ICML | 2022

北京大学智能仿生设计实验室
Intelligent Biomimetic Design Lab

Difference Advantage Estimation for Multi-Agent Policy Gradients

Yueheng Li[1], Guangming Xie[1,2], Zongqing Lu[2,3]

[1]College of Engineering, [2]Institute of Artificial Intelligence, [3]School of Computer Science, Peking University

Presented by Y. Li

June 20, 2022

Corresponding authors: G. Xie xiegming@pku.edu.cn, Z. Lu zongqing.lu@pku.edu.cn

Outlines

- Background
- Multi-Agent Credit Assignment
- Difference Advantage Estimation
- Experimental Results

Cooperative Multi-Agent Reinforcement Learning (MARL)

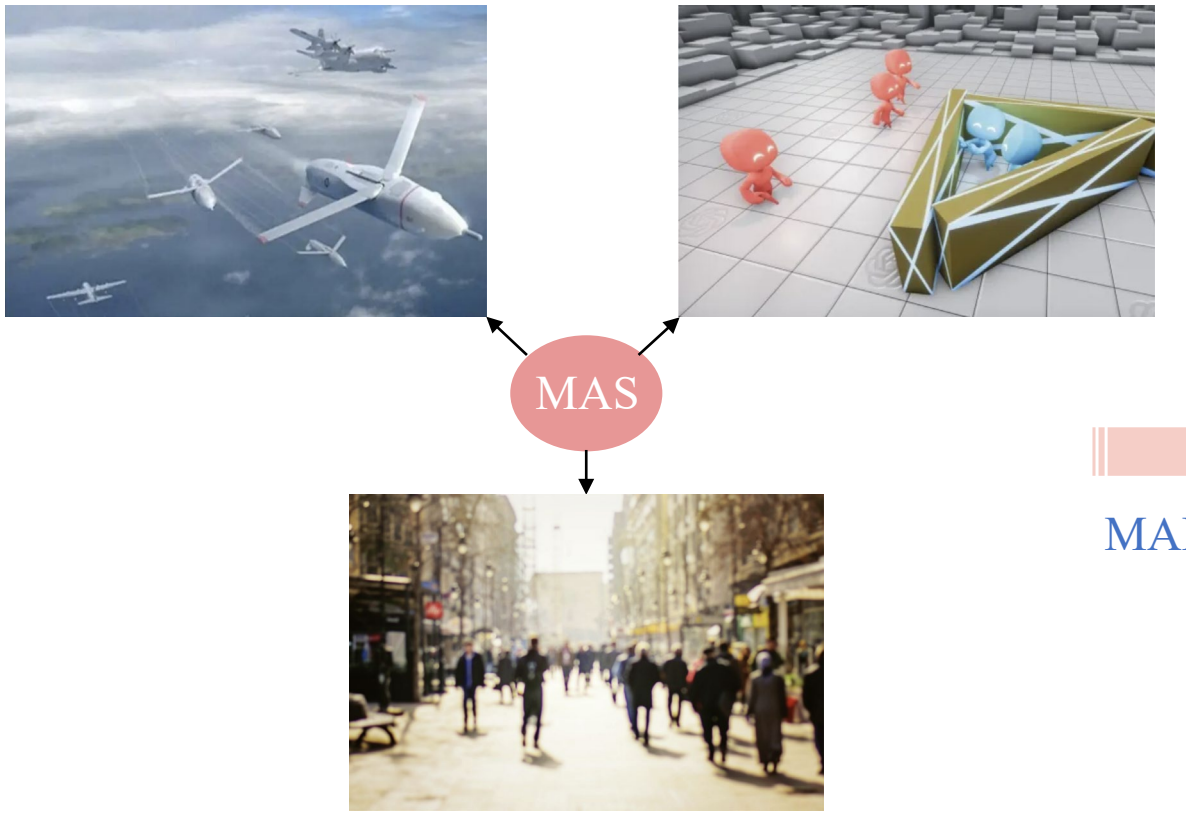


Fig. 1 Cooperative Multi-Agent System (MAS) (images from public domain)

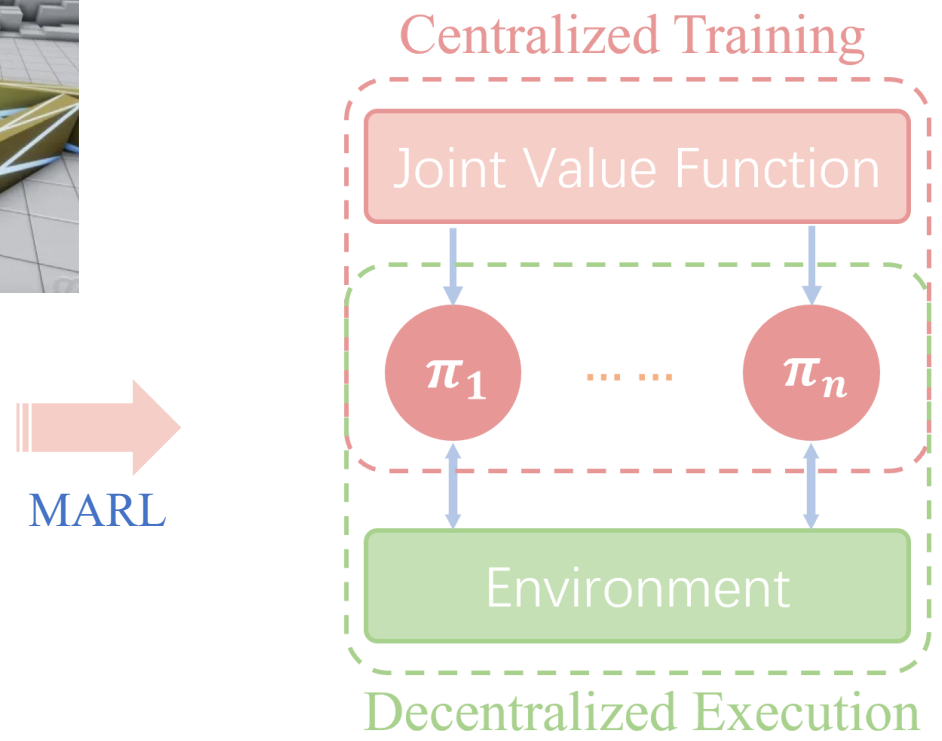
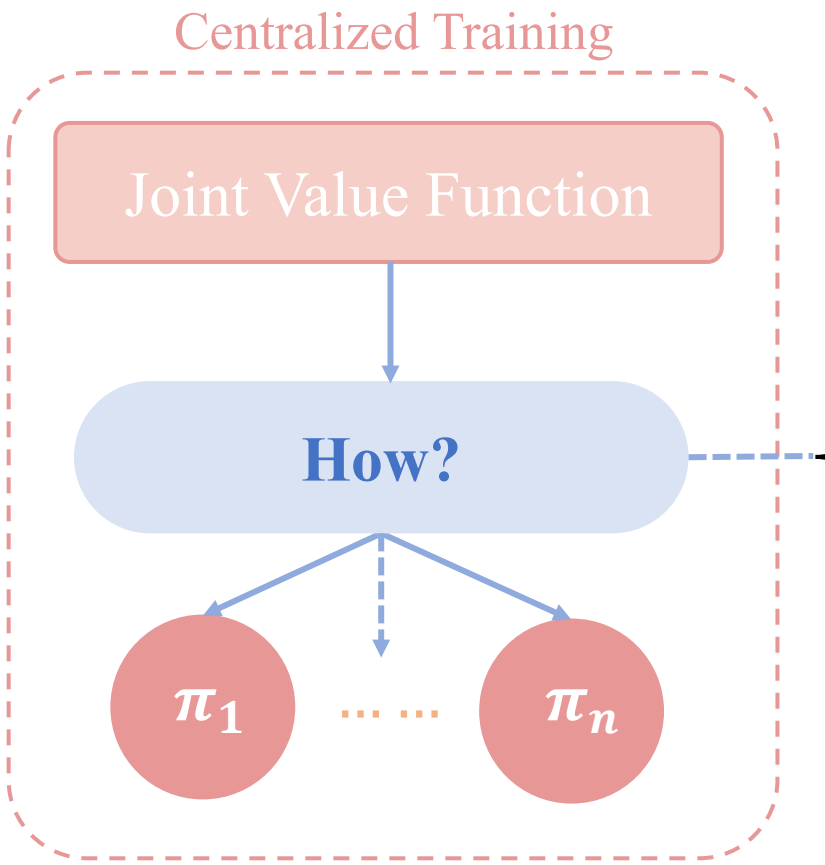


Fig. 2 Centralized Training with Decentralized Execution (CTDE)

Centralized Training with Decentralized Execution (CTDE)



value-based

$Q_{jt} \rightarrow [Q_i]_{i=1}^n$

value function decomposition

implicit credit assignment

policy-based

$V_{jt} \rightarrow [\pi_i]_{i=1}^n$

$\sum_{i=1}^n \nabla log \pi_i(a_i|s)(r + \gamma V(s') - V(s))$

How to assign credit?

Counterfactual Reward/Q-function

Multi-Agent Credit Assignment

- Difference rewards: $r(s,a) - E_{a_i \sim \pi_i}[r(s, a_i, a_{-i})]$ ← reward function
- Counterfactual baseline: $Q(s,a) - E_{a_i \sim \pi_i}[Q(s, a_i, a_{-i})]$ ← Q function

- hard to learn joint Q function directly
- highly biased gradient may not work well (e.g. COMA)

Counterfactual Reward/Q-function

- Potential-based difference rewards

$$r_i(s,a) = r(s,a) + \gamma E_{a_i' \sim \pi_i}[r(s', a_i', a_{-i}')] - E_{a_i \sim \pi_i}[r(s, a_i, a_{-i})]$$

✓ Policy Invariance

$$\tilde{Q}_i(s,a) = E_\pi \left[\sum_{t=0}^{\infty} \gamma^t \left(r_t + \gamma \phi_i(s_{t+1}, a_{t+1}^{-i}) - \phi_i(s_t, a_t^{-i}) \right) \right]$$

$$= Q(s,a) - \phi_i(s, a_{-i})$$

$$E_\pi[\sum_i \nabla_\theta log \pi_i(a_i|s) \phi_i(s, a_{-i})] = 0 \rightarrow$$ ✓ Unbiased

N-Step Reward Shaping

- If the potential-based difference rewards are helpful, can we apply it multiple times?

1-step

$$r_{i,t}^{(0)} = r_t$$

$$r_{i,t}^{(1)} = r_t + \gamma E_{a_i}[r_{t+1}] - E_{a_i}[r_t]$$

2-step

$$r_{i,t}^{(2)} = r_{i,t}^{(1)} + \gamma E_{a_i}\left[r_{i,t+1}^{(1)}\right] - E_{a_i}\left[r_{i,t}^{(1)}\right]$$

$$= r_t + \gamma^2 E_{a_i}[r_{t+2}] - E_{a_i}[r_t]$$

$$\vdots$$

∞-step

$$r_{i,t}^{(\infty)} = r_t - E_{a_i}[r_t]$$

$$G_{i,t}^{(0)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l}$$

$$G_{i,t}^{(1)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l} - E_{a_i}[r_t]$$

$$G_{i,t}^{(2)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l} - E_{a_i}[r_t + \gamma r_{t+1}]$$

$$\vdots$$

$$G_{i,t}^{(\infty)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l} - \sum_{l=0}^{\infty} \gamma^l E_{a_i}[r_{t+l}]$$

Difference Advantage Estimation

Bias and Credit Assignment Trade-Off

- k-step ($k \geq 2$) reward shaping bias the policy gradient

$$V^{\pi_\theta}(\mu') \geq V^*(\mu') - \frac{2}{1-\gamma}\left(\lambda \left\| \frac{d_{\mu'}^{\pi^*}}{\mu} \right\|_\infty + \sum_{t=1}^{k-1} \gamma^t\right)$$

- Control the bias and credit assignment by parameter $\beta \in [0,1]$

$$(1-\beta)\beta^0 \quad G_{i,t}^{(0)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l}$$

$$(1-\beta)\beta^1 \quad G_{i,t}^{(1)} = \sum_{l=0}^{\infty} \gamma^l r_{t+l} - E_{a_i}[r_t]$$

$$\vdots \qquad\qquad \vdots$$

$$G_{i,t}^{(\infty)} = \sum_{l=0}^{\infty} \gamma^l (r_{t+l} - E_{a_i}[r_{t+l}])$$

weighted sum

like TD(λ)

$$G_{i,t}^{(\beta)} = \sum_{l=0}^{\infty} \gamma^l (r_{t+l} - \beta^{l+1} E_{a_i}[r_{t+l}])$$

$$V^{\pi_\theta}(\mu') \geq V^*(\mu') - \frac{2}{1-\gamma}\left(\lambda \left\| \frac{d_{\mu'}^{\pi^*}}{\mu} \right\|_\infty + \frac{\beta\gamma}{1-\beta\gamma}\right)$$

Difference Advantage Estimator (DAE)

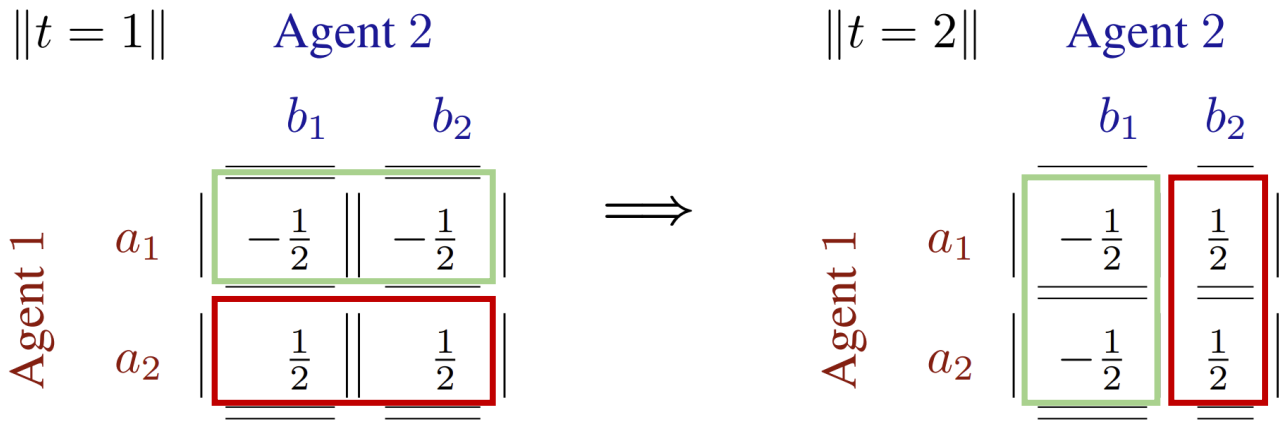
- Integrate with GAE(λ)

$$A_{i,t}^{DAE} = \sum_{l=0}^{\infty} (\gamma\lambda)^l (r_{t+l} - \beta^{l+1} E_{a_i}[r_{t+l}] + \gamma V_{t+l+1} - V_{t+l})$$

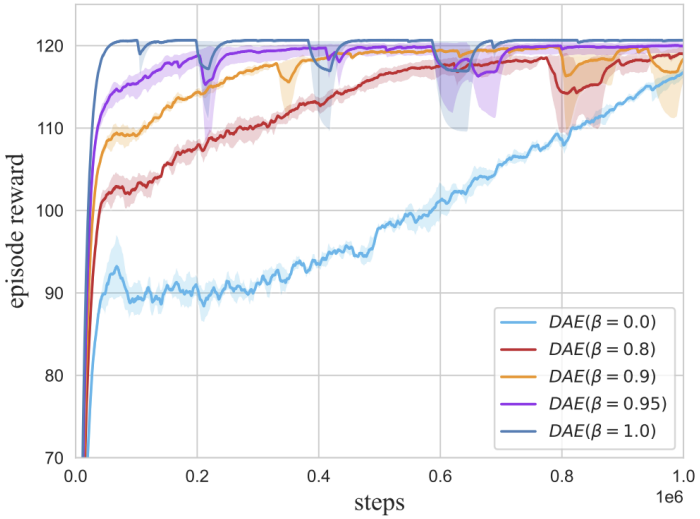
  - ✓ Bias and credit assignment trade-off

Experimental Results

Matrix Game



(a) 2-step matrix game

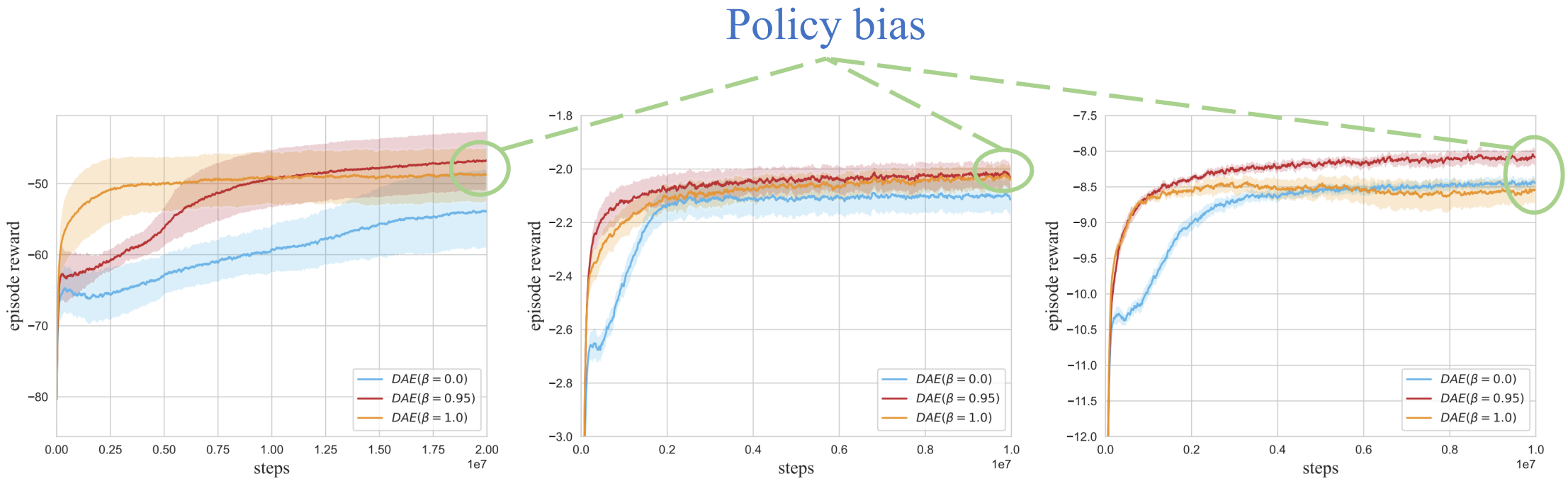


(b) learning curves of DAE with different β

Figure 1. (a) Payoff matrices for 2-step matrix game at $t = 1$ (left) and $t = 2$ (left), where all indices start from 1. (b) Learning curves of DAE with different β on the 16-step matrix game, where larger β gets better performance.

✓ Credit assignment is helpful

Multi-Agent Particle Environment

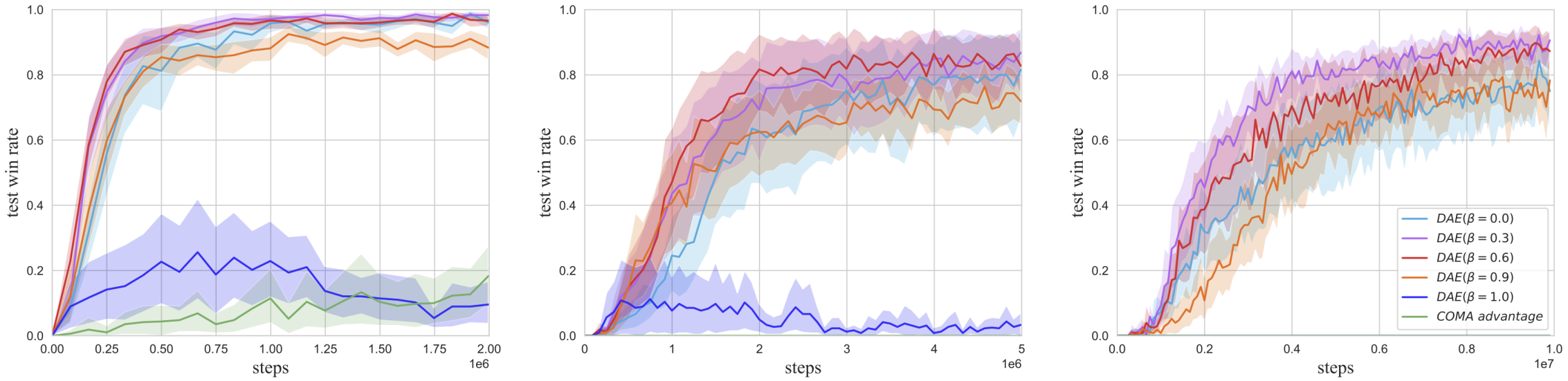


(a) cooperative navigation (b) line control (c) formation control

Figure 2. Learning curves for three tasks in MPE using DAE with varying values of β. Each curve is averaged over the settings $N = 3, 4, 5$, and separate results are available in Appendix D.

Experimental Results

StarCraft Multi-Agent Challenge (SMAC)



(a) easy (b) hard (c) super hard

Figure 3. Learning curves in terms of win rates of DAE with different β and COMA advantage in 'easy', 'hard' and 'super hard', each averaged over three corresponding maps. Separate results in each map are available in Appendix D.

✓ Credit assignment and bias trade-off result in good performance

Thanks for Your Attention!

Keep in touch:

Yueheng Li: liyueheng@pku.edu.cn
Guangming Xie: xiegming@pku.edu.cn
Zongqing Lu: zongqing.lu@pku.edu.cn