Event-triggered Multi-agent Optimal Regulation Using Adaptive Dynamic Programming

Xiangnan Zhong
Department of Computer & Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL, USA
xzhong@fau.edu

Haibo He
Department of Electrical, Computer and Biomedical Engineering
University of Rhode Island
Kingston, RI, USA
haibohe@uri.edu

***Abstract*—This paper develops an event-triggered multi-agent control method based on adaptive dynamic programming (ADP) techniques. Different from the traditional ADP-based multi-agent control with fixed sampling period, our method designs an adaptive controller only based on the efficiently reduced samples. The sampling instants are decided by an adaptive triggering condition to guarantee the stability of the event-triggered learning process. The theoretical analysis of the proposed method is also provided in this paper. It is proved that the designed event-triggered ADP controller can make all the agents synchronize to the leader's dynamics with reduced sampled data, and also reach Nash equilibrium at the same time. Therefore, the proposed method can save the computational resources in the learning process. Finally, the simulation results verify the theoretical analysis and also demonstrate the performance of the developed method.**

***Index Terms*—Event-triggered control, adaptive dynamic programming, multi-agent systems, and online learning.**

I. Introduction

Multi-agent systems are a group of autonomous systems, interacting with each other through communication or sensing networks [1], [2]. Such systems can perform certain challenge tasks which cannot be accomplished by a single agent [3]. For instance, multi-agent control techniques have been successfully used in robotics for replicating self-organized behaviors found in nature such as bird flocking, and fish schooling [4]. They are also used in developing applications such as formation control, rendezvous, robot coordination, and distributed estimation [5], [6]. A fundamental concept underlying these techniques is the notion of consensus [7], [8], [9]. In [10], information consensus of multi-agent systems was developed under the limited and unreliable information exchange with dynamically changing interaction topologies. The authors then provided a tutorial overview of information consensus in multivehicle cooperative control in [11], where theoretical results regarding consensus-seeking were summarized and several specific applications of consensus algorithms to multivehicle coordination were also described. The time-dependent communication links were considered in [12] and a novel approach was designed which was centered around the notion of convexity. The problem of cooperation among a collection of vehicles was considered in [13]. Such systems performed a shared task using intervehicle communication to coordinate their actions. In [14], strategic cyber attacks were investigated in multi-agent systems and then a distributed secure consensus tracking control method was established with a hybrid stochastic secure framework. So far, most of the studies on multi-agent system control are based on accurate system functions and/or models. In the real-world applications, however, the likelihood to access the complete knowledge of system functions is either infeasible or very difficult to obtain. Fortunately, adaptive dynamic programming (ADP) techniques give us an opportunity to solve this problem.

By approximating solutions of the Hamilton-Jacobi-Bellman (HJB) equation, ADP has attract significantly increasing attentions [15], [16]. It has been widely recognized as one of the "core methodologies" to achieve optimal control for intelligent systems [17], [18]. Extensive efforts have been dedicated to developing ADP method from both theoretical researches and real-world applications [19], [20], [21]. Recently, ADP method has been integrated into the multi-agent control designs to relax the requirements of accurate system functions [22], [23]. The authors in [24] developed a linear quadratic regulator-based optimal cooperative design for synchronization control of discrete-time multi-agent systems. The actor-critic network structures were provided in [25] for multi-agent graphical games depending only on the local information available to each agent. The stability and Nash equilibrium were discussed and proved in [26] with explicit theoretical foundation analysis. Then in [27], a data-driven learning-based method was designed for the discrete-time multi-agent systems with completely unknown dynamics. Later, this idea was developed for heterogeneous multi-agent system in [28], which proposed an optimal output regulation design for partially model-free heterogeneous linear multi-agent systems with disturbance. The authors in [29] considered fuzzy ADP structure for leader-based multi-agent differential games.

In the literature, ADP-based intelligent controls usually depends on the periodic transmitted data with fixed sampling period. The control laws are updated for every time instant.

This work was supported in part by the National Science Foundation under Grant ECCS 1917275 and ECCS 1947419.

The communication burden and computation load will be tremendously high during the learning process. In the situation with limited computation bandwidth or constrained sensor power resources, this disadvantage will become severe or sometimes seriously impact the learning performance. To solve this problem, event-triggered control method [30], [31], [32], [33], [34] is introduced into the ADP design to reduce the computation burden by transmitting the system data and updating the control laws only when it is necessary [35], [36], [37], [38], [39], [40], [41]. Recently, it has also been developed for the multi-agent systems. In [42], the authors studied a tracking control problem based on the event-triggered mechanics. The distributed optimal coordination control was designed in [43] based on the event-triggered ADP techniques.

Motivated by the above observations, this paper develops a multi-agent consensus control method under the event-triggered mechanism using ADP techniques. The theoretical foundation of the designed method is discussed in details. The major contributions of this paper can be summarized as follows: (1) different from the existing ADP-based multi-agent consensus designs, this paper develops an event-triggered scheme to save the computational resources and only updates the control signals when an event for certain agent is triggered. An adaptive triggering condition is designed for each agent to determine the sampling instants and also guarantee the stability of the entire system with reduced sampled data. Note that our proposed method triggered the system in an asynchronous mode, which means different agents will have their own triggering instants. This will make the learning process more flexible. (2) comparing with the event-triggered multi-agent ADP design in literature, this paper is focused on the consensus control problem and provides the explicit theoretical foundation for the proposed method. It shows that the agents can achieve consensus under the developed event-triggered control laws and all the agents can achieve Nash equilibrium at the same time. The Zeno-free behavior in this learning process is also discussed. We also compare our results with the traditional time-triggered ADP-based multi-agent consensus control design, which shows that our method can achieve competitive performance based on the reduced sampled system states.

The rest of this paper is organized as follows. Section II presents the preliminary of graphs and consensus for multi-agent systems. The proposed event-triggered ADP method is developed in Section III for multi-agent consensus control problem with the theoretical analysis of system stability and Nash equilibrium. Then, Section IV discusses the simulation results to demonstrate the effectiveness of the proposed method and verify the theoretical analysis. Finally, Section V concludes this paper.

II. Preliminary

A continuous-time multi-agent system is considered as

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

where $x_i(t) \in R^n$ is the state, and $u_i(t) \in R^{m_i}$ is the input coordination control for an arbitrary agent i, $i \in 1, 2, \cdots, N$. Let $A \in R^{n\times n}$, $B_i \in R^{m_i \times m_i}$ be the constant matrices.

The leader (target agent) is defined as

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

Usually, only a small percentage of the system is connected to the leader in multi-agent graph.

The goal of each agent is to approximate the optimal distributed control laws $u_i(t)$ based on its own information and local observations, such that all the agents can synchronize to the leader.

Therefore, a local neighborhood tracking error is defined as

$$\delta_i(t) = \sum_{j\in N_i} p_{ij}(x_i(t) - x_j(t)) + q_i(x_i(t) - x_0(t)) \tag{3}$$

where p_{ij} is the adjacency elements, $p_{ij} > 0$ when agent i and j have direct connection, otherwise, $p_{ij} = 0$, and $q_i \geq 0$ is the pinning gain [44], $q_i > 0$ when agent i has a direct path to the leader, otherwise $q_i = 0$. Here, we assume $q_i > 0$ for at least one agent.

Based on the discussion in [45], we have that the synchronization error $\eta_i(t) = x_i(t) - x_0(t)$ can be made arbitrarily small by making the neighborhood tracking errors $\delta_i(t)$ small. Different from [45], this paper will design the learning-based control signal based on the limited received data. Therefore, we further rewrite the dynamics of $\delta_i(t)$ as

$$\begin{aligned}\dot{\delta}_i(t) &= \sum_{j\in N_i} p_{ij}(\dot{x}_i(t) - \dot{x}_j(t)) + q_i(\dot{x}_i(t) - \dot{x}_0) \\ &= A\delta_i(t) + (d_i + q_i)B_i u_i(t) - \sum_{j\in N_i} p_{ij} B_j u_j(t).\end{aligned} \tag{4}$$

Here, N_i is defined as the set of neighbors of an agent i. It can be seen that dynamic system (4) has multiple control inputs, i.e., u_i from itself and u_j, $j \in N_i$, from all its neighbors. This means the agent can receive the control signals from their directly connected neighbors.

III. Event-triggered cooperative multi-agent systems on graph

A. Consensus under event-triggered condition

In order to save the resources, this paper updates the controller only when an event is triggered. Therefore, a sampled-data system is introduced with a monotonically increasing sequence of sampling instants $\{\tau_{i,k}\}_{k=0}^{\infty}$, where $\tau_{i,k} < \tau_{i,k+1}$, for $i = 1, 2, \cdots, N$, $k = 0, 1, 2, \cdots, \infty$. The time instant $\tau_{i,k}$ denotes the kth consecutive sampling instant of agent i. The outputs of the sampled-data system shall be $\hat{\delta}_{i,k} = \delta_i(\tau_{i,k})$, which are the local neighborhood tracking errors at the sampling instants. For simplicity, we assume the sampled-data system has zero task delay.

Define the gap function for $\forall t \in [\tau_{i,k}, \tau_{i,k+1})$ as

$$e_{i,k}(t) = \hat{\delta}_{i,k} - \delta_i(t) \tag{5}$$

which is the difference between the sampled and the current local neighborhood tracking errors.

Therefore, the problem becomes to find the distributed control laws $\gamma_i(\hat{\delta}_{i,k})$, which maps the sampled local neighborhood tracking error $\hat{\delta}_{i,k}$, rather than the current error $\delta_i(t)$, onto the control laws. Assume that $\gamma_i(\hat{\delta}_{i,k})$ is a Lipschitz continuous function. The obtained control sequence $\{\gamma_i(\hat{\delta}_{i,k})\}_{k=0}^{\infty}$ becomes a continuous signal for agent i through a zero-order hold (ZOH). In particular, this control signal can be seen as a piecewise constant function. For an arbitrary agent i, within any time interval $[\tau_{i,k}, \tau_{i,k+1})$, the controller is $u_i(t) = \gamma_i(\hat{\delta}_{i,k})$, $k = 0, 1, 2, \cdots, \infty$.

Therefore, we can rewrite the local neighborhood tracking error dynamics (4) as

$$\begin{aligned}\dot{\delta}_i(t) &= A\delta_i(t) + (d_i + q_i)B_i\gamma_i(\hat{\delta}_{i,k}) - \sum_{j\in N_i} p_{ij}B_j u_j(t)\\ &= A\delta_i(t) + (d_i + q_i)B_i\gamma_i(\delta(t) + e_{i,k}(t))\\ &\quad - \sum_{j\in N_i} p_{ij}B_j u_j(t), \qquad \forall t \in [\tau_{i,k}, \tau_{i,k+1})\end{aligned} \tag{6}$$

Note that the control signals in error dynamics (6) are only updated when an event is triggered. Now, our goal becomes to achieve optimality subject to (6), and hence according to Lemma 1, system (1) can achieve consensus.

First, let us recall the local performance indices for the traditional time-triggered multi-agent system,

$$\begin{aligned}J_i(\delta_i, u_i, u_{-i}) &= \int_{t_0}^{\infty}\left(\delta_i^T Q_{ii}\delta_i + u_i^T R_{ii}u_i + \sum_{j\in N_i} u_j^T R_{ij}u_j\right)dt\\ &= \int_{t_0}^{\infty} U(\delta_i, u_i, u_{-i})dt\end{aligned} \tag{7}$$

where $u_{-i} = \{u_j | j \in N_i\}$ is the set of control actions from the neighbors of agent i, $U(\delta_i, u_i, u_{-i}) = \delta_i^T Q_{ii}\delta_i + u_i^T R_{ii}u_i + \sum_{j\in N_i} u_j^T R_{ij}u_j$ is the reinforcement signal, and $Q_{ii} > 0$, $R_{ii} > 0$, and $R_{ij} > 0$ are the constant and symmetric matrices with appropriate dimensions.

When J_i are finite, a differential equivalent to (7) is given by Bellman's equation

$$\begin{aligned}&H(\delta_i, \nabla J_i, u_i, u_{-i})\\ &\equiv(\nabla J_i)^T\left(A\delta_i + (d_i + q_i)B_i u_i - \sum_{j\in N_i} p_{ij}B_j u_j(t)\right)\\ &\quad + \delta_i^T Q_{ii}\delta_i + u_i^T R_{ii}u_i + \sum_{j\in N_i} u_j^T R_{ij}u_j = 0\end{aligned} \tag{8}$$

where $\nabla J_i = \partial J_i(\delta_i, u_i, u_{-i})/\partial\delta_i$ is the partially derivatives of the performance indices $J_i(\delta_i, u_i, u_{-i})$ with respect to the local neighborhood tracking error δ_i, with boundary condition $J_i(\delta_i(0), u_i, u_{-i}) = 0$. That is, the solution of equation (8) serves as an alternative to evaluating the infinite integral (7) for finding the value associated to the current feedback controls. Therefore, according to Bellman's optimality equation, the optimal response of agent i to fixed laws u_{-i} can be derived by minimizing Hamiltonian function with respect to u_i as

$$u_i^* = arg\min_{u_i} H_i(\delta_i, \nabla J_i^*, u_i, u_{-i}) \tag{9}$$

where $\nabla J_i^* = \partial J_i^*(\delta_i, u_i^*, u_{-i}^*)/\partial\delta_i$. Assume that the minimum of $H_i(\delta_i, \nabla J_i^*, u_i, u_{-i})$ for agent i exists and is unique. Hence, the optimal controls u_i^* should satisfy

$$u_i^* = \frac{\partial H_i}{\partial u_i} = -\frac{1}{2}(d_i + q_i)R_{ii}^{-1}B_i^T\frac{\partial J_i^*}{\partial\delta_i} \equiv \gamma_i^*(\delta_i). \tag{10}$$

Note that, equations (8) and (10) show the HJB equations and the distributed control laws, respectively, in time-triggered condition. Both of them are updated periodically with fixed sampling periodic.

To reduce the computation load, we will develop the performance indices and the control laws under the event-triggered condition. This means the controller is designed only based on the sampled local neighborhood tracking error $\hat{\delta}_{i,k}$, rather than the current error δ_i. Therefore, we obtain the event-triggered control laws as

$$u_i^* = \gamma_i^*(\hat{\delta}_{i,k}) = -\frac{1}{2}(d_i + q_i)R_{ii}^{-1}B_i^T\frac{\partial J_i^*(\hat{\delta}_{i,k})}{\partial\delta_i} \tag{11}$$

Therefore, the corresponding performance indices under the event-triggered condition become

$$\begin{aligned}&J_i(\delta_i, \gamma_i(\hat{\delta}_{i,k}), u_{-i})\\ &= \int_{t_0}^{\infty}\left(\delta_i^T Q_{ii}\delta_i + \gamma_i^T(\hat{\delta}_{i,k})R_{ii}\gamma_i(\hat{\delta}_{i,k}) + \sum_{j\in N_i} u_j^T R_{ij}u_j\right)dt\\ &= \int_{t_0}^{\infty} U(\delta_i, \gamma_i(\hat{\delta}_{i,k}), u_{-i})dt\end{aligned} \tag{12}$$

Definition 1: Control laws u_i, $\forall i$ are said to be admissible if u_i are continuous, $u_i = 0$ when $\delta_i = 0$, u_i stabilize system (6) locally and values (12) are finite.

By taking derivative of (12) with respect to time t along with the trajectory of the local neighborhood tracking error δ_i, we obtain the event-triggered HJB equation,

$$\begin{aligned}&H_i(\delta_i, \nabla J_i^*, \gamma_i^*(\hat{\delta}_{i,k}), u_{-i}^*)\\ &=(\nabla J_i^*)^T\left(A\delta_i + (d_i + q_i)B_i\gamma_i^*(\hat{\delta}_{i,k}) - \sum_{j\in N_i} p_{ij}B_j u_j(t)\right)\\ &\quad + \delta_i^T Q_{ii}\delta_i + \gamma_i^*(\hat{\delta}_{i,k})^T R_{ii}\gamma_i^*(\hat{\delta}_{i,k}) + \sum_{j\in N_i} u_j^T R_{ij}u_j = 0\end{aligned} \tag{13}$$

Remark 1: Our goal is to make $\lim_{t\to\infty}\eta(t) = 0$. According to Lemma 1, one has when $\lim_{t\to\infty}\|\delta(t)\| = 0$, then $\lim_{t\to\infty}\|\eta(t)\| = 0$. Therefore, in the subsequent development, we will show the designed event-triggered control laws can make $\lim_{t\to\infty}\|\delta(t)\| = 0$.

B. Event-triggered controller design for multi-agent systems

In this subsection, an event-triggered ADP control scheme is provided for multi-agent continuous-time systems. A triggering condition is derived for each agent to guarantee the stability of the entire system. Before we develop the major results, let us start with the following assumption.

Assumption 1: The controller $\gamma(\delta_1)$ is Lipschitz continuous with respect to the gap,

$$\|\gamma_i(\delta_i) - \gamma_i(\hat{\delta}_{i,k})\| \le L\|e_{i,k}\| \tag{14}$$

where L is a positive real constant, and $e_{i,k} = \hat{\delta}_{i,k} - \delta_i$.

Theorem 1: Consider the dynamic system (6). If there exists a positive definite function J_i^* for agent i that satisfies the HJB equation (13) with $J_i^*(\delta_i(0), u_i, u_{-i}) = 0$, and the event-triggered control law is given in (11) with the triggering condition

$$\begin{aligned}&\|e_{i,k}\|^2\\ &> \frac{(1-\alpha^2)\underline{\lambda}(Q_{ii})\|\delta_i\|^2 + \|r_{ii}^T\gamma_i^*(\hat{\delta}_{i,k})\|^2 + \sum_{j\in N_i}\underline{\lambda}(R_{ij})\|u_j^*\|^2}{L^2\|r_{ii}\|^2}\\ &\triangleq \|e_{T_j}(\delta_i, \hat{\delta}_{i,k})\|^2\end{aligned} \tag{15}$$

then the event-triggered control law (11) can asympototically stabilize the system, where $\underline{\lambda}(x)$ is the minimal eigenvalue of x, and $\alpha \in (0,1)$ is the designed parameter.

Proof: With the event-triggered control law (11), the orbital derivative of J_i^* for agent i along the system trajectory can be given as

$$\begin{aligned}\dot{J}_i^* =& \left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T A\delta_i + \left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T (d_i + q_i)B_i\gamma_i^*(\hat{\delta}_{i,k})\\ &- \left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T \sum_{j\in N_i} p_{ij}B_j u_j\end{aligned} \tag{16}$$

Here, considering the HJB equation (8) and optimal control law (10) in the time-triggered ADP method for each agent, we have

$$\begin{aligned}&\left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T A\delta_i - \left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T \sum_{j\in N_i} p_{ij}B_j u_j\\ &= \frac{(d_i+q_i)^2}{4}\left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T B_i R_{ii}^{-1} B_i^T \left(\frac{\partial J_i^*}{\partial \delta_i}\right)\\ &\qquad - \delta_i^T Q_{ii}\delta_i - \sum_{j\in N_i} u_j^T R_{ij} u_j\end{aligned} \tag{17}$$

and

$$B_i^T\left(\frac{\partial J_i^*}{\partial \delta_i}\right) = -2(d_i+q_i)^{-1}R_{ii}\gamma_i^*(\delta_i) \tag{18}$$

Substitute (17) and (18) into (16), we obtain

$$\begin{aligned}\dot{J}_i^* =& \frac{(d_i+q_i)^2}{4}\left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T B_i R_{ii}^{-1} B_i^T \left(\frac{\partial J_i^*}{\partial \delta_i}\right) - \delta_i^T Q_{ii}\delta_i\\ &- \sum_{j\in N_i} u_j^T R_{ij} u_j + \left(\frac{\partial J_i^*}{\partial \delta_i}\right)^T (d_i+q_i)B_i\gamma^*(\hat{\delta}_{i,k})\\ =& \gamma_i^{*T}(\delta_i)R_{ii}\gamma_i^*(\delta_i) - \delta_i^T Q_{ii}\delta_i - \sum_{j\in N_i} u_j^T R_{ij}u_j\\ &- 2\gamma_i^{*T}(\delta_i)R_{ii}\gamma_i^*(\hat{\delta}_{i,k})\end{aligned} \tag{19}$$

Since R_{ii} is a symmetric positive definite matrix, we have R_{ii} as $R_{ii} = r_{ii}^T \cdot r_{ii}$. Therefore, we obtain

$$\begin{aligned}&\gamma_i^{*T}(\delta_i)R_{ii}\gamma_i^*(\delta_i) - 2\gamma_i^{*T}(\delta_i)R_{ii}\gamma^*(\hat{\delta}_{i,k})\\ &\quad = \|r_{ii}^T\gamma_i^{*T}(\delta_i) - r_{ii}^T\gamma_i^{*T}(\hat{\delta}_{i,k})\|^2 - \|r_{ii}^T\gamma^*(\hat{\delta}_{i,k})\|^2\end{aligned} \tag{20}$$

Substituting (20) into (19) and using the Lipschitz condition in Assumption 1, we have

$$\begin{aligned}\dot{J}_i^* =& \|r_{ii}^T\gamma_i^{*T}(\delta_i) - r_{ii}^T\gamma_i^{*T}(\hat{\delta}_{i,k})\|^2 - \|r_{ii}^T\gamma^*(\hat{\delta}_{i,k})\|^2\\ &- \delta_i^T Q_{ii}\delta_i - \sum_{j\in N_i} u_j^{*T}R_{ij}u_j^*\\ \le& L^2\|r_{ii}\|^2\|e_{i,k}\|^2 - \|r_{ii}^T\gamma_i^*(\hat{\delta}_{i,k})\|^2 - \underline{\lambda}(Q_{ii})\|\delta_i\|^2\\ &- \sum_{j\in N_i}\underline{\lambda}(R_{ij})\|u_j^*\|^2\\ =& -\alpha^2\underline{\lambda}(Q_{ii})\|\delta_i\|^2 - \Big[(1-\alpha^2)\underline{\lambda}(Q_{ii})\|\delta_i\|^2\\ &- L^2\|r_{ii}\|^2\|e_{i,k}\|^2 + \|r_{ii}^T\gamma^*(\hat{\delta}_{i,k})\|^2 + \sum_{j\in N_i}\underline{\lambda}(R_{ij})\|u_j^*\|^2\Big]\end{aligned} \tag{21}$$

where $\underline{\lambda}(x)$ is the minimal eigenvalue of x.

In order to guarantee the stability of the system, the algebraic summation of the last four terms in (21) should be positive. Consider the triggering condition (15), we know when $\|e_{i,k}\|^2 \le \|e_{T_j}(\delta_i, \hat{\delta}_{i,k})\|^2$, one obtains $(1-\alpha^2)\underline{\lambda}(Q_{ii})\|\delta_i\|^2 - L^2\|r_{ii}\|^2\|e_{i,k}\|^2 + \|r_{ii}^T\gamma^*(\hat{\delta}_{i,k})\|^2 + \sum_{j\in N_i}\underline{\lambda}(R_{ij})\|u_j^*\|^2 > 0$. Then, (21) can be further rewritten as $\dot{J}_i^* \le -\alpha^2\underline{\lambda}(Q_{ii})\|\delta_i\|^2 < 0$ for any $\delta_i \ne 0$. Therefore, $u_i^* = \gamma_i^*(\hat{\delta}_{i,k})$ can asymptotically stabilize the dynamic system (6), and hence according to Lemma 1, make all agents synchronize to the leader dynamics. The conclusion holds. ∎

From Theorem 1, we know the agents can synchronize to the leader's dynamics only with the efficiently reduced sampled data. In this process, the sampled-data systems will continuously monitor the triggering condition (15) and will be triggered to sample the new data from the dynamic system (6) when (15) is satisfied. Then the control laws are updated again based on the new sampled data. Now, we will prove that the designed event-triggered control laws can provide global Nash equilibrium solution for the multi-agent system.

Definition 2: A global Nash equilibrium solution for an N-agent system is given by N-tuple of control laws $\{u_1^*, u_2^*, \cdots, u_N^*\}$ if it satisfies

$$J_i^* \triangleq J(\delta_i(t_0), u_i^*, u_{\mathcal{G}-i}^*) \le J(\delta_i(t_0), u_i, u_{\mathcal{G}-i}^*) \tag{22}$$

where $u_{\mathcal{G}-1}$ denotes the actions of all other agents in the graph excluding agent i, i.e., $u_{\mathcal{G}-i} = \{u_j | j \in N, j \ne i\}$. The N-tuple $\{J_1^*, J_2^*, \cdots, J_N^*\}$ is called the Nash equilibrium of the N-agent system.

Theorem 2: Let the graph contains a spanning tree with at least one nonzero pinning gain. If for $\forall i$, the coupled HJB equation and the optimal event-triggered control laws are designed as (13) and (11), respectively, with the triggering condition (15), then all the agents can reach Nash equilibrium and the designed event-triggered control laws are global Nash equilibrium laws.

Proof: According to Theorem 1, we have $\delta_i(t) \to 0$, when $t \to \infty$ with the developed event-triggered control laws.

The optimal performance index $J_i^*(\delta_i(t_0),\gamma_i^*,u_{-i}^*)$ satisfies $J_i^*(0,0,0)=0$. Therefore, we have

$$\frac{dJ_i^*}{dt}=\nabla J_i^*\Big(A\delta_i+(d_i+q_i)B_i\gamma_i^*-\sum_{j\in N_i}p_{ij}B_ju_j^*\Big) \tag{23}$$

Based on (13), it becomes

$$\frac{dJ_i^*}{dt}+\delta_i^TQ_{ii}\delta_i+\gamma_i^{*T}R_{ii}\gamma_i^*+\sum_{j\in N_i}u_j^{*T}R_{ij}u_j^*=0 \tag{24}$$

which can be further rewritten as

$$J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)+\int_0^\infty\Big(\delta_i^TQ_{ii}\delta_i+\gamma_i^{*T}R_{ii}\gamma_i^*+\sum_{j\in N_i}u_j^{*T}R_{ij}u_j^*\Big)dt=0 \tag{25}$$

Therefore, performance index (7) can be described as

$$\begin{aligned}&J_i(\delta_i(0),\gamma_i,u_{-i})\\&=\int_0^\infty\Big(\delta_i^TQ_{ii}\delta_i+\gamma_i^TR_{ii}\gamma_{ii}+\sum_{j\in N_i}u_j^TR_{ij}u_j\Big)dt\\&\quad+J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)\\&\quad-\int_0^\infty\nabla J_i^*\Big(A\delta_i+(d_i+q_i)B_i\gamma_i^*-\sum_{j\in N_i}p_{ij}B_ju_j^*\Big)dt\end{aligned} \tag{26}$$

Consider the HJB equation (13) and the control law (11), we have

$$\begin{aligned}&J_i(\delta_i(0),\gamma_i,u_{-i})\\&=J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)+\int_0^\infty\Big((\gamma_i^*-\gamma_i)^TR_{ii}(\gamma_i^*-\gamma_i)\\&\quad+\sum_{j\in N_j}(u_j^*-u_j)^*R_{ij}(u_j-u_j^*)\\&\quad+(\nabla J_i)^T(d_i+q_i)B_i(\gamma_i-\gamma_i^*)\Big)dt\end{aligned} \tag{27}$$

If $\gamma_i=\gamma_i^*$ and $u_j=u_j^*$, then we obtain $J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)=J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)$. If only $u_j=u_j^*$, for $\forall\gamma_i$, equation (27) becomes

$$\begin{aligned}&J_i(\delta_i(0),\gamma_i,u_{-i}^*)\\&=J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)+\int_0^\infty\Big((\gamma_i^*-\gamma_i)^TR_{ii}(\gamma_i^*-\gamma_i)\\&\quad+(\nabla J_i)^T(d_i+q_i)B_i(\gamma_i-\gamma_i^*)\Big)dt\\&\geq J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)\end{aligned} \tag{28}$$

Then, it is clear that $J_i(\delta_i(0),\gamma_i,u_{-i}^*)\geq J_i^*(\delta_i(0),\gamma_i^*,u_{-i}^*)$ holds for all $i\in\{1,2,\cdots,N\}$. Therefore, according to Definition 2, all the agents can reach Nash equilibrium and the event-triggered control laws are global Nash equilibrium laws, which completes the proof. ∎

Remark 2: For the continuous-time systems with event-triggered controller, it is important to analyze the Zeno behavior, which is an infinite number of discrete transitions occurs in a finite time interval [41]. Consider the tracking error dynamics (4) with the triggering condition (15). According to the event-triggered mechanism, at the triggering instant, we have $\|e_{i,k+1}\|=e_{T_i}(\delta_i,\hat{\delta}_{i,k})$. Therefore, we have

$$\begin{aligned}L^2\|r_{ii}\|^2\|e_{i,k}\|^2=&(1-\alpha^2)\underline{\lambda}(Q_{ii})\|\delta_i\|^2+\|r_{ii}^T\gamma_i^*(\hat{\delta}_{i,k})\|^2\\&+\sum_{j\in N_i}\underline{\lambda}(R_{ij})\|u_j^*\|^2\\\geq&(1-\alpha^2)\underline{\lambda}(Q_{ii})\|\delta_i\|^2.\end{aligned} \tag{29}$$

Since at the kth triggering instant $e_{i,k}=0$, the time of $\|e_{i,k}\|/\|\delta_i\|$ growing from 0 to P provides a lower bound for the minimum interevent time [46], where $P=\sqrt{(1-\alpha^2)\underline{\lambda}(Q_{ii})}/(L\|r_{ii}\|)>0$, in which L is a positive real constant. This means the proposed method can achieve Zeno-free behavior.

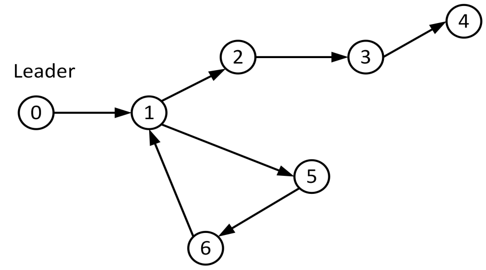

Fig. 1. Structure of communication network with one leader and six followers.

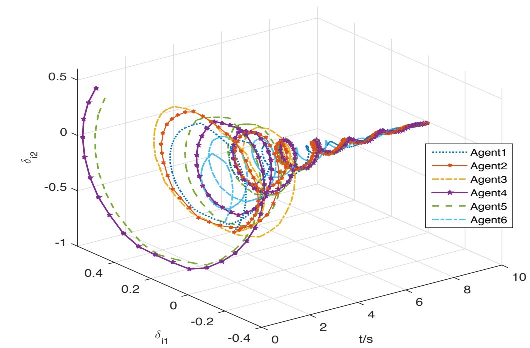

Fig. 2. Tracking errors for all the agents in 3D phase plane plot.

IV. Simulation

The simulation studies are provided in this section to show the effectiveness of the proposed method and also demonstrate the theoretical analysis in this paper. Furthermore, we compare our results with the traditional time-triggered ADP design with the same initial conditions. The results show that the proposed method can achieve competitive performance with the traditional method.

Consider a six-agent system with the communication network structure provided in Fig.1. Agent 0 is the leader with the system function as

$$\dot{x}_0(t)=Ax_0(t),\qquad A=\begin{bmatrix}0.995&-0.09983\\0.09983&0.995\end{bmatrix} \tag{30}$$

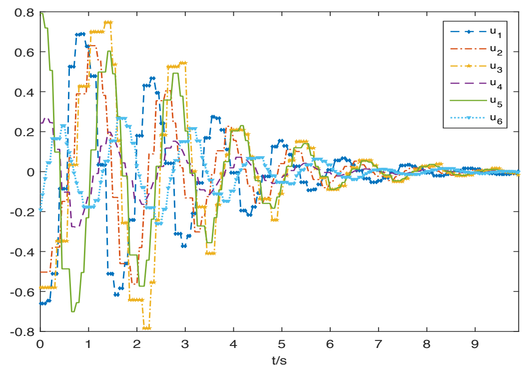

Fig. 3. Control signal trajectories for follower agents in event-triggered condition.

Note that, agent 1 can receive the information from the leader, while other follower agents can only receive the information from themselves and their neighbors. Our goal is to make all follower agents to follow the dynamics of the leader. The system functions for the follower agents are $x_i(t) = Ax_i(t) + B_iu_i(t)$ where A is the same with equation (30), $B_1 = \begin{bmatrix} 1 \\ 0.2 \end{bmatrix}$, $B_2 = \begin{bmatrix} 0 \\ 0.8 \end{bmatrix}$, $B_3 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$, $B_4 = \begin{bmatrix} 0 \\ 0.4 \end{bmatrix}$, $B_5 = \begin{bmatrix} 1 \\ 0.9 \end{bmatrix}$, $B_6 = \begin{bmatrix} 0 \\ 0.22 \end{bmatrix}$. Based on the communication network in Fig.1, we have the pinning gain as $q_1 = 1$, and $q_i = 0, i \neq 1$, and the edge weights as $p_{21} = p_{32} = p_{43} = p_{31} = p_{65} = p_{16} = 1$.

In order to save the resources, the proposed event-triggered ADP control method is implemented to this six-agent system. The weighting matrices in the performance index are selected as $Q_{ii} = I_{2\times2}$, $R_{ii} = 1$, for all $i \in \{1,2,3,4,5,6\}$, and $R_{21} = R_{32} = R_{43} = R_{31} = R_{65} = R_{16} = 1$. Neural network techniques are applied to implement the proposed method. Specifically, an action network is developed to estimate the event-triggered control law (11) based on the sampled data and a critic network is designed to approximate the value function (8) to evaluate the performance. These two neural networks work coordinately with each other to generate the results. The learning rates for both neural networks are set as $\beta_c = \beta_a = 0.005$.

Choose the triggering condition as (15) with $L = 3$, $\alpha = 0.1$. Therefore, we have the triggering condition as

$$||e_{i,k}||^2 > \frac{(1-0.1^2)||\delta_i||^2 + ||\gamma_i^*(\hat{\delta}_{i,k})||^2 + \sum_{j\in N_i} ||u_j^*||^2}{3^2}. \quad (31)$$

Only when the triggering condition (31) is satisfied, the controller is updated according to the sampled data.

The 3D phase plane plot of tracking errors for agents 1-6 are provided in Fig.2. We can observe that the tracking errors vanish after 9 seconds and therefore all the agents achieve synchronization thereafter. The trajectories of event-triggered control signal are provided in Fig.3. Under the event-triggered condition, the control signals are not updated periodically and hence, the trajectories in Fig.3 are piecewise. This means the control signals keep the same in $[\tau_{i,k}, \tau_{i,k+1})$ and are only updated when the triggering condition is satisfied. The relationships between the gap functions and the thresholds for all the follower agents are shown in Fig.4. It can be clearly observed that, for each agent, the gap is always smaller than the threshold to guarantee the stability of the system. Furthermore, to show the effectiveness of the proposed method, we compare our results with the traditional time-triggered ADP method, where the control signals are updated periodically with fixed sampling instants. The comparisons are presented in Fig.5 under the same initial conditions. We can observe that our proposed event-triggered ADP control method can achieve competitive performance with the traditional time-triggered ADP method. In this process, the required numbers of sampled data to update the control signals are provided in Fig. 6. It can be seen that the proposed methods need much less sampled data comparing with the traditional ADP method. It means by efficiently reducing the sampling instants, we can still obtain the competitive control performance.

V. Conclusion

In this paper, an event-triggered ADP method was developed for multi-agent continuous-time systems. The triggering conditions were designed for each agent to guarantee the stability and save the computation burden at the same time. The proposed method was totally data-driven and no explicit system models were required in the learning process. The stability was proved under the event-triggered condition. The simulation results verified the theoretical analysis and justified the efficiency of the proposed method.

References

[1] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial informatics*, vol. 9, no. 1, pp. 427–438, 2013.
[2] R. M. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, 2007.
[3] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on automatic control*, vol. 48, no. 6, pp. 988–1001, 2003.
[4] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.
[5] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical review letters*, vol. 75, no. 6, p. 1226, 1995.
[6] Y. Tang, H. Gao, W. Zou, and J. Kurths, "Distributed synchronization in networks of agent systems with nonlinearities and random switchings," *IEEE Transactions on Cybernetics*, vol. 43, no. 1, pp. 358–370, 2013.
[7] N. A. Lynch, *Distributed algorithms*. Morgan Kaufmann, 1996.
[8] P. Lin, W. Ren, and H. Gao, "Distributed velocity-constrained consensus of discrete-time multi-agent systems with nonconvex constraints, switching topologies, and delays," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5788–5794, 2017.
[9] J. Qin, H. Gao, and C. Yu, "On discrete-time convergence for general linear multi-agent systems under dynamic topology," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1054–1059, 2014.
[10] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on automatic control*, vol. 50, no. 5, pp. 655–661, 2005.
[11] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 71–82, 2007.

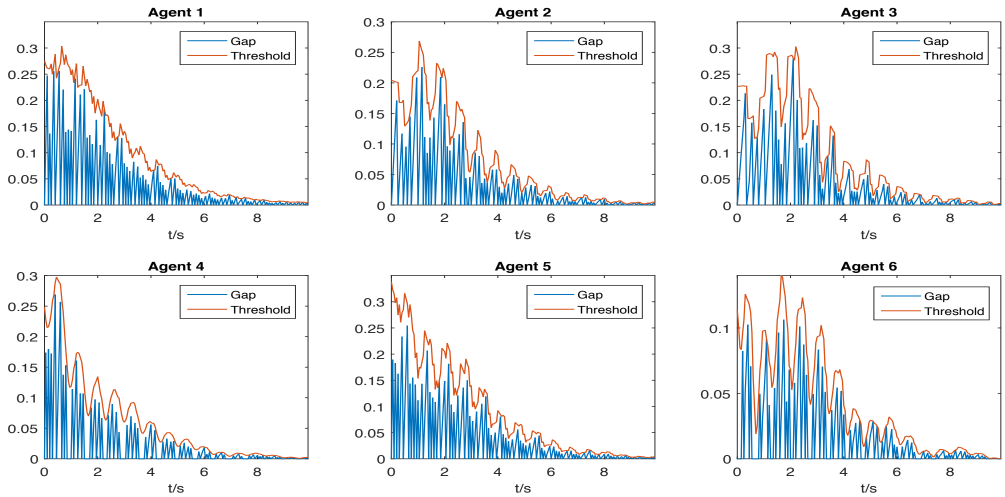

Fig. 4. Comparions of the gap functions and the thresholds.

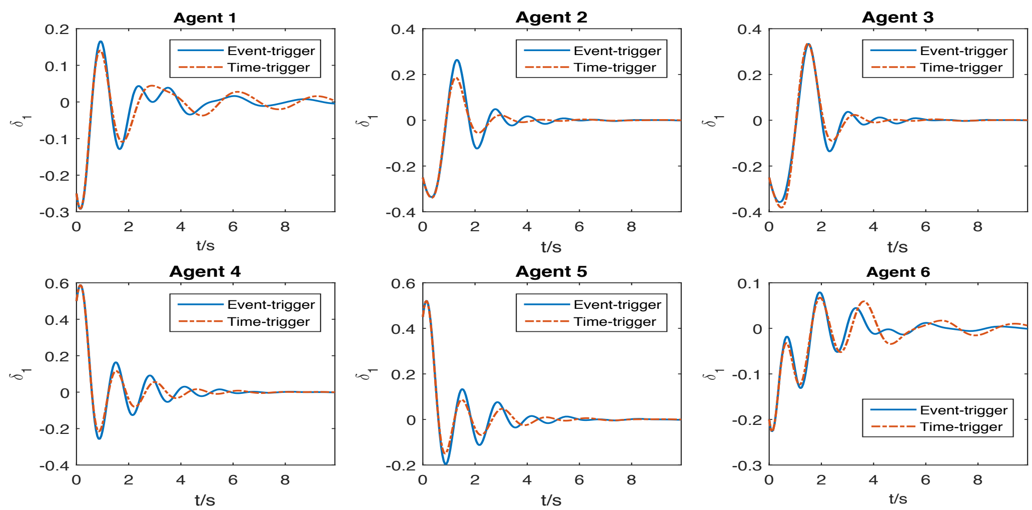

Fig. 5. Comparison of the tracking error dynamics of the event-triggered ADP method and the time-triggered ADP method.

Fig. 6. The required number of sampling data used in event-triggered ADP design and time-triggered ADP design.

[12] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on automatic control*, vol. 50, no. 2, pp. 169–182, 2005.

[13] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE transactions on automatic control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[14] Z. Feng, G. Wen, and G. Hu, "Distributed secure coordinated control for multiagent systems under strategic attacks," *IEEE transactions on cybernetics*, vol. 47, no. 5, pp. 1273–1284, 2017.

[15] D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, Massachusetts, 1995.

[16] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, ch. Neural Network Robot Control, pp. 173–219. Taylor & Francis, London, UK, 1999.

[17] P. J. Werbos, "Adp: The key direction for future research in intelligent control and understanding brain intelligence," *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, vol. 38, no. 4, pp. 898–900, 2008.

[18] Z. Hou, H. Gao, and F. L. Lewis, "Data-driven control and learning systems," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4070–4075, 2017.

[19] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive Dynamic Programming for Control: Algorithms and Stability* . London: Springer, 2013.

[20] Q. Wei, D. Liu, G. Shi, and Y. Liu, "Multibattery optimal coordination control for home energy management systems via distributed iterative adaptive dynamic programming," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 7, pp. 4203–4214, 2015.

[21] X. Yang, H. He, and X. Zhong, "Adaptive dynamic programming for robust regulation and its application to power systems," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5722–5732, 2018.

[22] Q. Wei, D. Liu, and F. L. Lewis, "Optimal distributed synchronization control for continuous-time heterogeneous multi-agent differential graphical games," *Information Sciences*, vol. 317, pp. 96–113, 2015.

[23] J. Zhang, H. Zhang, and T. Feng, "Distributed optimal consensus control for nonlinear multiagent system with unknown dynamic," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3339–3348, 2018.

[24] H. Zhang, T. Feng, H. Liang, and Y. Luo, "Lqr-based optimal distributed cooperative design for linear discrete-time multiagent systems," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 599–611, 2017.

[25] M. I. Abouheaf and F. L. Lewis, "Approximate dynamic programming solutions of multi-agent graphical games using actor-critic network structures," in *Proc. 2013 International Joint Conference on Neural Networks*, pp. 1–8, IEEE, 2013.

[26] M. I. Abouheaf, F. L. Lewis, K. G. Vamvoudakis, S. Haesaert, and R. Babuska, "Multi-agent discrete-time graphical games and reinforcement learning solutions," *Automatica*, vol. 50, no. 12, pp. 3038–3053, 2014.

[27] H. Zhang, H. Jiang, Y. Luo, and G. Xiao, "Data-driven optimal consensus control for discrete-time multi-agent systems with unknown dynamics using reinforcement learning method," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4091–4100, 2017.

[28] H. Zhang, H. Liang, Z. Wang, and T. Feng, "Optimal output regulation for heterogeneous multiagent systems via adaptive dynamic programming," *IEEE Transactions on neural networks and learning systems*, vol. 28, no. 1, pp. 18–29, 2017.

[29] H. Zhang, J. Zhang, G.-H. Yang, and Y. Luo, "Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 152–163, 2015.

[30] P. Tallapragada and N. Chopra, "On event triggered tracking for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 9, pp. 2343–2348, 2013.

[31] M. Lemmon, "Event-triggered feedback in control, estimation, and optimization," in *Networked Control Systems*, pp. 293–358, Springer, 2010.

[32] J. Zhang and G. Feng, "Event-driven observer-based output feedback control for linear systems," *Automatica*, vol. 50, no. 7, pp. 1852–1859, 2014.

[33] W. Heemels, M. Donkers, and A. Teel, "Periodic event-triggered control for linear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 847–861, 2013.

[34] Y. Zhu, D. Zhao, H. He, and J. Ji, "Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4101–4109, 2017.

[35] K. G. Vamvoudakis, "Event-triggered optimal adaptive control algorithm for continuous-time nonlinear systems," *IEEE/CAA Journal of Automatica Sinica*, vol. 1, no. 3, pp. 282–293, 2014.

[36] X. Zhong, Z. Ni, and H. He, "Event-triggered adaptive dynamic programming for continuous-time nonlinear system using measured input-output data," in *Proc. 2016 International Joint Conference on Neural Networks*, IEEE, 2016.

[37] A. Sahoo, H. Xu, and S. Jagannathan, "Neural network-based event-triggered state feedback control of nonlinear continuous-time systems," *IEEE Transactions on Neural Networks and Learning System*, vol. 27, no. 3, pp. 497 – 509, 2016.

[38] X. Zhong, Z. Ni, H. He, X. Xu, and D. Zhao, "Event-triggered reinforcement learning approach for unknown nonlinear continuous-time system," in *Proc. 2014 International Joint Conference on Neural Networks*, pp. 3677 – 3684, IEEE, 2014.

[39] X. Zhong and H. He, "An event-triggered adp control approach for continuous-time system with unknown internal states," *IEEE transactions on cybernetics*, vol. 47, no. 3, pp. 683–694, 2017.

[40] A. Sahoo, H. Xu, and S. Jagannathan, "Near optimal event-triggered control of nonlinear discrete-time systems using neurodynamic programming," *IEEE Transactions on Neural Networks and Learning System*, vol. 27, no. 9, pp. 1801 – 1815, 2016.

[41] Q. Zhang, D. Zhao, and D. Wang, "Event-based robust control for uncertain nonlinear systems using adaptive dynamic programming," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 37–50, 2018.

[42] Z. Wang, Q. Wei, and D. Liu, "A tracking control method based on event-triggered adaptive dynamic programming," in *2019 Chinese Control Conference (CCC)*, pp. 2454–2459, IEEE, 2019.

[43] W. Zhao and H. Zhang, "Distributed optimal coordination control for nonlinear multi-agent systems using event-triggered adaptive dynamic programming method," *ISA transactions*, vol. 91, pp. 184–195, 2019.

[44] G. Chen, "Pinning control and synchronization on complex dynamical networks," *International Journal of Control, Automation and Systems*, vol. 12.

[45] X. Zhong and Z. Ni, "Data-driven reinforcement learning design for multi-agent systems with unknown disturbances," in *Proc. 2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2018.

[46] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.