# Adaptive Event-Triggered Output Synchronization of Heterogeneous Multiagent Systems A Model-Free Reinforcement Learning Approach

Hu, Wenfeng; Wang, Xuan; Guo, Meichen; Luo, Biao; Huang, Tingwen

**DOI**
10.1109/TSIPN.2025.3578759

**Publication date**
2025

**Document Version**
Final published version

**Published in**
IEEE Transactions on Signal and Information Processing over Networks

**Citation (APA)**
Hu, W., Wang, X., Guo, M., Luo, B., & Huang, T. (2025). Adaptive Event-Triggered Output Synchronization of Heterogeneous Multiagent Systems: A Model-Free Reinforcement Learning Approach. *IEEE Transactions on Signal and Information Processing over Networks*, *11*, 604-616. https://doi.org/10.1109/TSIPN.2025.3578759

# Adaptive Event-Triggered Output Synchronization of Heterogeneous Multiagent Systems: A Model-Free Reinforcement Learning Approach

Wenfeng Hu, *Member, IEEE*, Xuan Wang, Meichen Guo, *Member, IEEE*, Biao Luo, *Senior Member, IEEE*, and Tingwen Huang, *Fellow, IEEE*

***Abstract*—This paper proposes a reinforcement learning approach to the output synchronization problem for heterogeneous leader-follower multi-agent systems, where the system dynamics of all agents are completely unknown. First, to solve the challenge caused by unknown dynamics of the leader, we develop an experience-replay learning method to estimate the leader's dynamics, which only uses the leader's past state and output information as training data. Second, based on the newly estimated leader's dynamics, we design an event-triggered observer for each follower to estimate the leader's state and output. Furthermore, the experience-replay learning method and the event-triggered leader observer are co-designed, which ensures the convergence and Zeno behavior exclusion. Subsequently, to free the followers from reliance on system dynamics, a data-driven adaptive dynamic programming (ADP) method is presented to iteratively derive the optimal control gains, based on which we design a policy iteration (PI) algorithm for output synchronization. Finally, the proposed algorithm's performance is validated through a simulation.**

***Index Terms*—Event-triggered mechanism (ETM), experience-replay, heterogeneous multi-agent systems (HMASs), output synchronization, reinforcement learning (RL).**

## I. Introduction

DISTRIBUTED cooperative control of multi-agent systems (MASs) has made remarkable progress, with widespread applications in autonomous aerial vehicles [1], power systems [2], satellite networks [3] and so on. Current research on multi-agent cooperative control primarily focuses on homogeneous agents, with the objective of achieving state synchronization [4], [5], [6]. In contrast, this paper emphasizes the output synchronization problem for heterogeneous multi-agent systems (HMASs). The complexity of HMASs arises from variations in the dimensions and dynamic matrices of each agent. Most existing studies [7], [8] on output synchronization are based on traditional control theory. They assume known system dynamics and continuous communication conditions to optimize performance. However, these assumptions are impractical in real-world scenarios and lead to wasted communication resources.

To tackle the issue of unknown models, reinforcement learning (RL) and adaptive dynamic programming (ADP) are model-free intelligent learning algorithms that autonomously interact with the environment to optimize rewards [9], has gained widespread attention in the field of MASs in recent years [10], [11], [12], [13]. In [14], a new off-policy RL algorithm was first proposed for HMASs output synchronization in online scenarios, and an adaptive leader observer was designed under a continuous-time communication. Experience-replay, as a method in RL, has been applied to system dynamics estimation in [15]. During the learning process, both past and current states of the leader are required. Ref. [16] proposed a model-free leader observer based on experience replay. Nevertheless, the observer is designed only after the system dynamics are learned, leading to a decoupled design and potential estimation errors. In [17], a two-layer data-based neural network RL framework is proposed for cooperative learning in MASs without requiring prior knowledge of system models. In [18], a value iteration-based method is proposed to solve the cooperative $H_\infty$ output regulation problem for HMASs. However, most current research on multi-agent RL is based on fixed signal sampling, which can result in excessive data transmission between agents, wasting bandwidth and computational power.

Event-triggered mechanism (ETM) sampling is a non-periodic sampling approach, where sampling only occurs when specific triggering conditions are met [19]. Nowadadys, ETM sampling is extensively used in first-order [20], [21], second-order [22], and higher-order [23] MASs. The output synchronization problem based on event-triggered control in HMASs is studied in [24] and [25], but they require complete system dynamics information. Ref. [26] proposed an adaptive distributed observer with event-triggered communication to estimate the leader's system model and observe its state. In [27], [28], [29], [30], RL and ETM are combined to explore tracking control. A fault-tolerant adaptive event-triggered tracking control scheme

Received 20 December 2024; revised 24 April 2025 and 27 May 2025; accepted 4 June 2025. Date of publication 11 June 2025; date of current version 26 June 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62273358, Grant 62373375, and Grant U2368204, in part by the Natural Science Foundation of Hunan Province under Grant 2023JJ20075, and in part by the Science and Technology Innovation Program of Hunan Province under Grant 2024RC1011. The associate editor coordinating the review of this article and approving it for publication was Dr. Aykut Koc. *(Corresponding author: Biao Luo.)*

Wenfeng Hu, Xuan Wang, and Biao Luo are with the School of Automation, Central South University, Changsha 410083, China (e-mail: wenfenghu@csu.edu.cn; w994842461@163.com; biao.luo@hotmail.com).

Meichen Guo is with the Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: M.Guo@tudelft.nl).

Tingwen Huang is with the Faculty of Computer Science and Control Engineering, Shenzhen University of Advanced Technology, Shenzhen 518055, China (e-mail: huangtw2024@163.com).

Digital Object Identifier 10.1109/TSIPN.2025.3578759

was introduced in [27], utilizing a multi-gradient recursive RL algorithm. Ref. [28] uses ADP algorithm with event-triggered output feedback to solve the output regulation for discrete systems. In [29], a Deep Q-Network (DQN) is used to online learn an event-triggered controller. In [30], a dynamic event-triggered control scheme based on RL and barrier Lyapunov functions is proposed for perturbed nonlinear MASs under state constraints and disturbances. Under completely unknown system models, how to use RL to achieve output synchronization control with ETM communication is a key motivation in our research. Current studies on this topic remain limited.

This paper is dedicated to output synchronization in a class of leader-follower HMASs under a directed graph. We introduce an RL-based event-triggered optimal control strategy, which avoids prior knowledge of the system's dynamics. The contributions are summarized as follows.

- Different from [14], [26], which use adaptive estimation methods, a learning-based experience-replay method is developed to estimate the leader's system dynamics in this paper. It is noted that ETM was not considered in [14].
- Different from existing observers, which require prior knowledge of the leader's dynamics [23], [31] or rely on continuous-time communication [32], a model-free observer is proposed based on ETM communication. The leader's dynamics estimation and event-triggered observation are co-designed simultaneously, ensuring that the learning method does not affect the system stability, and no Zeno behavior happens.
- A data-driven model-free RL algorithm is proposed for real-time optimization of output synchronization control protocols. By employing the ADP, the algorithm approximates the optimal solution without requiring the solution of the algebraic Riccati equation (ARE) and knowledge of the dynamics of the leader or followers.

The rest structure of this paper is as follows. Section II formulates the problem of output synchronization in leader-follower HMASs. Section III develops a learning-based event-triggered observer. Sections III-A and III-B present the design of the observer and its stability analysis, respectively. Section IV introduces an RL-based policy iteration (PI) algorithm using ADP to obtain the optimal control policy for output synchronization. Section V illustrates the algorithm's validity via simulations. Finally, Section VI concludes the paper.

## II. Problem Formulation and Preliminaries

### A. Graph Theory

A directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ consists of the set of nodes $\mathcal{V} = [v_1, v_2, \ldots, v_N]$, the set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and the adjacency matrix $\mathcal{A} = (a_{ij}) \in \mathbb{R}^{N\times N}$ with $a_{ij} \in \{0, 1\}$. If node $i$ receives the information from node $j$, namely $(v_j, v_i) \in \mathcal{E}, a_{ij} = 1$; otherwise $a_{ij} = 0$. The neighbor set is $N_i = \{j|(v_j, v_i) \in \mathcal{E}\}$. Define $D = \text{diag}\{\sum_{j=1}^N a_{1j}, \ldots, \sum_{j=1}^N a_{Nj}\}$ as the in-degree matrix and $L = D - A$ as the Laplace matrix. Suppose a diagraph has a root node with a directed path to others, it has a spanning tree. Node 0 is the leader, if node 0 sends information to node $i$ directly, $a_{i0} = 1$; otherwise $a_{i0} = 0$. In this case, we let the pinning matrix $O = \text{diag}\{a_{10}, \ldots, a_{N0}\}$. $\mathbb{Z}_+$ is the set of non-negative integers. $\otimes$ denotes the Kronecker product. For any symmetric matrix $P = [p_{ij}]_{n\times n} \in \mathbb{R}^{n\times n}$, $\text{vecs}(P) = [p_{11}, 2p_{12}, \ldots, 2p_{1n}, p_{22}, 2p_{23}, \ldots, 2p_{(n-1)n}, p_{nn}]^T \in \mathbb{R}^{\frac{n(n+1)}{2}}$. For any vector $b = [b_1, \ldots, b_n]^T \in \mathbb{R}^n$, and $\text{vecv}(b) = [b_1^2, b_1 b_2, \ldots, b_1 b_n, b_2^2, b_2 b_3, \ldots, b_{n-1} b_n, b_n^2]^T \in \mathbb{R}^{\frac{n(n+1)}{2}}$. For any matrix $A = [a_{ij}]_{n\times n} \in \mathbb{R}^{n\times n}$, $\text{vec}(A) = [a_{11}, \ldots, a_{n1}, a_{12}, \ldots, a_{n2}, \ldots, a_{1n}, \ldots, a_{nn}]^T \in \mathbb{R}^{n^2}$. If matrix $A$ is time-varying, $\dot{A}(t) = [\dot{a}_{ij}(t)]_{n\times n} \in \mathbb{R}^{n\times n}$.

The following lemmas will be used.

*Lemma 1 ([33]):* Under Assumption 2, every eigenvalue of $H = L + O$ has positive real part, with $L$ being the Laplace matrix and $O$ being the pinning matrix, and there exists a positive matrix $G = \text{diag}(g_1, g_2, \ldots, g_N)$ such that $GH + H^T G > 0$, where $g = \text{col}(g_1, g_2, \ldots, g_N) = H^{-1} 1_N$.

*Lemma 2 ([34]):* Consider the following system

$$\dot{x} = Fx + F_1(t)x + F_2(t) \tag{1}$$

where $x \in \mathbb{R}^n$, $F \in \mathbb{R}^{n\times n}$ is Hurwitz, $F_1(t) \in \mathbb{R}^{n\times n}$ and $F_2(t) \in \mathbb{R}^n$ are bounded and continuous for all $t \geq t_0$. If $\lim_{t\to\infty} F_1(t) = 0$ and $\lim_{t\to\infty} F_2(t) = 0$ exponentially, then for any $x(t_0)$, $\lim_{t\to\infty} x(t) = 0$ exponentially.

### B. Problem Formulation

Consider a class of continuous-time linear HMASs with one leader and $N$ followers, the leader's dynamics are modeled as

$$\dot{x}_0 = A_0 x_0 \tag{2}$$

$$y_0 = C_0 x_0 \tag{3}$$

where $x_0 \in \mathbb{R}^{n_0}$, $y_0 \in \mathbb{R}^{m_0}$ is the state and the output of the leader respectively. $A_0 \in \mathbb{R}^{n_0\times n_0}$ and $C_0 \in \mathbb{R}^{m_0\times n_0}$ are the leader's system matrices.

The followers' dynamics are described by

$$\dot{x}_i = A_i x_i + B_i u_i \tag{4}$$

$$y_i = C_i x_i \tag{5}$$

where $x_i \in \mathbb{R}^{n_i}$ is the state, $u_i \in \mathbb{R}^{m_i}$ is the control input, $y_i \in \mathbb{R}^{p_i}$ is the output of the follower $i$. $A_i \in \mathbb{R}^{n_i\times n_i}$, $B_i \in \mathbb{R}^{n_i\times m_i}$, and $C_i \in \mathbb{R}^{p_i\times n_i}$ are the followers' system matrices.

The interaction among all followers and the leader can be represented by a directed graph $\mathcal{G}$, where the leader is denoted as node 0, and the followers are the remaining nodes. In particular, only if $a_{ij} = 1$ for $i \neq 0$ and $j \neq 0$, follower $i$ receives the information (state/output) from follower $j$. Besides, we use $a_{i0}$ to denote the communication link between followers and the leader, and thus only part of the followers with $a_{i0} = 1, i = 1, \ldots, N$, have direct access to the leader's state and output, while other followers can only indirectly get the information from the leader. It is noted that the prior knowledge of agents' dynamics, no matter for followers or the leader, is not required.

The objective of this paper is to develop a distributed control protocol $u_i$ with event-triggered communication, where all agent dynamics are completely unknown for the HMASs under a directed spanning tree, described by (2)–(5), to achieve:

1) A leader observer is designed based on event-triggered communication such that every follower can estimate the leader's state and output. Specifically, for all $i = 1, \ldots, N$, the observer satisfies $\lim_{t\to\infty}(\eta_i(t) - x_0(t)) = 0$ and $\lim_{t\to\infty} (\zeta_i(t) - y_0(t)) = 0$. Here, $\eta_i \in \mathbb{R}^{n_0}$ and $\zeta_i \in \mathbb{R}^{m_0}$ denote the follower $i$'s estimated state and output of the leader, respectively.

2) All the followers' outputs are to synchronize with the leader's output without requiring any information of the dynamics, satisfying $\lim_{t\to\infty}(y_i(t) - y_0(t)) = 0, i = 1, \ldots, N$.

The following assumptions will be used.

*Assumption 1:* Each pair $(A_i, B_i)$ is stabilizable and $(A_i, C_i)$ is observable.

*Assumption 2:* The graph $\mathcal{G}$ has a spanning tree with the leader as a root.

*Assumption 3:* Every eigenvalue of $A_0$ is on the imaginary axis and each eigenvalue appears only once without repetition.

*Remark 1:* For Assumption 1, the stability condition is necessary for the feedback control, even for a single system. Similarly, from a theoretical viewpoint, the observability condition is also necessary for the design of the effective observers. Assumption 1 is widely used in some literature on cooperative control of MASs, like [14], [34], [35]. Assumption 2 is a necessary condition for directed graphs, ensuring that all followers can receive information from the leader. Assumption 3 guarantees that the leader is marginally stable, so that the leader's output remains a persistent periodic signal, and it also allows the use of LaSalle's invariance principle to derive the convergence of the closed-loop system.

*Remark 2:* The typical method for solving the output synchronization problem relies on the solution of the output regulation equation, which needs to know the agents' system dynamics. However, in this study, a novel model-free RL-based approach is introduced. Firstly, a model-free event-triggered leader observer is developed. Based on this observer, an RL-based algorithm is proposed to tackle the output synchronization. The new method alleviates the requirement of the system dynamics for all agents.

## III. Learning-Based Event-Triggered Leader Observer

In this section, we first propose a learning-based event-triggered observer for each follower to estimate the leader's state and output.

### A. The Design of the Leader Observer

First, we define the local combined measurement error of follower $i$ as

$$q_i = \sum_{j=0}^{N} a_{ij}(\eta_j - \eta_i) \tag{6}$$

where $\eta_i \in \mathbb{R}^{n_0}$ is the state of the observer to be designed later. When $j = 0$, define $\eta_0 = x_0$.

The event-triggered leader observer is designed as follows

$$\begin{cases} \dot{\eta}_i = \hat{A}_0(t)\eta_i + \beta q_i(t_k^i), t \in [t_k^i, t_{k+1}^i) \\ \zeta_i = \hat{C}_0(t)\eta_i \end{cases} \tag{7}$$

where $\zeta_i \in \mathbb{R}^{m_0}$ is the output of follower $i$. $\hat{A}_0(t)$ and $\hat{C}_0(t)$ are the estimation of the leader's dynamics $A_0$ and $C_0$ that need to be designed later, respectively. $\beta > 0$ is the positive weight to be designed later. And, $t_0^i, t_1^i, \ldots,$ are the triggering times for agent $i$.

The design of this observer consists of three steps: to get $\hat{A}_0(t)$ via Step 1, to get $\hat{C}_0(t)$ via Step 2, and to determine the triggering time sequence via Step 3.

*Step 1: To get $\hat{A}_0(t)$*

Traditional observers rely on the model information including $A_0$. To overcome this limitation, inspired by [15], we will develop a data-driven experience-replay learning method to learn $A_0$.

First, we apply the following filters

$$\dot{x}_l(t) = -kx_l(t) + x_0(t) \tag{8}$$

$$\dot{z}(t) = -kz(t) + \dot{x}_0(t) \tag{9}$$

$$\dot{\Lambda}(t) = -k\Lambda(t) + Y(t) \tag{10}$$

where $x_l(t)$, $z(t)$ and $\Lambda(t)$ are the variables after filtering with $x_0(t), \dot{x}_0(t)$, and $Y(t)$ with $Y(t) = x_0^T \otimes I_{n_0}$, respectively. $k$ is a positive gain to stabilize the filters and $x_l(0) = z(0) = \Lambda(0) = 0$.

And then, we set $\{z(t_s)\}_{s=1}^{p_a}$ and $\{\Lambda(t_s)\}_{s=1}^{p_a}$ as memory stacks to store the past data at different time instants $t = t_s$ with $t_1 < t_2 < \ldots < t_{p_a}$, where $p_a \geq n_0$ is the length of the stacks.

The data-driven update law to estimate $A_0$ is designed as

$$\dot{\hat{\mathscr{A}}}(t) = \iota_a \sum_{s=1}^{p_a} \Lambda^T(t_s)\phi_a^s(t) \tag{11}$$

where $\hat{\mathscr{A}}(t) \in \mathbb{R}^{{n_0}^2}$, $\iota_a > 0$ is a positive weight, $\Lambda(t)$ is given by (10), and $\phi_a^s(t) = z(t_s) - \Lambda(t_s)\hat{\mathscr{A}}(t)$ with $z(t_s) = x_0(t_s) - e^{-kt_s}x_0(0) - kx_l(t_s)$. Besides, the matrix form $\hat{A}_0(t)$ in (7) can be obtained through $\hat{\mathscr{A}}(t)$ by noting that $\hat{\mathscr{A}}(t) = \mathrm{vec}(\hat{A}_0(t))$.

To proceed the analysis, we reconstruct the leader's system matrix in (2), the matrix can be vectorized as

$$\dot{x}_0(t) = Y(t)\mathscr{A} \tag{12}$$

where $Y(t) = x_0^T \otimes I_{n_0} \in \mathbb{R}^{n_0 \times {n_0}^2}$, $\mathscr{A} = \mathrm{vec}(A_0) \in \mathbb{R}^{{n_0}^2}$.

The estimation error is defined as

$$\phi_a(t) = \Lambda(t)(\mathscr{A} - \hat{\mathscr{A}}(t)) \tag{13}$$

With such design, we can have the following lemma based on some standard assumption.

*Assumption 4:* The memory stack $\{\Lambda(t_s)\}_{s=1}^{p_a}$ is of full row rank, i.e., $\mathrm{rank}([\Lambda^T(t_1), \Lambda^T(t_2), \ldots, \Lambda^T(t_{p_a})]) = n_0^2$ with $p_a \geq n_0$.

*Remark 3:* Assumption 4 ensures that matrix $\sum_{s=1}^{p_a} \Lambda^T(t_s) \Lambda(t_s)$ is strictly positive definite, i.e., $\lambda_{\min}^{\Lambda} > 0$, where $\lambda_{\min}^{\Lambda}$ denote the minimum eigenvalue of $\sum_{s=1}^{p_a} \Lambda^T(t_s)\Lambda(t_s)$. In this case, according to (69) as given in the appendix, we have $\dot{V}_a \leq -2\iota_a\lambda_{\min}^{\Lambda}V_a < 0$, $\forall t > t_{p_a}$ and $V_a \neq 0$, where $V_a = \frac{1}{2}(\mathscr{A} - \hat{\mathscr{A}}(t))^T(\mathscr{A} - \hat{\mathscr{A}}(t))$ (also given in (67) later). This guarantees the exponential convergence of the estimation error. If this assumption is not satisfied, we still have $\dot{V}_a \leq 0$ for all

$t > t_{p_a}$, which only guarantees convergence and may result in slower convergence. In practical, since $\Lambda(t) \in \mathbb{R}^{n_0 \times n_0^2}$, the rank condition is generally fulfilled when $p_a$ is much greater than $n_0$, indicating that enough amount of data is collected. Besides, under Assumption 3 the leader state $x_0(t)$ is a marginally stable and persistent periodic signal, the rank condition is easily satisfied when $p_a \geq n_0$. To verify the importance of Assumption 4, we have also provided the simulation under the case if the required rank condition is violated, please see Fig. 4 in the simulation part.

*Lemma 3:* Under Assumption 4, consider the data-driven update law (11) only using the leader's past state information, then the estimation error given by (13) exponentially converge to zero.

*Proof:* The proof can be found in the Appendix. ■

*Remark 4:* The memory stacks $\{z(t_s)\}_{s=1}^{p_a}$ and $\{\Lambda(t_s)\}_{s=1}^{p_a}$ serve as training data. Thus, experience-replay method allows the agent to store and replay past data, which is different from [15]. In addition, due to the lag existed in followers acquiring state information from the leader, this method reduces reliance on real-time interactions, thereby enhancing learning efficiency. Prior to utilizing this approach, it is essential to gather a sufficient amount of data to fulfill the rank condition. Moreover, Lemma 3 ensures that $\hat{A}_0(t)$ will converge to $A_0$ exponentially.

*Step 2: To get* $\hat{C}_0(t)$

After estimating $A_0$, we proceed to estimate $C_0$. The method for estimating $C_0$ is similar to that of $A_0$ and also requires the use of experience-replay method.

We set $\{y_0(t_s)\}_{s=1}^{p_c}$ and $\{M(t_s)\}_{s=1}^{p_c}$ are memory stacks to store the past data at different time instants $t = t_s$ with $t_1 < t_2 < \ldots < t_{p_c}$, where $p_c \geq n_0$ is the length of the stacks, and $M(t) = x_0^T \otimes I_{m_0}$.

And then, the data-driven update law to estimate $C_0$ is designed as

$$\dot{\hat{\mathscr{C}}}(t) = \iota_c \sum_{s=1}^{p_c} M^T(t_s)\phi_c^s(t) \tag{14}$$

where $\hat{\mathscr{C}}(t) \in \mathbb{R}^{m_0 n_0}$, $\iota_c > 0$ is a positive weight, and $\phi_c^s(t) = y_0(t_s) - M(t_s)\hat{\mathscr{C}}(t)$.

Following the learning process using the updated law (14), the matrix form $\hat{C}_0(t)$ in (7) can be obtained by transforming the vectorized form $\hat{\mathscr{C}}(t)$, that is $\hat{\mathscr{C}}(t) = \text{vec}(\hat{C}_0(t))$.

To proceed the analysis, in respect of (3), we reconstruct as

$$y_0 = M(t)\mathscr{C} \tag{15}$$

where $M(t) = x_0^T \otimes I_{m_0} \in \mathbb{R}^{m_0 \times m_0 n_0}$, $\mathscr{C} = \text{vec}(C_0) \in \mathbb{R}^{m_0 n_0}$.

The estimation error is defined as

$$\phi_c = y_0(t) - M(t)\hat{\mathscr{C}}(t) \tag{16}$$

*Assumption 5:* The memory stack $\{M(t_s)\}_{s=1}^{p_c}$ is of full row rank, i.e., $\text{rank}([M^T(t_1), M^T(t_2), \ldots, M^T(t_{p_c})]) = m_0 n_0$ where $p_c \geq m_0$.

*Lemma 4:* Under Assumption 5, consider the data-driven update law (14) only using the leader's past state information, then the estimation error given by (16) exponentially converge to zero.

*Proof:* For the update law in (14), the computation of $M(t_s)$ and $\phi_c^s(t)$ can be directly obtained from the leader's past state and output. The proof of the convergence to $C_0$ is similar to Lemma 3 and thus is omitted here. ■

*Step 3: To Determine the triggering time sequence*

To reduce communication between agents, we will design an ETM. The combined measurement error is defined as

$$e_i(t) = q_i(t_k^i) - q_i(t), t \in \left[t_k^i, t_{k+1}^i\right). \tag{17}$$

Then, we design the ETM as

$$\begin{cases} t_{k+1}^i = \inf\{t > t_k^i \| \|e_i(t)\|^2 - \kappa_i\|q_i(t)\|^2 - \mu_i\xi_i(t) \geq 0\} \\ \dot{\xi}_i(t) = -\upsilon_{1i}\xi_i(t) + \upsilon_{2i}(\kappa_i\|q_i(t)\|^2 - \|e_i(t)\|^2) \end{cases} \tag{18}$$

where $\xi_i(0) > 0$, $\upsilon_{1i} > 0$, $\upsilon_{2i} \geq \beta\|H\|\|P_0\|$, $\mu_i > 0$ and $0 < \kappa_i \leq \frac{1}{\upsilon_{2i}}$ with $\beta > 0$.

*Remark 5:* In the observer (7), $\hat{A}_0(t)$ and $\hat{C}_0(t)$ represent the estimations of $A_0$ and $C_0$ learned via experience replay, respectively. The term $q_i(t_k^i)$ indicates that each agent communicates only at event-triggered time instants. These three components are co-designed to simultaneously achieve system model learning and event-triggered observation of the leader. In contrast, Ref. [16] adopts a decoupled design, where the observer is constructed only after completing system model estimation. Directly relying on the estimated system model can result in approximation errors, potentially affecting the stability of the closed-loop system. Furthermore, it is theoretically proven that the learning process does not compromise system stability or cause Zeno behavior.

### B. *Stability Analysis of the Leader Observer*

The state observation error and output observation error are defined as

$$\tilde{\eta}_i = \eta_i - x_0 \tag{19}$$

$$\tilde{\zeta}_i = \zeta_i - y_0. \tag{20}$$

*Theorem 1:* Under Assumptions 1–5, the learning-based event-triggered leader observer (7), (11), and (14) are co-designed simultaneously. Then, the state and output observation error (19) and (20) exponentially decay to zero if the triggering times are determined by the ETM (18).

*Proof:* Define the observer state $\eta = \text{col}(\eta_1, \eta_2, \ldots, \eta_N)$ and $\bar{q} = \text{col}(\bar{q}_1, \bar{q}_2, \ldots, \bar{q}_N)$, where $\bar{q}_j = q_j(t_{k_j'}^j)$ with $k_j' = \arg\max_{k \in \mathbb{N}}\{t_k^j | t_k^j \leq t\}$. It follows from (17) that $q_i(t_k^i) = e_i(t) + q_i(t)$, then rewrite (7) as the following compact form:

$$\begin{aligned} \dot{\eta} &= (I_N \otimes \hat{A}_0)\eta + \beta\bar{q} \\ &= ((I_N \otimes \hat{A}_0) - \beta(H \otimes I_{n_0}))\eta \\ &\quad + \beta(H \otimes I_{n_0})(1_N \otimes x_0) + \beta e \end{aligned} \tag{21}$$

where $e = \mathrm{col}(e_1, e_2, \ldots, e_N)$, $H = L + O$ with $L$ being the Laplace matrix and $O$ being the pinning matrix, and every eigenvalue of $H$ has positive real part according to Lemma 1.

It follows from (7) and (19) that

$$\begin{aligned}\dot{\tilde{\eta}}_i &= \hat{A}_0(t)\eta_i - A_0 x_0 + \beta q_i(t_k^i) \\ &= \hat{A}_0(t)\eta_i - A_0\eta_i + A_0\eta_i - A_0 x_0 + \beta \tilde{q}_i(t_k^i) \\ &= A_0\tilde{\eta}_i + \tilde{A}_0(t)\tilde{\eta}_i + \tilde{A}_0(t)x_0 + \beta\tilde{q}_i(t_k^i) \qquad (22)\end{aligned}$$

where $\tilde{q}_i = \sum_{j=0}^{N} a_{ij}(\tilde{\eta}_i - \tilde{\eta}_j) = q_i$, $\tilde{A}_0(t) = \hat{A}_0(t) - A_0$. Similar to (21), rewrite (22) into the compact form

$$\begin{aligned}\dot{\tilde{\eta}} &= ((I_N \otimes A_0) - \beta(H \otimes I_{n_0}))\tilde{\eta} + (I_N \otimes \tilde{A}_0(t))\tilde{\eta} \\ &\quad + (I_N \otimes \tilde{A}_0(t))(1_N \otimes x_0) + \beta e. \qquad (23)\end{aligned}$$

According to Lemma 2, (23) is analogous to (1), where $(I_N \otimes A_0) - \beta(H \otimes I_{n_0}) \triangleq F$, $(I_N \otimes \tilde{A}_0(t)) \triangleq F_1(t)$ and $(I_N \otimes \tilde{A}_0(t))(1_N \otimes x_0) + \beta e \triangleq F_2(t)$. Then, it is sufficient to show that (23) satisfies the condition of Lemma 2, which implies $\lim_{t\to\infty} \tilde{\eta} = 0$.

Since $H$ has positive real part according to Lemma 1, $(I_N \otimes A_0) - \beta(H \otimes I_{n_0})$ is Hurwitz if $\beta > 0$ under Assumption 3.

It follows from Lemma 3, that parameter estimation error $\delta_a(t) = \mathscr{A} - \hat{\mathscr{A}}(t)$ exponentially converges to 0, which implies $\hat{A}_0$ exponentially converge to $A_0$ for $\forall t \geq t_{p_a}$, that is $\lim_{t\to\infty} \tilde{A}_0(t) = 0$ exponentially. And under Assumption 3, it is easy to obtain that $(I_N \otimes \tilde{A}_0)(1_N \otimes x_0)$ will decay to zero exponentially, too.

Next, we will prove that $\beta e$ will decay to zero exponentially. Construct the following Lyapunov function candidate

$$V = \sum_{i=1}^{N} q_i^T(t)P_0 q_i(t) + \sum_{i=1}^{N} \xi_i(t) \qquad (24)$$

where $P_0$ is a positive definite symmetric matrix, such that

$$\hat{A}_0^T P_0 + P_0\hat{A}_0 - \alpha P_0^2 + I_{n_0} \leq 0 \qquad (25)$$

and define $V_1 = \sum_{i=1}^{N} q_i^T(t)P_0 q_i(t)$.

Referring to (18), we have $\|e_i\|^2 - \kappa_i\|q_i\|^2 \leq \mu_i\xi_i$, which implies that

$$\dot{\xi}_i \geq -\upsilon_{1i}\xi_i - \upsilon_{2i}\mu_i\xi_i. \qquad (26)$$

Using the comparison principle provides

$$\xi_i \geq \xi_i(0)e^{-(\upsilon_{1i}+\upsilon_{2i}\mu_i)t} > 0. \qquad (27)$$

Moreover, it is easy to prove that $V_1 \geq 0$ which leads to $V > 0$.

The compact form $q(t) = -(H \otimes I_{n_0})\eta(t)$. Then, based on (7), the derivative of $q(t)$ is given as

$$\dot{q} = (I_N \otimes \hat{A}_0 - \beta(H \otimes I_{n_0}))q(t) - \beta(H \otimes I_{n_0})e \qquad (28)$$

The derivative of $V_1$ along (28) is provided as

$$\begin{aligned}\dot{V}_1 &= \sum_{i=1}^{N} q_i^T P_0 \dot{q}_i = 2q^T(I_N \otimes P_0)\dot{q} \\ &= 2q^T(I_N \otimes P_0)(I_N \otimes \hat{A}_0)q - 2\beta q^T(H \otimes P_0)q \\ &\quad - 2\beta q^T(H \otimes P_0)e \\ &\leq q^T(I_N \otimes (\hat{A}_0^T P_0 + P_0\hat{A}_0))q + \beta e^T(H \otimes P_0)e \\ &\quad - 2\beta q^T(H \otimes P_0)q + \beta q^T(H \otimes P_0)q \\ &\leq q^T(I_N \otimes (\hat{A}_0^T P_0 + P_0\hat{A}_0))q - \alpha q^T(I_N \otimes P_0^2)q \\ &\quad + \beta e^T(H \otimes P_0)e \qquad (29)\end{aligned}$$

where $\alpha = \beta\frac{\lambda_{\min}(H)}{\lambda_{\max}(P_0)} > 0$. Then, we have

$$\begin{aligned}\dot{V}_1 &\leq q^T(I_N \otimes (\hat{A}_0^T P_0 + P_0\hat{A}_0 - \alpha {P_0}^2))q \\ &\quad + \beta e^T(H \otimes P_0)e \\ &\leq -\sum_{i=1}^{N}\|q_i\|^2 + \beta\|H\|\|P_0\|\sum_{i=1}^{N}\|e_i\|^2. \qquad (30)\end{aligned}$$

According to (18) and (30), the derivative of $V$ satisfies

$$\begin{aligned}\dot{V} &= \dot{V}_1 + \sum_{i=1}^{N}\dot{\xi}_i \\ &\leq -\sum_{i=1}^{N}(1 - \upsilon_{2i}\kappa_i)\|q_i\|^2 \\ &\quad + \sum_{i=1}^{N}(\beta\|H\|\|P_0\| - \upsilon_{2i})\|e_i\|^2 - \sum_{i=1}^{N}\upsilon_{1i}\xi_i \\ &\leq -\omega\sum_{i=1}^{N} q_i^T P_0 q_i - \sum_{i=1}^{N}\upsilon_{1i}\xi_i \\ &\leq -l_0 V \qquad (31)\end{aligned}$$

where $\upsilon_{2i} \geq \beta\|H\|\|P_0\|$ and $\omega = (1 - \upsilon_{2i}\kappa_i)/\lambda_{\max}(P_0) \geq 0$ by selecting $0 < \kappa_i \leq \frac{1}{\upsilon_{2i}}$. And $l_0 = \min\{\omega, \upsilon_{1m}\} > 0$ with $\upsilon_{1m} = \min_i \upsilon_{1i}$. Thus, $V(t)$ exponentially decays to zero. Since $\lambda_{\min}(P_0)\|q(t)\|^2 \leq V_1(t) < V(t)$, one has $\lim_{t\to\infty} q(t) = 0$ exponentially, and thus, $\lim_{t\to\infty} e(t) = 0$ exponentially.

Therefore, by Lemma 2, $\lim_{t\to\infty} \tilde{\eta}_i(t) = 0$ exponentially. And according to Lemma 4, $\hat{C}_0(t)$ exponentially converges to $C_0$, we obtain $\lim_{t\to\infty} \tilde{\zeta}_i(t) = 0$ exponentially. The proof is completed. ■

*Remark 6:* It is worth mentioning that the estimation approach designed in [34] requires parts of the followers exactly know the leader's system dynamics. In contrast, our experience-replay learning method does not require any agents have knowledge of the leader's system dynamics by using the past data. Furthermore, the observer in [34] relies on continuous communication, whereas our proposed observer is based on event-triggered communication. Moreover, the three steps are co-designed simultaneously in the observer to ensure the convergence.

*Lemma 5:* The learning-based event-triggered leader observer (7) under the ETM (18) ensures all agents are capable of avoiding Zeno behavior.

*Proof:* We will employ a proof by contradiction to establish that if agent $i$ has Zeno behavior, i.e., $\lim_{k\to\infty} t_k^i = T_0 < \infty$. Then we have for any $\varepsilon > 0$, there exists $k_0$ such that $t_k^i \in (T_0 - \varepsilon, T_0 + \varepsilon)$ for $\forall k \geq k_0$, implying $t_{k_0+1}^i - t_{k_0}^i < 2\varepsilon$.

Because $V_1 = \sum_{i=1}^{N} q_i^T(t)P_0 q_i(t)$, it is easy to acquire $\sum_{i=1}^{N} \|q_i(t)\|^2 = \|q(t)\|^2 \leq (V_1(t)/\lambda_{\min}(P_0))$. From (31), it follows $V(t) \leq V(0)e^{-l_0 t}$, thus $V_1(t) \leq V(t) \leq V(0)$. Then, one has

$$\|q_i(t)\| \leq \|q(t)\| \leq \sqrt{\frac{V(0)}{\lambda_{\min}(P_0)}} \doteq V_0 \tag{32}$$

In the interval $[t_k^i, t_{k+1}^i)$, $\|e_i(t)\|$ is piecewise continuously differentiable. Using (6)–(7), its Dini derivative is given by

$$\begin{aligned} D^+\|e_i(t)\| &\leq \frac{\|e_i^T\|}{\|e_i\|}\|\dot{e}_i\| = \|-\dot{q}_i(t)\| \\ &= \left\|\sum_{j=0}^{N} a_{ij}(\dot{\eta}_i(t) - \dot{\eta}_j(t))\right\| \\ &= \left\|-\hat{A}_0(t)q_i(t) + \beta\sum_{j=0}^{N} a_{ij}(q_i(t_k^i) - q_j(t_{k'}^j))\right\| \end{aligned} \tag{33}$$

Based on (32), it follows

$$\begin{aligned} D^+\|e_i(t)\| &\leq \|\hat{A}_0(t)\|\|q_i(t)\| \\ &\quad + \beta\left(\sum_{j=0}^{N} a_{ij}(\|q_i(t_k^i)\| + \|q_j(t_{k'}^j)\|)\right) \\ &\leq V_0(\|\hat{A}_0(t)\| + \beta(1+|N_i|)) \\ &\doteq \hat{V}_0 \end{aligned} \tag{34}$$

where $k' = \arg\max_{k\in\mathbb{N}}\{t_k^j | t_k^j \leq t, j \in N_i\}$ denotes the most recent triggering time of follower $j$ prior to the current time $t$.

According to the ETM (18), when $t = t_k^{i-}$, one has $\|e_i(t)\| \geq \sqrt{\kappa_i\|q_i(t)\|^2 + \mu_i\xi_i(t)} \geq \sqrt{\mu_i\xi_i(t)}$. Define $f(t^-) = \lim_{s\to t^-} f(s)$. When $t = t_k^{i-}$, one has

$$\begin{aligned} \|e_i(t_k^{i-})\| &\geq \sqrt{\kappa_i\|q_i\left(t_k^{i-}\right)\|^2 + \mu_i\xi_i(t_k^{i-})} \\ &\geq \sqrt{\mu_i\xi_i(t_k^{i-})} \\ &= \sqrt{\mu_i\xi_i(0)}e^{-\frac{\upsilon_{1i}+\upsilon_{2i}\mu_i}{2}t_k^{i-}} \end{aligned} \tag{35}$$

By (34) and (35), then

$$\begin{aligned} t_{k_0+1}^i - t_{k_0}^i &\geq \frac{1}{\hat{V}_0}\sqrt{\mu_i\xi_i(0)}e^{-\frac{\upsilon_{1i}+\upsilon_{2i}\mu_i}{2}t_{k_0+1}^{i-}} \\ &\geq \frac{1}{\hat{V}_0}\sqrt{\mu_i\xi_i(0)}e^{-\frac{\upsilon_{1i}+\upsilon_{2i}\mu_i}{2}(T_0+\varepsilon)} \\ &= 2\varepsilon \end{aligned} \tag{36}$$

where $\varepsilon > 0$ satisfies the equation $\frac{1}{\hat{V}_0}\sqrt{\mu_i\xi_i(0)}e^{-\frac{\upsilon_{1i}+\upsilon_{2i}\mu_i}{2}T_0} = 2\varepsilon e^{\frac{\upsilon_{1i}+\upsilon_{2i}\mu_i}{2}\varepsilon}$. This is contradictory of the fact that $t_{k_0+1}^i - t_{k_0}^i < 2\varepsilon$. Therefore, we can exclude the Zeno behavior. The proof is completed. ■

## IV. RL-Based Event-Triggered PI Algorithm for Output Synchronization

In this part, we propose a data-driven method based on ADP to obtain the optimal control gains $K_i^*$. Then, an RL-based event-triggered PI algorithm is developed by combining the observer and the ADP method, such that the output synchronization problem can be solved.

We establish the augmented system as

$$\dot{X}_i = \bar{A}_i X_i + \bar{B}_i u_i \tag{37}$$

where $X_i = \begin{bmatrix} x_i \\ x_0 \end{bmatrix} \in \mathbb{R}^{p_i}$ with $p_i = n_i + n_0$, and the dynamics $\bar{A}_i = \begin{bmatrix} A_i & 0 \\ 0 & A_0 \end{bmatrix}$, $\bar{B}_i = \begin{bmatrix} B_i \\ 0 \end{bmatrix}$.

We consider the augment distributed state-feedback controller as follows

$$u_i = K_i X_i \tag{38}$$

with $K_i = [K_{1i}\ K_{0i}]$. It is noted that $x_0$ is assumed to be available for feedback control. In the later, the estimation of $x_0$, namely, $\eta_i$ will be used to replace $x_0$ by combining with the leader observer.

The output error is given by

$$\tilde{y}_i = y_i - y_0 \tag{39}$$

Then the performance function can be proposed as

$$\begin{aligned} J_i(y_i, u_i) = \int_t^\infty & e^{-\gamma_i(\tau - t)}((y_i - y_0)^T Q_i(y_i - y_0) \\ &+ u_i^T R_i u_i)d\tau \end{aligned} \tag{40}$$

where $\gamma_i$ is a positive discount factor, $Q_i$ and $R_i$ are symmetric positive definite weight matrices. It is noted that the term $(y_i - y_0)^T Q_i(y_i - y_0)$ represents the tracking error between the follower and the leader, while the term $u_i^T R_i u_i$ is the control cost. Define $\bar{C}_i = [C_i\ \ -C_0]$, and based on (38), then (40) can be reformulated as

$$\begin{aligned} J_i(X_i) &= \int_t^\infty e^{-\gamma_i(\tau - t)} X_i^T(\bar{C}_i^T Q_i \bar{C}_i + K_i^T R_i K_i)X_i d\tau \\ &= X_i^T P_i X_i \end{aligned} \tag{41}$$

where $P_i = \int_t^\infty e^{-\gamma_i(\tau - t)}(\bar{C}_i^T Q_i \bar{C}_i + K_i^T R_i K_i)d\tau$. By minimizing the performance function, we can determine the optimal control gains $K_i^*$ by

$$K_i^* = -R_i^{-1}\bar{B}_i^T P_i^* \tag{42}$$

where $P_i^*$ satisfies the following discounted ARE

$$\bar{A}_i^T P_i + P_i\bar{A}_i - \gamma_i P_i + \bar{C}_i^T Q_i \bar{C}_i - P_i\bar{B}_i R_i^{-1}\bar{B}_i^T P_i = 0. \tag{43}$$

*Lemma 6 ([14]):* The control policy (38) with the optimal control gains $K_i^*$ in (42) makes the output error (39) locally asymptotically converge to zero if

$$\gamma_i \leq \gamma_i^* = 2\|(\bar{B}_i R_i^{-1}\bar{B}_i^T Q_i)^{\frac{1}{2}}\|$$

*Remark 7:* This problem can be formulated as a linear quadratic Regulator (LQR) problem, and the optimal control gains can be found by solving the ARE. However, this approach necessitates full understanding of the system's dynamics. Moreover, due to the nonlinearity of the ARE equation, directly solving the equation for $P_i^*$ is highly challenging. Moving forward, we will adopt ADP to determine the optimal control gains $K_i^*$ without relying on the system dynamics.

An iterative method is adopted based on the knowledge of the system dynamics [36], where $P_i^*$ is given by the following lemma.

*Lemma 7 ([36]):* Let $P_i^{(j)}$, where $j = 0, 1, \dots$ represents the number of iterations, be the unique solution of the following ARE

$$0 = \bar{A}_i^{(j)T} P_i^{(j)} + P_i^{(j)} \bar{A}_i^{(j)} - \gamma_i P_i^{(j)} + \bar{C}_i^T Q_i \bar{C}_i + K_i^{(j)T} R_i K_i^{(j)} \tag{44}$$

where $\bar{A}_i^{(j)} = \bar{A}_i + \bar{B}_i K_i^{(j)}$, $K_i^{(j)} = -R^{-1} \bar{B}^T P_i^{(j-1)}$. $K_i^{(0)}$ is chosen to satisfy $\bar{A}_i^{(0)}$ has negative real parts. And then, we have $\lim_{j\to\infty} P_i^{(j)} = P_i^*$, $\lim_{j\to\infty} K_i^{(j)} = K_i^*$.

Next, we will develop an ADP method to get $K_i^*$ without knowledge of the system dynamics. First, we can rewrite the augmented system (37) as

$$\dot{X}_i = \bar{A}_i^{(j)} X_i + \bar{B}_i (u_i - K_i^{(j)} X_i) \tag{45}$$

By (41), (44) and (45), the Bellman equation is given by

$$\begin{aligned}
& e^{-\gamma_i \delta t} X_i^T(t+\delta t) P_i^{(j)} X_i(t+\delta t) - X_i^T(t) P_i^{(j)} X_i(t) \\
&= \int_t^{t+\delta t} \frac{d}{d\tau} (e^{-\gamma_i(\tau - t)} X_i(\tau)^T P_i^{(j)} X_i(\tau)) d\tau \\
&= \int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} (X_i^T(-\gamma_i P_i^{(j)} + \bar{A}_i^{(j)T} P_i^{(j)} + P_i^{(j)} \bar{A}_i^{(j)}) X_i \\
&\quad + 2(u_i - K_i^{(j)} X_i)^T \bar{B}_i^T P_i^{(j)} X_i) d\tau \\
&= -\int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} X_i^T (\bar{C}_i^T Q_i \bar{C}_i + K_i^{(j)T} R_i K_i^{(j)}) X_i d\tau \\
&\quad = -2 \int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} (u_i - K_i^{(j)} X_i)^T R_i K_i^{(j+1)} X_i d\tau
\end{aligned} \tag{46}$$

Replace $X_i^T(\bar{C}_i^T Q_i \bar{C}_i) X_i$ with $(y_i - y_0)^T Q_i (y_i - y_0)$. And then, the Bellman equation is

$$\begin{aligned}
& e^{-\gamma_i \delta t} X_i^T(t+\delta t) P_i^{(j)} X_i(t+\delta t) - X_i^T(t) P_i^{(j)} X_i(t) \\
&= -\int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} (y_i - y_0)^T Q_i (y_i - y_0) d\tau \\
&\quad - \int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} X_i^T (K_i^{(j)T} R_i K_i^{(j)}) X_i d\tau \\
&\quad - 2 \int_t^{t+\delta t} e^{-\gamma_i(\tau - t)} (u_i - K_i^{(j)} X_i)^T R_i K_i^{(j+1)} X_i d\tau
\end{aligned} \tag{47}$$

By Kronecker product representation, there holds

$$\begin{aligned}
& X_i^T P_i^{(j)} X_i = (X_i^T \otimes X_i^T) \mathrm{vec}(P_i^{(j)}) \\
& X_i^T (K_i^{(j)T} R_i K_i^{(j)}) X_i = (X_i^T \otimes X_i^T) \mathrm{vec}(K_i^{(j)T} R_i K_i^{(j)}) \\
& (u_i - K_i^{(j)} X_i)^T R_i K_i^{(j+1)} X_i \\
& = (X_i^T \otimes u_i^T)(I_{p_i} \otimes R_i) \mathrm{vec}(K_i^{(j+1)}) \\
& - (X_i^T \otimes X_i^T)(I_{p_i} \otimes K_i^{(j)T} R_i) \mathrm{vec}(K_i^{(j+1)})
\end{aligned}$$

We define

$$\Pi_i = \left[ e^{-\gamma_i(\tau - t_0)} \mathrm{vecv}(X_i) \mid_{t_0}^{t_1}, \dots, e^{-\gamma_i(\tau - t_{r-1})} \mathrm{vecv}(X_i) \mid_{t_{r-1}}^{t_r} \right]^T \tag{48}$$

$$\Phi_i = \left[ \int_{t_0}^{t_1} e^{-\gamma_i(\tau - t_0)} X_i \otimes X_i d\tau, \dots, \int_{t_{r-1}}^{t_r} e^{-\gamma_i(\tau - t_{r-1})} X_i \otimes X_i d\tau \right]^T \tag{49}$$

$$\Psi_i = \left[ \int_{t_0}^{t_1} e^{-\gamma_i(\tau - t_0)} X_i \otimes u_i d\tau, \dots, \int_{t_{r-1}}^{t_r} e^{-\gamma_i(\tau - t_{r-1})} X_i \otimes u_i d\tau \right]^T \tag{50}$$

$$\Theta_i = \left[ \int_{t_0}^{t_1} e^{-\gamma_i(\tau - t_0)} (y_i - y_0)^T Q_i (y_i - y_0) d\tau, \dots, \int_{t_{r-1}}^{t_r} e^{-\gamma_i(\tau - t_{r-1})} (y_i - y_0)^T Q_i (y_i - y_0) d\tau \right]^T \tag{51}$$

Then, (47) can be rewritten in the compact form

$$\Omega_i^{(j)} \begin{bmatrix} \mathrm{vecs}(\mathrm{P}_\mathrm{i}^{(\mathrm{j})}) \\ \mathrm{vec}(K_i^{(j+1)}) \end{bmatrix} = -\Upsilon_i^{(j)} \tag{52}$$

where $\Omega_i^{(j)} = [\Pi_i, 2(\Psi_i(I_{p_i} \otimes R_i) - \Phi_i(I_{p_i} \otimes K_i^{(j)T} R_i))]$, $\Upsilon_i^{(j)} = \Theta_i + \Phi_i \mathrm{vec}(K_i^{(j)T} R_i K_i^{(j)})$.

To calculate the solution to (52), we further need the assumption.

*Assumption 6 ([37]):* $\Omega_i^{(j)}$ has full column rank for all $j \in \mathbb{Z}_+$, i.e., $\mathrm{rank}([\Phi_i, \Psi_i]) = \frac{p_i(p_i+1)}{2} + p_i m_i$.

*Remark 8:* Assumption 6 is to ensure the existence and uniqueness of the solution, so that the least-squares method yields a unique solution in each iteration, thereby allowing $P_i^{(j)}$ and $K_i^{(j+1)}$ to be computed. In each iteration, we collect data at time instants $t_1, t_2, \dots, t_r$, and as long as $r$ is sufficiently large, the rank condition can be easily guaranteed.

Thus, under Assumption 6, the modified form of (52) is

$$\begin{bmatrix} \mathrm{vecs}(\mathrm{P}_\mathrm{i}^{(\mathrm{j})}) \\ \mathrm{vec}(K_i^{(j+1)}) \end{bmatrix} = -(\Omega_i^{(j)T} \Omega_i^{(j)})^{-1} \Omega_i^{(j)T} \Upsilon_i^{(j)}. \tag{53}$$

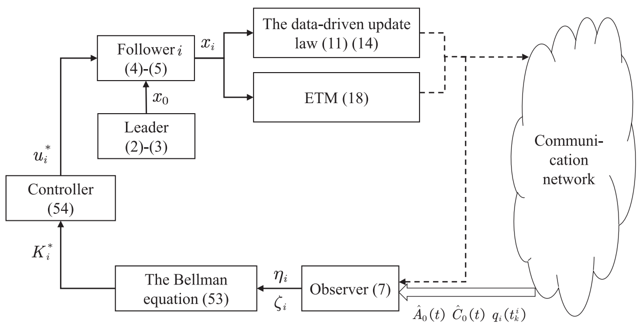

Fig. 1. Control block diagram.

*Remark 9:* By (53), it is evident that $P_i^{(j)}$ and $K_i^{(j+1)}$ can be obtained by computing the online input-output data. Furthermore, under Assumption 6, it follows from [37] that $P_i^{(j)}$ and $K_i^{(j+1)}$ converge to the optimal strategy $P_i^*$ and $K_i^*$ respectively.

The preceding analysis assumes that all followers can access to information from the leader. And then, we will integrate the designed event-triggered observer with the controller to address the output synchronization.

The observer (7) estimates the state and output of the leader for every follower. As a result, we can substitute $\eta_i$ for $x_0$ to represent the leader's state in (38) and $\zeta_i$ for $y_0$ to denote the leader's output. Thus, the modified optimal distributed controller is given as

$$u_i^* = K_i^* \hat{X}_i \tag{54}$$

where $\hat{X}_i = [x_i^T \ \ \eta_i^T]^T$ and $K_i^*$ is to be determined later.

According to (51), the calculation of $\Theta_i$ depends on $y_0(t)$, which is not available for every follower. We use $\zeta_i$ to replace $y_0$ in the calculation to get

$$\begin{aligned}\hat{\Theta}_i = \bigg[&\int_{t_0}^{t_1} e^{-\gamma_i(\tau-t_0)}\left(y_i-\zeta_i\right)^T Q_i\left(y_i-\zeta_i\right)d\tau, \ldots, \\ &\int_{t_{r-1}}^{t_r} e^{-\gamma_i(\tau-t_{r-1})}\left(y_i-\zeta_i\right)^T Q_i\left(y_i-\zeta_i\right)d\tau\bigg]^T\end{aligned} \tag{55}$$

Based on the above analysis, an RL-based event-Triggered PI algorithm for output synchronization will be given in Algorithm 1. The control block diagram is shown in Fig. 1.

*Theorem 2:* Under Assumptions 1–6, consider HMASs (2)–(5) with the event-triggered leader observer (7) and ETM (18). Let $u_i = u_i^*$ be the optimal control protocol obtained from Algorithm 1, then the output error $\tilde{y}_i$ in (39) will asymptotically converge to zero, provided that the discounted factor $\gamma_i$ satisfies $0 < \gamma_i \leq 2\|(\bar{B}_i R_i^{-1} \bar{B}_i^T Q_i)^{\frac{1}{2}}\|$.

*Proof:* The augment system dynamics is determined by

$$\begin{bmatrix}\dot{x}_i \\ \dot{\eta}_i\end{bmatrix} = \begin{bmatrix}A_i + B_i K_{1i} & B_i K_{0i} \\ 0 & \hat{A}_0(t)\end{bmatrix}\begin{bmatrix}x_i \\ \eta_i\end{bmatrix} + \begin{bmatrix}0 \\ \Delta_i\end{bmatrix} \tag{56}$$

where $\Delta_i = \beta q_i(t_k^i)$.

**Algorithm 1:** RL-Based Event-Triggered PI Algorithm for Output Synchronization.

```
Initialize:
Set $t_0^1 = t_0^2 = \ldots = 0$, $e_i(0) = 0$, $k = 0$, $j = 0$, $\forall i \in N$
Set the stable $\eta_i(0)$, $\mathscr{A}(0)$, $\mathscr{C}(0)$
Start with an arbitrary initial control policy:
    $u_i^{(0)} = K_i^{(0)} \hat{X}_i + \varsigma, i = 1, \ldots, N$
 where $\varsigma$ is an exploration signal, and $K_i^{(0)}$ is the
 stabilizing control gain
for $t = 0$ to $t_{end}$ (final time) do
  Compute $\hat{\mathscr{A}}(t)$ from (11) and $\hat{\mathscr{C}}(t)$ from (14)
  Compute $\eta_i(t)$ and $\zeta_i(t)$ from (7)
  if the ETM (18) is satisfied then
    Update $k \leftarrow k+1$, $q_i(t_k^i) = q_i(t)$, and $e_i(0) = 0$
  end if
  Using online data to compute $\Omega_i^{(j)}$ from (48), (49), and
  (50)
  Using online data to compute $\Upsilon_i^{(j)}$ from (49) and (55)
  if Assumption 6 holds then
    Solve $P_i^{(j)}$ and $K_i^{(j+1)}$ from the (53)
    $u_i^{(j+1)} = K_i^{(j+1)} \hat{X}_i + \varsigma, i = 1, \ldots, N$
    Update $j \leftarrow j+1$
  end if
  if $\|K_i^{(j)} - K_i^{(j-1)}\| \leq \varepsilon$ (a small positive constant) then
    Break
  end if
end for
Set $K_i^* = K_i^{(j)}$, $P_i^* = P_i^{(j-1)}$, and $u_i^* = K_i^{(j)} \hat{X}_i$.
```

According to the separation principle, because (56) is the block-triangular structure, the design of the event-triggered observer and the distributed controller can proceed independently of each other.

According to Theorem 1, the leader state estimation error $\tilde{\eta}$ in (19) and the leader output error $\tilde{\zeta}$ in (20) will exponentially converge to zero under the event-triggering mechanism (18), namely, $\lim_{t\to\infty}(\eta_i(t) - x_0(t)) = 0$ and $\lim_{t\to\infty}(\zeta_i(t) - y_0(t)) = 0, i = 1, \ldots, N$. Thus, it is obvious that $\lim_{t\to\infty}(\hat{\Theta}_i - \Theta_i) = 0, i = 1, \ldots, N$.

The uniqueness of solution to (53) is guaranteed under Assumption 6. It follows from Lemma 7 that $\lim_{j\to\infty} K_i^{(j)} = K_i^*$, which means that the solution calculated from (53) converges to (42). According to Lemma 6, under the boundary condition $\gamma_i \leq 2\|(\bar{B}_i R_i^{-1} \bar{B}_i^T Q_i)^{\frac{1}{2}}\|$, the output error (39) locally asymptotically converge to zero, i.e., $\lim_{t\to\infty}(y_i(t) - y_0(t)) = 0, i = 1, \ldots, N$. This completes the proof. ■

## V. Simulation Example

In this section, a simulation is used to validate the performance of the proposed model-free RL-based event-triggered output synchronization algorithm in HMASs.

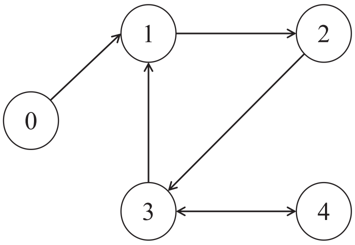

Fig. 2. Topology in the simulation.

We select a sinusoidal trajectoey generator as the leader, with its dynamics as follows:

$$A_0 = \begin{bmatrix} 0 & 2 \\ -2 & 0 \end{bmatrix}, \quad C_0 = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{57}$$

And the leader's initial state is $x_0 = [1{,}1]^T$. The four heterogeneous followers' dynamics are given as

$$\begin{aligned}
&A_1 = 0, && B_1 = 10, \quad C_1 = 1 \\
&A_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, && B_2 = \begin{bmatrix} 0 \\ 5 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & 2 \end{bmatrix} \\
&A_3 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, && B_3 = \begin{bmatrix} 0 \\ 2 \\ 6 \end{bmatrix}, \quad C_3 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\
&A_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}, && B_4 = \begin{bmatrix} 3 \\ 0 \\ 5 \end{bmatrix}, \quad C_4 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}
\end{aligned} \tag{58}$$

And the initial states of the four observers are set as $\eta_1 = [1, 2]^T$, $\eta_2 = [2, 1]^T$, $\eta_3 = [1, 0]^T$, $\eta_4 = [0, 1]^T$ and the simulation step size is $t = 0.001$ s.

The directed network topology graph of HMASs is shown in Fig. 2. In this context, agent 0 represents the leader, while agent 1-4 represent the followers.

In the first step, presented in Section II, we apply an experience-replay method to estimate $A_0$ and $C_0$. According to Lemmas 3 and 4, we select $\iota_a = 0.8$, $p_a = 500$ and $\iota_c = 0.8$, $p_c = 300$, respectively. The trajectory of $A_0$ and $C_0$ estimation is illustrated in Fig. 3. In the legend, $\hat{A}_{0i}$ and $\hat{C}_{0i}$ respectively represent the corresponding elements in the $\hat{A}_0$ and $\hat{C}_0$ matrices. From the figure, it can be observed that the method rapidly and precisely captures the system dynamics. To verify the importance of Assumptions 4 and 5, we set $p_a = 10$ and $p_c = 10$, which violates the required rank conditions. As shown in Fig. 4, although convergence can still be achieved, the convergence speed is significantly slower. The adaptive method in [14] is used to estimate the leader's system model, as illustrated in Fig. 5. Under the same initial conditions, the experience-replay approach proposed in this paper significantly improves the convergence speed. Specifically, our method achieves convergence within 3 seconds, whereas the method in [14] requires up to 10 seconds.

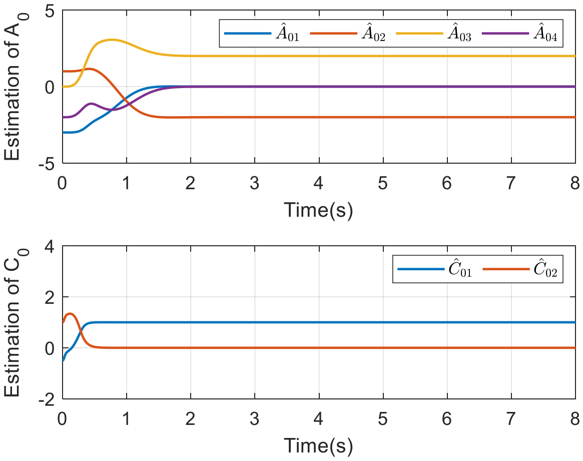

Fig. 3. Estimation of $A_0$ and $C_0$ besed on experience-replay with $p_a = 500$ and $p_c = 300$.

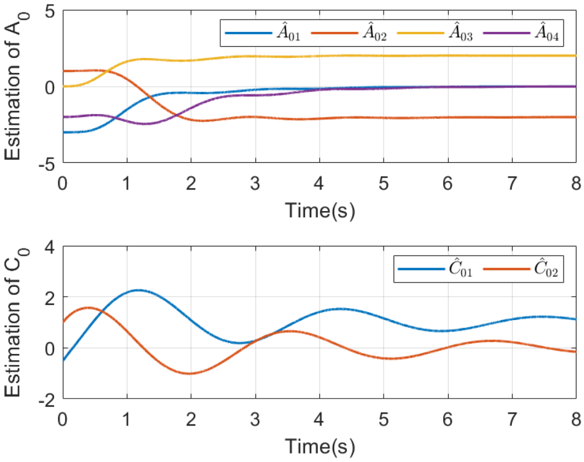

Fig. 4. Estimation of $A_0$ and $C_0$ besed on experience-replay with $p_a = 10$ and $p_c = 10$.

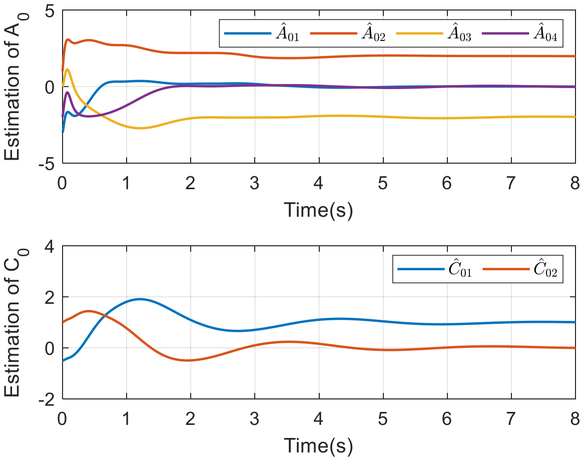

Fig. 5. Estimation of $A_0$ and $C_0$ in [14].

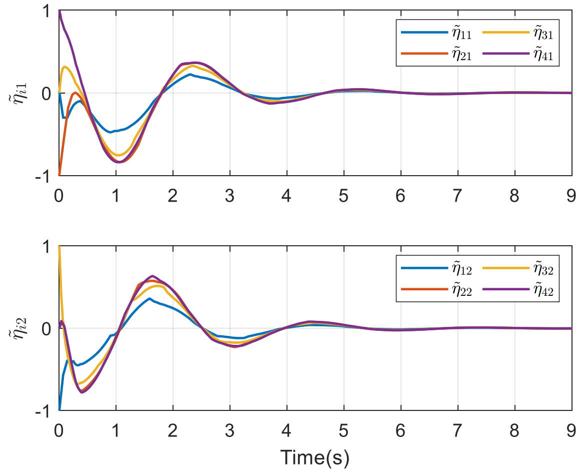

Fig. 6. Trajectory of state observation error.

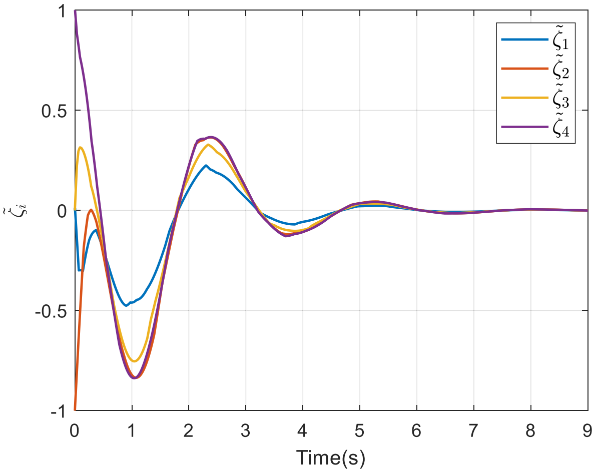

Fig. 7. Trajectory of output observation error.

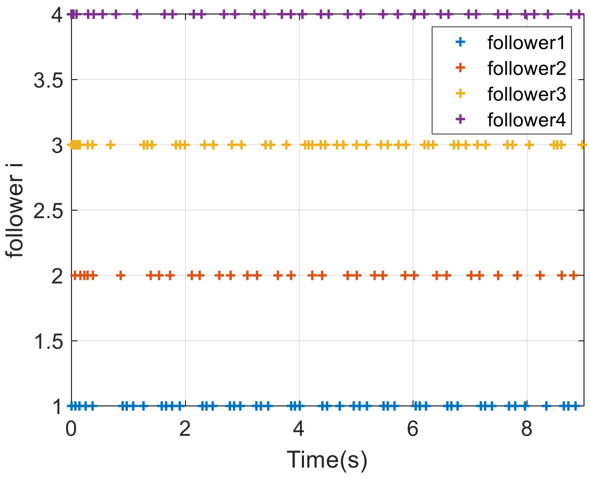

Fig. 8. Triggering time instants of each follower.

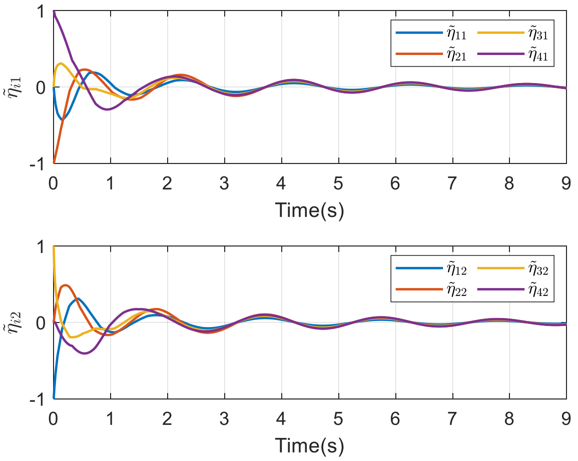

Fig. 9. Trajectory of state observation error in [26].

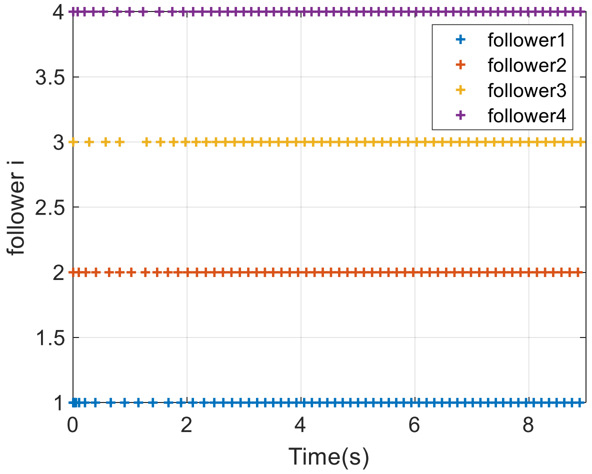

Fig. 10. Triggering time instants of each follower in [26].

In the presented event-triggered leader observer (7), we set $\beta = 3$ and in the ETM (18), we set $\kappa_i = 1, \mu_i = 1, \upsilon_{1i} = 0.5$ and $\upsilon_{2i} = 1$. The state observation error $\tilde{\eta}_i = \eta_i - x_0$ trajectories for the four observers are illustrated in Fig. 6, while the output observation error $\tilde{\zeta}_i = \zeta_i - y_0$ are presented in Fig. 7. In Fig. 6, $\tilde{\eta}_{i1}$ and $\tilde{\eta}_{i2}$ represent the first and second dimensions of the state estimation error, respectively. It is evident that this event-triggered observer can accurately estimate the leader's information. Furthermore, the event-triggering instants are recorded, displayed in Fig. 8. The communication of information between agents is effectively reduced. Compared to the event-triggered observer proposed in [26], we further assume the prior knowledge of the leader's system dynamics for the algorithm in [26]. The observer (7) demonstrates significant advantages in multiple aspects. The method in [26] exhibits a notably slower convergence rate of the state observation error (see Fig. 9), while also resulting in a significantly higher number of triggering events (see Fig. 10). Moreover, Table I further quantifies this advantage. Compared with the method in [26], the proposed observer in this chapter reduces the

TABLE I
COMPARISON BETWEEN OBSERVER (7) AND OBSERVER [26] IN TERMS OF TRIGGERING TIMES

| | Follower 1 | Follower 2 | Follower 3 | Follower 4 |
|---|---|---|---|---|
| Observer (7) | 35 | 30 | 34 | 25 |
| Observer [26] | 63 | 59 | 53 | 63 |

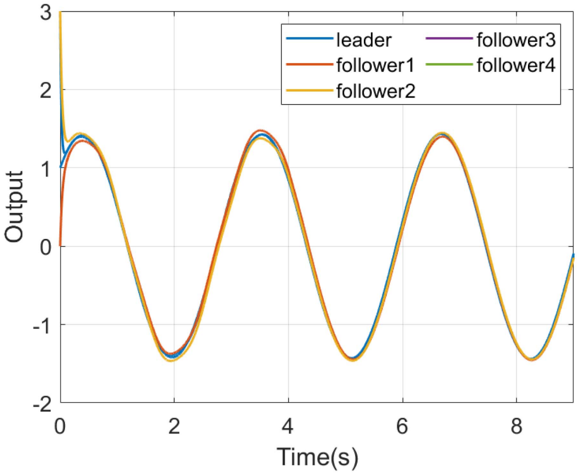

Fig. 11. Output synchronization trajectory of leader and followers.

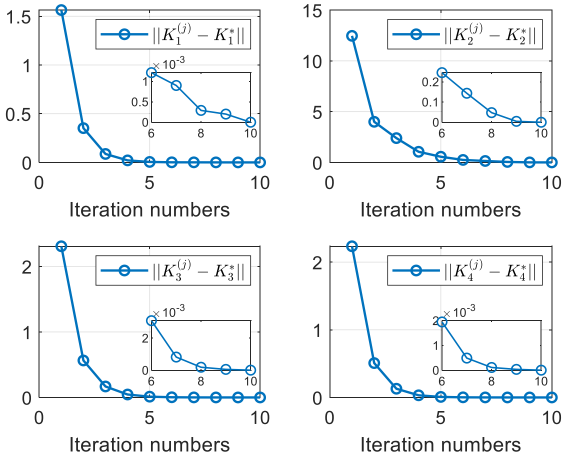

Fig. 12. Iterations of convergence of $\|K_i^{(j)} - K_i^*\|$.

number of triggering events by nearly half, effectively lowering communication burden and improving system efficiency.

Moving forward, we adopt the date-driven ADP algorithm to attain the optimal control strategy, consequently achieving output synchronization. Set the four followers' initial states to $x_1 = [2], x_2 = [1, 1]^T, x_3 = [1, -2, 1]^T, x_4 = [-1, 0, 4]^T$. The optimal control gains obtained through the iterative process of Algorithm 1 are

$$\begin{aligned}
K_1^* &= \begin{bmatrix} -3.5411 & 3.5364 & 0.0473 \end{bmatrix} \\
K_2^* &= \begin{bmatrix} -5.0888 & -10.2278 & 5.0711 & 0.0955 \end{bmatrix} \\
K_3^* &= \begin{bmatrix} -3.6079 & -3.6632 & -3.6311 & -3.6311 & 3.6297 \end{bmatrix} \\
K_4^* &= \begin{bmatrix} -3.5012 & -3.4620 & -3.5325 & 3.4911 & 0.0607 \end{bmatrix}
\end{aligned} \tag{59}$$

In the end, a comparison of the output synchronization trajectories between the followers and the leader is visualized in Fig. 11. The convergence of $\|K_i^{(j)} - K_i^*\|$ over iterations is shown in Fig. 12 and the convergence of $\|P_i^{(j)} - P_i^*\|$ over iterations is shown in Fig. 13. This algorithm is capable of learning optimal control gains and precisely tracking the leader's output.

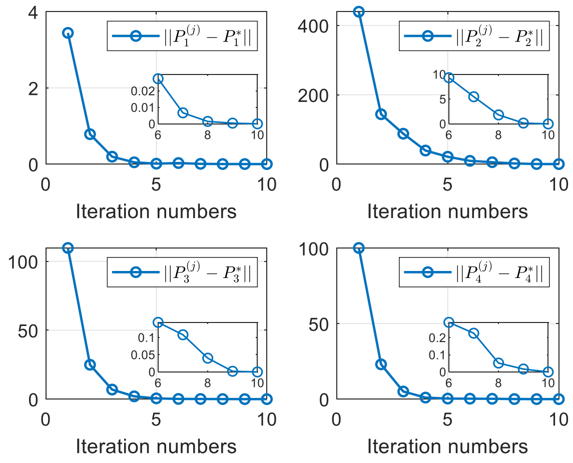

Fig. 13. Iterations of convergence of $\|P_i^{(j)} - P_i^*\|$.

In our method, both the observer and controller are designed in a distributed manner, where each agent performs its own computations and communicates only with its local neighbors. Therefore, the computational complexity does not scale with the total number of agents, and the performance is not significantly affected by network sparsity. This ensures the scalability and robustness of the proposed algorithm in large-scale multi-agent systems.

## VI. Conclusion

In summary, this paper has introduced a model-free event-triggered optimal control approach for output synchronization, employing RL-based algorithm for HMASs. The leader's system dynamics has been estimated using an experience-replay learning method. Building upon this, an event-triggered leader observer has been co-designed, and its effectiveness, as well as the exclusion of Zeno behavior, has been demonstrated. Subsequently, a data-driven algorithm based on ADP has been presented, enabling the derivation of an optimal control gains. The algorithm proposed in this paper not only achieves output synchronization control of HMASs without prior knowledge of systems, but also effectively reduces information exchange between agents. Finally, a simulation example has validated the proposed control strategy's performance.

## Appendix

### A. *Proof of Lemma 3*

*Proof:* Since (13) cannot be directly used because it requires the information of $A_0$. Thus, we introduce the filtered state $x_l(t)$, $z(t)$ and $\Lambda(t)$. Integrating (8), (9) and (10) yields

$$x_l(t) = e^{-kt} \int_0^t e^{k\tau} x_0(\tau) d\tau \tag{60}$$

$$z(t) = e^{-kt} \int_0^t e^{k\tau} \dot{x}_0(\tau) d\tau \tag{61}$$

$$\Lambda(t) = e^{-kt} \int_0^t e^{k\tau} Y(\tau) d\tau \tag{62}$$

According to (12), (61) and (62), it is not difficult to obtain

$$z(t) = \Lambda(t)\mathscr{A}. \tag{63}$$

It is noted that $z(t)$ cannot be calculated by (61) or (63) because they both depend on the knowledge of system dynamics. Thus, applying integration to (61) yields

$$\begin{aligned} z(t) &= x_0(t) - e^{-kt}x_0(0) - ke^{-kt}\int_0^t e^{k\tau}x_0(\tau)d\tau \\ &= x_0(t) - e^{-kt}x_0(0) - kx_l(t) \end{aligned} \tag{64}$$

where the last step is obtained by using (60).

It follows from (63) that (13) can be reformulated as

$$\phi_a(t) = z(t) - \Lambda(t)\hat{\mathscr{A}}(t) \tag{65}$$

from which $\phi_a^s(t)$ can be obtained only using the leader's past state information without knowing the leader's dynamics $A_0$.

Next, we prove that $\mathscr{A}$ can be estimated by using (11). Define the system parameter estimation error as

$$\delta_a(t) = \mathscr{A} - \hat{\mathscr{A}}(t) \tag{66}$$

And then consider the Lyapunov function

$$V_a = \frac{1}{2}\delta_a^T\delta_a \tag{67}$$

The error dynamics for $\delta_a$ is given by

$$\dot{\delta}_a(t) = -\iota_a\sum_{s=1}^{p_a}\Lambda^T(t_s)\phi_a^s(t) \tag{68}$$

According to (11) and (13), take time derivative of (67) as

$$\begin{aligned} \dot{V}_a &= -\iota_a\delta_a^T\sum_{s=1}^{p_a}\Lambda^T(t_s)\phi_a^s(t) \\ &= -\iota_a\delta_a^T\left(\sum_{s=1}^{p_a}\Lambda^T(t_s)\Lambda(t_s)\right)\delta_a \\ &\leq -2\iota_a\lambda_{\min}^{\Lambda}V_a \leq 0 \quad \forall t > t_{p_a} \end{aligned} \tag{69}$$

where $\lambda_{\min}^{\Lambda}$ is the minimum eigenvalue of the argument matrix $\sum_{s=1}^{p_a}\Lambda^T(t_s)\Lambda(t_s)$. Based on Lyapunov stability analysis, the estimation error $\delta_a(t)$ is convergent.

Moreover, under Assumption 4, the argument matrix is positive definite and then we have $\lambda_{\min}^{\Lambda} > 0$, we have

$$\dot{V}_a \leq -2\iota_a\lambda_{\min}^{\Lambda}V_a < 0 \quad \forall t > t_{p_a} \tag{70}$$

Therefore, $\delta_a(t)$ is exponentially decreasing $\forall t \geq t_{p_a}$, which means that $\hat{\mathscr{A}}$ exponentially converges to $\mathscr{A}$, namely, the estimation $\hat{A}_0$ exponentially converges to $A_0$ for $\forall t \geq t_{p_a}$. The proof is thus completed. ■

## References

[1] Y. Zheng, C. Zheng, X. Zhang, F. Chen, Z. Chen, and S. Zhao, "Detection, localization, and tracking of multiple MAVs with panoramic stereo camera networks," *IEEE Trans. Automat. Sci. Eng.*, vol. 20, no. 2, pp. 1226–1243, Apr. 2023.

[2] Z. Yan and Y. Xu, "A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4599–4608, Nov. 2020.

[3] W. Hu, Z. Li, M.-Z. Dai, and T. Huang, "Robust adaptive control for spacecraft attitude synchronization subject to external disturbances: A performance adjustable event-triggered mechanism," *Int. J. Robust Nonlinear Control*, vol. 33, no. 3, pp. 2392–2408, 2023.

[4] H. Modares, F. L. Lewis, and Z.-P. Jiang, "$H_\infty$ tracking control of completely unknown continuous-time systems via off-policy reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2550–2562, Oct. 2015.

[5] W. Hu, Y. Cheng, and C. Yang, "Leader-following consensus of linear multi-agent systems via reset control: A time-varying systems approach," *Automatica*, vol. 149, 2023, Art. no. 110824.

[6] Z. Chen, "Synchronization of frequency-modulated multiagent systems," *IEEE Trans. Autom. Control*, vol. 68, no. 6, pp. 3425–3439, Jun. 2023.

[7] J. Wang, Q. Wang, H.-N. Wu, and T. Huang, "Finite-time output synchronization and $H_\infty$ output synchronization of coupled neural networks with multiple output couplings," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 6041–6053, Dec. 2021.

[8] L. Zhao, S. Wen, M. Xu, K. Shi, S. Zhu, and T. Huang, "PID control for output synchronization of multiple output coupled complex networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1553–1566, May/Jun. 2022.

[9] K. G. Vamvoudakis, Y. Wan, F. L. Lewis, and D. Cansever, *Handbook of Reinforcement Learning and Control*. Berlin, Germany: Springer, 2021.

[10] P. Ning et al., "Diffusion-based deep reinforcement learning for resource management in connected construction equipment networks: A hierarchical framework," *IEEE Trans. Wireless Commun.*, vol. 24, no. 4, pp. 2847–2861, Apr. 2025.

[11] S. Sun, R. Chi, Y. Liu, and N. Lin, "Model-free adaptive iterative learning from communicable agents for nonlinear networks consensus," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 9, pp. 458–467, 2023.

[12] X. Shi, Y. Li, C. Du, C. Chen, G. Zong, and W. Gui, "Reinforcement learning-based optimal control for Markov jump systems with completely unknown dynamics," *Automatica*, vol. 171, 2025, Art. no. 111886.

[13] Y. Jiang, J. Fan, W. Gao, T. Chai, and F. L. Lewis, "Cooperative adaptive optimal output regulation of nonlinear discrete-time multi-agent systems," *Automatica*, vol. 121, 2020, Art. no. 109149.

[14] H. Modares, S. P. Nageshrao, G. A. D. Lopes, R. Babuška, and F. L. Lewis, "Optimal model-free output synchronization of heterogeneous systems using off-policy reinforcement learning," *Automatica*, vol. 71, pp. 334–341, 2016.

[15] C. Chen, H. Modares, K. Xie, F. L. Lewis, Y. Wan, and S. Xie, "Reinforcement learning-based adaptive optimal exponential tracking control of linear systems with unknown dynamics," *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4423–4438, Nov. 2019.

[16] Y. Xu and Z. Wu, "Data-efficient off-policy learning for distributed optimal tracking control of HMAS with unidentified exosystem dynamics," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 3, pp. 3181–3190, Mar. 2024.

[17] X. Wang, C. Zhao, T. Huang, P. Chakrabarti, and J. Kurths, "Cooperative learning of multi-agent systems via reinforcement learning," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 9, pp. 13–23, 2023.

[18] Y. Jiang, W. Gao, J. Wu, T. Chai, and F. L. Lewis, "Reinforcement learning and cooperative $H_\infty$ output regulation of linear continuous-time multi-agent systems," *Automatica*, vol. 148, 2023, Art. no. 110768.

[19] C. Nowzari, E. Garcia, and J. Cortés, "Event-triggered communication and control of networked systems for multi-agent consensus," *Automatica*, vol. 105, pp. 1–27, 2019.

[20] X. Yi, K. Liu, D. V. Dimarogonas, and K. H. Johansson, "Dynamic event-triggered and self-triggered control for multi-agent systems," *IEEE Trans. Autom. Control*, vol. 64, no. 8, pp. 3300–3307, Aug. 2019.

[21] W. Hu, Y. Hou, Z. Chen, C. Yang, and W. Gui, "Event-triggered consensus of multiagent systems with prescribed performance," *IEEE Trans. Autom. Control*, vol. 69, no. 8, pp. 5462–5469, Aug. 2024.

[22] H. Li, X. Liao, T. Huang, and W. Zhu, "Event-triggering sampling based leader-following consensus in second-order multi-agent systems," *IEEE Trans. Autom. Control*, vol. 60, no. 7, pp. 1998–2003, Jul. 2015.

[23] W. Hu, C. Yang, T. Huang, and W. Gui, "A distributed dynamic event-triggered control approach to consensus of linear multiagent systems with directed networks," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 869–874, Feb. 2020.

[24] H. Meng, L. Zhu, and H.-T. Zhang, "Output synchronization of linear heterogeneous multi-agent systems with periodic event-triggered output communication," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 4, pp. 1942–1951, Jul./Aug. 2023.

[25] M. S. Mahmoud and B. J. Karaki, "Output-synchronization of discrete-time multiagent systems: A cooperative event-triggered dissipative approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 1, pp. 114–125, Jan.–Mar. 2021.

[26] J. Sun, J. Zhang, H. Zhang, and R. Zhang, “Adaptive event-triggered control approach to the cooperative output regulation of heterogeneous multiagent systems under digraphs,” *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 3388–3395, May 2023.

[27] H. Li, Y. Wu, M. Chen, and R. Lu, “Adaptive multigradient recursive reinforcement learning event-triggered tracking control for multiagent systems,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 1, pp. 144–156, Jan. 2023.

[28] F. Zhao, W. Gao, T. Liu, and Z.-P. Jiang, “Adaptive optimal output regulation of linear discrete-time systems based on event-triggered output-feedback,” *Automatica*, vol. 137, 2022, Art. no. 110103.

[29] X. Shi, Y. Li, C. Du, Y. Shi, C. Yang, and W. Gui, “Fully distributed event-triggered control of nonlinear multiagent systems under directed graphs: A model-free DRL approach,” *IEEE Trans. Autom. Control*, vol. 70, no. 1, pp. 603–610, Jan. 2025.

[30] D. Tang, N. Pang, and X. Wang, “Reinforcement learning-based event-triggered constrained containment control for perturbed multiagent systems,” *IEEE Trans. Signal Inf. Process. Netw.*, vol. 10, pp. 820–832, 2024.

[31] J. Zhang and H. Zhang, “Adaptive event-triggered consensus of linear multiagent systems with resilience to communication link faults for digraphs,” *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 69, no. 7, pp. 3249–3253, Jul. 2022.

[32] Q. Li, L. Xia, R. Song, and L. Liu, “Output event-triggered tracking synchronization of heterogeneous systems on directed digraph via model-free reinforcement learning,” *Inf. Sci.*, vol. 559, pp. 171–190, 2021.

[33] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems: Optimal and Adaptive Design Approaches*. Berlin, Germany: Springer, 2013.

[34] H. Cai, F. L. Lewis, G. Hu, and J. Huang, “The adaptive distributed observer approach to the cooperative output regulation of linear multi-agent systems,” *Automatica*, vol. 75, pp. 299–305, 2017.

[35] Q. Ma, S. Xu, F. L. Lewis, B. Zhang, and Y. Zou, “Cooperative output regulation of singular heterogeneous multiagent systems,” *IEEE Trans. Cybern.*, vol. 46, no. 6, pp. 1471–1475, Jun. 2016.

[36] D. Kleinman, “On an iterative technique for riccati equation computations,” *IEEE Trans. Autom. Control*, vol. 13, no. 1, pp. 114–115, Feb. 1968.

[37] W. Gao and Z.-P. Jiang, “Adaptive dynamic programming and adaptive optimal output regulation of linear systems,” *IEEE Trans. Autom. Control*, vol. 61, no. 12, pp. 4164–4169, Dec. 2016.

**Wenfeng Hu** (Member IEEE) received the Ph.D. degree in mechanical and biomedical engineering from the City University of Hong Kong, Hong Kong, in 2016. He is currently an Associate Professor with the School of Automation, Central South University, Changsha, China. His research interests include multi-agent systems, networked control systems, event-triggered control, and high-speed train control and scheduling. He was the recipient of the Hunan Natural Science Foundation for Excellent Young Scholars and one of Huxiang Young Talents.

**Xuan Wang** was born in 1999. She received the B.S. degree in automation from the Beijing University of Technology, Beijing, China, in 2022, and the M.S. degree in artificial intelligence from Central South University, Changsha, China, in 2025. Her research interests include multi-agent systems, reinforcement learning, and event-triggered control.

**Meichen Guo** (Member, IEEE) is currently an Assistant Professor with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. She received the Ph.D. degree from the City University of Hong Kong, Hong Kong SAR, China, in 2017. During 2017–2018, she was a Postdoc Research Fellow with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, VIC, Australia. From 2018 to 2022, she was an FSE Fellow with Engineering and Technology Institute Groningen, University of Groningen, Groningen, The Netherlands. Her research interests include data-driven control, nonlinear control, distributed control, and agriculture applications.

**Biao Luo** (Senior Member, IEEE) received the Ph.D. degree from Beihang University, Beijing, China, in 2014. From 2014 to 2018, he was an Associate Professor and Assistant Professor with the Institute of Automation, Chinese Academy of Sciences, Beijing, China. He is currently a Professor with the School of Automation, Central South University, Changsha, China. His research interests include intelligent control, reinforcement learning, deep learning, and decision-making. Dr. Luo is an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, *Artificial Intelligence Review*, and the *Neurocomputing*. He is a also the Vice Chair of Adaptive Dynamic Programming and Reinforcement Learning Technical Committee, Chinese Association of Automation.

**Tingwen Huang** (Fellow, IEEE), photograph and biography not available at the time of publication.