SIAM J. CONTROL OPTIM.
Vol. 61, No. 2, pp. 543–559

LEARNING-ABILITY OF DISCRETE-TIME ITERATIVE LEARNING CONTROL SYSTEMS WITH FEEDFORWARD*

JIAN LIU†, XIAOE RUAN‡, YUANSHI ZHENG†, YINGMIN YI§, AND CONGSI WANG†

Abstract. This paper considers the learning-ability for discrete-time iterative learning control (ILC) systems with feedforward. More specifically, the relation between the output realizability and the feedforward matrix is first established. Then, the learning-ability of four ILC systems is considered. It is shown that the proportional type (P-type) update law can only ensure the fully asymptotic learning-ability. By only using the feedforward matrix, a more efficient point-wise P-type update law is developed, which can ensure the fully $(T+2)$-step learning-ability, where T is the trial length. In the case that the state is measurable and controllable, it is proven that the update law with current state feedback can ensure the fully monotone learning-ability and the fully 2-step learning-ability, respectively. In addition, by only using the output data at the previous trial, a full output feedback update law is proposed, which can respectively ensure the fully 2-step learning-ability and the fully monotonic learning-ability.

Key words. iterative learning control, output realizability, discrete-time systems, learning-ability, convergence performance

MSC codes. 93C55, 93B50, 93B52

DOI. 10.1137/22M1477258

1. Introduction.

Iterative learning control (ILC) technique is developed for systems repetitively operated over the finite time interval whose task is to track the given desired trajectory over the whole finite time interval. Since ILC can achieve the tracking task over the whole finite time interval with a satisfactory precision [1], it has been widely studied during the past three decades, and great progress has been made in both fundamental theories [2, 5, 10, 22] and practical applications [6, 7, 17].

The idea of ILC is to exploit the repetitive operation characteristics of control system and all the available information to design an update law for the system input, which can drive the control system to track the given desired trajectory. Here, the available information may include the known part of system dynamics, the input data, the measurable output data, and the measurable state data. Clearly, if it is expected that the ILC update law designed can drive the control system to track the given desired trajectory with any given precision, there must exist at least a desired initial state and a desired input such that supplying them to the control

*Received by the editors February 10, 2022; accepted for publication (in revised form) October 16, 2022; published electronically April 17, 2023.
https://doi.org/10.1137/22M1477258
Funding: This work was funded by the National Natural Science Foundation of China (62273267 and 51975447), National Key Research and Development Program of China (2021YFC2203600), the Fundamental Research Funds for the Central Universities (XJS210402), the Key R and D plan of Shaanxi Province (2020ZDLGY06-01), Science and Technology Innovation Guidance Project of Xi'an, China (21JY033), and the Natural Science Basic Research Program of Shaanxi, China (2022JC-46).

†Shaanxi Key Laboratory of Space Solar Power Station System, School of Mechano-electronic Engineering, Xidian University, Xi'an, 710071, China (liujianzym@outlook.com, zhengyuanshi2005@163.com, congsiwang@163.com).

‡School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, 710049, China (wruanxe@mail.xjtu.edu.cn).

§Faculty of Automation and Information Engineering, Xi'an University of Technology, Xi'an, 710049, China (yiym@xaut.edu.cn).

system yields the given desired trajectory, i.e., the given desired trajectory must be realizable over the whole finite time interval [14]. This means that the study of output realizability is crucial for ILC systems. On the output realizability, one can see that the existing literature falls into two categories. Some literature directly assumes that the given desired trajectory is realizable [3, 9, 20, 21], while others directly ignore the realizability of the given desired trajectory [5, 16, 18]. The main reason for this phenomenon is the technical route of convergence analysis for the system output sequence. When the indirect technical route is adopted, i.e., the convergence of the output sequence follows from the convergence of the input sequence, it requires the desired state and output equations to get the convergence of the input sequence. This is the immediate cause of the hypothesis that the desired trajectory is realizable. However, when we directly conduct the convergence analysis of the output sequence, i.e., the direct technical route is adopted, the desired state and output equations are not needed any more. This creates an illusion that the realizability assumption can be removed.

Up to now, the research of output realizability has made some progress. It was shown in [12, 13] that for a class of single-input-single-output (SISO) discrete-time systems, when the input-output coupling parameter (IOCP) is nonzero, any given desired trajectory is realizable and for any given initial state there is a unique desired input corresponding to the given desired trajectory. The article [11] extended the results in [12, 13] to discrete-time system with time-varying IOCP. The article [14] studied the output realizability for a class of multi-input-multioutput (MIMO) discrete-time systems, and built the relation between the output realizability and the input-output coupling matrix (IOCM). It reported that when the IOCM is full-row rank, any given desired trajectory is realizable and when the IOCM is not full-row rank, the output is unrealizable almost everywhere. However, the output realizability criterion for systems with feedforward has not been addressed yet, though it is a problem of great interest.

In ILC, under the premise of the given desired trajectory being realizable, the core problem becomes to design an effective ILC update law for the system input sequence that can drive the control system to track the given desired trajectory. The primary concern is the convergence properties of the iterative learning process: asymptotic convergence, monotone convergence, finite-step convergence, and so on. Obviously, the convergence properties of ILC process are mainly determined by the ILC update law, and the designs of the ILC update laws are inseparable from the information of dynamics structure or dynamics matrices. It can be found that the convergence performance of the designed ILC system will be better and better as the available information becomes more and more [4, 19, 23], in which the control system together with an ILC update law designed is termed as an ILC system. When the IOCM is full-row rank and available, for linear or local-Lipschitz-nonlinear-affine systems, the proportional type (P-type) update law can only ensure the exponential convergence of output sequence and for continuous-nonlinear-affine systems, the P-type update law can only guarantee the asymptotic convergence of output sequence [23]. When the full-row rank IOCM and the state-output coupling matrix (SOCM) are available, for the linear time-invariant (LTI) system, the P-type update law with current state feedback can ensure the 2-step convergence [8]. When the full-row rank IOCMs, the SOCMs, and the output matrices are available, for a class of linear time-varying continuous-time systems, the P-type update law with current state feedback can ensure the 2-step convergence [15]. Clearly, the convergence performance of an ILC system depends on not only the ILC update law designed but also the available information of system

dynamics. This motivates us to further investigate how to leverage the different parts of system dynamics to design an highly efficient update law, and whether the required system dynamics are learnable.

Inspired by the above observations, this paper intends to explore the output realizability, how to exploit the different parts of system dynamics to design highly efficient update law, and whether the dynamics required by the ILC update law designed are finite-step learnable. The main differences between this paper and [14] lie in two aspects. Clearly, the control systems considered in this paper and [14] are different. Note that [14] considered the relation between the output realizability and the IOCM, the learning mechanisms for system dynamics and the monotone convergence of D-type update law with current state feedback. Therefore, the problems considered in this paper and [14] are different. The main contributions of this paper are highlighted as follows.

- The output realizability over the whole time interval is considered. It gives the sufficient and necessary condition for the output to be fully realizable. Meanwhile, it presents an upper bound of the dimension of the realizable output space in the case the feedforward matrix is not full-row rank. Moreover, in the case the feedforward matrix is full column-rank and the input dimension is less than that of output, for a fixed initial state, any given desired trajectory corresponds to at most a desired input.
- It is shown that in the case the output dimension is less than that of the input, there are infinite many desired inputs and states corresponding to the given desired trajectory. And we show how to utilize the direct technique route of convergence analysis and the Cauchy convergence criterion to prove the exponential convergence of system input and state sequences for the classic P-type update law without resorting to the input transformation technique used in [14, 16, 23].
- It establishes the relations between the convergence performances of ILC systems and the output, feedforward, and lower triangle Toeplitz-type block matrices. The P-type update law can only ensure the fully asymptotic learning-ability, in which the feedforward matrix required for designing the gain matrix is $(p+1)$-step learnable. In addition, only using the feedforward matrix, we can get a more efficient point-wise update law that can ensure the fully $(T+2)$-step learning-ability. When the state is measurable and controllable, we can obtain a P-type update law with current state feedback that can ensure the fully 2-step learning-ability and the fully monotone learning-ability, respectively, in which the required output matrix is at least $(np+1)$-step learnable. In the case that the state is unmeasurable or uncontrollable, only using the output data at the previous trial can also get an update law that can, respectively, ensure the fully 2-step learning-ability and the fully monotonic learning-ability, in which the required lower triangle Toeplitz-type block matrix is $(p+1)$-step learnable.

The rest of this paper is organized as follows. In section 2, the output realizability is considered. Section 3 considers the learning-ability of ILC system. Section 4 concludes the paper.

Notation. Throughout this paper, $\mathscr{N} = \{1, 2, \ldots\}$, $\mathscr{N}^+ = \{2, 3, \ldots\}$, and $\mathbb{R}^n$ refers to the n-dimensional real column vector space. For a given vector or matrix x, $x^\top$ denotes its transpose and $\|x\|_\vartheta$ labels its ϑ-norm ($\vartheta = 1, 2, \infty$). Let $\mathscr{S} = \{0, 1, \ldots, T\}$, $\mathscr{S}^+ = \{1, \ldots, T\}$, and $\mathscr{S}^- = \{0, 1, \ldots, T-1\}$, where the positive integer T labels the trial length. For a matrix $D \in \mathbb{R}^{q\times p}$ with columns being $d_1, \ldots, d_p$,

span(D) denotes the linear space $\{\alpha_1 d_1 + \cdots + \alpha_p d_p : [\alpha_1, \ldots, \alpha_p]^\top \in \mathbb{R}^p\}$ and rank(D) refers to the rank of D. For given positive integers m and n, I_n denotes the $n \times n$ identify matrix, 0_n labels the n-dimensional zero column vector and $0_{m\times n}$ denotes the $m \times n$ matrix whose entries are all zero. $\rho(A)$ denotes the spectral radius of a square matrix $A \in \mathbb{R}^{m\times m}$.

2. Output realizability. Consider the following MIMO discrete-time repetitive system:

$$\begin{cases} x_k(t+1) = Ax_k(t) + Bu_k(t), t \in \mathscr{S}^-, \\ y_k(t) = Cx_k(t) + Du_k(t), t \in \mathscr{S}, \\ x_k(0) = x^0, \end{cases} \tag{2.1}$$

where t denotes the discrete time and $k \in \mathscr{N}$ labels the trial number. $u_k(t) \in \mathbb{R}^p$ is the input, $y_k(t) \in \mathbb{R}^q$ is the output, and $x_k(t) \in \mathbb{R}^n$ is the state. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{q\times n}$, and $D \in \mathbb{R}^{q\times p}$ are the unknown matrices. $x_k(0) = x^0$ means that the initial state can be reset to a fixed state x^0.

In ILC, the control objective is to exploit the available information to design an update law for the input sequence $u_k(\cdot)$ such that the output sequence $y_k(\cdot)$ can track the given desired trajectory/output $y_d(\cdot)$ when k tends to infinity, i.e., $\lim_{k\to\infty} y_k(t) = y_d(t)$ for all $t \in \mathscr{S}$. Here, the available information may include the known system dynamics information and the input, output, and state data. The control objective inherently requires $y_d(\cdot)$ to be realizable, i.e., there exists at least a desired input $u_d(\cdot)$ that can drive the control system to yield $y_d(\cdot)$. Note that the system state signals over the finite time interval $\mathscr{S}^+$ are only determined by the system input. Therefore, $y_d(\cdot)$ is realizable means that there at least exists a pair of $u_d(\cdot)$ and $x_d(\cdot)$ satisfying

$$\begin{cases} x_d(t+1) = Ax_d(t) + Bu_d(t), t \in \mathscr{S}^-, \\ y_d(t) = Cx_d(t) + Du_d(t), t \in \mathscr{S}, \\ x_d(0) = x^0. \end{cases} \tag{2.2}$$

In this paper, one of the issues we are interested in is to explore the criterion for the output realizability of system (2.1). The following definitions are useful.

Definition 2.1.

- *A given desired output $y_d(\cdot)$ for system* (2.1) *is said to be realizable if there exists at least a pair of $u_d(\cdot)$ and $x_d(\cdot)$ satisfying* (2.2).
- *The set $\mathscr{Y}_d$ defined by*

$$\left\{[(y_d(0))^\top, \ldots, (y_d(T))^\top]^\top : y_d(\cdot)\text{is realizable for system (2.1)}\right\}$$

  is termed as the realizable output space (ROS).
- *The output is said to be fully realizable if $\mathscr{Y}_d = \mathbb{R}^{q(T+1)}$.*

The following theorem gives the necessary and sufficient conditions for the output to be fully realizable.

Theorem 2.2. *The output of system* (2.1) *is fully realizable if and only if D is full-row rank.*

Proof. First, we prove the sufficiency. On account of rank$(D) = q$, we have $p \geq q$.

Case 1: $p = q$.

This case implies that D is nonsingular. Therefore, for any given $y_d(\cdot)$, there exists a unique pair of $u_d(\cdot)$ and $x_d(\cdot)$, given by the recursion formula (2.3), satisfying (2.2). This means that the output of system (2.1) is fully realizable:

$$\begin{cases} u_d(t) = D^{-1}[y_d(t) - Cx_d(t)], & t \in \mathscr{S}, \\ x_d(t+1) = Ax_d(t) + Bu_d(t), & t \in \mathscr{S}^-, \\ x_d(0) = x^0. \end{cases} \tag{2.3}$$

Case 2: $p > q$.

Let $d_i (1 \le i \le p)$ denote the ith column of D. Due to $\operatorname{rank}(D) = q < p$, there exist $d_{i_1}, \ldots, d_{i_q} (1 \le i_1 < \cdots < i_q \le p)$ such that the matrix $D_1 \triangleq [d_{i_1}, \ldots, d_{i_q}]$ is nonsingular. Let $D_2 \triangleq [b_{j_1}, \ldots, b_{j_{p-q}}]$, where $1 \le j_1 < \cdots < j_{p-q} \le p$,

$$\{i_1, \ldots, i_q\} \bigcap \{j_1, \ldots, j_{p-q}\} = \emptyset,$$
$$\{i_1, \ldots, i_q\} \bigcup \{j_1, \ldots, j_{p-q}\} = \{1, \ldots, p\}.$$

For $t \in \mathscr{S}$, $u_d^i(t)$ denotes the ith component of $u_d(t)$. Let

$$u'_d(t) \triangleq [u_d^{i_1}(t), \ldots, u_d^{i_q}(t)]^\top,$$
$$u''_d(t) \triangleq [u_d^{j_1}(t), \ldots, u_d^{j_{p-q}}(t)]^\top.$$

Then, (2.2) becomes

$$\begin{cases} x_d(t+1) = Ax_d(t) + Bu_d(t), & t \in \mathscr{S}^-, \\ y_d(t) = Cx_d(t) + D_1 u'_d(t) + D_2 u''_d(t), & t \in \mathscr{S}, \\ x_d(0) = x^0, \end{cases} \tag{2.4}$$

which together with the nonsingularity of D_1 yields

$$\begin{cases} u'_d(t) = \left[D_1\right]^{-1}\left[y_d(t) - Cx_d(t) - D_2 u''_d(t)\right], & t \in \mathscr{S}, \\ x_d(t+1) = Ax_d(t) + Bu_d(t), & t \in \mathscr{S}^-, \\ x_d(0) = x^0. \end{cases} \tag{2.5}$$

According to (2.5), for any given $y_d(\cdot)$ and $u''_d(\cdot)$, there exists a unique pair of $u'_d(\cdot)$ and $x_d(\cdot)$ satisfying (2.4). Therefore, the output of system (2.1) is fully realizable.

Next, we prove the necessity by contradiction.

What we need to do is to show that if $\operatorname{rank}(D) < q$, there exists at least a desired trajectory $y_d(t) (t \in \mathscr{S})$ that is not realizable.

Let $r = \operatorname{rank}(D)$ and $\operatorname{span}(D)$ be a vector space spanned by the column vectors of D. Clearly, $\operatorname{span}(D)$ is a r-dimensional true subspace of $\mathbb{R}^q$. Therefore, there exists $y'_d(0)$ that belongs to $\mathbb{R}^q$ and does not belong to $\operatorname{span}(D)$. It is easy to check that if $y_d(t) (t \in \mathscr{S})$ satisfies $y_d(0) = y'_d(0) + Cx_d(0)$, then it is not a realizable desired output for system (2.1). This completes the proof of necessity. ∎

The following theorem gives the upper bound of the dimension of the ROS $\mathscr{Y}_d$ in the case the rank of D is less than q.

THEOREM 2.3. *If the rank of D is less than q, then the dimension of the ROS $\mathscr{Y}_d$ is no more than $qT + r$, where $r = \operatorname{rank}(D)$.*

Proof. **Case 1:** $r = p < q$.

In this case, there exists a nonsingular matrix Q to ensure that $QD = \left[\begin{smallmatrix} I_r \\ 0_{q-r} \end{smallmatrix}\right]$.

Let $y'_d(0) = Qy_d(0)$. We partition $y'_d(0)$ and Q as

$$y'_d(0) = \begin{bmatrix} y'_d(1,0) \\ y'_d(2,0) \end{bmatrix} \quad \text{and} \quad Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}, \tag{2.6}$$

where $y'_d(1,0) \in \mathbb{R}^r$, $y'_d(2,0) \in \mathbb{R}^{q-r}$, $Q_1 \in \mathbb{R}^{r\times q}$, and $Q_2 \in \mathbb{R}^{(q-r)\times q}$. Then, we have

$$\begin{cases} y'_d(1,0) = Q_1 C x_d(0) + u_d(0), \\ y'_d(2,0) = Q_2 C x_d(0). \end{cases} \tag{2.7}$$

Obviously, for any given $y'_d(1,0)$, there exists a unique $u_d(0)$ satisfying (2.7). On account of $y_d(0) = Q^{-1} y'_d(0)$, $y_d(t)(t \in \mathscr{S})$ is not realizable if $y_d(0)$ does not take the form of (2.8). This means that the dimension of the ROS $\mathscr{Y}_d$ for system (2.1) is at most $qT + r$:

$$y_d(0) = Q^{-1} \begin{bmatrix} y'_d(1,0) \\ Q_2 C x_d(0) \end{bmatrix}. \tag{2.8}$$

Case 2: $r < p$.

In this case, there exist the nonsingular matrices Q and P satisfying

$$QDP = \begin{bmatrix} I_r & 0_{r\times(p-r)} \\ 0_{(q-r)\times r} & 0_{(q-r)\times(p-r)} \end{bmatrix}. \tag{2.9}$$

Let $u'_d(0) = P^{-1} u_d(0)$. Due to $r < p$, we partition $u'_d(0)$ as

$$u'_d(0) = \begin{bmatrix} u'_d(1,0) \\ u'_d(2,0) \end{bmatrix}, \tag{2.10}$$

where $u'_d(1,0) \in \mathbb{R}^r$ and $u'_d(2,0) \in \mathbb{R}^{p-r}$.

Using (2.6), (2.9), and (2.10) together with $y_d(0) = Cx_d(0) + Du_d(0)$ yields

$$\begin{cases} y'_d(1,0) = Q_1 C x_d(0) + u'_d(1,0), \\ y'_d(2,0) = Q_2 C x_d(0). \end{cases} \tag{2.11}$$

Then, for any given $y'_d(1,0)$, there is a unique $u'_d(1,0)$ to meet (2.11). Since $y_d(0) = Q^{-1} y'_d(0)$, $y_d(t)(t \in \mathscr{S})$ is not realizable if $y_d(0)$ does not take the form of (2.8). Therefore, the dimension of the ROS $\mathscr{Y}_d$ is less than $qT + r$.

This completes the proof of Theorem 2.3. ∎

The following theorem shows that when D is full-column rank and the output dimension is greater than that of the input for a given desired trajectory there exists at most a pair of desired input and state satisfying (2.2).

THEOREM 2.4. *Assume that* $\operatorname{rank}(D) = p < q$. *For an arbitrarily given desired trajectory* $y_d(\cdot)$, *there exists at most a pair of* $u_d(\cdot)$ *and* $x_d(\cdot)$ *satisfying* (2.2).

Proof. We will complete the proof by contradiction.

Assume that $u_d(\cdot)$ and $x_d(\cdot)$ is a pair of desired input and state satisfying (2.2) and $u'_d(\cdot)$ and $x'_d(\cdot)$ is another pair of desired input and state satisfying (2.12):

$$\begin{cases} x'_d(t+1) = Ax'_d(t) + Bu'_d(t), t \in \mathscr{S}^-, \\ y_d(t) = Cx'_d(t) + Du'_d(t), t \in \mathscr{S}, \\ x'_d(0) = x^0. \end{cases} \tag{2.12}$$

Since the state signals over $\mathscr{S}^+$ are only determined by the input signals over $\mathscr{S}^-$ and the initial state signal x^0, what we need to do is to show

$$u_d(t) = u'_d(t) \quad \text{for all}\, t \in \mathscr{S}. \tag{2.13}$$

It follows from (2.2) and (2.12) that

$$x_d(t+1) - x'_d(t+1) = A[x_d(t) - x'_d(t)] + B[u_d(t) - u'_d(t)], t \in \mathscr{S}^-, \tag{2.14}$$

$$0_q = C[x_d(t) - x'_d(t)] + D[u_d(t) - u'_d(t)], t \in \mathscr{S}, \tag{2.15}$$

$$x_d(0) - x'_d(0) = 0_n. \tag{2.16}$$

Using (2.15) and (2.16) leads to

$$D[u_d(0) - u'_d(0)] = 0_q. \tag{2.17}$$

By (2.17), together with the condition $q > p = \text{ rank}(D)$, we get

$$u_d(0) = u'_d(0).$$

What the rest we need to do is to show

$$u_d(s+1) = u'_d(s+1) \tag{2.18}$$

provided that

$$u_d(t) = u'_d(t) \quad \text{for all}\, t \in \{0, 1, \ldots, s\}, \tag{2.19}$$

where $0 \le s \le T-1$.

Using the induction hypothesis (2.19) together with the condition $x_d(0) = x'_d(0) = x^0$ gives

$$x_d(s+1) = x'_d(s+1). \tag{2.20}$$

By (2.15) and (2.20), we get

$$D[u_d(s+1) - u'_d(s+1)] = 0_q. \tag{2.21}$$

Using $\text{rank}(D) = p < q$ and (2.21) yields (2.18). This completes the proof. ∎

Remark 2.5. According to Theorem 2.3, it makes no sense to assume that a given desired output is realizable when system (2.1) contains unknown dynamics information. However, this does not deny the theoretical feasibility of the indirect technology route of the convergence analysis. According to Theorem 2.4, any realizable desired trajectory only has a pair of desired input and state satisfying (2.2) if D is full-column rank. This means we can use the desired state and output equations (2.2) to get the convergence of input and output sequences.

Remark 2.6. It follows from Theorems 2.2 and 2.3 that the output realizability does not depend on the trial length, the ILC update law designed, and the repetitiveness of the control system, which is an inherent property of the control system.

3. Learning-ability of ILC system. In the previous section, we have discussed the output realizability. It is observed that the output realizability only depends on the dynamics of control system including its initial state. The output realizability only reflects whether the control system has the potential ability to achieve the control objective. Clearly, that the output is realizable is only a prerequisite for the control objective to be feasible.

In this section, what we care about is the learning-ability of ILC system. We should first clarify what the learning-ability of ILC system is. System (2.1) together

with an ILC update law designed is termed as an ILC system. Since the dynamics matrices of system (2.1) are completely unknown, the connotation of the learning-ability of ILC system should include three aspects: the given desired trajectory to be realizable, the output sequence convergent to the given desired trajectory and the required system dynamics information to be learnable.

The following definitions are useful.

Definition 3.1.

- $\Upsilon \in \{A, B, C, D, H\}$ *is said to be* m*-step learnable if there is a learning scheme by which we can obtain* Υ *through running system* (2.1) m *times, where* m *is a positive integer and* H *is the lower triangle Toeplitz-type block matrix defined by*

$$H = \begin{bmatrix} D & & & & \\ CB & D & & & \\ CAB & CB & D & & \\ \vdots & \ddots & \ddots & \ddots & \\ CA^{T-1}B & \cdots & CAB & CB & D \end{bmatrix}. \tag{3.1}$$

- *System* (2.1) *with an ILC update law is said to have fully asymptotic learning-ability if* (1) *any given desired trajectory is realizable;* (2) *for any initial input, the output sequence can asymptotically converge to any given desired trajectory; and* (3) *the required system dynamics information is* m*-step learnable.*
- *System* (2.1) *with an ILC update law is said to have fully monotonic learning-ability in the sense of* ϑ*-norm if* (1) *any given desired trajectory is realizable;* (2) *for any initial input, the output sequence can monotonically converge to any given desired trajectory in the sense of* ϑ*-norm; and* (3) *the required system dynamics information is* m*-step learnable, where* $\vartheta = 1, 2, \infty$.
- *System* (2.1) *with an ILC update law is said to have fully* M*-step learning-ability if* (1) *any given desired trajectory is realizable;* (2) *for any initial input, the output at the* M*th trial can be any given desired trajectory;* and (3) *the required system dynamics information is* m*-step learnable, where* M *is a positive integer.*
- *The state of system* (2.1) *is said to be measurable if it can be obtained directly by a sensor or other means of measurement.*

3.1. P-type update law. We first show that if D is full-row rank, system (2.1) with P-type update law (3.2) can ensure the fully asymptotic learning-ability:

$$u_{k+1}(t) = u_k(t) + \Gamma \delta y_k(t), \tag{3.2}$$

where $\Gamma \in \mathbb{R}^{p \times q}$ is the learning gain matrix and $\delta y_k(t) = y_d(t) - y_k(t)$.

It is well known that under the condition $\operatorname{rank}(D) = q < p$, the output sequence of system (2.1) with update law (3.2) is exponentially convergent to the desired trajectory if and only if $\rho(I_q - D\Gamma) < 1$. It follows from the proof of Theorem 2.2 that for any given desired trajectory, there are infinitely many desired inputs and states satisfying (2.2). Therefore, we cannot directly use (2.2) to prove the convergence of system input and state sequences. However, by Cauchy convergence criterion together with the direct technique route of convergence analysis, we can get the exponential convergence of the input and state sequences, respectively. The above fact can be summed up in Theorem 3.2 whose proof can be found in the appendix.

THEOREM 3.2. *Assume that the update law* (3.2) *is applied to system* (2.1) *and* D *is full-row rank. Then, for any given* $y_d(\cdot)$ *and* $u_1(\cdot)$, $y_k(\cdot)$ *exponentially converges to* $y_d(\cdot)$ *when* k *tends to infinity, and* $u_k(\cdot)$ *and* $x_k(\cdot)$ *are exponentially convergent when* k *tends to infinity whose limits together with* $y_d(\cdot)$ *satisfy* (2.2) *if and only if* $\rho(I_q - D\Gamma) < 1$.

Let

$$Y_k = [(y_k(0))^\top, \ldots, (y_k(T))^\top]^\top, Y_d = [(y_d(0))^\top, \ldots, (y_d(T))^\top]^\top, \mathbb{G} = \mathrm{diag}\{\underbrace{\Gamma, \ldots, \Gamma}_{T+1}\}. \tag{3.3}$$

Then, the output sequence Y_k of system (2.1) with update law (3.2) satisfies

$$\delta Y_{k+1} = [I_{q(T+1)} - H\mathbb{G}]\delta Y_k, \tag{3.4}$$

where $\delta Y_k = Y_d - Y_k$ and H is given by (3.1).

It is easy to verify that for any initial input, the output sequence monotonically converges to the desired trajectory in the sense of ϑ-norm, i.e., $\lim_{k\to\infty} Y_k = Y_d$ and $\|\delta Y_{k+1}\|_\vartheta \le \|\delta Y_k\|_\vartheta$ if and only if $\rho(I_q - D\Gamma) < 1$ and $\|I_{q(T+1)} - H\mathbb{G}\|_\vartheta \le 1$, where $\vartheta = 1, 2, \infty$. However, by only tuning the learning gain Γ, the monotone convergence condition cannot be ensured.

Theorem 3.2 tells that for system (2.1) with update law (3.2), D is the only system dynamics information involved in designing the learning gain matrix Γ. Now, we show how to use the column-by-column learning mechanism given in [11] to get the feedforward matrix D through running the repetitive system (2.1) $p+1$ times.

Let d_j be the jth column of D, where $j = 1, \ldots, p$. $u_k^i(t)$ denotes the ith component of the column vector $u_k(t)$, where $1 \le i \le p$ and $t \in \mathscr{S}$.

Step 1. Arbitrarily choose an admissible initial system input $u_1(t)(t \in \mathscr{S})$. Supplying system (2.1) with it yields $y_1(t)(t \in \mathscr{S})$.

Step 2. Let $u_{j+1}(t)(t \in \mathscr{S})$ satisfy $u_{j+1}^j(0) = u_1^j(0) + \theta$, $u_{j+1}^i(0) = u_1^i(0)$ for $i \ne j$ and $u_{j+1}(t) = u_1(t)$ for $t \ne 0$, where θ is a nonzero constant. Using $u_{j+1}(t)(t \in \mathscr{S})$ to drive system (2.1) gives $y_{j+1}(t)(t \in \mathscr{S})$.

By steps 1 and 2 together with $x_{j+1}(0) = x_1(0) = x^0$, we have

$$y_{j+1}(0) - y_1(0) = \theta d_j. \tag{3.5}$$

Thanks to $\theta \ne 0$, using (3.5) gives $d_j = \frac{y_{j+1}(0) - y_1(0)}{\theta}$.

Following the facts above yields Theorem 3.3.

THEOREM 3.3. *The feedforward matrix* D *is* $(p+1)$*-step learnable.*

According to Theorems 2.2, 3.2, and 3.3 and the facts stated above, there holds Theorem 3.4.

THEOREM 3.4. *Assume that* D *is full-row rank. System* (2.1) *with update law* (3.2) *can ensure the fully asymptotic learning-ability if and only if* $\rho(I_q - D\Gamma) < 1$.

3.2. Pointwise update law. The following theorem shows that when D is available, we can get a better pointwise update law (3.6) such that the output of system (2.1) with update law (3.6) at the $(T+2)$th trial is the desired trajectory:

$$u_{k+1}(t) = \begin{cases} u_k(k-1) + \Gamma\delta y_k(k-1) & \text{for } t = k-1, \\ u_k(t) & \text{for } t \ne k-1, \end{cases} \tag{3.6}$$

where the initial input $u_1(t)(t \in \mathscr{S})$ is arbitrarily given and $1 \le k \le T+1$.

THEOREM 3.5. *Assume that D is full-row rank. System* (2.1) *with the pointwise update law* (3.6) *can ensure the fully $(T+2)$-step learning-ability if $D\Gamma = I_q$.*

Proof. Since the full-row rank matrix D is $(p+1)$-step learnable, what we need to do is to show that when $D\Gamma = I_q$, for any given $y_d(\cdot)$ and $u_1(\cdot)$, the output at the $(T+2)$th trial is the given desired trajectory, i.e., $y_{T+2}(t) = y_d(t)(t \in \mathscr{S})$.

At the 1st trial, supplying system (2.1) with $u_1(t)(t \in \mathscr{S})$ yields $y_1(t)(t \in \mathscr{S})$.

Since $u_2(0) = u_1(0) + \Gamma\delta y_k(0)$ and $u_2(t) = u_1(t)$ for $t \neq 0$, it follows from $D\Gamma = I_q$ that $y_2(0) = y_d(0)$.

Next, we show that when $u_{k+2}(t)(t \in \mathscr{S})$ is applied to system (2.1), the output $y_{k+2}(t)(t \in \mathscr{S})$ satisfies $y_{k+2}(t) = y_d(t)$ for $0 \le t \le k$ provided that when we apply $u_{k+1}(t)(t \in \mathscr{S})$ to system (2.1), the output $y_{k+1}(t)(t \in \mathscr{S})$ satisfies $y_{k+1}(t) = y_d(t)$ for $0 \le t \le k-1 (k \ge 1)$.

On account of $y_{k+1}(t) = y_d(t)$ for $0 \le t \le k-1$, using (3.6) yields

$$u_{k+2}(t) = \begin{cases} u_{k+1}(k) + \Gamma\delta y_{k+1}(k) & \text{for} t = k, \\ u_{k+1}(t) & \text{for} t \neq k. \end{cases} \tag{3.7}$$

Using (2.1) and (3.7) together with $D\Gamma = I_q$ gives rise to $y_{k+2}(t) = y_d(t)$ for $0 \le t \le k$. It follows from mathematical induction that $y_{T+2}(t) = y_d(t)(t \in \mathscr{S})$. ∎

Remark 3.6. It follows from Theorems 3.4 and 3.5 that if the full-row rank feedforward matrix D is available, from the point of view of efficiency, the learning-ability of system (2.1) with pointwise update law (3.6) outperforms that of system (2.1) with update law (3.2). Clearly, neither of the above two ILC systems can guarantee the fully monotone learning-ability.

3.3. P-type update law with current state feedback. When the system state is controllable and measurable, incorporating the state feedback mechanism into update law (3.2) gives rise to a more efficient update law (3.8). Applying it to system (2.1) achieves Theorem 3.7:

$$u_{k+1}(t) = u_k(t) + K[x_{k+1}(t) - x_k(t)] + \Gamma\delta y_k(t), \tag{3.8}$$

where K is the state gain matrix and Γ is the learning gain matrix.

THEOREM 3.7. *Assume that D is full-row rank and the state of system* (2.1) *is controllable, i.e., the controllability matrix $Q_c = [A^{n-1}B, A^{n-2}B, \ldots, AB, B]$ is full-row rank. Then, system* (2.1) *with update law* (3.8) *can ensure the fully* 2*-step learning-ability if $D\Gamma = I_q$ and $C + DK = 0_{q\times n}$. Meanwhile, system* (2.1) *with update law* (3.8) *can ensure the fully monotone learning-ability in the sense of ϑ-norm if and only if $\rho(I_q - D\Gamma) < 1$ and $\|I_{q(T+1)} - \bar{H}\mathbb{G}\|_\vartheta \le 1$, where $\vartheta = 1, 2, \infty$,*

$$\bar{H} = \begin{bmatrix} D & & & & \\ \bar{C}B & D & & & \\ \bar{C}\bar{A}B & \bar{C}B & D & & \\ \vdots & \ddots & \ddots & \ddots & \\ \bar{C}\bar{A}^{T-1}B & \cdots & \bar{C}\bar{A}B & \bar{C}B & D \end{bmatrix},$$

$\bar{C} = C + DK$, and $\bar{A} = A + BK$.

Proof. Since D is full-row rank, it follows from Theorem 2.2 that any given desired trajectory for system (2.1) is realizable.

The rest of what we need to do is show that (i) when $D\Gamma = I_q$ and $C+DK = 0_{q\times n}$ for any given $y_d(\cdot)$ and $u_1(\cdot)$, the output at the 2nd trial is the desired trajectory, i.e., $y_2(t) = y_d(t)(\forall t)$; (ii) for any given $y_d(\cdot)$ and $u_1(\cdot)$, the output sequence $y_k(t)$ is monotonically convergent to $y_d(t)$ in the sense of ϑ-norm if and only if $\rho(I_q - D\Gamma) < 1$ and $\|I_{q(T+1)} - \bar{H}\mathbb{G}\|_\vartheta \le 1$; and (iii) under the conditions that the state is measurable and Q_c is full-row rank, A, B, and C are M-step learnable, where M is some positive integer.

We first prove (i).

Since D is full-row rank, there exists the gain matrices Γ and K such that $D\Gamma = I_q$ and $C + DK = 0_{q\times n}$. Using (2.1) and (3.8) yields

$$\delta y_2(t) = [I_q - D\Gamma]\delta y_1(t) - [C + DK][x_2(t) - x_1(t)] \tag{3.9}$$

for $t \in \mathscr{S}$. It follows from (3.9) together with $D\Gamma = I_q$ and $C + DK = 0_{q\times n}$ that $y_2(t) = y_d(t)$ for $t \in \mathscr{S}$.

Next, we prove the sufficiency of (ii).

Using (2.1) and (3.8) yields

$$\delta Y_{k+1} = \Xi \delta Y_k, \tag{3.10}$$

where $\Xi = I_{q(T+1)} - \bar{H}\mathbb{G}$.

Since $\rho(I_q - D\Gamma) < 1$ and Ξ is a lower triangle Toeplitz-type block matrix with $I_q - D\Gamma$ being its main diagonal block entries, it follows that $\rho(\Xi) < 1$ which implies that $\lim_{k\to\infty} Y_k = Y_d$. Using (3.10) together with the condition $\|\Xi\|_\vartheta \le 1$ leads to $\|\delta Y_{k+1}\|_\vartheta \le \|\delta Y_k\|_\vartheta$, where $\vartheta = 1, 2, \infty$. This completes the proof of the sufficiency of (ii).

Now, we move to prove the necessity of (ii). What we need to do is to show that when $\rho(I_q - D\Gamma) \ge 1$ or $\|\Xi\|_\vartheta > 1$, there exist a desired trajectory Y_d and an initial input $u_1(t)$ such that the output sequence Y_k does not monotonically converge to Y_d in the sense of ϑ-norm, where $\vartheta = 1, 2, \infty$.

In the case $\rho(I_q - D\Gamma) \ge 1$, it follows that $\rho(\Xi) \ge 1$. Since D is full-row rank, using (3.10) and $\rho(\Xi) \ge 1$ together with the same arguments used in the proof of Theorem 3.2 gives that there does exist a desired trajectory Y_d and an initial input $u_1(t)$ such that the output sequence Y_k does not converge to Y_d. Therefore, $\rho(I_q - D\Gamma) \ge 1$ implies that there are Y_d and $u_1(\cdot)$ such that the output sequence Y_k does not monotonically converge to Y_d in the sense of $1/2/\infty$-norm.

In the case $\rho(I_q - D\Gamma) < 1$ and $\|\Xi\|_\vartheta > 1$, there exists a column vector $\xi_\vartheta \in \mathbb{R}^{q(T+1)}$ such that $\|\xi_\vartheta\|_\vartheta = 1$ and $\|\Xi\xi_\vartheta\|_\vartheta = \|\Xi\|_\vartheta$, where $\vartheta = 1, 2, \infty$. Since D is full-row rank, for a given initial input $u_1(\cdot)$ there exists a desired trajectory Y_d such that $\delta Y_1 = \xi_\vartheta$. Therefore, $\|\delta Y_2\|_\vartheta > \|\delta Y_1\|_\vartheta$. The above fact implies that the conditions $\rho(I_q - D\Gamma) < 1$ and $\|\Xi\|_\vartheta > 1$ cannot ensure the output sequence to be monotonically convergent in the sense of ϑ-norm. This completes the proof of the necessity of (ii).

Finally, we prove (iii).

Under the condition that the state is measurable, by the column-by-column learning scheme, we can obtain the input matrix B through running the repetitive system (2.1) $p+1$ times. When Q_c is full-row rank, by the learning scheme developed in [14] we can conclude that A and C are at least $(np+1)$-step learnable. This completes the proof of (iii). ∎

Remark 3.8. A direct computation yields

$$\|\Xi\|_1 \le \|I_q - D\Gamma\|_1 + \|C + DK\|_1 \left(\sum_{i=0}^{T-1} \|A + BK\|_1^i\right) \|B\Gamma\|_1,$$

$$\|\Xi\|_\infty \le \|I_q - D\Gamma\|_\infty + \|C + DK\|_\infty \left(\sum_{i=0}^{T-1} \|A + BK\|_\infty^i\right) \|B\Gamma\|_\infty,$$
$$\|\Xi\|_2 \le \sqrt{\|\Xi\|_1 \|\Xi\|_\infty},$$

where $\|A + BK\|_1^0 = 1$ and $\|A + BK\|_\infty^0 = 1$. Clearly, there are two options for designing the gain matrix K. One is to utilize the available information of A and B. However, it cannot ensure the monotone convergence even if A and B are completely available. Another is to leverage the available information of C and D to design K. Under the condition that D is full-row rank, this option can ensure the monotone convergence as long as the available information of D and C is accurate enough.

3.4. Full output feedback update law. When the state is unmeasurable, the update law (3.8) is unfeasible. Next, we will show that the ILC update law (3.11) using only the output signals of the previous trial, which is termed as full output feedback update law, can also guarantee the fully 2-step learning-ability and the fully monotone learning-ability in the sense of $1/2/\infty$-norm, respectively.

$$u_{k+1}(t) = u_k(t) + \sum_{s=0}^{t} \Gamma_{t-s} \delta y_k(s), \tag{3.11}$$

where $\Gamma_t \in \mathbb{R}^{p\times q} (0 \le t \le T)$ are the learning gain matrices.

THEOREM 3.9. *Assume that D is full-row rank. Then,* (a) *there exist the gain matrices $\Gamma_t (0 \le t \le T)$ such that system* (2.1) *with update law* (3.11) *can ensure the fully 2-step learning-ability;* (b) *system* (2.1) *with update law* (3.11) *can ensure the fully monotone learning-ability in the sense of ϑ-norm if and only if $\rho(I_q - D\Gamma) < 1$ and $\|I_{q(T+1)} - H\mathscr{G}\|_\vartheta \le 1$, where $\vartheta = 1, 2, \infty$ and*

$$\mathscr{G} = \begin{bmatrix} \Gamma_0 & & & & \\ \Gamma_1 & \Gamma_0 & & & \\ \Gamma_2 & \Gamma_1 & \Gamma_0 & & \\ \vdots & \ddots & \ddots & \ddots & \\ \Gamma_T & \cdots & \Gamma_2 & \Gamma_1 & \Gamma_0 \end{bmatrix}.$$

Proof. We first prove (a).

Since D is full-row rank, any given desired trajectory for system (2.1) is realizable.

Next, we illustrate that for any given desired trajectory and initial input, there exist the gain matrices $\Gamma_t (0 \le t \le T)$ such that the output at the 2nd trial is the given desired trajectory. Using (2.1) and (3.11) gives rise to

$$\delta Y_2 = [I_{q(T+1)} - H\mathscr{G}]\delta Y_1. \tag{3.12}$$

where H is given by (3.1). Let $\psi_0 = D\Gamma_0$ and $\varphi_i = \sum_{s=0}^{i-1} CA^{i-1-s}B\Gamma_s + D\Gamma_i$ for $i = 1, 2, \ldots, T$. Then, a direct computation yields

$$H\mathscr{G} = \begin{bmatrix} \psi_0 & & & & \\ \psi_1 & \psi_0 & & & \\ \psi_2 & \psi_1 & \psi_0 & & \\ \vdots & \ddots & \ddots & \ddots & \\ \psi_T & \cdots & \psi_2 & \psi_1 & \psi_0 \end{bmatrix}. \tag{3.13}$$

Therefore, what we need to do is to show that there exist the matrices $\Gamma_t (0 \le t \le T)$ solving $D\Gamma_0 = I_q$ and

$$\sum_{s=0}^{i-1} CA^{i-1-s} B\Gamma_s + D\Gamma_i = 0_{q\times q} \quad \text{for} i = 1, 2, \ldots, T. \tag{3.14}$$

On account of $\text{rank}(D) = q$, there is Γ_0 solving $D\Gamma_0 = I_q$. For such a Γ_0 there exists Γ_1 solving the matrix equation $CB\Gamma_0 + D\Gamma_1 = 0_{q\times q}$. Suppose that for $1 \le i_0 < T$ there exist $\Gamma_0, \Gamma_1, \ldots, \Gamma_{i_0}$ solving the matrix equation $\sum_{s=0}^{i_0-1} CA^{i_0-1-s} B\Gamma_s + D\Gamma_{i_0} = 0_{q\times q}$. For such a group of gain matrices $\Gamma_0, \Gamma_1, \ldots, \Gamma_{i_0}$, using the condition D is full-row rank yields that there exists Γ_{i_0+1} solving the matrix equation $\sum_{s=0}^{i_0} CA^{i_0-s} B\Gamma_s + D\Gamma_{i_0+1} = 0_{q\times q}$. It follows from mathematical induction that there exist the gain matrices $\Gamma_0, \Gamma_1, \ldots, \Gamma_T$ solving $D\Gamma_0 = I_q$ and (3.14).

Clearly, the design of the gain matrices $\Gamma_0, \Gamma_1, \ldots, \Gamma_T$ requires the information of the matrix H. Since the first p columns of H contain all of its information, what we need to do is to design a learning scheme for the first p columns of H.

Let $U_k = [(u_k(0))^\top, \ldots, (u_k(T))^\top]^\top$ and $\Phi = \begin{bmatrix} C^\top & (CA)^\top & \cdots & (CA^T)^\top \end{bmatrix}^\top$. Then, we can get

$$Y_k = HU_k + \Phi x^0, \tag{3.15}$$

where Y_k is defined by (3.3).

Let h_i denote the ith column of H, where $1 \le i \le p$. Arbitrarily choose an admissible initial system input $u_1(\cdot)$. Supplying system (2.1) with $u_1(\cdot)$ yields Y_1. Let the input $u_{i+1}(\cdot)$ satisfy $u_{i+1}^i(0) = u_1^i(0) + \theta$, $u_{i+1}^l(0) = u_1^l(0)$ for $l \ne i$ and $u_{i+1}(t) = u_1(t)$ for $t \ne 0$, where θ is a nonzero constant. Using $u_{i+1}(\cdot)$ to drive system (2.1) gives Y_{i+1}. By (3.15) together with a simple computation, we obtain

$$h_i = \frac{Y_{i+1} - Y_1}{\theta} (1 \le i \le p).$$

The above fact implies that we can get the first p columns of H through running the repetitive system (2.1) $p+1$ times. Therefore, the lower triangle Toeplitz-type block matrix H is $(p+1)$-step learnable. This completes the proof of (a).

Now, we move to prove (b).

In the proof of (a), we have shown that under the condition D is full-row rank, any given desired trajectory is realizable and the matrix H is $(p+1)$-step learnable. Therefore, the rest of what we need to do is show that for any given desired trajectory Y_d and initial input $u_1(\cdot)$, the output sequence Y_k of system (2.1) with update law (3.11) monotonically converges to Y_d in the sense of ϑ-norm if and only if $\rho(I_q - D\Gamma) < 1$ and $\|I_{q(T+1)} - H\mathscr{G}\|_\vartheta$, where $\vartheta = 1, 2, \infty$.

Using (2.1) and (3.11) yields

$$\delta Y_{k+1} = [I_{q(T+1)} - H\mathscr{G}]\delta Y_k. \tag{3.16}$$

Then, by (3.16) together with the same arguments as those used in the proof of Theorem 3.7, we can prove that Y_k monotonically converges to Y_d in the sense of ϑ-norm if and only if $\rho(I_q - D\Gamma_0) < 1$ and $\|I_{q(T+1)} - H\mathscr{G}\|_\vartheta \le 1$, where $\vartheta = 1, 2, \infty$. To save space, we omit it. This completes the proof of (b). ∎

Remark 3.10. The column-by-column learning scheme for D or H only requires the information of system input and output and does not require any additional information. Therefore, D and H are unconditionally learnable.

4. Conclusions. This paper first studies the output realizability over the whole operation time interval. In the case the feedforward matrix is full-row rank, the output is fully realizable, and the initial state has no influence on the output realizability. In the case the feedforward is not full-row rank, the dimension of the ROS is no more than $qT + \text{rank}(D)$, and the initial state may have influence on the realizability of the given desired trajectory. When the feedforward matrix is full-column rank and the output dimension is greater than the input dimension, for any given desired trajectory there exists at most a pair of desired input and state. Then, the learning-ability of ILC system is considered, in which we consider the fully asymptotic learning-ability, the fully monotone learning-ability, and the fully finite-step learning-ability. We show that the classic P-type ILC update law can only ensure the fully asymptotic learning-ability, in which the full-row rank feedforward matrix is $(p+1)$-step learnable. Meanwhile, we show how to utilize the Cauchy convergence criterion and the direct technique route of convergence analysis to prove the exponential convergence of system input and state sequences for the classic P-type update law without resorting to the input transformation technique. Furthermore, a pointwise P-type update law is given to show that we can only use the information of the feedforward matrix to design an update law for ensuring the fully $(T+2)$-step learning-ability. It should be pointed that the above two ILC update laws cannot ensure the monotone learning-ability. If the state is measurable and controllable, the P-type ILC update law with current state feedback can ensure the fully 2-step learning-ability and the fully monotone learning-ability, respectively, in which the required output matrix for designing the state gain matrix is $(np+1)$-step learnable. If the state is unmeasurable or uncontrollable, an ILC update law only using the output data at the previous trial can also ensure the fully 2-step learning-ability and the fully monotone learning-ability, respectively, in which the required Lower Toeplitz-type block matrix is $(p+1)$-step learnable. In brief, more available information of system dynamics may mean better learning-ability.

Appendix. The following is the proof of Theorem 3.2.

Proof. Using (2.1) gives rise to

$$\begin{cases} y_k(0) = Cx^0 + Du_k(0), \\ y_k(t) = CA^t x^0 + \sum_{s=0}^{t-1} CA^{t-s-1} Bu_k(s) + Du_k(t) \quad \text{for} t \in \mathscr{S}^+, \end{cases} \tag{4.1}$$

where A^0 refers to the identify matrix I_n. By (4.1) and (3.1), for all $k \in \mathscr{N}$ we have

$$\delta y_{k+1}(0) = [I_q - D\Gamma]\delta y_k(0), \tag{4.2}$$

$$\delta y_{k+1}(t) = [I_q - D\Gamma]\delta y_k(t) - \sum_{s=0}^{t-1} CA^{t-s-1} B\Gamma \delta y_k(s) \quad \text{for} t \in \mathscr{S}^+. \tag{4.3}$$

We first prove the sufficiency.

Under the condition $\rho(I_q - D\Gamma) < 1$, by direct technique route of convergence analysis we can get the exponential convergence of the output sequence $y_k(\cdot)$, i.e., there exist two positive constant numbers Φ and $\rho \in (0,1)$ such that

$$\|\delta y_k(t)\|_1 \le \Phi \rho^k \quad \text{for all} t \in \mathscr{S} \text{and} k \in \mathscr{N}. \tag{4.4}$$

In addition, it follows from (3.1) and (4.4) that for all $k \ge 1$, $m \ge 1$, and $t \in \mathscr{S}$,

$$\|u_{k+m}(t) - u_k(t)\|_1 \le \frac{\Phi \|\Gamma\|_1}{1-\rho} \rho^k, \tag{4.5}$$

which together with Cauchy convergence criterion yields that the system input sequence $u_k(t)$ is convergent when k tends to infinity for all $t \in \mathscr{S}$. Furthermore, taking

$$x_k(t) = A^t x^0 + \sum_{s=0}^{t-1} A^{t-s-1} B u_k(s) \quad \text{for all} t \in \mathscr{S}^+ \tag{4.6}$$

into consideration yields that $x_k(t)$ converges when k tends to infinity over $\mathscr{S}$. Let

$$u_d(t) \triangleq \lim_{k\to\infty} u_k(t)(t \in \mathscr{S}) \quad \text{and} \quad x_d(t) \triangleq \lim_{k\to\infty} x_k(t)(t \in \mathscr{S}).$$

It is easy to check that $u_d(\cdot)$ and $x_d(\cdot)$ together with the corresponding desired trajectory $y_d(\cdot)$ satisfy (2.2).

Next, we illustrate that $u_k(t)$ and $x_k(t)$ exponentially converge to $u_d(t)$ and $x_d(t)$, respectively. Since (4.5) holds for all $m \geq 1$, it follows that

$$\|u_d(t) - u_k(t)\|_1 \leq \frac{\Phi\|\Gamma\|_1}{1-\rho}\rho^k \quad \text{for all} k. \tag{4.7}$$

In addition, by (4.6) and (4.7) together with $x_d(t) = A^t x^0 + \sum_{s=0}^{t-1} A^{t-s-1} B u_d(s)$ for $t \in \mathscr{S}^+$, we get

$$\|\delta x_k(t)\|_1 \leq \left(\sum_{s=0}^{T-1} \|A\|_1^{T-s-1}\right) \frac{\Phi\|B\|_1\|\Gamma\|_1}{1-\rho}\rho^k \quad \text{for all} t \in \mathscr{S}^+, \tag{4.8}$$

where $\delta x_k(t) = x_d(t) - x_k(t)$ and $\delta u_k(t) = u_d(t) - u_k(t)$. Note that $x_d(0) = x_k(0) = x^0$ for all k. Therefore, we have

$$\|\delta x_k(t)\|_1 \leq \left(\sum_{s=0}^{T-1} \|A\|_1^{T-s-1}\right) \frac{\Phi\|B\|_1\|\Gamma\|_1}{1-\rho}\rho^k \quad \text{for all} t \in \mathscr{S}. \tag{4.9}$$

By (4.7) and (4.9), we get that $x_k(\cdot)$ and $u_k(\cdot)$ are exponentially convergent to $x_d(\cdot)$ and $u_d(\cdot)$, respectively. This completes the proof of sufficiency.

Next, we move to prove necessity.

What we need to do is to prove that if $\rho(I_q - D\Gamma) \geq 1$, then there at least exists a desired trajectory $y_d(\cdot)$ and an initial input signal $u_1(0)$ such that $y_k(0)$ does not converge to $y_d(0)$. According to Jordan canonical form theorem, there is a nonsingular matrix Ψ such that

$$\Psi(I_q - D\Gamma)\Psi^{-1} = \begin{bmatrix} J_{m_1}(\lambda_1) & & & \\ & J_{m_2}(\lambda_2) & & \\ & & \ddots & \\ & & & J_{m_\nu}(\lambda_\nu) \end{bmatrix},$$

where $J_{m_i}(\lambda_i)$ is an m_i-by-m_i Jordan block taking the form of

$$\begin{bmatrix} \lambda_i & & & \\ 1 & \lambda_i & & \\ & \ddots & \ddots & \\ & & 1 & \lambda_i \end{bmatrix},$$

where $\lambda_1, \ldots, \lambda_\nu$ are the eigenvalues of the matrix $I_q - D\Gamma$ and $m_1 + \cdots + m_\nu = q$. Due to $\rho(I_q - D\Gamma) = \max_{1\leq i\leq \nu} |\lambda_i|$ and $\rho(I_q - D\Gamma) \geq 1$, there exists $i_0 \in \{1, \ldots, \nu\}$ such

that $|\lambda_{i_0}| \geq 1$. Without loss of generality, assume that $i_0 = 1$, which implies $|\lambda_1| \geq 1$. Let φ_1 be the first row of Ψ. Then, using (4.2) leads to

$$\varphi_1 \delta y_{k+1}(0) = \lambda_1 \varphi_1 \delta y_k(0),$$

which together with $|\lambda_1| \geq 1$ implies

$$|\varphi_1 \delta y_k(0)| \geq |\varphi_1 \delta y_1(0)| \quad \text{for all} k \in \mathscr{N}. \tag{4.10}$$

Let $u_1(0) = 0_p$. Then, $y_1(0) = Cx^0$. On account of the nonsingularity of Ψ, φ_1 is a nonzero row vector. Let φ_1^i be the ith entry of the row vector φ_1. Therefore, there exists $i_1 \in \{1, \ldots, q\}$ such that $\varphi_1^{i_1} \neq 0$. Let $y_1^i(0)$ and $y_d^i(0)$ be the ith entries of $y_1(0)$ and $y_d(0)$, respectively. Choose $y_d(0)$ satisfying $y_d^i(0) = y_1^i(0)$ for $i \neq i_1$ and $y_d^{i_1}(0) = y_1^{i_1}(0) + 1$. Using (4.10) gives rise to

$$|\varphi_1 \delta y_k(0)| \geq |\varphi_1^{i_1}| > 0 \quad \text{for all} k \in \mathscr{N},$$

which implies that $y_k(0)$ do not converge to $y_d(0)$. This proves the necessity. ∎

REFERENCES

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, *Bettering operation of robots by learning*, J. Robotic Syst., 1 (1984), pp. 123–140.

[2] D. A. Bristow, M. Tharayil, and A. G. Alleyne, *A survey of iterative learning control*, IEEE Control Syst. Mag., 26 (2006), pp. 96–114.

[3] X. Bu, F. Yu, Z. Hou, and F. Wang, *Iterative learning control for a class of nonlinear systems with random packet losses*, Nonlinear Anal. Real World Appl., 14 (2013), pp. 567–580.

[4] H. Chen, *Almost sure convergence of iterative learning control for stochastic systems*, Sci. China, 46 (2003), pp. 67–79.

[5] Y. Chen, C. Wen, Z. Gong, and M. Sun, *An iterative learning controller with initial state learning*, IEEE Trans. Automat. Control, 44 (1999), pp. 371–376.

[6] D. Huang, V. Venkataramanan, J. Xu, and T. Huynh, *Contact-induced vibration in dual-stage hard disk drive servo systems and its compensator design*, IEEE Trans. Ind. Electron., 61 (2014), pp. 4052–4060.

[7] H. Ji, Z. Hou, and R. Zhang, *Adaptive iterative learning control for high-speed trains with unknown speed delays and input saturations*, IEEE Trans. Automat. Sci. Eng., 13 (2016), pp. 260–273.

[8] J. E. Kurek and M. B. Zaremba, *Iterative learning control synthesis based on 2-d system-theory*, IEEE Trans. Automat. Control, 38 (1993), pp. 121–125.

[9] X. Li, J. Xu, and D. Huang, *An iterative learning control approach for linear systems with randomly varying trial lengths*, IEEE Trans. Automat. Control, 59 (2014), pp. 1954–1960.

[10] X. F. Li, J. X. Xu, and D. H., *An iterative learning control approach for linear systems with randomly varying trial lengths*, IEEE Trans. Automat. Control, 59 (2014), pp. 1954–1960.

[11] J. Liu, W. Chen, X. Ruan, and Y. Zheng, *Data-based iterative learning mechanism for unknown input-output coupling parameters/matrices*, Intern. J. Robust Nonlinear Control, 30 (2020), pp. 1275–1297.

[12] J. Liu and X. Ruan, *Convergence properties of two networked iterative learning control schemes for discrete-time systems with random packet dropout*, Internat. J. Systems Sci., 49 (2018), pp. 2682–2694.

[13] J. Liu and X. Ruan, *Synchronous-substitution-type iterative learning control for discrete-time networked control systems with bernoulli-type stochastic packet dropouts*, IMA J. Math. Control Inform., 35 (2018), pp. 939–962.

[14] J. Liu, X. Ruan, and Y. Zheng, *Iterative learning control for discrete-time systems with full learnability*, IEEE Trans. Neural Netw. Learn. Syst., 33 (2022), pp. 629–643.

[15] D. Meng, Y. Jia, J. Du, and S. Yuan, *Feedback approach to design fast iterative learning controller for a class of time-delay systems*, IET Control Theory Appl., 3 (2009), pp. 225–238.

[16] D. Meng and J. Zhang, *Convergence analysis of robust iterative learning control against nonrepetitive uncertainties: System equivalence transformation*, IEEE Trans. Neural Netw. Learn. Syst., 32 (2021), pp. 3867–3879.

[17] C. Mi, H. Lin, and Y. Zhang, *Iterative learning control of antilock braking of electric and hybrid vehicles*, IEEE Trans. Vehicular Technol., 54 (2005), pp. 486–494.

[18] X. Ruan, Z. Bien, and Q. Wang, *Convergence properties of iterative learning control processes in the sense of the Lebesgue-p norm*, Asian J. Control, 14 (2012), pp. 1095–1107.

[19] D. Shen and Z. Hou, *Iterative learning control with unknown control direction: A novel data-based approach*, IEEE Trans. Neural Netw., 22 (2011), pp. 2237–2249.

[20] D. Shen and J. Xu, *An iterative learning control algorithm with gain adaptation for stochastic systems*, IEEE Trans. Automat. Control, 65 (2020), pp. 1280–1287.

[21] D. Wang, *Convergence and robustness of discrete time nonlinear systems with iterative learning control*, Automatica, 34 (1998), pp. 1445–1448.

[22] J. Wu, Y. Zhu, Y. Zheng, and H. Wang, *Resilient bipartite consensus of second-order multi-agent systems with event-triggered communication*, IEEE Syst. J., (2021), https://doi.org/10.1109/JSYST.2021.3132623.

[23] Y. Zhang, J. Liu, and X. Ruan, *Equivalence and convergence of two iterative learning control schemes with state feedback*, Internat. J. Robust Nonlinear Control, 32 (2022), pp. 1561–1582, https://doi.org/10.1002/rnc.5891.