Iterative learning control of multi-agent systems with random noises and measurement range limitations

Chen Liua, Dong Shenb and JinRong Wangb

aCollege of Information Science and Technology, Beijing University of Chemical Technology, Beijing, People’s Republic of China; bDepartment of Mathematics, Guizhou University, Guiyang, Guizhou, People’s Republic of China

ABSTRACT
In this paper, the iterative learning control is introduced to solve the consensus tracking problem of a multi-agent system with random noises and measurement range limitation. A distributed learning control algorithm is proposed for all agents by utilising its nearest neighbour measured information from previous iterations. With the help of the stochastic approximation technique, we first establish the consensus convergence of the input sequences in almost sure sense for fixed topology as the iteration number increases. Then, we extend the results to switching topologies case which is dynamically changing along the time axis. Illustrative simulations verify the effectiveness of the proposed algorithms.

ARTICLE HISTORY
Received 6 June 2018
Accepted 29 April 2019

KEYWORDS
Multi-agent system; iterative learning control; measurement saturation; communication noises; time-switching topologies

1. Introduction

Iterative learning control (ILC) is designed to improve the control performance by utilising past control experience for batched systems. Such control methodology generates the input signal for the current iteration based on input signals and tracking errors of previous iterations. ILC attracts the scholars and engineers by its simple but effective control structure, which can gradually improve the tracking performance along the iteration axis. The concept of ILC was proposed in Arimoto, Kawamura, and Miyazaki (1984) and greatly developed over the past three decades in both theoretical analysis and practical applications. Readers can refer to the recent surveys (Ahn, Chen, & Moore, 2007; Bristow, Tharayil, & Alleyne, 2006; Shen, 2018a; Shen & Wang, 2014; Xu, 2011) for more details. The advantage of ILC is that it can render the system to follow the desired reference trajectory precisely without using full dynamic model information. ILC requires less knowledge about the system dynamics and relatively less computational effort. Thus, it has been applied to solve the tracking problem with various issues such as quantised control (Zhang & Li, 2017; Zhang & Shen, 2018), event-triggered control (Xiong, Yu, Patel, & Yu, 2016), control under data dropouts (Shen, 2018b; Shen & Xu, 2017), adaptive switching learning strategy (Ouyang, Zhang, & Gupta, 2006), and iteration-varying lengths (Shen & Xu, 2019a, 2019b; Shen, Zhang, Wang, & Chien, 2016).

A multi-agent system (MAS) consists of a group of agents, which is required to complete group-level cooperative tasks. The consensus problem for the MAS has attracted considerable research interest due to its wide applications in many practical areas. In the literature, there are many results devoted to studying the distributed consensus problem, see Chen, Lu, Han, and Yu (2011) and Han, Lu, and Chen (2013) for cluster consensus, Liu, Xie, and Wang (2012), Ni and Cheng (2010) and Cheng, Hou, Tan, Lin, and Zhang (2010) for leader-following consensus, and Li and Zhang (2010), Liu and Chen (2008) for robust consensus. These mentioned results mainly contribute to the consensus problem of the MAS in a traditional control framework; that is, the involved systems mainly evolve along the time axis and generate a time-domain-based asymptotical property.

ILC was early employed for the MAS by Ahn and Chen (2009), where the communication graph is of fixed-type. Later, the learning consensus has attracted rapid developments such as Meng, Jia, Du, and Yu (2013), Yang and Xu (2012) and Shen and Xu (2018) due to its distinct difference from the conventional framework that the MAS is dynamically dependent on two independent variables: time instant and iteration number.
In particular, when using ILC, both time and iteration axes should be taken into account simultaneously for deriving the asymptotical property. Moreover, in Shen and Xu (2018), the output constraint was taken into account that the output of each agent should locate in a given bound while converging to the leader’s trajectory. However, we note from Meng et al. (2013), Yang and Xu (2012), Ahn and Chen (2009) and Shen and Xu (2018) that all of these results depends on the precision information exchange; that is, the received neighbour information is without any noises and quality loss. This observation motivates us to address the learning consensus with communication noise and measurement range limitation. In particular, the communication noise is inevitable in wireless networks. We assume that each agent measures its neighbour information using its own devices that certain measurement range limitation arises in these circumstances.

For the measurement range limitation, saturation is a common phenomenon for mechanic systems due to various range limitations in a device. The input saturation has been considered in a large amount of literature. The paper Xu, Tan, and Lee (2004) considers the input saturation problem and employs the composite energy function method to conduct convergence analysis. In Wei, Quan, and Cai (2016), the additive-state-decomposition-based ILC method is proposed to solve the tracking problem for a class of nonlinear systems with input saturation. We note that the output saturation exists in many practical areas but are less considered in the ILC field. Inspired by Shen and Zhang (2017), we notice that the connection between measured data and its original information can be held by a random variable. This technique is expected to be effective in dealing with learning consensus of a MAS with measurement range limitation and random noises.

The studies on ILC for MAS are usually under a fixed communication topology, where a topology indicates the information exchange structure among a group of agents. However, in many practical applications, the communication topologies among agents can be dynamically switching in a given set of topologies. Therefore, it is significant to develop methods for the switching topology case. In Yang, Xu, Li, and Shen (2017, p. 27), the iteration-varying topology was strongly connected along the iteration axis and at least one of the follower agents can obtain the leader's information. A D-type distributed learning rule was proposed to ensure consensus tracking performance. However, the results of ILC for MAS with switching topologies are far from complete; therefore, we will consider both fixed and the time-switching topology cases and provide the Lyapunov function-based analysis technique in the presence of random noises.

In this paper, we study the ILC for a class of MAS with both fixed and time-switching topologies using lower quality data. In particular, the output of each agent is measured with range limitation and random noise; that is, if the output which is involved with random noises is beyond the measurement range, the received information is saturated. To solve the asymptotical consensus problem in presence of the above randomness and nonlinearity, we introduce a decreasing gain sequence into the distributed update algorithms to suppress the influence of randomness asymptotically along the iteration axis. Distributed learning algorithms are proposed and strictly proved its effectiveness that the generated input sequence converges to the desired one almost surely.

The main contributions of this paper are as follows. First, a transform factor matrix is introduced to connect the measured information and the actual information exchanged in a MIMO MAS. This connection helps us understand the essential effect of the measured information in the convergence analysis. Second, we introduce a decreasing gain sequence in the updating law to suppress the effect of noises and ensure the almost sure convergence. Third, we extend the results from the fixed topology case to the time-switching topology case. Compared with existing results, we note that the main challenges in this paper include the coupling effect of randomness and nonlinearity as well as the information exchange over fixed and time-switching topologies under the ILC framework.

The rest of this paper is organised as follows. We give problem formulations of consensus tracking problem for a MAS in Section 2. In Section 3, we present the main result for fixed topology case, where the convergence condition is provided with strict analysis. In Section 4, we extend to the time-switching topology case. Illustrative simulations are given in Section 5. Section 6 concludes the paper. In the Appendix, we provide detailed proofs for main theorems.

Notations: $\mathbb{Z}_N = \{0, 1, \ldots, N\}$; the set of natural number is denoted by $\mathbb{N}$; the set of real numbers and the n-dimensional space are denoted by $\mathbb{R}$ and $\mathbb{R}^n$, respectively; $I$ denotes the identity matrix with required dimensions; $1$ denotes the column vector with required dimensions and all the elements equal to 1; $\text{diag}[d_1, d_2, \ldots, d_n]$ denotes a diagonal matrix with diagonal entries: $d_1, d_2, \ldots, d_n$ and the off-diagonal entries equal to 0; $\| \cdot \|$ denotes the Euclidean norm for vectors and the induced 2-norm for matrices; $\otimes$ denotes the Kronecker product; the superscript $T$ denotes the transpose of a vector or matrix; and $\mathbb{E}$ denotes the mathematical expectation of a random variable.

Graph Theory Nations: Let $\mathcal{G} = (\mathcal{V}(\mathcal{G}), \mathcal{E}(\mathcal{G}))$ be an nth order directed graph. The vertex set and the edge
set are denoted by $\mathcal{V}(\mathcal{G})$ and $\mathcal{E}(\mathcal{G})$, respectively. In addition, $\mathcal{V}(\mathcal{G}) = \{v_i : i \in G_i\}$ and $\mathcal{E}(\mathcal{G}) = \{(v_i, v_j) : v_i, v_j \in \mathcal{V}(\mathcal{G})\}$. If $(v_i, v_j)$ is an edge, it indicates that the vertex $v_j$ is able to receive the information from vertex $v_i$; that is, $v_j$ is a neighbour of $v_i$ or $v_i$ and $v_j$ are described to be adjacent. Moreover, the index set of all neighbours of $v_i$ is denoted by $\mathcal{N}_i = \{j : (v_i, v_j) \in \mathcal{E}(\mathcal{G})\}$. The graph $\mathcal{G}$ is described to have a spanning tree as if there is a special vertex which has paths to all other vertices, and the special vertex is said the root vertex. Moreover, $\mathcal{A} = [a_{ij}]$ is defined as the adjacency matrix of graph $\mathcal{G}$. $a_{ij} = 1$ if $(v_i, v_j) \in \mathcal{E}(\mathcal{G})$, otherwise $a_{ij} = 0$. The degree matrix and the Laplacian matrix are denoted by $\Delta$ and $\mathcal{L}$, respectively. The degree matrix $\Delta$ is defined as $\Delta = \text{diag}(\sum_{j \in \mathcal{N}_i} a_{ij}, \sum_{j \in \mathcal{N}_2} a_{2,j}, \ldots, \sum_{j \in \mathcal{N}_n} a_{n,j})$, and the Laplacian matrix $\mathcal{L}$ is defined as $\mathcal{L} = \Delta - \mathcal{A}$.

2. Problem formulation

In this section, we consider a MAS consisting of $n$ MIMO discrete-time agents, with the $j$th agent governed by the following linear time-variant system

$$\begin{align*}
x_{i,j}(t+1) &= A(t)x_{i,j}(t) + B(t)u_{k,j}(t), \\
y_{i,j}(t) &= C(t)x_{i,j}(t),
\end{align*}$$

(1)

where the iteration number and the agent number are denoted by $k \in \mathbb{N}_+$ and $j = 1, 2, \ldots, n$, respectively. $t \in \mathbb{Z}_N$ denotes different time instant of each iteration and $N$ is the trial length. The input vector, output vector and state vector are denoted by $u_{k,j}(t) \in \mathbb{R}^{s_i}$, $y_{i,j}(t) \in \mathbb{R}^{s_o}$ and $x_{i,j}(t) \in \mathbb{R}^{s_x}$, respectively, and $A(t) \in \mathbb{R}^{s_x \times s_x}$, $B(t) \in \mathbb{R}^{s_x \times s_i}$, and $C(t) \in \mathbb{R}^{s_o \times s_x}$ are system matrices with appropriate dimensions.

In this paper, we discuss three cases of the communication mechanism in MAS.

**Case 1 (Noisy output and precise transmission):** Assume the $i$th agent is in the neighbourhood of the $j$th agent. For the $i$th agent, it measures its output, involved with random measurement noise $v_{i,j}(t)$, using a range-limited sensor, and then broadcasts the measured output to its neighbour agents precisely. The communication mechanism between agents $i$ and $j$ is shown in Figure 1.

![Figure 1. Communication mechanism between agents in Case 1.](image)

The measured output for the $i$th agent is as follows

$$z_{k,i}(t) = \text{sat}(y_{k,j}(t) + v_{k,j}(t)),$$

where $\text{sat}(\cdot)$ is a saturation function denoting the measurement range limitation. For a vector $v = [v_1, v_2, \ldots, v_{s_o}]^T \in \mathbb{R}^{s_o}$, the saturation function is given as $\text{sat}(v) \triangleq [\text{sat}(v_1), \text{sat}(v_2), \ldots, \text{sat}(v_{s_o})]^T$, where $\text{sat}(v_i)$ is defined as follows:

$$\text{sat}(v_i) = \begin{cases} r_0, & v_i \geq r_0 \\ v_i, & -r_0 < v_i < r_0 \\ -r_0, & v_i \leq -r_0 \end{cases}$$

(2)

for some unknown positive constant $r_0 > 0$. If the actual value is beyond the range, then measured information is the boundary; that is, the sensor saturation guarantees that the output coupled with measurement noise will not exceed the measurement range.

**Case 2 (Precise output and noisy transmission):** For the $i$th agent, it measures its actual output using a range-limited sensor and then broadcasts to other agents through unreliable networks. Additive communication noise $\omega_{k,i}(t)$ exists during the transmission of $i$th agent’s output to the $j$th agent. The communication mechanism between agents $i$ and $j$ is shown in Figure 2.

The received output of the $i$th agent by the $j$th agent is described as follows:

$$z_{k,i}(t) = \text{sat}(y_{k,i}(t) + \omega_{k,i}(t)),$$

**Case 3 (Noisy output and noisy transmission):** This case is a combination of Cases 1 and 2. In other words, both measurement noise, $v_{k,i}(t)$, and communication noise, $\omega_{k,i}(t)$, are accounted for. The communication mechanism between agents $i$ and $j$ is shown in Figure 3.
The measured output by other agents (for example: the \( i \)th agent) is described by

\[
   z_{k,j}(t) = \text{sat}(y_{k,j}(t) + v_{k,j}(t)) + \omega_{k,j}(t).
\]

In this paper, we consider the case that only a subset of agents can know the desired trajectory and regard the desired trajectory as a virtual leader, which is indexed by vertex 0 in the graph. Therefore, the complete information flow can be denoted by \( \mathcal{G} = \{V(\mathcal{G}) \cup \mathcal{E}(\mathcal{G})\} \).

For the \( j \)th agent, the control objective is to design a distributed ILC scheme such that the input sequence \( u_{kj}(t), t \in \mathbb{Z}_{N-1} \), drives the \( j \)th agent to track the desired trajectory \( y_d(t) \) precisely as the iteration number goes to infinity, where the input sequence is generated using available neighbours’ information \( z_{k,j}(t) \) (in Case 1) or \( z_{k,j,i}(t) \) (in Cases 2 and 3), \( \forall i \in \mathcal{N}_j \).

The following assumptions are needed for MAS (1).

**Assumption 2.1:** For the \( j \)th agent, the initial states can be asymptotically reset in the sense that \( x_d(0) - x_{kj}(0) \to 0 \) as the iteration number increases.

**Remark 2.1:** In the ILC field, the identical initialisation condition (i.i.c.) is usually required for the MAS; that is, the MAS is reset to the same initial state for each iteration \( x_{kj}(0) = x_d(0), \forall j \). The i.i.c. is a standard assumption in the ILC design to ensure perfect tracking performance. Assumption 2.1 is a relaxation of this condition. In addition, many papers have studied various initial conditions such as (Chien, Hsu, & Yao, 2004; Park, 1999; Xu & Yan, 2005).

**Assumption 2.2:** The desired trajectory \( y_d(t) \) is realisable in the sense that there exist \( u_d(t) \) and \( x_d(0) \) such that

\[
   x_d(t+1) = A(t)x_d(t) + B(t)u_d(t),
   y_d(t) = C(t)x_d(t), \quad \text{(3)}
\]

Moreover, the desired trajectory \( y_d(t) \) should lie in the measurable range, i.e. \( |y_d(t)| \leq r_0, \forall t, i = 1, 2, \ldots, s_0 \), where

\[
   y_d(t) = [y_d^1(t), y_d^2(t), \ldots, y_d^{s_0}(t)]^T.
\]

**Assumption 2.3:** The matrix \( C(t+1)B(t) \) is assumed to be of full-column rank for all \( t \).

**Assumption 2.4:** The measurement noise \( \{v_{k,j}(t)\} \) is a sequence of independent and identical distributed random variables such that \( \mathbb{E}v_{k,j}(t) = 0, \sup_k \mathbb{E}\|v_{k,j}(t)\|^2 < \infty \), and \( \lim_{n \to \infty} (1/n) \sum_{k=1}^{n} v_{k,j}(t)v_{k,j}^T(t) = R_{ij}^v \), \( \forall t \), where \( R_{ij}^v \) is an unknown matrix. In addition, the communication noise \( \{\omega_{k,i,j}(t)\} \) has the same property as \( \{v_{k,j}(t)\} \).

**Remark 2.2:** Assumption 2.2 assumes the realisability of the desired reference. It should be emphasised that the desired reference should lie in the measurable range, otherwise, the perfect tracking is difficult to achieve due to the limitation of the saturated output. Furthermore, it is worthy noting that there exist suitable initial state \( x_d(0) \) and input \( u_d(t) \) which are mentioned in Assumption 2.2. Assumption 2.3 indicates that the relative degree is 1 and the dimension of output is larger than the dimension of input, i.e. \( s_0 \geq s_i \). Assumption 2.4 imposes the common condition to the unknown stochastic noises. Moreover, it is noted that the noise assumption is made according to the iteration axis rather than the time axis. In practical environments, the process is repeatable from iteration to iteration and thus the assumption is not restricted.

**Assumption 2.5:** The directed graph \( \mathcal{G} \) has a spanning tree with the virtual leader being the root.

**Remark 2.3:** In Assumption 2.5, the virtual leader \( v_0 \) generates the desired trajectory \( y_d(t) \) and its behaviour is not affected by other agents in the MAS. Moreover, Assumption 2.5 is a necessary requirement for the solvability of the consensus tracking problem. To be specific, if there is an isolated agent in the graph \( \mathcal{G} \), it is impossible for that agent to follow the desired trajectory.

In order to simplify the writing, for the \( j \)th agent we set

\[
   \delta u_{k,j}(t) \triangleq u_d(t) - u_{kj}(t), \delta x_{k,j}(t) \triangleq x_d(t) - x_{kj}(t).
\]

The following lemmas are required for the proof of convergence in the following sections.

**Lemma 2.1 (Hu & Hong, 2007):** When the virtual agent has a path to any follower agent, the eigenvalues of \( \mathcal{L} + \mathcal{D} \) have positive real parts, where \( \mathcal{L} \) is the Laplacian matrix of graph \( \mathcal{G} \) and \( \mathcal{D} \) is the diagonal matrix of the communication gain matrix. In addition, when the graph \( \mathcal{G} \) is undirected, the eigenvalues of \( \mathcal{L} + \mathcal{D} \) are positive real numbers.

**Lemma 2.2:** Assume that Assumptions 2.1–2.2 hold for the MAS (1). If \( \lim_{k \to \infty} \delta u_k(s) = 0, s = 0, 1, \ldots, t - 1 \), then at the time instant \( t, \|\delta x_k(t)\| \to 0 \), where

\[
   \delta u_k(t) = [\delta u_{k,1}^T(t), \delta u_{k,2}^T(t), \ldots, \delta u_{k,m}^T(t)]^T,
   \delta x_k(t) = [\delta x_{k,1}^T(t), \delta x_{k,2}^T(t), \ldots, \delta x_{k,m}^T(t)]^T.
\]
Proof: We prove this lemma by mathematical induction. By using (1) and (3), we can obtain the following equation
\[
\delta x_k(t + 1) = (I_n \otimes A(t))x_k(t) - (I_n \otimes A(t))x_k(t) \\
+ (I_n \otimes B(t))u_k(t) - (I_n \otimes B(t))u_k(t) \\
= (I_n \otimes A(t))\delta x_k(t) + (I_n \otimes B(t))\delta u_k(t),
\]
where \(x_k(t) = [x_{k,1}^T(t), x_{k,2}^T(t), \ldots, x_{k,n}^T(t)]^T, u_k(t) = [u_{k,1}^T(t), u_{k,2}^T(t), \ldots, u_{k,n}^T(t)]^T.\)

Initial step. For \(t = 0\), Assumption 2.1 implies that the first term on the right-hand side of the last equation tends to zero. In addition, \(\lim_{k \to \infty} \delta u_k(0) = 0\) leads to the second term on the right-hand side of the last equation also tends to zero. Therefore, we have \(\|\delta x_k(1)\| \to 0\) as \(k \to \infty\).

Inductive step. Assume the conclusion of the lemma is true for \(s = 0, 1, \ldots, t - 1\), then we have \(\|\delta x_k(s)\| \to 0\) as \(k \to \infty\). For the time instant \(t\), under the condition \(\lim_{k \to \infty} \delta u_k(t) = 0\), by using the same steps as that used above, we find that \(\|\delta x_k(t + 1)\| \to 0\) as \(k \to \infty\). That is, the conclusion is valid for \(t\). The proof is thus completed.

In view of Lemma 2.2, if the input errors at all previous time \(\delta u_k(s), s = 0, 1, \ldots, t - 1\) are close to zero, then the state error at current time instant \(\delta x_k(t)\) would be close to zero. Specifically, the asymptotical convergence of tracking error is guaranteed as long as the convergence of the input sequence can be realized asymptotically. Thus, the analysis objective in the next section is to show the precise convergence of the input sequence to the desired input defined in Assumption 2.2. The details of analysis will be elaborated in the next section.

3. Consensus convergence under fixed topology case

In this section, the distributed ILC algorithm will be designed with an appropriate gain matrix. Moreover, the design process will establish the connection between the actual output and the measured output, and provide the main theorem on the asymptotic convergence. The detailed proof is put in the Appendix for smooth readability. As is mentioned in the previous section, only the measured information rather than the actual output can be used in the algorithm design. In the remainder of this section, we elaborate three cases separately.

3.1. Case 1: Noisy output and precise transmission

Let the stacked vector of \(y_{k,j}\) and \(z_{k,j}\) for all agents be as follows:
\[
y_k(t) = [y_{k,1}^T(t), y_{k,2}^T(t), \ldots, y_{k,n}^T(t)]^T,
\]
\[
z_k(t) = [z_{k,1}^T(t), z_{k,2}^T(t), \ldots, z_{k,n}^T(t)]^T.
\]
For the \(j\)th agent, the measured error at the \(k\)th iteration is defined as
\[
e_{k,j}(t) = y_d(t) - z_{k,j}(t).
\]
For the \(j\)th agent, \(\xi_{k,j}(t)\) denotes the distributed information at the \(k\)th iteration:
\[
\xi_{k,j}(t) = \sum_{i \in N_j} a_{ij}(e_{k,i}(t) - e_{k,j}(t)) + d_j(y_d(t) - z_{k,j}(t)),
\]
where the first term on the right-hand side of (6) denotes the information communication between the \(j\)th agent and its neighbour agents. \(a_{ij} = 1\) if \((i, j) \in \mathcal{E}(\mathcal{G})\); that is, the \(i\)th agent is the neighbourhood of the \(j\)th agent and the neighbourhood set of the \(j\)th agent is denoted by \(N_j\). The last term on the right-hand side of (6) denotes the information communication between the virtual leader and the \(j\)th agent.

Inspired by Shen and Zhang (2017), using (6), we can define the distributed P-type ILC update law for the \(j\)th agent as follows:
\[
u_{k+1,j}(t) = u_{k,j}(t) + \lambda_k \Gamma(t)\xi_{k,j}(t + 1),\quad j = 1, 2, \ldots, n,
\]
where \(\lambda_k\) is a decreasing gain such that \(\lambda_k > 0, \lim_{k \to \infty} \lambda_k = 0, \sum_{k=1}^{\infty} \lambda_k = \infty\) and \(\sum_{k=1}^{\infty} \lambda_k^2 < \infty\). It is obvious that \(\lambda_k = c/k\) complies with the above requirements with \(c\) being a suitable positive constant. In addition, the gain matrix is denoted by \(\Gamma(t) \in \mathbb{R}^{n \times n}\) for all agents and \(\Gamma(t)\) is bounded, i.e. \(\|\Gamma(t)\| \leq u_{\Gamma}\) with \(u_{\Gamma}\) being a suitable positive constant for \(k \in \mathbb{N}_+\) and \(t \in \mathbb{Z}_{\mathcal{N}_-}\).

By using (5), the distributed information measured by the \(j\)th agent in (6) can be rewritten as
\[
\xi_{k,j}(t) = \sum_{i \in N_j} a_{ij}(e_{k,i}(t) - e_{k,j}(t)) + d_j(e_{k,j}(t)).
\]
In compact form, we can rewrite (8) as
\[
\xi_k(t) = [(\mathcal{L} + \mathcal{D}) \otimes I_n]e_k(t),
\]
where
\[
e_k(t) = [e_{k,1}^T(t), e_{k,2}^T(t), \ldots, e_{k,n}^T(t)]^T,
\]
\[
\xi_k(t) = [\xi_{k,1}^T(t), \xi_{k,2}^T(t), \ldots, \xi_{k,n}^T(t)]^T.
\]
For expressions simplicity, we consider homogeneous agents in this paper without loss of generality; that is, each agent has identical dynamics. Moreover, due to each agent has the same learning structure, we can reformulate (7) in a compact form,
\[
u_{k+1}(t) = u_k(t) + \lambda_k [(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]e_k(t + 1).
\]
Remark 3.1: For the $j$th agent, the initial control input $u_{0j}(t)$ can be arbitrarily selected. Without loss of generality, we can set the initial control input $u_{0j}(t)$ equals to zero for simplicity, i.e. $u_{0j}(t) \equiv 0$ for $\forall t$. Furthermore, feedback can be used in the MAS to construct $u_{0j}(t)$ to improve the transient behaviour in the learning process.

Remark 3.2: We consider the homogeneous MAS in this paper and thus the learning gain matrix is identical for all agents. Moreover, the learning gain matrix could be different for each agent as far as it satisfies the condition in Theorem 3.1. If one extends the results in the paper to the heterogeneous MAS, the learning gain matrix should be varying for different agents. By slight modifications in notation and definitions, the following derivations are valid for heterogeneous MAS.

From (5) we have
\[
e_{kj}(t + 1) = y_d(t + 1) - \text{sat}(y_{kj}(t + 1) + \nu_{kj}(t + 1)) = \gamma_{kj}(t)y_d(t + 1) - y_{kj}(t + 1) - \nu_{kj}(t + 1),
\]
where $\gamma_{kj}(t) = \text{diag}[\gamma_{kj}^{(1)}(t), \ldots, \gamma_{kj}^{(s_k)}(t)]$. Let $y_{kj}(t) + \nu_{kj}(t) = [m_{kj}^{(1)}(t), \ldots, m_{kj}^{(s_k)}(t)]^T$. The transform factor $\gamma_{kj}^{(i)}(t)$ is described as
\[
\gamma_{kj}^{(i)}(t) = \begin{cases} 
\frac{y_d(t + 1) - r_0}{y_d(t + 1) - \nu_{kj}(t + 1)}, & \text{if } m_{kj}^{(i)}(t + 1) \geq r_0 \\
1, & \text{if } -r_0 < m_{kj}^{(i)}(t + 1) < r_0 \\
\frac{y_d(t + 1) + r_0}{y_d(t + 1) - \nu_{kj}(t + 1)}, & \text{if } m_{kj}^{(i)}(t + 1) \leq -r_0.
\end{cases}
\]
(11)

Note that the desired reference $y_d(t)$ should locate in the measurable range, thus it is obvious that $0 < \gamma_{kj}^{(i)}(t) \leq 1$, for $j = 1, 2, \ldots, n$, $\forall t, k$ and $i = 1, \ldots, s_k$. Therefore, $\gamma_{kj}(t)$ (transform factor matrix) is a special diagonal matrix with all its diagonal elements being positive real numbers no larger than 1; that is, $\|\gamma_{kj}(t)\| \leq 1$.

Remark 3.3: The decreasing gain sequence is introduced to suppress the influence of stochastic noises and to guarantee the asymptotic convergence of the actual output. Moreover, although it is more effective to use the actual output information $y_k(t)$ rather than the measured output information $z_k(t)$ in the learning algorithm (7), the actual output information $y_k(t)$ is not available in MAS. Although the measured output information $z_k(t)$ might slow down the convergence speed, the consensus tracking problem in the MAS is indeed achieved.

Remark 3.4: From (11) we can easily obtain
\[
\|y_d(t + 1) - \text{sat}(y_{kj}(t + 1) + \nu_{kj}(t + 1))\|
\leq \|\gamma_{kj}(t)y_d(t + 1) - y_{kj}(t + 1) - \nu_{kj}(t + 1)\|
\leq \|y_d(t + 1) - y_{kj}(t + 1) - \nu_{kj}(t + 1)\|.
\]

Note that the stochastic noise $\nu_{kj}(t)$ is implicated in the MAS; therefore, the traditional contraction mapping method is unsuitable. To solve this problem, the transform factor $\gamma_{kj}^{(i)}(t)$ is proposed to describe the contraction coefficient for the $i$th dimension of the associated vectors. In addition, it should be emphasised that the contraction coefficient $\gamma_{kj}^{(i)}(t)$ is virtual; in other words, the coefficient is only used for the convergence analysis.

We define
\[
e_k'(t) = y_d(t) - y_{kj}(t) - \nu_{kj}(t).
\]
In compact form, we can reformulate the above equation as follows:
\[
e_k'(t) = e_k^T(t) = [e_{k1}^T(t), e_{k2}^T(t), \ldots, e_{kn}^T(t)]^T,
\]
where $\nu_k(t) = [\nu_{k1}^T(t), \nu_{k2}^T(t), \ldots, \nu_{kn}^T(t)]^T$. Using (11) and (13), we can rewrite (10) as follows:
\[
u_{k+1}(t) = u_k(t) + \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]y_k(t)e_k'(t + 1),
\]
(14)
where
\[
Y_k(t) = \text{diag}[Y_{k1}(t), Y_{k2}(t), \ldots, Y_{kn}(t)].
\]
From (1), (3), (4) and (11), we can reformulate the learning algorithm (14) as follows:
\[
u_{k+1}(t) = u_k(t) + \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]y_k(t)
\times [I_n \otimes (C(t + 1)B(t))]\delta u_k(t)
+ \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]u_k(t)
\times [I_n \otimes (C(t + 1)A(t))]\delta x_k(t)
- \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]Y_k(t)\nu_k(t + 1).
\]
(15)

The first theorem is presented below.
Theorem 3.1: Assume that Assumptions 2.1–2.5 hold for MAS (1) and the learning consensus algorithm (10). The communication mechanism between two neighbour agents is modelled in Case 1. If the learning gain matrix $\Gamma(t)$ satisfies that all eigenvalues of $(\mathcal{L} + \mathcal{D}) \otimes (\Gamma(t)C(t+1)B(t))$ are with positive real parts for all time instances, then the actual output $y_d(t)$ converges to the desired reference vector $y_d(t)$, $\forall t$, and the input vector $u(t)$ which is generated by the learning algorithm (10) converges to the desired input vector $y(t)$ as $k \to \infty$, $t = 0, 1, \ldots, N - 1$. That is, the consensus tracking problem for the MAS is solved as $k \to \infty$.

The proof is put in the Appendix.

3.2. Case 2: precise output and noisy transmission

The measured tracking error at the $k$th iteration for the $j$th agent is defined as follows:

$$e_{kj}(t) = y_d(t) - \text{sat}(y_{kj}(t)).$$

(16)

Note that this error is unavailable for neighbour agents and it is defined for convergence analysis.

Correspondingly, the distributed information measured at the $k$th iteration is described as

$$\xi_{kj}(t) = \sum_{i \in N_j} a_{ij}(\xi_{kj,i} - \text{sat}(y_{kj}(t)))$$

$$+ d_i(y_d(t) - \text{sat}(y_{kj}(t)))$$

$$= \sum_{i \in N_j} a_{ij}(\text{sat}(y_{kj}(t)) - \text{sat}(y_{kj,i}(t))) + d_i(y_d(t) - \text{sat}(y_{kj}(t)))$$

$$- \text{sat}(y_{kj}(t))) + \sum_{i \in N_j} a_{ij} \omega_{kj,i}(t),$$

(17)

In compact form, we can reformulate (17) as

$$\xi_k(t) = [(\mathcal{L} + \mathcal{D}) \otimes I_{s_0}]e_k(t) + [\tilde{A} \otimes I_{s_0}]\omega_k(t),$$

(18)

where

$$\omega_{kj}(t) = [\omega_{kj,1}^T(t), \omega_{kj,2}^T(t), \ldots, \omega_{kj,n}^T(t)]^T,$$

$$\omega_k(t) = [\omega_{k,1}^T(t), \omega_{k,2}^T(t), \ldots, \omega_{k,n}^T(t)]^T,$$

$$\tilde{A} = \begin{bmatrix}
A_1 & 0 & \cdots & 0 \\
0 & A_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & A_n
\end{bmatrix},$$

$A_i$ denotes the $i$th row of the adjacency matrix $A$.

Following similar steps as in Case 1, we present the learning consensus algorithm as follows:

$$u_{k+1}(t) = u_k(t) + \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]e_k(t + 1)$$

$$+ \lambda_k[\tilde{A} \otimes \Gamma(t)]\omega_k(t + 1).$$

(19)

For Case 2, no measurement noise exists in the output. To connect the measured tracking error $e_{kj}(t)$ and the actual tracking error $y_d(t) - y_{kj}(t)$, we modify the transform factor $y_{kj}^{(i)}(t)$ as follows:

$$y_{kj}^{(i)}(t) = \begin{cases}
1, & \text{if } -r_0 < y_{kj}(t + 1) < r_0 \\
y_{kj}^{(i)}(t + 1) - \frac{y_{kj}(t + 1) - r_0}{r_0}, & \text{if } y_{kj}(t + 1) \leq r_0 \\
y_{kj}^{(i)}(t + 1) - \frac{y_{kj}(t + 1) + r_0}{r_0}, & \text{if } y_{kj}(t + 1) \geq r_0 \\
y_{kj}^{(i)}(t + 1), & \text{otherwise}
\end{cases}$$

Then, we obtain

$$\|y_d(t + 1) - \text{sat}(y_{kj}(t + 1))\|$$

$$= \|\gamma_{kj}(t)(y_d(t + 1) - y_{kj}(t + 1))\|$$

$$\leq \|\gamma_{kj}(t)\| \|y_d(t + 1) - y_{kj}(t + 1)\|$$

$$\leq \|y_d(t + 1) - y_{kj}(t + 1)\|.$$ 

Consequently, we can rewrite (19) as follows:

$$u_{k+1}(t) = u_k(t) + \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]\gamma_{kj}(t)$$

$$\times [I_n \otimes (C(t + 1) + B(t))]\delta_k(t)$$

$$+ \lambda_k[(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)]\gamma_{kj}(t)$$

$$\times [I_n \otimes (C(t + 1)A(t))]\delta_k(t)$$

$$+ \lambda_k[\tilde{A} \otimes \Gamma(t)]\omega_k(t + 1).$$

(20)

The second theorem for Case 2 is presented below.

Theorem 3.2: Assume that Assumptions 2.1–2.5 hold for MAS (1) and learning consensus algorithm (19). The communication mechanism between two neighbour agents is modelled in Case 2. If the learning gain matrix $\Gamma(t)$ satisfies that all eigenvalues of $(\mathcal{L} + \mathcal{D}) \otimes (\Gamma(t)C(t+1)B(t))$ are with positive real parts for all time instances, then the actual output $y_k(t)$ will converge to the desired reference vector $1_n \otimes y_d(t)$, $\forall t$, and the input vector $u_k(t)$ converges to the desired input vector $1_n \otimes u_d(t)$ as $k \to \infty$, $t = 0, 1, \ldots, N - 1$.

The proof is put in the Appendix.
3.3. Case 3: noisy output and noisy transmission

Clearly, Case 3 is a combination of the above two cases; thus, the design and analysis of learning consensus algorithm can be conducted by combing previous two subsections. We present the main framework for this case. The measured error at the kth iteration for the jth agent is defined as

\[ e_{kj}(t) = y_d(t) - \text{sat}(y_{kj}(t) + v_{kj}(t)). \]  

(21)

The distributed information for the jth agent at the kth iteration is described as follows:

\[ \xi_{kj}(t) = \sum_{i \in N_j} a_{ij}(z_{kj,i}(t) - \text{sat}(y_{kj}(t) + v_{kj}(t))) \]
\[ + d_i(y_{d}(t) - \text{sat}(y_{kj}(t) + v_{kj}(t))) \]
\[ = \sum_{i \in N_j} a_{ij}(\text{sat}(y_{kj}(t) + v_{kj}(t)) - \text{sat}(y_{kj}(t)) \]
\[ + v_{kj}(t)) \]
\[ + d_i(y_{d}(t) - \text{sat}(y_{kj}(t) + v_{kj}(t))) \]
\[ + \sum_{i \in N_j} a_{ij} \omega_{kj,i}(t), \]  
\[ \xi_k(t) = [(\mathcal{L} + \mathcal{D}) \otimes I_y] e_{k}(t) + [\tilde{A} \otimes I_y] \omega_k(t). \]  
\[ (22) \]

Then, the compact form of the learning consensus algorithm is given as follows:

\[ u_{k+1}(t) = u_k(t) + \lambda_k [(\mathcal{L} + \mathcal{D}) \otimes \Gamma(t)] e_k(t + 1) \]
\[ + \lambda_k [\tilde{A} \otimes \Gamma(t)] \omega_k(t + 1). \]  
\[ (24) \]

Theorem 3.3: Assume that Assumptions 2.1–2.5 hold for MAS (1) and learning consensus algorithm (24). The communication mechanism between two neighbour agents is modelled in Case 3. If the learning gain matrix \( \Gamma(t) \) satisfies that all eigenvalues of \((\mathcal{L} + \mathcal{D}) \otimes (\Gamma(t)C(t + 1)B(t))\) are with positive real parts for all time instances, then the actual output \( y(t) \) will converge to the desired reference vector \( 1_n \otimes y_d(t) \), \( \forall t \), and the input vector \( u_k(t) \) converges to the desired input vector \( 1_n \otimes u_d(t) \) as \( k \to \infty \), \( t = 0, 1, \ldots, N - 1 \).

The proof can be completed by combining the analysis for Theorems 3.1 and 3.2. We omit it for the sake of brevity.

Theorems 3.1–3.3 reveal the connection between the asymptotic convergence of the input information and actual output information. In addition, with the help of a decreasing gain sequence \( \lambda_k \) and an appropriate gain matrix \( \Gamma(t) \), we can obtain an asymptotic convergence of the input \( u_k(t) \) to the desired one defined in (3). Furthermore, in view of Lemma 2.2, we get the the asymptotic convergence of the actual output in an almost sure sense.

The pivotal idea of the proof is to find a continuous contraction along the iteration axis (i.e. \( \Phi_{\lambda, t} \) defined in the Appendix, (A7)). In addition, it should be emphasised that the function \( \Phi_{\lambda, t} \) is a product of successive matrices when considering MIMO MAS. Moreover, the convergence condition can be guaranteed when we select \( \Gamma(t) \) such that all eigenvalues of \( \Gamma(t)C(t + 1)B(t) \) are positive real numbers by using the basic property of the Kronecker product. In the next section, we will discuss the switching topologies case; that is, the communication graph among agents is a dynamically changing directed graph.

4. Extensions to time-switching topologies case

In this section, the proposed ILC learning algorithm is extended to the time-switching topologies case. Assumption 2.5 will be slightly modified according to the switching topologies. It should be pointed out that the system equation is the same as (1).

Every agent corresponds to a vertex in the dynamically changing directed graph \( \mathcal{G}(t) \). Each edge \((v_i, v_j) \in \mathcal{E}(t)\) denotes the information flow between agent \( v_i \) and \( v_j \) at the time step \( t \). Moreover, the adjacency matrix of dynamically changing directed graph \( \mathcal{G}(t) \) is denoted by \( A(t) = [a_{ij}(t)] \). Therefore, the index set of all neighbours of \( v_i \) and the Laplacian matrix associated with \( \mathcal{G}(t) \) are denoted by \( N_i(t) \) and \( \mathcal{L}(t) \), respectively. In addition, the set of all directed graphs for the agents are denoted by \( \mathcal{G}_\alpha = \{G_{\alpha_1}, G_{\alpha_2}, \ldots, G_{\alpha_\mu}\} \); that is, \( \mathcal{G}(t) \in \mathcal{G}_\alpha \) for all \( t \in \mathbb{Z}_N \).
As mentioned in the previous sections, the desired trajectory is only available to a portion of follower agents. It should be emphasised that \( d_i(t) = 1 \) indicates that the agent \( v_j \) can get the desired trajectory \( y_d(t) \) at the time step \( t \), otherwise \( d_i(t) = 0 \). Then we can easily modify \( D \) as \( D(t) = \text{diag}\{d_1(t), d_2(t), \ldots, d_n(t)\} \). Note that \( G(t) \in \mathcal{G}_a \), and \( \mathcal{G}_a \) consists of \( q \) directed graphs. In addition, we can find that \( D(t) \in \mathcal{D}_a \), where \( \mathcal{D}_a = \{\mathcal{D}_{a_1}, \mathcal{D}_{a_2}, \ldots, \mathcal{D}_{a_q}\} \) and \( \mathcal{D}_{a_i} \) indicates the accessibility of the desired reference for the follower agents in \( \mathcal{G}_{a_i}, i = 1, 2, \ldots, q \).

The concept of the virtual leader is described in the previous sections. Then the directed graph \( \mathcal{G}(t) \in \mathcal{G}_a = \{\mathcal{G}_{a_1}, \mathcal{G}_{a_2}, \ldots, \mathcal{G}_{a_q}\} \) consists of the \( n+1 \) agents labelled \( v_0, v_1, \ldots, v_n \). It should be noted that the switching among all graphs can be in an arbitrary way with respect to the time step \( t \). In other words, there is no restriction on the switching between any two graphs.

Based on the above supplement, Assumption 2.5 can be modified as

**Assumption 4.1:** The directed graph \( \mathcal{G}_{a_i} \) has a spanning tree with the virtual leader being the root for \( i = 1, 2, \ldots, q \).

If Assumption 4.1 holds, \( v_0 \) is the root vertex of the directed graph \( \mathcal{G}(t) \) for all \( t \in \mathbb{Z}_N \), where \( \mathcal{G}(t) \in \mathcal{G}_a \). The virtual leader \( v_0 \) generates the desired trajectory and will not receive any information from any other agents.

We elaborate Case 3 in this section because the other two are special cases of Case 3. Based on the above analysis, for the \( j \)th agent we can rewrite (7) in compact forms as

\[
\begin{align*}
u_{k+1}(t) &= u_k(t) + \lambda_k[(L(t) + D(t)) \otimes \Gamma(t)]p(t + 1) \\
&+ \lambda_k[\bar{A}(t) \otimes \Gamma(t)]p(t + 1),
\end{align*}
\]

(26)

where \( \lambda_k \) is a decreasing gain, \( L(t) \) and \( D(t) \) are time-varying matrices, and the gain matrix is denoted by \( \Gamma(t) \in \mathbb{R}^{n \times n} \) for all agents. Moreover, \( \Gamma(t) \) is bounded, i.e. \( \|\Gamma(t)\| \leq u_T \) for all \( t \in \mathbb{Z}_N \).

The last theorem of this paper is present below.

**Theorem 4.1:** Assume that Assumptions 2.1–2.4 and 4.1 hold for MAS (1). The learning algorithm (26) is applied for MAS (1) with dynamic changing directed graphs \( \mathcal{G}(t) \in \mathcal{G}_a \). If the learning gain matrix \( \Gamma(t) \) satisfies that all eigenvalues of \( [(L(t) + D(t)) \otimes \Gamma(t)]C(t + 1)B(t)] \) are with positive real parts then the actual output \( y_k(t) \) will converge to the desired reference vector \( 1_n \otimes y_d(t) \), \( \forall t \), and the input \( u_k(t) \) which is generated by the learning algorithm (26) converges to the desired input \( 1_n \otimes u_d(t) \) as \( k \to \infty \), \( t = 0, 1, \ldots, N - 1 \). That is, the MAS solves the consensus tracking problem as \( k \to \infty \).

The proof is put in the Appendix.

We have now found that the convergence of ILC for the MAS with measurement range limitation and noises for both fixed topology and time-switching topologies case. Theorem 4.1 extends the learning algorithm results from the fixed topology case to the dynamic changing topology case.

**Remark 4.1:** For the switching topology case, in which the graph is dynamically changing along both time and iteration axes, if all eigenvalues of \( M_{ai} = [(L_{ai}(t) + D_{ai}(t)) \otimes (\Gamma(t)C(t + 1)B(t))] \) are positive real parts and there exists a common positive-definite matrix \( P \) such that \(-M_{ai}^TP + PM_{ai}\) is negative-definite for all \( i = 1, \ldots, q \), the actual output will converge to the desired one. However, it is a strong condition. In other words, the existence of such a matrix \( P \) actually is a requirement on the existence of a common Lyapunov function to ensure the successive contraction along the iteration axis.

5. Illustrative simulations

In this section, the leaning algorithms are simulated for both fixed and time-switching topologies case.

5.1. MIMO system with fixed topology

We consider the MAS with fixed topology \( \mathcal{G} \) as shown in Figure 4. Moreover, \( t \in \mathbb{Z}_{30} \) is considered. The virtual leader has directed edges to Agents 2 and 3. The communication among followers is directed. In addition, the Laplacian matrix and pinning gain matrix are described as follows:

\[
L = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.
\]

In this example, the system matrices for all agents are describe as follows:

\[
A(t) = \begin{bmatrix} -0.05 \sin(0.2t) & 0 & -0.01t \\ 0 & -0.01t & -0.02 \cos(0.5t) \\ 0.2 & 0.4 & 0.3 + 0.1 \cos(0.2t) \end{bmatrix},
\]

\[
B(t) = \begin{bmatrix} 1 - 0.1 \sin^2(0.5t) & 0 \\ 0.01t & 0.01t \\ 0.1 & 1 + 0.1 \sin(0.5t) \end{bmatrix},
\]

\[
C(t) = \begin{bmatrix} 1 + 0.1 \sin^2(0.5t) & 0 & 0 \\ 0 & 0 & 1 - 0.2 \sin^2(0.5t) \end{bmatrix}.
\]

The initial state is chosen as \( x_{1,j}(0) = [0, 0, 0]^T \) for \( j = 1, 2, 3, 4 \). Moreover, the zero initial input will be used
for all agents, i.e. $\mathbf{u}_{0,j} \equiv \mathbf{0}$ for all agents. The measurement range is set to be $(-2.5, 2.5)$. That is, the upper bound for the first and second dimensions is 2.5 and the lower bound for the first and second dimensions is $-2.5$.

In addition, for the $j$th agent the measurement noise is termed as $\mathbf{v}_{k,j}(t) = [v_{k,1,j}(t), v_{k,2,j}(t)]^T$ with each dimension satisfying that $v_{k,j}(t) \sim N(0,0.05^2)$. Similarly, the communication noise $\omega_{k,j}(t) \sim N(0,0.05^2)$, $\sigma = 1, 2$. The algorithm runs for 40 iterations. The desired trajectory is described as

$$\mathbf{y}_d(t) = \begin{bmatrix} 2 \sin(0.1t\pi) \\ 2 \cos(0.1t\pi) - \frac{2}{t+1} \end{bmatrix} \quad \text{for } t \in \mathbb{Z}_{30}. \tag{27}$$

Moreover, the learning gain matrix is devised as

$$\Gamma(t) = \begin{bmatrix} 0.8 & 0.5 \\ -0.2 & 1.4 \end{bmatrix}$$

for all $t$ and the decreasing gain $\lambda_k$ is designed as a piecewise function

$$\lambda_k = \begin{cases} \frac{1}{k}, & 0 < k \leq 10 \\ \frac{3}{k}, & k > 10. \end{cases}$$

For the sake of the measurement saturation, for Agent 2 in Case 1, the second dimension of the system actual output $y_{k,2}(t) + \omega_{k,2}(t)$ and the measured output $z_{k,3}(t)$ at the second iteration are shown in Figure 5. Here, we take the second iteration just for illustration the measurement saturation. As can be seen, the upper bound for the second dimension is 2.5 and the lower bound for the second dimension is $-2.5$; that is, only the actual output trajectories of agents which are located in the measurable range can be obtained.

We also plot the actual output trajectories at the 40th iteration for Cases 1 and 2 in Figures 6 and 7, respectively. It can be observed from this figure that the actual output trajectories of all agents do not exceed the measurement bound. In addition, after sufficient learning iterations, all agents track the desired trajectory perfectly. This figure has demonstrated the effectiveness of the proposed learning algorithm (10) and (19) under measurement saturation and noises.

For Agent 2, the maximum actual tracking error $\max_t \| \mathbf{y}_d(t) - \mathbf{y}_{k,1}(t) \|$ and the maximum measured tracking error $\max_t \| \mathbf{y}_d(t) - z_{k,3}(t) \|$ over the first 40 iterations for Cases 1 and 2 are shown in Figures 8–11 for two dimensions, respectively. Clearly, those figures show that the consensus tracking objective can be achieved. Moreover, the noise free case is also simulated where the decreasing gain $\lambda_k$ is replaced by a constant gain $\frac{1}{3}$ and the maximum tracking error profiles are also displayed.

### 5.2. MIMO system with switching topologies

Case 3 is simulated in this part. The system matrices, the initial state, the initial input, the noises and the
measurement range are defined as same as the previous case. In addition, the algorithm runs for 50 iterations.

In this case, the communication graphs among agents are dynamically changing at the time step $t$. The communication graphs for agents are considered to switch over four states as shown in Figure 12, i.e., $\mathcal{G}_\alpha = \{G_a, G_b, G_c, G_d\}$, where all of these graphs have a spanning tree with vertex $v_0$ being the root. Furthermore, in order to simulate the switching of the communication topologies, the switching signal is defined by $\alpha(t)$ varying over the interval $[0, 1]$ as a function of $t$. We use the MATLAB command ‘rand’ to produce the switching signal.

The switching rule is described as follows:

- If $\alpha(t) \in [0, 0.25)$, then $\mathcal{G}_\alpha = G_a$,
- If $\alpha(t) \in [0.25, 0.5)$, then $\mathcal{G}_\alpha = G_b$,
- If $\alpha(t) \in [0.5, 0.75)$, then $\mathcal{G}_\alpha = G_c$,
- If $\alpha(t) \in [0.75, 1]$, then $\mathcal{G}_\alpha = G_d$. 

---

**Figure 7.** The actual output for all agents (Case 2).

**Figure 8.** Maximal error profiles of first output (Case 1).

**Figure 9.** Maximal error profiles of second output (Case 1).

**Figure 10.** Maximal error profiles of first output (Case 2).

**Figure 11.** Maximal error profiles of second output (Case 2).
Based on Theorem 4.1, the learning gain matrix is selected as

\[ \Gamma(t) = \begin{bmatrix} 1.2 & 0.3 \\ -0.2 & 1.5 \end{bmatrix} \]

for all \( t \) and the decreasing gain \( \lambda_k \) is designed as a piecewise function

\[ \lambda_k = \begin{cases} 1 & 0 < k \leq 10 \\ \frac{5}{k} & k > 10. \end{cases} \]

The actual output trajectories at the 50th iteration are shown in Figure 13. As can be seen, the actual output trajectories of all agents are located in the measurable range. In addition, with the help of (26), all agents are able to track the desired trajectory well. This figure shows that the proposed learning algorithm (26) can solve the consensus tracking problem under measurement saturation and noises. Moreover, in order to show the measurement saturation, for Agent 3, the second dimension of the system actual output and the measured output at the second iteration are shown in Figure 14.

For agent 3, the maximum actual tracking error and the maximum measured tracking error over the first 50 iterations are shown in Figures 15 and 16 for two dimensions, respectively. Moreover, we also simulate the case of constant gain for the noise free system, where the constant learning gain is set to be \( \frac{1}{3} \).
Obviously, Figures 15 and 16 show that by using (26) we can solve the consensus tracking problem under switching topologies; however, the maximum error profiles are not perfectly smooth due to the noises and switching topologies. Nevertheless, the tracking performance is already good enough.

6. Conclusions

The ILC is addressed for the MAS with each agent being described by a MIMO discrete-time system with both fixed and time-switching topologies. Moreover, the measurement range limitation and noise are considered in the MAS. In order to deal with the above problem, a distributed ILC scheme is proposed by using the measured output information. In addition, a transform factor is introduced to establish the connection between the measured tracking error and the actual tracking error. By illustrative simulations, we have verified the effectiveness of the proposed learning scheme. For further research, it is of great significance to further relax the agent model such as nonlinear system and heterogeneous agents.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by National Natural Science Foundation of China [grant numbers 61673045, 11661016], the Fundamental Research Funds for the Central Universities [grant number XK1802-4], and the Major Research Project of Innovative Group in Guizhou Education Department [grant number [2018]012].

Notes on contributors

Chen Liu received the B.E. degree from Chang’ an University, Xi’an, China, in 2017. Now, he is pursuing a master degree at Beijing University of Chemical Technology. His research interests include adaptive control, iterative learning control and distributed control of MAS.

Dong Shen (M’10–SM’17) received the B.S. degree in mathematics from School of Mathematics, Shandong University, Jinan, China, in 2005. He received the Ph.D. degree in mathematics from the Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and System Science, Chinese Academy of Sciences (CAS), Beijing, China, in 2010. From 2010 to 2012, he was a Post-Doctoral Fellow with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, CAS. From 2016.02 to 2017.02, he was a visiting scholar at National University of Singapore (NUS), Singapore. Since 2012, he has been with the College of Information Science and Technology, Beijing University of Chemical Technology (BUCT), Beijing, China, where he now is a Professor. His current research interests include iterative learning control, stochastic control and optimization. He is an Associate Editor of IEEE Access.

JinRong Wang has been a Professor of the Department of Mathematics at the School of Mathematics and Statistics at Guizhou University, People’s Republic of China since 2011. He received his master’s degree from Guizhou University in 2006 and PhD from the same university in 2009. He is interested in impulsive differential equations, fractional differential equations, delay differential equations and iterative learning controls. He is the 2015–2018 Highly Cited Researcher in Mathematics.

References

multiagent systems with uncertainties. *IEEE Transactions on Neural Networks, 21*(8), 1351–1358.


**Appendix**

**Proof of Theorem 3.1:** It is sufficient to prove that $u_k(t) \rightarrow I_n \otimes u_k(t)$ a.s., $t = 0, 1, \ldots, N - 1$ based on (1) and Lemma 2.2. To this end, with the help of Lemmas 2.1 and 2.2, we use the mathematical induction method to obtain this proof along the time axis $t$.

**Initial step:** First, we prove the validity of the Theorem 3.1 for $t = 0$. In this case, (15) can be reformulated as

$$u_{k+1}(0) = u_k(0) + \lambda_k \bigl[ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \bigr] T_k(0)$$

$$\times \bigl[ I_n \otimes (C(1)B(0)) \bigr] \delta u_k(0)$$

$$+ \lambda_k \bigl[ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \bigr] T_k(0)$$

$$\times \bigl[ I_n \otimes (C(1)A(0)) \bigr] \delta x_k(0)$$

$$- \lambda_k \bigl[ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \bigr] T_k(0) u_k(1).$$

(A1)
Subtracting both sides of last equation from $1_n \otimes u_d(0)$, then we have
\[ \delta u_{k+1}(0) = (I - \lambda_k[(L + D) \otimes (\Gamma(0))]Y_k(0) \times \{I_n \otimes (C(1)B(0))\})\delta u_k(0) \]
\[ - \lambda_k[(L + D) \otimes (\Gamma(0))]Y_k(0) \times \{I_n \otimes (C(1)A(0))\})\delta x_k(0) \]
\[ + \lambda_k[(L + D) \otimes (\Gamma(0))]Y_k(0)\rangle w_k(1). \quad (A2) \]

It is noted that in system (1) the output $\gamma_k(j), j = 1, 2, 3, \ldots, n$ is bounded. Furthermore, combing with the Assumption 2.4, we can conclude that there exists a nonzero lower bound of $\gamma_k(j), 0$, that is, there is a suitable $0 < v < 1$ that $\gamma_k(j) > v, \forall k, j$. In addition, we can derive that $\langle Y_k(0) \rangle > v$.

Set
\[ \Phi_{l,m} \triangleq \{I - \lambda_m[(L + D) \otimes (\Gamma(0))]Y_m(0)[I_n \otimes (C(1)B(0))]\ldots \]
\[ [I - \lambda_m[(L + D) \otimes (\Gamma(0))]Y_m(0)[I_n \otimes (C(1)B(0)))]\}, \quad l \geq m, \Phi_{l+1,l} \triangleq I. \quad (A3) \]

It is noted that $\Phi_{l,m}$ is a successive production of the contraction $[I - \lambda_m[(L + D) \otimes (\Gamma(0))]Y_m(0)[I_n \otimes (C(1)B(0))]]$ with the subscript $k$ increasing from $m$ up to $l$. It is obvious that $\Phi_{l,l} = [I - \lambda_m[(L + D) \otimes (\Gamma(0))]Y_m(0)[I_n \otimes (C(1)B(0))]])$ is a special case. In addition, when $l < m$, the notation $\Phi_{l,m}$ has no sense, and thus, we define $\Phi_{l,m} = I$ for the completeness of the notation. It is clear that $\{I - \lambda_m[(L + D) \otimes (\Gamma(0))]Y_m(0)[I_n \otimes (C(1)B(0))]\} > 0$ for all sufficient large $m$.

Then for any $l \geq m, m > 0$, it is sufficient to verify that
\[ \|\Phi_{l,m}\| \leq c_2 \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right) \quad (A4) \]
for some suitable $c_1, c_2 > 0$.

Note that $\|Y_l(0)\| \geq vI$ and all eigenvalues of $[(L + D) \otimes (\Gamma(t)C(t+1)B(t))]$ are with positive real parts. Therefore, according to the Lyapunov stability for linear system (see e.g. Chen & Lin, 2004, p. 37), for any given positive definite matrix $Q$, there exists a positive-definite matrix $P$ such that
\[ -[P((L + D) \otimes (\Gamma(t)C(t+1)B(t)))] + [(L + D) \otimes (\Gamma(t)C(t+1)B(t))]^TP = -Q. \]

Here, we let $Q = 2I_{2n}$, and compute the corresponding matrix $P$. Moreover, $Y_k(0)$ is a diagonal matrix and for sufficient large number $k$ we have
\[ -[P((L + D) \otimes (\Gamma(0))]Y_k(0)[I_n \otimes (C(1)B(0))] + [(L + D) \otimes (\Gamma(0))]Y_k(0)[I_n \otimes (C(1)B(0))]^TP \leq -vI. \]

For simplicity, we denote $W_k \triangleq -[(L + D) \otimes (\Gamma(0))]Y_k(0)[I_n \otimes (C(1)B(0))].$ Thus, for sufficiently large number $l$,
\[ \Phi_{l,m}^T P\Phi_{l,m} = \Phi_{l-1,m}^T(I + \lambda_l W_l)^T P(I + \lambda_l W_l)\Phi_{l-1,m} \]
\[ = \Phi_{l-1,m}^T(P + \lambda_l^2 W_l^2 P W_l) + \lambda_l W_l^TP + \lambda_l P W_l)\Phi_{l-1,m} \]
\[ \leq \Phi_{l-1,m}^T(P + \lambda_l^2 W_l^2 P W_l - v \lambda_l I)\Phi_{l-1,m} \]
\[ = \Phi_{l-1,m}^T P^{l/2} (I - v \lambda_l I)^{P^{l/2}} + \lambda_l^2 P^{l/2} W_l P W_l^{P^{l/2}} P^{l/2} \Phi_{l-1,m}. \quad (A5) \]

Without loss of generality, when the iteration number $l$ is sufficiently large, we obtain
\[ \|I - v \lambda_l P^{-1} + \lambda_l^2 P^{-1/2} W_l^T P W_l P^{-1/2}\| \]
\[ \leq 1 - 2c_1 \lambda_l \leq \exp(-2c_1 \lambda_l), \quad (A6) \]
where $c_1 > 0$ is a suitable constant and the basic inequality $1 - x \leq e^{-x}$ is applied.

Noticing the boundedness of $P^{-1/2} W_l^T P W_l P^{-1/2}$ and combining (A5) and (A6) leads to
\[ \Phi_{l,m}^T P\Phi_{l,m} \leq c_3 \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right) \]
for some suitable $c_3 > 0$. Thus,
\[ \|\Phi_{l,m}\| \leq \lambda^{-1/2}_{\min}(P) \sqrt{c_3} \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right). \quad (A7) \]

As a result, we define $c_2 \triangleq \lambda^{-1/2}_{\min}(P) \sqrt{c_3}$, then (A4) is guaranteed.

Now from (A2) we have
\[ \delta u_{k+1}(0) = \Phi_{k+1,k} \delta u_k(0) - \sum_{m=1}^{k} \Phi_{k,m+1} \lambda_m \]
\[ \times \{I_n \otimes (C(1)A(0))\} \delta x_m(0) \]
\[ \times \{I_n \otimes (C(1)A(0))\} \delta x_m(0) \]
\[ + \sum_{m=1}^{k} \Phi_{k,m+1} \lambda_m \{I_n \otimes (C(1)B(0))\} \delta x_m(0) \]
\[ \|\Phi_{k,m+1}\| \leq \lambda^{-1/2}_{\min}(P) \sqrt{c_3} \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right). \quad (A8) \]

where the first term on the right-hand side of (A8) tends to 0 as $k \rightarrow \infty$ because of (A7).

From Assumptions 2.1, it is obvious that $\|\delta x_k(0)\| \rightarrow 0$ as iteration number increases. Hence, for any $\varepsilon > 0$, there exists a sufficient large integer $k_1$ such that $\|\delta x_k(0)\| < \varepsilon, \forall k \geq k_1$. Then for the second term of (A8), we have
\[ \sum_{m=1}^{k} \|\Phi_{k,m+1}\| \lambda_m \{I_n \otimes (C(1)B(0))\} \delta x_m(0) \]
\[ \times \{I_n \otimes (C(1)A(0))\} \delta x_m(0) \]
\[ \|\Phi_{k,m+1}\| \leq \lambda^{-1/2}_{\min}(P) \sqrt{c_3} \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right). \quad (A8) \]

where the first term on the right-hand side of (A8) tends to 0 as $k \rightarrow \infty$ because of (A7).

From Assumptions 2.1, it is obvious that $\|\delta x_k(0)\| \rightarrow 0$ as iteration number increases. Hence, for any $\varepsilon > 0$, there exists a sufficient large integer $k_1$ such that $\|\delta x_k(0)\| < \varepsilon, \forall k \geq k_1$. Then for the second term of (A8), we have
\[ \sum_{m=1}^{k} \|\Phi_{k,m+1}\| \lambda_m \{I_n \otimes (C(1)B(0))\} \delta x_m(0) \]
\[ \times \{I_n \otimes (C(1)A(0))\} \delta x_m(0) \]
\[ \|\Phi_{k,m+1}\| \leq \lambda^{-1/2}_{\min}(P) \sqrt{c_3} \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right). \quad (A8) \]

where the first term on the right-hand side of (A8) tends to 0 as $k \rightarrow \infty$ because of (A7).

From Assumptions 2.1, it is obvious that $\|\delta x_k(0)\| \rightarrow 0$ as iteration number increases. Hence, for any $\varepsilon > 0$, there exists a sufficient large integer $k_1$ such that $\|\delta x_k(0)\| < \varepsilon, \forall k \geq k_1$. Then for the second term of (A8), we have
\[ \sum_{m=1}^{k} \|\Phi_{k,m+1}\| \lambda_m \{I_n \otimes (C(1)B(0))\} \delta x_m(0) \]
\[ \times \{I_n \otimes (C(1)A(0))\} \delta x_m(0) \]
\[ \|\Phi_{k,m+1}\| \leq \lambda^{-1/2}_{\min}(P) \sqrt{c_3} \exp \left(-c_1 \sum_{k=m}^{l} \lambda_k \right). \quad (A8) \]
\[
\sum_{m=k_1}^{k} \|\Phi_{k,m+1}^{\dagger} \lambda_m \Pi_m [(\mathcal{L} + \mathcal{D}) \otimes \Gamma(0)] Y_m(0) \\
\times [I_n \otimes (\mathcal{C}(1) A(0))] \| \| \delta x_m(0) \|. 
\] (A9)

It is noted that the first term on the right-hand side of (A9) is a finite summation, and each term tends to 0 as \( k \to \infty \) because the starting number is bounded by \( k_1 \). Therefore, the finite summation tends to 0 as iteration number increases. Note that \( \lim_{k \to \infty} \lambda_k \to 0 \) and \( \| \Gamma \| \leq u_r \), then for the second term of (A9), we have the following estimation

\[
\sum_{m=k_1}^{k} \lambda_m \exp \left( -c_1 \sum_{i=1}^{k} \lambda_i \right) \leq 2c_2 \| (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \| I_n \otimes (\mathcal{C}(1) A(0)) \|. 
\]

where it is obvious that \( \| (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \| \theta_k - [ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta \to 0 \) as \( k \to \infty \). For the last term, we have

\[
\sum_{m=k_1}^{k} \Pi_m ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta) \\
= \left( 1 - e^{-c_1 \lambda_m} \right) \exp \left( -c_1 \sum_{i=1}^{k} \lambda_i \right) \\
= \frac{2c_2}{c_1} \| (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \| I_n \otimes (\mathcal{C}(1) A(0)) \|. 
\]

Hence, the second term on the right-hand side of (A9) tends to 0 as \( k \to \infty \). In summary, (A9) tends to 0 as \( k \to \infty \).

For the last term of (A8), based on Assumption 2.4 and the fact that the gain matrix is bounded, it is clear that \( \{ v_k(1) \} \) is a sequence of independent and identical distributed random variables. Furthermore,

\[
\sum_{k=1}^{\infty} \mathbb{E} \| \lambda_k Y_k(0) v_k(1) \|^2 \\
\leq \sum_{k=1}^{\infty} \mathbb{E} \| \lambda_k v_k(1) \|^2 \leq \| R_1^T \| \sum_{k=1}^{\infty} \lambda_k^2 < \infty, 
\]

where

\[
R_1^T = [ R_{1,1}^T, R_{1,2}^T, \ldots, R_{1,n}^T ]^T. 
\]

This fact implies \( \sum_{k=1}^{\infty} \lambda_k Y_k(0) v_k(1) < \infty \) a.s. by the Khintchine–Kolmogorov convergence theorem (Chow, 1997).

Now we set \( \theta_k = \sum_{m=k_1}^{k} \lambda_m Y_m(0) v_m(1) \), \( \theta_0 = 0 \). Then \( \theta_k \to \theta \) as \( k \to \infty \). For any \( \varepsilon > 0 \), there is a sufficient large \( k_2 \) such that \( \| \theta_k - \theta \| \leq \varepsilon, \forall k \geq k_2 \).

\[
\sum_{m=k_1}^{k} \Phi_{k,m+1}^{\dagger} \lambda_m ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] Y_m(0) v_m(1) \\
= \sum_{m=1}^{k} \Phi_{k,m+1}^{\dagger} [ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] (\theta_m - \theta_{m-1}) \\
= [(\mathcal{L} + \mathcal{D}) \otimes \Gamma(0)] \theta_k \\
- \sum_{m=1}^{k} (\Phi_{k,m+1}^{\dagger} - \Phi_{k,m}^{\dagger}) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta) \\
= [(\mathcal{L} + \mathcal{D}) \otimes \Gamma(0)] \theta_k \\
- \sum_{m=1}^{k} (\Phi_{k,m+1}^{\dagger} - \Phi_{k,m}^{\dagger}) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta) \\
+ \Phi_{k,1}^{\dagger} [ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta \\
- \sum_{m=1}^{k} (\Phi_{k,m+1}^{\dagger} - \Phi_{k,m}^{\dagger}) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta), 
\]

it is clear that \( \sum_{m=1}^{k_2} (\Phi_{k,m+1}^{\dagger} - \Phi_{k,m}^{\dagger}) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta) \to 0 \) as \( k \to \infty \). Due to the fact that \( m \) is bounded by \( k_2 \), we have \( \Pi_m \| \to 0 \) as \( k \to \infty \). On the other hand,

\[
\sum_{m=k_1+1}^{k} (\Phi_{k,m+1}^{\dagger} - \Phi_{k,m}^{\dagger}) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta) \\
= \sum_{m=k_1+1}^{k} \Phi_{k,m+1}^{\dagger} \lambda_m ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] Y_m(0) \\
\times [I_n \otimes (\mathcal{C}(1) B(0))] ) ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \theta_{m-1} - \theta) \\
\leq \varepsilon c_2 \| (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) \| \| I_n \otimes (\mathcal{C}(1) B(0)) \| \\
\times ([ (\mathcal{L} + \mathcal{D}) \otimes \Gamma(0) ] \| k \lambda_m \exp \left( -c_1 \sum_{i=1}^{k} \lambda_i \right) \\
\]

It is clear that the above equation tends to 0 as \( k \to \infty \) and \( \varepsilon \to 0 \). Above all, \( \delta u_{k+1}(t) \to 0 \) as \( k \to \infty \), the conclusion of this theorem is true for \( t = 0 \).
**Inductive step:** Assume that for \( s = 0, 1, \ldots, t - 1 \) the theorem is true, then we continue to prove that the theorem is true for \( t \). From (15), we can obtain the counterpart of (A2) for time instant \( t \) as

\[
\delta u_{k+1}(t) = [I - \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t))] Y_k(t) \\
\times [I_n \otimes (C(t + 1)B(t))] \delta u_k(t) \\
- \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t)) Y_k(t) \\
\times [I_n \otimes (C(t + 1)A(t))] \delta x_k(t) \\
+ \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t)) Y_k(t) w_k(t + 1). \quad (A10)
\]

It is easy to observe that \( \delta u_k(t) \) corresponds to \( \delta u_k(0), I - \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t))] Y_k(t) \) for \( k \to \infty \) as \( k \to \infty \). By the induction assumption we have \( \|\delta x_k(t)\| \to 0 \) as \( k \to \infty \). Hence, the proof can be following the same steps as far as we guarantee that \( \|\delta x_k(t)\| \to 0 \) as \( k \to \infty \). As a result, we can easily prove that \( \|\delta u_k(t)\| \to 0 \) as \( k \to \infty \). The proof of Theorem 3.1 is completed.

**Proof of Theorem 3.2:** We copy the learning consensus algorithm (20) as follows:

\[
u_{k+1}(t) = u_k(t) + \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t)) Y_k(t) \\
\times [I_n \otimes (C(t + 1)B(t))] \delta u_k(t) \\
+ \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t)) Y_k(t) \times [I_n \otimes (C(t + 1)A(t))] \delta x_k(t) \\
+ \lambda_k([\mathcal{L} + \mathcal{D}] \otimes \Gamma(t)) \omega_k(t + 1).
\]

It is noticed that the recursion is same to (15) except the last term on the form of additive noises. Thus, the proof can be completed following similar steps as in Theorem 3.1.

For the initial step, \( t = 0 \). Subtracting both sides of the above equation from the desired input \( u_d(t) \) yields

\[
\delta u_{k+1}(0) = \Phi_{k+1} \delta u_1(0) - \sum_{m=1}^{k} \Phi_{k,m+1} \lambda_m \\
\times [(\mathcal{L} + \mathcal{D}) \otimes (\Gamma(t))] Y_m(0) \\
\times [I_n \otimes (C(1)A(0))] \delta x_m(0) \\
- \sum_{m=1}^{k} \Phi_{k,m+1} \lambda_m [\mathcal{A} \otimes \Gamma(0)] \omega_m(1). \quad (A11)
\]

It is obvious that \( \Phi_{m,m} \) defined in (A3) is still valid for Case 2. Thus, the estimation (A7) can be applied. With this estimation of \( \Phi_{m,m} \), we can conclude that the first two terms on the right-hand side of (A11) follow the same steps as in Theorem 3.1. For the last term of (A11), we can verify \( \sum_{k=1}^{\infty} \lambda_k w_k(1) < \infty \) by noticing \( \sum_{k=1}^{\infty} E[|\lambda_k w_k(1)|^2] < \infty \).

We let \( \theta_k = \sum_{m=1}^{k} \lambda_m \omega_m(1) \), \( \theta_{\infty} = 0 \). Then \( \theta_k \to \theta \) as \( k \to \infty \) where \( \theta = \sum_{m=1}^{\infty} \lambda_m \omega_m(1) \). For any \( \epsilon > 0 \), there exists a sufficiently large \( k_3 \) such that \( |\theta_k - \theta| \leq \epsilon, \forall k \geq k_3 \). Then, we have

\[
\sum_{m=1}^{k} \Phi_{k,m+1} \lambda_m [\mathcal{A} \otimes \Gamma(0)] \omega_m(1)
\]

\[
= \sum_{m=1}^{k} \Phi_{k,m+1} [\mathcal{A} \otimes \Gamma(0)] (\theta_m - \theta_{m-1}) \]

\[
= [\mathcal{A} \otimes \Gamma(0)] (\theta_k - \sum_{m=1}^{k} (\Phi_{k,m+1} - \Phi_{k,m}) (\theta_m - \theta_{m-1})) \]

\[
+ \sum_{m=1}^{k} (\Phi_{k,m+1} - \Phi_{k,m}) [\mathcal{A} \otimes \Gamma(0)] (\theta_m - \theta_{m-1}) \]

\[
- \sum_{m=1}^{k} (\Phi_{k,m+1} - \Phi_{k,m}) [\mathcal{A} \otimes \Gamma(0)] (\theta_m - \theta_{m-1}).
\]

It is obvious that \( \|\mathcal{A} \otimes \Gamma(0)\| \theta_k - [\mathcal{A} \otimes \Gamma(0)] (\theta_m - \theta_{m-1}) \|	o 0 \) as \( k \to \infty \). For the last term at the right-hand side of the above equation, we have

\[
\sum_{m=1}^{k} (\Phi_{k,m+1} - \Phi_{k,m}) [\mathcal{A} \otimes \Gamma(0)] (\theta_m - \theta_{m-1})
\]

\[
= \left( \sum_{m=1}^{k_3} + \sum_{m=k_3+1}^{k} \right) (\Phi_{k,m+1} - \Phi_{k,m}) \times [\mathcal{A} \otimes \Gamma(0)] (\theta_{m-1} - \theta_{m-2}).
\]

It is clear that \( \sum_{m=1}^{k_3} ||(\Phi_{k,m+1} - \Phi_{k,m}) (\mathcal{A} \otimes \Gamma(0)) (\theta_{m-1} - \theta_{m-2})|| \to 0 \) as \( k \to \infty \). Due to the fact that \( m \) is bounded by \( k_3 \), we have \( \|\Phi_{m,m}\| \to 0 \) as \( k \to \infty \). The rest part can be proved converging to \( 0 \) by using the same step as in Theorem 3.1. In short, the convergence of \( \delta u_k(t) \to 0 \) is proved. The validation of the initial step is thus completed.

The Inductive step can be completed using completely the same steps in Theorem 3.1. This completes the proof.

**Proof of Theorem 4.1:** The proof can be performed similarly to the proof of Theorem 3.1, if we can establish the same estimation (A4) for the time-switching topologies case. In fact, the recursion for the switching topology case is as follows,

\[
u_{k+1}(t) = u_k(t) + \lambda_k([\mathcal{L}(t) + \mathcal{D}(t)] \otimes \Gamma(t)) Y_k(t) \\
\times [I_n \otimes (C(t + 1)B(t))] \delta u_k(t) \\
+ \lambda_k([\mathcal{L}(t) + \mathcal{D}(t)] \otimes \Gamma(t)) Y_k(t) \times [I_n \otimes (C(t + 1)A(t))] \delta x_k(t) \\
- \lambda_k([\mathcal{L}(t) + \mathcal{D}(t)] \otimes \Gamma(t)) Y_k(t) \omega_k(t + 1) \\
+ \lambda_k([\mathcal{A}(t) \otimes \Gamma(t)]) \omega_k(t + 1). \quad (A12)
\]

Thus, similar to \( \Phi_{m,m} \), we define \( \Psi_{m,m} \) for the time-switching topologies case, taking the initial step \( t = 0 \) as example,

\[
\Psi_{m,m} \triangleq [I - \lambda_1([\mathcal{L}(0) + \mathcal{D}(0)] \otimes \Gamma(0))] Y_0(0)
\]
\[ \times [I_n \otimes (C(1)B(0))] \cdots \]
\[ [I - \lambda_m ([L(0) + D(0)) \otimes \Gamma(0)] \nu_m(0) \]
\[ \times [I_n \otimes (C(1)B(0))] \]

when \( l \geq m \) and \( \Psi_{l+1} \equiv I \).

Note that all eigenvalues of \( [([\mathcal{L}(t) + D(t)) \otimes (\Gamma(t)C(t + 1)B(t))] \) are with positive real parts. Therefore, there exists a positive-definite matrix \( P(0) \) such that
\[ - [([L(0) + D(0)) \otimes \Gamma(0)] \nu_k(0)[I_n \otimes (C(1)B(0))]^T P(0) \]
\[ - P(0) ([L(0) + D(0)) \otimes \Gamma(0)] \nu_k(0) \]
\[ \times [I_n \otimes (C(1)B(0))] \leq -\nu' I \]

for some suitable constant \( \nu' > 0 \).

We denote \( W_k \equiv -([L(0) + D(0)) \otimes \Gamma(0)] \nu_k(0)[I_n \otimes (C(1)B(0))]. \)

Thus, for positive integers \( l \) and \( m \) with \( l \geq m \), we have
\[ \Psi_{l,m}^T P(0)\Psi_{l,m} \]
\[ = \Psi_{l-1,m}^T (I + \lambda_l W_l)^T P(0)(I + \lambda_l W_l) \Psi_{l-1,m} \]
\[ = \Psi_{l-1,m}^T (P(0) + \lambda_l^2 W_l^T P(0) W_l \]
\[ + \lambda_l W_l^T P(0) + \lambda_l P(0) W_l) \Psi_{l-1,m} \]
\[ \leq \Psi_{l-1,m}^T (P(0) + \lambda_l^2 W_l^T P(0) W_l - \nu' \lambda_l I) \Psi_{l-1,m} \]
\[ = \Psi_{l-1,m}^T P^{1/2}(0)(I - \nu' \lambda_l P^{-1}(0) \]
\[ + \lambda_l^2 P^{-1/2}(0) W_l^T P(0) W_l P^{-1/2}(0)) P^{1/2}(0) \Psi_{l-1,m} \].

(A13)

Without loss of generality, when the iteration number \( m \) is sufficiently large, we have
\[ \|I - \nu' \lambda_l P^{-1}(0) + \lambda_l^2 P^{-1/2}(0) W_l^T P(0) W_l P^{-1/2}(0)\| \]
\[ \leq 1 - 2c_4 \lambda_{l+1} \leq \exp(-2c_4 \lambda_{l+1}), \]

(A14)

where \( c_4 > 0 \) is a suitable constant and the basic inequality \( 1 - x \leq e^{-x} \) is applied.

Noticing the boundedness of \( \nu P^{-1/2}(0) W_l^T P(0) W_l P^{-1/2}(0) \)

and combining (A13) and (A14) leads to
\[ \Psi_{l,m}^T P(0)\Psi_{l,m} \leq c_5 \exp \left( -2c_4 \sum_{k=m}^l \lambda_k \right) \]

(A15)

for some suitable \( c_5 > 0 \). Thus,
\[ \|\Psi_{l,m}\| \leq \lambda_{\min}^{-1/2}(P(0)) \sqrt{c_5} \exp \left( -c_4 \sum_{k=m}^l \lambda_k \right) \]

The rest of the proof can be completed similarly to Theorems 3.1 and 3.2. This completes the proof. □