

Article

Fully- and Partially- Distributed Adaptive Consensus of Second-Order Multi-Agent Systems Using Only Relative Position Measurements

Bahram Taran and Mohammad Reza Homaeinezhad *

Faculty of Mechanical Engineering, K. N. Toosi University of Technology, P.O. Box 19395-1999, Tehran 19991-43344, Iran; bahram_taran@yahoo.com (B.T.)

* Corresponding author. E-mail: mrhomaeinezhad@kntu.ac.ir or mrh.kntu@gmail.com (M.R.H.); Tel.: +98-21-84063284 or +98-91-21899445 (M.R.H.); Fax: +98-21-88677274 (M.R.H.)

Received: 8 August 2024; Accepted: 1 November 2024; Available online: 7 November 2024

ABSTRACT: In this paper, the distributed leader-follower consensus of a group of agents with second-order dynamics under the undirected graph communication topology is studied. The main objective of this study is to solve a major practical multi-agent problem in which the acceleration of the leader is not communicated to each follower. In contrast, the follower agents include some unknown dynamics in their intrinsic structure. By assuming a linear regression structure for leader acceleration and agent's unknown dynamics, Lyapunov-based adaptive control algorithms are devised to control the network of agents in the presence of the communication loss and modeling uncertainties. The presented study describes two multi-agent control strategies called fully-distributed adaptive control (FDAC) and partially-distributed adaptive control (PDAC) systems in the first method, the followers do not have any a priori information about the communication graph, while in the second method, some information about the eigenvalues of the communication graph is available. The mathematical manipulations required to prove the stability of the FDAC and PDAC methods are presented. Finally, illustrative simulations are conducted to render the proposed algorithms' merits and efficiencies.

Keywords: Multi-agent systems; Consensus; Leader-follower; Distributed control; Adaptive control; Output feedback; Fully-distributed adaptive control (FDAC); Partially-distributed adaptive control (PDAC)



© 2024 The authors. This is an open access article under the Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cooperative control of multi-agent systems has been studied intensively over the last two decades. This is due to the fact that multi-agent systems have broad applications in diverse areas such as unmanned air vehicles [1], flocking [2], formation control [3] and sensor networks [4]. Consensus is an important problem in the area of multi-agent systems, and it has been studied comprehensively in the literature. The studies [5,6] are well-known references for the review of the most critical problems in the area of multi-agent systems and consensus problems.

In practice, each agent might be subjected to uncertainty or unknown dynamics. Also, many existing works on the leader-follower consensus assumed that the leader input is not zero and is unknown to the followers. In leader-follower multi-agent control algorithms, the term “leader” refers to a designated agent or agents that set the reference trajectory or control commands for the other agents, known as “followers”. The leader's role typically involves leading behavior, information dissemination, coordination and stability, dynamic response and control algorithms. The leader dictates the desired path, behavior, or objectives that the followers are supposed to follow. This can include position, velocity, or orientation. Also, the leader often shares information about the environment or the desired goals, enabling followers to make informed decisions. On the other hand, the presence of a leader can provide stability to the multi-agent system by ensuring that all agents move cohesively toward a common objective, reducing the risk of inconsistencies or chaotic behavior among the followers. The leader can adapt its behavior in response to external changes (like obstacles or new objectives), guiding the followers to adapt their behavior accordingly. In some applications, the leader may implement specific control strategies to maintain a desired formation or behavior. At the

same time, followers adjust based on pre-defined rules or algorithms referencing the leader's state. Finally, in practical applications, such as robotics, autonomous vehicles, and drone swarms, the concept of a leader-follower framework enhances coordination and cooperative behavior among agents, allowing them to achieve collective tasks efficiently. Depending on the existence of the modeling uncertainties, non-zero leader input and unawareness of followers from the leader's input, the consensus problem of multi-agent systems for coping with these problems has been studied in diverse studies. The formation problem of a leader-follower multi-agent system in which the agents translate together has been studied in [7,8]. In studies [7,8], the authors assumed that only the leader has access to the reference velocity. Accordingly, they proposed the decentralized adaptive method for each follower in order to exploit the reference velocity. In study [9], the robust decentralized adaptive control approach in which the neural network is used has been proposed to solve the consensus problem of the multi-agent systems with uncertainties and external disturbances. In the study [10], the neural network has been used to estimate the unknown nonlinearity of the first-order agent dynamics. The adaptive estimation approach has been applied to determine the neural network weights in order to compensate for the unknown nonlinearity of agent dynamics. As a major consequence, it has been proven that the errors of synchronization tracking and the estimation of neural network weights are both uniformly ultimately bound. Similar to the study [10], in the study [11], the neural network approach has been applied to the high-order dynamic multi-agent system with unknown nonlinear dynamics. In [12], the authors investigated the distributed adaptive output synchronization of the networked nonlinear SISO systems, which can be transformed into the semi-strict feedback form. In [12], the authors assumed the existence of parameter uncertainty in the dynamics of each agent. Then, by application of the back-stepping method and the adaptive parameter updating recursion, the output synchronization of the network is analyzed. Authors in a study [13] worked on the first-order leader-follower consensus in which the unknown dynamics of each follower and the velocity of the leader were described by the linearly parameterized models. In the study [13], consensus stability and parameter convergence have been proven under adaptive control algorithms. Similar to [13], in [14], the authors studied the finite time leader-following consensus of the high-order multi-agent systems with unknown nonlinear dynamics. In [15], the authors studied a decentralized adaptive tracking control of a second-order leader-follower system with unknown dynamics and relative position measurements. The two introduced research articles have been cited appropriately in the introduction section. Thanks to the information extracted from these research articles, the following text has been added to the introduction section. Consensus algorithms are crucial in many real-world applications, enabling decentralized systems to achieve agreement among distributed nodes. Understanding the size of a network is vital for optimizing resource allocation and ensuring efficient communication. Consensus algorithms can facilitate network size estimation by allowing nodes to exchange information about their local views of the network. Approaches like gossip protocols enable nodes to share their knowledge iteratively, leading to a collective understanding of the total number of nodes in the system. This is particularly useful in mobile ad-hoc networks (MANETs) and Internet of Things (IoT) environments, where network topology can dynamically change. In sensor networks and distributed data systems, data aggregation is essential for reducing the communication overhead and improving data quality. Consensus algorithms help aggregate data collected from various nodes to provide global insights without requiring each node to communicate its data directly to a central authority. Techniques like averaging, voting, and weighted sums allow nodes to collaboratively compute and share summary statistics, such as the average temperature or humidity readings while maintaining data privacy and minimizing bandwidth usage. Unmanned Aerial Vehicles (UAVs) often operate in teams, requiring coordinated behaviors such as formation flying or collaborative search and rescue missions. Consensus algorithms enable UAVs to agree on actions and trajectories in real-time, ensuring that they operate cohesively despite potential communication delays or interruptions. Techniques such as consensus-based flocking algorithms allow each UAV to adjust its path based on the positions and velocities of its neighbors, facilitating smooth and efficient formation changes in dynamic environments. Consensus mechanisms are fundamental to blockchain and distributed ledger technologies, ensuring that all nodes in a network agree on the validity of transactions. Various algorithms, such as Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT), facilitate trustless environments where multiple parties can securely agree on a shared state without relying on a central authority. This application extends to cryptocurrencies, supply chain management, and transparent voting systems. In multi-agent systems, consensus algorithms enable autonomous agents to make decisions collaboratively. Applications range from robotic swarms performing collective tasks, such as exploration and mapping, to smart grid systems managing energy distribution effectively. Agents employ consensus to synchronize their actions, exchange environmental information, and adapt to changes in real-time, which enhances the system's overall efficiency and resilience. Consensus algorithms are also integral to achieving fault tolerance in distributed systems. By ensuring agreement among healthy

nodes despite the presence of faulty nodes or communication failures, these algorithms enable robust operations in systems such as cloud computing and distributed databases. Algorithms like Raft and Paxos are designed to maintain a consistent state among nodes even in the face of partial failures, thereby maintaining system reliability. Consensus algorithms are foundational to many modern technologies, facilitating decentralized networks' coordination, data integrity, and resilience. Their diverse applications across fields such as sensor networks, UAV control, and blockchain underscore their importance in shaping the future of interconnected systems [16,17]. In recent years, the development of resilient control strategies for multi-agent systems has garnered significant attention, particularly in the face of potential cybersecurity threats. One notable contribution in this area is the work by Zhang et. al, [18]. The study [18] presents an innovative adaptive neural consensus tracking control strategy, specifically designed to ensure robust coordination among agents while mitigating the effects of actuator attacks. By leveraging jointly connected topologies, the proposed approach enhances the system's resilience and effectiveness, thereby providing a framework for future research and applications in dynamic and potentially compromised environments [18].

This paper describes the solution procedure of the distributed adaptive coordination problem for second-order leader-follower multi-agent systems with unknown nonlinear dynamics under the constraint which imposes the availability of only relative position measurements and loss of communication between agents. The authors in a similar study [15] assumed that the communication between agents is available, while in this study, the aforementioned assumption is not applied during the solution of the problem.

This paper is organized as follows. Section 2 reviews some basic notation and preliminaries of graph theory and formal statement of the problem. In Section 3, the main results of this study are presented. Finally, in Section 4, simulation results are shown to illustrate the proposed methods' effectiveness. The concluding remarks associated with PDAC and FDAC methods are given in Section 5.

2. Preliminaries and Problem Formulation

Let $G = (V, E, \mathbf{A})$ is a graph of order N with $V = \{1, 2, \dots, N\}$ which represents node-set, $E \subseteq V \times V$ is the set of edges, and \mathbf{A} is the weighted adjacency matrix with nonnegative elements. An edge (i, j) denotes that the node j has access to the information of the node i . The graph G is undirected if $(i, j) \in E$ then $(j, i) \in E$ for any pair of (i, j) . In the undirected graph, two nodes i and j are neighbors if $(i, j) \in E$. Set of neighbors of node i is shown by $N_i = \{j \in V : (j, i) \in E, j \neq i\}$. A path is a sequence of connected edges in a graph. An undirected graph is connected if between any pair of distinct nodes i and j a path exists. In the context of graph theory, a self-loop refers to an edge that connects a node to itself, denoted as (i, i) for some node $i \in V$. In many applications, including networked systems, self-loops can be significant as they may represent scenarios where an agent or node processes or reflects its information. Additionally, in the definition of this paper, the expression $E \subseteq V \times V$ is defined so as to imply a generally broad perspective on the types of connections that could exist in the graph. While it is technically correct to say that E can include self-loops, it might be recognized that the presence of self-loops might not always align with the specific applications that are considered. In undirected graphs, self-loops can exist, but they do not affect the concept of adjacency between distinct nodes. They are relevant for scenarios where it is required to allow a node to have a degree of connection with itself. This can be conceptually important in many applications, such as social networks, where individuals may have a relationship with their past actions or decisions. Accordingly, the $E_{distinct} \triangleq \{(i, j) \in E | i \neq j\}$ is defined to emphasize the edges connecting distinct nodes more clearly.

In addition, for the follower agents 1 to N , there exists a leader labeled by 0. Information is exchanged between the leader and the follower agents, which belong to the neighbors of the leader. Then, the graph $\bar{G} = (\bar{V}, \bar{E}, \bar{\mathbf{A}})$ with node set $\bar{V} = V \cup \{0\}$ and edge set $\bar{E} = \bar{V} \times \bar{V}$, represents the communication topology between the leader and the followers. The communication graph between the leader and the followers is assumed to be unidirectional. A network topology is assumed to be defined by a graph where nodes represent agents and edges represent communication links. While the leader has direct links to certain followers, non-directly linked agents can still receive information through a networked relay system. Each follower agent typically communicates within its communication range with its immediate neighbors. In situations where an agent does not have a direct link to the leader, it can relay messages. This means that the follower can pass on the leader's commands or state information through a communication chain (*i.e.*, from the leader to its direct neighbors and further down the line). To ensure that all agents, including those without direct lines to the leader, can achieve consensus, consensus protocols allow agents to update their states based on their information and the information received from neighboring agents. For instance, at each time step, a follower updates its state by considering the state of its neighbors, thus indirectly incorporating the leader's influence. The message

propagation process explicitly details how the leader's state or directives are broadcasted throughout the network. This may involve strategies such as flooding (where messages are rapidly disseminated) or controlled dissemination based on a time or distance metric. The implications of communication delays, packet loss, and the potential need for robust communication protocols (such as acknowledgments or redundancy) to ensure that critical information from the leader reaches all agents effectively might be further discussed.

Let the adjacency matrix $\bar{\mathbf{A}} = [a_{ij}] \in \mathcal{R}^{(N+1) \times (N+1)}$ is such that $a_{ij} > 0$ if $(j, i) \in \bar{E}$ and $a_{ij} = 0$, otherwise. It is assumed that $a_{ii} = 0$ for all $i \in \bar{V}$ and $a_{0j} = 0$ for all $j \in V$. The Laplacian matrix $\mathbf{L} = [l_{ij}] \in \mathcal{R}^{N \times N}$ is defined as $l_{ii} = \sum_{j=1}^N a_{ij}$ and $l_{ij} = -a_{ij}$ for $j \neq i$. Finally, the matrix \mathbf{H} is defined as $\mathbf{H} = \mathbf{L} + \mathbf{diag}(a_{10}, \dots, a_{N0})$.

Lemma 1. (Ni and Cheng [18]):

- (i) The matrix \mathbf{H} has nonnegative eigenvalues.
- (ii) The matrix \mathbf{H} is positive definite if and only if the graph \bar{G} is connected.

Let $\lambda_1, \lambda_2, \dots, \lambda_N$ are the eigenvalues of \mathbf{H} and $\lambda_{\min}(\mathbf{H})$ denotes the smallest eigenvalue of \mathbf{H} . The Lemma 1 implies that $\lambda_{\min}(\mathbf{H}) > 0$ if and only if the graph \bar{G} is connected.

Lemma 2. (Boyd *et al.* [19]): Let \mathbf{S} be a symmetric real matrix. Suppose that \mathbf{S} partitioned as $\mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{12}^T & \mathbf{S}_{22} \end{bmatrix}$

then $\mathbf{S} \succ 0$ if and only if

- (i) $\mathbf{S}_{11} \succ 0$ and $\mathbf{S}_{22} - \mathbf{S}_{12}^T \mathbf{S}_{11}^{-1} \mathbf{S}_{12} \succ 0$
- (ii) $\mathbf{S}_{22} \succ 0$ and $\mathbf{S}_{11} - \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \mathbf{S}_{12}^T \succ 0$

Consider a group of agents which consists of N followers and a leader. The second-order dynamics of the follower agents is given by

$$\begin{cases} \dot{x}_i = v_i, \\ \dot{v}_i = u_i + f_i(t), \quad i = 1, 2, \dots, N \end{cases} \quad (1)$$

where $x_i \in \mathcal{R}$, $v_i \in \mathcal{R}$, $u_i \in \mathcal{R}$ and $f_i \in \mathcal{R}$ are the position, velocity, control input and unknown time-varying dynamics of the i th follower, respectively.

Also, the dynamic of the leader is governed by the following equation

$$\begin{cases} \dot{x}_0 = v_0, \\ \dot{v}_0 = u_0(t) \end{cases} \quad (2)$$

where $x_0 \in \mathcal{R}$, $v_0 \in \mathcal{R}$ and $u_0 \in \mathcal{R}$ are the position, velocity and acceleration of the leader, respectively.

The tracking errors of position and velocity of the i th agent are defined as $\tilde{x}_i(t) = x_i(t) - x_0(t)$ and $\tilde{v}_i(t) = v_i(t) - v_0(t)$, respectively. Also, the relative position measurement and relative velocity measurement are defined as

$$\begin{cases} e_{xi} = \sum_{j=0}^N a_{ij}(x_i - x_j), \\ e_{vi} = \sum_{j=0}^N a_{ij}(v_i - v_j), \quad i = 1, 2, \dots, N \end{cases} \quad (3)$$

where a_{ij} is the (i, j) entry of the adjacency matrix $\bar{\mathbf{A}}$.

Definition 1. Second order leader-follower consensus of multi-agent system described by (1) and (2) is said to be achieved if

$$\lim_{t \rightarrow \infty} \tilde{x}_i(t) = 0, \quad \lim_{t \rightarrow \infty} \tilde{v}_i(t) = 0, \quad i = 1, 2, \dots, N \quad (4)$$

3. Solution of the Communication-Loss Problem

This section deals with the solution procedure for the communication-loss problem in the network of followers and leaders. Based on the techniques of classical adaptive control ([15,16]), the unknown acceleration of the leader, u_0 , and the unknown dynamics of each agent, $f_i(t)$, can be parameterized approximately as a linear regression form

$$u_0 \simeq \phi_0^T(t)\theta_0, \quad f_i \simeq \phi_i^T(t)\theta_i \tag{5}$$

where $\phi_0 \in \mathcal{R}^{m_0}$ and $\phi_i \in \mathcal{R}^{m_i}$ are the basis functions associated with the parameters θ_0 and θ_i . The numbers m_0 and m_i are the dimension of the basis functions associated with the parameter vectors θ_0 and θ_i , respectively. The arrays $\theta_0 \in \mathcal{R}^{m_0}$ and $\theta_i \in \mathcal{R}^{m_i}$ are time-invariant parameters of the leader’s acceleration and agents’ uncertainties which are unknown to each follower.

For compensating the terms u_0 and f_i , the parameter arrays θ_0 and θ_i should be estimated by each follower. Assume that the i th follower estimates the θ_0 and θ_i by $\hat{\theta}_{0i}$ and $\hat{\theta}_i$, respectively; and also u_0 and f_i by \hat{u}_{0i} and \hat{f}_i which so that

$$\hat{u}_{0i} = \phi_0^T(t)\hat{\theta}_{0i}, \quad \hat{f}_i = \phi_i^T(t)\hat{\theta}_i, \quad i = 1, 2, \dots, N \tag{6}$$

When the velocity measurement of agents is not available, a filter for each follower agent can be implemented according to the procedure of study 22,

$$w_i = \vartheta_i + be_{xi}, \quad \dot{\vartheta}_i = -aw_i, \quad i = 1, \dots, N \tag{7}$$

where $w_i \in \mathcal{R}$ and $\vartheta_i \in \mathcal{R}$ are the filter output and auxiliary filter vector, respectively. The coefficients a and b are positive constant quantities, which are design parameters. By eliminating the auxiliary filter vector, ϑ_i , (7) can be written as

$$\dot{w}_i = -aw_i + be_{vi}, \quad i = 1, \dots, N \tag{8}$$

Now, the following adaptive control input is introduced for system (1) as

$$u_i = -\alpha(w_i + e_{xi}) + \phi_0^T(t)\hat{\theta}_{0i} - \phi_i^T(t)\hat{\theta}_i, \quad i = 1, \dots, N \tag{9}$$

where α is a positive constant parameter. In (9), the term $-\alpha(w_i + e_{xi})$ is stabilizing part. The adaptive terms $\phi_i^T(t)\hat{\theta}_i$, $\phi_0^T(t)\hat{\theta}_{0i}$ compensate the unknown dynamics of each agent and leader acceleration, respectively.

Before giving the adaptive laws for estimated parameter arrays $\hat{\theta}_{0i}$, $\hat{\theta}_i$, primitively two auxiliary arrays are defined as

$$\begin{cases} \psi_{0i} = \hat{\theta}_{0i} + \beta\phi_0e_{xi}, \\ \psi_i = \hat{\theta}_i - \beta\phi_ie_{xi}, \end{cases} \quad i = 1, \dots, N \tag{10}$$

Design of the adaptive laws for $\hat{\theta}_{0i}$, $\hat{\theta}_i$ is equivalent to design of adaptive laws for auxiliary arrays ψ_{0i} , ψ_i which are given as

$$\begin{cases} \dot{\psi}_{0i} = (-\gamma\phi_0 + \beta\dot{\phi}_0)e_{xi} + \gamma\phi_0w_i, \\ \dot{\psi}_i = (\gamma\phi_i - \beta\dot{\phi}_i)e_{xi} - \gamma\phi_iw_i, \end{cases} \quad i = 1, \dots, N \tag{11}$$

where β and γ are positive constant parameters. By substituting (10) in (11), the following result appears

$$\begin{cases} \dot{\hat{\theta}}_{0i} = -\gamma\phi_0e_{xi} - \beta\phi_0e_{vi} + \gamma\phi_0w_i, \\ \dot{\hat{\theta}}_i = \gamma\phi_ie_{xi} + \beta\phi_ie_{vi} - \gamma\phi_iw_i, \end{cases} \quad i = 1, \dots, N \tag{12}$$

By application of the (7) and (9) to the system (1) and (2), the closed-loop system can be written as

$$\begin{cases} \dot{\tilde{x}} = \tilde{v}, \\ \dot{\tilde{v}} = -\alpha H\tilde{x} - \alpha w + \Phi_0^T\tilde{\theta}_0 - \Phi^T\tilde{\theta}, \\ \dot{w} = bH\tilde{v} - aw \end{cases} \tag{13}$$

With the adaptive laws

$$\begin{cases} \dot{\tilde{\theta}}_0 = -\gamma\Phi_0H\tilde{x} - \beta\Phi_0H\tilde{v} + \gamma\Phi_0w, \\ \dot{\tilde{\theta}} = \gamma\Phi H\tilde{x} + \beta\Phi H\tilde{v} - \gamma\Phi w \end{cases} \tag{14}$$

where $\tilde{x} = col\{\tilde{x}_1, \dots, \tilde{x}_N\}$, $\tilde{v} = col\{\tilde{v}_1, \dots, \tilde{v}_N\}$, $w = col\{w_1, \dots, w_N\}$, $\tilde{\theta}_0 = col\{\tilde{\theta}_{01}, \dots, \tilde{\theta}_{0N}\}$, $\tilde{\theta} = col\{\tilde{\theta}_1, \dots, \tilde{\theta}_N\}$, $\Phi_0 = I_N \otimes \phi_0(t)$, $\Phi = diag\{\phi_1, \dots, \phi_N\}$ and also $\tilde{\theta}_{0i} = \hat{\theta}_{0i} - \theta_0$, $\tilde{\theta}_i = \hat{\theta}_i - \theta_i$.

The following theorem presents a sufficient condition for the stability of the closed-loop system (13) and (14).

Theorem 1. (PDAC Method): It is supposed that in the leader-following system (1) and (2), G is undirected in \bar{G} the leader has directed paths to all followers, and the basic functions ϕ_0 and ϕ_i are uniformly bounded. Then, the leader-following consensus under (9) and (11) is achieved if the following conditions are satisfied

$$\left\{ \begin{array}{l} b > 1, \quad a > \frac{\gamma}{\beta}b, \\ \alpha > \max \left\{ \frac{1+b}{\lambda_{\min}} \left(\frac{\gamma}{\beta} \right)^2, \frac{a^2 \left(\frac{\gamma}{\beta} \right)}{4(b-1) \left(\frac{a}{b} - \frac{\gamma}{\beta} \right) \lambda_{\min}} \right\}, \quad \lambda_{\min} = \lambda_{\min}(\mathbf{H}) \end{array} \right. \quad (15)$$

Proof. If $\mathbf{z} = [\tilde{\mathbf{x}}^T \quad \tilde{\mathbf{v}}^T \quad \mathbf{w}^T]^T$, the following Lyapunov function can be introduced

$$\begin{aligned} V(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, \mathbf{w}, \tilde{\boldsymbol{\theta}}_0, \tilde{\boldsymbol{\theta}}) &= \alpha\beta\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{x}} + 2\gamma\tilde{\mathbf{x}}^T \mathbf{H} \tilde{\mathbf{v}} + \beta\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} - 2\gamma\tilde{\mathbf{v}}^T \mathbf{w} + \frac{\alpha\beta}{b} \mathbf{w}^T \mathbf{w} + \tilde{\boldsymbol{\theta}}_0^T \tilde{\boldsymbol{\theta}}_0 + \tilde{\boldsymbol{\theta}}^T \tilde{\boldsymbol{\theta}} \\ &= \mathbf{z}^T \underbrace{\begin{bmatrix} \alpha\beta\mathbf{H}^2 & \gamma\mathbf{H} & \mathbf{0}_{N \times N} \\ \gamma\mathbf{H} & \beta\mathbf{H} & -\gamma\mathbf{I}_N \\ \mathbf{0}_{N \times N} & -\gamma\mathbf{I}_N & \frac{\alpha\beta}{b}\mathbf{I}_N \end{bmatrix}}_{\mathbf{P}} \mathbf{z} + \tilde{\boldsymbol{\theta}}_0^T \tilde{\boldsymbol{\theta}}_0 + \tilde{\boldsymbol{\theta}}^T \tilde{\boldsymbol{\theta}} \end{aligned} \quad (16)$$

The function V is a positive definite function if and only if the matrix \mathbf{P} is a positive definite matrix. By application of Lemma 2, the matrix \mathbf{P} is a positive definite matrix if and only if $\begin{bmatrix} \alpha\beta\mathbf{H}^2 & \gamma\mathbf{H} \\ \gamma\mathbf{H} & \beta\mathbf{H} \end{bmatrix} - \frac{b}{\alpha\beta} \begin{bmatrix} \mathbf{0}_{N \times N} \\ -\gamma\mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{0}_{N \times N} & -\gamma\mathbf{I}_N \end{bmatrix} > 0$ and $\frac{\alpha\beta}{b} \mathbf{I}_N > 0$. And also, $\begin{bmatrix} \alpha\beta\mathbf{H}^2 & \gamma\mathbf{H} \\ \gamma\mathbf{H} & \beta\mathbf{H} \end{bmatrix} - \frac{b}{\alpha\beta} \begin{bmatrix} \mathbf{0}_{N \times N} \\ -\gamma\mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{0}_{N \times N} & -\gamma\mathbf{I}_N \end{bmatrix} = \begin{bmatrix} \alpha\beta\mathbf{H}^2 & \gamma\mathbf{H} \\ \gamma\mathbf{H} & \beta\mathbf{H} - \frac{b}{\alpha\beta}\gamma^2\mathbf{I}_N \end{bmatrix} > 0$ if and only if $\beta\mathbf{H} - \frac{(1+b)}{\alpha\beta}\gamma^2\mathbf{I}_N > 0$ and $\alpha\beta\mathbf{H}^2 > 0$. Since the graph G is undirected and in \bar{G} the leader has directed paths to all followers; we can get from Lemma 1 that the matrix \mathbf{H} is a symmetric positive definite matrix and also $\alpha, \beta, b > 0$, which leads to $\frac{\alpha\beta}{b} \mathbf{I}_N > 0, \alpha\beta\mathbf{H}^2 > 0$. Also, from Lemma 1, it is known that $\mathbf{H} > \lambda_{\min} \mathbf{I}_N$; then, if $\alpha > \frac{1+b}{\lambda_{\min}} \left(\frac{\gamma}{\beta} \right)^2$ it is concluded that $\beta\mathbf{H} - \frac{(1+b)}{\alpha\beta}\gamma^2\mathbf{I}_N > 0$, which consequently results $\mathbf{P} > 0$ and V is a positive definite function.

Now, by time differentiation of V along (13), (14),

$$\begin{aligned} \dot{V} &= 2\alpha\beta\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{v}} + 2\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\gamma\tilde{\mathbf{x}}^T \mathbf{H} [-\alpha\mathbf{H}\tilde{\mathbf{x}} - \alpha\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] \\ &\quad + 2\beta\tilde{\mathbf{v}}^T \mathbf{H} [-\alpha\mathbf{H}\tilde{\mathbf{x}} - \alpha\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] - 2\gamma\tilde{\mathbf{v}}^T [b\mathbf{H}\tilde{\mathbf{v}} - a\mathbf{w}] \\ &\quad - 2\gamma\mathbf{w}^T [-\alpha\mathbf{H}\tilde{\mathbf{x}} - \alpha\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] + 2\frac{c\beta}{b} \mathbf{w}^T [b\mathbf{H}\tilde{\mathbf{v}} - a\mathbf{w}] \\ &\quad + 2\tilde{\boldsymbol{\theta}}_0^T [-\gamma\boldsymbol{\Phi}_0 \mathbf{H} \tilde{\mathbf{x}} - \beta\boldsymbol{\Phi}_0 \mathbf{H} \tilde{\mathbf{v}} + \gamma\boldsymbol{\Phi}_0 \mathbf{w}] + 2\tilde{\boldsymbol{\theta}}^T [\gamma\boldsymbol{\Phi} \mathbf{H} \tilde{\mathbf{x}} + \beta\boldsymbol{\Phi} \mathbf{H} \tilde{\mathbf{v}} - \gamma\boldsymbol{\Phi} \mathbf{w}], \end{aligned} \quad (17)$$

By simplifying (17) and giving a pretty form to it,

$$\begin{aligned} \dot{V}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, \mathbf{w}, \tilde{\boldsymbol{\theta}}_0, \tilde{\boldsymbol{\theta}}) &= -2\alpha\gamma\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{x}} - 2(b-1)\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\alpha\gamma\tilde{\mathbf{v}}^T \mathbf{w} - 2\left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma\right) \mathbf{w}^T \mathbf{w} \\ &= -\mathbf{z}^T \underbrace{\begin{bmatrix} 2\alpha\gamma\mathbf{H}^2 & \mathbf{0}_{N \times N} & \mathbf{0}_{N \times N} \\ \mathbf{0}_{N \times N} & 2(b-1)\gamma\mathbf{H} & -\alpha\gamma\mathbf{I}_N \\ \mathbf{0}_{N \times N} & -\alpha\gamma\mathbf{I}_N & 2\left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma\right)\mathbf{I}_N \end{bmatrix}}_{\mathbf{Q}} \mathbf{z} \end{aligned} \quad (18)$$

The matrix \mathbf{Q} is a positive definite matrix if and only if $\begin{bmatrix} 2(b-1)\gamma\mathbf{H} & -\alpha\gamma\mathbf{I}_N \\ -\alpha\gamma\mathbf{I}_N & 2\left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma\right)\mathbf{I}_N \end{bmatrix} > 0$ and $2\alpha\gamma\mathbf{H}^2 >$

0. By application of the Lemma 2, $\begin{bmatrix} 2(b-1)\gamma\mathbf{H} & -\alpha\gamma\mathbf{I}_N \\ -\alpha\gamma\mathbf{I}_N & 2\left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma\right)\mathbf{I}_N \end{bmatrix} > 0$ if and only if $2(b-1)\gamma\mathbf{H} > 0$ and

$2(b - 1)\gamma\mathbf{H} - \frac{a^2\gamma^2}{2\alpha\beta(\frac{a-\gamma}{b-\beta})}\mathbf{I}_N > 0$. If condition (15) is satisfied, then the results $2(b - 1)\gamma\mathbf{H} - \frac{a^2\gamma^2}{2\alpha\beta(\frac{a-\gamma}{b-\beta})}\mathbf{I}_N > 0$,

$2(b - 1)\gamma\mathbf{H} > 0$ are concluded. Hence, the matrix \mathbf{Q} is a positive definite matrix and \dot{V} is negative semi-definite.

The function V is a positive quantity, so it is lower bound. Also \dot{V} is negative semi-definite. Since \dot{V} is negative semi-definite, it is concluded that \mathbf{z} , $\tilde{\boldsymbol{\theta}}_0$, $\tilde{\boldsymbol{\theta}}$ are bounded. Thus, from (13), (14) and uniformly boundedness condition of the basis functions $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_i$, $\dot{\mathbf{z}}$ is bounded. By taking the derivative of \dot{V} in (18) along (13), (14), the \ddot{V} is assessed to be bounded. Now, by resorting to the Barbalat's Lemma [20] that $\lim_{t \rightarrow \infty} \dot{V}(t) = 0$, the consequence $\lim_{t \rightarrow \infty} \mathbf{z}(t) = 0$ is achieved. \square

Specific Example 1. Consider a multi-agent system with one leader and follower. Assume that $u_0(t) = 1.0$ and $f_1(t) = 1.0$. Also assume that $a_{10} = 1.0$ then

$$e_{x1} = \tilde{x}_1 \tag{19}$$

By selecting the basis function $\phi_0 = 1.0$ and $\phi_1 = 1.0$, Adaptive laws based on Equation (11) can be written as given

$$\begin{cases} \dot{\psi}_{01} = -\gamma\tilde{x}_1 + \gamma w_1, \\ \dot{\psi}_1 = \gamma\tilde{x}_1 - \gamma w_1, \end{cases} \tag{20}$$

and from Equation (10), $\hat{\theta}_{01}$ and $\hat{\theta}_1$ can be written as

$$\begin{cases} \hat{\theta}_{01} = \psi_{01} - \beta\tilde{x}_1, \\ \hat{\theta}_1 = \psi_1 + \beta\tilde{x}_1 \end{cases} \tag{21}$$

Differentiating from the above equation gives the following set of differential equations

$$\begin{cases} \dot{\tilde{\theta}}_{01} = -\gamma\tilde{x}_1 - \beta\tilde{v}_1 + \gamma w_1, \\ \dot{\tilde{\theta}}_1 = \gamma\tilde{x}_1 + \beta\tilde{v}_1 - \gamma w_1 \end{cases} \tag{22}$$

Finally, the closed-loop system is given as follows

$$\begin{cases} \dot{\tilde{x}}_1 = \tilde{v}_1, \\ \dot{\tilde{v}}_1 = -\alpha\tilde{x}_1 - \alpha w_1 + \tilde{\theta}_{01} - \tilde{\theta}_1, \\ \dot{w}_1 = b\tilde{v}_1 - \alpha w_1 \end{cases} \tag{23}$$

By defining $\omega = \tilde{\theta}_{01} - \tilde{\theta}_1$ and selecting $a = 1, b = 2, \alpha = 2, \beta = 3, \gamma = 1$ such that the (15) satisfied, the (22) and (23) can be written as

$$\begin{bmatrix} \dot{\tilde{x}}_1 \\ \dot{\tilde{v}}_1 \\ \dot{w}_1 \\ \dot{\omega} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & 0 & -2 & 1 \\ 0 & 2 & -1 & 0 \\ -2 & -6 & 2 & 0 \end{bmatrix}}_{\mathbf{\Xi}} \begin{bmatrix} \tilde{x}_1 \\ \tilde{v}_1 \\ w_1 \\ \omega \end{bmatrix} \tag{24}$$

The stability of the system (24) depends on the eigenvalues of the matrix $\mathbf{\Xi}$. The eigenvalues of the matrix $\mathbf{\Xi}$ are $\lambda = -0.2445 \pm 3.3938i, -0.2555 \pm 0.3278i$ which implies the exponential stability of (24).

Remark 1. To satisfy the condition on control gain α in Theorem 1, centralized information depending on the communication graph must be known in advance for the control gain design. Due to this reason, the control algorithm (9), (11) proposed to the multi-agent system (1), (2) is not distributed. So, in order to design a fully distributed consensus algorithm, the adaptive gain method introduced in [21,22] is applied. The question regarding the feasibility of determining the necessary eigenvalues in a distributed manner is pertinent and highlights the challenges inherent in decentralized control systems. In multi-agent systems, the eigenvalues of the system's coupling matrix (often derived from the graph that represents the connectivity of agents) play a critical role in assessing system stability, convergence properties, and performance. These values may relate to the system's ability to achieve consensus, formation, and control objectives. While techniques are available for distributed computation of eigenvalues, such as consensus algorithms or distributed iterative methods (e.g., power iteration and methods based on local coordination), these approaches can be computationally intensive. They may require multiple iterations to converge to an accurate

value. The feasibility of such methods largely depends on the graph topology and the network’s connectivity. In many practical scenarios, having prior knowledge of certain eigenvalues (e.g., the largest eigenvalue correlated with the system’s stability) can significantly simplify the design and implementation of control strategies. When these eigenvalues are known, controllers can be designed with specific bandwidth and robustness properties in mind. However, in the absence of such knowledge, algorithms that enable distributed estimation of eigenvalues can be employed. Additionally, approximate methods for estimating eigenvalues or using local information to infer global properties can sometimes be effective. Distributed algorithms can leverage neighboring agents’ relative positions and velocities to asymptotically converge on the desired values. While it is certainly possible to determine eigenvalues in a distributed manner, the effectiveness and efficiency of such an approach can vary based on specific system dynamics and network topology. Therefore, if prior knowledge of these eigenvalues is available, it can greatly enhance the control system’s design and performance.

According to Remark 1, an adaptive control algorithm that is fully distributed is introduced. Consider the following control law

$$u_i = -c_i(t)(w_i + e_{xi}) + \phi_0^T(t)\hat{\theta}_{0i} - \phi_i^T(t)\hat{\theta}_i, \quad i = 1, \dots, N \tag{25}$$

where, $c_i(t)$ is the adaptive gain.

Before giving adaptive law for $c_i(t)$, an auxiliary parameter is defined as

$$\eta_i(t) = c_i(t) - \mu\beta e_{xi}w_i + \mu\beta(b - 1)e_{xi}^2, \quad i = 1, \dots, N \tag{26}$$

where μ is a positive constant parameter. The adaptation law for η_i is given as

$$\dot{\eta}_i = \mu(\gamma e_{xi}^2 + \beta a e_{xi}w_i - \gamma w_i^2), \quad i = 1, \dots, N \tag{27}$$

The parameters a , b , β and γ were introduced previously by (7) and (11). Similar to (12), by substituting (26) into (27), the following result is obtained

$$\dot{c}_i = \mu(\gamma e_{xi}^2 + \beta e_{xi}e_{vi} + \beta e_{vi}w_i - \gamma w_i^2), \quad i = 1, \dots, N \tag{28}$$

By application of (19) to system (1), (2), the feedback form of the system is obtained as

$$\begin{cases} \dot{\tilde{x}} = \tilde{v}, \\ \dot{\tilde{v}} = -C(t)H\tilde{x} - C(t)w + \phi_0^T\tilde{\theta}_0 - \phi^T\tilde{\theta}, \\ \dot{w} = bH\tilde{v} - aw \end{cases} \tag{29}$$

where $C(t) = \text{diag}\{c_1(t), \dots, c_N(t)\}$.

Now, the second theorem of this study is presented.

Theorem 2. (FDAC Method): It is supposed that in the leader-following system (1) and (2), G is undirected in \bar{G} the leader has directed paths to all followers, and the basis functions ϕ_0 and ϕ_i are uniformly bounded. Then, the leader-following consensus under (11), (25) and (27) is achieved if the following conditions are satisfied

$$b > 1, \quad a > \frac{\gamma}{\beta}b \tag{30}$$

Proof. If $z = [\tilde{x}^T \quad \tilde{v}^T \quad w^T]^T$, the following Lyapunov function can be introduced

$$\begin{aligned} V(\tilde{x}, \tilde{v}, w, \tilde{\theta}_0, \tilde{\theta}) &= \alpha\beta\tilde{x}^T H^2\tilde{x} + 2\gamma\tilde{x}^T H\tilde{v} + \beta\tilde{v}^T H\tilde{v} - 2\gamma\tilde{v}^T w + \frac{\alpha\beta}{b}w^T w + \tilde{\theta}_0^T\tilde{\theta}_0 + \tilde{\theta}^T\tilde{\theta} + \frac{1}{\mu}\sum_{i=1}^N (c_i(t) - \alpha)^2 \\ &= z^T \begin{bmatrix} \alpha\beta H^2 & \gamma H & \mathbf{0}_{N \times N} \\ \gamma H & \beta H & -\gamma I_N \\ \mathbf{0}_{N \times N} & -\gamma I_N & \frac{\alpha\beta}{b} I_N \end{bmatrix} z + \tilde{\theta}_0^T\tilde{\theta}_0 + \tilde{\theta}^T\tilde{\theta} + \frac{1}{\mu}\sum_{i=1}^N (c_i(t) - \alpha)^2 \end{aligned} \tag{31}$$

where, in the proposed Lyapunov function, the parameter α is constant which should be determined. The parameter α is selected such that $\alpha > \max \left\{ \frac{1+b}{\lambda_{\min}} \left(\frac{\gamma}{\beta} \right)^2, \frac{a^2 \left(\frac{\gamma}{\beta} \right)}{4(b-1) \left(\frac{a-\gamma}{b} \right) \lambda_{\min}} \right\}$. Similar to Theorem 1, it is shown that the function V is positive definite.

By time differentiation of V along (14), (28) and (29)

$$\begin{aligned} \dot{V}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, \mathbf{w}, \tilde{\boldsymbol{\theta}}_0, \tilde{\boldsymbol{\theta}}) &= 2\alpha\beta\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{v}} + 2\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\gamma\tilde{\mathbf{x}}^T \mathbf{H} [-\mathbf{C}(t)\mathbf{H}\tilde{\mathbf{x}} - \mathbf{C}(t)\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] \\ &+ 2\beta\tilde{\mathbf{v}}^T \mathbf{H} [-\mathbf{C}(t)\mathbf{H}\tilde{\mathbf{x}} - \mathbf{C}(t)\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] - 2\gamma\tilde{\mathbf{v}}^T [b\mathbf{H}\tilde{\mathbf{v}} - a\mathbf{w}] \\ &- 2\gamma\mathbf{w}^T [-\mathbf{C}(t)\mathbf{H}\tilde{\mathbf{x}} - \mathbf{C}(t)\mathbf{w} + \boldsymbol{\Phi}_0^T \tilde{\boldsymbol{\theta}}_0 - \boldsymbol{\Phi}^T \tilde{\boldsymbol{\theta}}] + 2\frac{\alpha\beta}{b} \mathbf{w}^T [b\mathbf{H}\tilde{\mathbf{v}} - a\mathbf{w}] \\ &+ 2\tilde{\boldsymbol{\theta}}_0^T [-\gamma\boldsymbol{\Phi}_0 \mathbf{H}\tilde{\mathbf{x}} - \beta\boldsymbol{\Phi}_0 \mathbf{H}\tilde{\mathbf{v}} + \gamma\boldsymbol{\Phi}_0 \mathbf{w}] + 2\tilde{\boldsymbol{\theta}}^T [\gamma\boldsymbol{\Phi} \mathbf{H}\tilde{\mathbf{x}} + \beta\boldsymbol{\Phi} \mathbf{H}\tilde{\mathbf{v}} - \gamma\boldsymbol{\Phi} \mathbf{w}] \\ &+ 2 \sum_{i=1}^N [(c_i(t) - \alpha)(\gamma e_{xi}^2 + \beta e_{xi} e_{vi} + \beta e_{vi} w_i - \gamma w_i^2)] \end{aligned} \tag{32}$$

By defining $\tilde{c}_i(t) = c_i(t) - \alpha$ and also $\tilde{\mathbf{C}}(t) = \text{diag}\{\tilde{c}_1(t), \dots, \tilde{c}_N(t)\}$, and substituting into the (32)

$$\begin{aligned} \dot{V} &= -2\alpha\gamma\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{x}} - 2(b-1)\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\alpha\gamma\tilde{\mathbf{v}}^T \mathbf{w} - 2 \left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma \right) \mathbf{w}^T \mathbf{w} + 2[-\gamma\tilde{\mathbf{x}}^T \mathbf{H} - \beta\tilde{\mathbf{v}}^T \mathbf{H} + \gamma\mathbf{w}^T] [\tilde{\mathbf{C}}(t)\mathbf{H}\tilde{\mathbf{x}} + \tilde{\mathbf{C}}(t)\mathbf{w}] \\ &+ 2 \sum_{i=1}^N [\tilde{c}_i(t)(\gamma e_{xi}^2 + \beta e_{xi} e_{vi} + \beta e_{vi} w_i - \gamma w_i^2)] \end{aligned} \tag{33}$$

If $\mathbf{e}_x = \text{col}\{e_{x1}, \dots, e_{xN}\}$ and $\mathbf{e}_v = \text{col}\{e_{v1}, \dots, e_{vN}\}$, then by using the definition of matrix \mathbf{H} , $\mathbf{e}_x = \mathbf{H}\tilde{\mathbf{x}}$ and $\mathbf{e}_v = \mathbf{H}\tilde{\mathbf{v}}$ are obtained. Hence, the term $[-\gamma\tilde{\mathbf{x}}^T \mathbf{H} - \beta\tilde{\mathbf{v}}^T \mathbf{H} + \gamma\mathbf{w}^T] [\tilde{\mathbf{C}}(t)\mathbf{H}\tilde{\mathbf{x}} + \tilde{\mathbf{C}}(t)\mathbf{w}]$ can be written in a summation form which is presented as

$$\begin{aligned} \dot{V}(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}, \mathbf{w}, \tilde{\boldsymbol{\theta}}_0, \tilde{\boldsymbol{\theta}}) &= -2\alpha\gamma\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{x}} - 2(b-1)\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\alpha\gamma\tilde{\mathbf{v}}^T \mathbf{w} - 2 \left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma \right) \mathbf{w}^T \mathbf{w} \\ &+ 2 \sum_{i=1}^N [(-\gamma e_{xi} - \beta e_{vi} + \gamma w_i) \tilde{c}_i(t) (e_{xi} + w_i)] + 2 \sum_{i=1}^N [\tilde{c}_i(t) (\gamma e_{xi}^2 + \beta e_{xi} e_{vi} + \beta e_{vi} w_i - \gamma w_i^2)] \end{aligned} \tag{34}$$

Consequently

$$\begin{aligned} \dot{V} &= -2\alpha\gamma\tilde{\mathbf{x}}^T \mathbf{H}^2 \tilde{\mathbf{x}} - 2(b-1)\gamma\tilde{\mathbf{v}}^T \mathbf{H} \tilde{\mathbf{v}} + 2\alpha\gamma\tilde{\mathbf{v}}^T \mathbf{w} - 2 \left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma \right) \mathbf{w}^T \mathbf{w} \\ &= -\mathbf{z}^T \underbrace{\begin{bmatrix} 2\alpha\gamma\mathbf{H}^2 & \mathbf{0}_{N \times N} & \mathbf{0}_{N \times N} \\ \mathbf{0}_{N \times N} & 2(b-1)\gamma\mathbf{H} & -\alpha\gamma\mathbf{I}_N \\ \mathbf{0}_{N \times N} & -\alpha\gamma\mathbf{I}_N & 2 \left(\frac{\alpha\alpha\beta}{b} - \alpha\gamma \right) \mathbf{I}_N \end{bmatrix}}_{\hat{\mathbf{Q}}} \mathbf{z} \end{aligned} \tag{35}$$

Similar to procedure of the Theorem 1, the following results can be obtained for the Theorem 2 according to which, $\lim_{t \rightarrow \infty} \mathbf{z}(t) = 0$. \square

To clarify, both conditions specified in Theorem 2 are necessary for ensuring the positive definiteness of the matrix in question. Each condition addresses critical aspects of the system’s dynamics and interaction structure, contributing to the overall stability and convergence towards consensus.

Condition 1 ensures that the communication topology allows for appropriate interactions among the agents, while Condition 2 guarantees that the leader can influence all followers effectively. The interplay between these conditions is essential for deriving the positive definiteness of the related matrix, which is a key step in proving the consensus result. If either condition were to be relaxed or violated, the positive definiteness may not hold, potentially jeopardizing the proof of consensus.

Finally, the proposed algorithms can be summarized. So, in the PDAC method, the controller is given from a set of equations of (36). Also the block diagram of this method is shown in Figure 1.

$$\begin{cases} \dot{\psi}_{0i} = (-\gamma\phi_0 + \beta\dot{\phi}_0)e_{xi} + \gamma\phi_0w_i, \\ \dot{\psi}_i = (\gamma\phi_i - \beta\dot{\phi}_i)e_{xi} - \gamma\phi_iw_i, \\ \hat{\theta}_{0i} = \psi_{0i} - \beta\phi_0e_{xi}, \\ \hat{\theta}_i = \psi_i + \beta\phi_ie_{xi}, \\ w_i = \vartheta_i + be_{xi}, \quad \dot{\vartheta}_i = -aw_i \\ u_i = -\alpha(w_i + e_{xi}) + \phi_0^T(t)\hat{\theta}_{0i} - \phi_i^T(t)\hat{\theta}_i, \quad i = 1, \dots, N \end{cases} \quad (36)$$

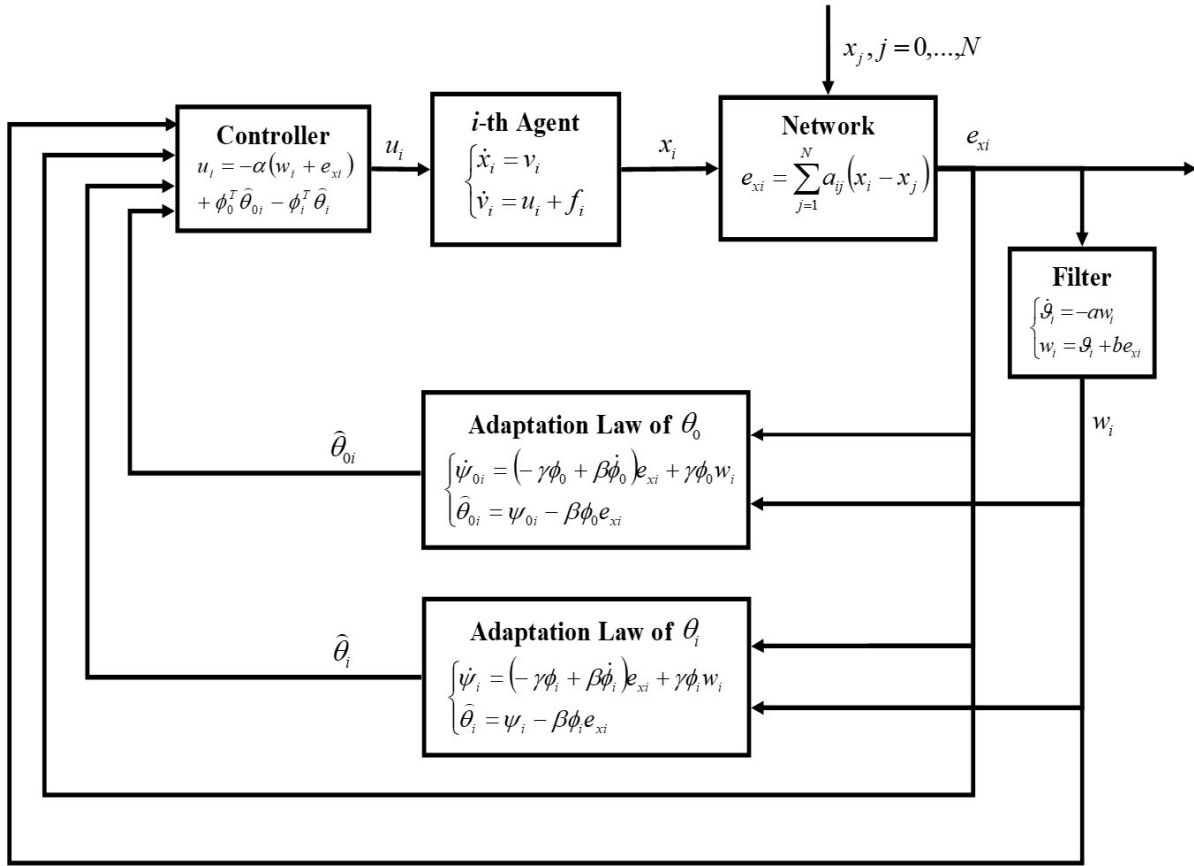


Figure 1. The feedback representation of the PDAC method.

Similar to PDAC method, the controller input in FDAC method is given from a set of equations of (37). Block diagram of this method is shown in Figure 2.

$$\begin{cases} \dot{\psi}_{0i} = (-\gamma\phi_0 + \beta\dot{\phi}_0)e_{xi} + \gamma\phi_0w_i, \\ \dot{\psi}_i = (\gamma\phi_i - \beta\dot{\phi}_i)e_{xi} - \gamma\phi_iw_i, \\ \hat{\theta}_{0i} = \psi_{0i} - \beta\phi_0e_{xi}, \\ \hat{\theta}_i = \psi_i + \beta\phi_ie_{xi}, \\ \dot{\eta}_i = \mu(\gamma e_{xi}^2 + \beta a e_{xi} w_i - \gamma w_i^2), \\ c_i(t) = \eta_i(t) + \mu\beta e_{xi} w_i - \mu\beta(b-1)e_{xi}^2, \\ w_i = \vartheta_i + be_{xi}, \quad \dot{\vartheta}_i = -aw_i \\ u_i = -c_i(t)(w_i + e_{xi}) + \phi_0^T(t)\hat{\theta}_{0i} - \phi_i^T(t)\hat{\theta}_i, \quad i = 1, \dots, N \end{cases} \quad (37)$$

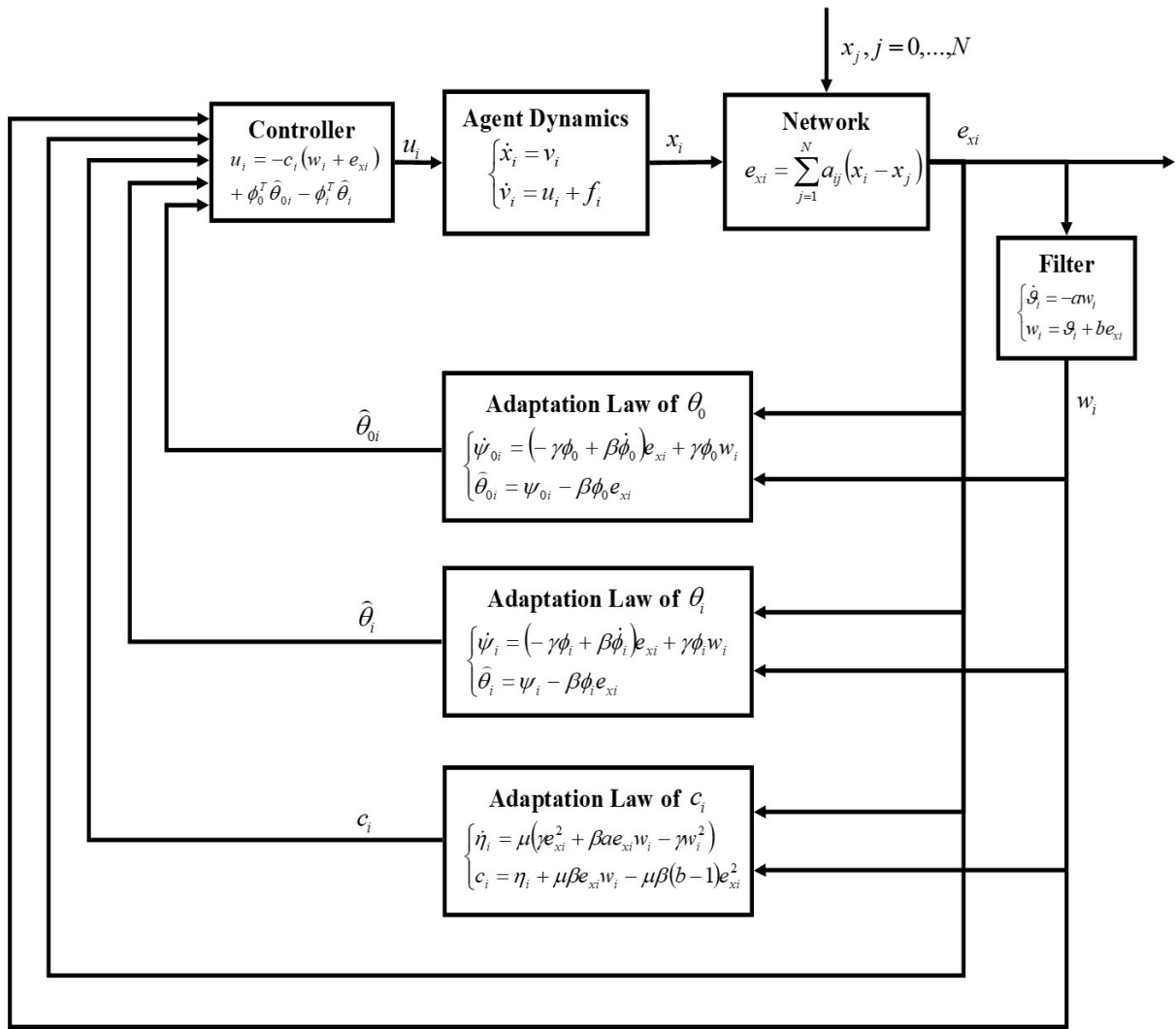


Figure 2. The feedback representation of the FDAC method.

4. Simulation Results and Discussions

4.1. Simulation Results

This section provides numerical simulations to show the effectiveness of proposed adaptive control algorithms. For numerical simulations, we consider 4 follower agents and one leader with the communication topology shown in Figure 3 and the matrix H described by (38).

$$H = \begin{bmatrix} 3 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \quad (38)$$

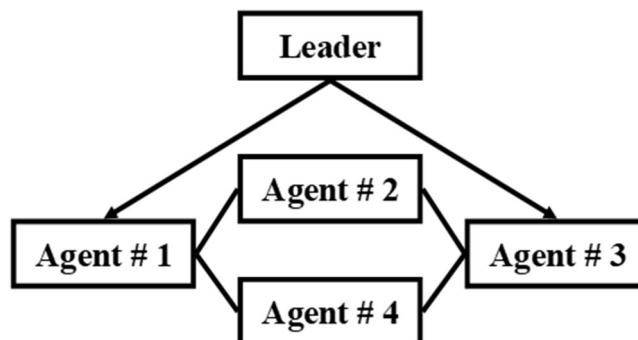


Figure 3. Communication topology of the illustrating example.

The examples included in this paper are intended to demonstrate the applicability and effectiveness of the proposed FDAC and PDAC systems in various scenarios. While diverse case studies are introduced, they are not exhaustive. Instead, they serve to illustrate the fundamental principles of the proposed approaches and their robustness under different conditions, including varying dynamics and communication constraints. The results obtained from these sample cases are consistent with the theoretical analysis presented in the paper, which establishes the stability and convergence properties of the FDAC and PDAC methods. The mathematical proofs and derivations provided in the paper underpin the generalizability of our findings, indicating that the proposed methods are not limited to the specific examples presented but are applicable to a broader class of distributed systems. It is acknowledged that additional examples would strengthen the empirical validation of our findings. The extension of analyses can be planned by incorporating a wider range of scenarios, including different network topologies, varying agent dynamics, and more complex communication patterns. This will provide further evidence of the effectiveness of the proposed methods in solving the distributed identification problem. While the examples serve as a foundation for the claims raised, the theoretical framework and stability proofs are critical components that support the conclusions that have been attained. It is realized that the combination of practical examples and rigorous theoretical analysis provides a solid basis for asserting that this study contributes to solving the identification problem in distributed systems.

Example 1. In this example, the leader acceleration and unknown nonlinear dynamics of each follower are given as

$$\begin{cases} u_0(t) = 4 \sin 2t + 3 \cos 2t, \\ f_1(t) = 2 \sin t + 2 \cos t, \\ f_2(t) = 2 \sin 3t - 3 \cos 3t, \\ f_3(t) = 5 \sin t - 2 \cos t, \\ f_4(t) = \sin 2t + 2 \cos 2t \end{cases} \quad (39)$$

In this example, we consider the structure of the u_0 and $f_i, i = 1, \dots, 4$ is known, so the basis functions are considered as $\phi_0 = \phi_4 = [\sin 2t \quad \cos 2t]^T, \phi_1 = \phi_3 = [\sin t \quad \cos t]^T, \phi_2 = [\sin 3t \quad \cos 3t]^T$. Based on the PDAC method, the controller parameters assumed to be $a = 5, b = 3, \alpha = 5, \beta = 4, \gamma = 2$. Figure 4 shows the simulation result of the controller based on the PDAC method without adaptive term compensator. Also, Figure 5 shows the adaptive controller result based on the PDAC method. Figures 6 and 7 show the convergence of the estimated parameter arrays $\hat{\theta}_{0i}, \hat{\theta}_i, i = 1, \dots, 4$. Figures 6 and 7 illustrate the convergence behavior of the estimated parameter arrays within the PDAC system over time. Each figure presents the evolution of the parameter estimates for different agents in the multi-agent network as they adapt to the dynamics of the environment. Initially, as depicted in both figures, the estimated parameters exhibit noticeable fluctuations. This behavior is expected in adaptive control systems, particularly in the early stages of operation when agents gather information and adjust their estimates based on local observations and interactions with neighboring agents. These fluctuations reflect the agents' attempts to respond to various uncertainties and disturbances in the system. After approximately 15 s, a significant change occurs, and the estimated parameters begin to stabilize. This convergence indicates that the PDAC algorithm effectively integrates the leader's and neighboring agents' information, allowing the followers to refine their parameter estimates. The stabilization of the estimated parameters signifies that the agents have reached a consensus on the underlying system dynamics, demonstrating the robustness and effectiveness of the PDAC approach. The convergence to constant values in the estimated parameters underscores the adaptability of the PDAC system in achieving consensus among agents, even in the presence of initial uncertainties. This behavior is crucial for ensuring that the multi-agent system can perform effectively in dynamic environments, as it allows the agents to maintain coordinated behavior and achieve collective goals. These figures exemplify the successful application of the PDAC algorithm, showcasing its capability to facilitate the convergence of estimated parameters in a partially distributed manner, thereby enhancing the overall performance of the multi-agent system.

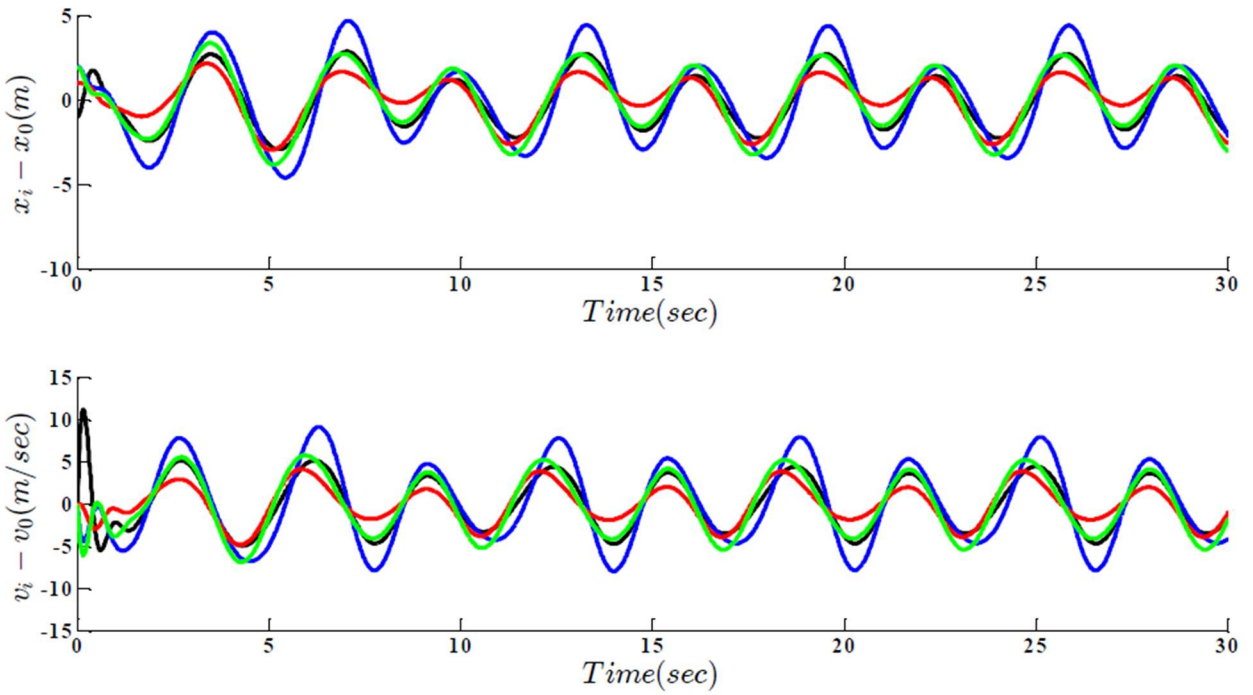


Figure 4. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, obtained from the controller without using adaptive term compensator, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

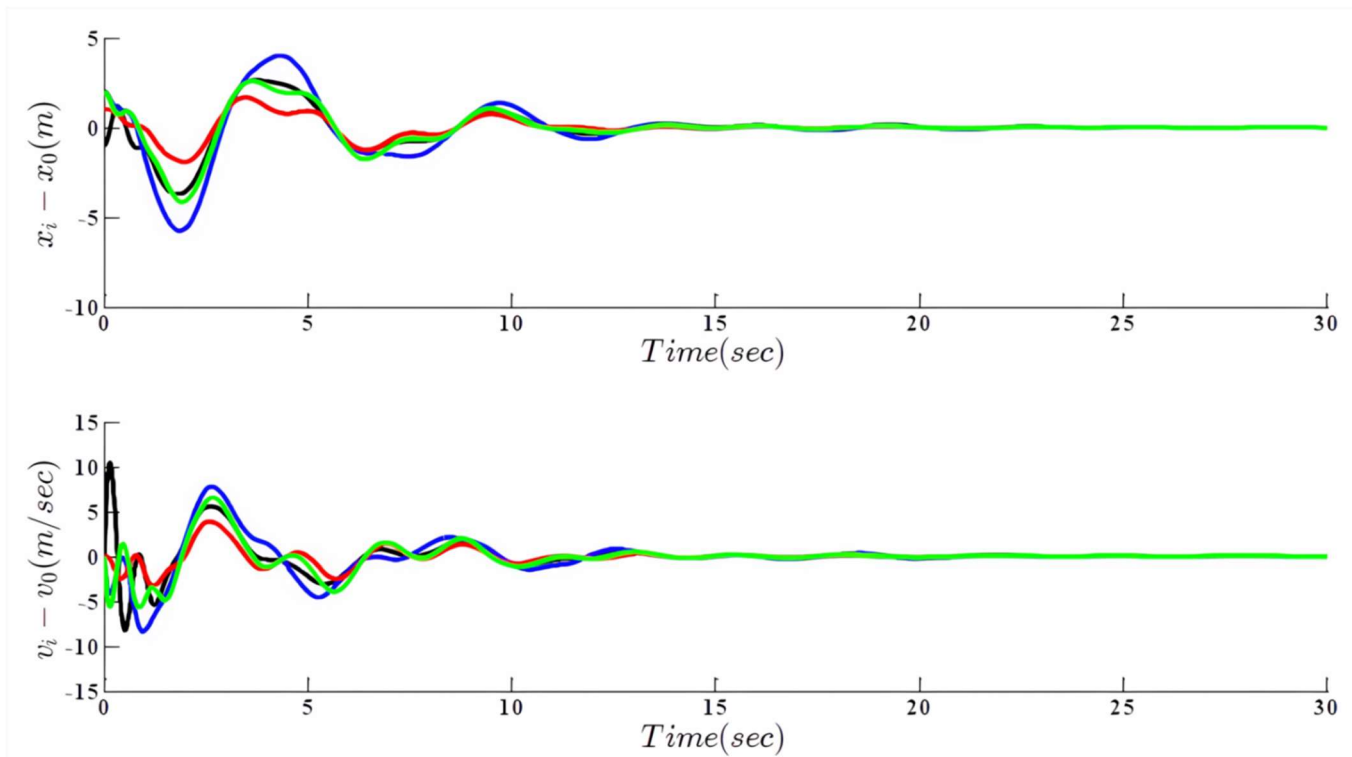


Figure 5. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the PDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

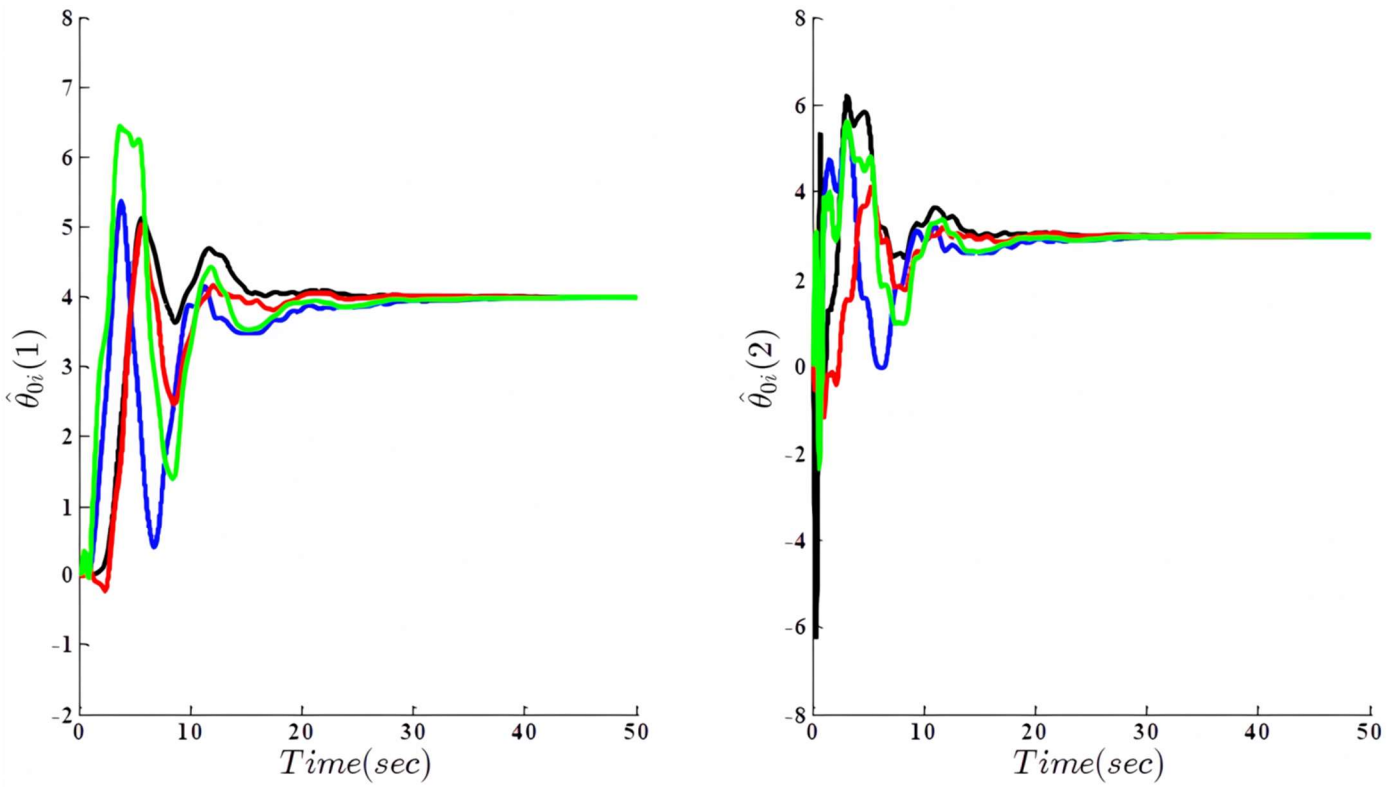


Figure 6. Convergences of the estimated parameter arrays $\hat{\theta}_{0i}, i = 1, \dots, 4$, based on the PDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

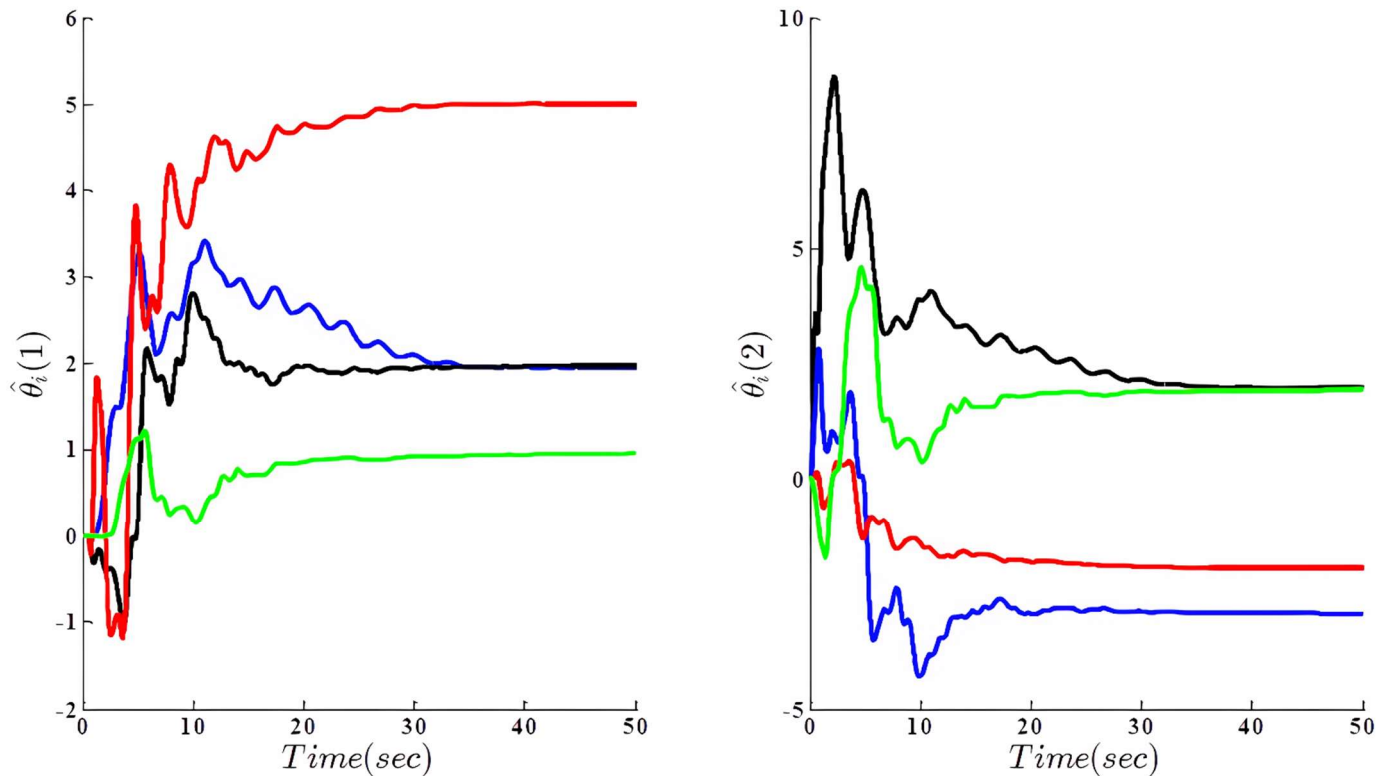


Figure 7. Convergences of the estimated parameter arrays $\hat{\theta}_i, i = 1, \dots, 4$, based on the PDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

Also, Figure 8 shows the simulation results of the adaptive controller based on the FDAC method with the control parameters $a = 5, b = 3, \mu = 0.2, \beta = 4, \gamma = 2$. Figure 9 shows the convergences of adaptive gain, *i.e.*, $c_i(t), i = 1, \dots, 4$. Also, Figures 10 and 11 show the convergence of the estimated parameters $\hat{\theta}_{0i}, \hat{\theta}_i, i = 1, \dots, 4$. Figures 9 and 10 illustrate the dynamic performance of the FDAC system by showing the convergence characteristics

of the adaptive gain and the estimated parameter arrays over time. Figure 9 presents the convergence of the adaptive gain utilized by the FDAC algorithm. Initially, it is observed that some fluctuations in the adaptive gains as the agents start to adapt to the system dynamics and the influence of the leader. However, after approximately 5 s, the adaptive gains stabilize and converge to constant values. This rapid convergence indicates the FDAC system's effectiveness in quickly adjusting the control parameters in response to the varying dynamics of the environment. The stabilization of the adaptive gains reflects the system's ability to achieve an appropriate balance between responsiveness and stability, ensuring that agents can effectively coordinate their actions while adapting to real-time conditions. Figure 10 depicts the convergence of the estimated parameter arrays for the agents. Initially, the parameter estimates exhibit high oscillations as the agents engage in adaptive processes, reflecting the ongoing adjustments to their understanding of the environment. After about 30 s, these oscillations diminish, and the estimated parameters converge towards constant values. High oscillations during the early adaptation phase suggest that the agents are actively refining their estimates based on local and neighbor information. The eventual convergence to stable parameter values illustrates the FDAC system's ability to facilitate cooperative learning and parameter estimation among agents, leading to coherent collective behavior. Together, Figures 9 and 10 demonstrate the robustness of the FDAC algorithm in achieving convergence in both adaptive gains and parameter estimates. The quick stabilization of adaptive gains, coupled with the eventual convergence of the parameter arrays, underscores the effectiveness of the fully-distributed approach in managing uncertainty and enabling agents to work in harmony within a multi-agent system.

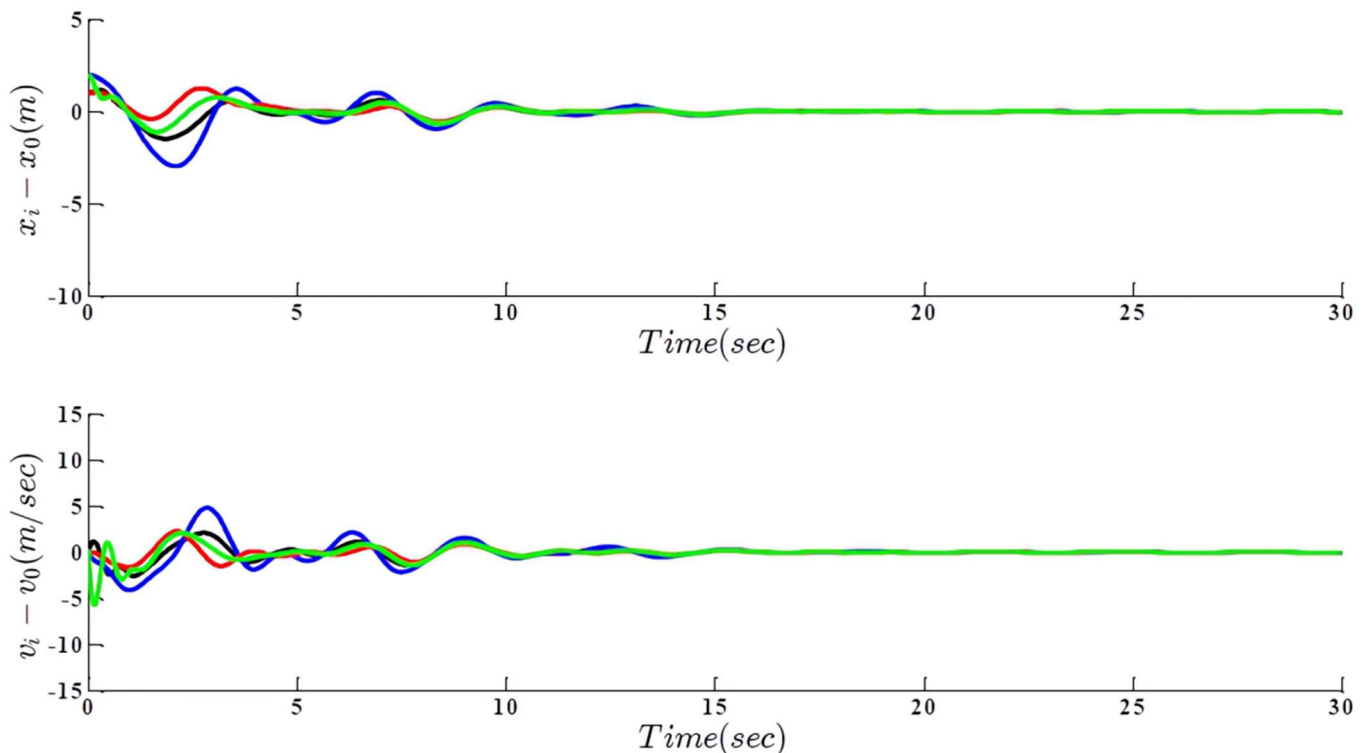


Figure 8. Trajectories of $x_i - x_0$ and $v_i - v_0$, $i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

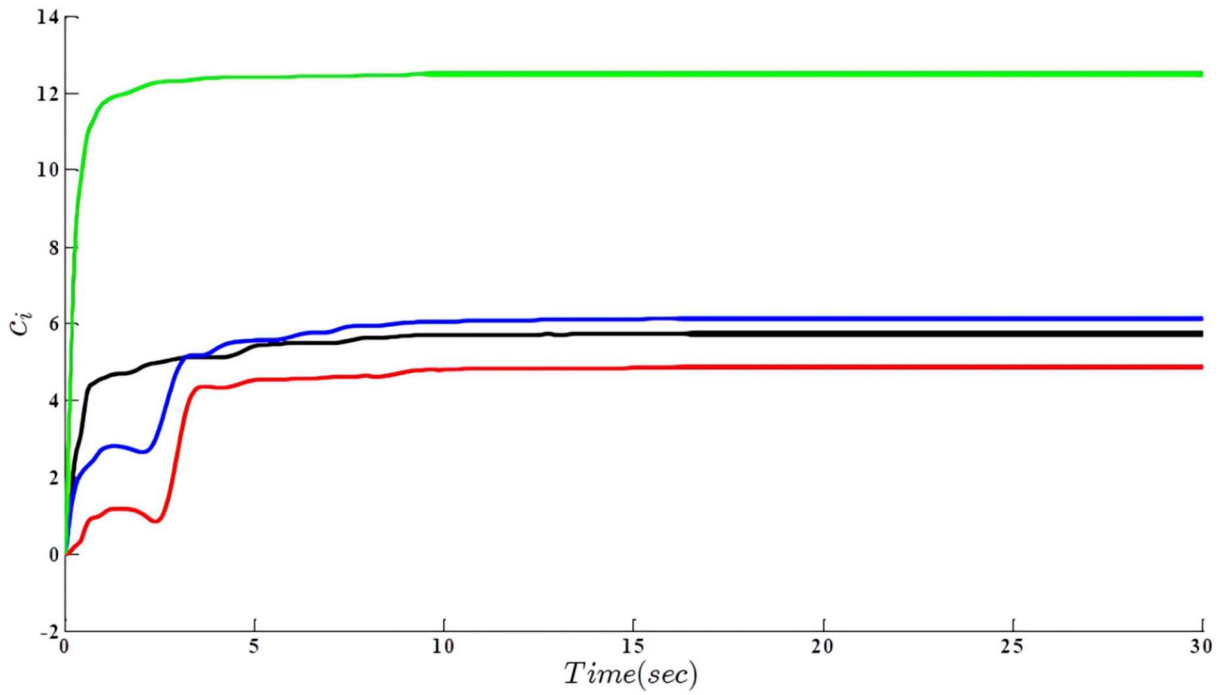


Figure 9. Convergence of the adaptive gain, $c_i(t), i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

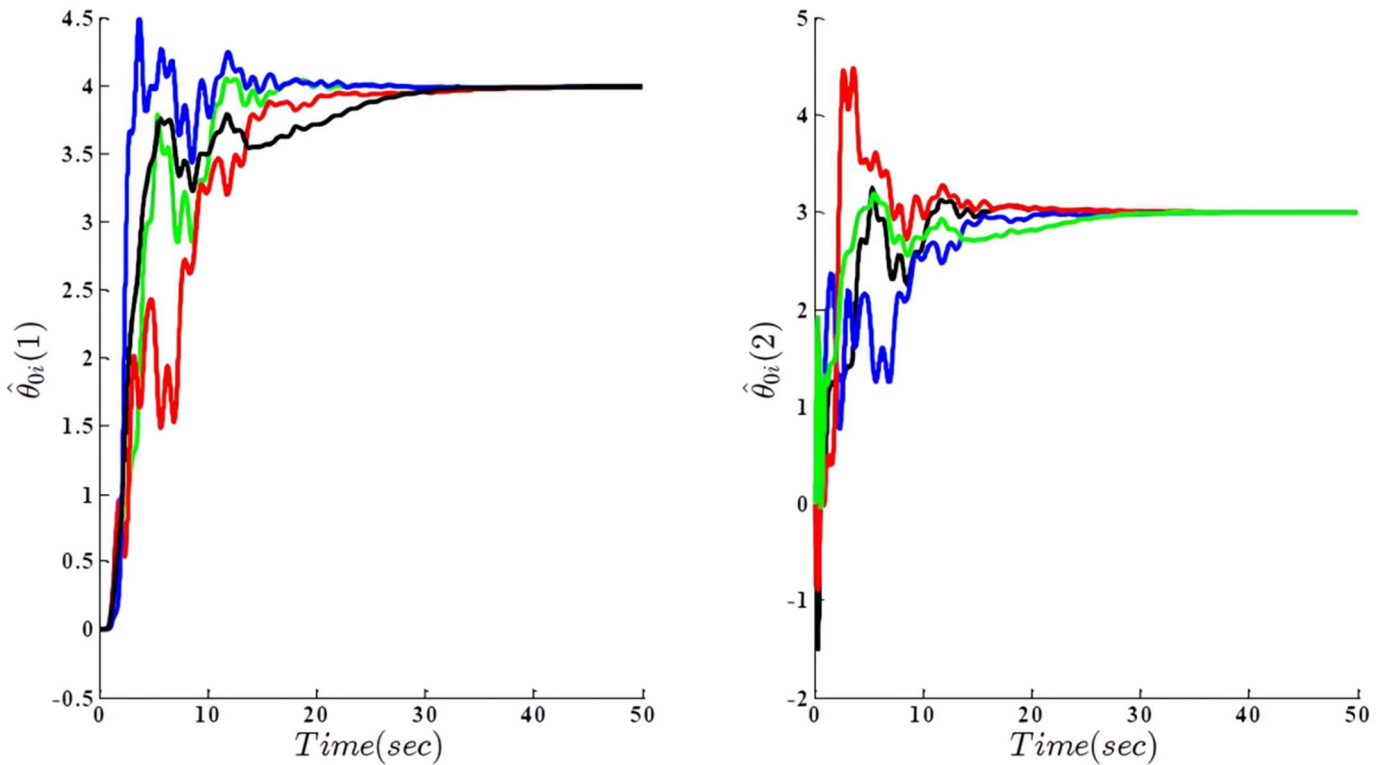


Figure 10. Convergences of the estimated parameter arrays $\hat{\theta}_{0i}, i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

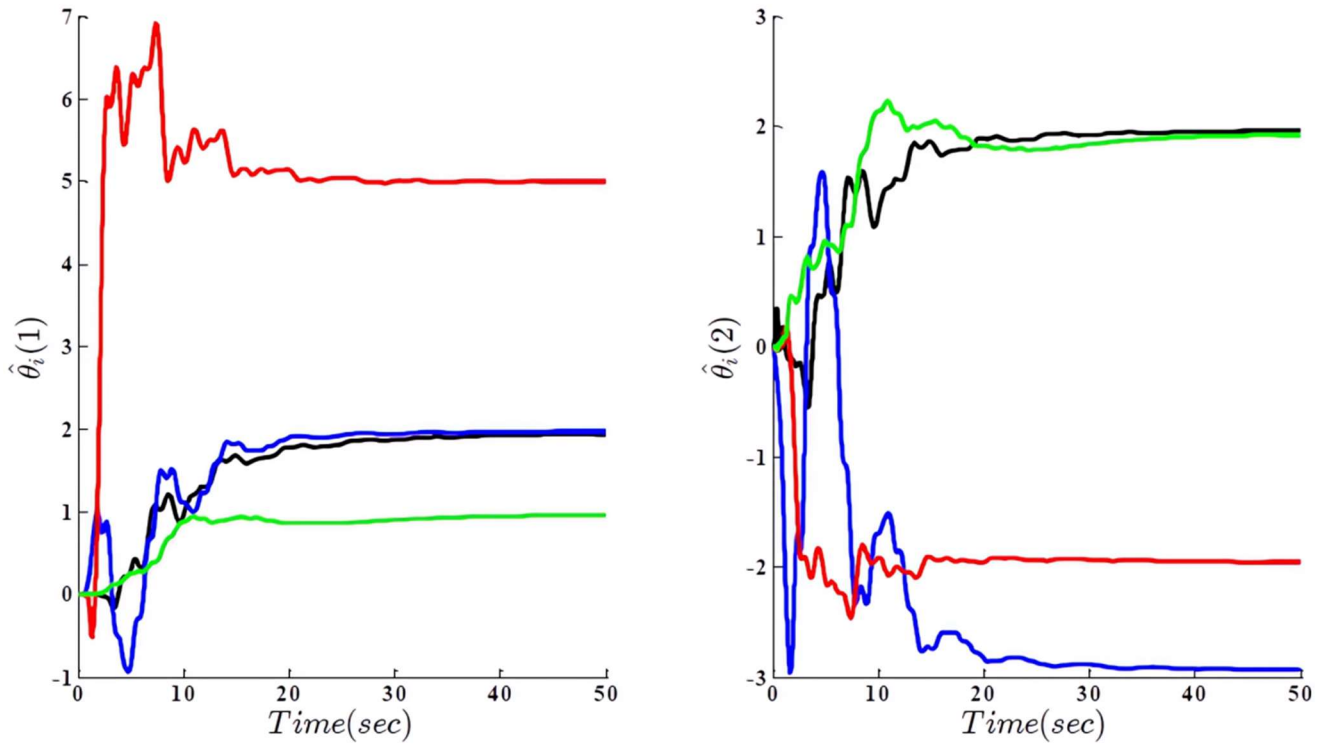


Figure 11. Convergences of the estimated parameter arrays $\hat{\theta}_i, i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

Example 2. In this example, the relative position measuring considered being unavailable for any agent; but each agent can measure its absolute position and communicate this with each other. Due to data transmitting speed limitations, each agent receives information from others with a time delay. Consequently, the relative position of each agent relative to its neighbors is given as

$$e_{xi}(t, \tau) = \sum_{j=0}^N a_{ij}(x_i(t) - x_j(t - \tau)), \quad i = 1, 2, \dots, N \quad (40)$$

The expression (40) can be used in proposed PDAC and FDAC algorithms. The time delay affected the closed loop system's stability and tracking performance. So, this example shows numerically the effect of time delay on tracking performance and, thus, on the performance of relative positioning with respect to absolute positioning. Figures 12 and 13 show the simulation results of the PDAC and the FDAC methods with controller parameters similar to Example 1 and also $\tau = 0.50$ s, respectively.

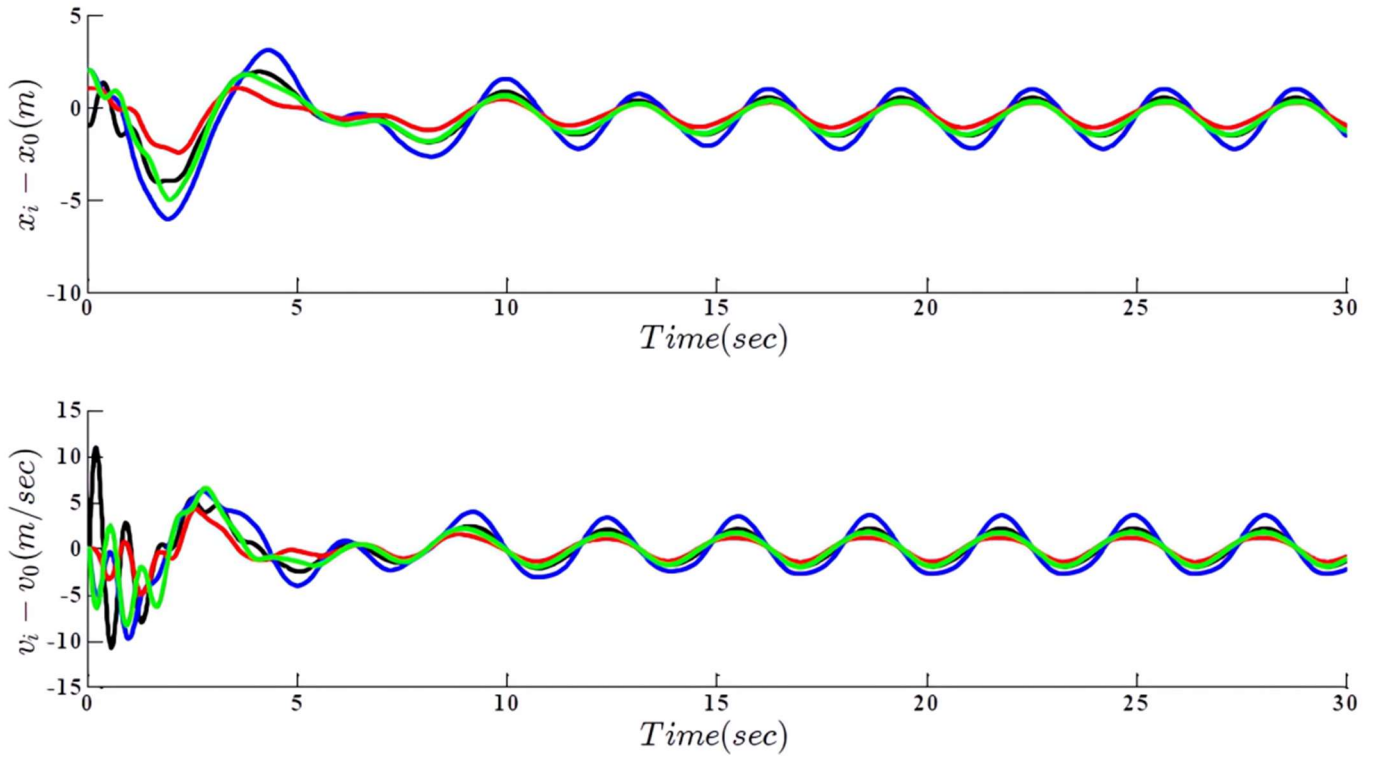


Figure 12. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the PDAC method with delayed communication, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent agent # 4).

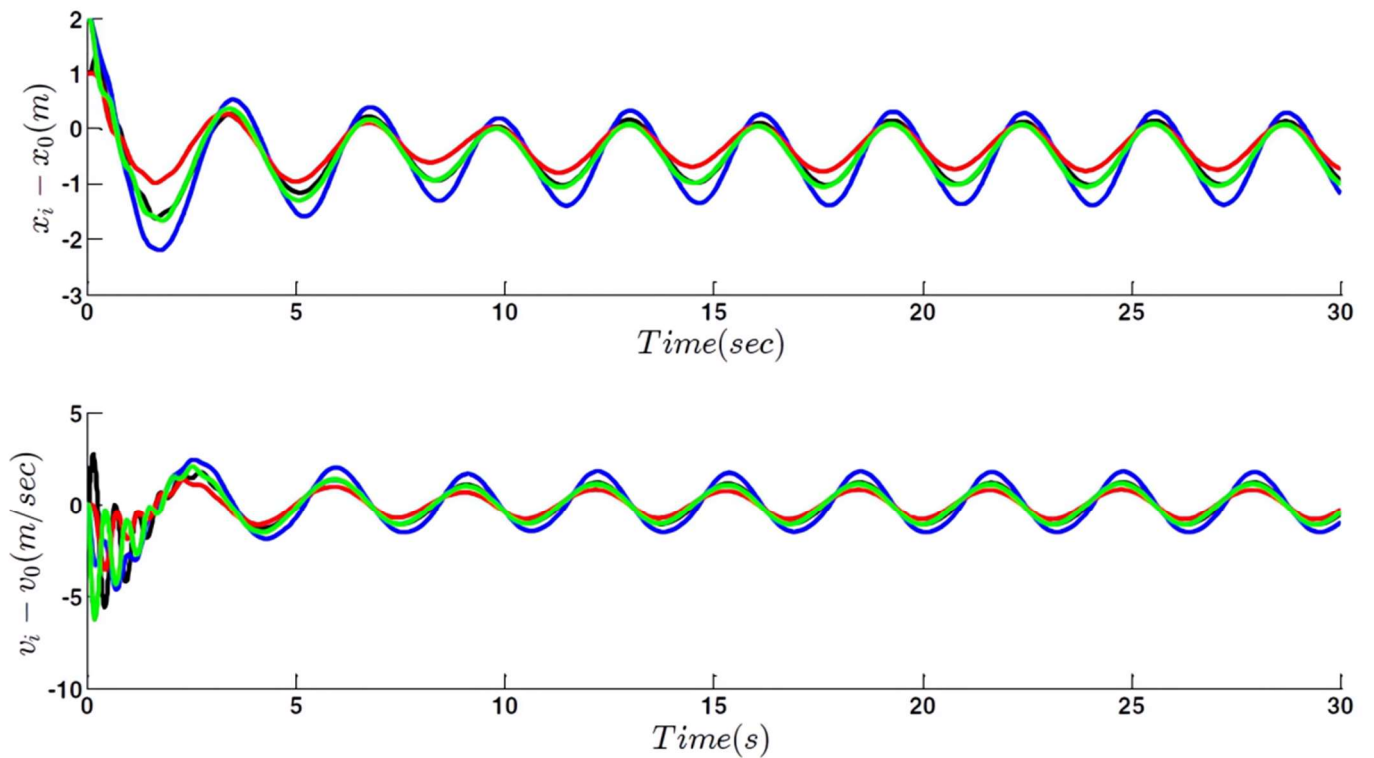


Figure 13. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the FDAC method with delayed communication, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent agent # 4).

In the Examples 1 and 2, the structure of the u_0 and $f_i, i = 1, \dots, 4$ is considered to be known. In the Examples 3 and 4, the structure of the u_0 and $f_i, i = 1, \dots, 4$ is unknown, but is assumed slow-varying. Then, it is reasonable to use the basis function $\phi_0 = \phi_i = 1.0$ in the formulation of the proposed algorithms.

Example 3. (performance evaluation of the controllers for slow-varying discontinuous disturbances): In this example, the leader acceleration is assumed to be a square function, $u_0(t) = 5.0 \times \text{square}(0.05t)$ as shown in Figure 14. For analysis of the simplicity of controller performance, the $f_i(t) = 0, i = 1, \dots, 4$ assumed. Figure 15 shows the simulation results of the PDAC method with controller parameters $a = 5, b = 2, \alpha = 10, \beta = 4, \gamma = 2$. Also, Figure 16 shows the result of the FDAC method with the controller parameters $a = 5, b = 2, \mu = 0.1, \beta = 4, \gamma = 2$.

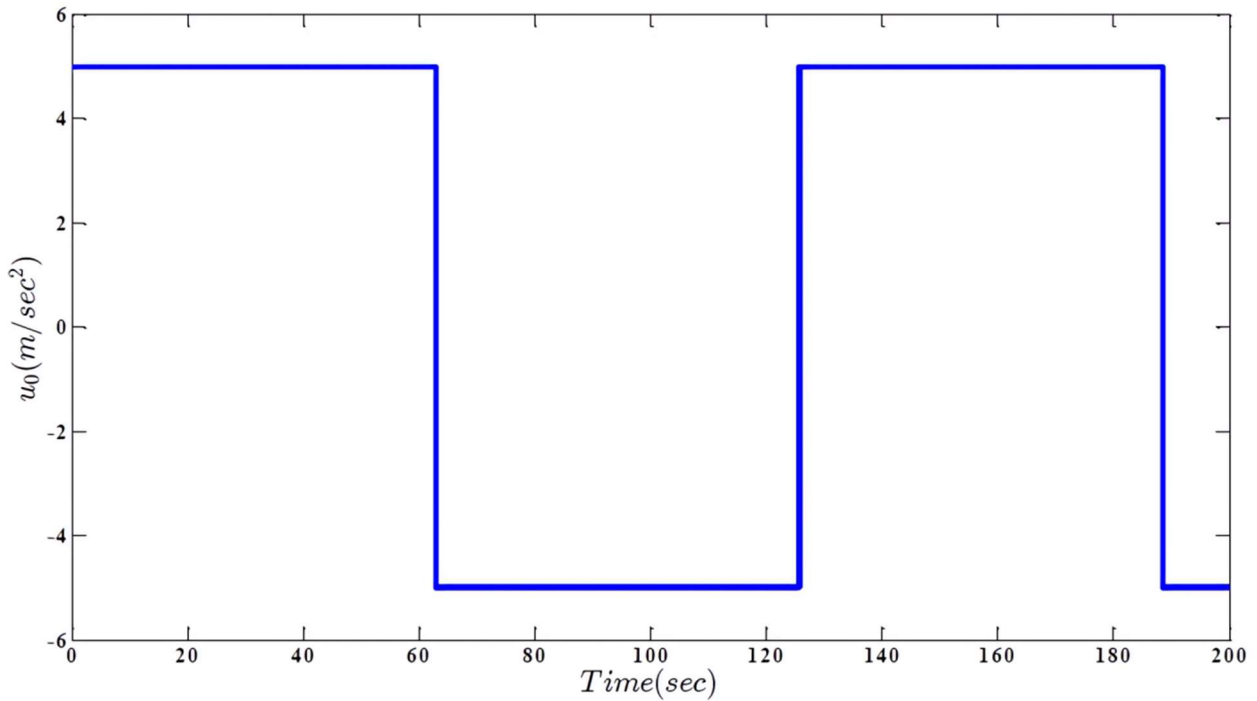


Figure 14. Leader acceleration, $u_0 = 5.0 \times \text{square}(0.05t)$ versus time.

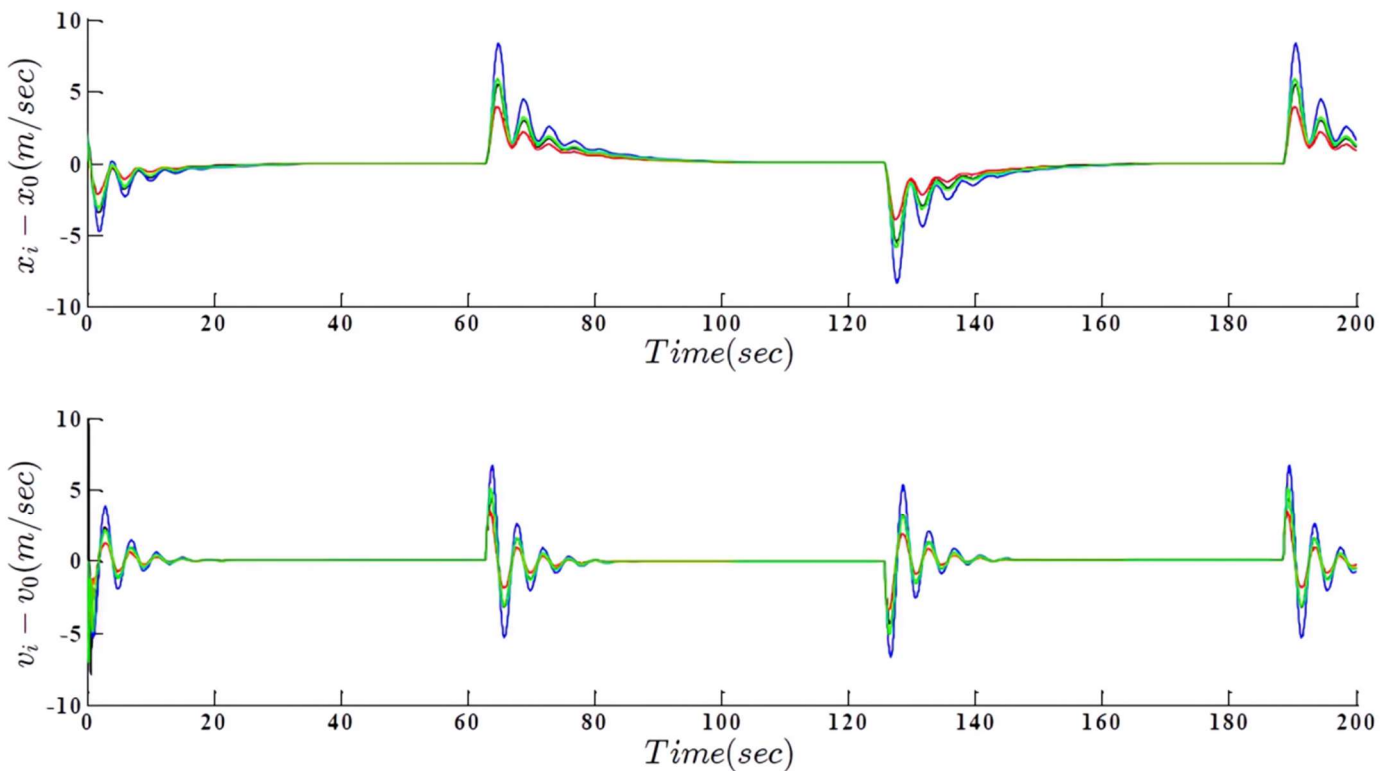


Figure 15. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the PDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

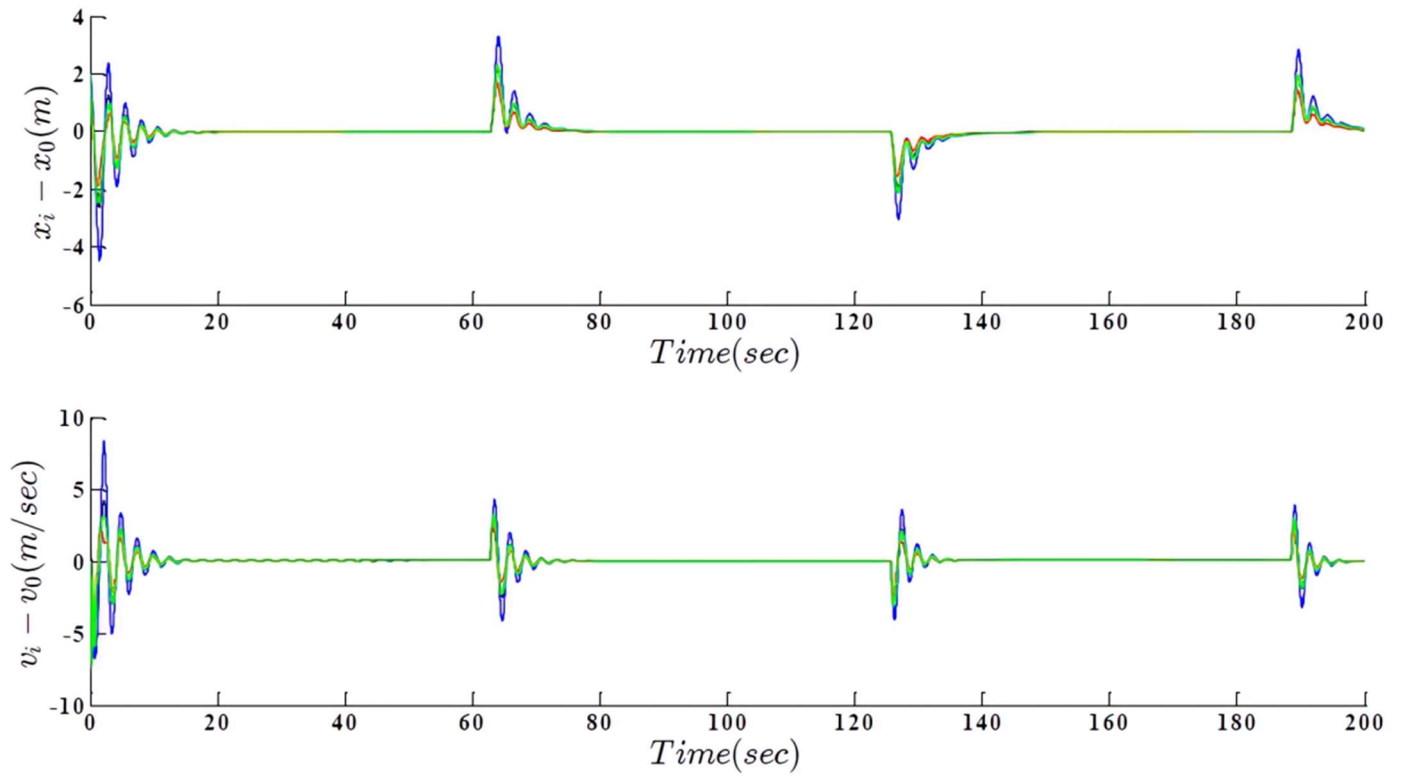


Figure 16. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent agent # 4).

Example 4. (performance evaluation of the controllers for fast-varying discontinuous disturbances): Similar to Example 3, the unknown dynamics of each follower, $f_i(t)$ assumed to be zero. Also the leader acceleration is considered to be $u_0(t) = 4.0 \times \text{sawtooth}(0.1t)$ as shown in Figure 17. Figures 18 and 19 show the simulation results of the PDAC and the FDAC methods with the controller parameters which are similar to Example 3.

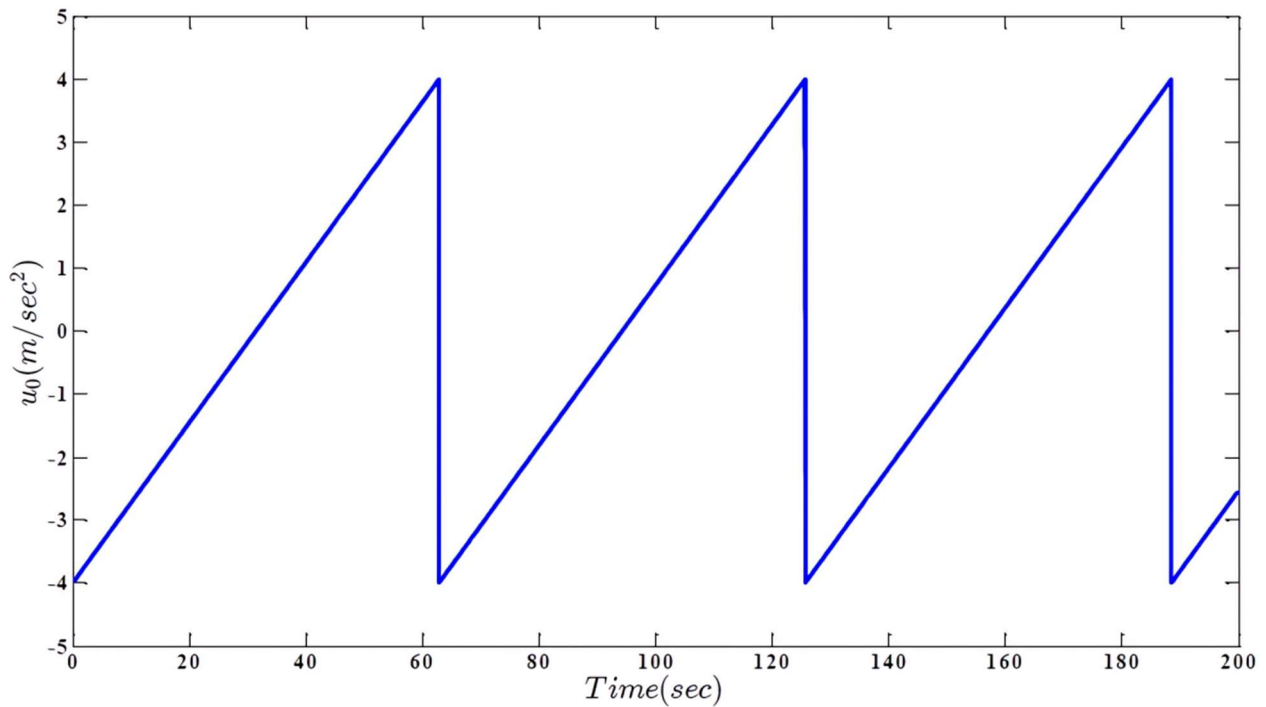


Figure 17. Leader acceleration, $u_0(t) = 4.0 \times \text{sawtooth}(0.1t)$ versus time.

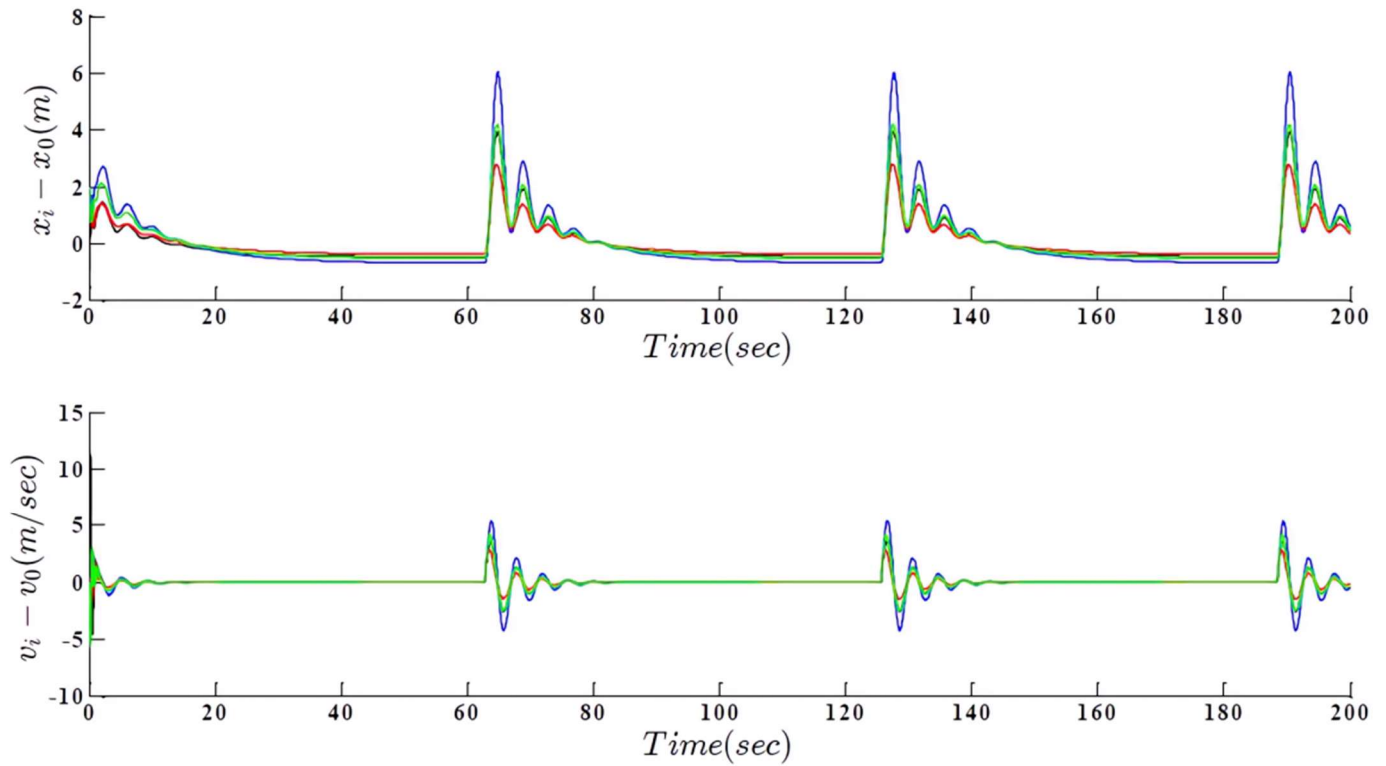


Figure 18. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the PDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

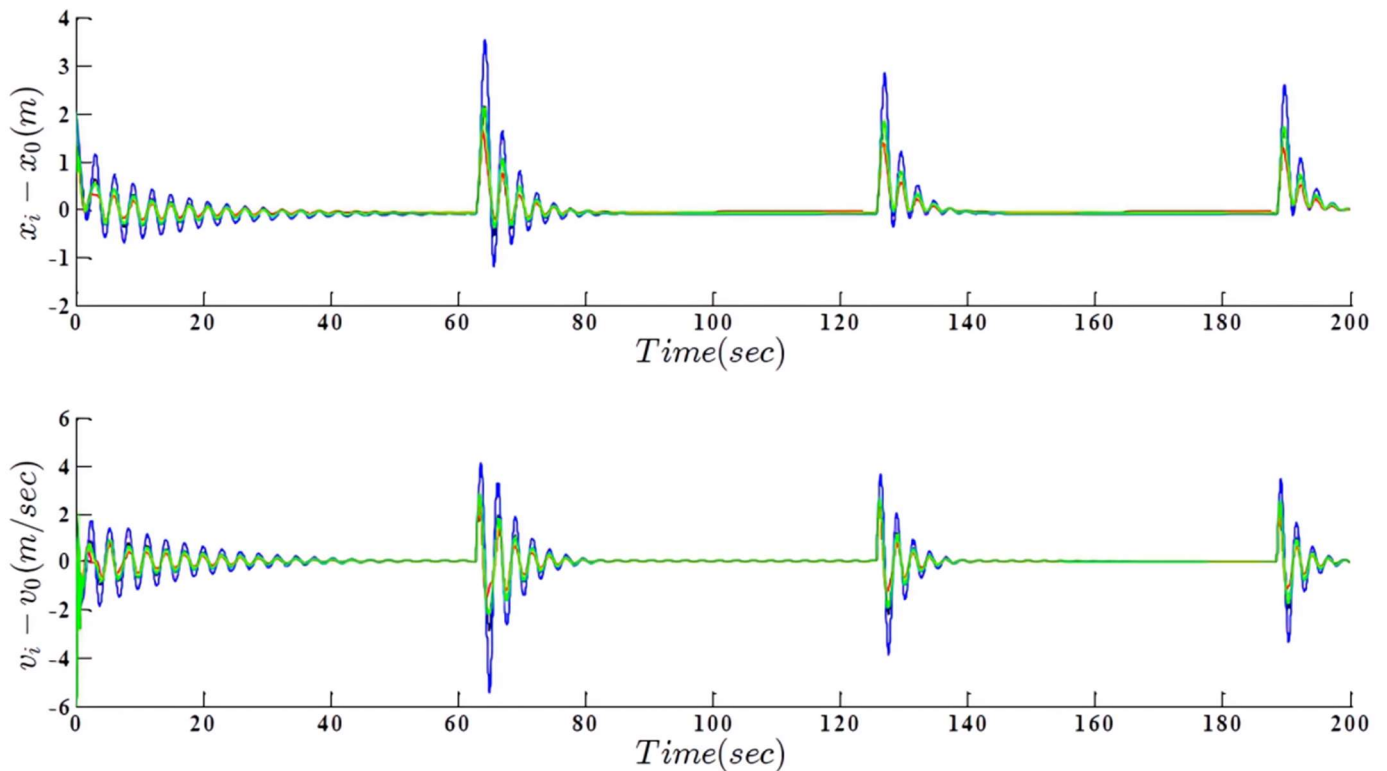


Figure 19. Trajectories of $x_i - x_0$ and $v_i - v_0, i = 1, \dots, 4$, based on the FDAC method, (Black: follower agent # 1, Blue: follower agent # 2, Red: follower agent # 3, Green: follower agent # 4).

4.2. Discussions

The present study focused on establishing the theoretical foundations underpinning the FDAC and PDAC methods through Lyapunov-based analysis and stability proofs. The theoretical theorems and their associated proofs provide a robust basis for understanding the performance of the FDAC and PDAC algorithms in the context of the

undirected graph communication topology. While the current theoretical analysis emphasizes the collective system behavior, it should be noted that a more detailed examination of individual components could further enhance the understanding of the dynamics involved as follows. As the proposed strategies utilize global information about the graph topology differently in the FDAC and PDAC methods, it can be concluded that an in-depth theoretical study on how these methods interact with various individual graph structures and their respective equations would be beneficial for strengthening the obtained results. This could involve analyzing how specific graph properties (e.g., connectivity, degree distributions) affect the performance of each algorithm on a granular level. According to simulations, it can be recognized that investigating various graph configurations, analyzing the control strategies concerning distinct types of inter-agent communication failures, and understanding how these factors can be integrated into the theoretical framework lead to a more comprehensive understanding. The numerical simulations conducted in this study involve a scenario with four follower agents and one leader governed by a defined communication topology represented by the matrix H . The aim is to evaluate the effectiveness of the FDAC and PDAC methods in facilitating consensus among agents with second-order dynamics, under varying conditions. In the Example 1, both the leader's acceleration and the unknown nonlinear dynamics of each follower agent are predetermined. For this simulation, it was assumed that the structure of the control input u_0 and the follower dynamics are known, allowing utilizing basis functions appropriately. Figure 4 illustrates the simulation results of the PDAC method implemented without an adaptive term compensator. This setup demonstrates the initial performance of the followers in tracking the leader's acceleration, highlighting the steady-state behavior after some transient response. Figure 5 shows the results when the adaptive controller is employed based on the PDAC method. The incorporation of the adaptive compensator significantly enhances the convergence behavior, as the follower agents can adjust their control efforts to account for discrepancies in the leader's acceleration. Figures 6 and 7 display the convergence of the estimated parameter arrays, showcasing how quickly and accurately the follower agents are able to adapt their parameters to match the leader's dynamics over time. Figure 8 presents the simulation results for the adaptive controller based on the FDAC method, where all controller parameters are assumed to be known. This visualization underscores the decentralized approach and its ability to maintain consensus among agents. Figure 9 depicts the convergence of the adaptive gain for the FDAC method, emphasizing the method's robustness in adjusting gains in real-time. Figures 10 and 11 highlight the convergence of the estimated parameters, further supporting the effectiveness of the FDAC in achieving the desired control objectives. In the Example 2, it was explored a more complex scenario where none of the followers can directly measure their relative positions. Instead, each agent can only measure its absolute position and communicate this information with its neighbors. Due to limitations in data transmission speeds, time delays are introduced, causing each agent to receive information from others with a known delay $\tau = 0.50$ s. These time delays' implications on stability and tracking performance are rigorously analyzed. Figures 12 and 13 display the simulation results for the PDAC and FDAC methods under this framework, illustrating how delays influence the system's response and overall effectiveness in maintaining a coordinated state. For Examples 3 and 4, it was assumed that the structures of u_0 and the follower dynamics are unknown, but we posit they are slow-varying. This allows us to apply the basis functions effectively to approximate the dynamics. In Example 3, the leader's acceleration is modeled as a square function, with the follower dynamics assumed zero for simplifying the analysis. Figure 15 exhibits the simulation results for the PDAC method under these assumptions, showcasing its performance against the leader's trajectory. FDAC in Unknown Conditions: Similarly, Figure 16 displays the results for the FDAC method, reaffirming its capability to perform effectively even when the agent dynamics are subject to uncertainty. Figures 18 and 19 delve deeper into the results of Examples 3 and 4, comparing the performance metrics of both control strategies. The findings indicate that the FDAC method consistently exhibits superior tracking performance compared to the PDAC method. The comparative analysis between FDAC and PDAC reveals critical insights:

Decentralization: The FDAC method's decentralized nature allows for a more flexible and robust response to disturbances and uncertainties, as it does not rely on global information.

Adaptive Gain Determination: The FDAC method employs an adaptive mechanism for controller gain determination, thus optimizing the control responses in real-time. In contrast, the PDAC method utilizes a fixed gain configuration, which can limit performance in dynamically changing environments.

The simulation results affirm that both methods provide viable solutions for leader-follower consensus. However, the FDAC method demonstrates a marked advantage in adaptability and robustness, particularly in scenarios involving unknown dynamics and communication delays.

According to the results obtained from the Examples 3 and 4, the FDAC method shows better tracking performance relative to the PDAC method. In summary, the main reasons for justification of this claim are:

- FDAC is completely decentralized, while PDAC is not.
- Adaptively determination of the controller gain in the FDAC method while in the PDAC method, the controller gain is chosen primitively.

5. Conclusions

The objective of this study was to solve the problem of distributed leader-follower consensus of a group of agents with the second-order dynamics under the undirected graph communication topology. When the leader's acceleration is not communicated to each follower, and the follower agents have some unknown dynamics in their intrinsic structure, implementing the proposed method practically results in delayed and faulty network control. To address this problem, a linear regression model is assumed for the leader's acceleration and the agents' unknown dynamics. Using this model, Lyapunov-based adaptive control algorithms are developed to manage the network of agents despite communication loss and modeling uncertainties. The presented study describes two multi-agent control strategies called FDAC and PDAC systems. In the first method, the followers do not have any a priori information about the communication graph, while in the second method, some information about the eigenvalues of the communication graph is available. Illustrative simulations were conducted to show the merits and efficiencies of the proposed algorithms relative to the case in which enough communication between leader and followers is available.

Future Studies

The future research associated with the present article can be summarized as follows:

- Future research could explore the leader-follower consensus problem in systems with higher-order dynamics, which may introduce additional complexities in control design and stability analysis.
- Investigating the effects of external noise and disturbances on the leader-follower consensus system would be crucial. Developing robust control strategies that maintain consensus under such conditions could enhance the practical applicability of the proposed methods.
- Future studies could aim to develop algorithms that allow for decentralized information gathering regarding the communication graph, enabling followers to adapt more dynamically to changes in the network topology.
- Examining the scalability of both FDAC and PDAC methods in larger multi-agent systems could provide insights into the limitations and performance trade-offs of the proposed algorithms as the number of agents increases.
- Implementing the proposed algorithms in real-world multi-agent systems, such as autonomous vehicles or robotic networks, would validate their performance and adaptability in practice, highlighting any practical challenges.
- Investigating the dynamics of multi-leader systems where multiple leaders influence their respective follower groups could broaden the applicability of the consensus algorithms, allowing for more complex coordination tasks.
- Analyzing the impact of time-varying communication topologies on consensus convergence could yield more flexibility and robustness in dynamic environments.

Author Contributions

B.T.: Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing—Original Draft, Writing—Review & Editing, Supervision. M.R.H.: Methodology, Software, Validation, Formal analysis, Investigation, Writing—Original Draft, Writing—Review & Editing, Visualization.

Ethics Statement

The authors of the article have adhered to the terms and conditions of ethical publications.

Informed Consent Statement

This study does not require human data or tests.

Funding

This research was not supported by any funding agency and was conducted based on authors personal interest.

Declaration of Competing Interest

This study includes not potential conflict of interest.

References

- Xu B. An Integrated Fault-Tolerant Model Predictive Control Framework for UAV Systems. Doctoral Dissertation, University of Victoria, Victoria, BC, Canada, 2024.
- Jadbabaie A, Lin J, Morse AS. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Autom. Control.* **2003**, *48*, 988–1001.
- Xu C, Deng Z, Liu J, Huang C, Hang P. Towards Safe and Robust Autonomous Vehicle Platooning: A Self-Organizing Cooperative Control Framework. *arXiv* **2024**, arXiv:2408.09468.
- Hu Y, Lin X, Peng K. Distributed Consensus Filtering Over Sensor Networks with Asynchronous Measurements. *Int. J. Adapt. Control. Signal Process.* **2024**, *in press*. doi:10.1002/acs.3924.
- He T, Goh HH, Yew WK, Kurniawan TA, Goh KC. Enhancing multi-agent system coordination: Fixed-time and event-triggered control mechanism for robust distributed consensus. *Ain Shams Eng. J.* **2024**, *103105*, *in press*. doi:10.1016/j.asej.2024.103105.
- Thattai K. Hierarchical Multi-Agent Deep Reinforcement Learning Based Energy Management System for Active Distribution Networks. UNSW Research. Doctoral Dissertation, UNSW Sydney, Kensington, Australia, 2024.
- Munir M, Khan Q, Ullah S, Syeda TM, Algethami AA. Control Design for Uncertain Higher-Order Networked Nonlinear Systems via an Arbitrary Order Finite-Time Sliding Mode Control Law. *Sensors* **2022**, *22*, 2748.
- Wen B, Huang J. Leader-Following Formation Tracking Control of Nonholonomic Mobile Robots Considering Collision Avoidance: A System Transformation Approach. *Appl. Sci.* **2022**, *12*, 1257.
- Ren Y, Liu S, Li D, Zhang D, Lei T, Wang L. Model-free adaptive consensus design for a class of unknown heterogeneous nonlinear multi-agent systems with packet dropouts. *Sci. Rep.* **2024**, *14*, 23093.
- Yi Z, Jiang M. Attitude collaborative control strategy for space gravitational wave detection. *Syst. Sci. Control. Eng.* **2024**, *12*, 2368661.
- Martinelli A, Aboudonia A, Lygeros J. Interconnection of (Q, S, R)-Dissipative Systems in Discrete Time. 2024. Available online: https://www.researchgate.net/profile/Andrea-Martinelli-7/publication/381671295_Interconnection_of_QSR-Dissipative_Systems_in_Discrete_Time/links/667ab7378408575b838a6717/Interconnection-of-Q-S-R-Dissipative-Systems-in-Discrete-Time.pdf (accessed on 3 August 2024).
- Chen R-Z, Li Y-X, Hou ZS. Distributed model-free adaptive control for multi-agent systems with external disturbances and DoS attacks. *Inf. Sci.* **2022**, *613*, 309–323.
- Zhang W, Yan J. Finite-Time Distributed Cooperative Guidance Law with Impact Angle Constraint. *Int. J. Aerosp. Eng.* **2023**, *2023*, 5568394.
- Li Y, Zhao H, Wang C, Huang J, Li K, Lei Z. A Novel Fuzzy Adaptive Consensus Algorithm for General Strict-feedback Nonlinear Multi-agent Systems Using Backstepping Approaches. In Proceedings of the 2023 42nd Chinese Control Conference (CCC), Tianjin, China, 24–26 July 2023; pp. 5829–5836.
- Kim Y, Ye H, Lim S, Kim SK. Observer-Based Nonlinear Proportional–Integral–Integral Speed Control for Servo Drive Applications via Order Reduction Technique. *Actuators* **2023**, *13*, 2.
- Kenyeres M, Kenyeres J. Distributed Network Size Estimation Executed by Average Consensus Bounded by Stopping Criterion for Wireless Sensor Networks. In Proceedings of the 2019 International Conference on Applied Electronics (AE), Pilsen, Czech Republic, 10–11 September 2019; pp. 1–6.
- Wang Y, Cheng Z, Xiao M. UAVs' Formation Keeping Control Based on Multi-Agent System Consensus. *IEEE Access* **2020**, *8*, 49000–49012.
- Zhang Y, Zhao Z, Ma S. Adaptive neural consensus tracking control for singular multi-agent systems against actuator attacks under jointly connected topologies. *Nonlinear Dyn.* **2024**, doi:10.1007/s11071-024-10418-z.
- Boyd S, Ghaoui LE, Feron E, Balakrishnan V. *Linear Matrix Inequalities in System and Control Theory*; SIAM: Philadelphia, PA, USA, 1994.
- Slotine JJE, Li W. *Applied Nonlinear Control*; Prentice Hall: Upper Saddle River, NJ, USA, 1991.
- Marino R, Tomei P. *Nonlinear Control Design-Geometric, Adaptive and Robust*; Prentice Hall: London, UK, 1995.
- Gu H, Lü J, Lin Z. On PID control for synchronization of complex dynamical network with delayed nodes. *Sci. China Technol. Sci.* **2019**, *62*, 1412–1422.