SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2025, Vol. 68, Iss. 1, 112206:1–112206:16
https://doi.org/10.1007/s11432-023-4183-4

Distributed optimal consensus control for multiagent systems based on event-triggered and prioritized experience replay strategies

Cuijuan ZHANG[1,2], Lianghao JI[1*], Shasha YANG[1], Xing GUO[1] & Huaqing LI[3]

[1]*Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China;*
[2]*Key Laboratory of Intelligent Perception and Computing of Anhui Province, Anqing Normal University, Anqing 246133, China;*
[3]*College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China*

Received 31 October 2023/Revised 20 June 2024/Accepted 23 August 2024/Published online 26 December 2024

Abstract This study uses event-triggered (ET) and reinforcement learning methods to investigate the optimal consensus control problem for cooperative-competitive multiagent systems. It proposes a novel distributed ET control strategy, which relies on a prioritized experience replay (PER) policy. This strategy not only conserves communication resources but also ensures acceptable system performance. To implement the proposed method, actor-critic (AC) dual-structured neural networks (NNs) are used to approximate the value function and control policy. In the AC NNs, the weight estimates for the NNs are updated at the moment of event triggering, resulting in a nonperiodic weight adjustment pattern. This approach decreases the computational cost in comparison with the traditional ET mechanism. The PER-based ET mechanism makes full use of valid historical data and effectively establishes a balance between system performance and communication resource conservation. Moreover, it does not require the following two conditions in most existing studies: (1) requirement of the system dynamics model to be known, and (2) persistent excitation. In addition, Zeno behavior is excluded from this study. Finally, a simulation is conducted to confirm the validity of the suggested approach.

Keywords multiagent systems, event-triggered mechanism, prioritized experience replay, dual-structured neural networks, optimal consensus control

Citation Zhang C J, Ji L H, Yang S S, et al. Distributed optimal consensus control for multiagent systems based on event-triggered and prioritized experience replay strategies. Sci China Inf Sci, 2025, 68(1): 112206, https://doi.org/10.1007/s11432-023-4183-4

1 Introduction

Consensus control of multiagent systems (MASs) has been used with great progress in various areas, such as formation control and obstacle avoidance maintenance [1,2], smart grids [3,4], autonomous submersibles [5,6], and flocking [7]. Optimal consensus control problems have received increased attention as a class of typical consensus problems. They aim to create a controller that achieves the desired control effect while balancing between system performance and usage of available resources. Thus far, optimization has been established as a primary guiding principle in the design of modern control. Therefore, the optimal consensus control problem is an active and intriguing topic associated with the analytical solution of the Hamilton-Jacobi-Bellman (HJB) equation [8]. Obtaining the exact model of the system is often difficult in practice. Moreover, the number of dimensions increases with the number of agents, considerably increasing the computational and storage costs. Consequently, obtaining an analytical solution to the HJB equation would be almost impossible. Werbos [9] proposed an approximation strategy based on adaptive learning to overcome these difficulties. The primary objective was to approximate the solution of the HJB equation by exploiting the approximation properties of the value function. Numerous notable results have been continuously published [10–12].

Reinforcement learning (RL) is an approximate strategy that has been receiving increasing attention [13–15]. The essential concept behind RL is that an agent interacts with its environment, receives

* Corresponding author (email: jilh@cqupt.edu.cn)

feedback, and then accordingly optimizes its control policy [16]. Actor-critic (AC) dual-structured neural networks (NNs) are a common method for the implementation of RL. In these networks, the critic network evaluates the control behavior and sends the results back to the actor network, which then uses the results to optimize the control behavior. The policy gradient and generalized policy gradient methods, which are based on the architectures of AC NNs, are used to solve HJB equations without requiring an exact model of the system. Lately, these methods have been widely used in linear systems [17–19] and nonlinear systems [20–24]. A novel policy iteration algorithm was employed to address the optimal consensus control problem for MASs [19]. Ref. [21] investigated the finite-time optimal consensus problem for nonlinear strict-feedback MASs, while Ref. [22] addressed the same problem for stochastic MASs. Refs. [23, 24], based on the identifier-AC architecture, focused on the design of finite-time optimal formation controllers for nonlinear MASs.

Notably, the existing RL algorithms require the traditional persistence of excitation conditions to fully enable the agent's interaction with the environment and prevent it from succumbing to local optimization, as a result of which the system can become unstable [8]. In continuous-time linear systems, experience replay (ER) is used to store historical experiences and use current data to determine critic weights instead of relying on the persistence of excitation conditions. Ref. [25] employed ER techniques to avoid the need for an initially admissible control policy. However, it might have introduced bias propagation, resulting in an underestimation of the value function. Underestimations are typically nonuniform, which poses challenges in the selection of the optimal control method.

Various methods should be explored to minimize estimation bias in order to achieve more precise tracking control. Schaul [26], a senior research scientist at DeepMind, proposed the prioritized ER (PER) mechanism. PER considerably increases the frequency of reusing the experiences that have a crucial impact on model learning and potential value by introducing a prioritized sorting strategy for experience data. This expedited learning processes and facilitated the optimization of model performance. The PER mechanism finds widespread use across various domains, including tracking control for autonomous underwater vehicles [27], decision optimization for autonomous driving systems [28], and data processing in the Internet of Things [29]. In the field of collaborative optimization in MASs, the introduction of PER has resulted in several novel approaches to address this complex issue. Ref. [30] innovatively integrated PER's efficient learning capabilities with heuristic dynamic programming, effectively increasing the efficiency of MASs in collaborative decision-making processes by prioritizing the replay of critical experience data. Ref. [31] further explored the optimal consensus control problem in multidelay MASs based on PER, enhancing the system's collaborative control performance via optimized use of experience data.

The abovementioned RL-based studies require real-time communication between agents in order to establish system stability and minimize system consumption. However, real-time communication might increase communication channel load and computing load [32–35]. Therefore, establishing a balance between communication efficiency and system stability is crucial. Event-triggered (ET) control is a realistic manner for determining the moment of control, as it ensures that state signals are collected only if a predefined trigger condition is broken. Consequently, only a small number of state signals are sent. Recently, RL based on the ET mechanism has been extensively used in MASs, and it has been developed to solve consensus problems [36–45]. However, in many cases, parameters associated with the trigger threshold are fixed and cannot be adjusted. This limitation can restrict the application scenarios of ET mechanisms (ETMs) [46]. Accordingly, dynamic ETMs [47] and adaptive ETMs [37] were devised to eliminate the limits of classic ETMs.

In most ET conditions, the differences between current-state data and the latest transmitted state data are evaluated to determine whether or not to transmit data. If the difference between these sampled data is sufficiently small, the data will not be transmitted and will be discarded. However, the abovementioned principle might not be valid when the value of the current sampled data is considerably larger than that of the other data. Considering transmitted data history, the system can exhibit better control performance, albeit with the potential to increase the number of triggers [48, 49]. Therefore, the following interesting question arises: "How to design a suitable ETM that effectively establishes a balance between system performance and communication resource conservation?" Attempting to answer this question is the main motivation for this study.

In the previous decade, most studies on consensus control of MASs have assumed that agents are cooperative [17,50,51]. Ref. [51] studied cooperative evolution of high-order temporal networks. However, in real-world scenarios, competition between agents is also common because of the limited resources.

Table 1 Abbreviations and full names.

Abbreviation	Full name
ET	Event-triggered
PER	Prioritized experience replay
AC	Actor-critic
NNs	Neural networks
MASs	Multiagent systems
HJB	Hamilton-Jacobi-Bellman
RL	Reinforcement learning
ER	Experience replay
NN	Neural network
ETMs	Event-triggered mechanisms
ET-ER	Event-triggered method based on experience replay
ET-PER	Event-triggered method based on prioritized experience replay

Ref. [52] examined the issue of collective optimal decision-making within anti-coordinated agent networks. Furthermore, cooperative-competitive interaction always exists in many complex systems, such as railway transportation [53–55]. Therefore, this relationship is both universal and practical.

In summary, we investigate the fully distributed optimal consensus problem for cooperative-competitive MASs. A PER-based ET method, referred to as the ET-PER method, is used to address this issue. The following are the main contributions of this study.

(1) A novel ET method based on PER is designed. In particular, we use the priority in PER as the basis for the selection of effective historical data and the designing of ET conditions. Compared with [36,37], the proposed method makes use of historical information on MASs and can effectively establish a balance between system performance and communication resource conservation.

(2) An ET AC NN is constructed to learn the optimal control policy. In comparison with [13], the NN weights are nonperiodically adjusted, meaning that the AC network is solely adjusted at the triggering moment, decreasing the computational complexity.

(3) The algorithm proposed in this study employs the PER approach, which offers several advantages versus traditional RL algorithms. First, it reduces estimation bias by breaking the strong correlation between data and satisfying the independent identically distributed assumption required by stochastic gradient algorithms. Second, it selects useful but infrequent experiences, shortening the transient response process and increasing system safety. Third, it allows for MASs to fully interact with the environment, avoiding the persistence of excitation conditions.

Table 1 presents the abbreviations, along with their expansions, used in this study.

The rest of this paper is structured as follows. Section 2 provides the problem statement, and Section 3 presents the proposed algorithm. Section 4 introduces an online solution method for NNs based on ET-PER. Section 5 experimentally validates the results obtained. Finally, Section 6 concludes this paper.

2 Problem statement

2.1 Communication topology

MASs with N agents communicating through the network can be modeled as a graph $\tilde{G} = (\tilde{\Pi}, \Pi, A_{\tilde{\Pi}})$, where $\tilde{\Pi} = (o_1, o_2, \ldots, o_N)$ represents the finite node-set and $A_{\tilde{\Pi}} = [a_{ij}]_{N\times N}$ represents the weighted adjacency matrix. $\Pi = \{\Pi_{ij} = (o_i, o_j)\}$ denotes the connection edge set of the agents. If $(o_i, o_j) \in \Pi$, then it is indicated that node o_i can receive information from node o_j, and the corresponding adjacency matrix element is defined as $a_{ij} \neq 0$. Otherwise, if $(o_i, o_j) \notin \Pi$, then $a_{ij} = 0$. In particular, when there exists a cooperative relationship between agents i and j, element a_{ij} of the adjacency matrix is positive; conversely, a_{ij} is negative if the two agents are in a competitive relationship.

As per the weighted adjacency matrix $A_{\tilde{\Pi}}$, the Laplacian matrix H can be obtained as $H = D - A_{\tilde{\Pi}}$, where $D = \operatorname{diag}\{d_i\}$. The diagonal elements are defined as $d_i = \sum_{j\in N_i} |a_{ij}|$.

We can employ a diagonal matrix $B = \operatorname{diag}\{b_1, b_2, \ldots, b_N\}$ to represent the relationships between the followers and the leader as follows: $b_i = 1$ if a follower can receive the leader's message, and $b_i = 0$ otherwise. It must be ensured that a small number of followers can receive messages from the leader, i.e.,

$b_1 + b_2 + \cdots + b_N > 0$.

Node-set $\tilde{\Pi} = (o_1, o_2, \ldots, o_N)$ can be split into two distinct subsets, $\tilde{\Pi}_1$ and $\tilde{\Pi}_2$, if the signed graph $\tilde{G}$ is structurally balanced. The following requirements are fulfilled in this case: (1) $\tilde{\Pi} = \tilde{\Pi}_1 \cup \tilde{\Pi}_2, \tilde{\Pi}_1 \cap \tilde{\Pi}_2 = \emptyset$; and (2) $a_{ij} \geqslant 0, \forall i, j \in \tilde{\Pi}_T$ $(T \in \{1, 2\})$ and $a_{ij} \leqslant 0, \forall i \in \tilde{\Pi}_T, j \in \tilde{\Pi}_{\hat{T}}, T \neq \hat{T}$ $(T, \hat{T} \in \{1, 2\})$.

2.2 Problem formulation

Considering the leader-follower MASs with N followers, where each agent is represented by the dynamic system given below:

$$\dot{x}_i(t) = Ax_i(t) + B_i\mu_i(t), \tag{1}$$

where $x_i(t) \in \mathbb{R}^n$ represents the position state, $\mu_i(t) \in \mathbb{R}^{m_i}$ is the control input vector of agent i. $A \in \mathbb{R}^{n\times n}$ and $B_i \in \mathbb{R}^{n\times m_i}$ are the dynamic and feedback gain matrices, respectively.

The reference signal for the position is given by

$$\dot{x}_0(t) = Ax_0(t), \tag{2}$$

where $x_0(t) \in \mathbb{R}^n$.

Definition 1 ([15]). For $i \in \{1, 2, \ldots, N\}$, the MASs (1) is said to reach consensus in the sense of cooperative-competitive if the following two conditions are satisfied: (1) $\lim_{t\to\infty} \|x_i(t) - x_0(t)\| = 0$, if $i \in \tilde{\Pi}_1$. (2) $\lim_{t\to\infty} \|x_i(t) + x_0(t)\| = 0$, if $i \in \tilde{\Pi}_2$.

Assumption 1 ([56]). The graph $\tilde{G}$ representing the connection topology has a spanning tree that includes the leader and does not contain any repeated edges.

Assumption 2 ([38]). For $x, y \in \mathbb{R}^n$, if there exists positive constant h which satisfies $\|f(x) - f(y)\| \leqslant h\|x - y\|$, then $f : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous.

The local neighbor state error is intended to prescribe consensus objectives in order to ensure that all agents track the reference signals.

Agent i's local state error is defined as

$$\varsigma_i(t) = \sum_{j\in N_i} |a_{ij}| (x_i(t) - \mathrm{sign}(a_{ij})x_j(t)) + b_i(x_i(t) - w_i x_0(t)), \tag{3}$$

where the sign function is

$$\mathrm{sign}(a_{ij}) = \begin{cases} 1, & a_{ij} > 0, \\ -1, & a_{ij} < 0. \end{cases}$$

N_i represents the set of agent i's neighboring nodes. The gauge transformation matrix is $W = \mathrm{diag}\{w_1, w_2, \ldots, w_N\}$, where $w_i = 1$ for $i \in \tilde{\Pi}_1$, and $w_i = -1$ for $i \in \tilde{\Pi}_2$.

Let $\varsigma(t) = \left(\varsigma_1^{\mathrm{T}}(t), \varsigma_2^{\mathrm{T}}(t), \ldots, \varsigma_N^{\mathrm{T}}(t)\right)^{\mathrm{T}} \in \mathbb{R}^{nN}$ be the global state error vector; then we can obtain the compact expression as

$$\varsigma(t) = ((H + B) \otimes I_n)\left(x(t) - \bar{W}\bar{x}_0(t)\right), \tag{4}$$

where $x(t) = \left(x_1^{\mathrm{T}}(t), x_2^{\mathrm{T}}(t), \ldots, x_N^{\mathrm{T}}(t)\right) \in \mathbb{R}^{nN}$, $\bar{x}_0(t) = \left(x_0^{\mathrm{T}}(t), \ldots, x_0^{\mathrm{T}}(t)\right)^{\mathrm{T}} \in \mathbb{R}^{nN}$, $\bar{W} = W \otimes I_n$, I_n is the n-dimensional identity matrix, and $\otimes$ is the Kronecker product.

The tracking error is defined as $\sigma_i(t) = x_i(t) - w_i x_0(t)$, and its compact form can be represented as

$$\sigma(t) = x(t) - \bar{W}\bar{x}_0(t). \tag{5}$$

Combining the system dynamics described in (1)–(3), the dynamic of $\varsigma_i(t)$ is formulated by

$$\begin{aligned} \dot{\varsigma}_i(t) =& \sum_{j\in N_i} |a_{ij}| (\dot{x}_i(t) - \mathrm{sign}(a_{ij})\dot{x}_j(t)) + b_i(\dot{x}_i(t) - w_i\dot{x}_0(t)) \\ =& A\varsigma_i(t) + \left(b_i + \sum_{j\in N_i} |a_{ij}|\right) B_i\mu_i(t) - \sum_{j\in N_i} |a_{ij}| (\mathrm{sign}(a_{ij})B_j\mu_j(t)) \\ =& f_i(\varsigma_i(t), \mu_i(t)). \end{aligned} \tag{6}$$

According to Assumption 1, (4), and (5), we can obtain that once $\varsigma(t) \to 0$, then $\sigma(t) \to 0$ as $t \to \infty$.

3 ET-PER based tracking controller design

In [15, 17], the absence of ET mechanisms for communication among agents resulted in unnecessary and frequent information exchanges. This can cause communication channels to become blocked, wasting resources or even interrupting communication. In [32], it introduces an ET mechanism by subtracting the error of the latest state information from the current state information. However, only the current error information is insufficient to reflect the full dynamic characteristics of the system, which can lead to the growth of transient processes and affect the system's stability. To address this issue, we propose the ET-PER-based algorithm, which effectively utilizes the historical trigger data.

To investigate the optimal consensus of the MASs, inspired by [17], the local cost function $J_i(\varsigma_i(0))$ is defined as

$$J_i(\varsigma_i(0)) = \frac{1}{2}\int_0^{+\infty} \Re_i(\varsigma_i(t), \mu_i(t), \mu_{N_i}(t))\,\mathrm{d}t, \tag{7}$$

where $\Re_i(\varsigma_i, \mu_i, \mu_{N_i}) = \varsigma_i^{\mathrm{T}}(t) Q_{ii}\varsigma_i(t) + \mu_i^{\mathrm{T}}(t) R_{ii}\mu_i(t) + \sum_{j\in N_i}\mu_j^{\mathrm{T}}(t) R_{ij}\mu_j(t)$ is the utility function, and μ_{N_i} denotes the set of the neighbors control of agent i. $Q_{ii} > 0$, $R_{ii} > 0$, and $R_{ij} > 0$ are symmetric positive definite matrices. As a result, the following constitutes a definition of the state-value function:

$$V_i(\varsigma_i(t)) = \frac{1}{2}\int_t^{+\infty} \Re_i(\varsigma_i, \mu_i, \mu_{N_i})\,\mathrm{d}\tau. \tag{8}$$

Definition 2 (Admissible control [12])**.** For agent i, the control policy $\mu_i(t)$ will be considered admissible if it can achieve both the stabilization of the tracking error system (6) and ensure the finiteness of (7).

Taking the derivative of the state-value function (8) with respect to time t, the local-coupled HJB equation for agent i is shown as

$$\begin{aligned} H_i(\varsigma_i, V_{\varsigma_i}, \mu_i) =& V_{\varsigma_i}^{\mathrm{T}} \cdot \dot{\varsigma}_i + \frac{1}{2}\left(\varsigma_i^{\mathrm{T}}(t) Q_{ii}\varsigma_i(t) + \mu_i^{\mathrm{T}}(t) R_{ii}\mu_i(t)\right) + \frac{1}{2}\sum_{j\in N_i}\mu_j^{\mathrm{T}}(t) R_{ij}\mu_j(t) \\ =& 0, \end{aligned} \tag{9}$$

where $V_{\varsigma_i}^{\mathrm{T}}$ refers to the partial derivative of the state-value function (8) with respect to ς_i.

On the base of Definition 2, Eq. (8) can become

$$V_i^*(\varsigma_i) = \min_{\mu_i}\frac{1}{2}\int_t^{+\infty} \Re_i(\varsigma_i, \mu_i, \mu_{N_i})\,\mathrm{d}\tau. \tag{10}$$

Then, from Bellman's optimality principle, the optimal control for agent i is given as

$$\mu_i^*(t) = -(b_i + d_i) R_{ii}^{-1} B_i V_{\varsigma_i}^*. \tag{11}$$

Substituting the optimal control policy $\mu_i^*(t)$ into (9) yields

$$\begin{aligned} H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*\right) =& V_{\varsigma_i}^{*\mathrm{T}} \cdot \left(A\varsigma_i(t) + (b_i + d_i) B_i\mu_i^*(t)\right) - V_{\varsigma_i}^{*\mathrm{T}} \cdot \left(\sum_{j\in N_i}|a_{ij}|\left(\mathrm{sign}(a_{ij})B_j\mu_j(t)\right)\right) \\ &+ \frac{1}{2}\left(\varsigma_i^{\mathrm{T}}(t) Q_{ii}\varsigma_i(t) + \mu_i^{*\mathrm{T}}(t) R_{ii}\mu_i^*(t)\right) + \frac{1}{2}\sum_{j\in N_i}\mu_j^{*\mathrm{T}}(t) R_{ij}\mu_j^*(t) \\ =& V_{\varsigma_i}^{*\mathrm{T}} \cdot \left(A\varsigma_i(t) - \sum_{j\in N_i}|a_{ij}|\left(\mathrm{sign}(a_{ij})B_j\mu_j(t)\right)\right) \\ &+ \frac{1}{2}\left(\varsigma_i^{\mathrm{T}}(t) Q_{ii}\varsigma_i(t) - \mu_i^{*\mathrm{T}}(t) R_{ii}\mu_i^*(t)\right) + \frac{1}{2}\sum_{j\in N_i}\mu_j^{*\mathrm{T}}(t) R_{ij}\mu_j^*(t) \\ =& 0. \end{aligned} \tag{12}$$

Let $w_i(t) = \|e_{i,k}\|^2 - \frac{\left(1-\alpha^2\right)\underline{\lambda}(Q_{ii})}{\bar{\lambda}(R_{ii})P^2}\|\varsigma_i(t)\|^2 - \frac{\underline{\lambda}(R_{ii})}{\bar{\lambda}(R_{ii})P^2}\|\mu_i^*(t_{i,k})\|^2$, where $0 < \alpha < 1$ and $\bar{\lambda}(R_{ii})$ represent the maximum eigenvalue of matrices R_{ii}. $\underline{\lambda}(R_{ii})$ and $\underline{\lambda}(Q_{ii})$ represent the minimum eigenvalues of

matrices R_{ii} and Q_{ii}, respectively. Thus, the definition of the expression $t_{i,k+1}$ is given as $t_{i,k+1} = \inf\{t : t > t_{i,k}, w_i(t) < 0\}$. $t_{i,k}$ represents the k-th ET instant for agent i. The sequence of triggering instants for each agent i is described as a strictly monotonically increasing sequence $\{t_{i,k}\}_{k=0}^{\infty}$ that satisfies the condition $t_{i,k} < t_{i,k+1}, k = 1, 2, \ldots$. To establish the trigger condition, the ET error between the current error state $\varsigma_i(t)$ and the weighted sampled disagreement error $\hat{\varsigma}_i(t_{i,k})$ is described below:

$$e_{i,k}(t) = \hat{\varsigma}_i(t_{i,k}) - \varsigma_i(t), \ t \in [t_{i,k}, t_{i,k+1}), \tag{13}$$

where $\hat{\varsigma}_i(t_{i,k}) = \alpha_i \cdot \varsigma_i(t_{i,s}) + \varsigma_i(t_{i,k})$, and α_i (see (51)) is the importance sampling weight in the prioritized experience replay. Following the control policy based on event triggers, the local error (6) can be expressed as

$$\dot{\varsigma}_i(t) = f_i(\varsigma_i(t), \mu_i(\hat{\varsigma}_i(t_{i,k}))). \tag{14}$$

Remark 1. The goal of the ET-PER method is to achieve a harmonious balance between enhancing system performance and minimizing the usage of communication resources. For $i \in \{1, 2, \ldots, N\}$, the design of the ET condition relies on the ET error as well as the state-dependent threshold. The control input $\mu_i(t_{i,k})$ can be updated at $t_{i,k}$ and retains its value until the next triggering instant. Then we have

$$\mu_i^* = \mu_i^*(t_{i,k}) = -(b_i + d_i) R_{ii}^{-1} B_i V_{\varsigma_i}^*(t_{i,k}). \tag{15}$$

Substituting (15) into (9), we can obtain the HJB equation based on ET:

$$\begin{aligned}
&H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*(t_{i,k})\right) \\
&= V_{\varsigma_i}^{*\mathrm{T}}\left(A\varsigma_i(t) - (b_i + d_i)^2 B_i R_{ii}^{-1} B_i^{\mathrm{T}} V_{\varsigma_i}^*(t_{i,k})\right) - V_{\varsigma_i}^{*\mathrm{T}}\left(\sum_{j\in N_i} |a_{ij}|(\mathrm{sign}(a_{ij}) B_j \mu_j(t))\right) \\
&\quad + \frac{1}{2}\sum_{j\in N_i} \mu_j^{*\mathrm{T}}(t) R_{ij} \mu_j^*(t) + \frac{1}{2}\varsigma_i^{\mathrm{T}}(t) Q_{ii} \varsigma_i(t) \\
&\quad + \frac{1}{2}(b_i + d_i)^2 V_{\varsigma_i}^{*\mathrm{T}}(t_{i,k}) B_i^{\mathrm{T}} R_{ii}^{-\mathrm{T}} R_{ii} R_{ii}^{-1} B_i V_{\varsigma_i}^*(t_{i,k}) \\
&= V_{\varsigma_i}^{*\mathrm{T}}\left(A\varsigma_i(t) - \sum_{j\in N_i} |a_{ij}|(\mathrm{sign}(a_{ij}) B_j \mu_j(t))\right) - \mu_i^{*\mathrm{T}} R_{ii} \mu_i^*(t_{i,k}) + \frac{1}{2}\varsigma_i^{\mathrm{T}}(t) Q_{ii} \varsigma_i(t) \\
&\quad + \frac{1}{2}\mu_i^{*\mathrm{T}}(t_{i,k}) R_{ii} \mu_i^*(t_{i,k}) + \frac{1}{2}\sum_{j\in N_i} \mu_j^{*\mathrm{T}}(t) R_{ij} \mu_j^*(t).
\end{aligned} \tag{16}$$

Remark 2. The problem of distributed optimal tracking control for MASs is well-known to be a multi-player zero-sum game. The objective is to find a set of optimal control strategies in a Nash equilibrium sense that minimizes the sum of consumption functions.

Definition 3 (Nash equilibrium)**.** For $i \in \{1, 2, \ldots, N\}$, if a set of optimal control policies $\{\mu_1^*, \mu_2^*, \ldots, \mu_N^*\}$ makes the following inequality hold:

$$J_i^*(\mu_1^*, \ldots, \mu_i^*, \ldots, \mu_N^*) \leqslant J_i(\mu_1^*, \ldots, \mu_i, \ldots, \mu_N^*), \tag{17}$$

then $\{\mu_1^*, \mu_2^*, \ldots, \mu_N^*\}$ is referred to as the solution of the Nash equilibrium.

Assumption 3 ([39])**.** Assume that the admissible control policy of agent i satisfies the Lipschitz condition and that there exists a positive real constant P such that

$$\|\mu_i^*(\varsigma_i) - \mu_i^*(\hat{\varsigma}_i(t_{i,k}))\| \leqslant P\|e_{i,k}\|. \tag{18}$$

Lemma 1 ([40])**.** Assuming that all agents have admissible control policies, we can derive the following relation associated with the HJB equation based on (12) and (16):

$$\begin{aligned}
&H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*\right) - H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*(t_{i,k})\right) \\
&\quad = -\frac{1}{2}(\mu_i^*(t) - \mu_i^*(t_{i,k}))^{\mathrm{T}} R_{ii}(\mu_i^*(t) - \mu_i^*(t_{i,k})).
\end{aligned} \tag{19}$$

Lemma 2 ([36]). Under Assumption 3, in (13) and (14), it is possible to define a positive constant L_f such that the inequality holds:

$$\left\|f_i\left(\varsigma_i\left(t\right),\mu_i\left(\hat{\varsigma}_i\left(t_{i,k}\right)\right)\right)\right\| \leqslant L_f\left\|e_{i,k}\left(t\right)\right\| + L_f\left\|\varsigma_i\left(t\right)\right\|. \tag{20}$$

Lemma 3 ([40]). According to Assumption 1 and Lemma 1, it yields that

$$\left\|H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*\right) - H_i\left(\varsigma_i, V_{\varsigma_i}^*, \mu_i^*\left(t_{i,k}\right)\right)\right\| \leqslant \bar{\lambda}\left(R_{ii}\right) P^2 \|e_{i,k}\|^2. \tag{21}$$

The following theorem gives an analysis of the stability of the cooperative-competitive local error system (6), which is based on the ET control policies. Furthermore, we demonstrate that the suggested ET control policies are optimal in the sense of Nash equilibrium.

Theorem 1. Assuming that Assumptions 1–3 are satisfied. For $t \in [t_{i,k}, t_{i,k+1})$, the control policies are presented in (15). When the following triggering condition holds:

$$\|e_{i,k}\|^2 \leqslant \frac{\left(1-\alpha^2\right)\underline{\lambda}\left(Q_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}\left\|\varsigma_i\left(t\right)\right\|^2 + \frac{\underline{\lambda}\left(R_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}\left\|\mu_i^*\left(t_{i,k}\right)\right\|^2, \tag{22}$$

the cooperative-competitive error systems (6) reach asymptotically stable. This implies that MASs tracking control can be established. In addition, the occurrence of Zeno behavior in each agent can be avoided.

Proof. According to the value function's definition, $V_i^*\left(\varsigma_i\right) \geqslant 0$. Then, $V_i^*\left(\varsigma_i\right)$ can be chosen as a Lyapunov function:

$$\begin{aligned}
\dot{V}_i^*\left(\varsigma_i\right) =& V_{\varsigma_i}^{*\mathrm{T}} \cdot \dot{\varsigma}_i \\
=& V_{\varsigma_i}^{*\mathrm{T}}\left(A\varsigma_i\left(t\right) + \left(b_i + d_i\right)B_i\mu_i^*\left(t\right)\right) - V_{\varsigma_i}^{*\mathrm{T}}\left(\sum_{j\in N_i}\left|a_{ij}\right|\left(\mathrm{sign}(a_{ij})B_j\mu_j\left(t\right)\right)\right) \\
=& \frac{1}{2}\mu_i^{*\mathrm{T}}\left(t\right)R_{ii}\mu_i^*\left(t\right) - \mu_i^{*\mathrm{T}}\left(t\right)R_{ii}\mu_i^*\left(t_{i,k}\right) - \frac{1}{2}\varsigma_i^{\mathrm{T}}\left(t\right)Q_{ii}\varsigma_i\left(t\right) - \frac{1}{2}\sum_{j\in N_i}\mu_j^{*\mathrm{T}}\left(t\right)R_{ij}\mu_j^*\left(t\right) \\
\leqslant& \frac{1}{2}\mu_i^{*\mathrm{T}}\left(t\right)R_{ii}\mu_i^*\left(t\right) - \mu_i^{*\mathrm{T}}\left(t\right)R_{ii}\mu_i^*\left(t_{i,k}\right) - \frac{1}{2}\varsigma_i^{\mathrm{T}}\left(t\right)Q_{ii}\varsigma_i\left(t\right).
\end{aligned} \tag{23}$$

By Lemma 3, we are able to derive

$$\dot{V}_i^*\left(\varsigma_i\right) \leqslant -\frac{1}{2}\underline{\lambda}\left(Q_{ii}\right)\left\|\varsigma_i\left(t\right)\right\|^2 + \frac{1}{2}\bar{\lambda}\left(R_{ii}\right)P^2\|e_{i,k}\|^2 - \frac{1}{2}\underline{\lambda}\left(R_{ii}\right)\left\|\mu_i^*\left(t_{i,k}\right)\right\|^2. \tag{24}$$

When the triggering condition $\|e_{i,k}\|^2 \leqslant \frac{1}{\bar{\lambda}(R_{ii})P^2}\left(\left(1-\alpha^2\right)\underline{\lambda}\left(Q_{ii}\right)\left\|\varsigma_i\left(t\right)\right\|^2 + \underline{\lambda}\left(R_{ii}\right)\left\|\mu_i^*\left(t_{i,k}\right)\right\|^2\right)$ holds, it results in $\lim_{t\to\infty} V_i^*\left(\varsigma_i\left(t\right)\right) = 0$. Therefore, it can be demonstrated that MASs (1) can reach the optimal consensus by the Lyapunov theorem.

In addition, for $t \in [t_{i,k}, t_{i,k+1})$, by Lemma 2, one has

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\left\|e_{i,k}\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|}\right) &\leqslant \left(1 + \frac{\left\|e_{i,k}\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|}\right)\cdot\frac{\left\|\dot{\varsigma}_i\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|} \\
&\leqslant \left(1 + \frac{\left\|e_{i,k}\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|}\right)\cdot\frac{L_f\left\|e_{i,k}\left(t\right)\right\| + L_f\left\|\varsigma_i\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|} \\
&= L_f\left(1 + \frac{\left\|e_{i,k}\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|}\right)^2.
\end{aligned} \tag{25}$$

From [35, Th.III.1], one can get

$$\frac{\left\|e_{i,k}\left(t\right)\right\|}{\left\|\varsigma_i\left(t\right)\right\|} \leqslant \frac{\left(t - t_{i,k}\right)L_f}{1 - \left(t - t_{i,k}\right)L_f}. \tag{26}$$

For (22), replacing t with $t_{i,k+1}$, it yields that

$$\begin{aligned}
\left\|e_{i,k+1}\left(t_{i,k+1}\right)\right\|^2 &= \frac{\left(1-\alpha^2\right)\underline{\lambda}\left(Q_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}\left\|\varsigma_i\left(t_{i,k+1}\right)\right\|^2 + \frac{\underline{\lambda}\left(R_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}\left\|\mu_i^*\left(t_{i,k}\right)\right\|^2 \\
&\geqslant \frac{\left(1-\alpha^2\right)\underline{\lambda}\left(Q_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}\left\|\varsigma_i\left(t_{i,k+1}\right)\right\|^2.
\end{aligned} \tag{27}$$

Algorithm 1 ET policy iteration algorithm.

1: Initialization: For each i, select a set of admissible control policies μ_i^0 and an appropriate threshold ε;
2: Policy evaluation: $l = l + 1$, taking into account the N-tuple of ET control policies $\hat{\mu}_1^l, \hat{\mu}_2^l, \ldots, \hat{\mu}_N^l$ to solve V_i^l using (9);

$$H\left(\varsigma_i(t), V_{\hat{\varsigma}_i}^l, \hat{\mu}_i^l, \hat{\mu}_{N_i}^l\right) = 0; \tag{30}$$

3: Policy improvement: The N-tuple of control policies can be updated by utilizing

$$\hat{\mu}_i^{l+1} = -\left(b_i + d_i\right) R_{ii}^{-1} B_i V_{\hat{\varsigma}_i}^l; \tag{31}$$

4: If $\|V_i^{l+1} - V_i^l\| \leqslant \varepsilon$, break the iteration; otherwise, return to 1 and continue.

Combining (26) with (27), we get

$$\xi_i = \sqrt{\frac{\left(1-\alpha^2\right)\underline{\lambda}\left(Q_{ii}\right)}{\bar{\lambda}\left(R_{ii}\right)P^2}} \leqslant \frac{\left\|e_{i,k+1}\left(t_{i,k+1}\right)\right\|}{\left\|\varsigma_i\left(t_{i,k+1}\right)\right\|} \leqslant \frac{\left(t_{i,k+1}-t_{i,k}\right)L_f}{1-\left(t_{i,k+1}-t_{i,k}\right)L_f}. \tag{28}$$

Then, we obtain

$$t_{i,k+1} - t_{i,k} \geqslant \frac{\xi_i}{L_f\left(\xi_i+1\right)} > 0. \tag{29}$$

Furthermore, Zeno behavior for any agent is excluded. The proof is complete.

Although the ET-based controller is used to reduce the computational effort, obtaining an analytical solution to the coupled HJB equation for the unknown system poses a challenge. Therefore, an ET policy iteration algorithm (Algorithm 1) is proposed as a solution to the coupled HJB equation.

Theorem 2. For agent i, $i \in \{1, 2, \ldots, N\}$, assume that the control policy is updated by Algorithm 1. The control policy and value function convergence to the optimum as $l \to \infty$, then

$$(1)\ \lim_{l\to\infty}\hat{\mu}_i^l = \hat{\mu}_i^*, (2)\ \lim_{l\to\infty} V_i^l = V_i^*.$$

Proof. In Algorithm 1, we use a distributed asynchronous update model. When an event is triggered for agent i, its control policy is updated and a zero-order-hold is maintained until the next triggering instant.

By (9),

$$\dot{V}_i^l = -\frac{1}{2}\left(\varsigma_i^{\mathrm{T}} Q_{ii}\varsigma_i + \left(\hat{\mu}_i^l\right)^{\mathrm{T}} R_{ii}\hat{\mu}_i^l + \sum_{j\in N_i}\left(\hat{\mu}_j^{l'}\right)^{\mathrm{T}} R_{ij}\hat{\mu}_j^{l'}\right), \tag{32}$$

$$\dot{V}_i^{l+1} = -\frac{1}{2}\left(\varsigma_i^{\mathrm{T}} Q_{ii}\varsigma_i + \left(\hat{\mu}_i^{l+1}\right)^{\mathrm{T}} R_{ii}\hat{\mu}_i^{l+1}\right) + \left(-\frac{1}{2}\sum_{j\in N_i}\left(\hat{\mu}_j^{l'}\right)^{\mathrm{T}} R_{ij}\hat{\mu}_j^{l'}\right). \tag{33}$$

Combining (32) with (33), one gets

$$\begin{aligned}\dot{V}_i^l - \dot{V}_i^{l+1} &= \frac{1}{2}\left(\left(\hat{\mu}_i^{l+1}\right)^{\mathrm{T}} R_{ii}\hat{\mu}_i^{l+1} - \left(\hat{\mu}_i^l\right)^{\mathrm{T}} R_{ii}\hat{\mu}_i^l\right)\\ &= -\frac{1}{2}\left(\hat{\mu}_i^{l+1} - \hat{\mu}_i^l\right)^{\mathrm{T}} R_{ii}\left(\hat{\mu}_i^{l+1} - \hat{\mu}_i^l\right) + \left(\hat{\mu}_i^{l+1}\right)^{\mathrm{T}} R_{ii}\left(\hat{\mu}_i^{l+1} - \hat{\mu}_i^l\right)\\ &= -\frac{1}{2}\left(\tilde{\hat{\mu}}_i\right)^{\mathrm{T}} R_{ii}\tilde{\hat{\mu}}_i + \left(\hat{\mu}_i^{l+1}\right)^{\mathrm{T}} R_{ii}\tilde{\hat{\mu}}_i.\end{aligned} \tag{34}$$

For $\dot{V}_i^l - \dot{V}_i^{l+1} \leqslant 0$, a sufficient condition is

$$\begin{aligned}\left(\tilde{\hat{\mu}}_i\right)^{\mathrm{T}} R_{ii}\tilde{\hat{\mu}}_i &\geqslant 2\left(\hat{\mu}_i^{l+1}\right)^{\mathrm{T}} R_{ii}\tilde{\hat{\mu}}_i\\ &= -2\left(b_i + d_i\right)\left(V_{\varsigma_i}^l\right)^{\mathrm{T}} B_i^{\mathrm{T}}\tilde{\hat{\mu}}_i,\end{aligned} \tag{35}$$

where $\tilde{\hat{\mu}}_i = \hat{\mu}_i^{l+1} - \hat{\mu}_i^l$.

For (35) to be valid, we require

$$\underline{\lambda}\left(R_{ii}\right)\left\|\tilde{\hat{\mu}}_i\right\| \geqslant 2\bar{\lambda}\left(B_i\right)\left(b_i+d_i\right)\left\|V_{\varsigma_i}^l\right\|. \tag{36}$$

By Definition 2 and (8), it is known that $V_i(\infty)=0$. Then, integrating both sides of the inequality $\dot{V}_i^l-\dot{V}_i^{l+1}\leqslant 0$ over the interval $[t,\infty)$, we obtain

$$\int_t^{+\infty}\left(\dot{V}_i^l-\dot{V}_i^{l+1}\right)\mathrm{d}\tau=V_i^{l+1}-V_i^l\leqslant 0. \tag{37}$$

According to (37), we conclude that the value function V_i^l is no-increasingly bounded by below 0. Thus, we have

$$V_i^{\infty}\leqslant V_i^*. \tag{38}$$

In addition, by (10), it can be obtained that $V_i^l\geqslant V_i^*$. Letting $l\to\infty$, we have

$$V_i^{\infty}\geqslant V_i^*. \tag{39}$$

Combining (38) with (39), we can obtain $\lim_{l\to\infty}V_i^l=V_i^*$. Meanwhile, according to (31), we can easily note that $\lim_{l\to\infty}\hat{\mu}_i^l=\hat{\mu}_i^*$. The proof is complete.

Remark 3. In [30,32], the ET condition was only related to current data, because of which a satisfactory balance could not be established between conservation of communication resources and optimization of system performance. Using (13), the ET-PER method in this study selects important historical data in order of priority through the PER technique. This can effectively establish a balance between system performance and communication resource conservation.

4 Online AC NN implementation using ET-PER

In this section, aiming at solving the coupled HJB equation (12), we use a dual NN based on the ET-PER method to approximate the value function and help optimize the control policy. Compared with the AC dual NNs in [15, 17], the weight estimates for the NN are updated at the moment of event trigger, and the frequency of system update is relatively low, thus reducing the amount of calculation.

4.1 Critic network

The optimal value function is approximated by the critic network using the ET-PER method,

$$V_i^*\left(\varsigma_i(t)\right)=W_{ci}^{*\mathrm{T}}\varphi_i\left(\phi_{ci}(t)\right)+\delta_{ci}\left(\varsigma_i\right), \tag{40}$$

where $W_{ci}^*\in\mathbb{R}^{N_c}$ is the unknown target weight matrix for critic NNs, $\varphi_i=\tanh(\cdot)$ is the activation function, $\phi_{ci}(t)\in\mathbb{R}^{N_c}$ contains input information $\left(\varsigma_i(t),\mu_i,\mu_{N_i}\right)$, $\delta_{ci}\left(\varsigma_i\right)$ is based on critic network reconstruction error, and N_c is the number of the hidden nodes.

Since W_{ci}^* is unknown, assuming $\hat{W}_{ci}$ is the estimation of W_{ci}^*, then the critic NN's actual output can be approximated as

$$\hat{V}_{i,k}\left(\varsigma_i(t)\right)=\hat{W}_{ci,k}^{\mathrm{T}}\varphi_i\left(\phi_{ci}(t)\right), \tag{41}$$

where $\hat{V}_{i,k}$ and $\hat{W}_{ci,k}^{\mathrm{T}}$ are estimates of V_i^* and $W_{ci}^{*\mathrm{T}}$ at the triggering instant $t_{i,k}$, respectively.

According to (12), we known $H_i\left(\varsigma_i,V_{\varsigma_i}^*,\mu_i^*\right)=0$, then the error function can be defined as

$$\begin{aligned}e_{ci,k}=&H_i\left(\varsigma_i,\hat{W}_{ci,k},\hat{\mu}_i\right)-H_i\left(\varsigma_i,V_{\varsigma_i}^*,\mu_i^*\right)\\=&\frac{1}{2}\left(\varsigma_i^{\mathrm{T}}(t)Q_{ii}\varsigma_i(t)+\mu_i^{\mathrm{T}}(t)R_{ii}\mu_i(t)\right)+\frac{1}{2}\sum_{j\in N_i}\mu_j^{\mathrm{T}}(t)R_{ij}\mu_j(t)+\hat{V}_{\varsigma_i}^{\mathrm{T}}\dot{\varsigma}_i(t),\end{aligned} \tag{42}$$

where $\hat{V}_{\varsigma_i}^{\mathrm{T}}\dot{\varsigma}_i(t)=\hat{W}_{ci,k}^{\mathrm{T}}\nabla\varphi_i\left(\phi_{ci}(t)\right)\cdot\left(\varsigma_i^{\mathrm{T}}(t)Q_{ii}\varsigma_i(t)+\mu_i^{\mathrm{T}}(t)R_{ii}\mu_i(t)+\sum_{j\in N_i}\mu_j^{\mathrm{T}}(t)R_{ij}\mu_j(t)\right)$ and $e_{ci,k}$ is the HJB equation error.

For agent i, the goal of the critic network is to find a suitable $\hat{W}_{ci,k}$ that minimizes the squared error function as below:

$$E_{ci,k} = \frac{1}{2} e_{ci,k}^2. \tag{43}$$

The critic NN weights will only be adjusted during ET instants under the ET sampling mechanism. The following is the critic NN weight estimation updating law:

$$\begin{cases} \dot{\hat{W}}_{ci,k} = 0, \ t \in (t_{i,k}, t_{i,k+1}), \\ \hat{W}_{ci,k}^{+} = \hat{W}_{ci,k} - \alpha_{ci} \rho_i \left(\rho_{i1}^{\mathrm{T}} \hat{W}_{ci,k} + \Re_i (\varsigma_i, \mu_i, \mu_{N_i}) \right), \ t = t_{i,k}, \end{cases} \tag{44}$$

where $\alpha_{ci} > 0$ is the learning rate, $\rho_{i1} = \nabla \varphi_i (\phi_{ci}(t)) \cdot f_i (\varsigma_i(t), \mu_i(t))$, and $\rho_i = \frac{\rho_{i1}}{(1+\rho_{i1}^{\mathrm{T}} \rho_{i1})^2}$.

4.2 Actor network

An approximation of the controller can be achieved through the usage of the actor network, whereby the actor network output is defined as

$$\hat{\mu}_i (\varsigma_i (t_{i,k})) = \hat{W}_{ai,k}^{\mathrm{T}} \phi_{ai} (\varsigma_i (t_{i,k})), \tag{45}$$

where $\hat{W}_{ai,k} \in \mathbb{R}^{N_a}$ is the unknown target weight matrix for actor network and $\mathbb{R}^{N_a}$ is the number of the hidden nodes. The error function of the actor network can be defined as

$$e_{ai,k} = \hat{V}_i (\varsigma_i (t_{i,k})) - U. \tag{46}$$

When the system achieves consensus, the control policy does not need to be updated in conjunction with the reinforcement learning algorithm, and U can be set to 0. The objective of an actor network, akin to that of a critic network, is to ascertain a suitable $\hat{W}_{ai,k}$ that minimizes the function of squared error, represented as follows:

$$E_{ai,k} = \frac{1}{2} e_{ai,k}^2. \tag{47}$$

In the ET sampling mechanism, the actor network weights will only be adjusted during ET instants. The actor NN weight estimation is updated by the following law:

$$\begin{cases} \dot{\hat{W}}_{ai,k} = 0, \ t \in (t_{i,k}, t_{i,k+1}), \\ \hat{W}_{ai,k}^{+} = \hat{W}_{ai,k} - \alpha_{ai} \frac{\partial E_{ai,k}}{\partial \hat{W}_{ai,k}}, \ t = t_{i,k}, \end{cases} \tag{48}$$

where $\frac{\partial E_{ai,k}}{\partial \hat{W}_{ai,k}} = \phi_{ci} (\varsigma_i (t_{i,k})) \cdot \hat{W}_{ci,k}^{\mathrm{T}} \nabla \varphi_i (\phi_{ci}(t)) \cdot \frac{\partial \phi_{ci}(t)}{\partial \hat{\mu}_i (\varsigma_i (t_{i,k}))} \cdot (\hat{W}_{ci,k}^{\mathrm{T}} \nabla \varphi_i (\phi_{ci}(t)))^{\mathrm{T}}$ and $\alpha_{ai} > 0$ represents learning rate.

4.3 Prioritized experience replay

In RL, agents interact with the environment and collect experiential data. These data often exhibit temporal dependencies, meaning that transitions from one state to another are continuous, and the current state is often influenced by preceding ones. This temporal dependency results in strong correlations between the data. When training RL models on highly correlated data, the models might fail to accurately capture the true distribution of the data. This can result in biases in the prediction of future states, thereby affecting the decision-making accuracy. The stochastic gradient algorithm typically assumes that input data are independently and identically distributed. However, in RL, this assumption often does not hold true because of strong correlations between data. This can result in bias in parameter updates, affecting the model's performance. The existing scholars have used various methods to break correlations between data.

For instance, Refs. [17,38] adopted the ER strategy. Although this approach, to some extent, reduced the correlation between data, it could not fully and efficiently exploit the data. This is because the agent's experiences were derived from previously collected data, but the values of these data were not uniform during the training process. In particular, the learning efficiency and effectiveness of the agent were higher in certain states than others, and traditional ER strategies failed to adequately account for

Algorithm 2 Optimal consensus of MASs via ET-PER method.

Step 1. Initialization: For each i, set the capacity parameter for the experience buffer $D_M = M$ to 200, iterative index $l = 0$;
Step 2. **If** ET occurred,
(i) Store the sampled disagreement error $\varsigma_i(t_{i,k})$ to the experience buffer D_M, with its initial priority set to the priority with the highest value taken in the experience buffer to ensure it is used in later updates;
(ii) For the i-th stored element, when $i > M$, remove the oldest element;
(iii) Data $\varsigma_i(t_{i,s})$ are taken from the experience buffer D_M with probability $P(i)$ (see (49)) and a new weighted error $\hat{\varsigma}_i(t_{i,k})$ is calculated from this data and the sampled disagreement error $\varsigma_i(t_{i,k})$ weighted using an importance sampling factor α_i (see (51));
(iv) Update the AC NN parameters;
(v) Update the priority of the selected error in the experience buffer by calculating the priority based on the temporal-difference error and the mapping function;
else
(vi) NN weights are not updated;
Step 3. Return to (2) until the weights converge.

this, thus failing to maximize the use of valuable information within these data. The essential idea of PER is to break up uniform sampling and assign a higher sampling weight to states that learn more efficiently. Now, the question arises "How to choose the weights?" An ideal criterion is that the more efficiently the agent learns, the higher the weight. We can employ the temporal-difference error to specify the degree of learning priority. The larger the error, the more the scope for improvement in prediction accuracy, and thus, the sample must be learned, i.e., the higher the learning priority.

In this part, PER is introduced to select effective historical data using the priority mechanism in PER. Essentially, PER functions by assigning varying priorities to individual transitions within the experience buffer, enabling more efficient and effective learning. Moreover, the purpose of employing the prioritization mechanism is to achieve an enhanced balance between communication resource usage and system performance. This mechanism selectively identifies high-priority historical data to establish ET conditions. Subsequently, during the sampling process, data holding a higher priority have a higher probability of being selected, and this probability can be mathematically expressed as

$$P(i) = \frac{p_i}{\sum_k p_k}, \tag{49}$$

where $p_i = \psi(\cdot)$ is the priority of the i-th transition in the experience buffer, and $\psi(\cdot)$ is the mapping function. PER uses non-uniform sampling, which inevitably alters the distribution of the original data. The data in the experience buffer is sampled with probability $P(i)$, and different samples have different sampling probabilities.

The temporal-difference error algorithm uses the following "stochastic gradient descent" method to update the NNs parameters:

$$W_{\text{new}} \leftarrow W_{\text{new}} - \alpha \cdot \boldsymbol{g}, \tag{50}$$

where α is the learning rate, $\boldsymbol{g}$ represents the gradient of the loss function with respect to W_{new}.

PER employs non-uniform sampling, which inevitably alters the distribution of the original data, potentially yielding different results than expected. Then, the learning rate α should be adjusted in accordance with the sampling probability. If the probability of a sample being sampled is large, its learning rate should be relatively small, then the learning rate is adjusted as follows, i.e., the importance-sampling weights:

$$\alpha_i = \frac{\alpha}{(M \cdot P(i))^{\beta}}, \tag{51}$$

where M denotes the total number of samples stored in the experience replay array, and $\beta \in (0,1)$ is a super parameter. The importance-sampling weights influence the NNs weights by merging into (49), which is modified as

$$\begin{cases} \dot{\hat{W}}_{ci,k} = 0, \ t \in (t_{i,k}, t_{i,k+1}), \\ \hat{W}^{+}_{ci,k} = \hat{W}_{ci,k} - \alpha_i \cdot \alpha_{ci} \rho_i (\rho_{i1}^{\mathrm{T}} \hat{W}_{ci,k} + \Re_i(\varsigma_i, \mu_i, \mu_{N_i})), \ t = t_{i,k}. \end{cases} \tag{52}$$

The framework of the AC NN implementation of the ET-PER method is shown in Algorithm 2.

Remark 4. In our approach, the implementation of PER necessitates the upkeep of a SumTree structure. Each leaf in the SumTree stores the sample's priority p_i; each branch node has exactly two branches,

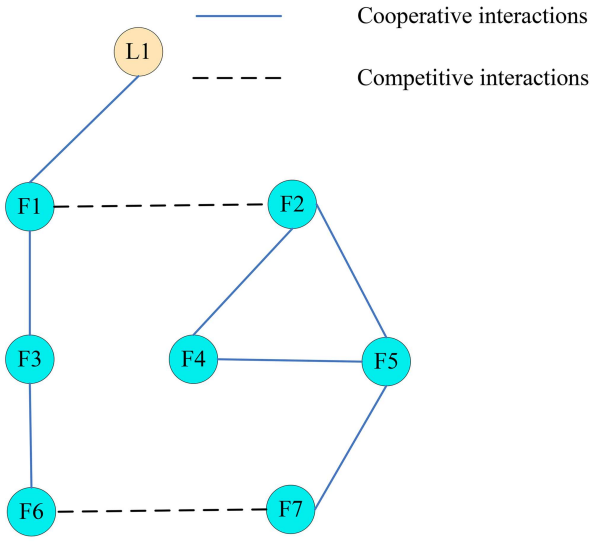

Figure 1 (Color online) Communication topology of 8 generators.

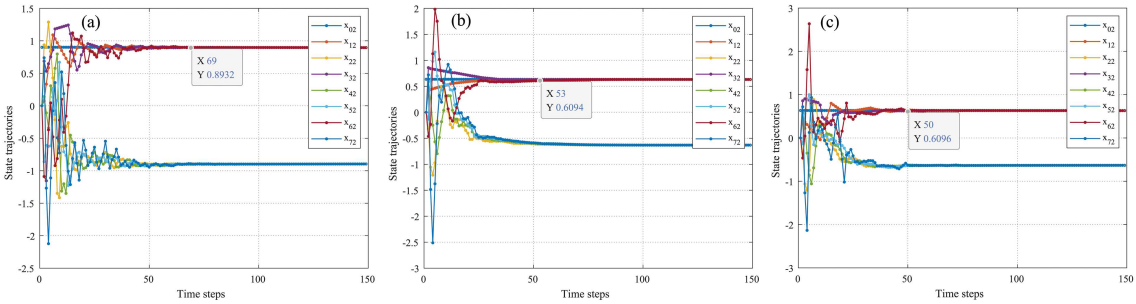

Figure 2 (Color online) Evolution trajectories of states. (a) Traditional ET-based algorithm; (b) ET-ER-based algorithm; (c) ET-PER-based algorithm.

and the node value is the summation of the two branches. Obviously, the top of the SumTree is the sum of all p_i values. During each iteration, upon drawing m samples, the total priority of the experience buffer samples is partitioned into m distinct segments. Subsequently, a random value is selected from each segment, and the respective data are retrieved from the SumTree using these randomly generated values. While the aforementioned approach necessitates additional memory allocation, it avoids the need for sorting p_i values at each sample, decreasing the time complexity of finding the data.

5 Simulation example

This section includes a numerical example demonstrating the effectiveness of the proposed ET-PER-based optimal control policy.

Let us consider MASs comprising one leader and seven followers. Figure 1 shows the cooperative-competitive network associated with the MASs. From Figure 1, one can see that the pinning gains and cooperative-competitive weights between the leader and followers are $b_1 = 1$, $a_{21} = a_{12} = a_{67} = a_{76} = -1$, and $a_{25} = a_{52} = a_{13} = a_{31} = a_{36} = a_{63} = a_{24} = a_{42} = a_{45} = a_{54} = a_{57} = a_{75} = 1$, respectively. The dimension-appropriate unit matrix is selected as the weighting matrix for the performance index function.

For the sake of generality, we chose some data at random to establish experimental parameters, then defined the dynamic and feedback gain matrices as follows: $A = \left[\begin{smallmatrix} 0.532 & 0.0335 \\ 0.00076 & 0.9999 \end{smallmatrix}\right]$, $B_1 = [0.017\ 0.36]^{\mathrm{T}}$, $B_2 = [0.025\ 0.29]^{\mathrm{T}}$, $B_3 = [0.042\ 0.32]^{\mathrm{T}}$, $B_4 = [0.027\ 0.41]^{\mathrm{T}}$, $B_5 = [0.033\ 0.40]^{\mathrm{T}}$ $B_6 = [0.021\ 0.30]^{\mathrm{T}}$, and $B_7 = [0.032\ 0.40]^{\mathrm{T}}$. The learning rate is $\alpha_{ai} = \alpha_{ci} = 0.01$. In the parameters setting of PER, $\beta = 0.5$, the capacity of the experience buffer is set to 200 and the priority mapping function $\psi(p_i) = |p_i|$.

Figures 2 and 3 depict the evolutionary trajectories of states and controls using the traditional ET-

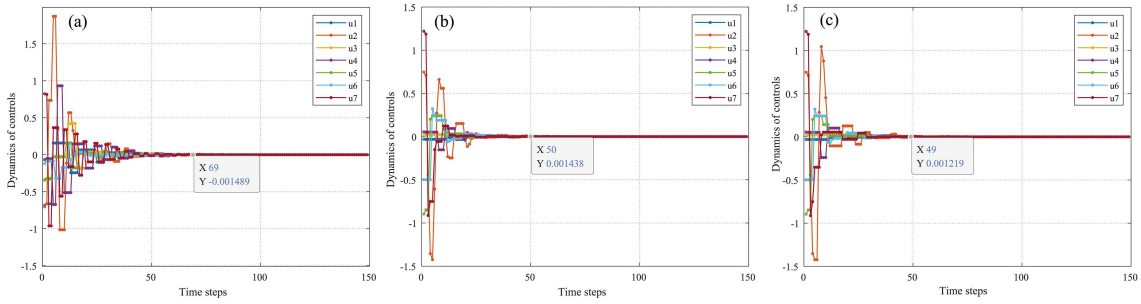

Figure 3 (Color online) Evolution trajectories of controls. (a) Traditional ET-based algorithm; (b) ET-ER-based algorithm; (c) ET-PER-based algorithm.

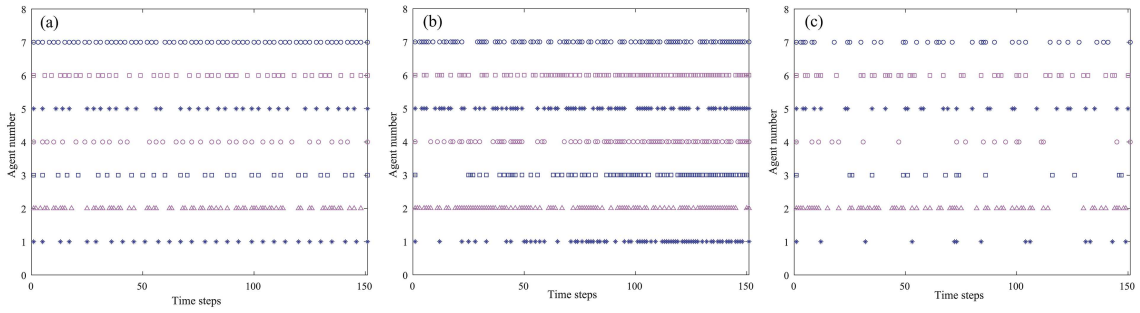

Figure 4 (Color online) Trigger moments of (a) the traditional ET-based algorithm, (b) the ET-ER-based algorithm, and (c) the ET-PER-based algorithm.

based, ET-ER-based, and ET-PER-based algorithms, respectively. From Figures 2 and 3, it can be observed that the convergence of agent states and controls under the traditional ET algorithm is slower compared to the other two algorithms. The reason for this difference lies in the fact that the algorithms based on ET-ER and ET-PER utilize the historical data of the system, enabling them to focus more on system performance, thereby enhancing the convergence rate.

Figure 4 illustrates the number of event triggers in MASs under the traditional ET-based algorithm, the ET-ER-based algorithm, and the ET-PER-based algorithm, respectively.

To intuitively illustrate the communication resource conservation benefits of the proposed algorithm (ET-PER-based algorithm), we have plotted a bar chart comparing the number of triggers during the convergence process for seven agents. In Figure 5, the green bars represent the number of triggers under the traditional ET-based algorithm, the blue bars represent the number of triggers under the ET-ER-based algorithm, and the orange bars represent the number of triggers under the proposed algorithm. It is evident from Figure 5 that the proposed algorithm results in the lowest number of triggers among all agents, followed by the traditional ET-based algorithm, while the ET-ER-based algorithm results in relatively higher trigger counts. This is mainly because, although the ET-ER-based algorithm utilizes historical data, it does not prioritize this data, leading to an increased number of triggers. Through comparative experiments, the advantage of the ET-PER-based algorithm in conserving communication resources is validated.

Figure 6 illustrates the energy consumption of agents during the consensus process under the traditional ET-based algorithm, the ET-ER-based algorithm, and the ET-PER-based algorithm, respectively.

To clearly demonstrate the advantages of the proposed algorithm in terms of system performance, we have created a pie chart to compare the total energy consumption of MASs under different algorithms. As shown in Figure 7, the green section represents the total energy consumption for the agents to reach consensus under the traditional ET-based algorithm, the blue section represents the total energy consumption under the ET-ER-based algorithm, and the orange section represents the total energy consumption under the ET-PER-based algorithm. It is evident from Figure 7 that the ET-PER-based algorithm performs the best in terms of total energy consumption, followed by the ET-ER-based algorithm, with the traditional ET-based algorithm having relatively higher total energy consumption. Specifically, compared to the total energy consumption under the traditional ET-based algorithm, the proposed algorithm with an integrated prioritized experience replay strategy improves performance by approximately 50%.

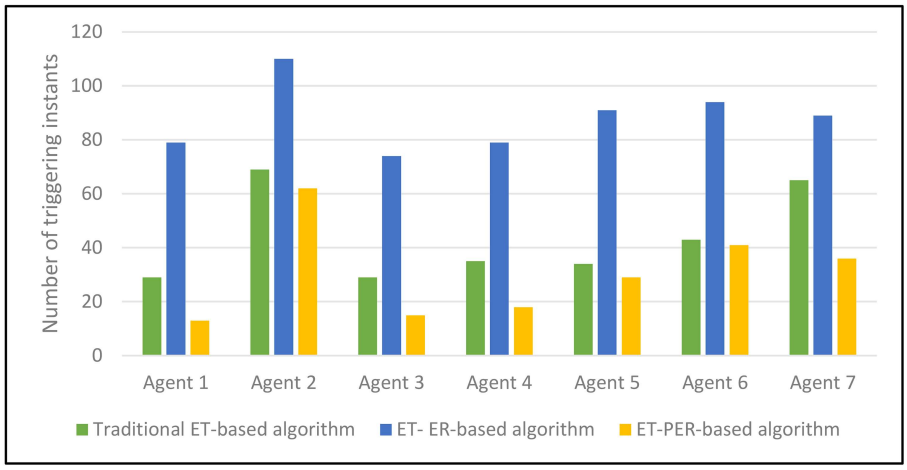

Figure 5 (Color online) Comparison of the number of sampled data is presented among the traditional ET-based algorithm, the ET-ER-based algorithm, and the ET-PER-based algorithm for each agent.

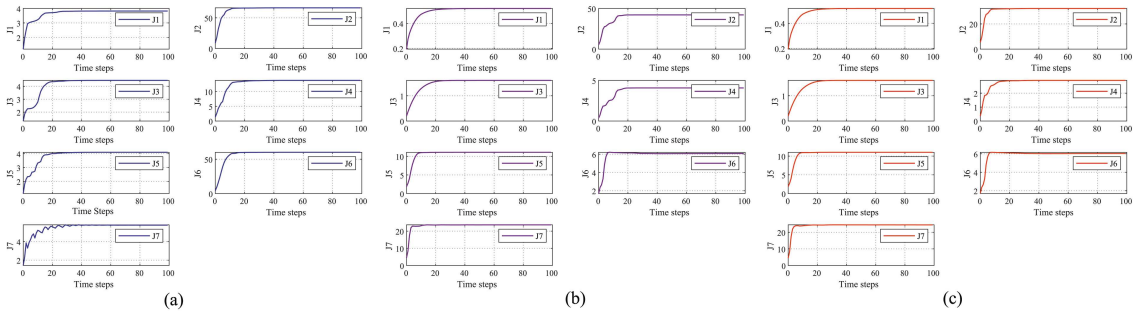

Figure 6 (Color online) Energy consumption using (a) the traditional ET-based algorithm, (b) the ET-ER-based algorithm, and (c) the ET-PER-based algorithm.

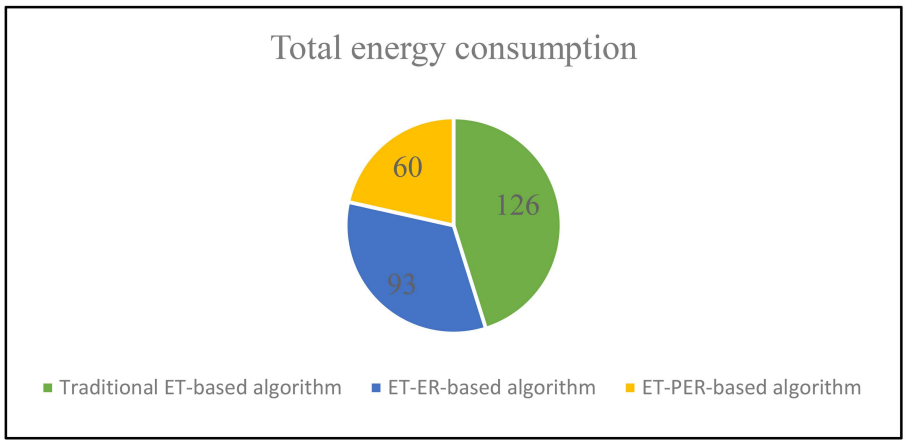

Figure 7 (Color online) Comparison of the total energy consumption is presented among the traditional ET-based algorithm, the ET-ER-based algorithm, and the ET-PER-based algorithm for MASs.

From the abovementioned analysis, it is evident that for the same NN structure and parameters, the proposed algorithm offers notable advantages versus the traditional ET-based algorithm across several key performance metrics. In particular, by integrating a priority-based ET mechanism, we can effectively reduce unnecessary communication, thereby resulting in increased convergence speed and reduced total energy consumption. However, compared with the ET-ER-based algorithm, while our algorithm also excels in communication resource conservation and energy consumption reduction, it does not considerably increase the convergence speed. This may be due to the current algorithm's design, which still requires refinement to more effectively balance communication resource conservation and convergence

speed increase.

6 Conclusion

We proposed the ET-PER-based algorithm, which solved the problem of optimal consensus control for cooperative-competitive MASs. As per the Lyapunov stability analysis, the estimated optimal control policies achieved MAS consensus and excluded Zeno behavior. The simulation results showed that the proposed ET-PER-based algorithm offered notable advantages in the cooperative optimization of MASs. Compared with the traditional ET-based algorithm, the proposed algorithm not only increased the convergence speed but also considerably reduced communication resource usage and energy consumption. However, in comparison with the ET-ER-based algorithm, while the proposed algorithm did conserve communication resources and reduced energy consumption, the convergence speed could not be substantially increased.

Results of further analysis revealed that although the ET-PER-based algorithm frequently selects sparse yet important samples for training, its advantage in using these samples could not be fully realized. This could be due to the failure of the current priority mapping function to fully capture the importance and sparsity of the samples, thereby limiting the performance of the ET-PER algorithm in practical applications. Furthermore, this study did not thoroughly explore the impact of cooperative and competitive strengths on the performance of MASs. In practical applications, the cooperative and competitive relationships among agents considerably affect the overall system performance. Therefore, we plan to (1) further investigate the mechanism by which cooperative and competitive strengths affect system performance and (2) design a priority mapping function that incorporates cooperative and competitive strengths to dynamically adjust priorities. Through this improvement, we expect to more effectively leverage the advantages of the PER technology and enhance the overall performance of MASs in cooperative optimization.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (Grant Nos. 62276036, 62221005, 62006031), Major Project of Scientific and Technological Research Program of Chongqing Municipal Education Commission (Grant No. KJZD-M202100602), Project of Natural Science Foundation of Chongqing (Grant Nos. cstc2021jcyj-msxmX1043, CSTB2024NSCQ-LZX0118), Anhui Provincial Research Programming Project (Grant Nos. 2022AH051039, 2022AH051054), and Doctoral Talent Training Project of Chongqing University of Posts and Telecommunications (Grant No. BYJS202210).

References

1 Wen G X, Chen C L P, Dou H, et al. Formation control with obstacle avoidance of second-order multi-agent systems under directed communication topology. Sci China Inf Sci, 2019, 62: 192205
2 Lu Y, Wen C, Shen T, et al. Bearing-based adaptive neural formation scaling control for autonomous surface vehicles with uncertainties and input saturation. IEEE Trans Neural Netw Learn Syst, 2020, 32: 4653–4664
3 Dai H, Jia J, Yan L, et al. Distributed fixed-time optimization in economic dispatch over directed networks. IEEE Trans Ind Inf, 2020, 17: 3011–3019
4 Li Q, Gao D W, Zhang H, et al. Consensus-based distributed economic dispatch control method in power systems. IEEE Trans Smart Grid, 2017, 10: 941–954
5 Wang J, Wang C, Wei Y, et al. Neuroadaptive sliding mode formation control of autonomous underwater vehicles with uncertain dynamics. IEEE Syst J, 2019, 14: 3325–3333
6 Qian Z, Lyu W, Dai Y, et al. A consensus-based model predictive control with optimized line-of-sight guidance for formation trajectory tracking of autonomous underwater vehicles. J Intell Robot Syst, 2022, 106: 15
7 Zhu J, Lu J, Yu X. Flocking of multi-agent non-holonomic systems with proximity graphs. IEEE Trans Circuits Syst I, 2012, 60: 199–210
8 Lewis F L, Vrabie D, Syrmos V L. Optimal Control. Hoboken: John Wiley and Sons, 2012
9 Werbos P J. Backpropagation through time: what it does and how to do it. Proc IEEE, 1990, 78: 1550–1560
10 Huang Z, Li Y, Zhang C, et al. A data-driven approximate solution to the model-free HJB equation. Optim Control Appl Methods, 2018, 39: 835–844
11 Li X X, Peng Z H, Jiao L, et al. Online adaptive Q-learning method for fully cooperative linear quadratic dynamic games. Sci China Inf Sci, 2019, 62: 222201
12 Zhang H G, Luo Y H, Liu D R. Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints. IEEE Trans Neural Netw, 2009, 20: 1490–1503
13 Peng Z, Luo R, Hu J, et al. Optimal tracking control of nonlinear multiagent systems using internal reinforce Q-learning. IEEE Trans Neural Netw Learn Syst, 2021, 33: 4043–4055
14 Wen G, Chen C L P, Ge S S, et al. Optimized adaptive nonlinear tracking control using actor-critic reinforcement learning strategy. IEEE Trans Ind Inf, 2019, 15: 4969–4977
15 Zhang C, Ji L, Yang S, et al. Optimal antisynchronization control for unknown multiagent systems with deep deterministic policy gradient approach. Inf Sci, 2023, 622: 946–961
16 Sutton R S, Barto A G. Reinforcement Learning: An Introduction. Cambridge: MIT Press, 2018
17 Zhang H, Jiang H, Luo Y, et al. Data-driven optimal consensus control for discrete-time multi-agent systems with unknown dynamics using reinforcement learning method. IEEE Trans Ind Electron, 2016, 64: 4091–4100
18 Peng Z, Zhao Y, Hu J, et al. Data-driven optimal tracking control of discrete-time multi-agent systems with two-stage policy iteration algorithm. Inf Sci, 2019, 481: 189–202
19 Xu W K, Wang L, Sun S W, et al. A novel policy iteration algorithm for solving the optimal consensus control problem of a discrete-time multiagent system with unknown dynamics. Sci China Inf Sci, 2023, 66: 189204

20 Wen G, Li B. Optimized leader-follower consensus control using reinforcement learning for a class of second-order nonlinear multiagent systems. IEEE Trans Syst Man Cybern Syst, 2021, 52: 5546–5555
21 Xu J, Wang L, Liu Y, et al. Finite-time adaptive optimal consensus control for multi-agent systems subject to time-varying output constraints. Appl Math Computation, 2022, 427: 127176
22 Xu S, Cao J, Liu Q, et al. Optimal control on finite-time consensus of the leader-following stochastic multiagent system with heuristic method. IEEE Trans Syst Man Cybern Syst, 2019, 51: 3617–3628
23 Wang P, Yu C, Lv M, et al. Adaptive fixed-time optimal formation control for uncertain nonlinear multiagent systems using reinforcement learning. IEEE Trans Netw Sci Eng, 2024, 11: 1729–1743
24 Zhang J, Fu Y, Fu J. Adaptive finite-time optimal formation control for second-order nonlinear multiagent systems. IEEE Trans Syst Man Cybern Syst, 2023, 53: 6132–6144
25 Luo B, Yang Y, Liu D. Adaptive Q-learning for data-based optimal output regulation with experience replay. IEEE Trans Cybern, 2018, 48: 3337–3348
26 Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay. 2015. ArXiv:1511.05952
27 Li T, Yang D, Xie X. Prioritized experience replay based reinforcement learning for adaptive tracking control of autonomous underwater vehicle. Appl Math Computation, 2023, 443: 127734
28 Yuan W, Li Y, Zhuang H, et al. Prioritized experience replay-based deep Q learning: multiple-reward architecture for highway driving decision making. IEEE Robot Automat Mag, 2021, 28: 21–31
29 Liu P, Ma X, Ding J, et al. Multi-agent collaborative path planning algorithm with reinforcement learning and combined prioritized experience replay in Internet of Things. Comput Electrical Eng, 2024, 116: 109193
30 Ni Z, Malla N, Zhong X. Prioritizing useful experience replay for heuristic dynamic programming-based learning systems. IEEE Trans Cybern, 2018, 49: 3911–3922
31 Ji L, Lin Z, Zhang C, et al. Data-based optimal consensus control for multiagent systems with time delays: using prioritized experience replay. IEEE Trans Syst Man Cybern Syst, 2024, 54: 3244–3256
32 Heemels W P M H, Donkers M C F, Teel A R. Periodic event-triggered control for linear systems. IEEE Trans Automat Contr, 2012, 58: 847–861
33 Fan S, Yan H, Zhang H, et al. Dynamic event-based non-fragile dissipative state estimation for quantized complex networks with fading measurements and its application. IEEE Trans Circ Syst I, 2020, 68: 856–867
34 Wu Y, Zhang H, Wang Z, et al. Output consensus of heterogeneous linear multiagent systems with directed graphs via adaptive dynamic event-triggered mechanism. IEEE Trans Cybern, 2023, 53: 4606–4618
35 Tabuada P. Event-triggered real-time scheduling of stabilizing control tasks. IEEE Trans Automat Contr, 2007, 52: 1680–1685
36 Peng Z, Luo R, Hu J, et al. Distributed optimal tracking control of discrete-time multiagent systems via event-triggered reinforcement learning. IEEE Trans Circ Syst I, 2022, 69: 3689–3700
37 Lu J, Wei Q, Liu Y, et al. Event-triggered optimal parallel tracking control for discrete-time nonlinear systems. IEEE Trans Syst Man Cybern Syst, 2021, 52: 3772–3784
38 Liu C, Liu L, Cao J, et al. Intermittent event-triggered optimal leader-following consensus for nonlinear multi-agent systems via actor-critic algorithm. IEEE Trans Neural Netw Learn Syst, 2023, 34: 3992–4006
39 Zhao W, Yu W, Zhang H. Event-triggered optimal consensus tracking control for multi-agent systems with unknown internal states and disturbances. Nonlinear Anal-Hybrid Syst, 2019, 33: 227–248
40 Zhao W, Zhang H. Distributed optimal coordination control for nonlinear multi-agent systems using event-triggered adaptive dynamic programming method. ISA Trans, 2019, 91: 184–195
41 Xu B, Li Y X, Hou Z, et al. Dynamic event-triggered reinforcement learning-based consensus tracking of nonlinear multi-agent systems. IEEE Trans Circ Syst I, 2023, 70: 2120–2132
42 Chen Z, Chen K, Zhang Y. Distributed observer-based hierarchical optimal consensus tracking with dynamic event-triggered adaptive dynamic programming. Nonlinear Dyn, 2023, 111: 12319–12337
43 Li Y F, Wang X, Sun J, et al. Data-driven consensus control of fully distributed event-triggered multi-agent systems. Sci China Inf Sci, 2023, 66: 152202
44 Wang X, Sun J, Deng F, et al. Event-triggered consensus control of heterogeneous multi-agent systems: model- and data-based approaches. Sci China Inf Sci, 2023, 66: 192201
45 Jin W, Zhang H, Ming Z. Optimal bipartite consensus for discrete-time multi-agent systems with event-triggered mechanism based on adaptive dynamic programming. Neurocomputing, 2024, 564: 126965
46 Zhou W, Wang Y, Ahn C K, et al. Adaptive fuzzy backstepping-based formation control of unmanned surface vehicles with unknown model nonlinearity and actuator saturation. IEEE Trans Veh Technol, 2020, 69: 14749–14764
47 Girard A. Dynamic triggering mechanisms for event-triggered control. IEEE Trans Automat Contr, 2015, 60: 1992–1997
48 Wang K, Tian E, Liu J, et al. Resilient control of networked control systems under deception attacks: a memory-event-triggered communication scheme. Intl J Robust Nonlinear, 2020, 30: 1534–1548
49 Xie L, Cheng J, Wang H, et al. Memory-based event-triggered asynchronous control for semi-Markov switching systems. Appl Math Comput, 2022, 415: 126694
50 Yang X, Zhang H, Wang Z. Data-based optimal consensus control for multiagent systems with policy gradient reinforcement learning. IEEE Trans Neural Netw Learn Syst, 2021, 33: 3872–3883
51 Xu Y, Wang J, Xia C Y, et al. Higher-order temporal interactions promote the cooperation in the multiplayer snowdrift game. Sci China Inf Sci, 2023, 66: 222208
52 Zhu Y, Zhang Z, Xia C, et al. Equilibrium analysis and incentive-based control of the anticoordinating networked game dynamics. Automatica, 2023, 147: 110707
53 Feng F, Xu Y, Tang Z. Research on the charge rate of railway value C guaranteed transportation based on competitive and cooperative relationships. Adv Mech Eng, 2018, 10: 1–11
54 Feng F, Li W, Jiang Q. Railway traffic accident forecast based on an optimized deep auto-encoder. Promet-Zagreb, 2018, 30: 379–394
55 Zhai S, Zheng W X. On survival of all agents in a network with cooperative and competitive interactions. IEEE Trans Automat Contr, 2019, 64: 3853–3860
56 Ren W, Beard R W. Consensus seeking in multiagent systems under dynamically changing interaction topologies. IEEE Trans Automat Contr, 2005, 50: 655–661