2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Trackability-Based Distributed Learning Control for Multi-Agent Systems Under Switching Topologies

Yuxin Wu, Deyuan Meng, and Jing Wang

***Abstract*— This paper aims to address the distributed learning control problem for irregular multi-agent systems subject to switching topologies. The cooperative trackability property for the desired reference is discussed, which ensures the existence of the desired inputs for realizing the cooperative perfect tracking objective. Then, a trackability-based distributed learning control algorithm is presented with the integration of the complete experience information from the previous iteration. It is shown that for the cooperatively trackable desired reference, all agents learn to achieve the cooperative perfect tracking objective in the presence of the developed distributed learning control algorithm despite their irregular dynamics, provided that their associated directed graphs jointly have a spanning tree. The simulation is implemented to illustrate the validity of the trackability-based distributed learning control algorithm.**

***Index Terms*— Cooperative perfect tracking, cooperative trackability property, distributed learning control, multi-agent system, switching topology.**

I. INTRODUCTION

Distributed learning control problems have become one of the most popular topics for multi-agent systems, which adopt the cooperative learning of agents from the past experiences to effectively improve their transient performances (see, e.g., [1]–[3]). The distributed learning control mechanism merges the advantages of the distributed control mechanism based on the local information usage and the learning control mechanism based on the experience information usage. As a result, all agents can learn from the experiences of both themselves and other agents iteration by iteration to establish the desired transient performances over the finite time interval of interest subject to the repetitive tasks. This may promote most steady-state cooperative tracking results of multi-agent systems from the perspective of the control precision (see, e.g., [4]). The distributed learning control idea has been applied to solving many practical engineering problems with the high-precision collaborative requirements, such as the satellite formation [5] and the quadrotor tracking [6].

For the design of distributed learning control algorithms, the most widely adopted idea is to leverage the local neighbor information for one time instant from the previous iteration to establish a class of the P-type distributed learning control algorithms (see, e.g., [1], [2]). The P-type distributed learning control algorithms may effectively work for the cooperative perfect tracking task of multi-agent systems when the inputs directly impact the outputs (see, e.g., [7]). Nonetheless, when the inputs indirectly affect the outputs, the P-type distributed learning control algorithms may apply to multi-agent systems only having the quasi-regular dynamics with the first nonzero Markov parameter matrices of full rank due to their limited information utilization. For the implementation of distributed learning control algorithms, the fixed topologies are generally concerned in, e.g., [1]–[3]. But owing to the communication limitations in some engineering applications, the interactions among agents may not always be invariable. Therefore, some affords have been made to exploit some distributed learning control algorithms under switching topologies (see, e.g., [8]–[10]), which, however, may have some specific connectivity requirements at every iteration. These connectivity conditions may be too strict. Besides, the selection on the gain matrices for distributed learning control algorithms in, e.g., [8]–[10] may rely heavily on the global information of their associated graphs. This makes them not in the fully distributed manners and hence constrains their applicability. Of particular note is that in most of the existing distributed learning control results, the implementability of the cooperative perfect tracking objective for multi-agent systems with respect to the desired reference is generally assumed, which may not be reasonable.

Motivated by the above discussions, we intend to develop some effective distributed learning control algorithm with the wider applicability by focusing on the essential properties of multi-agent systems. The cooperative trackability property of the desired reference is first revealed, which ensures the implementability of the cooperative perfect tracking objective for multi-agent systems. Moreover, we develop a trackability-based distributed learning control algorithm under the usage of the complete local information from the previous iteration. The cooperative perfect tracking objective can be established for multi-agent systems subject to the cooperatively trackable desired reference, provided the joint spanning tree condition holds. Due to the introduction of the cooperative trackability property, the proposed distributed learning control algorithm applies to any linear multi-agent system in spite of whether it is regular or not, which improves the applicability of those in, e.g., [1]–[3]. Besides, it only imposes a joint connectivity requirement without the utilization of the global information for the gain matrix selection, which has more relaxed design conditions than those in, e.g., [8]–[10].

We organize the rest of this paper as follows. In Section II,

Y. Wu and D. Meng are with the Seventh Research Division, Beihang University (BUAA), Beijing 100191, P. R. China, and also with the School of Automation Science and Electrical Engineering, Beihang University (BUAA), Beijing 100191, P. R. China (e-mail: wuyuxin@buaa.edu.cn, dymeng@buaa.edu.cn).

J. Wang is with the School of Electrical and Control Engineering, North China University of Technology, Beijing 100144, P. R. China (e-mail: jwang@ncut.edu.cn).

This work was supported in part by the National Natural Science Foundation of China under Grant 62273018 and in part by the Science and Technology on Space Intelligent Control Laboratory under Grant HTKJ2022KL502006.

we introduce some preliminaries on graphs and then present our concerned problem. In Section III, the cooperative trackability property is proposed, with which a distributed learning control algorithm is exploited. We give a simulation example in Section IV and make some conclusions in Section V.

Notations: We denote $\mathbb{Z}=\{0,1,\cdots\}$, $\mathbb{Z}_N=\{0,1,\cdots,N\}$, $\mathscr{I}_n=\{1,2,\cdots,n\}$, and $1_n=[1,1,\cdots,1]^{\mathrm{T}}\in\mathbb{R}^n$. Let $I_n\in\mathbb{R}^{n\times n}$ and $\operatorname{diag}\{d_1,d_2,\cdots,d_n\}\in\mathbb{R}^{n\times n}$ denote an identity matrix and a diagonal matrix with its diagonal entries as $d_1,d_2,\cdots,d_n$, respectively. Given any matrix $A=[a_{ij}]\in\mathbb{R}^{m\times n}$, let $\operatorname{span}(A)$ be its image space. Particularly for $m=n$, $\lambda_i(A)$ represents its eigenvalue having the ith maximum modulus. $\otimes$ describes the Kronecker product of two matrices.

II. Preliminaries and Problem Statement

In this section, we first discuss some necessary preliminaries on directed graphs and then show our concerned problem.

A. Directed Graphs

A switching directed graph is represented by a triple $\mathscr{G}_k=\{\mathscr{V},\mathscr{E}_k,\mathscr{A}_k\}$, where $\mathscr{V}=\{v_i:i\in\mathscr{I}_n\}$, $\mathscr{E}_k=\{(v_j,v_i):i,j\in\mathscr{I}_n\}$, and $\mathscr{A}_k=\left[\alpha_{ij,k}\right]^{n\times n}\geq 0$ are the node set, the edge set, and the adjacency matrix, respectively. If the node v_i receives the information from the node v_j, then $(v_j,v_i)\in\mathscr{E}_k$ follows with $\alpha_{ij,k}>0$ and v_j is called a neighbor of v_i. The neighbor set of v_i is denoted as $\mathscr{N}_{i,k}=\left\{v_j:(v_j,v_i)\in\mathscr{E}_k\right\}$. Let $\mathscr{G}_k$ have no self-loops, that is, $\alpha_{ii,k}\equiv 0$, $\forall k\in\mathbb{Z}$, $i\in\mathscr{I}_n$. A directed path from the node v_j to the node v_i is an edge sequence composed of some distinct nodes as $\left\{(v_j,v_{k_1}),(v_{k_1},v_{k_2}),\cdots,(v_{k_{l-1}},v_i)\right\}$. If there exists a node v_i with directed paths to all other nodes, then $\mathscr{G}_k$ is said to have a spanning tree, for which v_i is called the root node of $\mathscr{G}_k$. We assume that $\mathscr{G}_k$ switch among finite directed graphs. If there exists an infinite iteration sequence $\{k_i:k_0=0,0<k_{i+1}-k_i\leq\tau,\forall i\in\mathbb{Z}\}$ for some finite integer $\tau>0$ such that the union directed graph $\bigcup_{p=k_i}^{k_{i+1}-1}\mathscr{G}_p$, $\forall i\in\mathbb{Z}$ has a spanning tree, then $\mathscr{G}_k$ is said to jointly have a spanning tree, where the root node of $\bigcup_{p=k_i}^{k_{i+1}-1}\mathscr{G}_p$, $\forall i\in\mathbb{Z}$ is called the union root node. The Laplacian matrix of $\mathscr{G}_k$ is defined as

$$L_k=[l_{ij,k}]\in\mathbb{R}^{n\times n}=\begin{cases}\sum\limits_{v_j\in\mathscr{N}_{i,k}}\alpha_{ij,k}, & i=j\\ -\alpha_{ij,k}, & i\neq j.\end{cases}$$

B. Problem Statement

We consider a multi-agent system consisting of n agents in the presence of a switching directed graph $\mathscr{G}_k$, which evolves simultaneously along the time axis $t\in\mathbb{Z}_N$ and the iteration axis $k\in\mathbb{Z}$. The dynamics of the ith agent are described by

$$\begin{cases}x_{i,k}(t+1)=Ax_{i,k}(t)+Bu_{i,k}(t)\\ y_{i,k}(t)=Cx_{i,k}(t)\end{cases},\quad\forall t\in\mathbb{Z}_N,k\in\mathbb{Z},i\in\mathscr{I}_n \tag{1}$$

where $x_{i,k}(t)\in\mathbb{R}^s$, $u_{i,k}(t)\in\mathbb{R}^q$, and $y_{i,k}(t)\in\mathbb{R}^p$ are the state, input, and output of the ith agent; and A, B, and C are some system matrices with compatible dimensions. Without loss of generality, let the relative degree of all agents be $r=1$ such that $CB\neq 0$ is fulfilled and the initial states of all agents be iteration-invariant as $x_{i,k}(0)=0$, $\forall k\in\mathbb{Z}$, $i\in\mathscr{I}_n$ (for the case $x_{i,k}(0)=x_0\neq 0$, $\forall k\in\mathbb{Z}$, $i\in\mathscr{I}_n$, the following analysis is still effective by replacing $y_{i,k}(t)$ with $y_{i,k}(t)-CA^tx_0$ instead).

For the multi-agent system (1), this paper targets to design some distributed learning control algorithm such that for the given desired reference $y_d(t)\in\mathbb{R}^p$, $t\in\mathbb{Z}_N$, which is available only to a portion of agents, the cooperative perfect tracking objective can be realized, namely,

$$\lim_{k\to\infty}y_{i,k}(t)=y_d(t),\quad\forall t\in\mathbb{Z}_N\setminus\{0\},i\in\mathscr{I}_n. \tag{2}$$

Note that for most of the existing distributed learning control results in, e.g., [1]–[3], the implementability of the cooperative perfect tracking objective (2) subject to the given desired reference is directly supposed, the reasonability of which is doubtful. Besides, the full-row or full-column rank condition on the first nonzero Markov parameter matrix CB is generally required, which may limit the applicability of the distributed learning control algorithms. To overcome the aforementioned limitations, we first focus on exploring under what conditions the multi-agent system (1) is able to achieve the cooperative perfect tracking objective (2) for the given desired reference and then develop some effective distributed learning control algorithm to realize the cooperative tracking objective (2).

III. Main Results

In this section, we first discuss the cooperative trackability property of the desired reference for the multi-agent system (1), which guarantees the implementability of the cooperative perfect tracking objective (2). Given the cooperatively trackable desired reference, we propose some distributed learning control algorithm to achieve the cooperative perfect tracking objective (2) for the multi-agent system (1).

A. Cooperative Trackability Property

We first show the definition of the cooperative trackability property of the desired reference $y_d(t)$, $t\in\mathbb{Z}_N$ for the multi-agent system (1) as follows.

Definition 1: The desired reference $y_d(t)$, $t\in\mathbb{Z}_N$ is said to be cooperatively trackable for the multi-agent system (1) if for any agent, there exists some desired input $u_{i,d}(t)$, $t\in\mathbb{Z}_N$, $i\in\mathscr{I}_n$ such that under the initial state $x_{i,d}(0)=0$, $\forall i\in\mathscr{I}_n$,

$$\begin{cases}x_{i,d}(t+1)=Ax_{i,d}(t)+Bu_{i,d}(t)\\ y_d(t+1)=Cx_{i,d}(t+1)\end{cases},\quad\forall t\in\mathbb{Z}_{N-1},i\in\mathscr{I}_n. \tag{3}$$

The cooperative trackability property implies the existence of some desired input generating the desired reference for the multi-agent system (1), which ensures the implementability of the cooperative perfect tracking objective (2). Before we explore the distributed learning control algorithm design for the multi-agent system (1), it is necessary to validate whether the desired reference is cooperatively trackable. Otherwise, it is impossible to accomplish the cooperative perfect tracking objective (2) despite any inputs.

For exploiting the cooperative trackability property of the given desired reference, we may resort to the lifting form of the multi-agent system (1). To be specific, we, respectively, denote the supervectors of the input and the output for the ith agent as $\boldsymbol{U}_{i,k}=\left[u_{i,k}^{\mathrm{T}}(0),u_{i,k}^{\mathrm{T}}(1),\cdots,u_{i,k}^{\mathrm{T}}(N-1)\right]^{\mathrm{T}}$ and $\boldsymbol{Y}_{i,k}=\left[y_{i,k}^{\mathrm{T}}(1),y_{i,k}^{\mathrm{T}}(2),\cdots,y_{i,k}^{\mathrm{T}}(N)\right]^{\mathrm{T}}$. Then, by leveraging the lifting technique (see, e.g., [11], [12]), the dynamics of the ith agent for the multi-agent system (1) can be transformed into

$$\boldsymbol{Y}_{i,k}=\boldsymbol{G}\boldsymbol{U}_{i,k},\quad \forall k\in\mathbb{Z},i\in\mathscr{I}_n \tag{4}$$

where $\boldsymbol{G}$ is the block Toeplitz matrix in the form of

$$\boldsymbol{G}=\begin{bmatrix} CB & 0 & \cdots & 0\\ CAB & CB & \ddots & \vdots\\ \vdots & \vdots & \ddots & 0\\ CA^{N-1}B & CA^{N-2}B & \cdots & CB\end{bmatrix}.$$

As a result, the cooperative perfect tracking objective (2) can be correspondingly transformed into

$$\lim_{k\to\infty}\boldsymbol{Y}_{i,k}=\boldsymbol{Y}_d,\quad \forall i\in\mathscr{I}_n \tag{5}$$

for $\boldsymbol{Y}_d=\left[y_d^{\mathrm{T}}(1),y_d^{\mathrm{T}}(2),\cdots,y_d^{\mathrm{T}}(N)\right]^{\mathrm{T}}$. Clearly, (4) yields the direct input-to-output relation for the ith agent of the multi-agent system (1). It is obvious to validate from (4) that owing to the homogeneity of all the agents, whether the cooperative perfect tracking objective (5) (or (2)) can be achieved is only dependent on the relation between the block Toeplitz matrix $\boldsymbol{G}$ and the desired reference $\boldsymbol{Y}_d$. Next, we present a criterion to verify the cooperative trackability property of the desired reference in the following theorem.

Theorem 1: The desired reference $y_d(t)$, $t\in\mathbb{Z}_N$ is cooperatively trackable for the multi-agent system (1) if and only if its related linear algebraic equation

$$\boldsymbol{Y}_d=\boldsymbol{G}\boldsymbol{U}_d \tag{6}$$

is solvable.

Proof: We consider the lifting form of the multi-agent system (1) as (4). By Definition 1, the desired reference $y_d(t)$, $t\in\mathbb{Z}_N$ is cooperatively trackable if and only if some desired input supervectors $\boldsymbol{U}_{i,d}=\left[u_{i,d}^{\mathrm{T}}(0),u_{i,d}^{\mathrm{T}}(1),\cdots,u_{i,d}^{\mathrm{T}}(N-1)\right]^{\mathrm{T}}$, $i\in\mathscr{I}_n$ exist to ensure

$$\boldsymbol{Y}_d=\boldsymbol{G}\boldsymbol{U}_{i,d},\quad \forall i\in\mathscr{I}_n$$

which is equivalent to the solvability of the linear algebraic equation (6). The proof of this theorem is complete. ■

Remark 1: Thanks to the homogeneous dynamics and the same initial states of all agents in the multi-agent system (1), the criterion on the cooperative trackability property of the desired reference is similar to the trackability criterion for a learning control system in, e.g., [13], [14] regardless of the existence of multiple interactive systems. Based on Theorem 1, we only need to check whether $\boldsymbol{Y}_d\in\operatorname{span}(\boldsymbol{G})$ is satisfied to verify whether the given desired reference is cooperatively trackable for the multi-agent system (1). Once $\boldsymbol{Y}_d\in\operatorname{span}(\boldsymbol{G})$ holds, we proceed to develop the distributed learning control algorithm to reach the cooperative perfect tracking objective (2). Otherwise, the cooperative perfect tracking objective (2) can never be established in spite of any inputs. In particular, if $\boldsymbol{G}$ is of full-row rank, the cooperative trackability property of any desired reference can always be ensured.

B. Cooperative Perfect Tracking

When the desired reference is validated to be cooperatively trackable with Theorem 1, some distributed learning control algorithms can be established for the multi-agent system (1) to achieve the cooperative perfect tracking objective (2). It is worth noticing that for the design of most distributed learning control algorithms, the full-row or full-column rank property on the block Toeplitz matrix $\boldsymbol{G}$ is generally needed, i.e., the dynamics of agents have to be quasi-regular (see, e.g., [1]–[3]). As a benefit, the partial information from the previous iteration is only required at every time instant to exploit the P-type distributed learning control algorithms. But if $\boldsymbol{G}$ does not have full-row rank nor have full-column rank, the P-type distributed learning control algorithms may not be applicable anymore. Thus, we adopt the complete information collected from the previous iteration to develop the distributed learning control algorithm as

$$u_{i,k+1}(t)=u_{i,k}(t)+\gamma_{i,k}\sum_{l=1}^{N}K_{t+1,l}\Big\{\sum_{v_j\in\mathscr{N}_{i,k}}\alpha_{ij,k}\left[y_{j,k}(l)-y_{i,k}(l)\right] + d_{i,k}\left[y_d(l)-y_{i,k}(l)\right]\Big\},\forall t\in\mathbb{Z}_{N-1},k\in\mathbb{Z},i\in\mathscr{I}_n \tag{7}$$

where $K_{t,l}\in\mathbb{R}^{q\times p}$ and $\gamma_{i,k}>0$ denote the gain matrix and the gain parameter to be designed, respectively; and $\alpha_{ij,k}\geq 0$ and $d_{i,k}\geq 0$ denote the adjacency weight in $\mathscr{G}_k$ and the interaction for the desired reference and the ith agent, respectively. Let $\Gamma_k\triangleq\operatorname{diag}\left\{\gamma_{1,k},\gamma_{2,k},\cdots,\gamma_{n,k}\right\}$ and $\mathscr{D}_k\triangleq\left[d_{1,k},d_{2,k},\cdots,d_{n,k}\right]^{\mathrm{T}}$. We construct an enlarged directed graph $\widetilde{\mathscr{G}_k}=\left\{\widetilde{\mathscr{V}},\widetilde{\mathscr{E}_k},\widetilde{\mathscr{A}_k}\right\}$ with the desired reference involved as a virtual node v_0 such that it follows $\widetilde{\mathscr{V}}=\{v_0\cup\mathscr{V}\}$, $\widetilde{\mathscr{E}_k}=\left\{\mathscr{E}_k^0\cup\mathscr{E}_k\right\}$ with $\mathscr{E}_k^0$ in the form of $\mathscr{E}_k^0=\left\{(v_0,v_i):d_{i,k}\neq 0\right\}$, and

$$\widetilde{\mathscr{A}_k}=\left[\widetilde{\alpha}_{ij,k}\right]=\begin{bmatrix}0 & 0\\ \mathscr{D}_k & \mathscr{A}_k\end{bmatrix}.$$

Due to the unsatisfaction of the full rank condition on the block Toeplitz matrix $\boldsymbol{G}$, there exist some static (redundant) input-to-output channels. Thus, it may not be easy to directly explore the selection condition on the gain matrix based on the Markov parameter matrix CB. Instead, we may resort to the decomposition of the multi-agent system (1) and dig out the active input-to-output channels to investigate the selection condition on the gain matrix. Towards this end, we assume $m\triangleq\operatorname{rank}(\boldsymbol{G})\leq\min\{Np,Nq\}$ without loss of any generality. Then, we can construct some matrices $\boldsymbol{P}_1\in\mathbb{R}^{Np\times m}$ and $\boldsymbol{P}_2\in\mathbb{R}^{Np\times(Np-m)}$ such that $\boldsymbol{P}\triangleq[\boldsymbol{P}_1,\boldsymbol{P}_2]\in\mathbb{R}^{Np\times Np}$ is nonsingular and $\operatorname{span}(\boldsymbol{P}_1)=\operatorname{span}(\boldsymbol{G})$ is fulfilled, with which denote the inverse of $\boldsymbol{P}$ as $\boldsymbol{F}\triangleq\left[\boldsymbol{F}_1^{\mathrm{T}},\boldsymbol{F}_2^{\mathrm{T}}\right]^{\mathrm{T}}$ for $\boldsymbol{F}_1\in\mathbb{R}^{m\times Np}$ and $\boldsymbol{F}_2\in\mathbb{R}^{(Np-m)\times Np}$. Thanks to the introduction of $\boldsymbol{P}$ and $\boldsymbol{F}$, we can provide the cooperative perfect tracking result for the multi-agent system (1) in the following theorem.

Theorem 2: Let the distributed learning control algorithm (7) be applied to the multi-agent system (1), where the gain matrix $\boldsymbol{K}\triangleq[K_{ij}]\in\mathbb{R}^{Nq\times Np}$ is selected to fulfill

$$\begin{cases}0<\lambda_i(\boldsymbol{F}_1\boldsymbol{G}\boldsymbol{K}\boldsymbol{P}_1)\leq 1,\quad \forall i\in\mathscr{I}_m\\ \boldsymbol{F}_1\boldsymbol{G}\boldsymbol{K}\boldsymbol{P}_1=(\boldsymbol{F}_1\boldsymbol{G}\boldsymbol{K}\boldsymbol{P}_1)^{\mathrm{T}}\end{cases} \tag{8}$$

and the gain parameter $\gamma_{i,k}$ is selected to fulfill

$$\gamma_{i,k}\left(\sum_{v_j\in\mathcal{N}_{i,k}}\alpha_{ij,k}+d_{i,k}\right)<1,\quad \forall k\in\mathbb{Z}, i\in\mathscr{I}_n. \tag{9}$$

Given the cooperatively trackable desired reference $y_d(t)$, $t\in\mathbb{Z}_N$, the cooperative perfect tracking objective (2) is achieved if $\widetilde{\mathscr{G}}_k$ jointly has a spanning tree.

Proof: We consider the supervector $\boldsymbol{U}_{i,k}$ and can obtain from (7) that

$$\boldsymbol{U}_{i,k+1}=\boldsymbol{U}_{i,k}+\gamma_{i,k}\boldsymbol{K}\Big[\sum_{v_j\in\mathcal{N}_{i,k}}\alpha_{ij,k}\left(\boldsymbol{Y}_{j,k}-\boldsymbol{Y}_{i,k}\right)+d_{i,k}\left(\boldsymbol{Y}_d-\boldsymbol{Y}_{i,k}\right)\Big],\quad \forall k\in\mathbb{Z}, i\in\mathscr{I}_n.$$

For $\boldsymbol{U}_k=\left[\boldsymbol{U}_{1,k}^{\mathrm{T}},\boldsymbol{U}_{2,k}^{\mathrm{T}},\cdots,\boldsymbol{U}_{n,k}^{\mathrm{T}}\right]^{\mathrm{T}}$, we can further arrive at

$$\boldsymbol{U}_{k+1}=\boldsymbol{U}_k+(H_k\otimes\boldsymbol{K})\left(1_n\otimes\boldsymbol{Y}_d-\boldsymbol{Y}_k\right),\quad \forall k\in\mathbb{Z} \tag{10}$$

where $H_k=\Gamma_kL_k+\Gamma_k\mathrm{diag}\{d_{1,k},d_{2,k},\cdots,d_{n,k}\}$ with L_k as the Laplacian matrix of $\mathscr{G}_k$ and $\boldsymbol{Y}_k=\left[\boldsymbol{Y}_{1,k}^{\mathrm{T}},\boldsymbol{Y}_{2,k}^{\mathrm{T}},\cdots,\boldsymbol{Y}_{n,k}^{\mathrm{T}}\right]^{\mathrm{T}}$. By noting (4), it is easy to derive from (10) that

$$\boldsymbol{Y}_{k+1}=\boldsymbol{Y}_k+(H_k\otimes\boldsymbol{GK})\left(1_n\otimes\boldsymbol{Y}_d-\boldsymbol{Y}_k\right),\quad \forall k\in\mathbb{Z}.$$

We denote the tracking error supervector as $\boldsymbol{E}_k=1_n\otimes\boldsymbol{Y}_d-\boldsymbol{Y}_k$ and then can obtain

$$\boldsymbol{E}_{k+1}=(I_{Npn}-H_k\otimes\boldsymbol{GK})\boldsymbol{E}_k,\quad \forall k\in\mathbb{Z}. \tag{11}$$

We perform a linear transformation for the system (11) such that $\overline{\boldsymbol{E}}_k=(I_n\otimes\boldsymbol{F})\boldsymbol{E}_k$ with $\overline{\boldsymbol{E}}_k^1=(I_n\otimes\boldsymbol{F}_1)\boldsymbol{E}_k$ and $\overline{\boldsymbol{E}}_k^2=(I_n\otimes\boldsymbol{F}_2)\boldsymbol{E}_k$. It is clear to see from (4) that

$$\begin{aligned}\overline{\boldsymbol{E}}_k^2&=(I_n\otimes\boldsymbol{F}_2)\left(1_n\otimes\boldsymbol{Y}_d-\boldsymbol{Y}_k\right)\\&=(I_n\otimes\boldsymbol{F}_2)\left[1_n\otimes\boldsymbol{Y}_d-(I_n\otimes\boldsymbol{G})\boldsymbol{U}_k\right].\end{aligned}$$

With Theorem 1, there exists some desired input $\boldsymbol{U}_d\in\mathbb{R}^{Nq}$ to ensure (6) for the cooperatively trackable desired reference $\boldsymbol{Y}_d$. Consequently, it further follows

$$\overline{\boldsymbol{E}}_k^2=(I_n\otimes\boldsymbol{F}_2)\left[1_n\otimes\boldsymbol{GU}_d-(I_n\otimes\boldsymbol{G})\boldsymbol{U}_k\right]=0,\quad \forall k\in\mathbb{Z} \tag{12}$$

where $\boldsymbol{F}_2\boldsymbol{G}=0$ is inserted thanks to $\mathrm{span}(\boldsymbol{P}_1)=\mathrm{span}(\boldsymbol{G})$ and $\boldsymbol{F}_2\boldsymbol{P}_1=0$. Based on (11) and (12), we can obtain

$$\begin{aligned}\overline{\boldsymbol{E}}_{k+1}^1&=\overline{\boldsymbol{E}}_k^1-(H_k\otimes\boldsymbol{F}_1\boldsymbol{GK})\boldsymbol{E}_k\\&=\overline{\boldsymbol{E}}_k^1-(H_k\otimes\boldsymbol{F}_1\boldsymbol{GK})\\&\quad\times\left[(I_n\otimes\boldsymbol{P}_1)\overline{\boldsymbol{E}}_k^1+(I_n\otimes\boldsymbol{P}_2)\overline{\boldsymbol{E}}_k^2\right]\\&=\left[I_{mn}-(H_k\otimes\boldsymbol{F}_1\boldsymbol{GKP}_1)\right]\overline{\boldsymbol{E}}_k^1,\quad \forall k\in\mathbb{Z}.\end{aligned} \tag{13}$$

Since $\boldsymbol{K}$ is selected to satisfy (8), there exists some transformation matrix $\boldsymbol{T}\in\mathbb{R}^{m\times m}$ to render

$$\boldsymbol{T}(\boldsymbol{F}_1\boldsymbol{GKP}_1)\boldsymbol{T}^{-1}=\mathrm{diag}\{\lambda_1,\lambda_2,\cdots,\lambda_m\} \tag{14}$$

for $\lambda_i\triangleq\lambda_i(\boldsymbol{F}_1\boldsymbol{GKP}_1)$, $\forall i\in\mathscr{I}_m$. Then, we implement a linear transformation for the system (13) with $\widehat{\overline{\boldsymbol{E}}}_k^1=\boldsymbol{Q}(I_n\otimes\boldsymbol{T})\overline{\boldsymbol{E}}_k^1$, where $\boldsymbol{Q}=[\boldsymbol{e}_1,\boldsymbol{e}_2,\cdots,\boldsymbol{e}_m]$ for $\boldsymbol{e}_i=\left[e_i,e_{m+i},\cdots,e_{(n-1)m+i}\right]$, $\forall i\in\mathscr{I}_m$ with $e_j\in\mathbb{R}^{mn}$ as the jth column of I_{mn}. It is direct to deduce from (13) and (14) that

$$\widehat{\overline{\boldsymbol{E}}}_{k+1}^1=\left[I_{mn}-\begin{bmatrix}\lambda_1H_k&0&\cdots&0\\0&\lambda_2H_k&\ddots&\vdots\\\vdots&\ddots&\ddots&0\\0&\cdots&0&\lambda_mH_k\end{bmatrix}\right]\widehat{\overline{\boldsymbol{E}}}_k^1,\quad \forall k\in\mathbb{Z}.$$

We denote $\widehat{\overline{\boldsymbol{E}}}_k^1=\left[\left(\widehat{\overline{\boldsymbol{E}}}_{1,k}^1\right)^{\mathrm{T}},\left(\widehat{\overline{\boldsymbol{E}}}_{2,k}^1\right)^{\mathrm{T}},\cdots,\left(\widehat{\overline{\boldsymbol{E}}}_{m,k}^1\right)^{\mathrm{T}}\right]^{\mathrm{T}}$ with $\widehat{\overline{\boldsymbol{E}}}_{i,k}^1\in\mathbb{R}^n$, $\forall i\in\mathscr{I}_m$. As a result, we further have

$$\widehat{\overline{\boldsymbol{E}}}_{i,k+1}^1=(I_n-\lambda_iH_k)\widehat{\overline{\boldsymbol{E}}}_{i,k}^1,\quad \forall k\in\mathbb{Z}, i\in\mathscr{I}_m. \tag{15}$$

By resorting to (8) and (9), it is obvious that $I_{n+1}-\lambda_i\widetilde{\Gamma}_k\widetilde{L}_k$ is a stochastic matrix for any $k\in\mathbb{Z}$ and $i\in\mathscr{I}_m$, where $\widetilde{\Gamma}_k=\mathrm{diag}\{1,\Gamma_k\}$ and $\widetilde{L}_k$ is the Laplacian matrix of $\widetilde{\mathscr{G}}_k$. Since $\widetilde{\mathscr{G}}_k$ jointly has a spanning tree, it follows from [15] that

$$\lim_{k\to\infty}\left(I_{n+1}-\lambda_i\widetilde{\Gamma}_k\widetilde{L}_k\right)^k=1_{n+1}f_i^{\mathrm{T}},\quad \forall i\in\mathscr{I}_m$$

for some $f_i\in\mathbb{R}^{n+1}$. In addition, owing to the specific form of $\widetilde{\Gamma}_k\widetilde{L}_k$ as

$$\widetilde{\Gamma}_k\widetilde{L}_k=\begin{bmatrix}0&0\\-\Gamma_k\mathscr{D}_k&H_k\end{bmatrix}$$

we can deduce

$$\left(I_{n+1}-\lambda_i\widetilde{\Gamma}_k\widetilde{L}_k\right)^k=\begin{bmatrix}1&0\\(*)&(I_n-\lambda_iH_k)^k\end{bmatrix},\quad \forall k\in\mathbb{Z}, i\in\mathscr{I}_m$$

with $(*)$ as some unknown matrix, which leads to

$$\lim_{k\to\infty}(I_n-\lambda_iH_k)^k=0,\quad \forall i\in\mathscr{I}_m.$$

Together with (15), it directly yields

$$\lim_{k\to\infty}\widehat{\overline{\boldsymbol{E}}}_{i,k}^1=\lim_{k\to\infty}(I_n-\lambda_iH_k)^k\widehat{\overline{\boldsymbol{E}}}_{i,0}^1=0,\quad \forall i\in\mathscr{I}_m$$

and thus, we have

$$\lim_{k\to\infty}\widehat{\overline{\boldsymbol{E}}}_k^1=0.$$

As a consequence, we can conclude

$$\begin{aligned}\lim_{k\to\infty}\boldsymbol{E}_k&=(I_n\otimes\boldsymbol{P}_1)\lim_{k\to\infty}\overline{\boldsymbol{E}}_k^1+(I_n\otimes\boldsymbol{P}_2)\lim_{k\to\infty}\overline{\boldsymbol{E}}_k^2\\&=(I_n\otimes\boldsymbol{P}_1\boldsymbol{T}^{-1})\boldsymbol{Q}^{-1}\lim_{k\to\infty}\widehat{\overline{\boldsymbol{E}}}_k^1\\&=0\end{aligned}$$

which implies that the cooperative perfect tracking objective (5) is achieved. Equivalently, the cooperative perfect tracking objective (2) is accomplished for the multi-agent system (1). The proof of this theorem is complete. ■

Remark 2: From Theorem 2, it shows that the cooperative perfect tracking objective (2) can be achieved for the multi-agent system (1) under the distributed learning control algorithm (7) only with the joint spanning tree requirement on its related directed graph $\widetilde{\mathscr{G}}_k$, provided the gain matrix $\boldsymbol{K}$ and the gain parameter $\gamma_{i,k}$ are properly selected. Note that the gain matrix $\boldsymbol{K}$ is selected identically for all the agents thanks to the homogeneity of them. Because $\boldsymbol{F}_1\boldsymbol{G}$ is of full-row rank, there always exists some gain matrix $\boldsymbol{K}$ to fulfill the selection condition (8). By contrast, the gain parameter $\gamma_{i,k}$ is selected differently, which only relies on the local information of each agent. It is applied to ensuring that $I_{n+1} - \lambda_i \widetilde{\Gamma}_k \widetilde{L}_k$, $\forall i \in \mathscr{I}_m$ is a stochastic matrix. Owing to the fact that the virtual node v_0 can not receive information from any other nodes, it is the only union root of $\widetilde{\mathscr{G}}_k$ under the joint spanning tree condition, which indicates that all the agents can directly or indirectly receive the information of the desired reference.

Remark 3: The distributed learning control algorithm (7) may have more general applicability than most of the existing distributed learning control algorithms. By comparison with the distributed learning control results for linear multi-agent systems in, e.g., [1]–[3], the full rank condition on $\boldsymbol{G}$ (or CB) is relaxed such that the distributed learning control algorithm applies to agents with the irregular dynamics. This attributes to the integration of the cooperative trackability property for the desired reference, which can be determined by Theorem 1. In contrast with the distributed learning control results for multi-agent systems subject to switching topologies in, e.g., [8]–[10], not only is the joint connectivity condition adopted, but also the selection on the gain matrix/parameter only relies on the local information instead of the global information of multi-agent systems.

The selection condition (8) may not be easy to validate for the selection of the gain matrix $\boldsymbol{K}$. In the following corollary, some feasible candidates for the gain matrix $\boldsymbol{K}$ are provided to ensure the realization of the cooperative perfect tracking.

Corollary 1: Let the distributed learning control algorithm (7) be applied to the multi-agent system (1), where the gain matrix is selected as

$$\boldsymbol{K} = \kappa (\boldsymbol{F}_1\boldsymbol{G})^{\mathrm{T}} \left[\boldsymbol{F}_1\boldsymbol{G}(\boldsymbol{F}_1\boldsymbol{G})^{\mathrm{T}}\right]^{-1} \left(\boldsymbol{P}_1^{\mathrm{T}}\boldsymbol{P}_1\right)^{-1} \boldsymbol{P}_1^{\mathrm{T}} \qquad (16)$$

for some $0 < \kappa \leq 1$ and the gain parameter $\gamma_{i,k}$ is selected to fulfill (9). Given the cooperatively trackable desired reference $y_d(t)$, $t \in \mathbb{Z}_N$, the cooperative perfect tracking objective (2) is achieved if $\widetilde{\mathscr{G}}_k$ jointly has a spanning tree.

Proof: Because $\boldsymbol{F}_1\boldsymbol{G}$ has full-row rank and $\boldsymbol{P}_1$ has full-column rank, the selection of $\boldsymbol{K}$ in (16) makes sense. Under (16), we can validate $\boldsymbol{F}_1\boldsymbol{G}\boldsymbol{K}\boldsymbol{P}_1 = \kappa I_m$, which, combined with $0 < \kappa \leq 1$, ensures the satisfaction of (8). Then, the result in this corollary follows directly from Theorem 2. ■

If the full rank property on $\boldsymbol{G}$ (or CB) is satisfied, we can simplify the design manner of the distributed learning control algorithm for the multi-agent system (1), which is disclosed in the following corollary.

Corollary 2: Let the distributed learning control algorithm

$$u_{i,k+1}(t) = u_{i,k}(t) + \gamma_{i,k} K \Big\{ \sum_{v_j \in \mathscr{N}_{i,k}} \alpha_{ij,k}\left[y_{j,k}(t+1) - y_{i,k}(t+1)\right] + d_{i,k}\left[y_d(t+1) - y_{i,k}(t+1)\right] \Big\}, \forall t \in \mathbb{Z}_{N-1}, k \in \mathbb{Z}, i \in \mathscr{I}_n$$

be applied to the multi-agent system (1) with its associated directed graph $\widetilde{\mathscr{G}}_k$ jointly having a spanning tree, where the gain parameter $\gamma_{i,k}$ fulfills (9). Then, given the cooperatively trackable desired reference, the cooperative perfect tracking objective (2) is reached if any of the conditions below holds:

1) CB is of full-row rank with the gain matrix $K \in \mathbb{R}^{q\times p}$ being selected to fulfill

$$\begin{cases} 0 < \lambda_i(CBK) \leq 1, \quad \forall i \in \mathscr{I}_p \\ CBK = (CBK)^{\mathrm{T}}; \end{cases}$$

2) CB is of full-column rank with the gain matrix $K \in \mathbb{R}^{q\times p}$ being selected to fulfill

$$\begin{cases} 0 < \lambda_i(KCB) \leq 1, \quad \forall i \in \mathscr{I}_q \\ KCB = (KCB)^{\mathrm{T}}. \end{cases}$$

Proof: The results in this corollary can be obtained by adopting the induction analysis method in, e.g., [7], together with the analysis ideas in the proof of Theorem 2. ■

Under the full rank condition on CB, the partial information from the previous iteration only needs to be leveraged for the development of the distributed learning control algorithm. According to Theorem 1, the full-row rank condition on CB ensures that any desired reference is cooperatively trackable. But for the case that CB has full-column rank, we still need to check whether the given desired reference is cooperatively trackable, which leads to a unique desired input.

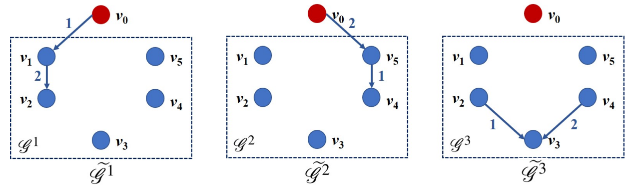

Fig. 1. Three candidate directed graphs $\left\{\widetilde{\mathscr{G}}^1, \widetilde{\mathscr{G}}^2, \widetilde{\mathscr{G}}^3\right\}$ for the switching of the directed graph $\widetilde{\mathscr{G}}_k$.

IV. Simulation Example

In this section, we perform a simulation example to show how to design the distributed learning control algorithm (7) for the multi-agent system (1).

We consider the multi-agent system (1) consisting of five agents, where the interactions among agents are described by a directed graph $\mathscr{G}_k$ switching among three candidate directed graphs $\left\{\mathscr{G}^1, \mathscr{G}^2, \mathscr{G}^3\right\}$ in Fig. 1. Let the system matrices of all agents be the same as

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & -1 \\ 2 & -2 \\ 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & -1 \end{bmatrix}$$

and the time interval of interest be $N = 100$. It is easy to see $\operatorname{rank}(CB) = 1$ such that the full rank condition on CB is not satisfied. Let the desired reference be

$$y_d(t) = [\sin(0.06t) + 1, 2\sin(0.06t) + 2]^{\mathrm{T}}, \quad \forall t \in \mathbb{Z}_{100}.$$

Under the zero initial states, we can validate that the desired reference is cooperatively trackable according to Theorem 1. Let the related directed graph $\widetilde{\mathscr{G}}_k$ of the multi-agent system (1) switch among three candidates in Fig. 1 to make $\widetilde{\mathscr{G}}_k = \widetilde{\mathscr{G}}^1$,

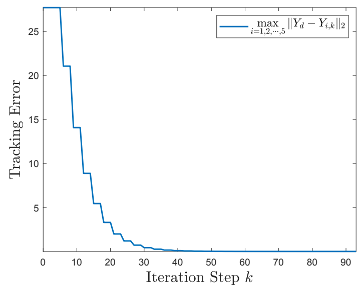

Fig. 2. The evolution of the tracking error norm $\max_{i\in\mathscr{I}_5} \|\boldsymbol{Y}_d - \boldsymbol{Y}_{i,k}\|_2$ along the iteration axis.

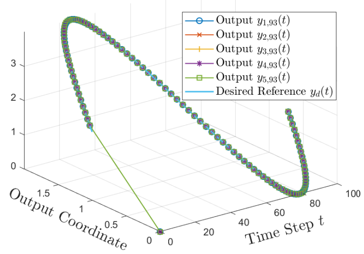

Fig. 3. The output evolutions of five agents at the 93rd iteration.

$\forall k = 3j$; $\widetilde{\mathscr{G}}_k = \widetilde{\mathscr{G}}^2$, $\forall k = 3j+1$; and $\widetilde{\mathscr{G}}_k = \widetilde{\mathscr{G}}^3$, $\forall k = 3j+2$ for any $j \in \mathbb{Z}$. Thus, $\widetilde{\mathscr{G}}_k$ jointly has a spanning tree. For designing the distributed learning control algorithm (7), we choose the odd columns of $\boldsymbol{G}$ to construct $\boldsymbol{P}_1 \in \mathbb{R}^{200\times 100}$, under which we take the even columns of I_{200} to construct $\boldsymbol{P}_2 \in \mathbb{R}^{200\times 100}$. Then, according to Corollary 1, we can select the gain matrix as $\boldsymbol{K} = (\boldsymbol{F}_1\boldsymbol{G})^{\mathrm{T}} \left[\boldsymbol{F}_1\boldsymbol{G}(\boldsymbol{F}_1\boldsymbol{G})^{\mathrm{T}}\right]^{-1} \left(\boldsymbol{P}_1^{\mathrm{T}}\boldsymbol{P}_1\right)^{-1}\boldsymbol{P}_1^{\mathrm{T}}$ and the gain parameter as $\gamma_{i,k} = 0.4$, $\forall i \in \mathscr{I}_5$ for $\widetilde{\mathscr{G}}_k \in \left\{\widetilde{\mathscr{G}}^1, \widetilde{\mathscr{G}}^2\right\}$ and $\gamma_{i,k} = 0.25$, $\forall i \in \mathscr{I}_5$ for $\widetilde{\mathscr{G}}_k \in \left\{\widetilde{\mathscr{G}}^3\right\}$, which ensure both (8) and (9).

We perform the simulation for the multi-agent system (1) under the distributed learning control algorithm (7) with the zero initial inputs for all agents, where the tolerance is chosen as 10^{-5} with regard to the tracking error norm $\max_{i\in\mathscr{I}_5} \|\boldsymbol{Y}_d - \boldsymbol{Y}_{i,k}\|_2$. We depict the evolution of the tracking error norm in Fig. 2, which indicates that the tracking error norm decreases into the tolerance at the 93rd iteration. Note that in Fig. 2, the tracking error norm is unchanged for some iterations owing to the nonpersistent interactions among agents. Thanks to the joint connectivity property of $\widetilde{\mathscr{G}}_k$, all agents can still achieve the cooperative perfect tracking objective (2). We also show the output evolutions of all five agents at the 93rd iteration in Fig. 3. From Fig. 3, it is clear that the cooperative perfect tracking objective (2) is accomplished in the presence of the distributed learning control algorithm (7). Hence, the validity of the results in Theorem 2 and Corollary 1 is illustrated.

V. Conclusions

In this paper, the distributed learning control problem has been considered for multi-agent systems subject to switching topologies. We have investigated the cooperative trackability property for the desired reference, which guarantees the implementability of the cooperative perfect tracking objective in the presence of some desired inputs. Given the cooperatively trackable desired reference, the full experience information from the previous iteration has been leveraged to develop a trackability-based distributed learning control algorithm. The cooperative perfect tracking objective is achieved for multi-agent systems under the proposed distributed learning control algorithm, provided that the joint spanning tree condition is fulfilled for their related directed graphs, where the ill effects from the irregular dynamics of agents can be overcome via the proper selection on the gain matrix. We have performed a simulation example to show the validity of the trackability-based distributed learning control algorithm.

References

[1] S. Yang, J.-X. Xu, and X. Li, "Iterative learning control with input sharing for multi-agent consensus tracking," *Systems & Control Letters*, vol. 94, pp. 97–106, Aug. 2016.

[2] C. Liu, D. Shen, and J. Wang, "Iterative learning control of multi-agent systems with random noises and measurement range limitations," *International Journal of Systems Science*, vol. 50, no. 7, pp. 1465–1482, May 2019.

[3] X. Zhao and Y. Wang, "Distributed point-to-point iterative learning control for multi-agent systems with quantization," *Journal of the Franklin Institute*, vol. 358, no. 13, pp. 6508–6525, Sept. 2021.

[4] J. Qin, Q. Ma, Y. Shi, and L. Wang, "Recent advances in consensus of multi-agent systems: A brief survey," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4972–4983, Jun. 2017.

[5] H.-S. Ahn, K. L. Moore, and Y. Chen, "Trajectory-keeping in satellite formation flying via robust periodic learning control," *International Journal of Robust and Nonlinear Control*, vol. 20, no. 14, pp. 1655–1666, Sept. 2010.

[6] A. Hock and A. P. Schoellig, "Distributed iterative learning control for a team of quadrotors," in *Proceedings of IEEE 55th Conference on Decision and Control*, Las Vegas, NV, USA, pp. 4640–4646, Dec. 12-14, 2016.

[7] D. Meng and K. L. Moore, "Learning to cooperate: Networks of formation agents with switching topologies," *Automatica*, vol. 64, pp. 278–293, Feb. 2016.

[8] X. Bu, J. Liang, Z. Hou, and R. Chi, "Data-driven terminal iterative learning consensus for nonlinear multiagent systems with output saturation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 5, pp. 1963–1973, May 2021.

[9] N. Lin, R. Chi, and B. Huang, "Event-triggered learning consensus of networked heterogeneous nonlinear agents with switching topologies," *Journal of the Franklin Institute*, vol. 358, no. 7, pp. 3803–3821, May 2021.

[10] J. Feng, W. Song, H. Zhang, and W. Wang, "Data-driven robust iterative learning consensus tracking control for MIMO multiagent systems under fixed and iteration-switching topologies," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 2, pp. 1331–1344, Feb. 2022.

[11] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control: A learning-based method for high-performance tracking control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[12] H.-S. Ahn, Y. Chen, and K. L. Moore, "Iterative learning control: Brief survey and categorization," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1099–1121, Nov. 2007.

[13] D. Meng and Y. Wu, "Control design for iterative methods in solving linear algebraic equations," *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5039–5054, Oct. 2022.

[14] D. Meng, "Feedback of control on mathematics: Bettering iterative methods by observer system design," *IEEE Transactions on Automatic Control*, vol. 68, no. 4, pp. 2498–2505, Apr. 2023.

[15] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655-661, May 2005.