OPTIMAL SYNCHRONIZATION CONTROL OF MULTIPLE EULER-LAGRANGE SYSTEMS VIA EVENT-TRIGGERED REINFORCEMENT LEARNING

YUAN Xu
The Key Laboratory of Advanced Control and Optimization for Chemical Processes
Ministry of Education, East China University of Science and Technology
Shanghai 200237, China

XIN Jin, SAIWEI Wang and YANG Tang*
The Key Laboratory of Advanced Control and Optimization for Chemical Processes
Ministry of Education, East China University of Science and Technology
Shanghai 200237, China

ABSTRACT. In this paper, an event-triggered reinforcement learning-based method is developed for model-based optimal synchronization control of multiple Euler-Lagrange systems (MELSs) under a directed graph. The strategy of event-triggered optimal control is deduced through the establishment of Hamilton-Jacobi-Bellman (HJB) equation and the triggering condition is then proposed. Event-triggered policy iteration (PI) algorithm is then borrowed from reinforcement learning algorithms to find the optimal solution. One neural network is used to represent the value function to find the analytical solution of the event-triggered HJB equation, weights of which are updated aperiodically. It is proved that both the synchronization error and the weight estimation error are uniformly ultimately bounded (UUB). The Zeno behavior is also excluded in this research. Finally, an example of multiple 2-DOF prototype manipulators is shown to validate the effectiveness of our method.

1. Introduction. Recent years have witnessed compelling attention in the field of multi-agent systems (MASs) owing to their promising applications, including natural science such as bird flocking [9], fish schooling [14] and other areas involving power systems [34] and mobile robots [2]. Optimal synchronization control [36], being a popular research issue in this field, requires MASs to accomplish an agreement while minimizing the given performance cost.

Multiple Euler-Lagrange systems (MELSs) has also aroused great interest of study, since the Euler-Lagrange model is one effective way to model a huge number of mechanical systems consisting of robotic arms [21], automated vehicles [7] and spacecrafts [33]. Recently, extensive results have been studied on the synchronization problems of MELSs [24], [16], [1]. In [16], a decentralized and continuous controller is designed to realize the synchronization problem of MELSs with an added robust integral sign of the error (RISE) strategy. Besides, there exist some studies, which are concerned about the containment control and tracking control...
using sliding mode control [22], [6]. Output tracking control is also studied in [35]. Further, some researchers pay attention to the optimal synchronization control of MELSs [32], [26] when they begin to consider minimizing certain performance function. In [32], a newly proposed distributed gradient-based control algorithm is introduced to drive both the position and velocity of robotics to reach synchronization. The optimal synchronization problem under kinematic constraints, the objective of which includes the synchronization of acceleration is also studied in [26].

Reinforcement learning, a branch of artificial intelligence, aims at finding the optimal policy through continuous interactions with the external environment [28]. Instead of applying a correct action directly in the above-mentioned works [32], [26], the control policy will be learned by agents themselves through trials and errors and adapts to the environment gradually, which is the most notable benefit of reinforcement learning method. Summing up the above discussions on the optimal synchronization control problems of MELSs and the merit of reinforcement learning method constitute the first motivation of our work. Adaptive dynamic programming (ADP) [23] depending on reinforcement learning algorithm, has been a class of popular methods in addressing the optimal control problems including synchronization control problems of MASs [25], [30]. In ADP methods, neural networks are generally applied to represent the system model, reinforcement signals and control inputs. Specifically, the reinforcement signal is applied to evaluate the current control signal so that the control input can be improved gradually, which means the policy can adapt to the environment better. Both synchronization of MASs with linear and nonlinear dynamics via ADP are discussed in [25], [39], respectively. In particular, the optimal control problem is generally transformed to the solution of HJB equation, which is tough to be solved. PI [4], as one category of reinforcement learning algorithms, is known as a kind of computational intelligence to handle this issue. This algorithm mainly includes policy evaluation and policy improvement, and the two steps will iterate continuously until the convergence of the value function. Consequently, solving the HJB equation directly is avoided.

However, one has to guarantee the stability of the close-looped systems while applying reinforcement learning method. Additionally, with the increase of the dimensionality of the state, curse of dimension will be caused when using neural network as approximation of the value function.

With the rapid growth of both scale and complexity of Euler-Lagrange systems like multiple industrial manipulators, the computational burden is largely increased consequently. At the same time, with the growing number of agents in one network, the continuous interactions among agents also result in heavy communication pressure. As a result, the second motivation of our work is to reduce the above-mentioned computational and transmission burden. For the purpose of reducing the overall resource cost, event-triggered control is introduced in the field of MASs and MELSs [8], [10], [13], [11], [37]. So far as we know, most previous works have not focused on the optimal synchronization control of MELSs under event-triggered mechanism, which inspires our work. Event-triggered systems are one kind of hybrid systems, as well as impulsive systems [29], [38], [19]. Compared to time-triggered situations, with event-triggered mechanism, the policy is renewed aperiodically and will be only triggered, when the pre-designed condition is satisfied. Inspired by [39], [31], the event-triggered control design for the optimal synchronization issue of MELSs is proposed in this research, which only involves the local and neighbouring agents’ information.
We consider the reinforcement learning-based optimal synchronization control of MELs with event-triggered strategies in this research. To the best of our knowledge, this issue has not been studied in recent works yet. Specifically, the main merits are summarized as the following.

1) This paper firstly addresses the optimal synchronization control of MELs through event-triggered reinforcement learning method. The event-triggered policy iteration algorithm and the triggering condition are newly proposed for MELs and the stability of which is guaranteed according to the Lyapunov technique. Meanwhile, the Zeno behaviour is proved to be excluded in this work.

2) Compared to the consideration that the neighbouring control policies of one agent should remain constant during the triggering interval of one agent in the previous literature [39], all neighbours in this work can update their control policies in this interval, which is more reasonable in the event-triggered setting and can apply to most practical situations.

We summarize the remaining of this work as below. In section 2, fundamental necessities are introduced and formulated. Section 3 gives the triggering condition and the event-triggered PI algorithm. The implementation of PI with critic-only neural network is presented in section 4. Section 5 gives one numerical example of multiple 2-DOF prototype robots. In section 6, there involves the conclusion and outlook for the upcoming research.

2. Preliminaries and problem formulation. This part firstly gives several useful notations and fundamental concepts of graph theories. The Euler-Lagrange system and the optimal synchronization control problem of MELs will be then introduced and formulated.

2.1. Notations. Throughout this paper, $\mathbb{R}^n$ and $\mathbb{R}^{n \times m}$ represent the n-dimensional vector set and $n \times m$ real matrix set. $\mathbb{N}$ is the set of positive integers. For vector $x = [x_1, x_2, ..., x_n] \in \mathbb{R}^n$, $\|x\|$ represents the Euclidean norm of vector $x$, which is denoted as $\|x\| = \sqrt{x_1^2 + x_2^2 + ... + x_n^2}$. With a given matrix $A = [a_{ij}] \in \mathbb{R}^{N \times N}$, $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ stand for the minimum eigenvalue and the maximum eigenvalue of matrix $A$, respectively. $A > 0$ shows that matrix $A$ is positive definite. $\text{tr}(A)$ represents the trace of $A$. $\|A\|_F$ denotes the Frobenius norm of matrix $A$ which is represented as $\|A\|_F = \sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} |a_{ij}|^2}$.

2.2. Graph theories. With a directed communication graph denoted as $G = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, it contains $N$ nodes with weighted edges. $\mathcal{V} = \{1, 2, ..., N\}$ denotes a nonempty node set, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents a set of directed edges. $\mathcal{A}$ is called as the adjacency matrix. Specifically, it can be represented as $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$, in which $a_{ij} > 0$ if there exists one directed edge starting from node $j$ to node $i$ and $a_{ij} = 0$ otherwise. For node $i$, $\mathcal{N}_i = \{j \in \mathcal{V}, a_{ij} > 0\}$ is written as a set of the neighbours of node $i$. $\mathcal{N}_i$ indicates the set of agent $i$ with its neighbours. Here, self connections are excluded, i.e., $a_{ii} = 0$, $\mathcal{D} = \text{diag}\{d_1, d_2, ..., d_N\} \in \mathbb{R}^{N \times N}$ denotes the in-degree matrix, in which $d_i = \sum_{j=1}^{N} a_{ij}$. Then we define the Laplacian matrix as $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{N \times N}$, with $l_{ij} = d_i$ if $i = j$ and $l_{ij} = -a_{ij}$ otherwise.

One directed path beginning from node $i$ to node $j$ is denoted as a limited sequence of edges such as $(i, k_1), (k_1, k_2), ..., (k_l, j) \subset \mathcal{E}$. In case there exists a directed path within any two distinct nodes, then this graph is said to be strongly connected.
In this note, the directed graph which is strongly connected is considered. The choice of the graph condition will be discussed later.

2.3. Euler-Lagrange system model. Consider a system with \( N \) agents, each of which is governed by the Euler-Lagrange equation as below,

\[
M_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + G_i(q_i) = \tau_i, \quad i = 1, \ldots, N, \tag{1}
\]

in which \( q_i \in \mathbb{R}^n \) denotes the generalized configuration coordinate vector. Specifically, \( M_i(q_i) \in \mathbb{R}^{n \times n} \) indicates the generalized inertia matrix. Besides, \( C_i(q_i, \dot{q}_i)\dot{q}_i \in \mathbb{R}^n \) and \( G_i(q_i) \in \mathbb{R}^n \) represent Centrifugal/Coriolis force vector and gravitational force vector, respectively. \( \tau_i \in \mathbb{R}^n \) is the control torque implemented by the actuator.

The model information of the Euler-Lagrange model is assumed to be known here, that is, the above-mentioned \( M_i, C_i \) and \( G_i \) are available when computing. Besides, we have the following assumptions throughout this work.

Assumption 1. [16]: Both the generalized configuration coordinate \( q_i \) and the first-order derivative \( \dot{q}_i \) can be measured. As a result, its second-order derivative \( \ddot{q}_i \) can be inferred from the known model and all measurable information.

Assumption 2. [13], [17]: The inverse matrix of \( M_i(q_i) \) exists, i.e., \( M_i^{-1}(q_i) \in \mathbb{R}^{n \times n} \) exists. Besides, \( M_i^{-1}(q_i) \) is bounded. Specifically, there are positive constants \( k_{mi}, k_{M1} \) such that \( k_{mi} \leq \| M_i^{-1}(q_i) \| \leq k_{M1} \).

2.4. Problem formulation. Our goal here is to design a controller which ensures that all Euler-Lagrange agents can reach an agreement including \( q_i \) and \( \dot{q}_i \). In addition, there exists specific performance functional constraint. Hence, the following definition is given.

The local neighbourhood position synchronization error (NPSE) \( e_{1,i} \in \mathbb{R}^n \) is firstly introduced here, which is described as follows,

\[
e_{1,i} = \sum_{j \in \mathcal{N}_i} a_{ij}(q_i - q_j).
\]

According to the goal, the definition of the auxiliary synchronization error \( e_{2,i} \in \mathbb{R}^n \) is proposed as

\[
e_{2,i} = \dot{e}_{1,i} + \alpha_{1,i}e_{1,i}, \tag{2}
\]

where \( \alpha_{1,i} > 0 \) [16].

Then the dynamics of \( e_{2,i} \) is represented as

\[
\dot{e}_{2,i} = \sum_{j \in \mathcal{N}_i} a_{ij}(\ddot{q}_i - \dot{q}_j) + \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij}(\dot{q}_i - \dot{q}_j). \tag{3}
\]

From (1), the dynamics (3) can be converted into the following form

\[
\dot{e}_{2,i} = \sum_{j \in \mathcal{N}_i} a_{ij}[M^{-1}_i(\tau_i - C_i(q_i, \dot{q}_i)\dot{q}_i - G_i(q_i))]
- M^{-1}_j(\tau_j - C_j(q_j, \dot{q}_j)\dot{q}_j - G_j(q_j))] + \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij}(\dot{q}_i - \dot{q}_j), \tag{4}
\]

and we use \( E_i(q_i, \dot{q}_i) \) to represent \(-C_i(q_i, \dot{q}_i)\dot{q}_i - G_i(q_i)\) for all \( i \in \mathcal{V} \) in the remaining work.
where \( V(\text{Optimal Synchronization Control of MELS}) \) is given as follows.

\[
V_i(e_{2,i}(t_0), \tau_i, \tau_{\mathcal{K}_i}) = \int_{t_0}^{\infty} \left[ e_{2,i}^T Q_i e_{2,i} + \tau_i^T S_{ii} \tau_i + \sum_{j \in \mathcal{K}_i} \tau_j^T S_{ij} \tau_j \right] dt,
\]

where \( \tau_{\mathcal{K}_i} = \{ \tau_j | j \in \mathcal{K}_i \} \) means the control inputs of the \( i \)th agent’s neighbours. \( Q_{ii} \in \mathbb{R}^{n \times n}, Q_{ii} > 0 \) denotes the weighted matrix of the synchronization error in the overall performance cost. Meanwhile, \( S_{ii} \in \mathbb{R}^{n \times n}, S_{ij} \in \mathbb{R}^{n \times n} \) are both positive definite matrices [5].

To obtain the control input which results in the optimal synchronization control, a definition of admissible synchronization control policy is introduced here.

**Definition 2.1.** (Admissible Synchronization Control [20]) A control input \( \tau_i \) is defined as an admissible synchronization control policy with regard to (5) on a given set \( \mathcal{Y} \in \mathbb{R}^n \) if \( \tau_i \) is continuous with \( \tau_i(0) = 0 \), drives the auxiliary synchronization error \( e_{2,i} \) to zero locally and the local performance function (5) is finite, i.e., \( J_i < +\infty \).

Further, the reinforcement learning signal which corresponds to (5), i.e., the value function is defined as follows,

\[
V_i(e_{2,i}(t)) = \int_{t}^{\infty} \left[ e_{2,i}^T Q_i e_{2,i} + \tau_i^T S_{ii} \tau_i + \sum_{j \in \mathcal{K}_i} \tau_j^T S_{ij} \tau_j \right] dt.
\]

With the definition of value function (6), the admissible control policy can be designed to drive all agents to reach synchronization and minimize the value function (6) at the same time. Hence, the definition of optimal synchronization control of MELSs is given as follows.

**Definition 2.2.** (Optimal Synchronization Control of MELSs) For \( i = 1, 2, ..., N \), the local admissible control input \( \tau_i^* \) is designed to drive all Euler-Lagrange agents to reach synchronization, i.e., \( \lim_{t \to \infty} ||q_a(t) - q_b(t)||_2 = 0, \lim_{t \to \infty} ||\dot{q}_a(t) - \dot{q}_b(t)||_2 = 0 \), for \( \forall a, b \in \mathcal{V} \) and minimize the value function \( V_i \).

Combining (2) and (6), for agent \( i \), we have the definition of the Hamiltonian function [3] as

\[
H_i(e_{2,i}, V_{e_{2,i}}, \tau_i, \tau_{\mathcal{K}_i}) = e_{2,i}^T Q_i e_{2,i} + \tau_i^T S_{ii} \tau_i + \sum_{j \in \mathcal{K}_i} \tau_j^T S_{ij} \tau_j + V_{e_{2,i}}^T
\]

\[
\left\{ \sum_{j \in \mathcal{K}_i} a_{ij} M_i^{-1}(\tau_i + E_i(q_i, \dot{q}_i)) - M_j^{-1}(\tau_j + E_j(q_j, \dot{q}_j)) \right\} + \alpha_{1,i} \sum_{j \in \mathcal{K}_i} a_{ij}(\dot{q}_i - \dot{q}_j),
\]

where \( V_{e_{2,i}} = \partial V_i / \partial e_{2,i} \) represents the partial derivative of \( V_i(e_{2,i}(t)) \) regarding to \( e_{2,i} \).

By applying the stationary condition, i.e., \( \partial H_i(e_{2,i}) / \partial e_{2,i} = 0 \), one can obtain the optimal control as follows,

\[
\tau_i^* = -\frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T V_{e_{2,i}}^*.
\]
Substituting the optimal control input (8) into (7), the local time-triggered coupled HJB equation will be then obtained as

\[
H_i(e_{2,i}, V_{e_{2,i}}^*, \tau_i^*, \tau_{0}^*) \\
= e_{2,i}^T Q_{ii} e_{2,i} + \left(-\frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T V_{e_{2,i}}^* \right)^T S_{ii} \left( -\frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T V_{e_{2,i}}^* \right) \\
+ \sum_{j \in N_i} \left(-\frac{1}{2} d_j S_{jj}^{-1}(M_j^{-1})^T V_{e_{2,j}}^* \right)^T S_{ij} \left(-\frac{1}{2} d_j S_{jj}^{-1}(M_j^{-1})^T V_{e_{2,j}}^* \right) \\
+ V_{e_{2,i}}^* \left( \sum_{j \in N_i} a_{ij} \left[M_i^{-1}\left(-\frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T V_{e_{2,i}}^* + E_i(q_i, \hat{q}_i) \right) \right] + \alpha_{1,i} \sum_{j \in N_i} a_{ij}(\hat{q}_i - \hat{q}_j) \right) \\
= 0.
\]

With the solution \( F_i(e_{2,i}) \) to the local coupled HJB equation (9) and the control policy (8), the asymptotical stability of \( e_{2,i} \)-dynamics can be guaranteed. Moreover, for MELSs, \( \{\tau_1^*, \ldots, \tau_N^*\} \) constitutes the interactive Nash equilibrium (INE) policy profile [30, 20], and the definition of INE and the according proof can be seen in the above works. At the same time, the game values are \( J_i^* (e_{2,i}(t_0), \tau_i^*, \tau_{0}^*) = F_i(e_{2,i}(t_0)), \forall i. \) The similar proof is omitted here for brevity.

**Remark 1.** The strong connectivity condition of the graph in this work is a sufficient and necessary condition to the existence of the INE policy profile [30]. Further, one strongly connected graph can be reduced to that the graph has a directed spanning tree. We will consider the graph condition of directed spanning tree in our future work.

3. **Design of event-triggered reinforcement learning method.** In the traditional time-triggered control system, the system bears huge transmission data and computation resources resulting from the continuous data-sampling. In this section, a reinforcement learning algorithm and specifically policy iteration will be introduced for the optimal synchronization control of MELSs with event-triggered mechanism.

3.1. **Event-triggered setting.** With event-triggered setting, the designed controller of agent \( i \) is only updated at a sequence of triggering instants \( \{t_{i,k}\}_{k=0}^\infty \), with \( t_{i,k} < t_{i,k+1}, k \in \mathbb{N}. \) We assume that the first moment is triggered, i.e., \( t_{i,0} = t_0 \), and \( t_{i,k} \) represents the \( k \)th triggering instant of the \( i \)th agent.

For the \( i \)th agent, the event-triggered error vector \( \delta_i(t) \) is defined as,

\[
\delta_i(t) = \hat{e}_{2,i}(t_{i,k}) - e_{2,i}(t), \forall t \in [t_{i,k}, t_{i,k+1}),
\]

where \( e_{2,i}(t) \) denotes the synchronization error of agent \( i \) at current and \( \hat{e}_{2,i}(t_{i,k}) \) denotes the sampled error vector. At the triggering instants, the event-triggered error \( \delta_i(t) \) will be reset to zero, and the sampled data keeps constant by application of zero-order holder(ZOH) during the triggering interval \( (t_{i,k}, t_{i,k+1}), k \in \mathbb{N}. \)

The triggering instants of agent \( i \) are decided by the violation of the following condition,

\[
\|\delta_i(t)\| < \delta^h_i(t),
\]

in which \( \delta^h_i(t) \) denotes the time-varying threshold of event-triggered setting, and we will introduce it later.
where $\hat{\tau}_j$ denotes the event-triggered control input of agent $j$, i.e., $\hat{\tau}_j = \tau_j(\hat{e}_{2,i}(t_{j,k}))$, $j \in \mathcal{N}_i$.

Accordingly, the local event-triggered coupled HJB equation of agent $i$ is

$$H_i(e_{2,i}, V_{e_{2,i}}^{*}, \hat{\tau}_i^*, \hat{\tau}_j) = e_{2,i}^T Q_i e_{2,i} + \hat{\tau}_i^T S_i \hat{\tau}_i^* + \sum_{j \in \mathcal{N}_i} \hat{\tau}_j^T S_{ij} \hat{\tau}_j^* + V_{e_{2,i}}^T$$

$$= \left\{ \sum_{j \in \mathcal{N}_i} a_{ij} [M_j^{-1}(\hat{\tau}_j^* + E_j(q_j, \dot{q}_j))] - M_j^{-1}(\hat{\tau}_j^* + E_j(q_j, \dot{q}_j))] + \alpha_1 i \sum_{j \in \mathcal{N}_i} a_{ij}(\dot{q}_i - \dot{q}_j) \right\}$$

$$= 0.$$  \hfill (11)

Hence, the according optimal local event-triggered synchronization control is then obtained as follows,

$$\hat{\tau}_i^* = -\frac{1}{2} d_i S_i^{-1}(M_i^{-1})^T V_{e_{2,i}}^*.$$  \hfill (12)

**Assumption 3 [31].** There exists one value $P$ which satisfies $P > 0$ so that

$$\|\tau_i(e_{2,i}(t)) - \tau_i(\hat{e}_{2,i}(t_{i,k}))\| \leq P\|d_i\|,$$  

which indicates the controller for the Euler-Lagrange system is Lipschitz continuous.

### 3.2. Event-triggered PI algorithm

With the event-triggered setting, the local event-triggered coupled HJB equation (11) for agent $i$ is hard to solve in an analytical manner. Though the form of the optimal policy input is deduced, the optimal $V_{e_{2,i}}^*$ is unknown unless the HJB function is solved. To obtain $V_{e_{2,i}}^*$, PI algorithm is used to seek for the solution. PI generally involves two main steps: policy evaluation plus policy improvement. These two procedures will continuously iterate until the value function converges, that is to say the second step will not improve the control policy anymore. Specifically, the event-triggered policy iteration algorithm for MELSs is given in Algorithm 1.

**Theorem 3.1.** Under the event-triggered mechanism and by using Algorithm 1, then the control policy of each agent $\hat{\tau}_i^{(h)}$ will converge to the optimal one $\hat{\tau}_i^*$, i.e., $\hat{\tau}_i^{(h)} \to \hat{\tau}_i^*$, \(\forall i \in \mathcal{V}\), as $h \to \infty$.

**Proof.** From (13), we can deduce as

$$\dot{V}_i^{(h+1)}(e_{2,i}, \hat{\tau}_i^{(h)}, \hat{\tau}_j^{(h)})$$

$$= -e_{2,i}^T Q_i e_{2,i} - (\hat{\tau}_i^{(h)})^T S_i \hat{\tau}_i^{(h)} + \sum_{j \in \mathcal{N}_i} (\hat{\tau}_j^{(h)})^T S_{ij} \hat{\tau}_j^{(h)},$$
Algorithm 1 Model-based Policy Iteration Algorithm with event-triggered mechanism

**Step 1 (Initialization):** Initialize admissible control policy for all agents $\hat{x}_i^{(0)}$, $\forall i$ and set the initial iteration time $h = 0$;

**Step 2 (Policy Evaluation):** Solve the event-triggered HJB equation for $V_{e_2,i}^{(h+1)}$ at the triggering instant of agent $i$ as

$$H_i(e_{2,i}, V_{e_2,i}^{(h+1)}, \hat{x}_i^{(h)}, \hat{x}_i^{(h-1)}) = e_{2,i}^T Q_{ii} e_{2,i} + (\hat{x}_i^{(h)})^T S_{ii} \hat{x}_i^{(h)} + \sum_{j \in \mathcal{N}_i} (\hat{x}_j^{(h)})^T S_{ij} \hat{x}_j^{(h)} + V_{e_2,i}^{(h+1)} \dot{e}_{2,i} = 0. \tag{13}$$

**Step 3 (Policy Improvement):**

$$\hat{x}_i^{(h+1)} = \frac{1}{2} d_i S_{ii}^{-1} (M_{ii}^{-1})^T V_{e_2,i}^{(h+1)}. \tag{14}$$

Back to Step 2 until the following condition for convergence satisfies,

$$\|V_i^{(h+1)} - V_i^{(h)}\| \leq \kappa,$$

where $\kappa$ denotes an extremely small value.

and

$$\dot{V}_i^{(h)}(e_{2,i}, \hat{x}_i^{(h-1)}, \hat{x}_i^{(h-1)}) = -e_{2,i}^T Q_{ii} e_{2,i} - (\hat{x}_i^{(h-1)})^T S_{ii} \hat{x}_i^{(h-1)} - \sum_{j \in \mathcal{N}_i} (\hat{x}_j^{(h-1)})^T S_{ij} \hat{x}_j^{(h-1)}$$

Then,

$$\dot{V}_i^{(h)} - \dot{V}_i^{(h+1)} = -(\hat{x}_i^{(h-1)})^T S_{ii} \hat{x}_i^{(h-1)} - \sum_{j \in \mathcal{N}_i} (\hat{x}_j^{(h-1)})^T S_{ij} \hat{x}_j^{(h-1)} + (\hat{x}_i^{(h)})^T S_{ii} \hat{x}_i^{(h)}$$

$$+ \sum_{j \in \mathcal{N}_i} (\hat{x}_j^{(h)})^T S_{ij} \hat{x}_j^{(h)}$$

$$= (\hat{x}_i^{(h-1)})^T S_{ii} \hat{x}_i^{(h-1)} - 2(\hat{x}_i^{(h-1)})^T S_{ii} (\hat{x}_i^{(h)} - \hat{x}_i^{(h-1)})$$

$$+ \sum_{j \in \mathcal{N}_i} (\hat{x}_j^{(h-1)})^T S_{ij} (\hat{x}_j^{(h+1)} - \hat{x}_j^{(h-1)}) + \sum_{j \in \mathcal{N}_i} 2(\hat{x}_j^{(h-1)})^T S_{ij} (\hat{x}_j^{(h)} - \hat{x}_j^{(h-1)}).$$

The sufficient condition of $(\dot{V}_i^{(h)} - \dot{V}_i^{(h+1)}) \leq 0$ is to guarantee the following two inequalities hold,

$$\Delta \hat{x}_i^T S_{ii} \Delta \hat{x}_i \leq -2(\hat{x}_i^{(h-1)})^T S_{ii} \Delta \hat{x}_i,$$

$$\sum_{j \in \mathcal{N}_i} \Delta \hat{x}_j^T S_{ij} \Delta \hat{x}_j \leq - \sum_{j \in \mathcal{N}_i} 2(\hat{x}_j^{(h-1)})^T S_{ij} \Delta \hat{x}_j,$$

where $\Delta \hat{x}_i = \hat{x}_i^{(h)} - \hat{x}_i^{(h-1)} \in \mathbb{R}^n$, $\Delta \hat{x}_j = \hat{x}_j^{(h)} - \hat{x}_j^{(h-1)} \in \mathbb{R}^n$. 
From the form of control policy and Assumption 2, and let $q_{ij} = \lambda_{\min}(S^{-1}_{jj}S_{ij})$, the sufficient condition becomes
\[ \lambda_{\max}(S_{ii})\|\Delta \hat{\tau}_i\| \leq k_{mi}d_i\|V_{e_{2,i}}^{(h-1)}\|, \]
\[ \lambda_{\max}(S_{ij})\|\Delta \hat{\tau}_j\| \leq k_{mj}d_j\|V_{e_{2,j}}^{(h-1)}\|. \]

Regarding to the definition of $V_i$, it can be deduced that $V_i^{(\infty)} = 0$. By integrating $\dot{V}_i^{(h)} \leq \dot{V}_i^{(h+1)}$ over $[t, \infty)$, then it follows that
\[ V_i^{(h+1)} \leq V_i^{(h)}, \]
which indicates that it is a non-increasing function, which is lower bounded by zero.

According to (6) and the optimality principle [18], it can be easily obtained that
\[ V_i^{(h)} \geq \int_{t_0=0}^\infty (e_{2,i}^TQ_{ii}e_{2,i} + (\hat{\tau}_i^*)^T S_{ii} \hat{\tau}_i^* + \sum_{j \in N_i} (\hat{\tau}_j^*)^T S_{ij} \hat{\tau}_j^*)d\tau = V_i^*, \]
where $h$ tends to $\infty$, $V_i^{(\infty)} \geq V_i^*$. As $V_i^{(\infty)} \leq V_i^*$, it can be inferred that $V_i^{(\infty)} = V_i^*$, which means $V_i$ will have a convergence to the optimal one. Hence, one can easily get $\dot{\tau}_i^{(\infty)} = \dot{\tau}_i^*$.

In the policy evaluation step of PI algorithm, it is still hard to solve the event-triggered HJB equation (13) analytically, i.e., the first-order derivative of $V_i$ with regard to $e_{2,i}$. Inspired by the strong approximation capability of NN, NN-based implementation of the PI algorithm will be proposed in the next section, in which one neural network is utilized to represent the value function and the control policy can be then obtained.

4. Implementation of event-triggered PI algorithm via critic-only neural network.

4.1. Critic-only neural network implementation. Actor-critic method, being popular in reinforcement learning, uses two separate neural networks, which represent the value function and control input, respectively. However, with model information and value function available, the control signal will be easily obtained by (12) such that only one critic neural network is sufficient in this work. Furthermore, with the event-triggered scheme, the weights of the critic neural network will only be renewed at triggering moments.

In accordance with the great universal approximation performance of the neural network, the value function of agent $i$ can be expressed as a three-layer neural network
\[ V_i^*(e_{2,i}) = W_{c,i}^T \phi_{c,i}(\sigma_{c,i}^T e_{2,i}) + \epsilon_{c,i}, \]
where $\sigma_{c,i} \in \mathbb{R}^{n \times n}$ represents the weight matrix between the input to the hidden layer, which is always set with identity matrix $I$ for brevity. $\phi_{c,i}(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$ represents the activation function. The concrete form of $\phi_{c,i}(\cdot)$ will be given in the simulation part. $W_{c,i} \in \mathbb{R}^m$ denotes the weight matrix from the hidden to the output layer. $\epsilon_{c,i}$ denotes the reconstruction error of the critic network.

Hence, the first-order derivative of $V_i(e_{2,i})$ regarding to $e_{2,i}$ is
\[ V_{e_{2,i}}^* = \frac{\partial (W_{c,i}^T \phi_{c,i})}{\partial e_{2,i}} + \frac{\partial \epsilon_{c,i}}{\partial e_{2,i}} = \nabla \phi_{c,i}^T W_{c,i} + \nabla \epsilon_{c,i}, \]
where \( \nabla \phi_{c,i}^{T} = \frac{\partial \phi_{c,i}^{T}}{\partial \sigma_{c,i}}, \nabla \epsilon_{c,i} = \frac{\partial \epsilon_{c,i}}{\partial \sigma_{c,i}}. \)

Hence, the local time-triggered coupled HJB equation (9) becomes

\[
H_{i} = W_{c,i}^{T} \nabla \phi_{c,i} \dot{e}_{2,i} + e_{2,i}^{T} Q_{ii} e_{2,i} + (\tau_{i}^{*})^{T} S_{ii} \tau_{i}^{*} + \sum_{j \in \mathcal{N}_{i}} (\tau_{j}^{*})^{T} S_{ij} \tau_{j}^{*} = \epsilon_{c,i},
\]

where \( \epsilon_{c,i} = -(\nabla \epsilon_{c,i})^{T} e_{2,i} \) represents the residual error and we have \( \hat{\epsilon}_{c,i} \) and \( \nabla \hat{\epsilon}_{c,i} \) with the event-triggered mechanism.

Then, the estimated value function of agent \( i \) during \( t \in [t_{i,k}, t_{i,k+1}) \) is as follows,

\[
V_{i}(e_{2,i}) = \hat{W}_{c,i}^{T} \phi_{c,i}(\sigma_{c,i}^{T} e_{2,i}),
\]

where \( \hat{W}_{c,i} \) denotes the estimated weight matrix between the hidden to the output layer at event-triggering instants.

Therefore, the event-triggered HJB equation error \( \xi_{c,i} \) becomes

\[
\dot{\xi}_{c,i} = \hat{W}_{c,i}^{T} \nabla \phi_{c,i} \dot{e}_{2,i} + e_{2,i}^{T} Q_{ii} e_{2,i} + (\hat{\tau}_{i})^{T} S_{ii} \hat{\tau}_{i} + \sum_{j \in \mathcal{N}_{i}} (\hat{\tau}_{j})^{T} S_{ij} \hat{\tau}_{j}.
\]

For the critic neural network, the objective function to minimize can be designed as

\[
E_{c,i} = \frac{1}{2} \hat{\xi}_{c,i}^{T} \hat{\xi}_{c,i}.
\] (15)

The estimated weights are only updated at discrete triggering instants \( t_{i,k} \) and hold constant during triggering interval \( t \in [t_{i,k}, t_{i,k+1}) \) under the event-triggered control.

From the objective function (15), the weight update law can be deduced as follows

\[
\begin{align*}
\dot{\hat{W}}_{c,i} &= 0, t \in (t_{i,k}, t_{i,k+1}), \\
\dot{W}_{c,i}^+ &= \hat{W}_{c,i} - l_{c} K_{i} (K_{i}^{T} \hat{W}_{c,i} + Y_{i}(e_{2,i}, \hat{\tau}_{i}, \hat{\tau}_{\mathcal{N}_{i}})), t = t_{i,k},
\end{align*}
\] (16)

where \( \hat{W}_{c,i} \) denotes the renewed estimated weight at the triggering time \( t_{i,k}, l_{c} \) denotes the critic network's learning rate, \( K_{i} = K_{ii} / (K_{ii}^{1/2} K_{ii} + 1)^{2} \) with \( K_{ii} = \nabla \phi_{c,i}^{T} \dot{e}_{2,i} \), and \( Y_{i}(e_{2,i}, \hat{\tau}_{i}, \hat{\tau}_{\mathcal{N}_{i}}) = e_{2,i}^{T} Q_{ii} e_{2,i} + (\hat{\tau}_{i})^{T} S_{ii} \hat{\tau}_{i} + \sum_{j \in \mathcal{N}_{i}} (\hat{\tau}_{j})^{T} S_{ij} \hat{\tau}_{j} \). It should be noticed that the identity matrices between the input layer and the hidden layer keep instant.

Then, with the estimated value function, we can obtain the control policy under event-triggered mechanism as

\[
\hat{\tau}_{i} = -\frac{1}{2} d_{i}^{T} s_{ii}^{-1} (M_{i}^{-1})^{T} V_{e_{2,i}} = -\frac{1}{2} d_{i}^{T} s_{ii}^{-1} (M_{i}^{-1})^{T} \nabla \phi_{c,i}^{T} \hat{W}_{c,i}.
\]

With the application of ZOH and the update of the weight of the critic neural network, the control sequence is continuous. In particular, the control policy is a piecewise constant function, which keeps constant during the triggering interval.

Moreover, with the definition of the estimation error \( \tilde{W}_{c,i} = \hat{W}_{c,i} - W_{c,i} \) of the critic NN, the first-order derivative of \( \tilde{W}_{c,i} \) becomes

\[
\begin{align*}
\dot{\tilde{W}}_{c,i} &= 0, t \in (t_{i,k}, t_{i,k+1}), \\
\dot{\tilde{W}}_{c,i}^+ &= \tilde{W}_{c,i} - l_{c} K_{i} (K_{i}^{T} \tilde{W}_{c,i} + \tilde{\epsilon}_{c,i}), t = t_{i,k}.
\end{align*}
\] (17)

Next, one has to prove the stability of the system which contains two situations: event-triggering instant and the triggering interval. At the same time, the Zeno
behavior can be avoided in this work, which will be proved later. Before the proof in the next part, we will firstly give the assumption as follows.

**Assumption 4.** In the critic network, the weights \( W_{c,i} \), the activation function \( \phi_{c,i} \) and reconstruction error \( \epsilon_{c,i} \) are all bounded by positive constants \( W_m, \phi_m \) and \( \epsilon_m \), i.e., \( \|W_{c,i}\| \leq W_m, \|\phi_{c,i}\| \leq \phi_m, \|\epsilon_{c,i}\| \leq \epsilon_m \). Also, the critic network’s residual error \( \epsilon_{c,i} \) is bounded by \( \epsilon_{cm} > 0 \), i.e., \( \|\epsilon_{c,i}\| \leq \epsilon_{cm} \). The gradients of residual error and the activation function are also bounded by positive constants \( \nabla \epsilon_m \) and \( \nabla \phi_m \), i.e., \( \|\nabla \epsilon_{c,i}\| \leq \nabla \epsilon_m, \|\nabla \phi_{c,i}\| \leq \nabla \phi_m \). Under the event-triggered setting, we have the same assumption, which is omitted here for brevity.

The activation function \( \phi_{c,i} \) of the critic network should maintain the persistence of excitation (PE) condition to make the estimated weights converges eventually [27].

**Assumption 5.** From the PE condition, it can be inferred that both \( K_{1i} \) and \( K_i \) are bounded such that \( \|K_{1i}\| \leq K_{ni}, \|K_i\| \leq K_{Mi} \), with \( K_{ni} > 0 \) and \( K_{Mi} > 0 \).

### 4.2. Stability analysis of neural-network-based event-triggered PI algorithm

Firstly, the definition of uniformly ultimately bounded (UUB) is reviewed here.

**Definition 4.1.** (Uniformly Ultimately Bounded) With the solution \( x \) of one differential system and a given set which contains the origin \( W \subset \mathbb{R}^n \), for all \( x(t_0) = x^0 \in W \), it is said to be UUB in case there has a bound \( B \) which is positive and a non-negative \( T(x^0, B) \) so that

\[
\|x(t)\| \leq B, \forall t \geq t_0 + T.
\]

**Theorem 4.2.** Considering \( e_{2,i} \)-dynamics (10), the given value function (6) and Assumptions 1-5 with the event-triggered policy (12) under the event-triggered update of estimated weights (16), both the dynamics of local synchronization error \( e_{2,i} \) (10) and the weight estimation error \( \tilde{W}_{c,i} \) of the target weight \( W_{c,i} \) are UUB when the event-triggering condition which requires information of both agent \( i \) and the neighbours satisfies as follows,

\[
\|\delta_i\|^2 \leq \frac{(1 - \gamma_i^2)\lambda_{\min}(Q_{ii})\|e_{2,i}\|^2 + \|s_i^T \tau_i\|^2 + \sum_{j \in K_i} \lambda_{\min}(S_{ij})\|\tau_j\|^2}{P^2\|s_i\|^2}.
\]

where \( s_i s_i^T = S_{ii} \), \( s_i \) represents a matrix which is semi-positive definite, which depends on the choice of \( S_{ii} \) and \( \gamma_i \in (0, 1) \) is a constant parameter.

**Proof.** The proof will be divided into the next two situations.

**Situation 1:** Event-triggering interval, \( t \in (t_{i,k}, t_{i,k+1}), \forall i \in \mathcal{V} \).

Choose the Lyapunov candidate function as follows,

\[
L^i = L^i_1 + L^i_2,
\]

where \( L^i_1 = V^*_i(e_{2,i}) \), and \( L^i_2 = \frac{1}{l_c} \text{tr}(\tilde{W}_{c,i}^T \tilde{W}_{c,i}) \). Next, the first-order derivative of the \( L^i \) goes

\[
\dot{L}^i = \dot{L}^i_1 + \dot{L}^i_2 = \dot{V}^*_i(e_{2,i}) + \frac{2}{l_c} \text{tr}(\tilde{W}_{c,i}^T \dot{\tilde{W}}_{c,i}).
\]
As $\tilde{W}_{e,i} = 0$, we can get $\tilde{L}_2 = 0$. Hence,

$$\dot{L}_1 = V^*_i(e_{2,i}) = V^{*T}_{e2,i} \dot{e}_{2,i}$$

$$= V^{*T}_{e2,i} \left\{ \sum_{j \in \mathcal{N}_i} a_{ij} [M_i^{-1}(\hat{\tau}_i^* + E_i(q_i, \hat{q}_i)) - M_j^{-1}(\hat{\tau}_j^* + E_j(q_j, \hat{q}_j))] + \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij}(\hat{q}_i - \hat{q}_j) \right\} \tag{20}$$

Using the local coupled time-triggered HJB equation (9), it follows that

$$V^{*T}_{e2,i} \left\{ \sum_{j \in \mathcal{N}_i} a_{ij} [M_i^{-1}E_i(q_i, \hat{q}_i) - M_j^{-1}E_j(q_j, \hat{q}_j)] + \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij}(\hat{q}_i - \hat{q}_j) \right\}$$

$$= -e^{T}_{2,i}Q_{ii}e_{2,i} - \tau_i^{*T}S_{ii}\hat{\tau}_i^* - \sum_{j \in \mathcal{N}_i} \tau_j^{*T}S_{ij}\hat{\tau}_j^* - V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}\tau_j^*$$

$$+ V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}\hat{\tau}_j^*. \tag{21}$$

Substituting (21) into (20), (20) can be converted into the form as below,

$$V^*_i(e_{2,i}) = V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_i^{-1}\hat{\tau}_i^* - e^{T}_{2,i}Q_{ii}e_{2,i} - \tau_i^{*T}S_{ii}\hat{\tau}_i^* - \sum_{j \in \mathcal{N}_i} \tau_j^{*T}S_{ij}\hat{\tau}_j^*$$

$$- V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_i^{-1}\tau_i^* + V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}(\tau_j^* - \hat{\tau}_j^*).$$

Combining optimal control policy (8) and $d_i = \sum_{j=1}^{N} a_{ij}$, one can get

$$V^{*T}_{e2,i} \sum_{j \in \mathcal{N}_i} a_{ij}M_i^{-1} = -2\tau_i^{*T}S_{ii} = -2\tau_i^{*T}S_{ii}.$$

Hence, the first-order derivative of $V_i(e_{2,i})$ becomes

$$\dot{V}_i^*(e_{2,i}) = -2\tau_i^{*T}S_{ii}\hat{\tau}_i^* - e^{T}_{2,i}Q_{ii}e_{2,i} - \tau_i^{*T}S_{ii}\tau_i^* - \sum_{j \in \mathcal{N}_i} \tau_j^{*T}S_{ij}\tau_j^*$$

$$+ 2\tau_i^{*T}S_{ii}\tau_i^* - \frac{2}{d_i} \tau_i^{*T}S_{ii}M_i \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}(\tau_j^* - \hat{\tau}_j^*)$$

$$= \tau_i^{*T}S_{ii}\tau_i^* - 2\tau_i^{*T}S_{ii}\hat{\tau}_i^* - e^{T}_{2,i}Q_{ii}e_{2,i} - \sum_{j \in \mathcal{N}_i} \tau_j^{*T}S_{ij}\tau_j^*$$

$$- \frac{2}{d_i} \tau_i^{*T}S_{ii}M_i \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}(\tau_j^* - \hat{\tau}_j^*)$$

$$= \tau_i^{*T}S_{ii}\tau_i^* - 2\tau_i^{*T}S_{ii}\hat{\tau}_i^* - e^{T}_{2,i}Q_{ii}e_{2,i} - \sum_{j \in \mathcal{N}_i} \tau_j^{*T}S_{ij}\tau_j^*$$

$$+ \tau_i^{*T}C_i \sum_{j \in \mathcal{N}_i} a_{ij}M_j^{-1}(\tau_j^* - \hat{\tau}_j^*), \tag{22}$$

where $C_i = -\frac{2}{d_i} S_{ii} M_i$.

In (22), combined with Assumption 3, then it becomes

$$\tau_i^{*T}S_{ii}\hat{\tau}_i^* - 2\tau_i^{*T}S_{ii}\tau_i^* = \| s_i^T \hat{\tau}_i^* - s_i^T \tau_i^* \|^2 - \| s_i^T \hat{\tau}_i^* \|^2 \leq P^2 \| s_i^T \|^2 \| \delta_i \|^2 - \| s_i^T \hat{\tau}_i^* \|^2.$$
Hence, (20) becomes

\[
\dot{V}_i^*(e_{2,i}) \leq P^2\|s_i^T\|^2 \|\delta_i\|^2 - \gamma_i^2 \lambda_{\text{min}}(Q_{ii}) \|e_{2,i}\|^2 - (1 - \gamma_i^2)\lambda_{\text{min}}(Q_{ii})\|e_{2,i}\|^2 \\
- \sum_{j \in N_i} \lambda_{\text{min}}(S_{ij})\|\tau_j\|^2 + \|\tau_j^*T\|C_i \sum_{j \in N_i} a_{ij}\|M_j^{-1}\| (\tau_j^* - \hat{\tau}_j)\| - \|s_i^T\hat{\tau}_j\|^2.
\]

(23)

With the neural network approximating value function, let \(z_i = \frac{1}{2}d_i S_{ii}^{-1} (M_i^{-1})^T \in \mathbb{R}^{n \times n}\), \(\tau_i^* = \tau_i + z_i (\nabla \phi_{c,i}^T \tilde{W}_{c,i} - \nabla \epsilon_{c,i}) \in \mathbb{R}^n\), \(\hat{\tau}_i = \tau_i + \hat{z}_i (\nabla \phi_{c,i}^T \tilde{W}_{c,i} - \nabla \epsilon_{c,i}) \in \mathbb{R}^n\), where \(\tau_i^*, \tau_i, \hat{\tau}_i, \hat{\tau}_i\) denote the optimal control policies and the estimated ones at the time-triggered and event-triggered setting, respectively.

From Assumption 2 and 4, we can deduce that \(\|z_i\| \leq z_m, \|\hat{z}_i\| \leq z_m, \|\tau_i\| \leq \tau_m\) and \(\|\hat{\tau}_i\| \leq \tau_m\), with positive constants \(z_m, \tau_m\).

Then, with neural network approximation, (23) can be converted into

\[
\dot{L}_i = V_{e_{2,i}} \left\{ \sum_{j \in N_i} a_{ij} \|M_j^{-1}\| (\hat{\tau}_j + E_j(q_j, \hat{q}_j)) - M_j^{-1}(\hat{\tau}_j + E_j(q_j, \hat{q}_j)) \right\} \\
+ \alpha_{1,i} \sum_{j \in N_i} a_{ij} (\hat{q}_j - \hat{q}_j) \leq P^2\|s_i^T\|^2 \|\delta_i\|^2 - \gamma_i^2 \lambda_{\text{min}}(Q_{ii})\|e_{2,i}\|^2 - (1 - \gamma_i^2)\lambda_{\text{min}}(Q_{ii})\|e_{2,i}\|^2 \\
- \sum_{j \in N_i} \lambda_{\text{min}}(S_{ij})\|\tau_j\|^2 - \|s_i^T\hat{\tau}_i\|^2 + 6\lambda_{\text{max}}(S_{ii})\tau_m z_m (\nabla \phi_m \|\tilde{W}_{c,i}\| + \nabla \epsilon_m) \\
+ \nabla \epsilon_m + 3\lambda_{\text{max}}(S_{ii}) z_m^2 (\nabla \phi_m \|\tilde{W}_{c,i}\| + \nabla \epsilon_m) \\
+ 2 \sum_{j \in N_i} \lambda_{\text{max}}(S_{ij})\tau_m z_m (\nabla \phi_m \|\tilde{W}_{c,j}\| + \nabla \epsilon_m) \\
- \sum_{j \in N_i} \lambda_{\text{min}}(S_{ij})\|z_j (\nabla \phi_{c,j}^T \tilde{W}_{c,j} - \nabla \epsilon_{c,j})\|^2 \\
+ \|\tau_j^* T\|C_i \sum_{j \in N_i} a_{ij}\|M_j^{-1}\|\tau_j^* \| + \|\tau_j^* T\|C_i \sum_{j \in N_i} a_{ij}\|M_j^{-1}\|\hat{\tau}_j^* \| \\
+ (W_m \nabla \phi_m + \nabla \epsilon_m) \left( d_i k_{M\ell} (\nabla \phi_m \|\tilde{W}_{c,i}\| + \nabla \epsilon_m) \right) \\
+ \sum_{j \in N_i} a_{ik} k_{M\ell} (\nabla \phi_m \|\tilde{W}_{c,j}\| + \nabla \epsilon_m) \right).
\]

(24)

Based on Assumption 2 and 4, it can be known that \(\|\tau_j^* T\|C_i \sum_{j \in N_i} a_{ij}\|M_j^{-1}\|\tau_j^* \| + \|\tau_j^* T\|C_i \sum_{j \in N_i} a_{ij}\|M_j^{-1}\|\hat{\tau}_j^* \|\) has an upper bound \(B_m\).

Hence, (24) becomes

\[
\dot{L}_i \leq P^2\|s_i^T\|^2 \|\delta_i\|^2 - \gamma_i^2 \lambda_{\text{min}}(Q_{ii})\|e_{2,i}\|^2 - (1 - \gamma_i^2)\lambda_{\text{min}}(Q_{ii})\|e_{2,i}\|^2 \\
- \sum_{j \in N_i} \lambda_{\text{min}}(S_{ij})\|\tau_j\|^2 - \|s_i^T\hat{\tau}_i\|^2 + D_m + B_m \\
- \sum_{j \in N_i} \lambda_{\text{min}}(S_{ij})\|z_j (\nabla \phi_{c,j}^T \tilde{W}_{c,j} - \nabla \epsilon_{c,j})\|^2.
\]
where
\[
D_m = 6\lambda_{\max}(S_{ii})\tau_m z_m (\nabla \phi_m \| \mathbf{W}_{c,i} \| + \nabla \epsilon_m) + 3\lambda_{\max}(S_{ii}) z_m^2 (\nabla \phi_m \| \mathbf{W}_{c,i} \| + \nabla \epsilon_m) \\
+ 2 \sum_{j \in \mathcal{N}_i} \lambda_{\max}(S_{ij})\tau_m z_m (\nabla \phi_m \| \mathbf{W}_{c,j} \| + \nabla \epsilon_m) + (W_m \nabla \phi_m + \nabla \epsilon_m)
\]
\[
(\lambda_{\max}(S_{ii}) + \lambda_{\max}(S_{ij}))\tau_m z_m (\nabla \phi_m \| \mathbf{W}_{c,i} \| + \nabla \epsilon_m) + (W_m \nabla \phi_m + \nabla \epsilon_m) + D_m = 6\lambda_{\max}(S_{ii})\tau_m z_m (\nabla \phi_m \| \mathbf{W}_{c,i} \| + \nabla \epsilon_m) + 3\lambda_{\max}(S_{ii}) z_m^2 (\nabla \phi_m \| \mathbf{W}_{c,i} \| + \nabla \epsilon_m) \\
+ 2 \sum_{j \in \mathcal{N}_i} \lambda_{\max}(S_{ij})\tau_m z_m (\nabla \phi_m \| \mathbf{W}_{c,j} \| + \nabla \epsilon_m) + (W_m \nabla \phi_m + \nabla \epsilon_m)
\]

\[
\mathbf{L} = \mathbf{tr}\left[\mathbf{W}_{\tilde{c},i}\right)
\]

\[
\mathbf{L}^{T}c,i\mathbf{W}_{\tilde{c},i} + \nabla \epsilon_m) + \sum_{j \in \mathcal{N}_i} a_{ij} k_{Mj} (\nabla \phi_m \| \mathbf{W}_{c,j} \| + \nabla \epsilon_m)
\]

When the event-triggering condition (18) and the following condition satisfy, \( \dot{L}_i < 0 \),
\[
\|e_{2,i}\| > \sqrt{\frac{D_m + B_m}{\lambda_{\min}(Q_{ii})}}
\]

It implies that the local synchronization error \( e_{2,i} \) is UUB. During the event-triggering interval, the critic network’s weight estimation error keeps constant, which indicates that the weight estimation error \( \mathbf{W}_{c,i} \) is UUB.

Situation 2: Event-triggering instant, \( t = t_{i,k}, \forall i \in \mathcal{V} \).

At this situation, the Lyapunov function is chosen as (19)
\[
\Delta \mathbf{L}^i = \Delta \mathbf{L}_1^i + \Delta \mathbf{L}_2^i,
\]

where
\[
\Delta \mathbf{L}_1^i = V^i_1(e_{2,i}^+ - e_{2,i}) = 0 \quad \text{as } e_{2,i}^+ = e_{2,i} \quad \text{for } t = t_{i,k}.
\]

Hence, we only have to consider \( \Delta \mathbf{L}_2^i \), which is represented as follows,
\[
\Delta \mathbf{L}_2^i = \frac{1}{\mathbf{L}_c} [\mathbf{tr}(\mathbf{W}_{c,i}^T \mathbf{W}_{c,i}^+ - \mathbf{tr}(\mathbf{W}_{c,i}^T \mathbf{W}_{c,i}))].
\]

Using (17) in (25), it becomes
\[
\Delta \mathbf{L}_2^i = \frac{1}{\mathbf{L}_c} [\mathbf{tr}(\mathbf{W}_{c,i}^T \mathbf{W}_{c,i}^+ - \mathbf{tr}(\mathbf{W}_{c,i}^T \mathbf{W}_{c,i}))].
\]
It can be deduced that
\[
\Delta L_2 \leq -2\alpha \|\tilde{W}_{c,i}\|^2 + \epsilon_{cm}^2 + \|\tilde{W}_{c,i}\|^2 K_{Mi}^2 + l_c \beta^2 \|\tilde{W}_{c,i}\|^2 \\
+ \epsilon_{cm}^2 + K_{Mi}^2 \|\tilde{W}_{c,i}\|^2 K_{mi}^2 + l_c K_{Mi}^2 \epsilon_{cm}^2.
\]

Merging all similar terms in the above inequality, the first-order difference of the candidate Lyapunov function goes
\[
\Delta L_2 \leq P \|\tilde{W}_{c,i}\|^2 + Q,
\]
where
\[
P = -2\alpha + K_{Mi}^2 + l_c \beta^2 + l_c K_{Mi}^2 K_{mi}^2, \quad Q = \epsilon_{cm}^2 + l_c \epsilon_{cm}^2 + l_c K_{Mi}^2 \epsilon_{cm}^2.
\]

Hence, \(\Delta L_2 < 0\) can be guaranteed as long as the following condition satisfies,
\[
\|\tilde{W}_{c,i}\| > \sqrt{-\frac{Q}{P}},
\]
on condition that \(P < 0\), i.e.,
\[
l_c < \frac{2\alpha - K_{Mi}^2}{\beta^2 + K_{Mi}^2 K_{mi}^2}.
\]

Then, the overall Lyapunov functional candidate \(\Delta L^j < 0\) can be obtained. At this situation, this implies that the weight estimation error \(\tilde{W}_{c,i}\) is UUB.

From situation 1 and 2, we conclude that both the local synchronization error \(e_{2,i}\) and the weight estimation error \(\tilde{W}_{c,i}\) are UUB. \(\square\)

**Remark 2.** Compared with [39], all neighbouring agents can update their own control policy during \(i\)th agent’s triggering interval in this paper, which is more reasonable in the event-triggered mechanism and can apply to most applications, see the derivation of Theorem 4.2.

**Remark 3.** It should be noticed that the ultimate bound for the local synchronization error \(e_{2,i}\) and the estimation error \(\tilde{W}_{c,i}\) are related to the learning rate, the reconstruction error and other corresponding bounds. By adjusting the parametric setting and increasing the number of hidden neurons [12], the ultimate bounds can be accordingly driven to an arbitrarily small value.

**Assumption 6.** In order to avoid the Zeno behaviour, we assume that
\[
\|M_i^{-1}(q_i)E_i(q_i, \dot{q}_i) + \alpha_{1,i} \dot{q}_i\| \leq b_c \|\dot{q}_i + \alpha_{1,i} \dot{q}_i\|,
\]
where \(b_c > 0\).

**Remark 4.** In Assumption 6, it can be easily known that all terms in the left side corresponds to \(\dot{q}_i\) and \(q_i\) such that it is reasonable to find a proper \(b_c\) which satisfies the above condition. To prove its reasonability, we will implement a validation in the simulation part.

**Theorem 4.3.** With Assumption 6, the triggering condition (18) and the conclusion of Theorem 4.2, then the interval between the adjacent two triggering moments \(T_i = t_{i,k+1} - t_{i,k}, \forall i \in \mathcal{V}\) has a positive lower bound.

**Proof.** The dynamics of the event-triggered error \(\delta_i\) are as follows,
\[
\dot{\delta}_i = \dot{e}_{2,i} - \dot{\epsilon}_{2,i}, \forall t \in (t_{i,k}, t_{i,k+1}], i \in \mathcal{V}.
\]
By combining (10), it becomes

\[
\|\hat{\delta}_i\| = \|e_{2,i}\| \\
= \left\| \sum_{j \in \mathcal{N}_i} a_{ij} [M_i^{-1}(\hat{\tau}_i + E_i(q_i, \hat{q}_i)) - M_j^{-1}(\hat{\tau}_j + E_j(q_j, \hat{q}_j))] \right\| \\
+ \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij} (\hat{q}_i - \hat{q}_j) \\
\leq \left\| \sum_{j \in \mathcal{N}_i} a_{ij} [M_i^{-1}E_i(q_i, \hat{q}_i) - M_j^{-1}E_j(q_j, \hat{q}_j)] + \alpha_{1,i} \sum_{j \in \mathcal{N}_i} a_{ij} (\hat{q}_i - \hat{q}_j) \right\| \\
+ \| \sum_{j \in \mathcal{N}_i} a_{ij} M_i^{-1}\hat{\tau}_i \| + \| \sum_{j \in \mathcal{N}_i} a_{ij} M_j^{-1}\hat{\tau}_j \| \\
\leq b_c \|e_{2,i}\| + d_i k_M \left\| \frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T \nabla \phi_{c,i}^T \hat{\xi}_{c,i} \right\| \\
+ \left\| \sum_{j \in \mathcal{N}_i} a_{ij} k_{M_j} \frac{1}{2} d_j S_{jj}^{-1}(M_j^{-1})^T \nabla \phi_{c,j}^T \hat{\xi}_{c,j} \right\| \\
= b_c \|\hat{\delta}_i\| + \Gamma_{i,j},
\]

where

\[
\Gamma_{i,j} = b_c \|e_{2,i}\| + d_i k_M \left\| \frac{1}{2} d_i S_{ii}^{-1}(M_i^{-1})^T \nabla \phi_{c,i}^T \hat{\xi}_{c,i} \right\| \\
+ \left\| \sum_{j \in \mathcal{N}_i} a_{ij} k_{M_j} \frac{1}{2} d_j S_{jj}^{-1}(M_j^{-1})^T \nabla \phi_{c,j}^T \hat{\xi}_{c,j} \right\|.
\]

In the light of Assumption 2, 4 and Theorem 4.2, it can be deduced that \(\Gamma_{i,j}\) is bounded.

According to the comparison lemma in [15], it can be obtained,

\[
\|\hat{\delta}_i\| \leq e^{b_c (t - t_{i,k}^+)} \delta_i(t_{i,k}^+) + \frac{1}{2} \int_{t_{i,k}}^{t} e^{b_c (t - \tau)} \Gamma_{i,j} d\tau.
\] (26)

At the triggering moment \(t = t_{i,k}\), the event-triggered error is reset to zero with \(\delta_i(t_{i,k}^+) = 0\), it can be concluded that

\[
\|\hat{\delta}_i\| \leq \frac{1}{2} \frac{\Gamma_{i,j}}{b_c} (e^{b_c (t - t_{i,k}^+)} - 1), \forall t \in (t_{i,k}, t_{i,k+1}], \forall i \in \mathcal{V}.
\] (27)

Based on the triggering condition (18), it can be written as,

\[
\|\hat{\delta}_i\|^2 \leq \frac{(1 - \gamma^2)\lambda_{\min}(Q_{ii}) \|e_{2,i}\|^2}{P^2 \|s_i\|^2} + \frac{\|s_i^T \hat{\xi}_i\|^2}{P^2 \|s_i\|^2} + \sum_{j \in \mathcal{N}_i} \lambda_{\min}(S_{ij}) \|\tau_j\|^2 \\
= \frac{\Pi_{1i} + \Pi_{2i} + \Pi_{3i}}{P^2 \|s_i\|^2},
\]
where
\[
\Pi_{1i} = (1 - \gamma_i^2) \lambda_{\min}(Q_{ii}) \|e_{2,i}\|^2, \\
\Pi_{2i} = \frac{1}{2} d_i s_i^T S_{ii}^{-1} (M_i^{-1})^T V e_{2,i}, \\
\Pi_{3i} = \sum_{j \in \mathcal{N}_i} \lambda_{\min}(S_{ij}) \frac{1}{2} d_j s_j^T S_{jj}^{-1} (M_j^{-1})^T V e_{2,j},
\]

At the triggering instant \(t_{i,k+1}\), then it becomes
\[
\|\delta_i(t_{i,k+1})\| \leq \sqrt{\frac{\Pi_{1i} + \Pi_{2i} + \Pi_{3i}}{P^2 \|s_i\|^2}}. \tag{28}
\]

By combining (27) and (28), at time \(t_{i,k+1}\), we can derive that
\[
\sqrt{\frac{\Pi_{1i} + \Pi_{2i} + \Pi_{3i}}{P^2 \|s_i\|^2}} \leq \frac{1}{2} \frac{\Gamma i,j}{b_c} (e^{b_c(t_{i,k+1} - t_{i,k})} - 1).
\]

Hence, the inter-event time interval is as follows,
\[
T_i = t_{i,k+1} - t_{i,k} \geq \frac{1}{b_c} \log \frac{\Pi_{1i} + \Pi_{2i} + \Pi_{3i}}{P^2 \|s_i\|^2} + \frac{\Gamma i,j}{\Gamma i,j} > 0.
\]

Therefore, the inter-event interval has a positive lower bound, we can then conclude that the Zeno behaviour can be excluded in this work.

5. Simulation. In order to validate the utility of our method in this section, we give a numerical example. Specifically, we are concerned with a multiple Euler-Lagrange system which includes four agents [13], each of which is modeled as a 2-DOF prototype robot. Figure 1 illustrates the topology of the communication network, which is a strongly connected directed graph.

![Communication graph of MELSs.](image)

The Laplacian matrix of the graph in this paper is set as
\[
L = \begin{bmatrix}
1.5 & -1.5 & 0 & 0 \\
-0.5 & 2.0 & 0 & -1.5 \\
-1.5 & 0 & 2.5 & -1.0 \\
0 & 0 & -2.5 & 2.5
\end{bmatrix}.
\]
Each manipulator is modeled as follows,

\[ M_i(q_i) \ddot{q}_i + C_i(q_i, \dot{q}_i) \dot{q}_i + G_i(q_i) = \tau_i, \quad i = 1, 2, 3, 4, \]

in which \( q_i = [q_{i1}, q_{i2}]^T \) represents the joint position vector.

Define the vector parameter \( \mathcal{W} = [\varpi_1, \varpi_2, \varpi_3]^T \) as

\[
\begin{bmatrix}
\varpi_1 \\
\varpi_2 \\
\varpi_3
\end{bmatrix} = 
\begin{bmatrix}
I_{ca} + m_al_{ca}^2 + m_bl_{a}^2 + I_{cb} + m_bl_{cb}^2 \\
m_bl_{cb}
I_{cb} + m_bl_{cb}^2
\end{bmatrix},
\]

where \( m_a, m_b \) denote the mass of link \( a \) and \( b \), respectively, \( l_{ca}, l_{cb} \) represent the distance from the center of the mass including both links, \( l_a, l_b \) show the length and \( I_{ca}, I_{cb} \) denote the moment of inertia to center of mass for both links. The values of the mentioned notations and their units are given in the Table 1.

With the definition, \( M_i(q_i) \) and \( C_i(q_i, \dot{q}_i) \) are respectively denoted as follows,

\[
M_i(q_i) = 
\begin{bmatrix}
\varpi_1 + 2\varpi_2 \cos(q_{i2}) & \varpi_3 + \varpi_2 \cos(q_{i2}) \\
\varpi_3 + \varpi_2 \cos(q_{i2}) & \varpi_3
\end{bmatrix},
\]

\[
C_i(q_i, \dot{q}_i) = 
\begin{bmatrix}
-\varpi_2 \sin(q_{i2}) \dot{q}_{i2} & -\varpi_2 \sin(q_{i2})(\dot{q}_{i1} + \dot{q}_{i2}) \\
3\varpi_2 \sin(q_{i2}) \dot{q}_{i1} & 0
\end{bmatrix}.
\]

\[ \text{Figure 2. Triggering instants for all agents.} \]
Table 1. Notations, values and units of the according physical parameters.

| Notations | Values | Units  |
|-----------|--------|--------|
| $m_a$     | 1.2    | kg     |
| $m_b$     | 1      | kg     |
| $l_{ca}$  | 0.75   | m      |
| $l_{cb}$  | 0.75   | m      |
| $l_a$     | 0.26   | m      |
| $l_b$     | 0.5    | m      |
| $I_{ca}$  | 0.125  | kg·m²  |
| $I_{cb}$  | 0.188  | kg·m²  |
| $g$       | 9.81   | m/s²   |

In this simulation, the matrices $Q_{ii}$, $S_{ii}$ and $S_{ij}$ in the performance index are appointed as $Q_{11} = Q_{22} = Q_{33} = Q_{44} = \text{diag}\{5, 5\}$, $R_{11} = R_{22} = R_{33} = R_{44} = \text{diag}\{4, 4\}$, $R_{12} = R_{21} = R_{24} = R_{42} = R_{34} = R_{43} = \text{diag}\{1, 1\}$ and $\alpha_{1,i}$ in the auxiliary synchronization error is chosen as 0.4. The whole simulation time is up to 40 seconds, and the sampling interval $\Delta t$ here is set to 0.01 s.

![Figure 3](image)

**Figure 3.** Position trajectories of the first and second component of each EL agent.

The parameters in condition (18) are chosen as below,

$$\gamma_i = 0.9, \ i = 1, 2, 3, 4,$$

$$P = 2.5,$$

and $\kappa$ is chosen as $10^{-6}$ in this work.

The critic neural network is modeled as a three-layer neural network structure with three hidden units and one output unit. The input for the critic neural network
of agent $i$ is $e_{2,i}$ and the learning rate here is set as 0.4. We select the activation
functions as follows, where $e_{2,i} = [e_{2,1,i}, e_{2,2,i}]^T$, $i = 1, 2, 3, 4$,

\[
\phi_{c,1} = [\tanh((e_{2,1}^1)^2), \tanh(e_{2,1}^1 e_{2,1}^2), \tanh((e_{2,1}^2)^2)],
\]
\[
\phi_{c,2} = [\tanh((e_{2,2}^1)^2), \tanh(e_{2,2}^1 e_{2,2}^2), \tanh((e_{2,2}^2)^2)],
\]
\[
\phi_{c,3} = [\tanh((e_{2,3}^1)^2), \tanh(e_{2,3}^1 e_{2,3}^2), \tanh((e_{2,3}^2)^2)],
\]
\[
\phi_{c,4} = [\tanh((e_{2,4}^1)^2), \tanh(e_{2,4}^1 e_{2,4}^2), \tanh((e_{2,4}^2)^2)],
\]
where \( \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \). The choice of the initial weights of four critic networks are randomized within \([0, 2]\). It should be noticed that we have to guarantee the inertia matrices \( M_i(q_i), \forall i \) are reversible to obtain a reasonable control input as (12). The initial states of each agent are given as \( q_i(0) = [0.1i, 0.1i]^T \), and both the first-order and second-order derivative of \( q_i \) are set to zero vectors.

Figure 2 denotes the triggering instants of four agents. The number of un-triggered times of four agents are 3921, 3899, 3866 and 3858 respectively, which can be calculated that 97.15\% triggering times are saved. The trajectories of the time-varying state for all agents are indicated in Figure 3, 4. Figure 3 shows the position vectors and Figure 4 shows the velocity vectors. It can be seen that both \( q_i \) and \( \dot{q}_i \) reach an agreement, which is also obvious in the result of synchronization error, see Figure 5. It should be noticed that Figures 3, 4, 5 are shown under time-triggered situation. In fact, the sampled states are kept constant through ZOH during the triggering interval under event-triggered mechanism. Figure 6 indicates the evolution of the event-triggered control inputs of all agents under event-triggered setting, which converges to zero eventually. The norm of the estimated weights is also shown in Figure 7, which infers that the weight estimation error is UUB. In summary, our proposed method can synchronize MELSs while minimizing the performance index. Meanwhile, the computation and transmission resources are greatly reduced due to the event-triggered mechanism. Finally, take agent 1 for example, it can be seen that Assumption 6 is valid in Figure 8, in which \( b_c \) equals to 0.229 and the remaining agents can be validated similarly. Specifically, we find the value of \( b_c \) through several trials to validate its existence.

6. Conclusion and future work. In this work, we have studied the optimal synchronization problem of multiple Euler-Lagrange systems with reinforcement learning under event-triggered mechanism. Compared to the previous work, all agents can renew their control policies at one agent’s triggering interval in this work. With the event-triggered setting, the transmission cost and computational
resources are greatly reduced as all agents update their control policies aperiodically. Critic-only neural network has been used to approximate the value function and the control input can be then obtained. The absence of the Zeno behaviour has also been proved. In future works, we may concern with model-free optimal synchronization control of MELSs with event-triggered setting and the graph condition of directed spanning tree.

Acknowledgments. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0809302, in part by the National Natural Science Foundation of China under Grant 61751305, Grant 61673176, Grant 61873294, and Grant 61603133, in part by the Programme of Introducing Talents of Discipline to Universities (the 111 Project) under Grant B17017.

REFERENCES
[1] A. Abdessameud, A. Tayebi and I. G. Polushin, Leader-follower synchronization of Euler-Lagrange systems with time-varying leader trajectory and constrained discrete-time communication, IEEE Trans. Autom. Control, 62 (2017), 2539–2545.
[2] C. Amato, G. Konidaris, A. Anders, G. Cruz, J. P. How and L. P. Kaelbling, Policy search for multi-robot coordination under uncertainty, Int. J. Robot. Res., 35 (2016), 1760–1778.
[3] M. Bardi and I. Capuzzo-Dolcetta, Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations, Systems & Control: Foundations & Applications. Birkhäuser Boston, Inc., Boston, MA, 1997.
[4] D. P. Bertsekas, J. N. Tsitsiklis and A. Volgenant, Neuro-Dynamic Programming, Second edition. Athena Scientific Optimization and Computation Series. Athena Scientific, Belmont, MA, 1999.
[5] D. P. Bertsekas, Dynamic Programming and Optimal Control, Athena scientific, Belmont, MA, 1995.
[6] G. Chen, Y. Yue and Y. Song, Finite-time cooperative-tracking control for networked Euler-Lagrange systems, IET Control Theory Appl., 7 (2013), 1487–1497.
[7] S. J. Chung and J. J. E. Slotine, Cooperative robot control and concurrent synchronization of Lagrangian systems, IEEE Trans. Robot., 25 (2009), 686–700.
[8] D. V. Dimarogonas, E. Frazzoli and K. H. Johansson, Distributed event-triggered control for multi-agent systems, IEEE Trans. Autom. Control, 57 (2012), 1291–1297.
[9] F. Heppner and U. Grenander, A stochastic nonlinear model for coordinated bird flocks, Proc. Ubiquity Chaos, 233 (1990), 238.
[10] W. Hu, L. Liu and G. Feng, Consensus of multi-agent systems by distributed event-triggered control, Proc. IFAC, 47 (2014), 9768–9773.
[11] N. Huang, Z. Duan and Y. Zhao, Distributed consensus for multiple Euler-Lagrange systems: An event-triggered approach, Sci. China Technol. Sci., 59 (2016), 33–44.
[12] B. Igelnik and Y. H. Pao, Stochastic choice of basis functions in adaptive function approximation and the functional-link net, IEEE Trans. Neural Netw., 6 (1995), 1320–1329.
[13] X. Jin, D. Wei, W. He, L. Kocarev, Y. Tang and J. Kurths, Twisting-based finite-time consensus for Euler-Lagrange systems with an event-triggered strategy, IEEE Trans. Netw. Sci. Eng., (2019), 1–1.
[14] Y. Katz, K. Tunstrom, C. C. Ioannou, C. Huepe and I. D. Couzin, Inferring the structure and dynamics of interactions in schooling fish, Proc. Natl Acad. Sci., 108 (2011), 18720–18725.
[15] H. K. Khalil, Nonlinear Systems, Upper Saddle River, NJ: Prentice Hall, 2002.
[16] J. R. Klotz, Z. Kan, J. M. Shea, E. L. Pasiliao and W. E. Dixon, Asymptotic synchronization of a leader-follower network of uncertain Euler-Lagrange systems, IEEE Trans. Control Network Syst., 2 (2014), 174–182.
[17] J. R. Klotz, S. Obuz, Z. Kan and W. E. Dixon, Synchronization of uncertain Euler-Lagrange systems with uncertain time-varying communication delays, IEEE Trans. Cybern., 48 (2018), 807–817.
[18] F. L. Lewis, D. Vrabie and V. L. Syrmos, Optimal Control, John Wiley & Sons, New Jersey, 2012.
[19] X. Li, X. Yang and T. Huang, Persistence of delayed cooperative models: Impulsive control method, *Appl. Math. Comput.*, 342 (2019), 130–146.

[20] J. Li, H. Modares, T. Chai, F. L. Lewis and L. Xie, Off-policy reinforcement learning for synchronization in multiagent graphical games, *IEEE Trans. Neural Netw. Learn. Syst.*, 28 (2017), 2434–2445.

[21] A. Loria and H. Nijmeijer, Bounded output feedback tracking control of fully actuated Euler-Lagrange systems, *Syst. Control Lett.*, 33 (1998), 151–161.

[22] J. Mei, W. Ren and G. Ma, Distributed containment control for Lagrangian networks with parametric uncertainties under a directed graph, *Automatica*, 48 (2012), 653–659.

[23] J. J. Murray, C. J. Cox, G. G. Lendaris and R. Saeks, Adaptive dynamic programming, *IEEE Trans. Syst. Man Cybern.*, 32 (2002), 140–153.

[24] E. Nuno, R. Ortega, L. Basanez and D. Hill, Synchronization of networks of nonidentical Euler-Lagrange systems with uncertain parameters and communication delays, *IEEE Trans. Autom. Control*, 56 (2011), 935–941.

[25] J. Qin, M. Li, Y. Shi, Q. Ma and W. X. Zheng, Optimal synchronization control of multiagent systems with input saturation via off-policy reinforcement learning, *IEEE Trans. Neural Netw. Learn. Syst.*, 30 (2018), 85–96.

[26] Z. Qiu, Y. Hong and L. Xie, Optimal consensus of Euler-Lagrangian systems with kinematic constraints, *Proc. IFAC*, 49 (2016), 327–332.

[27] J. Sarangapani, *Neural Network Control of Nonlinear Discrete-Time Systems*, CRC press, Boca Raton, 2006.

[28] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, Second edition. Adaptive Computation and Machine Learning, MIT Press, Cambridge, MA.

[29] Y. Tang, X. Wu, P. Shi and F. Qian, Input-to-state stability for nonlinear systems with stochastic impulses, *Automatica*, 113 (2020), 108766, 12pp.

[30] K. G. Vamvoudakis, F. L. Lewis and G. R. Hudas, Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality, *Automatica*, 48 (2012), 1598–1611.

[31] K. G. Vamvoudakis, Event-triggered optimal adaptive control algorithm for continuous-time nonlinear systems, *IEEE/CAA J. Auton. Sinica*, 1 (2014), 282–293.

[32] X. F. Wang, Z. Deng, S. Ma and X. Du, Event-triggered design for multi-agent optimal consensus of Euler-Lagrangian systems, *Kybernetika*, 53 (2017), 179–194.

[33] C. Wei, J. Luo, H. Dai and J. Yuan, Adaptive model-free constrained control of postcapture flexible spacecraft: A Euler–Lagrange approach, *J. Vib. Contr.*, 24 (2018), 4885–4903.

[34] S. Weng, D. Yue and J. Shi, Distributed cooperative control for multiple photovoltaic generators in distribution power system under event-triggered mechanism, *J. Franklin Inst.*, 353 (2016), 3407–3427.

[35] D. Yang, X. Li and J. Qiu, Output tracking control of delayed switched systems via state-dependent switching and dynamic output feedback, *Nonlinear Anal. Hybrid Syst.*, 32 (2019), 294–305.

[36] H. Zhang, F. L. Lewis and A. Das, Optimal design for synchronization of cooperative systems: State feedback, observer and output feedback, *IEEE Trans. Autom. Control*, 56 (2011), 1948–1952.

[37] W. Zhang, Y. Tang, T. Huang and A. V. Vasilakos, Consensus of networked Euler-Lagrange systems under time-varying sampled-data control, *IEEE Trans. Ind. Inform.*, 14 (2018), 535–544.

[38] W. Zhang, Q. Han, Y. Tang and Y. Liu, Sampled-data control for a class of linear time-varying systems, *Automatica*, 103 (2019), 126–134.

[39] W. Zhao and H. Zhang, Distributed optimal coordination control for nonlinear multi-agent systems using event-triggered adaptive dynamic programming method, *ISA Trans.*, 91 (2019), 184–195.

Received January 2020; revised January 2020.

E-mail address: 852067159@qq.com  
E-mail address: jx_98109163.com  
E-mail address: 842376017@qq.com  
E-mail address: tangtany@gmail.com