aerospace

MDPI

Article

A Distributed Reinforcement Learning Method for Output Consensus of Heterogeneous Multi-Agent Systems with Event-Triggered Mechanisms

Mengna Quan [1,†], Bin Lan [2,3,†], Shike Long [2,3,†], Yongjun Wang [2,3] and Shanlin Sun [2,3,*]

[1] School of Electronic and Information Engineering, Guangxi Normal University, Guilin 541004, China
[2] Guangxi Key Laboratory of Low-Altitude Unmanned Autonomous System, Guilin University of Aerospace Technology, Guilin 541004, China
[3] University Engineering Research Center of Low-Altitude Intelligent Inspection Technology Application, Guilin University of Aerospace Technology, Guilin 541004, China
* Correspondence: sunsl@guat.edu.cn; Tel.: +86-136-2783-2672
† These authors contributed equally to this work.

Abstract

This paper investigates the output consensus problem in heterogeneous multi-agent systems. To address the challenges of traditional analytical methods in handling unknown dynamics and disturbances, a control framework is proposed that combines known model structures with a data-driven adaptive mechanism. The framework uses a distributed internal model to compensate for system heterogeneity and incorporates an event-triggered mechanism to reduce communication burden. To improve transient tracking performance, a reinforcement learning strategy based on centralized training and decentralized execution is introduced to adaptively optimize local feedback gains. Simulation results show that the proposed method effectively bounds closed-loop signals, achieves relatively fast convergence, and demonstrates some robustness and communication efficiency under process noise.

Keywords: heterogeneous multi-agent systems; output consensus; event-triggered control; reinforcement learning

1. Introduction

Over the past decades, multi-agent systems (MAS) have found widespread applications in cooperative robotics, sensor fusion, distributed generation, and cyber–physical systems [1–4]. A key objective of MAS is output consensus, where follower agents coordinate to track a leader using only locally available information [5,6]. In practical scenarios, MAS often consist of heterogeneous agents with diverse dynamical characteristics, ranging from simple linear behaviors to complex nonlinear dynamics. Such heterogeneity, combined with complex communication topologies and time-varying chirp-type disturbances, significantly complicates distributed controller design. Conventional output regulation methods provide a rigorous mathematical foundation by analytically solving regulator equations. However, obtaining precise analytical solutions can be highly demanding in the presence of unmodeled dynamics and stochastic noise, occasionally necessitating conservative parameter designs to maintain system stability. To address these practical challenges, hybrid control architectures have been explored. These methods combine the structural compensation of nominal models with data-driven learning. They aim to achieve robust

Check for updates

Academic Editor: Gokhan Inalhan

Received: 23 March 2026
Revised: 7 May 2026
Accepted: 18 May 2026
Published: 22 May 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

coordination across heterogeneous agents. Furthermore, they reduce the reliance on precise parameter tuning.

Classical approaches address heterogeneity through feedback linearization [6], adaptive backstepping [7], internal-model-based control [8,9], and observer-based estimation techniques [10,11]. Finite-time and prescribed-performance control strategies have also been developed to guarantee transient tracking accuracy and disturbance rejection [12]. Regulation is typically achieved by compensating model mismatches or designing robust feedback structures. However, while these methods effectively guarantee asymptotic tracking, they generally assume continuous exchange of state information. This requirement leads to increased communication and computation burdens, particularly in large scale networks. Although event-triggered mechanisms (ETM) have been widely adopted to alleviate this burden by transmitting data only when necessary, they inherently introduce intermittent and aperiodic reference signals into the closed-loop system. Under the combined effects of stochastic disturbances and ETM induced signal jumps, tuning the feedback gains becomes a delicate task. Traditional Lyapunov-based controllers elegantly ensure system stability. However, it is difficult to analytically derive a gain that balances tracking precision and communication efficiency. This non-trivial problem motivates the use of data-driven optimization. These techniques can adaptively tune control parameters through system interactions.

To alleviate communication load, several communication reduction strategies have been explored. Periodic or time-sampled schemes transmit data at fixed intervals, regardless of the system's evolution. In contrast, sparse or event-driven schemes transmit information only when necessary. Among these approaches, event-triggered control has received significant attention. In such schemes, agents broadcast information only when locally evaluated conditions exceed predefined thresholds. Distributed event-triggered designs guarantee Zeno-free behavior while preserving consensus performance [13–15]. Furthermore, novel adaptive event-triggered schemes have been actively developed to accommodate complex environments, such as achieving fault-tolerant H_∞ stabilization in networked cascade control systems [16]. Event-triggered mechanisms have also been combined with learning-based or neural-network-based controllers to handle disturbances and reduce update frequency, for example in quadrotor UAV trajectory tracking problems [17]. Extensions to heterogeneous and nonlinear MAS demonstrate effective leader-following and synchronization [18–20]. Compared with periodic sampling, event-triggered mechanisms adapt communication to system states and often achieve higher communication efficiency. Nonetheless, most existing designs assume known agent dynamics or require global network parameters. These assumptions limit applicability in uncertain environments. Furthermore, the intermittent nature of ETM updates makes it highly challenging to tune optimal feedback gains using traditional analytical methods.

To overcome these limitations, reinforcement learning (RL) offers a powerful data-driven tool. Instead of requiring precise algebraic models, RL learns optimal control policies through interactions with the environment [21]. RL has been successfully applied in MAS for consensus and disturbance rejection [22–24]. Compared with traditional adaptive methods, RL is uniquely suited for multi-objective optimization. It can simultaneously evaluate tracking precision, control effort, and communication frequency. This makes RL highly compatible with distributed event-triggered architectures [25,26]. However, achieving optimal output consensus for heterogeneous MAS under both stochastic noise and ETM constraints remains an open challenge.

Motivated by these observations, this paper proposes a data driven framework for heterogeneous MAS. The proposed scheme achieves robust output consensus under dynamic event-triggered communication. First, it utilizes a distributed internal model to compensate

for structural heterogeneity [27–29]. Second, a centralized training decentralized execution (CTDE) RL architecture is introduced to handle system uncertainties. Specifically, acting as a data-driven optimizer, the RL agent continuously tunes the local feedback gains using only available local information [30]. Rather than replacing the controller, the RL agent dynamically optimizes the gains to balance transient tracking and resource utilization. Consequently, the proposed framework improves control efficiency, reduces communication demand, and maintains robust consensus against process noise. The main contributions are summarized as follows:

1. The integration of the internal model principle and reinforcement learning is explored. The internal model addresses agent heterogeneity, while the RL agent adapts feedback gains, offering a data-driven alternative to traditional model-based parameter tuning.
2. Event-triggered communication introduces intermittent reference signals, complicating control parameter tuning. In response, the RL agent adjusts the gains to accommodate these discontinuous updates.
3. The method is evaluated through numerical simulations involving a 6-agent directed network. The simulation results demonstrate the feasibility and effectiveness of the proposed scheme.

2. Preliminaries and Problem Formulation

2.1. Overview of the Proposed Framework

This paper develops an event-triggered output consensus framework for heterogeneous multi-agent systems. The proposed architecture consist of three main components. First, each follower employs a local tracking controller. This controller driver the agent's output toward a locally generated reference signal. Second, a distributed event-triggered mechanism is introduced. It transmits state information only when specific conditions are violated, which helps reduce communication burdens. Third, a reinforcement learning module is integrated to dynamically tune the feedback gains. The RL agent utilizes local system data to continuously adjust these parameters and the overall information flow is illustrated in Figure 1.

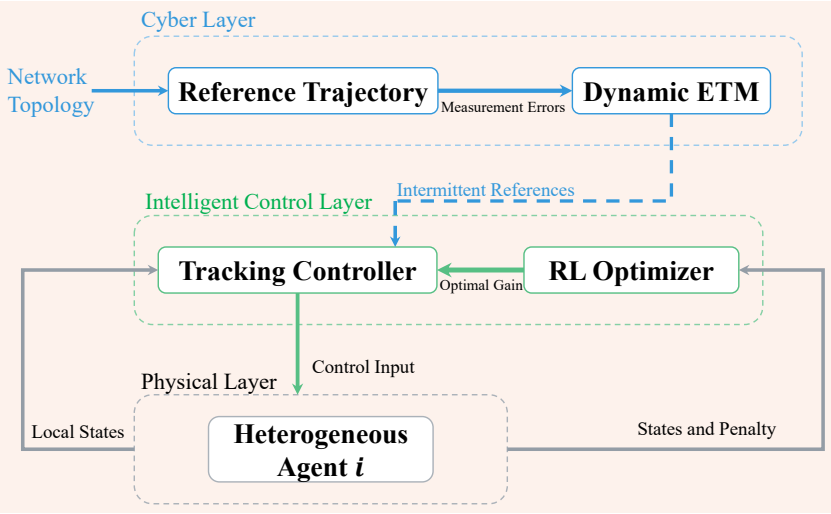


Figure 1. Overall framework of the proposed output consensus scheme.

2.2. Graph Theory and Notation

In this paper, $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ denotes a directed communication graph, consisting of a vertex set $\mathcal{V} = \{1, \ldots, n\}$ representing the followers, an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and an adjacency matrix $\mathcal{A}$. Vertices i and j are adjacent if $(i, j) \in \mathcal{E}$, implying that agent i can

sample information from agent j. The adjacency matrix is defined as $\mathcal{A} = [a_{ij}]_{n\times n}$, where $a_{ij} = 1$ if $(i,j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The neighborhood of vertex i is denoted by $\mathcal{N}_i = \{j \in \mathcal{V} \,|\, (i,j) \in \mathcal{E}\}$. The graph is undirected if $(i,j) \in \mathcal{E} \Leftrightarrow (j,i) \in \mathcal{E}$, and it is connected if a path exists between any two vertices. For leader-follower communication, the pinning gain $l_i \in \{0,1\}$ indicates whether follower i can directly access the leader's information; specifically, $l_i = 1$ if direct communication exists, and $l_i = 0$ otherwise. The Laplacian matrix of the follower graph $\mathcal{G}$ is defined as $L = D - \mathcal{A}$, with the degree matrix $D = \mathrm{diag}(|\mathcal{N}_1|, \ldots, |\mathcal{N}_n|)$. To capture the topology of the entire system including the leader, we define the matrix $H = L + D_l$, where $D_l = \mathrm{diag}(l_1, \ldots, l_n)$. The extended matrix over the state space is given by $\mathcal{H} = H \otimes I_q$, where $\otimes$ denotes the Kronecker product and I_q is the $q \times q$ identity matrix. Finally, the blkdiag operator constructs block-diagonal matrices from individual local matrices, which is used to assemble aggregated controller gains.

Assumption 1. *For the heterogeneous multi-agent system (1) and (2) to achieve output consensus, it is necessary that the communication graph contains a directed path from at least one leader to each follower agent.*

2.3. Problem Formulation

This paper considers a multi-agent system consisting of a leader and n followers. The dynamics of the leader are given by

$$\begin{cases} \dot{x}_0(t) = Sx_0(t) \\ y_0(t) = C_0 x_0(t) \end{cases} \tag{1}$$

where $x_0 \in \mathbb{R}^q$ denotes the leader state, $S \in \mathbb{R}^{q\times q}$ and $C_0 \in \mathbb{R}^{m\times q}$ are constant matrices.

The dynamics of the n heterogeneous followers are described as

$$\begin{cases} \dot{x}_i(t) = f_i(x_i,t) + B_i u_i(t) \quad i = 1, \ldots, n \\ y_i(t) = C_i x_i(t) \end{cases} \tag{2}$$

where $x_i \in \mathbb{R}^q$ and $u_i \in \mathbb{R}^p$ denote the state and control input of the ith follower, respectively. The function $f_i(\cdot) : \mathbb{R}^q \times \mathbb{R} \to \mathbb{R}^q$ represents the local heterogeneous dynamics of the ith agent, which are assumed to be known to its local controller. $B_i \in \mathbb{R}^{q\times p}$, $C_i \in \mathbb{R}^{m\times q}$ are constant matrices. The followers are heterogeneous in the sense that $f_i(\cdot)$, B_i, and C_i may differ across agents.

Assumption 2. *The matrix B_i is column full rank for $i = 1, \ldots, n$.*

Remark 1. *If B_i is not of full column rank, without loss of generality, its rank can be assumed to be k, $k \leq \min(q,p)$. According to the full-rank decomposition property of the matrix, one can always find a matrix $B'_{q\times k}$ with full column rank and a matrix $B''_{k\times p}$ with full row rank such that $B_{q\times p} = B'_{q\times k} B''_{k\times p}$. So $B_{q\times p} u_{p\times 1}(t) = B'_{q\times k} u'_{k\times 1}(t)$ can be obtained, where $u'_{k\times 1}(t) = B''_{k\times p} u_{p\times 1}(t)$. Since the matrix $B''_{k\times p}$ is full row rank, it follows that for any given $u'_{k\times 1}(t)$, there must exist $u_{p\times 1}(t)$ such that $u'_{k\times 1}(t) = B''_{k\times p} u_{p\times 1}(t)$ holds, for example by taking $u'_{k\times 1}(t) = {B''_{p\times k}}^T ({B''_{k\times p}} {B''_{p\times k}}^T)^{-1} u'_{k\times 1}(t)$. Based on the above analysis, Assumption 2 can in fact be relaxed as long as a full rank decomposition of matrix $B_{q\times p}$ can be found. But for the brevity of the main results section of this article, Assumption 2 is added to omit how to find the full-rank decomposition of the matrix $B_{q\times p}$.*

Assumption 3. *There exists a function $r_i(x_i, t) : R^q \times R \longrightarrow R^{p_i}$ such that $Sx_i = f_i(x_i, t) + B_i r_i(x_i, t)$ for $i = 1, \ldots, n$.*

Remark 2. *When $f_i(x_i, t) = A_i x_i(t)$, the follower agents are governed by linear dynamics. In [31], the output consensus problem for general linear multi-agent systems was studied under the assumption that there exists a matrix U_i such that $S = A_i + B_i U_i$. Under this condition, one can naturally define $r_i(x_i, t) = U_i x_i$, which satisfies Assumption 3. Therefore, the class of general linear systems considered in [31] constitutes a special case of the system model investigated in this paper. Moreover, assumptions analogous to Assumption 3 have also been adopted in the study of nonlinear output regulation problems,similar to [21].*

Remark 3. *For a follower agent i that is not directly connected to the leader, the leader dynamics matrix S is generally unknown, which renders the function $r_i(x_i, t)$ unknown as well. For general linear systems, an adaptive tuning law was proposed in [32] to estimate the feedback gain U_i online by exploiting an estimate of S. In contrast, this paper focuses on the online estimation of the function $r_i(x_i, t)$ for nonlinear systems by leveraging an estimated leader dynamics matrix, thereby extending existing adaptive consensus results from linear to nonlinear settings.*

Assumption 4. *The matrix pair (S, B_i) is stabilizable for $i = 1, \ldots, n$.*

The output consensus for the multi-agent system consisting of (1) and (2) is formally defined as follows.

Definition 1. *The heterogeneous multi-agent system achieves asymptotic output consensus if, under the designated control law $u_i(t)$, that is, the output of each follower agent tracks the leader's output trajectory asymptotically.*

$$\lim_{t \to \infty} \|y_i(t) - y_0(t)\| = 0 \quad i = 1, \ldots, n \tag{3}$$

then the multi-agent system (1) and (2) achieves output consensus.

2.4. Distributed Tracking Controller

First, consider the following dynamic state-feedback controller designed for each agent as

$$\begin{cases} u_i(t) = \hat{r}_i(x_i, t) - K_i\,(x_i - z_i) \\ \dot{z}_i(t) = \check{S}_i(t)\, z_i + \eta_i(t) \\ \eta_i(t) = -\varsigma_1 \Big[\sum_{j \in \mathcal{N}_i} a_{ij} \big(z_i(t) - \hat{z}_j(t)\big) + l_i \big(z_i(t) - x_0(t)\big) \Big] \end{cases} \tag{4}$$

where $u_i(t)$, $z_i(t)$, and $\eta_i(t)$ denote the control input, the local reference trajectory (observer state), and the distributed consensus feedback of follower i, respectively. The feedforward compensation term is defined as $\hat{r}_i(x_i, t) = (B_i^T B_i)^{-1} B_i^T \big(\check{S}_i(t) x_i - f_i(x_i, t)\big)$. Note that Assumption 2 guarantees that $B_i^T B_i$ is invertible. Under Assumption 4, the feedback gain matrix $K_i \in \mathbb{R}^{p \times q}$ is selected such that $S - B_i K_i$ is Hurwitz. The observer initial condition is set as $z_i(0) = 0$. The matrix $\check{S}_i(t)$ represents the local adaptive estimate of the leader's system matrix.

To alleviate communication load, the state information is exchanged via an event-triggered mechanism. On each sampling interval $t \in [t_k^j, t_{k+1}^j)$, the broadcast variable $\hat{z}_j(t)$ is initialized at the trigger instant as $\hat{z}_j(t_k^j) = z_j(t_k^j)$ and dynamically evolves according to $\dot{\hat{z}}_j(t) = \hat{S}_j(t) \hat{z}_j(t)$ rather than remaining static. At the k-th broadcast instant ($k \in \mathbb{N}^+$), agent i transmits its current state $z_i(t_k^i)$ to its neighbors. The coupling gain $\varsigma_1 > 0$ must be selected such that the nominal consensus generator matrix $A_\xi = I_n \otimes S - \varsigma_1 \mathcal{H}$ is Hurwitz.

Remark 4. *Due to the unknown dynamics of the leader's system matrix S for follower i, determining a feedback gain K_i that renders the error dynamics Hurwitz is challenging. Traditional adaptive methods often rely on solving complex Riccati equations online or requiring persistent excitation conditions, which are computationally expensive and difficult to verify in distributed settings. To address this, this paper proposes a data-driven reinforcement learning approach to adaptively learn optimal gains K_i online, bypassing the need for explicit solution of regulator equations. This learning-based strategy will be detailed in the subsequent section.*

Remark 5. *Similar to Remark 2, when the system model reduces to to a general linear model, that is, $f_i(x_i, t) = A_i x_i$, there is $r_i(x_i, t) = U_i x_i(t)$. If the leader coefficient matrix S is known, then the control input $u_i(t)$ in (4) reduces to $u_i(t) = (U_i - K_i)x_i + K_i z_i$. This controller is equivalent to that proposed in [33]. Moreover, since $S = A_i + B_i U_i$, the condition that $A_i + B_i(U_i - K_i)$ is Hurwitz is equivalent to requiring $S - B_i K_i$ to be Hurwitz.*

Remark 6. *The condition that $A_{\tilde{\xi}}$ is Hurwitz can always be satisfied by choosing a sufficiently large coupling gain ς_1. Let λ_k $(k = 1, \ldots, q)$ and μ_s $(s = 1, \ldots, n)$ denote the eigenvalues of S and H, respectively. According to the properties of the Kronecker product, the eigenvalues of $A_{\tilde{\xi}} = (I_n \otimes S) - \varsigma_1(H \otimes I_q)$ are given by $\lambda_k - \varsigma_1 \mu_s$. Under Assumption 1, the real parts of all eigenvalues μ_s of H are strictly positive. Therefore, choosing $\varsigma_1 > \max_{k,s} \frac{Re(\lambda_k)}{Re(\mu_s)}$ strictly guarantees that $A_{\tilde{\xi}}$ is a Hurwitz matrix. Although the exact bounds of the eigenvalues are typically difficult to compute globally, the condition is robustly met by simply selecting ς_1 to be sufficiently large.*

Remark 7. *It is worth noting that the nominal consensus generator matrix $A_{\tilde{\xi}}$ is structurally decoupled from the local feedback gains K_i. Therefore, the real-time adaptation of K_i by the RL optimizer does not affect the stability margins of the distributed observer. The stability sensitivity is instead concentrated on the closed-loop error matrix A_d, which is managed by a specific stability reward design detailed subsequently.*

Since most followers cannot directly access the leader's coefficient matrix S, each follower estimates it in a distributed manner using the following update law:

$$\dot{\check{S}}_i(t) = -\varsigma_2 \left[\sum_{j \in \mathcal{N}_i} a_{ij} \left(\check{S}_i(t) - \hat{S}_j(t) \right) + l_i \left(\check{S}_i(t) - S \right) \right] \tag{5}$$

where $\varsigma_2 > 0$ is a scalar gain, $\check{S}_i(0) = 0$ denotes the initial condition, and $\hat{S}_i(t) = \check{S}_i(s_k^i)$ for $t \in [s_k^i, s_{k+1}^i)$, $i = 1, \ldots, n$. Similar to the treatment of $z_i(t)$ in (4), each agent broadcasts $\check{S}_i$ to its neighbors only at the triggering instants s_k^i, thereby reducing communication load.

Remark 8. *It is worth noting that even follower agents with direct access to the leader's coefficient matrix S employ the estimated value $\check{S}_i(t)$. This design choice maintains a unified formulation of the estimator and controller across all follower agents, which facilitates potential hardware implementation. A similar design philosophy can be found in [32]. Nevertheless, it is feasible to allow follower agents with direct access to S to use its true value. In this case, define $l_i^0 = 1$ if $l_i > 0$, and $l_i^0 = 0$ otherwise. Then, $\check{S}_i(t)$ can be replaced by $\check{S}_i^0(t)$, where $\check{S}_i^0(t) = l_i^0 S + (1 - l_i^0)\check{S}_i(t)$. This modification does not affect the subsequent theoretical analysis.*

2.5. Event-Triggered Mechanism

Second, an event-triggered mechanism is designed to determine when follower agent i broadcasts information to its neighbors, thereby avoiding continuous communication and reducing the communication load. The corresponding measurement errors are defined as

$$\hat{e}_{z_i}(t) = \hat{z}_i(t) - z_i(t) \tag{6}$$

$$\hat{e}_{\check{S}_i}(t) = \hat{S}_i(t) - \check{S}_i(t) \tag{7}$$

To adaptively adjust the triggering threshold, a dynamic internal variable $\hat{\sigma}_i(t)$ is introduced and evolves as

$$\dot{\hat{\sigma}}_i(t) = -\lambda_i \hat{\sigma}_i(t) + \|\hat{e}_{z_i}\| \tag{8}$$

where $\lambda_i > 0$ is a design parameter. The triggering instants are determined by the following conditions

$$\begin{cases} t_{k+1}^i = \inf\left\{t > t_k^i \mid \|\hat{e}_{z_i}(t)\| \geq v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t)\right\} \\ s_{k+1}^i = \inf\left\{t > s_k^i \mid \|\hat{e}_{\check{S}_i}(t)\| \geq v_{2i}e^{-\alpha_{2i}t} + \gamma_{2i}e^{-\alpha'_{2i}t}\|\check{S}_i(t)\|\right\} \end{cases} \tag{9}$$

where $t_1^i = 0$ and $s_1^i = 0$ denote the initial triggering instants for agent i. The threshold components $v_{1i}e^{-\alpha_{1i}t}$, $v_{2i}e^{-\alpha_{2i}t}$, $\gamma_{1i}\hat{\sigma}_i(t)$, and $\gamma_{2i}e^{-\alpha'_{2i}t}\|\check{S}_i(t)\|$ are designed to satisfy the following analytical properties:

1. They are bounded and piecewise continuous for all $t \geq 0$.
2. As $t \to \infty$, all triggering thresholds asymptotically decay to zero, thereby ensuring asymptotic convergence.

To exclude Zeno behavior, the following conditions are imposed

1. For any finite time interval $[0, T]$, there exists a uniform constant $\varepsilon_1 > 0$, such that $v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t) \geq \varepsilon_1, \forall t \in [0, T)$.
2. Similarly, for any finite time interval $[0, T]$, there exists a uniform constant $\varepsilon_2 > 0$, such that $v_{2i}e^{-\alpha_{2i}t} + \gamma_{2i}e^{-\alpha'_{2i}t}\|\check{S}_i(t)\| \geq \varepsilon_2, \forall t \in [0, T)$.

Remark 9. *The proposed event-triggered scheme is intrinsically dynamic due to the introduction of the internal variable $\hat{\sigma}_i(t)$, which evolves according to $\dot{\hat{\sigma}}_i(t) = -\lambda_i\hat{\sigma}_i(t) + \|\hat{e}_{z_i}(t)\|$. As demonstrated in [34], this structure ensures that $\hat{\sigma}_i(t) \geq 0$ for all $t \geq 0$. By combining an exponentially decaying baseline with this adaptive component, such as $v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t)$, the triggering threshold becomes sensitive to recent measurement errors. Specifically, it increases during transient phases to reduce triggering frequency, and gradually tightens as the system approaches consensus. Similarly, in the matrix-triggered channel, the term $e^{-\alpha'_{2i}t}\|\check{S}_i(t)\|$ captures the evolution of the estimator and preserves sufficient responsiveness during parameter adaptation. Consequently, all threshold terms remain strictly positive over any finite time interval, successfully excluding Zeno behavior while relying solely on local quantities without needing global spectral information.*

Remark 10. *Many existing event-triggered mechanisms [35–37] employ triggering conditions that require continuous monitoring of neighboring agents' real-time states $x_j(t)$. Such designs implicitly necessitate continuous communication, which contradicts the fundamental objective of resource conservation. In contrast, the triggering mechanism (9) proposed in this study utilizes error vectors $\hat{e}_{z_i}(t)$ and $\hat{e}_{\check{S}_i}(t)$ that are computed strictly from the local information of agent i. This self-contained design significantly improves communication efficiency.*

Remark 11. *The terms $v_{1i}e^{-\alpha_{1i}t}$, $v_{2i}e^{-\alpha_{2i}t}$, $\gamma_{1i}\hat{\sigma}_i(t)$ and $\gamma_{2i}e^{-\alpha'_{2i}t}$ are typically designed as non-negative functions that asymptotically decay to zero. The selection of initial values and decay rates*

directly determines the triggering frequency. In particular, appropriate tuning of decay rates is critical for minimizing communication usage while preserving convergence performance.

2.6. Reinforcement Learning Based Gain Adaptation

This section investigates the feedback gain tuning problem for the heterogeneous multi-agent system. The objective is to design a learning based policy that balances tracking precision and control effort. Classical optimal control methods typically require solving coupled Hamilton–Jacobi–Bellman (HJB) equations analytically [38]. However, finding exact analytical solutions is generally highly complex, particularly in heterogeneous networks subjected to intermittent communication updates. To address this computational challenge, a reinforcement learning approach is adopted. Serving as a data-driven tuner, the RL algorithm allows the feedback gains K_i to be adjusted directly through interaction data. This pragmatic alternative effectively bypasses the necessity of analytically solving the complex HJB equations [39].

2.6.1. Formulation of State and Action Spaces

In the proposed leader–follower framework, each follower constructs its RL state using only locally available information, thereby ensuring a fully decentralized and data-driven execution. The local observation vector of follower i is defined as:

$$o_i(t) = \left[x_i^T(t), z_i^T(t), \mathrm{vec}\left(\check{S}_i(t)\right)^T\right]^T \tag{10}$$

where $\mathrm{vec}(\cdot)$ denotes the matrix vectorization operator.

The action space is defined to directly specify the feedback gain used in the local controller. Specifically, at each decision step, the RL action of follower i directly determines the gain matrix K_i in the control law

$$u_i(t) = \hat{r}_i(x_i, t) - K_i(x_i(t) - z_i(t)) \tag{11}$$

where $\hat{r}_i(x_i, t)$ is the feedforward term. In other words, the RL policy outputs the complete gain matrix K_i, rather than performing incremental adjustments.

This design allows the RL module to directly determine the feedback behavior of each follower while leaving the event-triggering condition unchanged. Consequently, the triggering frequency is entirely governed by the predefined distributed event-triggered mechanism.

2.6.2. Design of the Reward Function

In multi-agent reinforcement learning, the reward function fundamentally shapes the behavior of the learned policy. For the heterogeneous leader–follower output consensus problem, the reward function should encourage accurate tracking and efficient control effort, while ensuring closed-loop stability of the learned gains. Importantly, the reinforcement learning module is used solely to optimize the feedback gain K_i and does not affect the distributed event-triggering mechanism or its triggering frequency. The reward function for follower i at time t is defined as

$$r_i(t) = \beta_1 r_{\text{tracking}} + \beta_2 r_{\text{control}} + \beta_3 r_{\text{stability}} + \beta_4 r_{\text{trigger}} \tag{12}$$

where β_1, β_2 β_3 and $\beta_4 > 0$ are weighting coefficients.

The tracking reward penalizes deviations from the leader's output

$$r_{tracking} = -e_i(t)^T W_i e_i(t) \tag{13}$$

where $e_i(t) = x_i(t) - x_0(t)$ denotes the state tracking error, and $W_i > 0$ is a positive definite weight matrix. This quadratic penalty monotonically discourages large tracking errors and promotes output consensus.

To prevent excessive actuation and promote energy efficiency, the control effort is penalized by

$$r_{control} = -u_i^T(t) R_i u_i(t) \tag{14}$$

where $R_i > 0$. This term stabilizes training by suppressing aggressive control actions and mitigating actuator saturation.

Since the RL agent directly outputs the feedback gain matrix $K_i(t)$, it is crucial to constrain the search space to prevent aggressive or destabilizing control policies. To this end, a structural regularization penalty is introduced into the reward function:

$$r_{stability} = -\kappa \cdot \mathbb{I}_{\text{stability}}(K_i(t)) \tag{15}$$

where $\kappa > 0$ denotes the penalty magnitude. The term $\mathbb{I}_{\text{stability}}(K_i(t))$ equals a positive integer if the generated matrix $K_i(t)$ fails to be positive definite, and equals 0 otherwise.

An event-triggering penalty $r_{\text{trigger}} = -\mathbb{I}_{trigger}(t)$ is included, where $\mathbb{I}_{trigger}(t) = 1$ if a communication event is triggered at time t and 0 otherwise. While the RL agent does not directly govern the ETM thresholds, the inclusion of this penalty is intended to guide the policy toward smoother control actions.

2.6.3. Architecture of the Neural Networks

The neural network architecture follows the CTDE paradigm, which is well suited for multi-agent reinforcement learning in output consensus problems. Figure 2a illustrates the architecture of the Actor network used for policy learning [40]. Each follower agent i is equipped with a dedicated actor network that maps its local observations to control actions [41]. Figure 2b depicts the architecture of the shared Critic network. A centralized critic network is employed during training to estimate the value function using global information.

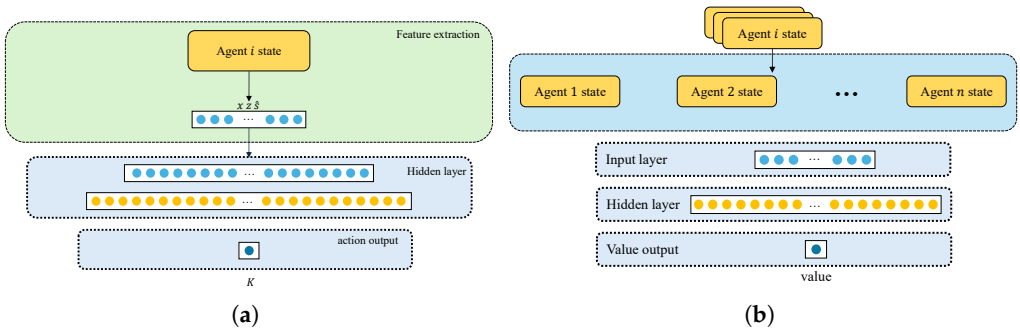


Figure 2. Architecture of the actor and critic networks used in the RL framework. (**a**) Actor network architecture. (**b**) Critic network architecture.

2.6.4. Training Procedure

The training framework follows the CTDE paradigm, as illustrated in Figure 3. During training, all agents access global information through the centralized critic network, enabling more stable and coordinated learning. Each agent selects its action based on its local observation and policy parameters. After executing the action at time t, the agents receive the reward r_t^i, their local observation o_t^i, and the joint observations and actions. These signals guide the learning process and help each agent choose control actions that achieve output consensus while respecting system constraints.

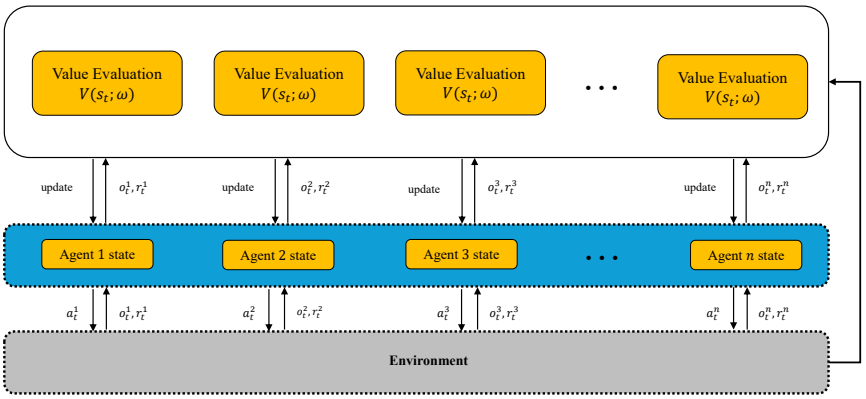


Figure 3. Centralized training and distributed execution framework.

2.7. Theoretical Analysis

The following analysis will focus on the convergence of the estimates $\check{S}_i(t)$ to the leader's coefficient matrix S, and the correctness of the reference trajectory $z_i(t)$.

Lemma 1. *For the multi-agent system* (1) *and* (2) *satisfying Assumption* 1*, asymptotic estimation of the leader's coefficient matrix S by each follower agent i is achieved under the combined action of the* (5) *and the event-triggered mechanism* (9)*. In particular,*

$$\lim_{t\to\infty}\left\|\check{S}_i(t)-S\right\|=0 \quad i=1,\ldots,n \tag{16}$$

Proof. Let $\tilde{S}_i(t)=\check{S}_i(t)-S$ denote the local estimation error of the ith follower and define the aggregated error vector

$$\tilde{S}(t)=\left[\tilde{S}_1^T(t),\ldots,\tilde{S}_n^T(t)\right]^T \tag{17}$$

Using the estimator law (5) and stacking all agents, the estimation error dynamics can be written in compact form as

$$\dot{\tilde{S}}(t)=-\varsigma_2(\mathcal{H}\otimes I_q)\tilde{S}(t)+E_S(t) \tag{18}$$

where $\mathcal{H}$ is the extended Laplacian matrix and $E_s(t)$ collects sampling-induced perturbations due to the event-triggered communication.

By Assumption 1 the matrix $\mathcal{H}$ is a positive definite M-matrix; therefore, the matrix

$$A=-\varsigma_2(\mathcal{H}\otimes I_q) \tag{19}$$

is Hurwitz. Hence there exists a symmetric positive definite matrix P solving the Lyapunov equation

$$A^TP+PA=-Q,\quad Q=Q^T>0 \tag{20}$$

Consider the quadratic Lyapunov function

$$V_s(\tilde{S})=\tilde{S}^TP\tilde{S} \tag{21}$$

Taking time derivative along trajectories gives

$$\dot{V}_S=\tilde{S}^T(A^TP+PA)\tilde{S}+2\tilde{S}^TPE_S(t)=-\tilde{S}^TQ\tilde{S}+2\tilde{S}^TPE_S(t) \tag{22}$$

Applying Young's inequality yields, for any $\varepsilon > 0$,

$$2\tilde{S}^T P E_S(t) \leq \lambda_{max}(P)(\varepsilon\|\tilde{S}\|^2 + \varepsilon^{-1}\|E_S(t)\|^2) \tag{23}$$

Choosing $\varepsilon > 0$ sufficiently small we obtain constants $\alpha > 0$ and $c > 0$ such that

$$\dot{V}_S \leq -\alpha\|\tilde{S}\|^2 + c\|E_S(t)\|^2 \tag{24}$$

By the triggering design (9), the sampling perturbation satisfies $\|E_S(t)\| \to 0$ as $t \to \infty$. Hence the comparison principle implies $V_S(t) \to 0$ and therefore $\tilde{S}(t) \to 0$.

Thus $\check{S}_i(t) \to S$ for every follower. □

Lemma 2. *Let the multi-agent system (1) and (2) satisfying Assumption 1. Assume the reference generator and estimator dynamics are implemented as in the text and that the closed-loop signals. are piecewise continuous and uniformly bounded on* $[0, \infty)$.

Consider the state-triggering error

$$\hat{e}_{z_i}(t) = \hat{z}_i(t_k^i) - z_i(t), \quad t \in [t_k^i, t_{k+1}^i] \tag{25}$$

and the internal dynamic variable

$$\dot{\hat{\sigma}}_i(t) = -\lambda_i\hat{\sigma}_i(t) + \|\hat{e}_{z_i}\|, \quad \lambda_i > 0 \tag{26}$$

Define the state-triggering threshold

$$\sigma_i(t) = v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t), \quad v_{1i}, \alpha_{1i}, \gamma_{1i} > 0 \tag{27}$$

and assume the matrix-triggering thresholds are chosen so that the corresponding sampling errors are uniformly bounded and vanish asymptotically. If the nominal generator matrix A_ξ *is Hurwitz, then each follower's generated reference* $z_i(t)$ *asymptotically converges to the leader's state* $x_0(t)$.

$$\lim_{t\to\infty} \|z_i(t) - x_0(t)\| = 0, \quad i = 1, \ldots, n \tag{28}$$

Proof. Let $\xi_i(t) = z_i(t) - x_0(t)$ and stack $\xi(t) = \left[\xi_1^T, \ldots, \xi_n^T\right]^T$. Using (4) and the leader dynamics we obtain the compact perturbed linear form

$$\dot{\xi}(t) = A_\xi\xi(t) + \Delta(t), \tag{29}$$

where $A_\xi = I_n \otimes S - \varsigma_1(\mathcal{H} \otimes I_q)$ and $\Delta(t)$ collects perturbation terms due to estimator errors and sampling errors.

By assumption the closed-loop quantities satisfy uniform bounds

$$\|z_i(t)\| \leq Z_{max}, \quad \|\check{S}_j(t)\| \leq S_{max}, \quad \forall t \geq 0, \forall i, j \tag{30}$$

hence the sampling errors $\hat{e}_{z_i}(t)$ and matrix-sampling errors are bounded on every finite interval. From the triggering design we have for the state channel

$$\|\hat{e}_{z_i}(t)\| \leq v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t) \tag{31}$$

Solving the linear ODE for $\hat{\sigma}_i(t)$ by variation of constants gives

$$\hat{\sigma}_i(t) = \hat{\sigma}_i(0)e^{-\lambda_i t} + \int_0^t e^{-\lambda_i(t-s)}\|\hat{e}_{z_i}\|ds \tag{32}$$

which implies $\hat{\sigma}_i(t)$ is uniformly bounded on any finite interval. If $\|\hat{e}_{z_i}\| \to 0$ then $\hat{\sigma}_i(t) \to 0$; combined with the exponential term we obtain $\sigma_i(t) \to 0$ as $t \to \infty$. Consequently each local perturbation component entering $\Delta(t)$ can be bounded by terms proportional to $\|\check{S}(t)\|$, $\sigma_i(t)$ and the estimator errors. In particular there exist positive constants C_1, C_2, C_3 and aggregate functions $\Sigma_1(t), \Sigma_2(t)$ constructed from $\sigma_i(t)$ and matrix thresholds such that

$$\|\Delta_i(t)\| \leq C_1\|\check{S}_i(t)\| + C_2\Sigma_1(t) + C_3\Sigma_2(t) \tag{33}$$

and by the previous discussion the right-hand side tends to zero as $t \to \infty$.

Since by design A_ξ is Hurwitz, there exists $P_\xi = P_\xi^T > 0$ solving

$$A_\xi^T P_\xi + P_\xi A_\xi = -Q_\xi, \quad Q_\xi = Q_\xi^T > 0 \tag{34}$$

Consider the quadratic function $V_\xi(\xi) = \xi^T P_\xi \xi$. Its derivative along trajectories satisfies

$$\dot{V}_\xi = -\xi^T Q_\xi \xi + 2\xi^T P_\xi \Delta(t) \tag{35}$$

Using Young's inequality with a small $\varepsilon > 0$ we obtain constants $\varkappa > 0$ and $c_\xi > 0$ such that

$$\dot{V}_\xi \leq -\varkappa\|\xi\|^2 + c_\xi\|\Delta(t)\|^2 \tag{36}$$

Hence, with $\mu = \varkappa/\lambda_{max}(P_\xi) > 0$,

$$\dot{V}_\xi(t) \leq V_\xi(0)e^{-\mu t} + c_\xi \int_0^t e^{-\mu(t-s)}\|\Delta(s)\|^2 ds \tag{37}$$

As $\|\Delta(t)\| \to 0$ and $\|\Delta(t)\|$ is bounded, the convolution integral tends to zero, hence $V_\xi \to 0$ and therefore $\overline{\xi} \to 0$. This implies $z_i(t) \to x_0(t)$ or all i. which completes the proof. □

Lemma 3. *For each follower i, consider the event-triggering rule (9) with the internal dynamics*

$$\dot{\hat{\sigma}}_i(t) = -\lambda_i\hat{\sigma}_i(t) + \|\hat{e}_{z_i}\|, \quad \lambda_i > 0 \tag{38}$$

and the threshold

$$\sigma_i(t) = v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t), \quad v_{1i}, \alpha_{1i}, \gamma_{1i} > 0 \tag{39}$$

Assume all closed-loop signals are piecewise continuous and uniformly bounded on any finite interval, and that the matrix-triggered channel yields sampling errors that are bounded on finite intervals. Then there exists a constant $\varepsilon_i > 0$ such that every inter event time satisfies

$$t_{k+1} - t_k \geq \varepsilon_i > 0 \tag{40}$$

for some constant ε_i and hence Zeno behavior is excluded.

Proof. Fix any follower i and any triggering instant t_k^i. Immediately after the broadcast at t_k^i the sampled error is reset so that $\hat{e}_{z_i}(t_k^i) = 0$. On the subsequent interval the error evolves continuously and, by the boundedness assumption on closed-loop signals, there exists $M_i > 0$ such that

$$\|\dot{\hat{e}}_{z_i}(t)\| \leq M_i, \quad t \in \left[t_k^i, t_k^i + \tau\right] \tag{41}$$

for some small $\tau > 0$. Hence, for $t \in \left[t_k^i, t_k^i + \tau\right]$

$$\|\hat{e}_{z_i}(t)\| \leq M_i(t - t_k^i) \tag{42}$$

Next we show the threshold $\sigma_i(t)$ admits a positive lower bound on any fixed finite interval. From the linear dynamics of $\hat{\sigma}_i$ and boundedness of $\|\hat{e}_{z_i}(\cdot)\|$ on $\left[t_k^i, t_k^i + T\right)$ it follows that $\hat{\sigma}_i(t)$ is bounded and satisfies

$$\hat{\sigma}_i(t) \geq \hat{\sigma}_i(0)e^{-\lambda_i t} > 0, \quad t \in [0, T) \tag{43}$$

Therefore there exists $\underline{\sigma}_i > 0$ such that

$$\dot{\hat{\sigma}}_i(t) = v_{1i}e^{-\alpha_{1i}t} + \gamma_{1i}\hat{\sigma}_i(t) \geq \underline{\sigma}_i > 0, \quad t \in [t_k^i, t_k^i + T] \tag{44}$$

Therefore, an event cannot occur before the time τ_i solving $M_i\tau_i = \underline{\sigma}_i/M_i > 0$. Hence the inter event time satisfies

$$t_{k+1}^i - t_k^i \geq \tau_i = \frac{\underline{\sigma}_i}{M_i} > 0 \tag{45}$$

Since k was arbitrary, a uniform positive lower bound on inter event times exists and Zeno behavior is excluded. □

Theorem 1. *Under Assumptions 1–4, the control law (4), the distributed estimator (5), and the event-triggered mechanism (9) together guarantee that the tracking error*

$$e_i(t) = y_i(t) - y_0(t) \tag{46}$$

converges to zero for each follower.

$$\lim_{t\to\infty} \|e_i(t)\| = 0, \quad i = 1, \ldots, n \tag{47}$$

Proof. Define the local tracking state error

$$d_i(t) = x_i(t) - z_i(t) \tag{48}$$

and aggregate into

$$d(t) = \left[d_1^T(t), \ldots, d_n^T(t)\right]^T \tag{49}$$

Using the closed-loop control law (4) and system dynamics (2), the stacked tracking dynamics can be written as the perturbed linear system

$$\dot{d}(t) = A_d d(t) + \Theta(t), \quad A_d = I_n \otimes S - blkdiag(B_i K_i) \tag{50}$$

where $\Theta(t)$ collects all perturbation terms originating from the estimator error $\tilde{S}_i(t)$, reference generation errors $\xi_i(t) = z_i(t) - x_0(t)$, and sampling-induced errors in both reference and estimator exchanges.

By Assumption 4 the matrix A_d is Hurwitz. We provide an explicit bound for $\Theta(t)$. From the construction of the control law and the linearity of the perturbation contributions, each local perturbation admits the decomposition

$$\|\Theta_i(t)\| \leq D_{1i}\|\tilde{S}_i(t)\| + D_{2i}\|\xi_i(t)\| + D_{3i}\sum_{j\in\mathcal{N}_i}\|e_{z_j}(t)\| + D_{4i}\sum_{j\in\mathcal{N}_i}\|e_{S_j}(t)\| \tag{51}$$

for nonnegative constants $D_{1i}, \ldots, D_{4i}$ depending only on the agent dynamics and network weights. Stacking these bounds yields the global inequality

$$\|\Theta(t)\| \leq D_1\|\tilde{S}(t)\|_\infty + D_2\|\xi(t)\|_\infty + D_3E_z(t) + D_4E_S(t) \tag{52}$$

where $D_1, \ldots, D_4 > 0$ are network-dependent constants, $\|\cdot\|_\infty$ denotes the maximum over agents, and $E_z(t) = max_j\|e_{z_j}(t)\|$, $E_S(t) = max_j\|e_{S_j}(t)\|$.

By Lemmas 1 and 2 we have $\|\tilde{S}(t)\|_\infty \to 0$ and $\|\xi(t)\|_\infty \to 0$. Moreover, the triggering thresholds guarantee $E_z(t) \to 0$ and $E_S(t) \to 0$ as $t \to \infty$. Hence

$$\|\Theta(t)\| \to 0, \quad t \to \infty \tag{53}$$

Since A_d is Hurwitz, choose $P_d > 0$ solving

$$A_d^T P_d + P_d A_d = -Q_d, \quad Q_d = Q_d^T > 0 \tag{54}$$

Consider the Lyapunov function $V_d(d) = d^T P_d d$. Its derivative along trajectories satisfies

$$\dot{V}_d = -d^T Q_d d + 2d^T P_d \Theta(t) \tag{55}$$

Using Young's inequality as before we obtain constants $\varkappa > 0$ and $c_v > 0$ such that

$$\dot{V}_d \leq -\varkappa\|d\|^2 + c_d\|\Theta(t)\|^2 \tag{56}$$

Equivalently, with $\lambda_{max}(P_d)$ denoting the largest eigenvalue of P_d, there exists $\mu > 0$ such that

$$\dot{V}_d(t) \leq -\mu V_d(t) + c_d\|\Theta(t)\|^2 \tag{57}$$

Applying the variation of constants formula yields

$$V_d(t) \leq V_d(0)e^{-\mu t} + c_d \int_0^t e^{-\mu(t-s)}\|\Theta(s)\|^2 ds \tag{58}$$

Since $\Theta(t) \to 0$ and is bounded, the convolution integral tends to zero as $t \to \infty$. Therefore, $V_d(t) \to 0$ and consequently $d(t) \to 0$. According to the output regulation theory for heterogeneous systems, the reference z_i is constructed such that $C_i z_i = C_0 x_0$ holds. Thus, the output tracking error can be decomposed as $e_i = C_i x_i - C_0 x_0 = C_i(x_i - z_i) + (C_i z_i - C_0 x_0) = C_i d_i$. Finally, since $d_i(t) \to 0$, we obtain

$$\lim_{t\to\infty} \|e_i(t)\| = 0, \quad i = 1, \ldots, n \tag{59}$$

This completes the proof. □

3. Simulation Results

3.1. Simulation Initialization

The communication structure of the leader–follower network is illustrated in Figure 4.

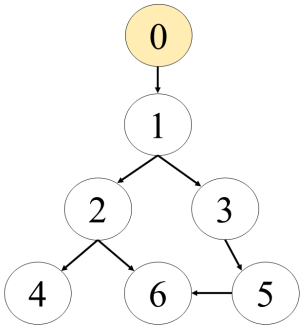


Figure 4. The communication topology for a group of agents with six followers and one leader.

The dynamics of the follower agents are described by

$$\dot{x}_1 = \begin{bmatrix} 8 & -\cos(2t^2) \\ 3x_{1,1} & 3 \end{bmatrix} x_1 + u_1$$
$$\dot{x}_2 = \begin{bmatrix} 0.3 & 0 \\ 0 & -0.2 \end{bmatrix} x_2 + \begin{bmatrix} 1.8 & 0 \\ 0 & 1.6 \end{bmatrix} u_2$$
$$\dot{x}_3 = u_3$$
$$\dot{x}_4 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} x_4 + \begin{bmatrix} 1.5 & 0.1 \\ 0 & 1.2 \end{bmatrix} u_4$$
$$\dot{x}_5 = \begin{bmatrix} 0.2 & -0.1 \\ 0.1 & 0.2 \end{bmatrix} x_5 + \begin{bmatrix} 0.9 & 0 \\ 0 & 0.9 \end{bmatrix} u_5$$
$$\dot{x}_6 = \begin{bmatrix} x_{6,1}^2 - 2x_{6,2} + \sin(t) \\ (1+t)x_{6,2} \end{bmatrix} + \begin{bmatrix} 1.4 & 0 \\ 0 & 1.4 \end{bmatrix} u_6$$

where $x_{i,1}$ of x_i for $i = 1, 2, 3, 4$. The leader dynamics follow the system in (1) with the system matrix $S = \begin{bmatrix} 1 & -3 \\ 1 & -1 \end{bmatrix}$. $C_0 = C_1 = C_2 = C_3 = C_4 = C_5 = C_6 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T$. The initial state of the leader is randomly sampled from the interval $[-40, 40]$ for each state component. Similarly, the initial states of all follower agents $x_i(0)$, $i = 1, 2, 3, 4, 5, 6$, are independently sampled from the same interval. This choice introduces heterogeneous initial conditions and enables evaluation of convergence from dispersed configurations. The event-triggered mechanism is implemented according to the state-triggered and matrix-triggered conditions in (9), which are specified as follows

$$t_{k+1}^i = \inf\left\{ t > t_k^i \mid \|\hat{e}_{z_i}(t)\| \geq 1.2e^{-0.7t} + 0.8\hat{\sigma}_i(t) \right\}$$

$$s_{k+1}^i = \inf\left\{ t > s_k^i \mid \|\hat{e}_{S_i}(t)\| \geq e^{-5t} + 0.5e^{-t}\|\check{S}_i(t)\| \right\}$$

For each follower agent i, the internal variable evolves as $\dot{\hat{\sigma}}_i(t) = -1.2\hat{\sigma}_i(t) + \|\hat{e}_{z_i}(t)\|$, with $\hat{\sigma}_i(0) = 0$. In addition, the design parameters are chosen as $\varsigma_1 = 3$ and $\varsigma_2 = 4$. This threshold design satisfies the regularity and positivity conditions stated in Remark 9, thereby guaranteeing strictly positive thresholds over any finite time interval. Furthermore, since all triggering conditions depend solely on locally available quantities, the proposed implementation is fully consistent with the distributed structure emphasized in Remark 10.

Table 1 summarizes the primary hyperparameter settings of the reinforcement learning algorithm utilized in the proposed method. A MAPPO-based actor–critic framework is adopted, featuring a moderate discount factor and conservative learning rates to guarantee stable policy updates across the multi-agent network during training. The episode length and the total number of training episodes are deliberately configured to allow for sufficient state-space exploration while preserving computational efficiency. Furthermore, identical parameter settings are uniformly applied to all follower agents throughout the training phase to ensure algorithmic fairness and experimental reproducibility.

Figure 5 illustrates the MAPPO training convergence across five independent seeds. The solid line and shaded region represent the mean cumulative reward and standard deviation, respectively. Following an initial high-variance exploration phase, the reward rapidly converges and stabilizes with negligible variance, demonstrating the robustness and reproducibility of the proposed learning scheme.

Table 1. Reinforcement learning parameter settings used in the proposed method.

Parameter	Value
RL algorithm	MAPPO
Discount factor	0.95
Actor learning rate	1×10^{-4}
Critic learning rate	1×10^{-4}
Episode length	1000 steps
Training episodes	300

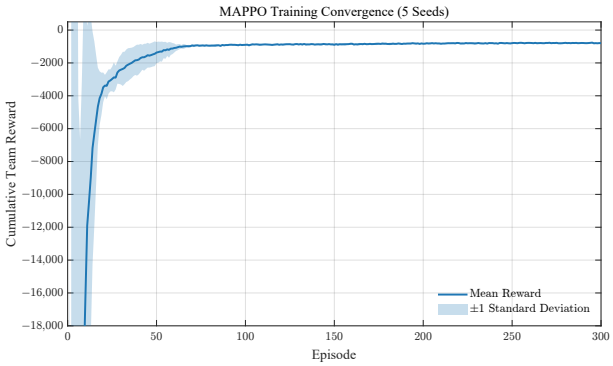


Figure 5. MAPPO training convergence.

3.2. Simulation Results and Performance Evaluation

Numerical simulations are conducted to evaluate the proposed data-driven consensus strategy. The test system is a directed heterogeneous multi-agent network. It consists of one leader and six followers. The leader follows the autonomous linear dynamics defined in (1). Meanwhile, each follower obeys the distinct nonlinear dynamics described in (2). In this framework, each follower utilizes continuous local measurements. It also intermittently receives state information from its neighbors. This information exchange is strictly governed by the distributed event-triggered protocol defined in (9). During the execution phase, each agent relies on a dedicated actor network. This network dynamically outputs the adaptive feedback gain matrix K_i. The local tracking controller in (4) then uses K_i to compute the continuous control input $u_i(t)$. During the training phase, a shared critic network is employed. It coordinates the multi-agent policy updates under the CTDE framework, as previously shown in Figure 3. To ensure a rigorous evaluation, a fixed-gain controller serves as the baseline. Unlike the proposed RL method, which learns online without prior knowledge of the leader's dynamics S, the baseline gains were computed offline using global model knowledge. Specifically, they were initialized via LQR and empirically fine-tuned, providing a proper static benchmark for evaluating the RL method's dynamic adaptability.

Figure 6 illustrates the tracking performance of the multi-agent system under different gain configuration strategies. As shown in Figure 6a, when the RL based tuning mechanism is applied, the states of all follower agents align with the leader's trajectory smoothly. The convergence is achieved within approximately 3 s, exhibiting a highly efficient transient adjustment. In contrast, Figure 6b displays the system response under the fixed gain controller. While the baseline method successfully ensures the final asymptotic output consensus, the trajectories of certain agents experience larger initial deviations. Consequently, the system requires a relatively longer settling period to fully synchronize with the leader.

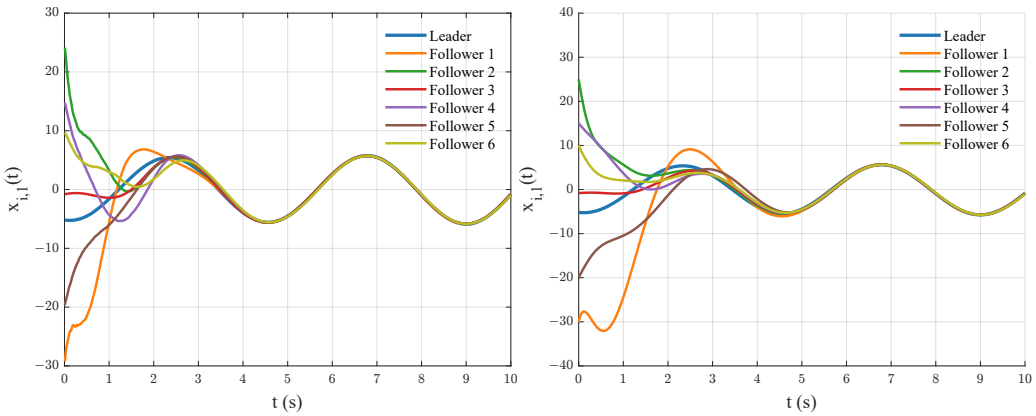


Figure 6. Output trajectory comparison. (**a**) Output trajectories under the RL based controller. (**b**) Output trajectories under the fixed gain controller.

Figure 7 illustrates the evolution of the output tracking errors for all follower agents. As shown in Figure 7a, under the guidance of the RL based tuning mechanism, the tracking errors decay continuously and rapidly approach zero within approximately 4 s. The error curves exhibit a smooth downward trend, reflecting a highly efficient and stable initial synchronization process. In contrast, Figure 7b depicts the tracking errors when the fixed gain controller is employed. While the baseline method successfully drives all errors to zero eventually, the decay process is relatively prolonged.

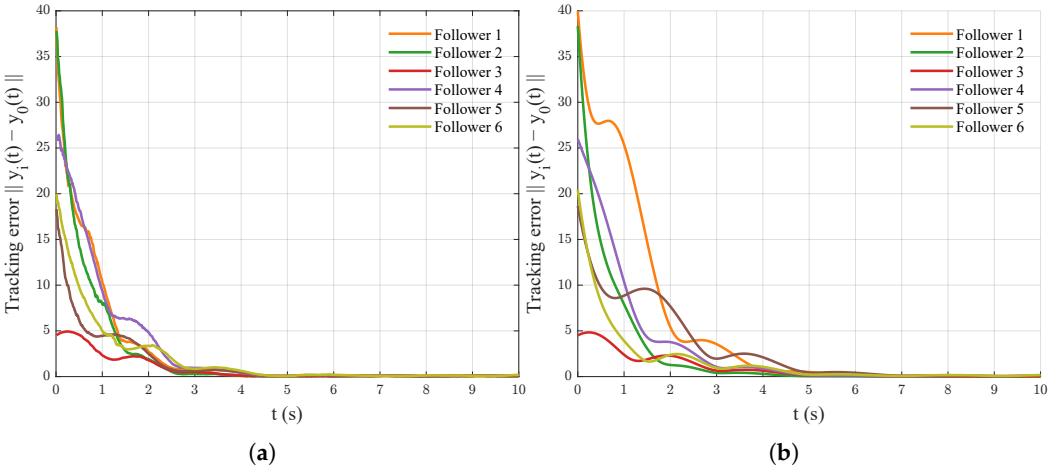


Figure 7. Tracking error comparison. (**a**) Tracking errors under the RL based controller. (**b**) Tracking errors under the fixed gain controller.

As illustrated in Figure 8, the event-triggering patterns of the RL-based and fixed-gain controllers are largely comparable. While the RL controller exhibits a slightly higher triggering frequency during the initial transient phase in Figure 8a to facilitate rapid adaptation, both methods maintain similar and highly efficient communication frequencies in the steady state. Overall, the RL approach provides enhanced transient adaptability without increasing the long-term communication burden.

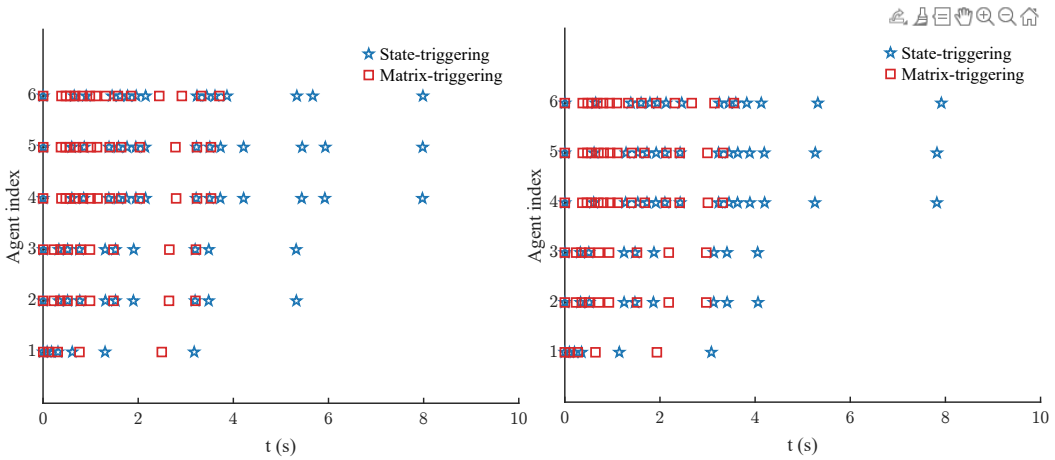


Figure 8. Triggering events comparison. (**a**) Triggering events under the RL based controller. (**b**) Triggering events under the fixed gain controller.

To evaluate the system's robustness, 20 independent trials were conducted. The leader and followers were initialized using zero-mean normal distributions with standard deviations of 40 and 5, respectively. As summarized in Table 2, the system consistently achieves output consensus despite high initial spatial randomness. The convergence time averages 5.87 ± 0.96 s, demonstrating strong stability against large initial deviations. Furthermore, by employing a data-bundling broadcasting strategy, the total triggered events averaged 671.10 ± 232.24. This yields an 88.81% communication savings rate compared to traditional periodic sampling.

Table 2. Statistical performance of the proposed method under random initial conditions.

Metric	Mean Value	Standard Deviation
Convergence Time (s)	5.87	0.96
Total Triggered Events	671.10	232.24
Communication Savings (%)	88.81	-

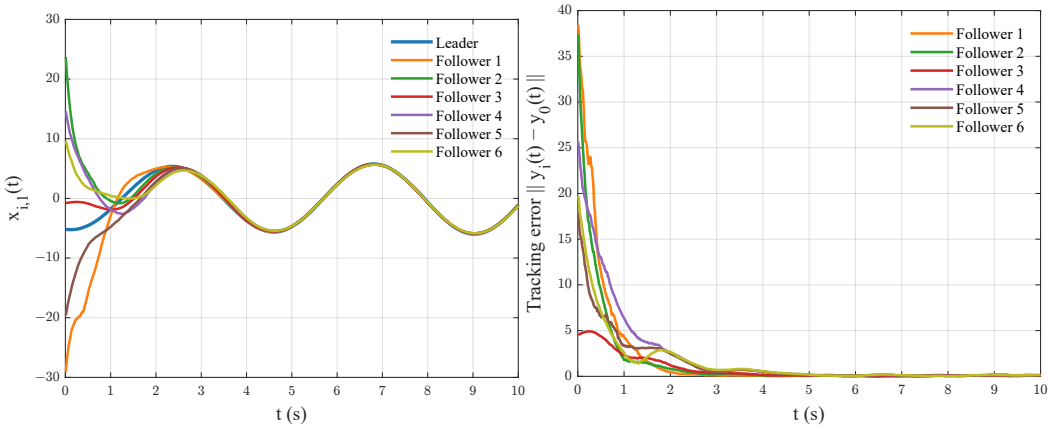
Overall, the comparative results indicate similar steady-state communication frequencies for both controllers. However, the RL-based approach achieves a significantly faster and smoother transient response. This improvement must be weighed against the added computational complexity of MARL. The proposed CTDE architecture effectively resolves this trade-off. The computationally intensive training phase is executed entirely offline. During real-time operation, each agent simply evaluates a lightweight feedforward Actor network. This adds negligible computational burden to the local controller. In practical physical deployments, such rapid initial synchronization is critical. It prevents actuator saturation and mitigates collision risks among neighboring agents.

3.3. Robustness Evaluation Under Process Noise

To evaluate the robustness of the proposed event-triggered adaptive consensus scheme, small process noise is introduced into the follower dynamics. The follower agents and leader are initialized at fixed positions. This experiment investigates whether the learned RL policy and the event-triggered mechanism can maintain stable closed-loop behavior in the presence of disturbances.

As shown in Figure 9, the injected disturbance causes minor fluctuations in the initial stage. However, the follower trajectories remain stable and quickly realign with the leader

trajectory. The tracking errors remain bounded and converge to a small neighborhood around zero, suggesting that the proposed scheme demonstrates robustness in the presence of disturbances.



(**a**) (**b**)

Figure 9. System performance under process noise. (**a**) Output trajectories of the leader and followers. (**b**) Tracking errors with respect to the leader.

Figure 10 illustrates the triggering instants under process noise. Compared with the noise free case, the presence of small process noise slightly increases the number of triggering events, reflecting minor deviations in the follower trajectories. However, all inter event times remain strictly positive, confirming that the event-triggered mechanism operates in a Zeno-free.

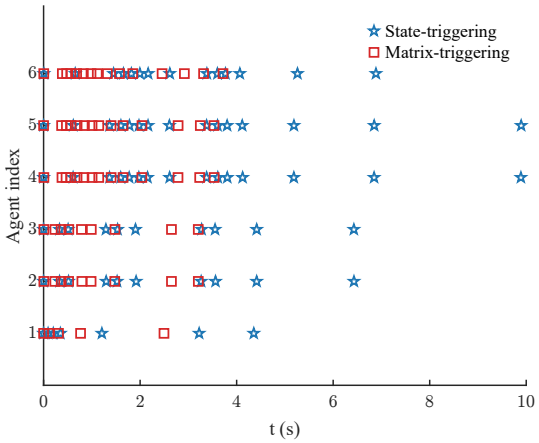


Figure 10. Triggering instants under process noise.

Overall, these results suggest that the proposed strategy effectively maintains high communication efficiency while ensuring robust tracking performance in the presence of continuous disturbances and highly uncertain initial conditions.

4. Discussion

The proposed framework demonstrates significant advantages in transient performance and communication efficiency. However, several limitations warrant further analysis. First, the system performance depends on empirically tuned rewards and thresholds. This dependence may restrict its broader applicability. Second, this study focuses on small-scale networks. Its performance in larger or more dynamic systems remains to be investigated.

Furthermore, while the current reward based $r_{stability}$ is computationally lightweight, it acts only as a soft constraint. For strict aerospace applications, this might lead to high-

frequency control chatter. Implementing Lyapunov-based constraints in the actor's output layer could eliminate such chatter, though at the cost of increased computational burden. Finally, the framework assumes ideal control inputs. Practical deployments face actuator saturation, rate limits, and delays. The current reward function $r_{control}$ suppresses excessive energy, but it cannot mathematically prevent saturation-induced instability.

It should also be noted that this study does not claim absolute superiority over advanced analytical methods. Approaches like Robust H_∞ Control or Model Reference Adaptive Control (MRAC) provide rigorous mathematical guarantees. However, implementing distributed H_∞ control often requires solving complex coupled Linear Matrix Inequalities. Similarly, MRAC generally necessitates strict structural matching conditions. The proposed RL-based approach offers a flexible data-driven alternative. It bypasses heavy algebraic computations and structural restrictions. This provides tuning flexibility at the cost of hard analytical optimality.

5. Conclusions

This paper presents an adaptive control framework with event-triggered mechanisms for heterogeneous multi-agent systems. Simulation results confirm that followers can track the leader's trajectory with bounded errors, even under disturbances. The integration of reinforcement learning effectively optimizes local feedback gains and improves transient performance. Meanwhile, the distributed internal model updates successfully alleviate structural heterogeneity. The event-triggered mechanism significantly reduces communication frequency without inducing Zeno behavior. Overall, the proposed framework provides a practical solution for robust and communication-efficient coordination.

Future work will focus on several key areas. We intend to explore dynamic threshold adjustment in the ETM to achieve a fully autonomous tuning process. Extending the framework to larger, more complex networks is also a priority. Furthermore, current noise evaluations remain preliminary. Defining strict robustness limits requires comprehensive statistical testing, such as Monte Carlo analyses across varying Signal-to-Noise Ratios (SNR). Finally, incorporating hard non-smooth actuator constraints and anti-windup mechanisms is essential for future physical implementations. Although there are areas for improvement, the results suggest that the proposed framework holds potential for practical multi-agent coordination.

Author Contributions: Conceptualization, M.Q. and S.S.; Methodology, B.L. and M.Q.; Software, M.Q.; Validation, M.Q.; Formal analysis, M.Q.; Investigation, M.Q., S.L. and Y.W.; Resources, M.Q.; Data curation, M.Q. and Y.W.; Writing—original draft preparation, M.Q.; Writing—review and editing, M.Q. and S.L.; Visualization, M.Q., S.L. and B.L.; Supervision, S.S. and S.L.; Project administration, S.S.; Funding acquisition, S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Science and Technology Program, grant numbers AB24010110 and AB25069501.

Data Availability Statement: The data supporting the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ota, J. Multi-agent robot systems as distributed autonomous systems. *Adv. Eng. Inform.* **2006**, *20*, 59–70. [CrossRef]
2. Fuentes-Fernández, R.; Guijarro, M.; Pajares, G. A multi-agent system architecture for sensor networks. *Sensors* **2009**, *9*, 10244–10269. [CrossRef]
3. Pinnarelli, A.; Vizza, P.; Brusco, G.; Menniti, D.; Sorrentino, N.; Shagar, G.V.; Soleimani, A. Multi-agent approach for synthetic inertia and fast reserve service provided by a distributed energy storage system. *Results Eng.* **2025**, *27*, 105474. [CrossRef]

4. Owoputi, R.; Ray, S. Security of multi-agent cyber-physical systems: A survey. *IEEE Access* **2022**, *10*, 121465–121479. [CrossRef]
5. Ni, J.; Wen, C.; Zhao, Y. Fixed-time leader-follower quantized output consensus of high-order multi-agent systems over digraph. *Inf. Sci.* **2022**, *587*, 408–434. [CrossRef]
6. Mousavi, A.; Markazi, A.H. A new control method for leader–follower consensus problem of uncertain constrained nonlinear multi-agent systems. *J. Frankl. Inst.* **2024**, *361*, 106889. [CrossRef]
7. Wen, F.; Chen, Y.; Pan, C. Cooperative tracking control of cross-domain heterogeneous unmanned systems via adaptive neural backstepping. In *Proceedings of the 2024 7th International Symposium on Autonomous Systems (ISAS)*; IEEE: New York, NY, USA , 2024; pp. 1–6. [CrossRef]
8. Zuo, S.; Song, Y.; Lewis, F.L.; Davoudi, A. Output containment control of linear heterogeneous multi-agent systems using internal model principle. *IEEE Trans. Cybern.* **2017**, *47*, 2099–2109. [CrossRef]
9. Gao, W.; Mynuddin, M.; Wunsch, D.C.; Jiang, Z.P. Reinforcement learning-based cooperative optimal output regulation via distributed adaptive internal model. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 5229–5240. [CrossRef]
10. Huang, J. The cooperative output regulation problem of discrete-time linear multi-agent systems by the adaptive distributed observer. *IEEE Trans. Autom. Control* **2016**, *62*, 1979–1984. [CrossRef]
11. Wang, B.; Shen, C. A time-varying observer-based approach to equilibrium estimation and compensation for synchronization of heterogeneous nonlinear cyber-physical systems. *IEEE Trans. Autom. Control* **2023**, *68*, 6176–6183. [CrossRef]
12. Cao, C.; Wei, C.; Liao, Y.; Zhang, Y.; Li, J. On novel trajectory tracking control of quadrotor UAV: A finite-time guaranteed performance approach. *J. Frankl. Inst.* **2022**, *359*, 8454–8483. [CrossRef]
13. Dimarogonas, D.V.; Frazzoli, E.; Johansson, K.H. Distributed event-triggered control for multi-agent systems. *IEEE Trans. Autom. Control* **2011**, *57*, 1291–1297. [CrossRef]
14. Yi, X.; Liu, K.; Dimarogonas, D.V.; Johansson, K.H. Dynamic event-triggered and self-triggered control for multi-agent systems. *IEEE Trans. Autom. Control* **2018**, *64*, 3300–3307. [CrossRef]
15. Yi, X.; Liu, K.; Dimarogonas, D.V.; Johansson, K.H. Distributed dynamic event-triggered control for multi-agent systems. In *Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC)*; IEEE: New York, NY, USA, 2017 ; pp. 6683–6698. [CrossRef]
16. Du, Z.; Chen, C.; Li, C.; Yang, X.; Li, J. Fault-tolerant H-infinity stabilization for networked cascade control systems with novel adaptive event-triggered mechanism. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 22597–22608. [CrossRef]
17. Xue, M.; Ye, M.; Dai, M.Z.; Liao, Y.; Cao, C. A general prescribed performance trajectory tracking control policy for quadrotor UAVs under event-triggered sampling. *Int. J. Robust Nonlinear Control* **2026**, *36*, 525–547. [CrossRef]
18. Yao, Y.; Luo, Y.; Cao, J.; Li, A. Sample-based event-triggered consensus control for cooperative-competitive leader-follower linear multi-agent systems based on disturbance observers: State and output feedback design. *Inf. Sci.* **2025**, *694*, 121701. [CrossRef]
19. Yang, R.; Zhang, H.; Feng, G.; Yan, H. Distributed event-triggered adaptive control for cooperative output regulation of heterogeneous multiagent systems under switching topology. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 4347–4358. [CrossRef]
20. Wang, Y.; Zhu, F. Distributed dynamic event-triggered control for multi-agent systems with quantization communication. *IEEE Trans. Circuits Syst. II Express Briefs* **2023**, *71*, 2054–2058. [CrossRef]
21. Farooq, A.; Iqbal, K. A survey of reinforcement learning for optimization in automation. In *Proceedings of the 2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*; IEEE: New York, NY, USA, 2024; pp. 2487–2494. [CrossRef]
22. Fard, N.E.; Selmic, R. Consensus of multi-agent reinforcement learning systems: The effect of immediate rewards. *J. Robot. Control* **2022**, *3*, 115–127. [CrossRef]
23. Zhang, K.; Sun, T.; Tao, Y.; Genc, S.; Mallya, S.; Basar, T. Robust multi-agent reinforcement learning with model uncertainty. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 10571–10583.
24. Long, M.; Su, H.; Zeng, Z. Model-free event-triggered consensus algorithm for multiagent systems using reinforcement learning method. *IEEE Trans. Syst. Man, Cybern. Syst.* **2021**, *52*, 5212–5221. [CrossRef]
25. Zhang, C.H.; Yang, G.H. Event-triggered global finite-time control for a class of uncertain nonlinear systems. *IEEE Trans. Autom. Control* **2019**, *65*, 1340–1347. [CrossRef]
26. Cao, L.; Pan, Y.; Liang, H.; Huang, T. Observer-based dynamic event-triggered control for multiagent systems with time-varying delay. *IEEE Trans. Cybern.* **2022**, *53*, 3376–3387. [CrossRef]
27. Francis, B.; Wonham, W. The internal model principle of control theory. *Automatica* **1976**, *12*, 457–465. [CrossRef]
28. Qian, Y.Y.; Liu, L.; Feng, G. Distributed event-triggered adaptive control for consensus of linear multi-agent systems with external disturbances. *IEEE Trans. Cybern.* **2018**, *50*, 2197–2208. [CrossRef]
29. Wang, Y.; Zhu, F. Distributed hybrid dynamic event-triggered consensus control for nonlinear multi-agent systems. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 10470–10483. [CrossRef]
30. Yuan, Z.; Yao, C.; Liu, X.; Gao, Z.; Zhang, W. Multiagent formation control and dynamic obstacle avoidance based on deep reinforcement learning. *IEEE Trans. Ind. Inform.* **2025**, *21*, 4672–4682. [CrossRef]

31. Su, Y.; Huang, J. Cooperative output regulation of linear multi-agent systems. *IEEE Trans. Autom. Control* **2011**, *57*, 1062–1066. [CrossRef]
32. Zuo, S.; Song, Y.; Lewis, F.L.; Davoudi, A. Adaptive output containment control of heterogeneous multi-agent systems with unknown leaders. *Automatica* **2018**, *92*, 235–239. [CrossRef]
33. Haghshenas, H.; Badamchizadeh, M.A.; Baradarannia, M. Containment control of heterogeneous linear multi-agent systems. *Automatica* **2015**, *54*, 210–216. [CrossRef]
34. Cheng, J.; Liang, L.; Park, J.H.; Yan, H.; Li, K. A dynamic event-triggered approach to state estimation for switched memristive neural networks with nonhomogeneous sojourn probabilities. *IEEE Trans. Circuits Syst. Regul. Pap.* **2021**, *68*, 4924–4934. [CrossRef]
35. Girard, A. Dynamic triggering mechanisms for event-triggered control. *IEEE Trans. Autom. Control* **2014**, *60*, 1992–1997. [CrossRef]
36. Zhu, G.; Liu, K.; Gu, H.; Luo, W.; Lu, J. Observer-based event-triggered formation control of multi-agent systems with switching directed topologies. *IEEE Trans. Circuits Syst. Regul. Pap.* **2021**, *69*, 1323–1332. [CrossRef]
37. Hu, W.; Liu, L.; Feng, G. Output consensus of heterogeneous linear multi-agent systems by distributed event-triggered/self-triggered strategy. *IEEE Trans. Cybern.* **2016**, *47*, 1914–1924. [CrossRef]
38. Rajnikant, P.N.; Khanna, R. The Hamilton-jacobi-bellman equation for optimal control in multi-agent systems. *Int. J. Math. Trends Technol.* **2015**, *71*, 9–17. [CrossRef]
39. Mei, D.; Sun, J.; Xu, Y.; Dou, L. Model-free learning-based distributed cooperative tracking control of human-in-the-loop multi-agent systems. *Int. J. Robust Nonlinear Control* **2024**, *34*, 7029–7044. [CrossRef]
40. Zhu, H.y.; Mao, W. Multi-agent reinforcement learning control for consensus problems of uncertain nonlinear multi-agent systems. In *Proceedings of the 2021 China Automation Congress (CAC)*; IEEE: New York, NY, USA, 2021; pp. 6561–6566. [CrossRef]
41. Zhang, H.; Jiang, H.; Luo, Y.; Xiao, G. Data-driven optimal consensus control for discrete-time multi-agent systems with unknown dynamics using reinforcement learning method. *IEEE Trans. Ind. Electron.* **2016**, *64*, 4091–4100. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.