Combining PPO and Evolutionary Strategies for Better Policy Optimization

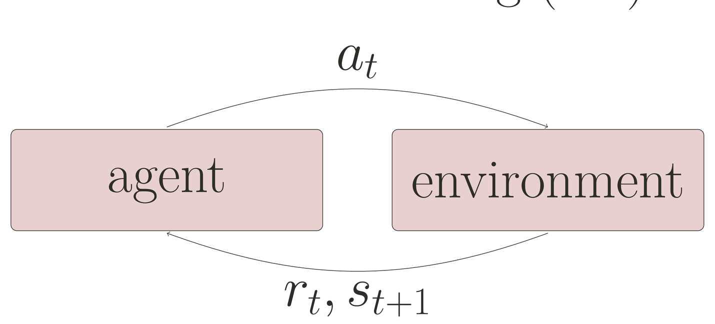
Jennifer She
Computer Science, Stanford University
jenshe@stanford.edu

Objective

- Propose and implement hybrid policy optimization methods inspired by Proximal Policy Optimization **(PPO)** and Natural Evolutionary Strategies **(ES)** in order to leverage their individual strengths
- Compare hybrid methods against PPO and ES in two OpenAI environments: **CartPole** and **BipedalWalker**

Background

Under the reinforcement learning (RL) framework



the goal of **policy optimization** is to find a policy $\pi_\theta : S \times A \to [0,1]$ defining $\Pr(a_t = a|s_t = s)$ that maximizes the expected return

$$J(\theta) = \mathbb{E}_{\tau \sim p(\tau;\theta)}[\Sigma r_t]$$

PPO updates π_θ via an approximation of $\nabla_\theta J(\theta)$

- *pro:* it uses **gradient information** to guide its updates, which helps it to zero-in on potential solutions
- *con:* it may get stuck at a local optima as a result

ES parameterizes θ with

$$\Theta = \bar{\theta} + \sigma\epsilon, \epsilon \sim \mathcal{N}(0, I)$$

which it updates by sampling $\{\theta^{(1)}, ..., \theta^{(k)}\}$ weighted by their return

$$\frac{1}{k\sigma}\Sigma_{i=1}^{k}\{\epsilon_t \Sigma r_t|_{\tau\sim p(\tau;\theta^{(i)})}\} \quad (1)$$

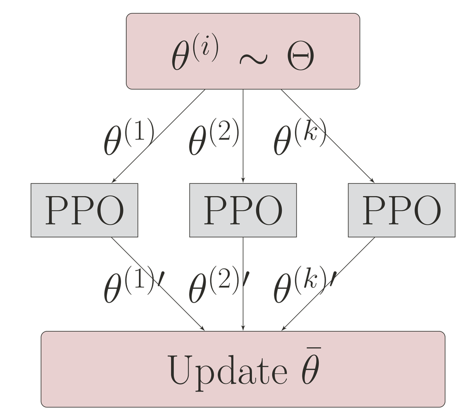
- *pro:* it incorporates **stochasticity** in the space of θ for better exploration of π_θ
- *con:* it treats the RL problem as a black-box

The goal is then:

To build hybrid methods that both leverage **gradient information** and are **stochastic** in θ

Methods

ES-PPO



- Sample $\theta^{(i)}$ as in ES, but instead, run PPO with these as initializations to obtain $\theta^{(i)\prime}$
- Update π_θ by (1) with modified perturbations

$$\epsilon_t' = \frac{1}{\sigma}(\theta^{(i)\prime} - \bar{\theta})$$

MAX-PPO

- Run ES/PPO as above but directly set $\bar{\theta}$ to $\theta^{(i)\prime}$ with the highest return

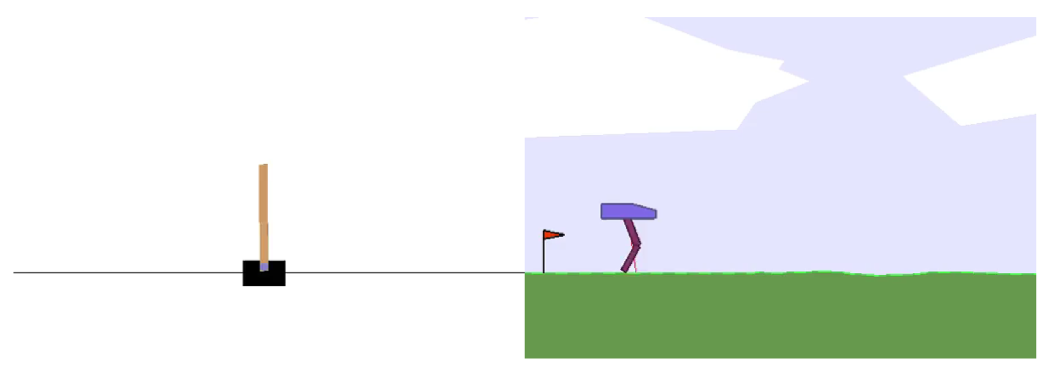
$$\text{argmax}_i \Sigma r_t|_{\tau\sim p(\tau;\theta^{(i)\prime})}$$

ALT-PPO

- Run ES every j PPO iterations

We compare these methods to **ES** and **PPO**

Environments



(a) CartPole (b) BipedalWalker

CartPole-v0 (CP)

- $S \subset \mathbb{R}^4$, $A = \{0, 1\}$
- **Objective**: Move cart to keep pole upright
- **Rewards**: +1 every timestep for a max of 200
- **Termination**: Pole falls / cart goes off screen or episode reaches max of 200 timesteps

BipedalWalker-v2 (BW)

- $S \subset \mathbb{R}^{24}$, $A = [-1, 1]^4$
- **Objective:** Maneuver walker to right-most side of environment (target) without falling
- **Rewards:** $+\epsilon$ for moving forward, for a total of 300 on agent reaching target; -100 for falling
- **Termination:** Walker reaches target or falls

Architecture Details

ES

$$\pi_\theta(a|s) = \mathbf{1}[a = f_\theta(s)]$$

where f_θ is a fully-connected neural network

1. FC(dim$(s) \times 100$) + ReLU
2. FC($100 \times$ dim(a))
3. Sigmoid + $\mathbf{1}[.]$ (**CP**) or Tanh (**BW**)

PPO/Hybrids

$$\pi_\theta(a|s) \sim \text{Bernoulli}(g_\theta(s)) \quad \textbf{(CP)}$$
$$\pi_\theta(a|s) \sim \mathcal{N}(g_\theta(s), \sigma) \quad \textbf{(BW)}$$

where g_θ is a fully-connected neural network

1. FC(dim$(s) \times 100$) + ReLU
2. FC(100×100) + ReLU
3. FC($100 \times$ dim(a))
4. Sigmoid (**CP**) or Tanh (**BW**)

Results

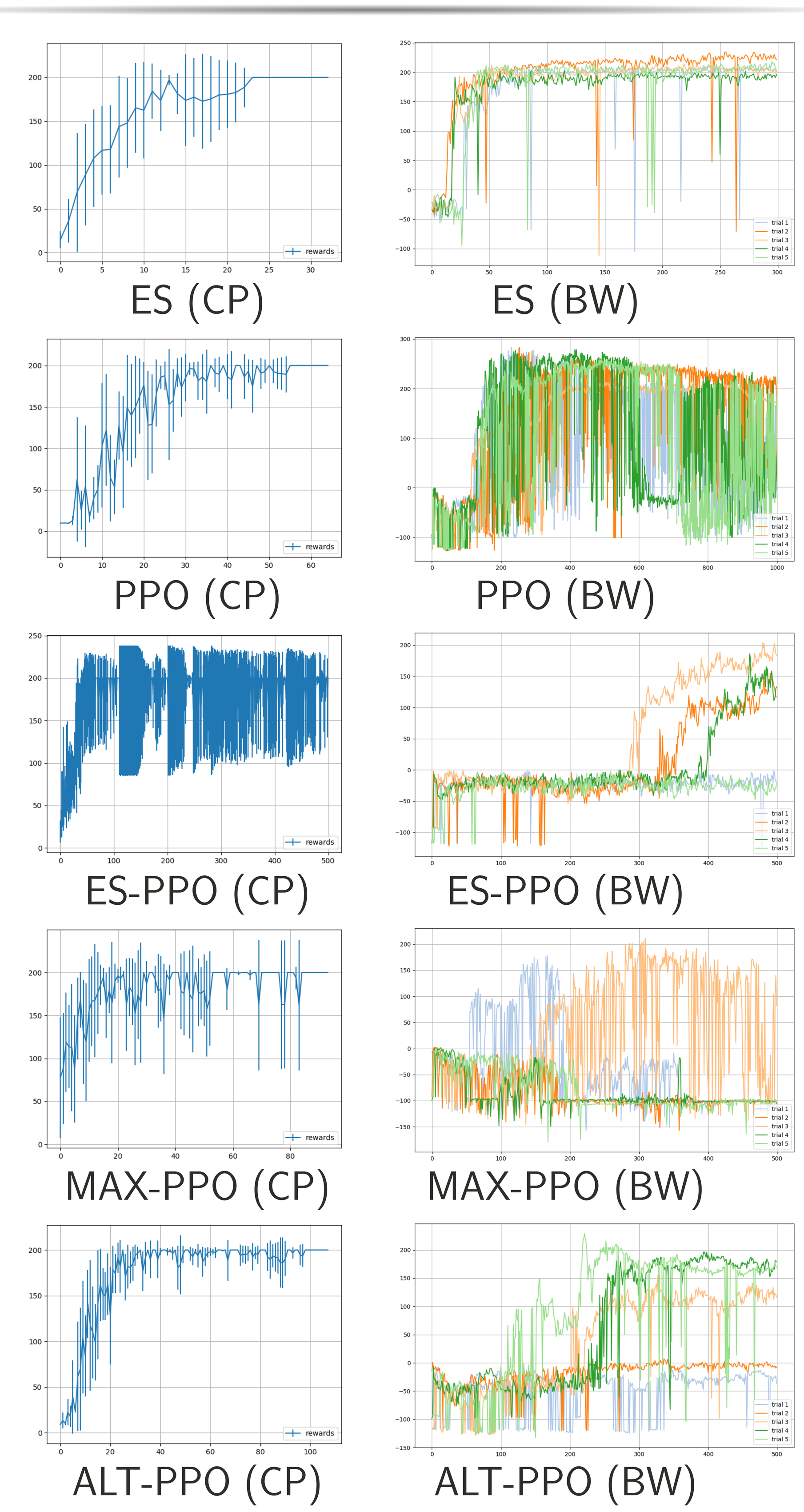


Figure: Episode returns over training across 5 trials each

	Return	Training Time
ES	200.0	60.59
PPO	200.0	53.74
ES-PPO	200.0	515.03
MAX-PPO	200.0	363.52
ALT-PPO	200.0	131.24

Table: Final results from CP averaged across 5 trials

Discussion

- **PPO** and **ES** performed well on both tasks
- **PPO:** Training instability (**BW**) likely a result of reusing samples from $\pi_{\theta_{\text{old}}}$
- **ES:** Evaluating $\theta^{(i)}$ is slow without leveraging large-scale parallel compute $\to$ extending **ES-PPO** and **MAX-PPO** from ES exponentiated this problem, and forced us to choose max sample size $k = 5$ for **BW**
- **ES-PPO:** PPO calls may drive $\theta^{(i)\prime}$ far from $\bar{\theta}$; thus a weighted average of returns at $\theta^{(i)\prime}$ may no longer be a good predictor of return at weighted average of $\theta^{(i)\prime} \to$ misleading update signals
- **MAX-PPO:** Mitigates averaging problem of ES-PPO but may lead away from a good solution when all neighbouring $\theta^{(i)\prime}$ have low returns $\to$ high variance
- **ALT-PPO:** Mitigates high computation cost of ES-PPO and MAX-PPO but its stochasticity may lead away from a good solution when neighbour $\theta^{(i)\prime}$ have low (but different) returns

Future Directions

- Investigate trade-offs in **sample efficiency** and variance in the case of **PPO**
- Investigate ways to leverage **high-compute** in the case of **ES-PPO** and **MAX-PPO**
- Investigate stochasticity with **adaptive variance** (using gradient information) to avoid moving away from good solutions
- Investigate more **complex environments** where **ES** and **PPO** fail

Acknowledgements

We thank **Mario Srouji (TA)** for the project idea, and help during the project.