Optimal Group Consensus Control for Multi-agent Systems in Coopetition Networks via Dynamic Event-Triggered Methods

Xi Li[1], Lianghao Ji[1](✉), Shasha Yang[1], and Yong Wang[2]

[1] Chongqing Key Laboratory of Image Cognition,
Chongqing University of Posts and Telecommunications,Chongqing 400065, China
jilh@cqupt.edu.cn

[2] Guangxi Key Laboratory of Cryptography and Information Security,
Guilin University of Electronic Technology, Guilin 541004, China

Abstract. In this paper, the optimal group consensus control of second-order multi-agent systems is investigated by using the adaptive dynamic programming (ADP) method under the event-triggered mechanism. In order to meet the needs of group consensus, a novel tracking error protocol involving coopetition interaction is proposed. On this basis, the optimal group consensus problem is established by using the Bellman optimality principle. Then, the dynamic event-triggered mechanism is introduced into the ADP method to save computing and communication resources. The stability of the system is proved by using Lyapunov stability theory. Finally, the simulation results show the effectiveness of the proposed method.

Keywords: Optimal consensus control · Second-order multi-agent systems · Dynamic event-triggered · Group consensus

1 Introduction

In recent years, the optimal control of multi-agent systems(MASs) has attracted wide attention of scholars. Note that most of the work on ADP is periodically triggered or time-triggered, which leads to the inefficient use of communication and computing resources. This problem will be improved with the introduction of event-triggered control methods. Different from the time-triggered control, the event-triggered control method means that the agent will update its control policy only when the measurement error exceeds the threshold set by the engineer [1]. This will greatly reduce the use of resources. Therefore, it is a meaningful work to combine the ADP method with the event-triggered mechanism [2–4]. Sahoo et al. [3] proposed event-based near optimal control for uncertain nonlinear discrete-time systems by using input-output data and ADP method. Wei et al. [4] use critical neural network (NN) updated at the event-triggered instants to approximate

Z. Ren et al. (Eds.): *Proceedings of 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control*, LNEE 934, pp. 134–145, 2023.
https://doi.org/10.1007/978-981-19-3998-3_14

the value function and search for the optimal control policy, resulting in aperiodic weight adjustment law to reduce the computational cost.

It should be pointed out that most of the existing work uses the static event-triggered mode, that is, the trigger instant is the time sequence in which the measurement error is equal to or exceeds the threshold [2–4]. At the beginning, the measurement error is not easy to meet the static event-triggered conditions, so it can effectively reduce resource consumption [5]. However, as the system approaches the consensus state, the threshold becomes smaller and smaller, causing the static event-triggered condition to be triggered excessively. Therefore, the main motivation of this paper is to develop a more efficient and flexible dynamic event-triggered optimal control based on ADP method.

With the increase of MASs scale and complexity, it may need to be divided into different subnets to achieve multiple consensus values to cope with the changes of the environment. As a generalization of multi-agent consensus control, group consensus is first mentioned in [6]. Thereafter, there are more and more researches on group consensus, such as [7,8]. In view of the advantages of group consensus, it seems to be a very interesting topic to put group consensus into the optimal control problem. In addition, the increase of system complexity means that the interaction between agents is not only cooperation. Due to the limited resources, competitive interaction is also essential [9]. Moreover, coopetition interaction is not uncommon in the real environment, such as Unmanned Aerial Vehicles [10], railway transportation [11] and so on. However, the work mentioned previously on the optimal control problem has few in relation to coopetition interaction, most of them are single cooperation. Therefore, the coopetition interaction in the optimal control problem is also a research focus of this paper.

To sum up, we will discuss optimal group consensus control for multi-agent systems in the coopetition networks via dynamic event-triggered methods. The main contributions are summarized mainly as follows: (1) Compared with the traditional time-triggered method to investigate the optimal control problem, this paper will use the event-triggered mechanism. Firstly, the static event-triggered condition is proposed, and then it is extended to the dynamic event-triggered condition to solve the excessive trigger problem caused by the static one. (2) A new tracking error control protocol for the MASs in coopetition networks is proposed. And the group consensus of second-order discrete MASs is investigated based on the ADP method in this paper. (3) The weight estimation errors of Actor-Critic neural network can be guaranteed uniformly ultimately bounded (UUB).

2 Preliminaries

2.1 Graph Theory

Consider the discrete-time MASs with one leader and N followers, let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ be a simple directed communication topology digraph where $\mathcal{V} = \{v_1, ..., v_N\}$ is a set of agents, $\mathcal{E} = \{(v_i, v_\mathrm{j})|v_i, v_i \in \mathcal{V}\} \subseteq \mathcal{V} \times \mathcal{V}$ denotes a set of directed edges and $\mathcal{A} = [a_{ij}]_{N \times N}$ represents a weighted adjacency matrix. $N_i = \{j \in \mathcal{V}|(v_j, v_i) \in \mathcal{E}\}$ denotes the neighbors set of the agent i.

The diagonal matrix $D = \text{diag}\{d_1, d_2, ..., d_N\}$ is the in-degree matrix of $\mathcal{G}$. $\mathcal{B} = \text{diag}\{b_i, b_2, ..., b_N\}$ is pinning matrix.

2.2 Problems Statement

Consider the second-order MASs with one leader and N followers. Then the followers. Dynamics is represented as

$$\begin{cases} x^i_{k+1} = Ax^i_k + Bv^i_k \\ v^i_{k+1} = Cv^i_k + T_i u^i_k \end{cases}, i \in \{1, 2, 3, ..., N\} \tag{1}$$

where $x_{i:k}$, $u_{i:k}$ and $v_{i:k}$ denote the position state, the control input and the velocity state, respectively. The system matrices A, B, C and T_i are supposed to be unknown constant matrices completely. And the dynamics of the leader is given as

$$\begin{cases} x^0_{k+1} = Ax^0_k + Bv^0_k \\ v^0_{k+1} = Cv^0_k \end{cases} \tag{2}$$

where $x_{0:k}$ is position state and $v_{0:k}$ is velocity state of the leader. Here the leader only generates desired signals that enable the followers to track, but it does not receive signals from other agents.

Define the local neighbor tracking error as

$$\delta^i_k = \sum_{j \in N_i} a_{ij}(y^i_k - \varGamma_{ij} y^j_k) + b_i(y^i_k - y^0_k) \tag{3}$$

where $y^i_k = \begin{bmatrix} x^i_k \ v^i_k \end{bmatrix}^{\mathrm{T}}$, $y^0_k = \begin{bmatrix} x^0_k \ v^0_k \end{bmatrix}^{\mathrm{T}}$, $\varGamma_{ij}$ is the coopetition coefficient. We utilize $\varGamma_{ij} < 0$ to denote the competitive interaction between the agent i and j. Otherwise, $\varGamma_{ij} > 0$ represents the cooperative interaction.

Notes and Comments. Note that a single equilibrium state can no longer meet the requirements of complex environments and distributed tasks. Therefore, we divide the MASs into different subnets and achieve multiple consensus values by designing a reasonable coopetition coefficient.

Based (1) and (2), we can get

$$\begin{cases} y^i_{k+1} = \varUpsilon y^i_k + \mathrm{P}_i u^i_k \\ y^i_{k+1} = \varUpsilon y^i_k \end{cases} \tag{4}$$

where $\varUpsilon = \begin{bmatrix} A \ B \\ 0 \ C \end{bmatrix}$ and $\mathrm{P}_i = \begin{bmatrix} 0 \\ T_i \end{bmatrix}$. Based on (4) and (5), we have the iterative local neighbor tracking error as below

$$\delta^i_{k+1} = \varUpsilon \delta^i_k + (d_i + b_i)\mathrm{P}_i u^i_k - \sum_{j \in N_i} a_{ij} \varGamma_{ij} \mathrm{P}_i u^j_k \tag{5}$$

Then the global tracking error can be deduced as

$$\begin{aligned} \delta_k &= (D - A \circ \varGamma) y_k + \mathcal{B}(y - \bar{y}^0_k) \\ &= (\bar{L} + \mathcal{B})(y_k - \bar{y}^0_k) \end{aligned} \tag{6}$$

where $\bar{L} = D - A \circ \Gamma$, $y_k = \left[{y_k^1}^{\mathrm{T}}, ..., {y_k^N}^{\mathrm{T}} \right]^{\mathrm{T}}$ and $\bar{y}_k^0 = \left[{y_k^0}^{\mathrm{T}}, ..., {y_k^0}^{\mathrm{T}} \right]^{\mathrm{T}}$; $\Gamma = [\Gamma_{ij}]$ is the coopetition coefficient matrix; $\circ$ represents Hadamard product; $\zeta_k = y_k - \bar{y}_k^0$ is defined as global consensus error.

Assumption 1. Suppose that the matrix $\bar{L}$ satisfies $\sum\limits_{j \in N_i} a_{ij}(1 - \Gamma_{ij}) = 0$.

Based on the work of [12] and Assumption 1, we can know $\lim\limits_{k \to \infty} \|\zeta_k\| = 0$ once $\lim\limits_{k \to \infty} \|\delta_k\| = 0$, i.e., the agents in each subnet will be synchronized with their group leader. Therefore, one of our goals is to minimize the global tracking error δ_k by ADP method.

In order to find the optimal control u_i^*, the discounted value function is designed with the local neighbor tracking errors as

$$J_i\left(\delta_k^i\right) = U_i(\delta_k^i, u_k) + \alpha J_i\left(\delta_{k+1}^i\right) \tag{7}$$

where $U_i(\delta_k^i, u_k) = \delta_k^{i\mathrm{T}} Q_{ii} \delta_k^i + u_k^{i\mathrm{T}} R_{ii} u_k^i + \sum\limits_{j \in N_i} u_k^{j\mathrm{T}} R_{ij} u_k^j$ denotes the reward function; $\alpha \in (0, 1]$ is the discount factor; $Q_{ii} \geq 0$, $R_{ii} \geq 0$ and $R_{ij} \geq 0$ are positive symmetric weighting matrices.

By Bellman optimality principle, we can know the optimal discounted value function $J_i^*\left(\delta_k^i\right)$ satisfies the following discrete-time Hamilton-Jacobi-Bellman (DT-HJB) equation

$$J_i^*\left(\delta_k^i\right) = \min_{u_k^i} \left\{ U_i(\delta_k^i, u_k) + \alpha J_i^*\left(\delta_{k+1}^i\right) \right\} \tag{8}$$

Under the necessary condition $\partial J_i^*\left(\delta_k^i\right) / \partial u_k^i = 0$, we can give the optimal control policy as

$$u_k^{i*} = -\frac{\alpha}{2}\left(b_i + d_i\right) R_{ii}^{-1} T_i^{\mathrm{T}} \partial J_i^*\left(\delta_{k+1}^i\right) / \partial \delta_{k+1}^i \tag{9}$$

3 Main Results

3.1 Event-Triggered Framework

The traditional ADP method usually adopts periodic sampling, which will cause a huge computation and a waste of resources with the increase of system scale and complexity. Therefore, we introduce event-triggered mechanism into the ADP method to reduce the Sampling frequency. The measurement error is defined as

$$e_k^i = \delta_k^i - \delta_{k_s^i}^i, k \in \left[k_s^i, k_{s+1}^i\right) \tag{10}$$

where k_s^i is the sth triggering instants of agent i. The monotonically increasing triggering instants sequence is determined by

$$\begin{cases} k_1^i = 0 \\ k_{s+1}^i = \inf\limits_{l > k_s^i,} \left\{ \begin{aligned} & l : \left\|e_k^i\right\|^2 - \tfrac{1-2F^2}{2F^2-\sigma_i}\left\|\delta_k^i\right\|^2 \\ & + \frac{\sum\limits_{j \in N_i} \left\|\hat{\omega}_{aj}(k_{s+1}^j)\psi_{aj}(z_{aj}(k_{s+1}^j))\right\|^2}{2F^2-\sigma_i} \geq 0, \forall k \in \left(k_s^i, l\right] \end{aligned} \right\} \end{cases} \tag{11}$$

where F and σ_i are positive constants, $\hat{\omega}_{aj}$ is the weight of the actor network for agent j , $\psi_{aj}(\cdot)$ is $tanh(\cdot)$ which denotes the activation function, z_{aj} is input vector combine with $\delta_j(k_s^j)$.

The above condition (11) is a static event-triggered condition that contains constant threshold parameters that should be determined by the operation engineer or designer. However, this may cause agents to communicate unnecessarily when consensus is approaching. The event-triggered condition (11) is over-triggered. Taking this into account, we propose a dynamic event-triggered law that introduces an external threshold. The triggering instants sequence determined by the following dynamic event-triggered conditions

$$\begin{cases} k_1^i = 0 \\ k_{s+1}^i = \inf\limits_{l > k_s^i,} \left\{ \begin{aligned} & l : \theta_i \left(\left\| e_k^i \right\|^2 - \frac{1-2F^2}{2F^2-\sigma_i} \left\| \delta_k^i \right\|^2 \right. \\ & \left. + \frac{\sum\limits_{j \in N_i} \left\| \hat{\omega}_{aj}(k_{s+1}^j) \psi_{aj}(z_{aj}(k_{s+1}^j)) \right\|^2}{2F^2-\sigma_i} \right) \geq \eta_k^i, \forall k \in \left(k_s^i, l\right] \end{aligned} \right\} \end{cases} \tag{12}$$

where F, σ_i, $\hat{\omega}_{aj}$, $\psi_{aj}(\cdot)$ and z_{aj} are defined in (11), θ_i is a positive constants, η_k^i is external threshold satisfying

$$\dot{\eta}_k^i = -\rho_i \eta_k^i + \sigma_i \left(\frac{1-2F^2}{2F^2-\sigma_i} \left\| \delta_k^i \right\|^2 - \left\| e_k^i \right\|^2 - \frac{1}{2F^2-\sigma_i} \sum_{j \in N_i} \left\| \hat{\omega}_{aj}(k_{s+1}^j) \psi_{aj}(z_{aj}(k_{s+1}^j)) \right\|^2 \right) \tag{13}$$

with $\eta_0^i > 0$ and $\rho_i > 0$.

Notes and Comments. In (12), η_k^i varies in real time according to the measurement error, local neighbor tracking error and the weight of actor network. Compared with the static event-triggered condition (11), the dynamic one is essentially a static trigger mechanism with "history" information. If η_k^i is set to 0, the dynamic event-triggered condition in (12) becomes a static one, which can be seen as its special case. Note that the external threshold η_k^i will improve the excessive trigger problem caused by static trigger conditions, thus saving communication and computing resources.

3.2 Implementation of the ADP Method

In order to find the optimal discounted value function and control policy, the online-learning policy iteration (PI) algorithm is presented.

PI Algorithm: Let $k_s = k = 0$ and $s = 0$, choose the initial admissible control policy u_0^i and a small enough positive number $\iota_i > 0$.

1) compute the measurement error and tracking error;
2) if the dynamic event-triggered condition (12) is satisfied, compute the iterative local value function by (7).
3) set $s = s + 1$, $k_s = k$. Update the control policy by (10).

4) End until $\left|J_i\left(\delta_k^i\right) - J_i\left(\delta_{k+1}^i\right)\right| < \iota_i$, else $k = k+1$ and back to step 1;
5) return $u_{k_s}^i$ and $J_i\left(\delta_k^i\right)$.

Through the continuous iteration of the above algorithm, u^* can be get finally. The process of proving the convergence of the above PI algorithm is similar to the work in [13], so it is no longer proved in this paper.

Since the discrete HJB equation is very difficult to resolve, we use the Actor-Critic neural network to obtain an approximate optimal solution of the equation in the PI algorithm. Figure 1 designs the structure diagram of the optimal consensus control network based on event trigger. When the event-triggered condition is satisfied, the current time and state are recorded as sampling time and state respectively, and then sent to the controller. The optimal control policy is approximated by the Actor-Critic network.

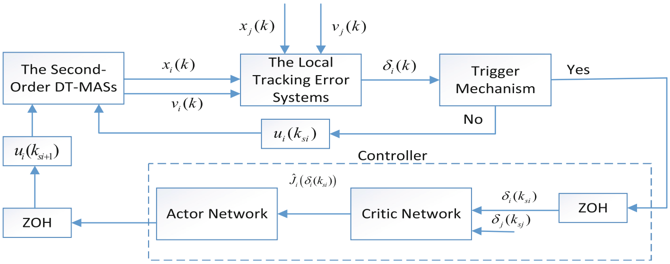


Fig. 1. Schematic of event-triggered ADP method.

(1) Critic NN

The discounted value function is approximated by the following critic network

$$\hat{J}_i(k_s^i) = \hat{\omega}_{ci}^{\mathrm{T}}(k_s^i)\psi_{ci}(z_{ci}(k_s^i)) \tag{14}$$

where $\hat{\omega}_{ci}$ is the weight of critic network; z_{ci} is an input vector with $\delta_{k_s}^i$, $u_{k_s}^i$ and $u_{k_s}^{N_i}$; $\psi_{ci}(\cdot)$ is $tanh(\cdot)$ which denotes the activation function.
Based on the discounted value function (7) and approximation discounted value function (14), we define the error function as

$$\bar{\vartheta}_{ci}(k) = \frac{1}{2}\vartheta_{ci}^{\mathrm{T}}(k)\vartheta_{ci}(k) \tag{15}$$

where $\vartheta_{ci}(k) = U_i(\delta_k^i, u_{k_s}) + \alpha\hat{J}_i(k_s^i + 1) - \hat{J}_i(k_s^i)$. The goal of our work is to minimize the error function by adjusting the weights of the neural network. The weight update policy based on gradient is derived as

$$\hat{\omega}_{ci}(k) {=} \begin{cases} \hat{\omega}_{ci}(k-1) - \kappa_{ci}\alpha\psi_{ci}(z_{ci}(k-1))\vartheta_{ci}(k-1), k{=}k_s^i; \\ \hat{\omega}_{ci}(k-1), k \in \left(k_s^i, k_{s+1}^i\right) \end{cases} \tag{16}$$

(2) Actor NN

The control policy is approximated by the following actor network

$$\hat{u}^i_{k_s} = \hat{\omega}^{\mathrm{T}}_{ai}(k^i_s)\psi_{ai}(z_{ai}(k^i_s)) \tag{17}$$

where $\hat{\omega}_{ai}$, $\psi_{ai}(\cdot)$ and z_{ai} are defined in (11). Define the output error function with the approximate control policy (17) as

$$\bar{\vartheta}_{ai}(k) = \frac{1}{2}\vartheta^{\mathrm{T}}_{ai}(k)\vartheta_{ai}(k) \tag{18}$$

with $\vartheta_{ai}(k) = \hat{u}^i_k - \tilde{u}^i_k$ and $\tilde{u}^i_k$ is the target of the actor network. The actor network weight update policy can be derived by the gradient descent method as

$$\hat{\omega}_{ai}(k) = \begin{cases} \hat{\omega}_{ai}(k-1) - \kappa_{ai}\psi_{ai}(z_{ai}(k-1)) \\ \times\left[\hat{\omega}_{ai}(k-1)\psi_{ai}(z_{ai}(k-1)) - \tilde{u}^i_k\right], k{=}k^i_s; \\ \hat{\omega}_{ai}(k-1), k \in \left(k^i_s, k^i_{s+1}\right) \end{cases} \tag{19}$$

3.3 Stability Analysis

We will analyze the stability of the MASs which combines the event-triggered mechanism and the neural network structure in two cases whether the events are triggered or not. In order to facilitate the following analysis, we first define two approximation errors. The critic network weight approximation error $\tilde{\omega}_{ci}(k^i_s)$ is define as

$$\tilde{\omega}_{ci}(k^i_s) = \hat{\omega}_{ci}(k^i_s) - \omega^*_{ci} \tag{20}$$

where ω^*_{ci} is the target weight of the critic network. The actor network weight approximation error $\tilde{\omega}_{ai}(k^i_s)$ is define as

$$\tilde{\omega}_{ai}(k^i_s) = \hat{\omega}_{ai}(k^i_s) - \omega^*_{ai} \tag{21}$$

where ω^*_{ai} is the target weight of the actor network. Then, when the measurement error of the MASs satisfies the proposed dynamic event-triggered condition (12), we have the following theorem.

Theorem 1. *When the dynamic event-triggered condition is satisfied, the MASs updates the control policy of the current agent and the weights of the neural network associated with it. The approximation error $\tilde{\omega}_{ci}(k^i_s)$ and $\tilde{\omega}_{ai}(k^i_s)$can be guaranteed UUB.*

Proof. Define the following Lyapunov function

$$L_i(k^i_s) = L_{i1}(k^i_s) + L_{i2}(k^i_s) \tag{22}$$

where $L_{i1}(k^i_s) = \tilde{\omega}^T_{ci}(k^i_s)\tilde{\omega}_{ci}(k^i_s)$ and $L_{i2}(k^i_s) = \tilde{\omega}^T_{ai}(k^i_s)\tilde{\omega}_{ai}(k^i_s)$. According to (15) and (16), we can get

$$\begin{aligned} \tilde{\omega}_{ci}(k^i_{s+1}) &= \hat{\omega}_{ci}(k^i_s) - \kappa_{ci}\alpha\psi_{ci}(z_{ci}(k))\vartheta_{ci}(k) - \omega^*_{ci} \\ &= (1-\lambda_1)\tilde{\omega}_{ci}(k^i_s) - \varepsilon_1 \end{aligned} \tag{23}$$

where $\Delta\psi_{ci}(z_{ci}(k)) = \psi_{ci}(z_{ci}(k+1)) - \psi_{ci}(z_{ci}(k))$, λ_1 is an eigenvalue of $\alpha\kappa_{ci}\psi_{ci}(z_{ci}(k))(1-\alpha)\Delta\psi_{ci}^T(z_{ci}(k))$ and $\lambda_1 \in (-\infty, 1)$ can be designed by choosing the appropriate κ_{ci}, and $\varepsilon_1 = \alpha\kappa_{ci}\psi_{ci}(z_{ci}(k))\left((\alpha-1)\Delta\psi_{ci}^T(z_{ci}(k))\omega_{ci}^* - U_i(\delta_k^i, u_{k_s})\right)$. $\Delta\psi_{ci}(z_{ci}(k))$ and ε_1 are bounded since $\psi_{ci}(z_{ci}(k))$ is bounded. Then the difference of $L_{i1}(k_s^i)$ is given by

$$\begin{aligned}
&\Delta L_{i1}(k_s^i)\\
&= \left((1-\lambda_1)\tilde{\omega}_{ci}(k_s^i) - \varepsilon_1\right)^{\mathrm{T}}\left((1-\lambda_1)\tilde{\omega}_{ci}(k_s^i) - \varepsilon_1\right) - \tilde{\omega}_{ci}^T(k_s^i)\tilde{\omega}_{ci}(k_s^i)\\
&\le {\sigma_1}^2\left\|\tilde{\omega}_{ci}(k_s^i)\right\|^2 + \|\varepsilon_1\|^2 + (\lambda_1 - 1)\left(\left\|\tilde{\omega}_{ci}(k_s^i)\right\|^2 + \|\varepsilon_1\|^2\right) - \left\|\tilde{\omega}_{ci}(k_s^i)\right\|^2\\
&\le \left(\lambda_1^2 - \lambda_1 - 1\right)\left\|\tilde{\omega}_{ci}(k_s^i)\right\|^2 + \lambda_1\|\varepsilon_1\|^2
\end{aligned} \tag{24}$$

Based on the work in [2], the following condition can be satisfied when λ_1=0.

$$\Delta L_{i1}(k_s^i) \le -\left\|\tilde{\omega}_{ci}(k_s^i)\right\|^2 \tag{25}$$

Since $\left\|\tilde{\omega}_{ci}(k_s^i)\right\| > 0$, $L_{i1}(k_s^i) < 0$. Then the difference of $L_{i2}(k_s^i)$ is given by

$$\begin{aligned}
\Delta L_{i2}(k_s^i) &= \tilde{\omega}_{ai}^{\mathrm{T}}(k_{s+1}^i)\tilde{\omega}_{ai}(k_{s+1}^i) - \tilde{\omega}_{ai}^{\mathrm{T}}(k_s^i)\tilde{\omega}_{ai}(k_s^i)\\
&= \left[\left(1 - \kappa_{ai}\psi_{ai}^2(z_{ai}(k_s^i))\right)^2 - 1\right]\left\|\tilde{\omega}_{ai}(k_s^i)\right\|^2
\end{aligned} \tag{26}$$

Due to $\kappa_{ai} \in (0,1)$, $\psi_{ai}(\cdot) \in (-1,1)$ and $\left\|\tilde{\omega}_{\mathrm{a}i}(k_s^i)\right\| > 0$, the difference $\Delta L_{i2}(k_s^i) < 0$. Based on (25) and (26), we can know $\Delta L_i(k_s^i) = \Delta L_{i1}(k_s^i) + \Delta L_{i2}(k_s^i) < 0$. That is to say, $\tilde{\omega}_{ci}(k_s^i)$ and $\tilde{\omega}_{ai}(k_s^i)$ are UUB and the system is ultimately bounded at the event- triggered instants. The proof is completed.

Assumption 2. There exists a constant $F > 0$ such that $\left\|\delta_{k+1}^i\right\| \le F\left\|\delta_k^i\right\| + F\left\|e_k^i\right\|$.

Under the above assumptions, when the measurement error of the MASs does not meet the proposed dynamic event-triggered condition, we have the following theorem.

Theorem 2. *When the measurement error of agent i does not meet the condition (12), the agent will not update its relevant control policy and the weights of neural network. The MASs can be guaranteed to be asymptotically stable if there exist positive scalar σ_i, F, ρ_i and θ_i such that*

$$\begin{cases} \sigma_i > 0\\ 1 - 2F^2 < 0\\ 0 < 2F^2 - \sigma_i < \rho_i\theta_i \end{cases} \tag{27}$$

Proof. Consider the following Lyapunov function

$$L_i^2(k) = \delta_k^{i\mathrm{T}}\delta_k^i + \sum_{j\in N_i} u_k^{j\mathrm{T}}u_k^j + \eta_k^i \tag{28}$$

Since the dynamic event-triggered condition (12) is dissatisfied, we can know

$$\theta_i \left(\left\| e_k^i \right\|^2 - \tfrac{1-2F^2}{2F^2-\sigma_i} \left\| \delta_k^i \right\|^2 + \tfrac{1}{2F^2-\sigma_i} \times \sum_{j \in N_i} \left\| \hat{\omega}_{aj}(k_{s+1}^j) \psi_{aj}(z_{aj}(k_{s+1}^j)) \right\|^2 \right) < \eta_k^i \tag{29}$$

Based on (13), (29) and Assumption 2, the difference of $L_i^2(k)$ can be derived as

$$\begin{aligned} &\mathit{\Delta} L_i(k) \\ &= \left\| \delta_{k+1}^i \right\|^2 - \left\| \delta_k^i \right\|^2 + \sum_{j \in N_i} \left(\left\| u_{k+1}^j \right\|^2 - \left\| u_k^j \right\|^2 \right) + \dot{\eta}_k^i \\ &\leq \left(\tfrac{\sigma_i(1-2F^2)}{2F^2-\sigma_i} \right) \left\| \delta_k^i \right\|^2 + \tfrac{(2F^2-\sigma_i)-\rho_i\theta_i}{\theta_i} \eta_k^i - \tfrac{\sigma_i}{2F^2-\sigma_i} \sum_{j \in N_i} \left\| u_{k+1}^j \right\|^2 - \sum_{j \in N_i} \left\| u_k^j \right\|^2 \end{aligned} \tag{30}$$

When σ_i, F, ρ_i and θ_i satisfy the condition (27), it can be known that $\mathit{\Delta} L_i^2(k) \leq 0$ and the system is asymptotically stable. The proof is completed.

4 Simulation

We will verify the effectiveness of our proposed ADP method under the dynamic event-triggered mechanism through the simulation experiments of three trigger modes, namely, time trigger, static event trigger and dynamic event trigger. Consider a MASs (4) with the topological structure of Fig. 2. In practical applications, we can abstract each UAV, robot and so on into an agent node. Given the following initial parameters in the event-triggered mechanism. Pinning gain: $b_1 = 1$, $b_2 = b_3 = b_4 = b_5 = b_6 = b_7 = 0$. Competition coefficients: $\mathit{\Gamma}_{21} = -0.1$, $\mathit{\Gamma}_{25} = 2.1$, $\mathit{\Gamma}_{31} = 1$, $\mathit{\Gamma}_{42} = 1$, $\mathit{\Gamma}_{54} = 1$, $\mathit{\Gamma}_{63} = 3$, $\mathit{\Gamma}_{67} = -1$. Discount factor: $\alpha = 0.95$, $\kappa_{ci} = \kappa_{ai} = 0.03$. Event-triggered parameters: $\theta_i = 0.6$, $\rho_i = 0.59$, $F = 0.729$, $\sigma_i = 0.72$. In the simulation experiments under the three trigger modes, we set the initial position and initial speed of the agent to be the same, and finally the MASs reached the bipartite consensus.

Figure 3 depicts that under the dynamic event-triggered mechanism we designed, the control policies will remain unchanged for a certain period of time because of the threshold we set. And Figs. 4 and 5 show that even if the event trigger reduces the communication of agents, it still reaches the bipartite consensus. Under the event-triggered modes we have proposed, whenever the measurement error exceeds the event-triggered condition (11) and (12), the agent is triggered, and then the control policy is updated. Therefore, compared with the time-triggered mechanism in which every moment is triggered, the event-triggered mechanism proposed in this paper can significantly save the resource consumption. However, it is clear from Fig. 6 that due to the external threshold added to the dynamic event-triggered condition, the effect is better than that of the static event-triggered mechanism.

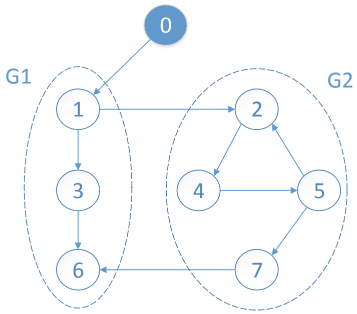


Fig. 2. Topology of the complex systems.

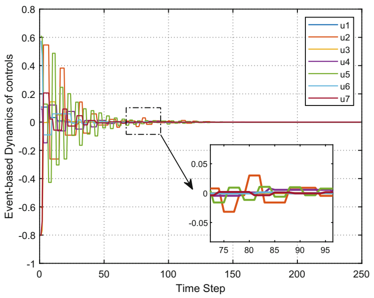
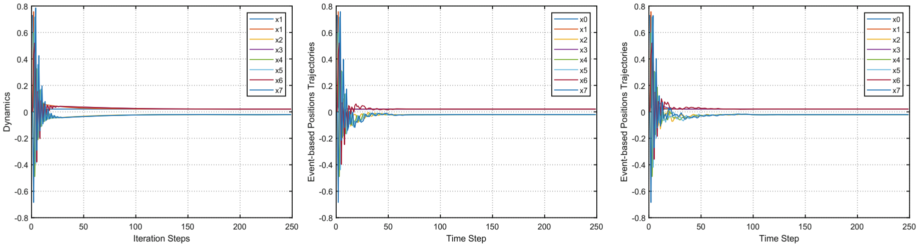
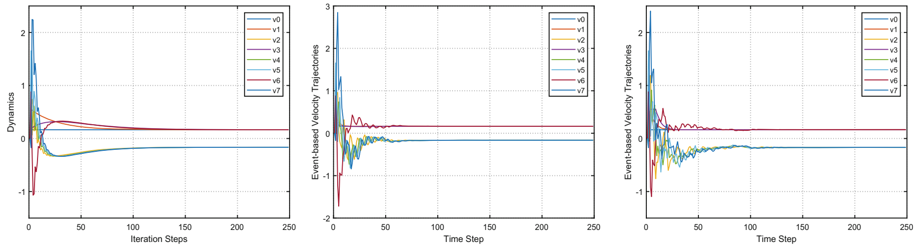


Fig. 3. Trajectories of the control policies.



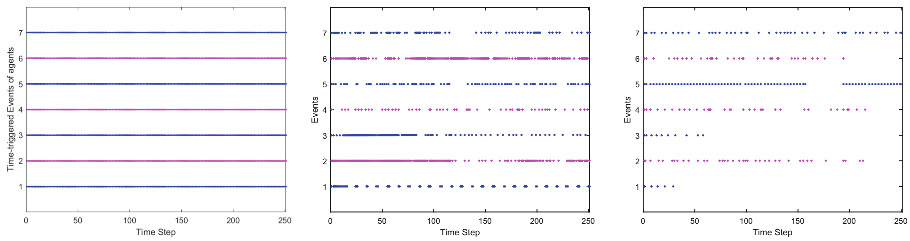
(a) (b) (c)

Fig. 4. Trajectories of Position States under three trigger modes: (a) time-triggered mode; (b) static event-triggered mode; (c) dynamic event-triggered mode.



(a) (b) (c)

Fig. 5. Trajectories of Velocity States under theree trigger modes: (a) time-triggered mode; (b)static event-triggered mode;(c) dynamic event-triggered mode.



(a) (b) (c)

Fig. 6. Event-triggered instants of agents under three trigger modes: (a) time-triggered mode; (b) static event-triggered mode; (c) dynamic event-triggered mode.

5 Conclusion

In this paper, the optimal tracking control problem of discrete second-order MASs under event-triggered mechanism is developed. We propose dynamic event-triggered mechanism to solve the problem of excessive triggering caused by static event-triggered conditions. Then, under this mechanism, the optimal control problem is studied by using ADP method, and the stability of the system is proved.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61876200, 62072066 and 62006031, in part by the Foundation of Guangxi Key Laboratory of Cryptography and Information Security(GCIS201908), in part by the Major Scientific and Technological Research Program of Chongqing Municipal Education Commission under grant No. KJZD-M202100602 and in part by the Natural Science Foundation Project of Chongqing Science and Technology Commission under Grant Nos. cstc2018jcyjAX0112 and cstc2019jcyj-msxmX0545.

References

1. Luo, B., Yang, Y., Liu, D., Wu, H.: Event-triggered optimal control with performance guarantees using adaptive dynamic programming. IEEE Trans. Neural Netw. Learn. Syst. **31**(1), 76–88 (2020)
2. Yang, W., Wei, Q., Liu, D.: Event-triggered adaptive dynamic programming for discrete-time multi-player games. Inf. Sci. **506**, 457–470 (2019)
3. Sahoo, A., Xu, H., Jagannathan, S.: Near optimal event-triggered control of nonlinear discrete-time systems using neurodynamic programming. IEEE Trans. Neural Netw. Learn. Syst. **27**(9), 1801–1815 (2016)
4. Wei, Z., Zhang, H.: Distributed optimal coordination control for nonlinear multi-agent systems using event-triggered adaptive dynamic programming method. ISA Trans. **91**, 184–195 (2019)
5. He, W., Xu, B., Han, Q.-L., Qian, F.: Adaptive Consensus control of linear multiagent systems with dynamic event-triggered strategies. IEEE Trans. Cybern. **50**(7), 2996–3008 (2020)

6. Yu, J., Wang, L.: Group consensus of multi-agent systems with undirected communication graphs. In: 2009 7th Asian Control Conference, pp. 105–110 (2009)
7. Qin, J., Yu, C.: Group consensus of multiple integrator agents under general topology. In: 52nd IEEE Conference on Decision and Control, pp. 2752–2757 (2013)
8. Xie, D., Liang, T.: Second-order group consensus for multi-agent systems with time delays. Neurocomputing **153**, 133–139 (2015)
9. Ji, L., Yu, X., Li, C.: Group consensus for heterogeneous multiagent systems in the competition networks with input time delays. IEEE Trans. Syst. Man Cybern. Syst. **50**(11), 4655–4663 (2020)
10. Liu, J., Shi, T., Li, P., Ren, X., Ma, H.: Trajectories planning for multiple UAVs by the cooperative and competitive PSO algorithm. In: 2015 IEEE Intelligent Vehicles Symposium (IV), pp. 107–114 (2015)
11. Feng, F., Xu, Y., Tang, Z.: Research on the charge rate of railway value-guaranteed transportation based on competitive and cooperative relationships. Adv. Mech. Eng. **10**(1), 168781401774769 (2018)
12. Li, J., Ji, L., Li, H.: Optimal consensus control for unknown second-order multi-agent systems: using model-free reinforcement learning method. Appl. Math. Comput. **410**, 126451 (2021). https://doi.org/10.1016/j.amc.2021.126451
13. Peng, Z., Hu, J., Ghosh, B.K.: Optimal tracking control of heterogeneous multi-agent systems with switching topology via actor-critic neural networks. In: 2018 37th Chinese Control Conference (CCC), pp. 7037–7042 (2018)