

# Local Optimization Achieves Global Optimality in Multi-Agent Reinforcement Learning

Yulai Zhao

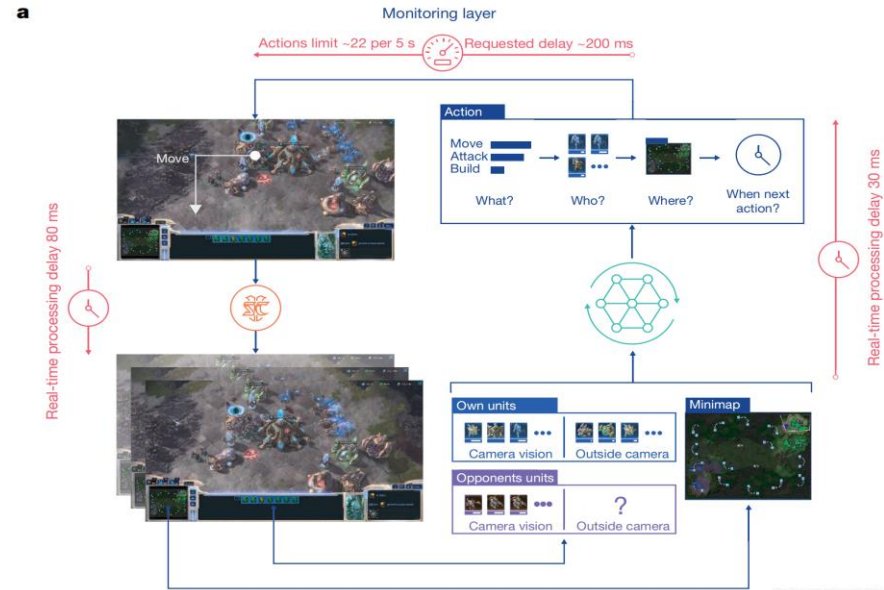
Zhuoran Yang

Zhaoran Wang

Jason Lee

# Backgrounds

- Multi-agent reinforcement learning (MARL) has demonstrated many empirical successes, e.g. strategic games (Go, StarCraft II...)
- Policy optimization methods are widely used in MARL (AlphaGo, LOLA...)



## Main challenges in MARL (Zhang 2021)

1. non-stationarity: each action taken by one agent affects the total reward and the transition of state.
2. scalability: taking other agents into consideration, each individual agent would face the joint action space, whose dimension increases exponentially with the number of agents
3. function approximation: closely related to the scalability issue, the state space and joint action space are often immense in MARL

# Motivation

Despite the empirical successes, theoretical studies of policy optimization in MARL are very limited. Even for the cooperative setting where the agents share a common goal: maximizing the total reward function

In this paper, we aim to answer the following fundamental question:

*Can we design a provably convergent multi-agent policy optimization algorithm in the cooperative setting with function approximation?*

# Contributions

1. We answer the above question affirmatively.
2. We propose a multi-agent PPO algorithm in which the local policy of each agent is updated sequentially in a similar fashion as vanilla PPO algorithm (Schulman et al., 2017).
3. We adopt the log-linear function approximation for the policies. We prove that multi-agent PPO converges at a sublinear  $O\left(\frac{N}{1-\gamma} \sqrt{\frac{\log(|A|)}{K}}\right)$  rate up to some statistical errors incurred in evaluating/improving policies.
4. Moreover, we propose an off-policy variant of the multi-agent PPO algorithm and introduce pessimism into policy evaluation.

# Problem Setup

- Fully-cooperative Markov Games

- a tuple  $M = (N, \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ : A party of participants  $N$ , a set of states  $\mathcal{S}$ , a set of actions  $\mathcal{A}$ , a transition probability  $\mathcal{P}: \mathcal{S} \times \mathcal{A} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$ , a reward function  $r: \mathcal{S} \times \mathcal{A} \times \mathcal{A} \rightarrow [0, 1]$ , a discounted factor  $\gamma \in [0, 1)$ .
- define policies as probability distributions over action space:  $\pi \in \mathcal{S} \rightarrow \Delta(\mathcal{A})$ .

- Value function

$$V^\pi(s) = E_{a \sim \pi} \left[ \sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, b_t) \mid s_0 = s \right]$$

# Multi-agent Notations

- We write index  $k$  on superscript when we refer to the specific  $k$ -th agent. When bold symbols are used without any superscript (e.g.,  $\mathbf{a}$ ), they consider all agents. For simplicity, let  $(m:m')$  be shorthand for set:  $\{i|m \leq i \leq m', i \in N\}$ .
- Definition 3.1. Let  $P$  be a subset in  $N$ . The multi-agent action value function associated with agents in  $P$  is

$$Q_{\pi}^P(s, \mathbf{a}^P) = E_{\tilde{\mathbf{a}} \sim \tilde{\pi}}[Q_{\pi}(s, \mathbf{a}^P, \tilde{\mathbf{a}})]$$

here we use a tilde over symbols to refer to the complement agents, namely  $\tilde{\mathbf{a}} = \{a^i | i \notin P, i \in N\}$ .

# Advantage function and Bellman operators

- Let  $P, P' \subset N$  be two disjoint subsets of agents. The multiagent advantage function is defined below. Essentially, it accounts for the improvements of setting agents  $a^{P'}$  upon setting agents  $a^P$ , while all other agents follow  $\pi$

$$A_{\pi}^{P'}(s, \mathbf{a}^P, \mathbf{a}^{P'}) = Q_{\pi}^{P \cup P'}(s, \mathbf{a}^P, \mathbf{a}^{P'}) - Q_{\pi}^P(s, \mathbf{a}^P)$$

- For  $m \in N$  and any function  $f: S \times A^m \rightarrow R$  we define multi-agent Bellman operator  $T_{\pi}^{1:m}: R^{S \times A^m} \rightarrow R^{S \times A^m}$  as

$$T_{\pi}^{1:m} f(s, a^{1:m}) = E_{\tilde{a} \sim \tilde{\pi}} r(s, a^{1:m}, \tilde{a}) + \gamma E_{\tilde{a} \sim \tilde{\pi}, s' \sim P} f(s', \pi^{1:m})$$

where  $f(s', \pi^{1:m})$  is shorthand for  $E_{a' \sim \pi^{1:m}} f(s', a')$



# Multi-agent PPO for online setting

**Parametrization** For the  $m$ -th agent ( $m \in \mathcal{N}$ ), its conditional policy depends on all prior ordered agents  $\mathbf{a}^{1:m-1}$ . Given a coefficient vector  $\theta^m \in \Theta$ , where  $\Theta = \{\|\theta\| \leq R | \theta \in \mathbb{R}^d\}$  is a convex, norm-constrained set. The probability of choosing action  $a^m$  under state  $s$  is

$$\pi_{\theta^m}(a^m | s, \mathbf{a}^{1:m-1}) = \frac{\exp(\phi^\top(s, \mathbf{a}^{1:m-1}, a^m)\theta^m)}{\sum_{a^m \in \mathcal{A}} \exp(\phi^\top(s, \mathbf{a}^{1:m-1}, a^m)\theta^m)} \quad (2)$$

**Policy Improvement** At the  $k$ -th iteration, we define  $\hat{\pi}_{k+1}^m$  as the ideal update based on  $\hat{Q}_{\pi_{\theta_k}}^{1:m}$  (for agent  $m \in \mathcal{N}$ ), which is an estimator of  $Q_{\pi_{\theta_k}}^{1:m}$ . The ideal update is obtained via the following update

$$\hat{\pi}_{k+1}^m \leftarrow \arg \max_{\pi^m} \hat{F}(\pi^m) \quad (3)$$

$$\hat{F}(\pi^m) = \mathbb{E}_{\sigma_k} \left[ \langle \hat{Q}_{\pi_{\theta_k}}^{1:m}(s, \mathbf{a}^{1:m-1}, \cdot), \pi^m(\cdot | s, \mathbf{a}^{1:m-1}) \rangle - \beta_k KL \left( \pi^m(\cdot | s, \mathbf{a}^{1:m-1}) \| \pi_{\theta_k}^m(\cdot | s, \mathbf{a}^{1:m-1}) \right) \right]$$

where  $\theta_k^m$  is the parameter of the current conditional policy of the  $m$ -th agent. In above equation, the distribution is taken over  $(s, \mathbf{a}^{1:m-1}) \sim \nu_k \pi_{\theta_k}^{1:m-1}$ , we write  $\sigma_k$  for simplicity.

# Multi-agent PPO for online setting

- In practice, we approximate such policy within a certain parametrization class, which is the log-linear function class.

To approximate the ideal  $\hat{\pi}_{k+1}^m$  using a parameterized  $\pi_{\theta_{k+1}^m} \propto \exp\{\phi^\top \theta_{k+1}^m\}$ , we minimize the following mean-squared error (MSE) as a sub-problem

$$\theta_{k+1}^m \leftarrow \arg \min_{\theta^m \in \Theta} L(\theta^m) \quad (4)$$

where  $L(\theta^m)$  is defined as

$$L(\theta^m) = \mathbb{E}_{\sigma_k} \left( (\theta^m - \theta_k^m)^\top \phi(s, \mathbf{a}^{1:m-1}, a^m) - \frac{\hat{Q}_{\pi_{\theta_k}}^{1:m}(s, \mathbf{a}^{1:m-1}, a^m)}{\beta_k} \right)^2$$

- Note that this MSE can be minimized using the classic SGD updates.

# Multi-agent PPO for online setting

**Policy Evaluation** In this step, we aim to examine the quality of the attained policy. Thereby, a  $Q$ -function estimator is required. We make the following assumption.

**Assumption 4.3.** Assume we can access an estimator of  $Q$  function that returns  $\hat{Q}$ . The returned  $\hat{Q}$  satisfies the following condition for all  $m \in \mathcal{N}$  at the  $k$ -th iteration

$$\left[ \mathbb{E}_{\sigma_k} \left( \hat{Q}_{\pi_{\theta_k}}^{1:m}(s, \mathbf{a}^{1:m-1}, a^m) - Q_{\pi_{\theta_k}}^{1:m}(s, \mathbf{a}^{1:m-1}, a^m) \right)^2 \right]^{1/2} \leq \xi_k^m.$$

# Algorithm

---

**Algorithm 1** Multi-Agent PPO

---

**Input:** Markov game  $(\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ , penalty parameter  $\beta$ , stepsize  $\eta$  for sub-problem, number of SGD iterations  $T$ , number of iterations  $K$ .

**Output:** Uniformly sample  $k$  from  $0, 1, \dots, K-1$ , return  $\bar{\pi} = \pi_{\theta_k}$ .

- 1: Initialize  $\theta_0^m = 0$  for every  $m \in \mathcal{N}$ .
  - 2: **for**  $k = 0, 1, \dots, K-1$  **do**
  - 3:   Set parameter  $\beta_k \leftarrow \beta\sqrt{K}$
  - 4:   **for**  $m = 1, \dots, N$  **do**
  - 5:     Sample  $\{s_t, \mathbf{a}_t^{1:m-1}, a_t^m\}_{t=0}^{T-1}$  from  $\sigma_k = \nu_k \pi_{\theta_k}$ .
  - 6:     Obtain  $\hat{Q}_{\pi_{\theta_k}}^{1:m}(s, \mathbf{a}^{1:m-1}, a^m)$  for each sample.
  - 7:     Feed samples into Algorithm 3, obtain  $\theta_{k+1}^m$ .
  - 8:   **end for**
  - 9: **end for**
- 

---

**Algorithm 3** Policy Improvement Solver for MA-PPO

---

**Input:** MG  $(\mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ , iterations  $T$ , stepsize  $\eta$ , samples  $\{s_t, \mathbf{a}_t^{1:m-1}, a_t^m\}_{t=0}^{T-1}$ .

**Output:** Policy update  $\theta$ .

- 1: Initialize  $\theta_0 = 0$ .
  - 2: **for**  $t = 0, 1, \dots, T-1$  **do**
  - 3:   Let  $(s, \mathbf{a}^{1:m-1}, a) \leftarrow (s_t, \mathbf{a}_t^{1:m-1}, a_t^m)$ .
  - 4:    $\theta(t + \frac{1}{2}) \leftarrow \theta(t) - 2\eta \phi(s, \mathbf{a}^{1:m-1}, a) \left( (\theta(t) - \theta_k^m)^\top \phi(s, \mathbf{a}^{1:m-1}, a^m) - \beta_k^{-1} \hat{Q}_{\pi_k}^{1:m}(s, \mathbf{a}^{1:m-1}, a^m) \right)$ .
  - 5:    $\theta(t+1) \leftarrow \Pi_{\Theta} \theta(t + \frac{1}{2})$
  - 6: **end for**
  - 7: Calculate average:  $\bar{\theta} \leftarrow \frac{1}{T} \sum_{t=1}^T \theta_t$ .
-

# Theoretical results

- Theorem 1 (informal): For this setting, after  $K$  iterations, we have  $J(\pi^*) - J(\bar{\pi})$  upper bounded by

$$\mathcal{O} \left( \frac{B\sqrt{N}}{1-\gamma} \sqrt{\frac{N \log |\mathcal{A}| + \sum_{m=1}^N \sum_{k=0}^{K-1} (\Delta_k^m + \delta_k^m)}{K}} \right)$$

where  $\Delta_k^m = \sqrt{2}(\phi_k^m + \phi_k^{m-1}) \cdot \left( \epsilon_k^m + \frac{\xi_k^m}{\beta_k} \right)$  and  $\delta_k^m = 2\phi_k^{m-1}\epsilon_k^m$ . Here  $\epsilon_k^m$  is the statistical error of a PPO iteration: for agent  $m \in \mathcal{N}$ ,

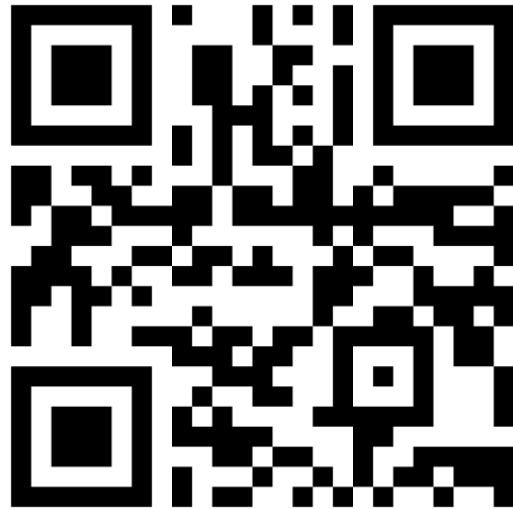
$$\mathbb{E}_{\sigma_k} \left( (\theta_{k+1}^m - \theta_k^m)^\top \phi - \beta_k^{-1} \hat{Q}_{\pi_{\theta_k}}^{1:m} \right)^2 \leq (\epsilon_k^m)^2$$

# Pessimistic MA-PPO with Linear Function Approximation

- We perform pessimistic policy evaluation via regularization to reduce such overestimation aligning with experimental works.
- Theorem 1 (informal): For this setting, after  $K$  iterations, we have  $J(\pi^*) - J(\bar{\pi})$  upper bounded by

$$\mathcal{O} \left( \frac{N}{(1-\gamma)^2} \sqrt{\frac{\log |\mathcal{A}|}{K}} + \frac{C_{\mu}^{d\pi^*}}{(1-\gamma)^2} \sqrt[3]{\frac{d \log \frac{nLR}{\delta}}{n}} \right)$$

# Thank you and some more information



**paper**



Yulai Zhao  
PhD student @ Princeton