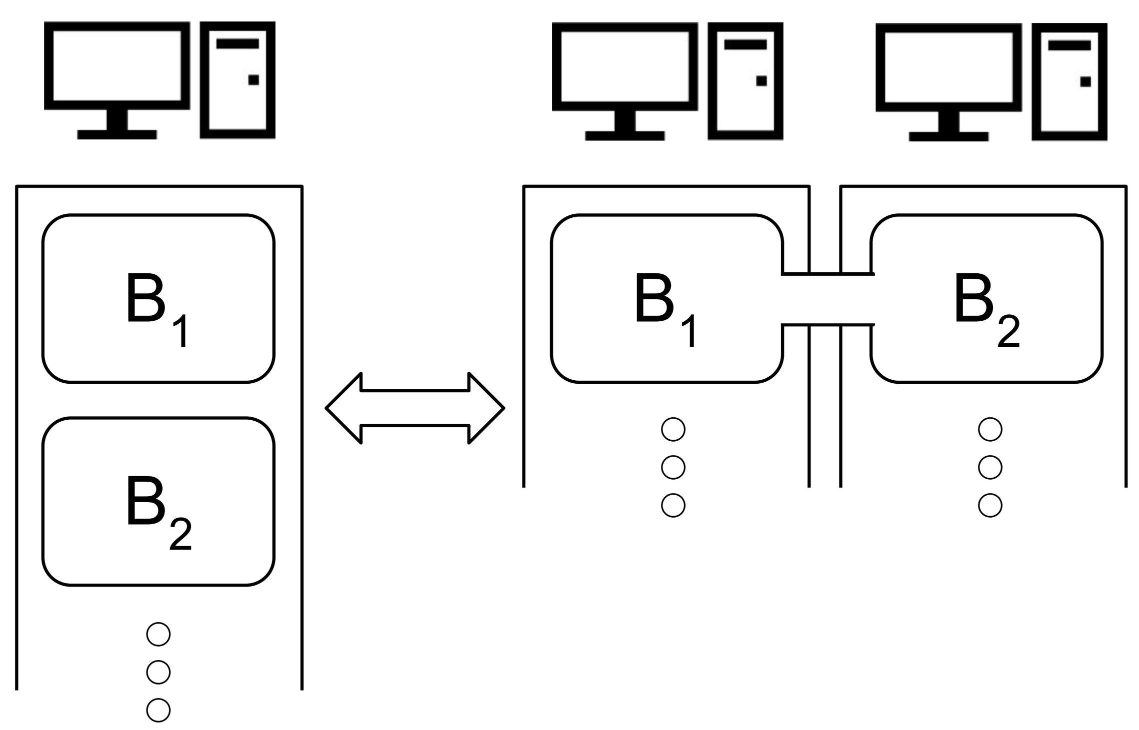
Batch size-invariance for policy optimization

OpenAI — NeurIPS 2022

Jacob Hilton, Karl Cobbe, John Schulman

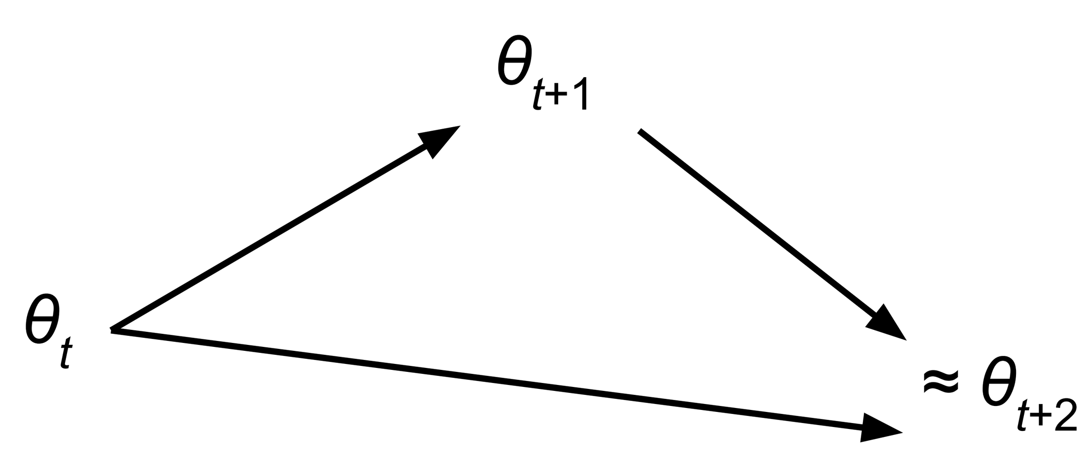
Batch size-invariance

This means that changes to the batch size can be compensated for by changes to other hyperparameters, such as the learning rate.



SGD is batch size-invariant at small batch sizes

two steps, batch size n, learning rate α
≈ one step, batch size $2n$, learning rate 2α



See Mandt et al. [2017] for a thorough explanation.

PPO has two batch sizes

optimization batch size = per gradient step
iteration batch size = per alternation between rollout and optimization

We consider simultaneously changing both.

Decoupled PPO objective

PPO [Schulman et al., 2017] uses π_{old} in two ways, which can be decoupled:

- Importance sampling – must use behavior policy
- KL penalty – can use another "proximal" policy

$$L^{\text{KLPEN}}_{\text{decoupled}}(\theta) := \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{behav}}}(a_t \mid s_t)}\hat{A}_t - \beta\,\mathrm{KL}\left[\pi_{\theta_{\text{prox}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)\right]\right]$$

PPO is not iteration batch-size invariant because the iteration batch size determines the age of the proximal policy used as the KL penalty target.

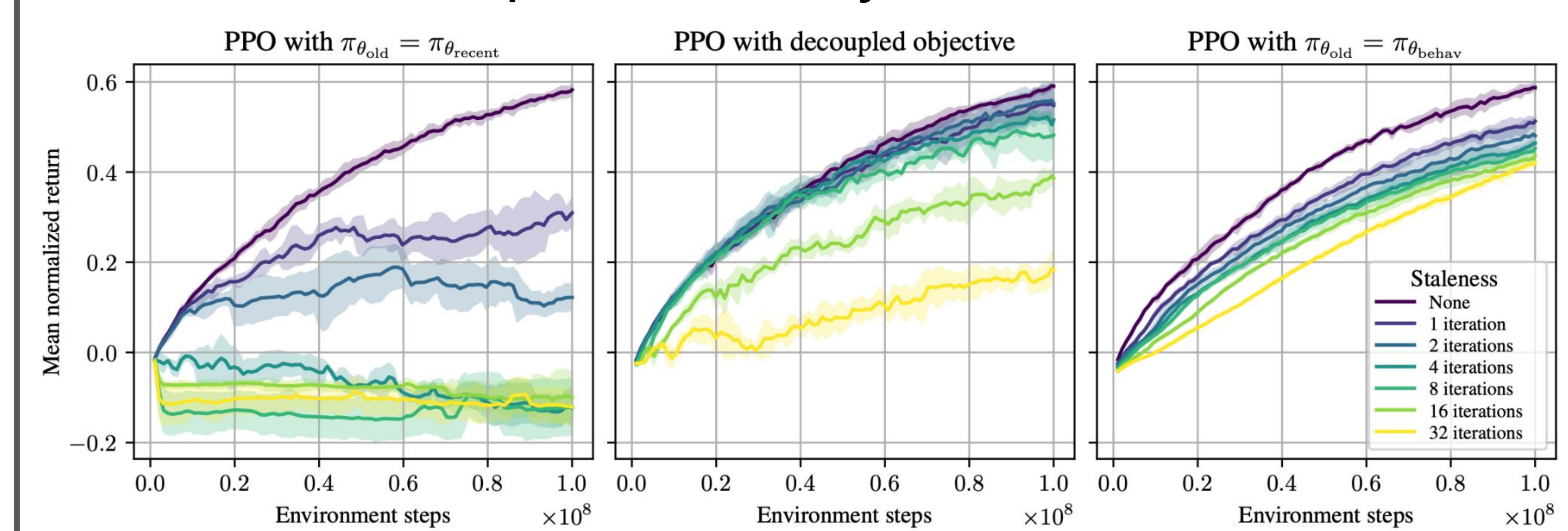
PPO-EWMA

This is the same as PPO, but using the decoupled objective with π_{prox} = EWMA(π) (exponentially-weighted moving average of weights).

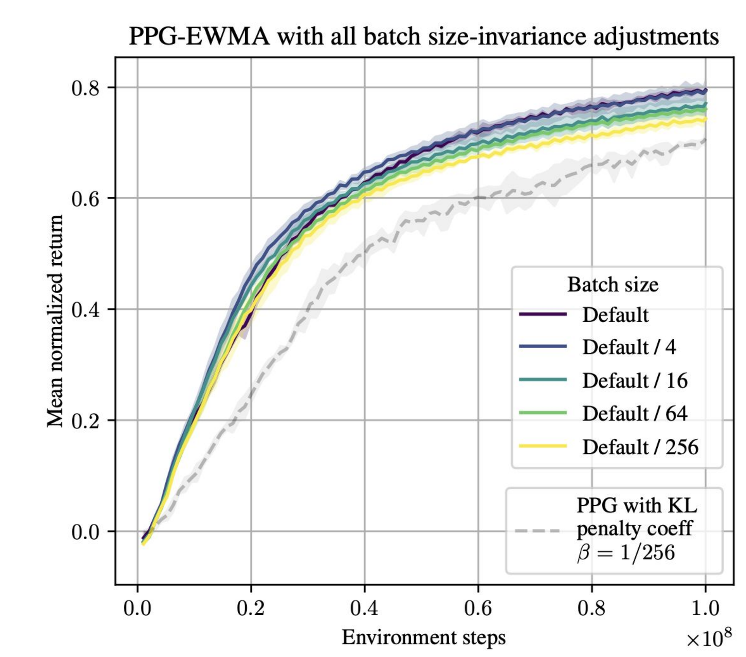
To make PPO-EWMA iteration batch size-invariant, adjust the decay rate of the EWMA when changing the iteration batch size to keep the age of the proximal policy constant.

Experiments

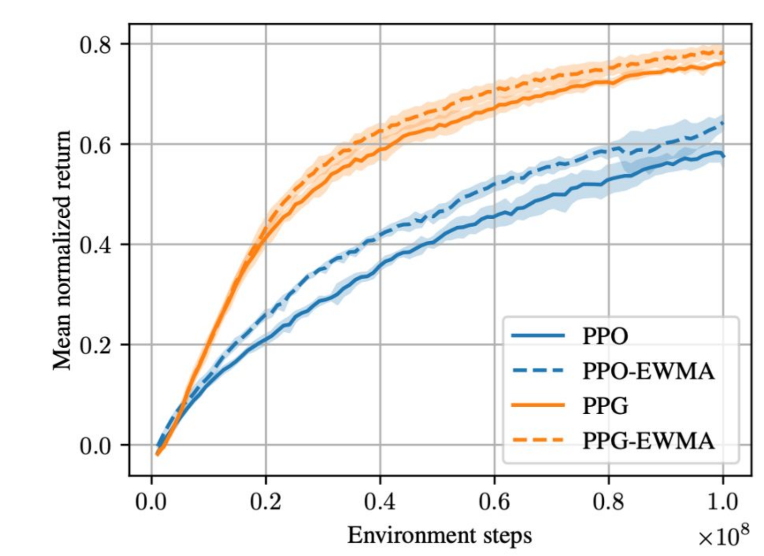
Staleness experiment (with buffered rollout data) validates decoupled PPO objective



High degree of batch size-invariance



EWMA also improves performance slightly



Experiments use PPO and PPG [Cobbe et al., 2020] on Procgen Benchmark.

Takeaways

Need to control *how fast* the policy changes, but do not need to keep the policy that close to the behavior policy specifically. So PPO is more of a natural gradient method than a trust region method.

When scaling computational resources, either keep iteration batch size constant, or use PPO-EWMA.

References

K. Cobbe, J. Hilton, O. Klimov, and J. Schulman. Phasic policy gradient. *arXiv:2009.04416*, 2020.

S. Mandt, M. D. Hoffman, and D. M. Blei. Stochastic gradient descent as approximate Bayesian inference. *arXiv:1704.04289*, 2017.

J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.