RACE: Improve Multi-Agent Reinforcement Learning with Representation Asymmetry and Collaborative Evolution

Pengyi Li, Jianye Hao[†], Hongyao Tang, Yan Zheng, Xian Fu

[†]Corresponding authors.

Any Question, Concat: lipengyi@tju.edu.cn

Code: https://github.com/yeshenpy/RACE

Necessary Background and Problem Statement

Multi-Agent Reinforcement Learning

- In MARL, individual agents interact with the environment and with each other, collecting samples and receiving reward signals to evaluate their decisions. By leveraging value function approximation, MARL optimizes policies through gradient updates. However, MARL often faces the following challenges:
  - (**Low-quality reward signals**) The reward signals are often of low quality (e.g., deceptive, sparse, delayed, and team-level), making it challenging to obtain accurate value estimates.
  - (**Low exploration for collaboration**) The gradient-based optimization approach may struggle to efficiently explore the multi-agent policy space and facilitate collaboration.
  - (**Non-stationarity**) As the agents learn concurrently and continuously influence each other, breaking the Markov assumption on which most single agent RL algorithms are based.
  - (**Partial observations**) When agents have partial observations of their environment, making policy optimization even more challenging.

Evolutionary Algorithm

- Evolutionary Algorithm (EA) simulates the natural process of genetic evolution and does not rely on gradient information for policy optimization, which **has been demonstrated to be competitive** with RL. Unlike RL which typically maintains only one policy, EA **maintains a population of individuals** and performs iterative evolution according to **policy fitness**. The fitness is typically defined as the average Monte Carlo (MC) return over some episodes

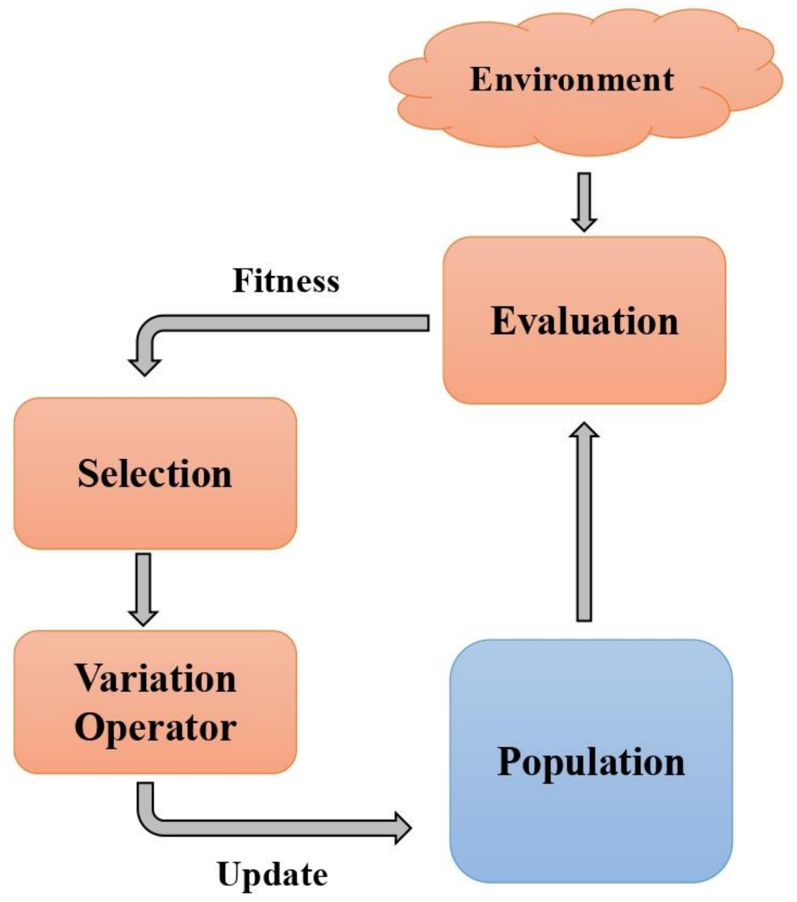

Figure 1: Evolutionary Algorithm optimization process.

- Evolutionary Algorithm (EA) possesses several key strengths:
  - EA does not require RL value function approximation and directly evolves individuals within the population according to fitness, i.e., the cumulative rewards. This makes EA **more robust to reward signals.**
  - EA is not reliant on the Markov property in problem formulation and evolves policies from the team perspective, thereby **circumventing the non-stationarity problem encountered in MARL.**
  - EA has **strong exploration ability, good robustness, and stable convergence.**

Motivation

- (**Complementarity**) EA offers numerous strengths that can complement the weaknesses of MARL.
- (**Research Gap**) However, the efficient integration of both approaches in complex multiagent collaborative tasks has not been thoroughly investigated.
- To this end, we propose a novel framework called **R**epresentation **A**symmetry and **C**ollaborative **E**volution (**RACE**) which combines EA with MARL to achieve efficient collaboration.

Method

Representation-Asymmetry Team Construction

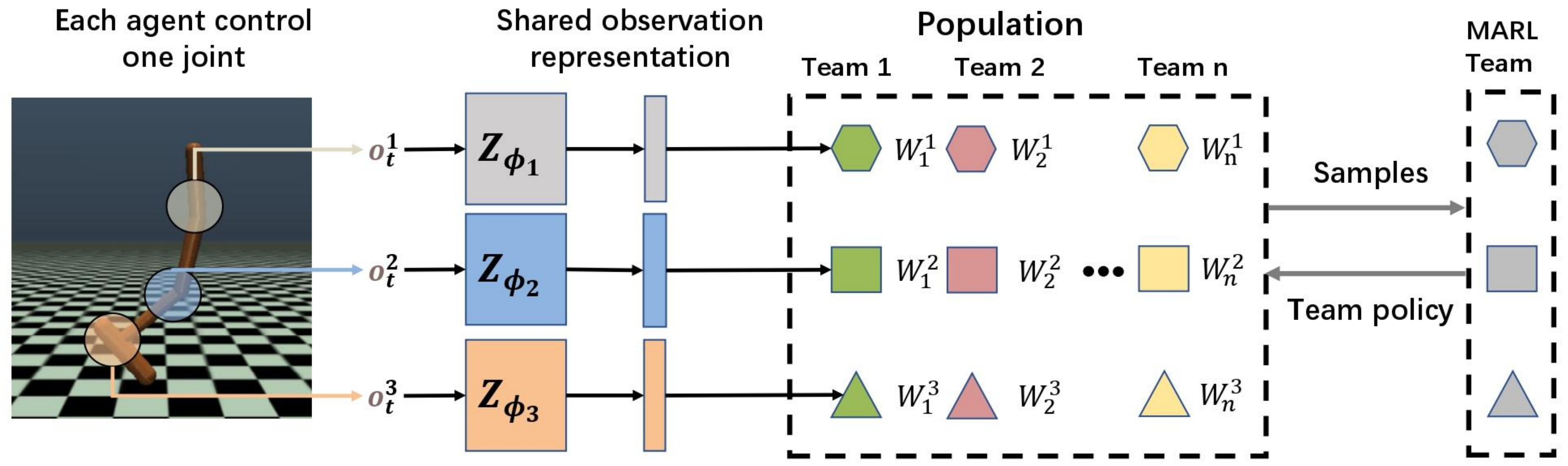

Figure 2: The conceptual illustration of Representation-Asymmetry Team Construction on 3-Agent Hopper task.

- RACE introduces a population of teams. Typically, each team maintains **separate policies** for decision-making and optimization. However, this independent policy construction limits knowledge sharing across teams and makes exploration in large policy spaces inefficient.
- We propose **Representation-Asymmetry Team Construction** to enable efficient knowledge sharing and policy exploration. Specifically, the policies that control the same agent in different teams are composed of a shared nonlinear observation representation encoder. Formally, we summarize the construction of individual, team, and population in RACE below:

Policy i of team j: $\boldsymbol{\pi_j^i(o_t^i) = \mathrm{act}\left(Z_{\phi_i}(o_t^i)^\top W^i_{j,[1:d]} + W^i_{j,[d+1]}\right)}$

Team policy of team j: $\boldsymbol{\pi_j(s_t) = \{\pi_j^1(o_t^1), \cdots, \pi_j^N(o_t^N)\}}$

Construction of team j: $\boldsymbol{W_j = \{W_j^1, W_j^2, \cdots, W_j^N\}}$

Team Population: $\mathbb{P} = \{\boldsymbol{W_1, W_2, \cdots, W_n}\}$

Shared Representation: Value Function Maximization

- The team architecture poses two demands: 1) The shared observation representation encoder Z_{Φ_i} should provide **useful knowledge about collaboration and tasks**. 2) The knowledge is required to **be beneficial to all teams**, not just one particular team.
- Thus we propose Value Function Maximization to extract the information:

$$\mathcal{L}_{\phi_i}^{\mathrm{VFM}} = -\mathbb{E}_{\mathcal{D},\mathbb{P}}\left[Q_\psi(s, \pi_{marl}(s)) + \mathbb{Q}_\theta(s, \pi_j(s), W_j)\right]$$

- Except for $Q(s,u)$, We learn a centralized Policy-extended Value Function Approximator (PeVFA) to estimate the value for the policy representations in $\mathbb{P}$:

$$\mathcal{L}_\theta = \mathbb{E}_{\mathcal{D}}\left[\left(r + \gamma\mathbb{Q}_{\theta'}(s', \pi_j(s'), W_j) - \mathbb{Q}_\theta(s, u, W_j)\right)^2\right]$$

Shared Representation: Value-Aware Mutual Information Maximization

- Only using the value information is inadequate since most tasks in Multi-Agent systems are **partially observable,** and thus **non-stationarity** throughout the execution and learning phases is exacerbated.
- Thus we propose to maximize the mutual information (MI) between the shared observation representations $z_i = Z_{\phi_i}(o^i)$ and global state s to make z_i reflect global information thus alleviating the problem of partial observations.

$$I(z_i; s) \geq \sup_{\omega\in\Omega} \underbrace{\mathbb{E}_{\mathbb{P}_{\mathcal{S}Z_i}}\left[-sp\left(-T_\omega(s_t, z_{i,t})\right)\right] - \mathbb{E}_{\mathbb{P}_{\mathcal{S}}\otimes\mathbb{P}_{z_i}}\left[sp\left(T_\omega(s_t, z_{i,k})\right)\right]}_{I_{lb}(z_i;s)}$$

- However, **maximizing MI with inferior states may induce a negative influence on shared observation representations** from the global information of poor collaboration, leading to suboptimality (PMIC, Li et al., 2022).
- To this end, we propose a novel Value-Aware MI Maximization to extract the superior global information into z_i:

$$\mathcal{L}_{\phi_i}^{\mathrm{VMM}} = -\mathbb{E}_D\left[\frac{V_\zeta(s_t) - \min_{s_j\sim D}\left(V_\zeta(s_j)\right)}{\max_{s_j\sim D}\left(V_\zeta(s_j)\right) - \min_{s_j\sim D}\left(V_\zeta(s_j)\right)} I_t(z_i^t, s_t)\right]$$

where $V_\zeta(s_t)$ is a value function and optimized by $\mathcal{L}_\zeta = \mathbb{E}_{s\sim D}[(V(s) - F(s))^2]. F(s) = r + \max\left(Q_{\psi'}\left(s', \pi_{\mathrm{marl}}(s')\right), \mathbb{Q}_{\theta'}(s', \pi_j(s'), W_j)\right)$

- Finally, the loss function of $Z_{\phi_i}(o^i)$ is defined as:

$$\mathcal{L}_{\phi_i} = \mathcal{L}_{\phi_i}^{\mathrm{VFM}} + \beta\mathcal{L}_{\phi_i}^{\mathrm{VMM}}$$

Improving MARL with Collaborative Evolution

- Based on the shared observation representation encoder Z_{Φ_i} the policies of different teams controlling the same agents optimize their policy representations more efficiently Since 1) Policy optimization occurs **in a linear policy space**. 2) The optimization utilizes **diverse samples collected by all teams**.

$$\mathcal{L}_{\mathrm{MARL}}(\mathrm{W}_{\mathrm{marl}}) = -\mathbb{E}_{s\sim\mathcal{D}}[\mathrm{Q}_{\psi_i}(s, \pi_{\mathrm{marl}}(s))]$$

- To achieve more efficient evolution: we design the agent-level crossover and mutation for both team and individual exploration.

$$W_i', W_j' = \left(\left(W_i - W_i^{d_i}\right) \cup W_j^{d_i}, \left(W_j - W_j^{d_j}\right) \cup W_i^{d_j}\right) = \mathrm{crossover}(W_i, W_j),$$

$$W_j' = \left(W_j - W_j^{d_j}\right) \cup P\left(W_j^{d_j}\right) = \mathrm{Mutation}(W_j)$$

Experiments (Both Continuous and Discrete Tasks)

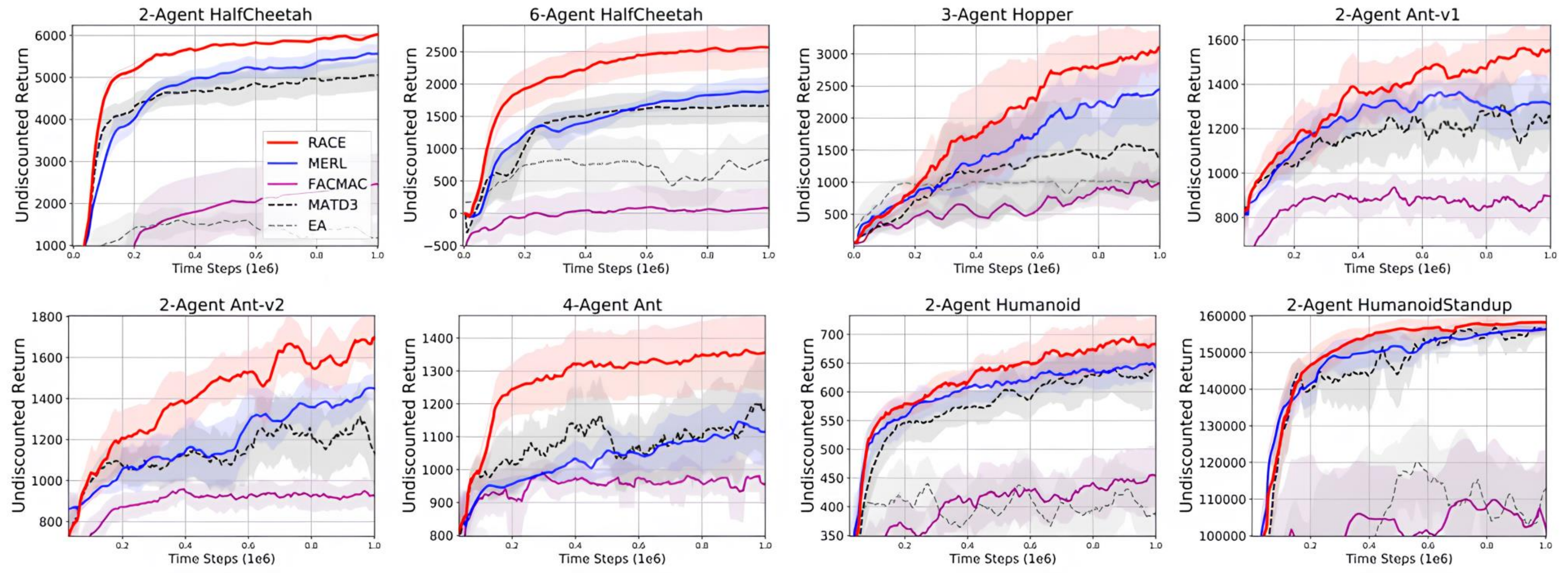

Figure 3: Performance comparison between RACE and baselines in Multi-Agent MuJoCo (All in MATD3 version).

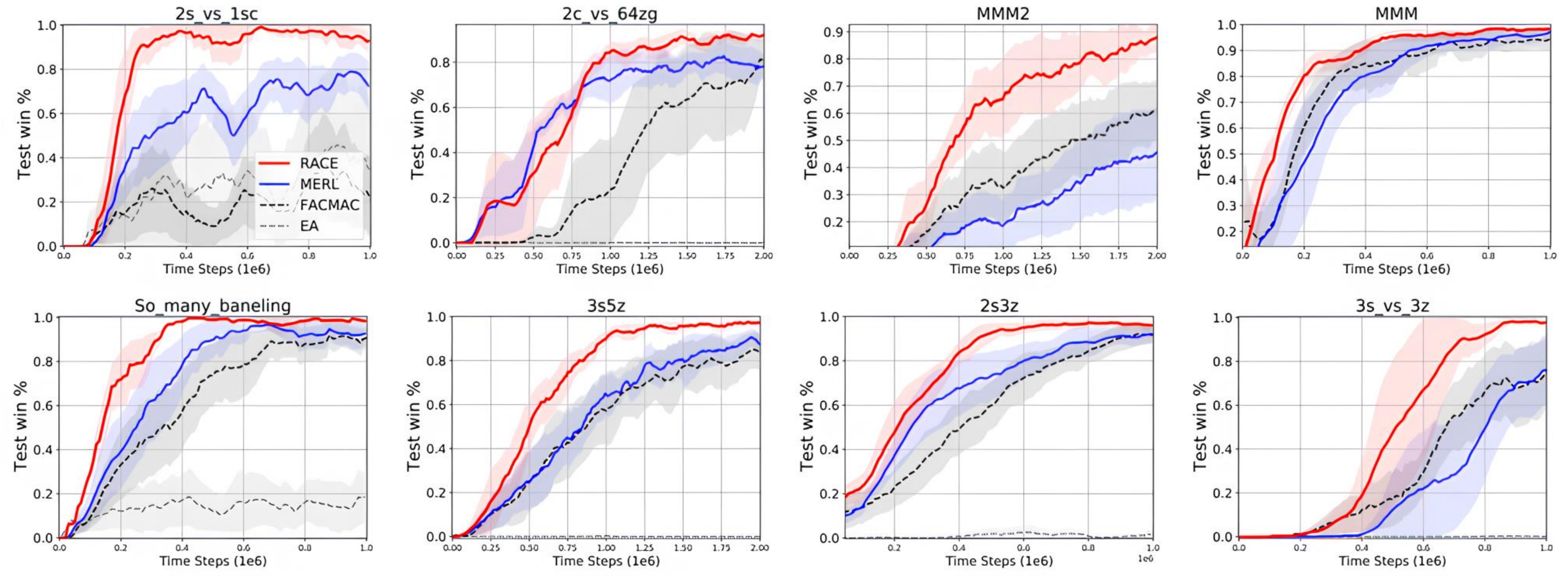

Figure 4: Test win rate comparison between RACE and baselines in SMAC (All in FACMAC version).

- RACE can significantly improve the basic MARL methods and outperform other baselines. Notably, our work demonstrates, **for the first time,** that EA has the capability to significantly enhance MARL performance **in complex collaborative tasks**.