Letter

Reinforcement Learning-Based MAS Interception in Antagonistic Environments

Siqing Sun ⓘ, Defu Cai ⓘ, Hai-Tao Zhang ⓘ,
Senior Member, IEEE, and Ning Xing ⓘ

Dear Editor,

As a promising multi-agent systems (MASs) operation, autonomous interception has attracted more and more attentions in these years, where defenders prevent intruders from reaching destinations. So far, most of the relevant methods are applied in ideal environments without agent damages. As a remedy, this letter proposes a more realistic interception method for MASs suffered by damages, where the defenders are fewer than the intruders. Firstly, a multi-agent interception frame (MAIF) is proposed, enabling the defenders to take actions and interact with the environments. To address non-stationarity issue induced by MAIF, a multi-agent reinforcement learning-based interception method (MAIM) is developed by sophisticatedly designing a reward function. Sufficient conditions are derived to guarantee the convergence of MAIM. Finally, numerical simulations are conducted to verify the effectiveness of the proposed method.

These years have witnessed the tremendous development of the research on autonomous interception of MASs [1], [2]. Lowe *et al.* [3] establish a multiple particle environment, where predators seek to intercept preys from foraging food. Zhang *et al.* [4] investigate a pursuit-evasion game for multi-quadcopters to intercept a random drone. Yu *et al.* [5] address a typical dynamic combat scenario for two hostile drone swarms intercepting each other from destroying their military bases. To achieve more agile collective interception, some scholars seek assistance from reinforcement learning [6], [7].

The main challenge lies in designing suitable a reward function, which has a significant influence on the interception performance [8]. In this regard, some interception studies define shape rewards by distances [9]–[11], so as to find an action to maximize the distances between the intruders and the defensive area However, oversimplification of aforementioned studies have hindered their further applications. For example, intruders are less than defenders, agent damages are not considered, etc. This motivate us to develop a more realistic collective interception scheme to address the challenging antagonistic interception problem for MASs, where the shape reward may change frequently upon emergence of events such as agent damages. Besides, reward function is usually non-convex in such a scenario [12], making the learning procedure apt to be trapped in local optima.

To address the dilemma, we propose a MAIM. Firstly, shape rewards are sophisticatedly designed and assigned to the agents with attention weights. In this way, each defender does not have to consider too many intruders simultaneously, reducing its decision space dimension. Additionally, to ensure learning process convergence

upon the emergence of intermediate events, an event reward is designed to revise the reward history sequence, instead of directly adding a large value to the reward of a moment.

In brief, the contribution of this work is two-fold: 1) Developing a reinforcement learning-based MAIM with a suitable reward function, which enables defenders to intercept intruders in antagonistic environments; 2) Deriving sufficient convergence conditions for MAS governed by the proposed MAIM.

Notations: Throughout the letter, $\mathbb{N}^+$ denotes the positive integer set. $\|*\|$ is the 2-norm of a vector $*$, and $*^T$ represents the transpose of a matrix $*$. The symbol $*_{i,j}^t$ denotes the relative value of $*$ between agents i and j at a temporal instant t. The symbol k is an index of the training iteration.

Preliminaries: As shown in Fig. 1, a MAIF is established, which is composed of a destination c with center coordinates ρ_c and radius d_c, and two opponent agent groups, i.e., the defenders $\mathcal{V}_d = \{v_1, v_2, \ldots, v_m\}$, and the intruders $\mathcal{V}_r = \{v_{m+1}, v_{m+2}, \ldots, v_{m+n}\}$ ($m < n$, $m, n \in \mathbb{N}^+$). where the defenders $\mathcal{V}_d$ aim to prevent the intruders $\mathcal{V}_r$ from reaching the defense region c. All the agents $\mathcal{V}_d \cup \mathcal{V}_r$ have the following kinetics:

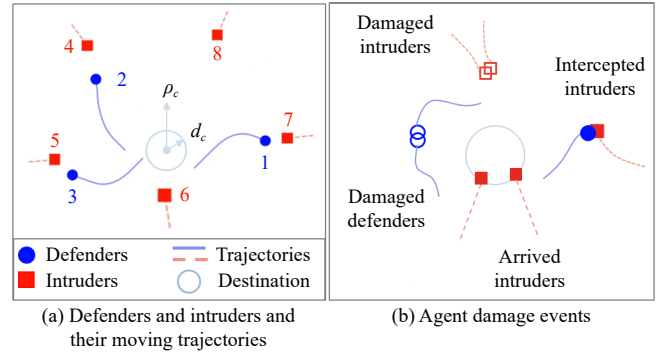$$\dot{\rho}_i^t = v_i^t, \quad \dot{v}_i^t = a_i^t. \tag{1}$$

Fig. 1. Illustration of the present MAIF.

The symbols ρ_i^t, v_i^t, and a_i^t are coordinate, velocity, and acceleration of agent i in x- and y- directions at moment t, respectively. $a_i^{\max}$, $v_i^{\max}$ are the maximum accelerations and velocities of agent i, respectively. We set $a_i^{\max} = \frac{n}{m} a_j^{\max}$, $v_i^{\max} = \frac{n}{m} v_j^{\max}$, $i \in \mathcal{V}_d$, $j \in \mathcal{V}_r$. Since the defenders are stronger than the intruders, intruder i is regarded as damaged once $\rho_i^t = \rho_j^t$, $j \in \mathcal{V}_d \cup \mathcal{V}_r$, $j \neq i$, whereas a defender is regarded as damaged only when $\rho_i^t = \rho_j^t$, $j \in \mathcal{V}_d$, $j \neq i$. When an agent is damaged or arrived (Arr$_j$ = 1, if $\|\rho_i - \rho_c\| < d_c$), it is regarded as done and then removed from MAIF. The flag Run$_i$ = 0 if agent i is done, and Run$_i$ = 1 otherwise. An interception episode ω begins with $t = 0$, and completes when all the intruders $\mathcal{V}_r$ are done or a given overall running period T is used up. The objective of the defenders $\mathcal{V}_d$ is to win the confrontation. Now, it is necessary to provide the following definition first.

Definition 1 (Winning or losing condition of defenders): Defenders $\mathcal{V}_d$ are considered to win (Win = 1), if and only if all the defenders $\mathcal{V}_d$ are alive (Run$_i$ = 1, $\forall i \in \mathcal{V}_d$), and all the intruders $\mathcal{V}_r$ are intercepted before arriving at the destination (Run$_j$ = 0, Arr$_j$ = 0, $\forall j \in \mathcal{V}_r$). Otherwise, defenders $\mathcal{V}_d$ lose (Win = 0).

For the intruders $\mathcal{V}_r$, a fixed strategy is designed, including destination attraction and collision avoidance. For defenders, the interception process is approximated as a multi-agent extension of Markov chain decision processes. As shown in Fig. 2, defender i generate an action a_i^t by its policy π_i and the state s_i^t, i.e.,

$$a_i^t = \pi_i(s_i^t). \tag{2}$$

Thereby, defender i gets to next state s_i^{t+1} with a reward r_i^t for the next interaction with MAIF. A series of state transition quintuples $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$ of defender i are stored into an experience replay
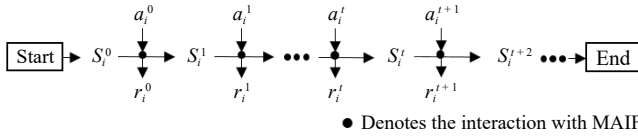
Fig. 2. The online Markov chain decision procedure in one interception episode of the present MAIF.

buffer D. The goal of each defender becomes optimizing its policy $\pi_i(s_i^t)$ for generating an action a_i^t with the maximized long-term discounted cumulative reward R_i^t, i.e.,

$$R_i^t = \sum_{\tau \in [0, T-t]} \gamma^\tau r_i^{t+\tau+1} \tag{3}$$

where $\gamma (\in [0,1])$ denotes a discount factor. To obtain the optimum policy π_i^*, the index is expressed as

$$\pi_i^* = \underset{a_i \in \mathcal{A}_i}{\arg\max} \, E_{\pi_i}(R_i^t | s_i). \tag{4}$$

where $E_{\pi_i}(R_i^t | s_i)$ is the expectation of R_i^t with the state s_i, and $\mathcal{A}_i$ is the action space. Since the cumulative reward R_i^t is hard to solve, a critic neural network $Q_i^\pi(s,a)$ parameterized by ε_i is trained to approximate R_i^t by an value $\widehat{Q}_i$. Meanwhile, an actor network $\pi_{\theta i}(s_i)$ parameterized by θ_i is adopted to approach the policy π_i for autonomous decision. Now, we are ready to develop the main technical problem for this letter.

Problem 1: For each defender governed by (1), how to obtain the optimal action policy $\pi_{\theta i}^*$ by means of actor-critic algorithm to maximum the index $E_{\pi_i}(Q_i^t | s_i)$, i.e.,

$$\pi_{\theta i}^* = \arg \max_{a_i = \pi_{\theta i}(s_i)} E_{\pi_{\theta i}}(\widehat{Q}_i | s_i).$$

Design a critic network parameter training law $\varepsilon_i = f_1(\widehat{Q}_i, R_i)$ such that

$$\lim_{k \to \infty} \varepsilon_i^k - \varepsilon_i^* = 0$$

where the optimum ε_i^* satisfies $\varepsilon_i^* = \arg\min_{\epsilon_i} E[(\widehat{Q}_i - R_i)^2]$.

MAIM: To solve Problem 1, the policy gradient $\nabla_{\theta_i} J(\theta_i)$ is derived by $\nabla_{\theta_i} J(\theta_i) = E[\nabla_{\theta_i} \pi_{\theta i}(a_i | s_i) \nabla_{\pi_{\theta i}} Q_i^\pi(s, a_{1,\dots,N}|_{a_i = \pi_i(s_i)})]$. To approximate R_i^t, the goal of the critic network Q_i^π is to minimize the sum of square loss $L(\varepsilon_i) = \min_{\epsilon_i} E[(\widehat{Q}_i - R_i)^2]$. Thus, the gradient of critic networks $\nabla_{\varepsilon_i} L(\varepsilon_i)$ is calculated by $\nabla_{\varepsilon_i} L(\varepsilon_i) = E[(Q_i^\pi(s, a_{1,\dots,Ni}) - y)\nabla_{\varepsilon_i} Q_i^\pi(s, a_{1,\dots,Ni})]$, where $y = r_i^t + \gamma Q_i^\pi(s, a_{1,\dots,N})$ is calculated by the temporal difference [3]. Sampling transitions $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$ from the replay buffer D, the network parameters θ_i, ε_i are iteratively updated by $\theta_i^{k+1} = \theta_i^k + \alpha_c \nabla_{\theta_i} J(\theta_i)$, $\varepsilon_i^{k+1} = \varepsilon_i^k - \alpha_c \nabla L(\varepsilon_i)$. Finally, the optimal policies π_θ^* are obtained.

Still, it is critical to design the action space $\mathcal{A}_i$ and the state s_i^t. A continuous acceleration vector is considered as the action space $a_i \in \mathcal{A}_i$. The new state s_i^t can be written as

$$s_i^t = [p_i^{t-1}, p_i^t] \tag{5}$$

where the property p_i^t includes 1) the information about the given defender itself $[\rho_i^t, v_i^t, \text{Run}_i]$, and 2) the relative states with other agents $[\rho_{i,j}^t, v_{i,j}^t, \text{Run}_{i,j}^t], j \in \mathcal{V}_d \cup \mathcal{V}_r, j \neq i$.

The main challenge lies in how to design a suitable reward function $r_i^t = r_{\text{shape}_i}^t + r_{\text{event}_i}^t$ in such a nonstationary MAIF, to guide the policy optimization proposed in Problem 1, where $r_{\text{shape}_i}^t$, $r_{\text{event}_i}^t$ denote the shape and the event rewards, respectively. Shape reward $r_{\text{shape}_i}^t$ is a distance-based function, presented in

$$r_{\text{shape}_i}^t = -k_\alpha \left\| \rho_{i,\tilde{i}}^t \right\| + k_\alpha \left\| \rho_{\tilde{i},c}^t \right\| + \sum_{j \in \mathcal{V}_r} \left\| \rho_{j,c}^t \right\| / n$$
$$- \sum_{j \in \mathbf{nei}_i^t} (1 - \left\| \rho_{i,j}^t \right\| / d)^2 \tag{6}$$

where $\tilde{i}$ is the nearest intruder of defender i, and $\tilde{i}$ is related to the time. $\tilde{i}$ is used to improve the attention weight k_α, which avoids defenders $\mathcal{V}_d$ constantly swinging among different intruders $\mathcal{V}_r$. $\mathbf{nei}_i^t$ is the neighbor set of defender i, i.e., $\mathbf{nei}_i^t = \{j | \|\rho_{i,j}^t\| \leq d\}, j \in \mathcal{V}_d$, and $d \geq 0$ is a set distance threshold parameter. Thus, the last term of $r_{\text{shape}_i}^t$ is a collision punishment to avoid the defender damage.

Upon intermediate events, the reward $r_{\text{shape}_i}^t$ is typically modified by a sharp increase or decrease. To avoid such dramatic changes, an event reward $r_{\text{event}_i}^t$ is designed in (7) to modify the corresponding reward histories in the replay-buffer D. The sharp changes in the reward are decomposed into difference sequence, where sequences closer to the event are given larger absolute values.

$$r_{\text{event}_i} = k_{\text{event}} \times (T - T_c + \Delta t) / \Delta t$$
$$r_{\text{event}_i}^t = r_{\text{event}_i} \times 2\Delta t / T_c (T_c + \Delta t) \tag{7}$$

where r_{event_i}, $r_{\text{event}_i}^t$ are the total event reward and the event reward at moment t, respectively. $r_{\text{event}_i} = \sum_{t \in T_c} r_{\text{event}_i}^t$. T_c denotes event-happening instant, and Δt is the time interval. k_{event} are coefficients of the four key events a)−d), calculated as: a) $k_{\text{event}} = -1$, if the target intruder arrives. $k_{\text{event}} = -1/n$, if other intruders arrives. b) $k_{\text{event}} = 2$, if the target intruder is intercepted. c) $k_{\text{event}} = -2$, if the defender is damaged. d) $k_{\text{event}} = -(N_{\text{arr}} + N_{\text{int1}}/n)$, if defenders lose. Here, N_{arr} is the number of arrival intruders, and N_{int1} is the number of running intruders. $k_{\text{event}} = n$, if defenders win.

In brief, $r_{\text{shape}_i}^t$ encourages defenders to pursue intruders, whereas $r_{\text{event}_i}^t$ update defender behavior according to interception results. Both of the two terms cooperatively to prevent the policy $\pi_{\theta i}$ from falling into local optima. By constantly performing interception episodes ω, the training process is divided into an outer loop with the time t as the label and an inner loop with the training iteration k as the label. An outer loop ends when ω ends, whereas an inner loop ends when k reaches a given threshold $\varsigma(\in \mathbb{N}^+)$. Thus, at every instant t, the network parameters θ_i, ε_i are updated ς times, and the training process of MAIM is proposed in Algorithm 1.

Algorithm 1 The Training Process of MAIM

Step 1: Select actions $a_i^t = \pi_{\theta i}(o_i^t) + \eta_t$ for defenders, w.r.t. the current policy $\pi_{\theta i}$ and exploration η_t, and execute a_i^t in MAIF by (1);

Step 2: Obtain the state s_i^t and the shape reward $r_{\text{shape}_i}^t$ by (5) and (6), and store the transitions $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$ in replay buffer D;

Step 3: Calculate the event rewards r_{event_i} by (7), if the intermediate events happen, and add $r_{\text{event}_i}^t$ to revise the buffer D;

Step 4: Sample a random minibatch samples from D at every instant t, and train the actor-critic network parameters θ_i, ε_i, i.e., $\theta_i^{k+1} = \theta_i^k + \alpha_c \nabla_{\theta_i} J(\theta_i)$, $\varepsilon_i^{k+1} = \varepsilon_i^k - \alpha_c \nabla L(\varepsilon_i), k \in [0, \varsigma]$;

The main analytical result concerning the convergence of the proposed MAIM is given below.

Theorem 1: For an MAS (1) governed by the (2) with learning process of critic networks by $\varepsilon_i^{k+1} = \varepsilon_i^k - \alpha_c \nabla_{\varepsilon_i} L(\varepsilon_i)$, where α_c is the critic networks' learning rate. Assuming that the time interval Δt is constant, the parameters ε_i^k of each defender asymptotically converges to the optimum value ε_i^* with iteration index $k \to \infty$, if the learning rate α_c satisfies

$$0 < \alpha_c < 2. \tag{8}$$

In other words, Problem 1 is solved by Algorithm 1.

Proof: For defender i, denote the critic network output $\widehat{Q}_i^k(x)$ by $\widehat{Q}_i^k(x) = (\varepsilon_i^k)^T \phi(x)$, where x is the critic network input, including the states s and the actions a of all the defenders $\mathcal{V}_d$, and

$$\phi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{9}$$

is the hyperbolic activation function of critic networks. For conciseness, we omit the symbol "(x)" in the following context. The error e_i^k between the reward prediction $\widehat{Q}_i^k$ and the real reward R_i writes $e_i^k = \widehat{Q}^k - R_i$, $L_i^k = \frac{1}{2}\left(e_i^k\right)^2$, where L_i^k is the square loss for obtaining

gradients. To minimize L_i^k, ε_i^k is updated by $\varepsilon_i^{k+1} = \varepsilon_i^k - \alpha_c \partial L_i^k / \partial \varepsilon_i^k = \varepsilon_i^k - \alpha_c e_i^k$, where α_c is the learning rate of critic networks, and $\alpha_c \geq 0$. then the error of the parameters $\bar{\varepsilon}_i^k$ can be expressed as $\bar{\varepsilon}_i^k = \varepsilon_i^k - \varepsilon_i^*$. Thus, one has $\bar{\varepsilon}_i^{k+1} = \bar{\varepsilon}_i^k - \alpha_c e_i^k \phi$. Now, design a discrete-time Lyapunov candidate as $V(k) = (\bar{\varepsilon}_i^k)^T \bar{\varepsilon}_i^k$, whose difference writes: $\Delta V(k) = V(k+1) - V(k) = (\bar{\varepsilon}_i^{k+1})^T \bar{\varepsilon}_i^{k+1} - (\bar{\varepsilon}_i^k)^T \bar{\varepsilon}_i^k = [(\bar{\varepsilon}_i^k)^T - \alpha_c \phi^T (e_i^k)^T] (\bar{\varepsilon}_i^k - \alpha_c e_i^k \phi) - (\bar{\varepsilon}_i^k)^T (\bar{\varepsilon}_i^k) = -\alpha_c \phi^T (e_i^k)^T \bar{\varepsilon}_i^k - \alpha_c (\bar{\varepsilon}_i^k)^T e_i^k \phi + \alpha_c^2 (e_i^k \phi)^T e_i^k \phi = -\alpha_c \phi^T (e_i^k)^T (\varepsilon_i^k - \varepsilon_i^*) - \alpha_c (\varepsilon_i^k - \varepsilon_i^*)^T e_i^k \phi + \alpha_c^2 \|e_i^k\|^2 \|\phi\|^2 = \alpha_c \|e_i^k\|^2 (-2 + \alpha_c \|\phi\|^2)$.

According to (8) and (9), one has $-2 + \alpha_c \|\phi\|^2 < 0$, and hence $\Delta V < 0$, implying that ε_i^k asymptotically converges to ε_i^*. Problem 1 is thus solved by Algorithm 1. ∎

Remark 1: The learning rate α_c can be generally picked as a small value among $[0.01, 0.03]$ to guarantee the convergence of MAIM.

Numerical simulations: The network parameters are selected referring to [4], listed in Table 1. The present MAIF is conducted in a bounded coordinate system with a range $([0, 200]^2)$, and a region $(\rho_c = (100, 100), d_c = 15)$. The numbers of intruders and defenders are $n = 5$ and $m = 3$. To ensure the generalization of MAIM, initial positions ρ_i^0 of agents are randomly generated, satisfying $d_c < \|\rho_i^0 - \rho_c\| < 2d_c, i \in \mathcal{V}_d$, and $\|\rho_j^0 - \rho_c\| > 6d_c, j \in \mathcal{V}_r$. Besides, kinetic parameters are set as $v_i^{max} = a_i^{max} = 10$, $v_j^{max} = a_j^{max} = 6$, $i \in \mathcal{V}_d$, $j \in \mathcal{V}_r$.

Table 1. Structures of Actor and Critic Networks

Networks	Activation function and neurons			Learning rates 0.02
	Input layer	3 hidden layers	Output layer	
Actor	ReLu, 80	ReLu, 64	SoftMax, 5	Optimizer
Critic	ReLu, 255	ReLu, 64	Tanh, 1	Adam

The training processes are presented in Fig. 3, ten independent interception simulations are performed after every training episode to reduce the influence of the randomness of the initial values. Cumulative rewards R_i of MAIM gradually increase, whereas R_i of MAL-S fluctuate continuously, which is caused by the high environmental nonstationarity. In the statistical results of 1×10^5 simulations, MAIM improves the win rate from 0 to 89.76%, whereas MAL-S usually intercepts 2 and 3 intruders, which has lost the confrontation. To further explore the reason of performance differences, Fig. 4 presents interception examples. For MAIM, intruders 5, 7 and 4 are intercepted by defenders 1, 2 and 3 in the initial period. Then, defender 1 hits its nearest intruder 6, and all defenders head to the last running intruder 8. Finally, MAIM safely guides defenders to win. By contrast, defenders approach the targets during the initial period for MAL-S. However, according to the trajectories of defender 1 and intruder 6, defenders are apt to lose interception ability, once intruders execute some evasive moves. Although defenders slow down and turn around, it is too late to catch up with intruders. Finally, no intruder is intercepted. In this scenario, MAL-S is trapped in a local optimum. In summary, the above numerical simulations have verified the effectiveness of the present MAIM.

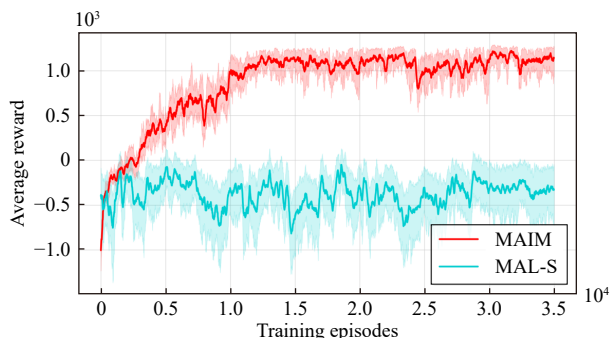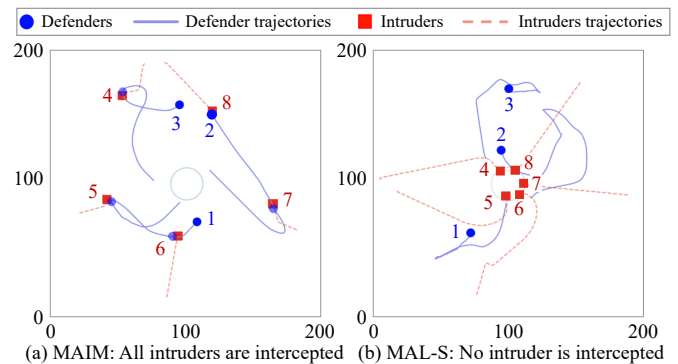**Conclusion and future works:** This letter proposes a more effi-

Fig. 4. An example of interception processes of MAIM and MAL-S.

cient reinforcement learning interception method in antagonistic environments, namely MAIM, by sophisticatedly designing a reward function. Significantly, sufficient conditions are derived to guarantee the convergence of MAIM. Finally, the effectiveness of the proposed method is verified by extensive numerical simulations. Future work will focus on more challenging collective interception version in higher dimensional spaces with dynamic obstacles.

References

[1] Y. Zheng, A. Lai, X. Yu, and W. Lan, "Early-awareness collision avoidance in optimal multi-agent path planning with temporal logic specifications," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 5, pp. 1346–1348, 2023.

[2] B. Ning, Q.-L. Han, Z. Zuo, L. Ding, Q. Lu, and X. Ge, "Fixed-time and prescribed-time consensus control of multiagent systems and its applications: A survey of recent trends and methodologies," *IEEE Trans. Industr. Inform.*, vol. 19, no. 2, pp. 1121–1135, 2023.

[3] R. Lowe, Y. Wu, A. Tamar, J. Harb, I. Pieter, and A. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6382–6393.

[4] R. Zhang, Q. Zong, X. Zhang, L. Dou, and B. Tian, "Game of drones: Multi-UAV pursuit-evasion game with online motion planning by deep reinforcement learning," *IEEE Trans. Neural. Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7900–7909, Oct. 2023.

[5] Y. Yu, J. Liu, and C. Wei, "Hawk and pigeon's intelligence for UAV swarm dynamic combat game via competitive learning pigeon-inspired optimization," *Sci. China Technol. Sci.*, vol. 65, no. 5, pp. 1072–1086, 2022.

[6] Z. Zhang and D. Zhao, "Clique-based cooperative multiagent reinforcement learning using factor graphs," *IEEE/CAA J. Autom. Sinica*, vol. 1, no. 3, pp. 248–256, 2014.

[7] L. Xue, C. Sun, D. Wunsch, Y. Zhou, and F. Yu, "An adaptive strategy via reinforcement learning for the prisoner's dilemma game," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 301–310, 2017.

[8] T. Rupprecht and Y. Wang, "A survey for deep reinforcement learning in markovian cyber-physical systems: Common problems and solutions," *Neural Netw.*, vol. 153, pp. 12–36, 2022.

[9] L. Huang, M. Fu, H. Qu, S. Wang, and S. Hu, "A deep reinforcement learning-based method applied for solving multi-agent defense and attack problems," *Expert Syst. Appl.*, vol. 176, p. 114896, 2021.

[10] B. Wang, S. Li, X. Gao, and T. Xie, "UAV swarm confrontation using hierarchical multiagent reinforcement learning," *Int. J. Aerosp. Eng.*, vol. 2021, p. 3360116, 2021.

[11] T. Zhang, L. Chai, S. Wang, J. Jin, X. Liu, A. Song, and Y. Lan, "Improving autonomous behavior strategy learning in an unmanned swarm system through knowledge enhancement," *IEEE Trans. Reliab.*, vol. 71, no. 2, pp. 763–774, 2022.

[12] Z. Chen and N. Li, "An optimal control-based distributed reinforcement learning framework for a class of non-convex objective functionals of the multi-agent network," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 11, pp. 2081–2093, 2023.

Fig. 3. Average cumulative reward of MAIM and MARL-S.