Research Article

Convergence Analysis of Iterative Learning Control for Two Classes of 2-D Linear Discrete Fornasini–Marchesini Model

Kai Wan

School of Electronic Information and Electrical Engineering, Huizhou University, Huizhou 516007, China

Correspondence should be addressed to Kai Wan; wankai606815@163.com

Received 2 July 2020; Revised 23 August 2020; Accepted 1 September 2020; Published 14 September 2020

Academic Editor: Guangchen Zhang

Copyright © 2020 Kai Wan. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper first investigates convergent property of two iterative learning control (ILC) laws for two kinds of two-dimensional linear discrete systems described by the first Fornasini–Marchesini model (2-D LDFFM with a direct transmission from inputs to outputs and 2-D LDFFM with input delay). Different from existing ILC results for 2-D LDFFM, this paper provides convergence analysis in a three-dimensional (3-D) framework. By using row scanning approach (RSA) or column scanning approach (CSA), it is theoretically proved no matter which method is adopted, perfect tracking on the desired reference surface is accomplished. In addition, linear matrix inequality (LMI) technique is utilized to compute the learning gain of the ILC controller. The effectiveness and feasibility of the designed ILC law are illustrated through numerical simulation on a practical thermal process.

1. Introduction

In practical industrial applications for two-dimensional (2-D) dynamical systems, for example, in form-closure grasps, the immobilized manipulation of 2-D serial chains could be regarded as a repetitive control problem [1]. Also, mold processing and material manufacturing usually require repeated operations of the controlled processing units to obtain a high-precision reference surface [2]. With regard to this kind of repetitive tracking problem, the traditional control methods in infinite coordinate domain, such as $H_{\infty}$ control [3], stochastic stability analysis [4], and sliding mode control [5], are difficult to be used because the precise model information is known in advance. Clearly, for the repetitively tracking cases mentioned above, the traditional tracking control approaches for 2-D dynamical systems in infinite coordinate domain are difficult to be applicable. Clearly, for the repetitively tracking cases mentioned above, the traditional tracking control approaches for 2-D dynamical systems in infinite coordinate domain are difficult to be applicable. Clearly, for the repetitively tracking cases mentioned above, the traditional tracking control approaches for 2-D dynamical systems in infinite coordinate domain are difficult to be applicable.

To the best of our knowledge, iterative learning control (ILC), as a data-driven and unsupervised control approach, does not require accurate knowledge of the controlled system, which makes ILC be widely prevalent in practical applications. A large number of ILC research results reported in the past few decades have fundamentally designed for one-dimensional (1-D) dynamical systems [6–14], only very few results involved in 2-D dynamical systems [15–22], which concentrate on mainly 2-D linear discrete first Fornasini–Marchesini model (2-D LDFFM). An optimal ILC algorithm was proposed in [16], such that the ILC tracking error converges to zero monotonically. Unfortunately, the system parameters of 2-D LDFFM are known. In [15], five ILC algorithms were proposed, the effectiveness and feasibility of which is verified through only numerical example. Afterwards, by using the column scanning approach (CSA), the iterative process of 2-D LDFFM is transformed into an iterative process of 1D systems; rigorously, theoretical proof on convergence and robustness of a two-gain ILC law is provided in [17]. Also, using the CSA as [18], to track a class of nonrepetitive reference surface described by a high-order internal model operator (HOIM), two HOIM-based ILC laws were, respectively, investigated in [23] for 2-D LDFFM by using 2-D HOIM-based linear inequality theory, but the ultimate ILC tracking error can only converge to a bounded range. To this end, adaptive ILC approach was proposed in [21, 22] to identify all unknown system parameters of 2-D
LDFFM, and the ILC result of perfect tracking on iteration-varying reference surface can be obtained. Unfortunately, the gain matrix in 2-D LDFFM is required to be positive definite, such that the proposed adaptive ILC algorithm, in practical applications, is greatly restricted. Notably, the aforementioned ILC results have concerned mainly on 2-D LDFFM without input delay.

Time delay, recognized as one of the main sources for poor performance or even instability of control systems, is frequently encountered in the signal transmission process of engineering and biological systems [24]. Over the past few years, with respect to 2-D systems with time delay on horizontal direction and vertical direction, some initial results have been reported on the states estimation [25], $H_{\infty}$ control and filtering [26, 27], and nonfragile robust optimal guaranteed cost control [28]. For instance, the authors in [23] have investigated the robust state estimation problem for 2-D linear discrete systems with state delay, and a robust $H_{\infty}$ filter for 2-D discrete systems with time delays has been designed in [23]. A delay-dependent $H_{\infty}$ controller for 2-D discrete state delay systems was designed in [27]. Nonfragile robust optimal guaranteed cost control subject to both state delay and input delay is discussed in [28]. However, no works have considered the ILC tracking problem for 2-D linear discrete systems with input delay, which motivates our current study.

This paper first investigates the convergence property of two ILC laws for 2-D LDFFM with a direct transmission from inputs to outputs and 2-D LDFFM with input delay to track the desired reference surface, respectively. The 2-D LDFFM is very typical, and many practical 2-D engineering systems can be represented as the form of 2-D LDFFM [29, 30]. Under a 3-D framework, a 3-D dynamical process is converted into a 2-D dynamical process by using CSA or row scanning approach (RSA), and it is theoretically proved no matter which method is used; perfect tracking on the desired reference surface is fulfilled. Additionally, this paper theoretically provides a linear matrix inequality (LMI) technique to compute the learning gain matrix of the ILC law. The main contributions of this paper are summarized as follows:

1. All of the existing ILC results for 2-D dynamical systems have not considered input delay. To the best of our knowledge, it is the first time to investigate ILC algorithms for 2-D LDFFM with a direct transmission from inputs to outputs and for 2-D LDFFM with input delay.

2. Compared with the adaptive ILC algorithm for 2-D LDFFM in [21, 22], two ILC algorithms proposed in this paper have no restriction on the numbers of system inputs and outputs.

3. Different from the existing ILC work for 2-D LDFFM [17, 18], a 3-D framework learning mechanism is presented in this paper and can reveal the dynamical behavior of the 2-D LDFFM in the horizontal dynamical direction, vertical dynamical direction, and iteration direction.

4. It is theoretically proved that RSA or CSA can guarantee perfect tracking on the desired reference surface.

5. Compared with the literature [20], this paper has addressed the ILC issue on nonidentical boundary states. In addition, this paper successfully avoids strong Assumption 2.2 on desired reference surface and Assumption 2.3 on the system matrix, and LMI technique is utilized to computer the learning gain of the ILC controller.

The rest of this paper is arranged as follows: Section 2 presents problem formulation and some preliminaries. Sections 3 and 4 provide convergence analysis of a symmetrical P-type ILC law and extend to 2-D LDFFM with input delay, respectively. Two simulation examples are introduced in Section 5. Finally, a conclusion on this article is drawn in Section 6. Notations: $A>0$ ($A<0$) denotes positive-definite (negative-definite) matrix. $[i]_{m \times n}^m$ represents $i \in \{m_1, m_1 + 1, m_1 + 2, \ldots, m_2\}$ with positive integer $m_2$, and $[i]_{m}^{\infty}$ represents $i \in \{0, 1, 2, \ldots\}$. $I_{m}$ denotes identity matrix with dimension $m \times m$. $\rho(\cdot)$ represents spectral radius of matrix.

2. Problem Formulation and Some Preliminaries

Consider the following 2-D LDFFM with a direct transmission from inputs to outputs [23], executing designated tracking tasks repetitively over dynamical region $[n_1]_{0}^{N_1}$ and $[n_2]_{0}^{N_2}$:

$$\begin{align*}
  x_k(n_1 + 1, n_2 + 1) &= A_1 x_k(n_1 + 1, n_2) + A_2 x_k(n_1, n_2)
  + A_3 (n_1, n_2 + 1) + B u_k(n_1, n_2),
  
y_k(n_1, n_2) &= C x_k(n_1, n_2) + D u_k(n_1, n_2),
\end{align*}$$

(1)

where $u_k(n_1, n_2) \in R^m$, $x_k(n_1, n_2) \in R^d$, and $y_k(n_1, n_2) \in R^l$ represent, respectively, control input, system state, and control output; $A_1, A_2, A_3, B, C,$ and $D$ are real matrices with appropriate dimensions. $[k]_0^\infty$ denotes the $k$-th repetitive number (or iteration number) of 2-D LDFFM (1); $n_1$ and $n_2$ are, respectively, discrete indexes along the horizontal dynamical direction and vertical dynamical direction. The independent indexes $n_1$ and $n_2$ in practical 2-D LDFFM, i.e., chemical reactors, heater exchangers, and pipe furnaces, usually represent space locations and time instants, respectively [31]. As $D = 0$, 2-D LDFFM (1) has been investigated in [17, 18].

For $[n_1]_{0}^{N_1}$ and $[n_2]_{0}^{N_2}$, let the desired trajectory surface and the corresponding tracking error at $k$th iteration, respectively, be denoted as $y_*(n_1, n_2)$ and $e_k(n_1, n_2)$, and the control objective of ILC for 2-D LDFFM is to update the control input $u_k(n_1, n_2)$ iteratively, such that the system output $y_k(n_1, n_2)$ converges to desired reference surface $y_*(n_1, n_2)$ asymptotically, i.e.,

$$\lim_{k \to \infty} y_k(n_1, n_2) = y_*(n_1, n_2), \quad [n_1]_{1}^{N_1}, [n_2]_{1}^{N_2}. \quad (2)$$
For the convenience of discussing the ILC problem for 2-D LDFFM, the property on a block matrix, assumptions, and Lemma is provided as follows.

**Property 1.** (see [32]). Give the following block matrix $H$ with a nonsingular matrix $H_{j} \in \mathbb{R}^{m \times m}$:

$$H = \begin{bmatrix}
H_{1} & 0 & 0 & \cdots & 0 \\
-H_{2} & H_{1} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
0 & \cdots & \cdots & \cdots & -H_{2} H_{1}
\end{bmatrix} \in \mathbb{R}^{m \times m \times N},$$

(3)

and the inverse matrix of which is given as

$$H^{-1} = \begin{bmatrix}
\overline{H}_{1} & 0 & 0 & \cdots & 0 \\
\overline{H}_{2} & \overline{H}_{1} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\overline{H}_{N} & \cdots & \overline{H}_{2} & \overline{H}_{1} & 0
\end{bmatrix} \in \mathbb{R}^{m \times m \times N},$$

(4)

with

$$\overline{H}_{1} = H_{1}^{-1},$$

$$\overline{H}_{2} = H_{1}^{-1} H_{2} H_{1}^{-1},$$

$$\overline{H}_{3} = (H_{1}^{-1} H_{2}) H_{1}^{-1},$$

$$\vdots$$

$$\overline{H}_{N} = (H_{1}^{-1} H_{2})^{N-1} H_{1}^{-1}.$$  

(5)

In particular, as $H_{1} = I_{m}$, $H^{-1}$ becomes

$$H^{-1} = \begin{bmatrix}
I_{m} & 0 & 0 & \cdots & 0 \\
H_{2} I_{m} & 0 & \cdots & \vdots \\
H_{2}^{2} H_{2} I_{m} & 0 & \cdots & \vdots \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
H_{2}^{N-1} \cdots H_{2}^{2} H_{2} I_{m}
\end{bmatrix}.$$

(6)

**Property 2** (see [32]). For $H^{-1}$ in Property 1, according to matrix theory, there is $\rho(H^{-1}) = \rho(\overline{H}_{1})$, which is irrelevant to $\overline{H}_{2}, \overline{H}_{3}, \ldots,$ and $\overline{H}_{N}$.

**Remark 1.** Properties 1 and 2 are often used in the ILC convergence analysis for 2-D linear discrete dynamical systems to track reference surface described by a high-order internal model or varying uniformly between $(k + 1)$-th iteration and $k$-th iteration [18]. Certainly, Properties 1 and 2 are also suitable to track repetitive reference surface considered in this paper.

**Assumption 1.** The matrix $D$ is full-row rank.

**Assumption 2.** The boundary states $x_{k}(0, n_{2})$ and $x_{k}(n_{1}, 0)$ of 2-D LDFFM (1) are assumed to satisfy $x_{k}(0, n_{2}) = x_{0}(0, n_{2}),$ $[n_{2}]_{0}^{N_{2}}$ and $x_{k}(n_{1}, 0) = x_{0}(n_{1}, 0),$ $[n_{1}]_{0}^{N_{1}}$, where $x_{0}(0, n_{2})$ and $x_{0}(n_{1}, 0)$ are iteration-invariant functions with respective to $n_{2}$ and $n_{1}$.

**Lemma 1.** Consider the following 3-D linear discrete system for $[n_{1}]_{0}^{N_{1}}, [n_{2}]_{0}^{N_{2}},$ and $\{k\}^{\infty}$:

$$\theta_{k+1}(n_{1}, n_{2}) = \alpha_{1} \theta_{k}(n_{1}, n_{2}) + \alpha_{2} \delta_{k}(n_{1}, n_{2}),$$

(7)

where $\theta_{k}(n_{1}, n_{2}) \in \mathbb{R}^{4}$ and $\delta_{k}(n_{1}, n_{2}) \in \mathbb{R}^{4}$, respectively, are state and control input; $\alpha_{1}$ and $\alpha_{2}$ denote real matrices. Suppose the boundary state $\theta_{0}(n_{1}, n_{2}) = f(n_{1}, n_{2})$, where $f(n_{1}, n_{2})$ is a bounded vector function. When $k \to \infty$, $\delta_{k}(n_{1}, n_{2}) = 0$, if $\rho(\alpha_{1}) < 1$ is satisfied, then the 3-D system is asymptotically stable along the dynamical direction $k$, i.e.,

$$\lim_{k \to \infty} \theta_{k}(n_{1}, n_{2}) = 0, \quad [n_{1}]_{0}^{N_{1}}, [n_{2}]_{0}^{N_{2}}.$$  

(8)

In particular, when $\delta_{k}(n_{1}, n_{2}) = 0$, for $[n_{1}]_{0}^{N_{1}}$ and $[n_{2}]_{0}^{N_{2}}$,

$$\lim_{k \to \infty} \theta_{k}(n_{1}, n_{2}) = 0$$

can still be obtained for $[n_{1}]_{0}^{N_{1}}$ and $[n_{2}]_{0}^{N_{2}}$. The proof process of Lemma 1 is shown in Appendix.

**3. Convergence Analysis of a Symmetrical P-Type ILC Law**

The following symmetrical P-type ILC law is applied to 2-D LDFFM (1) for $[n_{1}]_{0}^{N_{1}}$ and $[n_{2}]_{0}^{N_{2}}$:

$$u_{k+1}(n_{1}, n_{2}) = u_{k}(n_{1}, n_{2}) + \Gamma e_{k}(n_{1}, n_{2}),$$

(9)

where $e_{k}(n_{1}, n_{2}) = y_{r}(n_{1}, n_{2}) - y_{k}(n_{1}, n_{2})$. The learning gain $\Gamma \in \mathbb{R}^{m \times 1}$ is to be designed.

**Remark 2.** It is worth noting that 2-D LDFFM (1) is a zero relative degree system; the proposed ILC law (9) is extended on the existing ILC work for 1D linear discrete dynamical systems with zero relative degree [33]. Additionally, 2-D LDFFM (1) and the ILC law (9) are symmetrical on the discrete indexes $[n_{1}]_{0}^{N_{1}}$ and $[n_{2}]_{0}^{N_{2}}$; therefore, row scanning approach (RSA) or column scanning approach (CSA) can guarantee the same ILC tracking result, which is demonstrated in Theorem 1 and Appendix.

**Theorem 1.** Consider the 2-D LDFFM with a direct transmission from inputs to outputs (1) under Assumptions 1 and 2 and use the symmetrical P-type ILC law (9). If the learning gain $\Gamma$ satisfies

$$\rho(I_{1} - DI) < 1,$$

(10)

then the tracking error $e_{k}(n_{1}, n_{2})$ converges to zero, i.e.,

$$\lim_{k \to \infty} e_{k}(n_{1}, n_{2}) = 0, \quad [n_{1}]_{0}^{N_{1}}, [n_{2}]_{0}^{N_{2}}.$$  

(11)

**Proof 1.** For $[n_{1}]_{0}^{N_{1}-1}$ and $[n_{2}]_{0}^{N_{2}-1}$, let

$$\delta x_{k}(n_{1}, n_{2}) = x_{k+1}(n_{1}, n_{2}) - x_{k}(n_{1}, n_{2}),$$

$$\delta u_{k}(n_{1}, n_{2}) = u_{k+1}(n_{1}, n_{2}) - u_{k}(n_{1}, n_{2}).$$

(12)
Using (1), for \( \{n_{10}\}^{N_{1}-1} \) and \( \{n_{20}\}^{N_{2}-1} \), there is
\[
\delta x_k(n+1, n_2 + 1) = x_{k+1}(n+1, n_2 + 1) - x_k(n+1, n_2 + 1) = A_1 \delta x_k(n_1, n_2) + A_2 \delta x_k(n_1, n_2) + A_3 \delta x_k(n_1, n_2) + B \delta u_k(n_1, n_2).
\]

(13)

Applying the ILC law (9) into (13), it generates
\[
\delta x_k(n+1, n_2 + 1) = A_3 \delta x_k(n_1, n_2) + A_2 \delta x_k(n_1, n_2) + A_3 \delta x_k(n_1, n_2) + B \Gamma \epsilon_k(n_1, n_2).
\]

(14)

On the contrary, according to \( \epsilon_k(n_1, n_2) = y_r(n_1, n_2) - y_k(n_1, n_2) \) and (1), we have
\[
\epsilon_{k+1}(n_1, n_2) - \epsilon_k(n_1, n_2) = \epsilon_{r}(n_1, n_2) - \epsilon_{k+1}(n_1, n_2) - y_{k}(n_1, n_2) + y_{k+1}(n_1, n_2)
\]
\[
= -C[x_{k+1}(n_1, n_2) - x_k(n_1, n_2)] - D[u_{k+1}(n_1, n_2) - u_k(n_1, n_2)].
\]

(15)

where \( \{n_{10}\}^{N_{1}} \) and \( \{n_{20}\}^{N_{2}} \). Substituting the ILC law (9) into (15), it yields
\[
\epsilon_{k+1}(n_1, n_2) - \epsilon_k(n_1, n_2) = -C[x_{k+1}(n_1, n_2) - x_k(n_1, n_2)] - D \Gamma \epsilon_k(n_1, n_2).
\]

(16)

Rearranging (16), there is
\[
\epsilon_{k+1}(n_1, n_2) = -C \delta x_k(n_1, n_2) + (I_1 - D \Gamma) \epsilon_k(n_1, n_2).
\]

(17)

Let
\[
\delta X_k(n_2) = \begin{bmatrix} \delta x_k^T(1, n_2) & \delta x_k^T(2, n_2) & \cdots & \delta x_k^T(N_1, n_2) \end{bmatrix}^T,
\]

(18)

\[
E_k(n_2) = \begin{bmatrix} e_1^T(0, n_2) & e_1^T(1, n_2) & \cdots & e_1^T(N_1, n_2) \\
\end{bmatrix}^T,
\]

(19)

where \( \{n_{10}\}^{N_{1}} \). According to (18) and (19), using the CSA on horizontal dynamical direction \( n_1 \), (14) can be rewritten as
\[
\Phi_1 \delta X_k(n_1, n_2 + 1) = \Phi_2 \delta X_k(n_1, n_2) + \Phi_3 E_k(n_2) + \Phi_4 \delta X_k(0, n_2)
\]
\[
+ \Phi_5 \delta x_k(0, n_2 + 1),
\]

(20)

where
\[
\Phi_1 = \begin{bmatrix} I_p & 0 & 0 & \cdots & 0 \\
-A_3 & I_p & 0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & -A_3 & I_p \\
\end{bmatrix} \in R^{pN_1 \times pN_1},
\]

\[
\Phi_2 = \begin{bmatrix} A_1 & 0 & 0 & \cdots & 0 \\
0 & A_2 & A_1 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & A_2 & A_1 \\
\end{bmatrix} \in R^{pN_1 \times pN_1},
\]

\[
\Phi_3 = \begin{bmatrix} B \Gamma & 0 & 0 & \cdots & 0 \\
0 & B \Gamma & \cdots & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & B \Gamma \\
\end{bmatrix} \in R^{pN_1 \times (N_1 + 1)},
\]

\[
\Phi_4 = \begin{bmatrix} A_2 \\
0 \\
\vdots \\
0 \\
\end{bmatrix} \in R^{pN_1 \times p},
\]

\[
\Phi_5 = \begin{bmatrix} A_3 \\
0 \\
\vdots \\
0 \\
\end{bmatrix} \in R^{pN_1 \times p}.
\]

Then, since \( \Phi_1 \) is a nonsingular matrix, premultiplying \( \Phi_1^{-1} \) on both sides of (20), we get
\[
\delta X_k(n_2 + 1) = \Phi_1^{-1} \delta X_k(n_2) + \Phi_2 E_k(n_2) + \Phi_3 \delta X_k(0, n_2)
\]
\[
+ \Phi_4 \delta x_k(0, n_2 + 1),
\]

(22)

where \( \Phi_1^{-1} = \Phi_1^{-1} \Phi_2 \) and \( \Phi_2 = \Phi_1^{-1} \Phi_3 \). Similarly, (17) can also rewritten as
\[
E_{k+1}(n_2) = \varphi_1 \delta X_k(n_2) + \varphi_2 E_k(n_2) - C \delta x_k(0, n_2),
\]

(23)

where
Remark 3. The convergence condition $\rho(I_1 - D\Gamma) < 1$ in Theorem 1 only depends on the matrix $D$ and is irrelevant to $A_1$, $A_2$, $A_3$, $C$, and $B$, which is in accordance with the ILC convergence condition of the following 1D linear discrete iterative system to track repetitive reference trajectory [33]:

$$
\begin{align*}
&x_k(t+1) = Ax_k(t) + Bu_k(t), \\
y_k(t) = Cx_k(t) + Du_k(t),
\end{align*}
$$

under a P-type ILC law $u_{k+1}(t) = u_k(t) + \Gamma e_k(t)$ and initial iterative state $x_k(0) = x_0(t)$.

Remark 4. Taking advantage of the estimated information on $D$, the learning gain matrix $\Gamma$ of $\rho(I_1 - D\Gamma) < 1$ can be determined by using LMI technique. For the following 1D linear discrete dynamical system described by $I_1 - D\Gamma$,

$$x(k+1) = \left(I_1 - D\Gamma\right)x(k), \quad [k]_0^{\infty}. \tag{32}$$

Asymptotically stable condition $\rho(I_1 - D\Gamma) < 1$ is satisfied if and only if there exists a $Q>0$ such that

$$\begin{bmatrix}
-Q^{-1} & I_1 - D\Gamma \\
I_1 - \Gamma^T D^T & -Q
\end{bmatrix} < 0. \tag{34}$$

Using Schur complement theorem [34], it yields

$$
\begin{bmatrix}
-Q^{-1} & \Gamma^T D^T \\
Q^{-1} - \Gamma^T D^T & -Q^{-1}
\end{bmatrix} < 0. \tag{35}
$$

Pre- and postmultiplying by $\text{diag}(I_1, Q^{-1})$ on both sides of (34), we have

$$
\begin{bmatrix}
-Q^{-1} & \Gamma^T D^T \\
Q^{-1} - \Gamma^T D^T & -Q^{-1}
\end{bmatrix} < 0. \tag{36}
$$

with $X = \Gamma Q^{-1}$. Therefore, the selection problem of the learning gain matrix $\Gamma$ is transformed into the feasibility problem on solving LMI (36).

Remark 5. Theorem 1 employs the CSA on dynamical direction $n_1$ to obtain ILC result under Assumption 2. As the RSA on dynamical direction $n_2$ is applied to 2-D LDFFM (1) and the ILC law (9), we can still acquire the same ILC result with Theorem 1 due to the symmetrical characteristics on $n_1$ and $n_2$ in 2-D LDFFM (1) and the ILC law (9), and see Appendix.

### 4. Extension to 2-D LDFFM with Input Delay

In this section, under Assumption 2, we investigate the ILC issues for 2-D LDFFM with input delay [23], which is described in the following for $[n_1]_{0}^{N_1-1}$ and $[n_2]_{0}^{N_2-1}$:

$$
\begin{align*}
x_k(n_1 + 1, n_2) &= A_1 x_k(n_1 + 1, n_2) + A_2 x_k(n_1, n_2) \\
&+ A_3 x_k(n_1, n_2 + 1) + B u_k(n_1 - \tau_1, n_2 - \tau_2), \\
y_k(n_1, n_2) &= C x_k(n_1, n_2),
\end{align*}
$$

(37)
where \(\tau_1\) and \(\tau_2\) are two positive integers. \(x_k(n_1, n_2)\), 
\(u_k(n_1, n_2)\), and \(y_k(n_1, n_2)\) have been described in (1).

**Remark 6.** It is worthwhile noting that ILC tracking issues for 2-D LDFFM with \(\tau_1 = 0\) and \(\tau_2 = 0\) in (37) have been investigated in [17, 18]. It is the first time that ILC issues for 2-D LDFFM with input delay are investigated in this paper.

**Assumption 3.** The matrix CB is full-row rank.

Next, we discuss the convergence property of the ILC law for 2-D LDFFM with input delay (37) under Assumptions 2 and 3, and there is the following Theorem 2.

**Theorem 2.** Consider the 2-D LDFFM with input delay (37) under Assumptions 2 and 3 and use the following ILC law for \(\{n_1\}_{N_1}^{N_1-\tau_1-1}\) and \(\{n_2\}_{N_2}^{N_2-\tau_2-1}\):

\[
u_{k+1}(n_1, n_2) = u_k(n_1, n_2) + \Gamma e_k(n_1 + \tau_1 + 1, n_2 + \tau_2 + 1).
\]

(38)

If the learning gain \(\Gamma\) satisfies

\[
\rho(I - C\Gamma) < 1,
\]

(39)

then the tracking error \(e_k(n_1, n_2)\) converges to zero, i.e.,

\[
\lim_{k \to +\infty} e_k(n_1, n_2) = 0, \quad \{n_1\}_{1}^{N_1}, \{n_2\}_{1}^{N_2}.
\]

(40)

**Proof 2.** For \(\{n_1\}_{0}^{N_1}\) and \(\{n_2\}_{0}^{N_2}\), let

\[
\delta x_k(n_2) = x_{k+1}(n_1, n_2) - x_k(n_1, n_2).
\]

From (37), there is

\[
\delta x_k(n_1, n_2 + 1) = x_{k+1}(n_1, n_2 + 1) - x_k(n_1, n_2 + 1),
\]

\[
= A_1 \delta x_k(n_1 + 1, n_2) + A_2 \delta x_k(n_1, n_2) + B\nu_{k+1}(n_1 - \tau_1, n_2 - \tau_2 - 1) - u_k(n_1 - \tau_1, n_2 - \tau_2 - 1),
\]

(41)

where \(\{n_1\}_{0}^{N_1-1}\) and \(\{n_2\}_{0}^{N_2-1}\). Inserting the ILC law (38) into (42), it generates

\[
\delta x_k(n_1, n_2 + 1) = A_1 \delta x_k(n_1 + 1, n_2) + A_2 \delta x_k(n_1, n_2) + B\nu_{k+1}(n_1 - \tau_1, n_2 - \tau_2 - 1) - u_k(n_1 - \tau_1, n_2 - \tau_2 - 1),
\]

(43)

On the contrary, according to \(e_k(n_1, n_2) = y_r(n_1, n_2) - y_k(n_1, n_2)\) and using (37), we have

\[
e_{k+1}(n_1 + 1, n_2) - e_k(n_1 + 1, n_2) = y_r(n_1 + 1, n_2) - y_{k+1}(n_1 + 1, n_2) - y_r(n_1 + 1, n_2) + y_k(n_1 + 1, n_2)
\]

\[
= -C[x_{k+1}(n_1 + 1, n_2) - x_k(n_1 + 1, n_2)]
\]

\[
= -C\delta x_k(n_1 + 1, n_2 + 1),
\]

(44)

where \(\{n_1\}_{0}^{N_1-1}\) and \(\{n_2\}_{0}^{N_2-1}\). Let

\[
\delta x_k(n_2) = \begin{bmatrix} \delta x_k^T(1, n_2) & \delta x_k^T(2, n_2) & \cdots & \delta x_k^T(N_1, n_2) \end{bmatrix}^T,
\]

(45)

\[
E_k(n_2) = \begin{bmatrix} e_k^T(1, n_2) & e_k^T(2, n_2) & \cdots & e_k^T(N_1, n_2) \end{bmatrix}^T,
\]

(46)

where \(\{n_2\}_{1}^{N_2} \). From (45) and (46), (43) can be rewritten as

\[
\phi_1 \delta x_k(n_2 + 1) + \phi_2 E_k(n_2) + \phi_3 \delta x_k(0, n_2) + \bar{A}_3 \delta x_k(0, n_2 + 1),
\]

(47)

where

\[
\phi_1 = \begin{bmatrix} I_p & 0 & \cdots & 0 \\ -A_3 & I_p & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -A_3 \end{bmatrix} \in \mathbb{R}^{N_1 \times pN_1},
\]

(48)

\[
\phi_2 = \begin{bmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & A_1 \end{bmatrix} \in \mathbb{R}^{N_1 \times pN_1},
\]

(49)

and \(\bar{A}_3\) and \(\tilde{A}_3\) have been described in (20). Similarly, (44) can also be reformulated as

\[
E_{k+1}(n_2) = \psi_1 \delta x_k(n_2 + 1) + \psi_2 E_k(n_2),
\]

(50)

where

\[
\psi_1 = \begin{bmatrix} -C & 0 & \cdots & 0 \\ 0 & -C & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -C \end{bmatrix} \in \mathbb{R}^{N_1 \times pN_1},
\]

(51)

\[
\psi_2 = \begin{bmatrix} I_l & 0 & \cdots & 0 \\ 0 & I_l & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & I_l \end{bmatrix} \in \mathbb{R}^{N_1 \times lN_1},
\]

(52)

Then, since \(\phi_1^{-1}\) is a nonsingular matrix, premultiplying \(\phi_1^{-1}\) on both sides of (47), we get
\[\delta X_k(n_2 + 1) = \psi_1 \delta X_k(n_2) + \phi_2 E_k(n_2) + \phi_1^{-1} A_2 \delta X_k(0, n_2)
+ \phi_1^{-1} A_1 \delta X_k(0, n_2 + 1),\]

where \(\phi_1 = \phi_1^{-1} \phi_2\) and \(\phi_2 = \phi_1^{-1} \phi_3\). Substituting (51) into (49), it yields
\[-E_{k+1}(n_2) = \psi_1 \delta X_k(n_2) + (\psi_1 \delta_2 + \psi_2) E_k(n_2)
+ \psi_1^{-1} A_2 \delta X_k(0, n_2) + \psi_1^{-1} A_1 \delta X_k(0, n_2 + 1).\]

(52)

where \([n_2]_{1}^{N} \). From Assumption 2, we obtain \(\delta x_k(0, n_2 + 1) = x_k(1, 0, n_2) - x_k(0, n_2) = 0\) for \([n_2]_{1}^{N} \). Thus, for \([n_2]_{1}^{N} \), (51) and (52) can be represented as
\[
\begin{align*}
\delta X_k(n_2 + 1) &= \phi_1 \delta X_k(n_2) + \phi_2 E_k(n_2), \\
E_{k+1}(n_2) &= \psi_1 \phi_1^{-1} A_2 \delta X_k(0, n_2) + \psi_1^{-1} A_1 \delta X_k(0, n_2 + 1).
\end{align*}
\]

(53)

In (53), based on Assumption 2, we obtain \(\delta x_k(n_1, 1) = x_{k+1}(n_1, 0) - x_k(n_1, 0) = 0\). Furthermore,
\[\psi_1 \phi_2 + \psi_2 =
\begin{bmatrix}
-C & 0 & \cdots & 0 \\
0 & -C & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & -C
\end{bmatrix}
\begin{bmatrix}
I_p & 0 & 0 & \cdots & 0 \\
-A_3 & I_p & 0 & \cdots & 0 \\
0 & -A_3 & I_p & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & -A_3 & I_p
\end{bmatrix}^{-1}
\begin{bmatrix}
I & 0 & 0 & \cdots & 0 \\
0 & I & \cdots & \cdots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & I
\end{bmatrix}
+ \begin{bmatrix}
B & 0 & \cdots & 0 \\
B & 0 & \cdots & \vdots \\
\vdots & \ddots & \cdots & \ddots \\
0 & \cdots & 0 & B
\end{bmatrix}
\begin{bmatrix}
0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & \vdots \\
\vdots & \ddots & \cdots & \ddots \\
0 & \cdots & 0 & 0
\end{bmatrix}
\end{align*}
\]

(57)

From (57) and Property 2, we can know \(\rho(\psi_1 \phi_2 + \psi_2) = \rho(I - CBG)\). Consequently, \(\rho(\psi_1 \phi_2 + \psi_2) < 1\) can be replaced by a more convenient condition \(\rho(I - CBG) < 1\) without using any knowledge on the system matrix \(A_1\) and the interval length \(N_1\) in the 2-D LDFFM with input delay (37).

Remark 8. In Remark 7, using the estimated information on \(C\) and \(B\), the learning gain matrix \(\Gamma\) of \(\rho(I - CBG) < 1\) can be decided by using the LMI technique. Therefore, a similar Remark 4 can be obtained.

Remark 9. When the RSA on dynamical direction \(n_2\) is applied to 2-D LDFFM with input delay (37) and the ILC law (38), the same ILC result with Theorem 2 can be obtained, the proof process of which is shown in Appendix 7.3.

Remark 10. It is noted that 2-D LDFFM (1) and (37) contain 2-D Attasi model as a special case, i.e., \(A_2 = -A_1 A_3 = -A_3 A_1\). And the 2-D Roesser model under some specified coefficient matrix requirements can also be converted into 2-D LDFFM [29]. Consequently, depending on the ILC results obtained from 2-D LDFFM (1) and (37), it is easy to extend 2-D Roesser model and 2-D Attasi model with input delay.

5. Illustrative Examples

To show the effectiveness and feasibility of the proposed ILC algorithms (9) and (38) for the 2-D LDFFM with a direct transmission from inputs to outputs (1) and 2-D LDFFM with input delay (37), two simulation examples are introduced in this section.

Example 1. Consider the following 2-D LDFFM with a direct transmission from inputs to outputs over dynamical region \([n_1]_{10}^{20}\) and \([n_2]_{10}^{20}\).
which is required to repetitively track a desired reference surface 

\[ y_r(n_1, n_2) = \left\lfloor \cos(2\pi (n_1 + n_2))/10 \sin(\pi (n_1 + n_2))/10 \right\rfloor T \] 

for \([n_{1,0}^{20}]\) and \([n_{2,0}^{20}]\) shown in Figure 1. Under Assumption 2, the boundary states of 2-D LDFFM (58) are given as \(x_k(0, n_2) = \left[ 0.1 \sin(n_2) \right]^T\), \([n_{2,1}^{20}]\), and \(x_k(n_1, 0) = \left[ -0.1 \sin(n_1) \ 0 \right]^T\), \([n_{1,0}^{20}]\). Using the proposed ILC law (9) with the initial control input \(u_0(n_1, n_2) = \left[ 0 \right]^T\) for \([n_{1,0}^{20}]\) and \([n_{2,0}^{20}]\), the gain matrix \(\Gamma\) is selected as \(\Gamma = \text{diag}(0.3, 0.3)\), which satisfies the convergence condition \(\rho(I - D\Gamma) = 0.6 < 1\) in Theorem 1. In order to evaluate the accuracy of ILC tracking, the following maximum absolute tracking error index

\[
\begin{align*}
\begin{bmatrix} x_k^{(1)}(n_1 + 1, n_2 + 1) \\ x_k^{(2)}(n_1 + 1, n_2 + 1) \end{bmatrix} &= \begin{bmatrix} 0.01 & 0.01 \\ 0.01 & 0.03 \end{bmatrix} \begin{bmatrix} x_k^{(1)}(n_1 + 1, n_2) \\ x_k^{(2)}(n_1 + 1, n_2) \end{bmatrix} + \begin{bmatrix} 0.01 & 0.02 \\ 0.01 & 0.04 \end{bmatrix} \begin{bmatrix} x_k^{(1)}(n_1, n_2) \\ x_k^{(2)}(n_1, n_2) \end{bmatrix} \\
&+ \begin{bmatrix} 0.01 & 0.02 \\ 0.01 & 0.3 \end{bmatrix} \begin{bmatrix} x_k^{(1)}(n_1, n_2 + 1) \\ x_k^{(2)}(n_1, n_2 + 1) \end{bmatrix} + \begin{bmatrix} 1 & 0.2 \\ 0.3 & 0.5 \end{bmatrix} \begin{bmatrix} u_k^{(1)}(n_1, n_2) \\ u_k^{(2)}(n_1, n_2) \end{bmatrix} .
\end{align*}
\]

\[
\begin{align*}
\begin{bmatrix} y_k^{(1)}(n_1, n_2) \\ y_k^{(2)}(n_1, n_2) \end{bmatrix} &= \begin{bmatrix} 1 & 0.2 \\ -0.2 & 1 \end{bmatrix} \begin{bmatrix} x_k^{(1)}(n_1, n_2) \\ x_k^{(2)}(n_1, n_2) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} u_k^{(1)}(n_1, n_2) \\ u_k^{(2)}(n_1, n_2) \end{bmatrix} .
\end{align*}
\]

is used. As a result, the ILC tracking error surfaces \(e_k^{(1)}(n_1, n_2)\) at \(k = 3, 5, 7, 20\), and \(e_k^{(2)}(n_1, n_2)\) at \(k = 4, 6, 8, 24\) are depicted in Figures 2 and 3, respectively. Also, Figure 4 presents the profiles of the ILC tracking error indexes \(\text{MATE}_k^{(1)}\) and \(\text{MATE}_k^{(2)}\) in iteration domain \(k\) by utilizing the ILC law (9). Obviously, it is observed from Figures 2–4 that the convergent effect of the ILC law (9) against the iteration-invariant boundary states is illustrated.

\[
\text{MATE}_k^{(h)} = \max_{n_1, n_2} \max_{[n_{1,0}^{m}], [n_{2,0}^{m}]} \left| y_r^{(h)}(n_1, n_2) - y_k^{(h)}(n_1, n_2) \right|, \quad [h]_1^2
\]
Example 2. Some practical thermal processes can be represented by the following partial differential equation (PDE) \[31\]:

\[
\begin{align*}
\frac{\partial^2 s_k(w, t)}{\partial w \partial t} &= a_1 \frac{\partial s_k(w, t)}{\partial t} + a_2 \frac{\partial s_k(w, t)}{\partial w} + a_3 s_k(w, t) \\
&+ bf_k (w - d_w, t - d_t),
\end{align*}
\]  

where \(d_w\) and \(d_t\), respectively, denote space delay and time delay; \(a_0, a_1, a_2,\) and \(b\) are real coefficients. The repetitive index \(k\) implies that the ILC issue is investigated in the example. Through the discretization approximation on \[60\], system parameters and delay factors of 2-D LDFFM with input delay \(37\) are given as follows:
\[ A_1 = 1 + a_2 \Delta t, \]
\[ A_2 = a_0 \Delta w \Delta t - a_1 \Delta w - a_2 \Delta t - 1, \]
\[ A_3 = 1 + a_1 \Delta w, \]
\[ B = b \Delta w \Delta t, \]
\[ r_1 = \text{int} \left( \frac{d_w}{\Delta t} \right), \]
\[ r_2 = \text{int} \left( \frac{d_i}{\Delta t} \right), \]

where is shown in Figure 5. And the boundary states satisfy \( x_k(0, n_2) = 0.5 \sin(2\pi n_2/10) \), \( [n_{11}\, n_{12}] \), and \( x_k(n_1, 0) = 0.5 \sin(2\pi n_1/10) \), \( [n_{21}\, n_{22}] \). Under the ILC law (38) with initial control input \( u_0(n_1, n_2) = 0 \) for \( \{n_{11}\} \) and \( \{n_{21}\} \), let the learning gain \( \Gamma \) be selected as \( \Gamma = 0.9 \) and compute \( \rho(L_1 - CB) = 0.7885 < 1 \), which satisfies the convergent condition in Theorem 2. The maximum absolute tracking error index \( \text{MATE}_k \) is adopted to evaluate the ILC tracking performance, which is given as follows:

\[ \text{MATE}_k = \max_{\{n_{11}\}, \{n_{21}\}} \max \left| y_r(n_1, n_2) - y_k(n_1, n_2) \right|. \]

And Figure 6 displays the profile of \( \text{MATE}_k \) with iteration number \( k \) by using the ILC law (38). The tracking error surface \( e_k(n_1, n_2) \) at \( k = 5, 10, 15, 40 \) is shown in Figure 7. Apparently, perfect tracking on repetitive reference surface can be observed from Figures 6 and 7.

### 6. Conclusions

To date, ILC tracking issues for 2-D LDFFM with a direct transmission from inputs to outputs and with input delay are first concerned in this paper. Different from existing ILC results for 2-D LDFFM, a 3-D analysis approach is first applied to the convergent property of the proposed ILC law, which supplies a new tool for follow-up ILC development for 2-D systems. Additionally, the LMI technique is used to compute the learning gain. In our future work, reinforcement learning (RL) [35–38] will be incorporated into the ILC designs for 2-D dynamical systems such that more ILC results are made.

### Appendix

**Proof of Lemma 1.** The solution to \( \theta_{k+1}(n_1, n_2) = \alpha_1 \theta_k(n_1, n_2) + \alpha_2 \delta_k(n_1, n_2) \) is given as follows:

\[ \theta_k(n_1, n_2) = \alpha_1^{k-1} \theta_0(n_1, n_2) + \sum_{i=0}^{k-1} \alpha_1^{k-i-1} \alpha_2 \delta_i(n_1, n_2). \]  
(A.1)

Then, taking the limit on both sides of (A.1), there is

\[ \lim_{k \to \infty} \theta_k(n_1, n_2) = \lim_{k \to \infty} \alpha_1^{k-1} \theta_0(n_1, n_2) + \sum_{i=0}^{k-1} \alpha_1^{k-i-1} \alpha_2 \delta_i(n_1, n_2). \]  
(A.2)

Under \( \lim \theta_k(n_1, n_2) = 0 \) in Lemma 1, if \( \rho(\alpha_1) < 1 \) holds, we obtain

\[ \lim_{k \to \infty} \theta_k(n_1, n_2) = 0. \]  
(A.3)

Lemma 1 is completed.
Proof of Theorem 1 by using the RSA. Let

\[
\delta \mathbf{X}_k(n_i) = \left[ \delta \mathbf{x}_k^T(n_i, 1) \delta \mathbf{x}_k^T(n_i, 2) \cdots \delta \mathbf{x}_k^T(n_i, N_2) \right]^T,
\]

(A.4)

\[
\mathbf{E}_k(n_i) = \left[ \mathbf{e}_k^T(n_i, 0) \mathbf{e}_k^T(n_i, 1) \cdots \mathbf{e}_k^T(n_i, N_2) \right]^T. \tag{A.5}
\]

From (A.4) and (A.5), using the RSA on vertical direction \(n_2\), for \(\{n_1\}_{0}^{N_1-1}\), (14) can be rewritten as

\[
\begin{align*}
\Phi_1 \delta \mathbf{X}_k(n_i + 1) &= \Phi_1 \Phi_2 \delta \mathbf{X}_k(n_i) + \Phi_1 \mathbf{E}_k(n_i) + \tilde{\mathbf{A}}_1 \delta \mathbf{x}_k(n_i, 0) \\
&\quad + \tilde{\mathbf{A}}_2 \delta \mathbf{x}_k(n_i + 1, 0),
\end{align*}
\]

(A.6)

where

\[
\Phi_1 = \begin{bmatrix}
I_p & 0 & 0 & \cdots & 0 \\
-A_1 & I_p & 0 & \cdots & 0 \\
0 & -A_1 & I_p & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & -A_1 & I_p
\end{bmatrix} \in \mathbb{R}^{pN_1 \times pN_2},
\]

\[
\Phi_2 = \begin{bmatrix}
A_1 & 0 & 0 & \cdots & 0 \\
A_2 & A_1 & 0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & A_2 & A_1
\end{bmatrix} \in \mathbb{R}^{pN_2 \times pN_2},
\]

\[
\Phi_3 = \begin{bmatrix}
B^T & 0 & 0 & \cdots & 0 \\
0 & B^T & \ddots & \ddots & \vdots \\
0 & 0 & B^T & \ddots & 0 \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & 0 & B^T
\end{bmatrix} \in \mathbb{R}^{pN_1 \times p(N_1+1)},
\]

(A.7)

\[
\tilde{\mathbf{A}}_1 = \begin{bmatrix}
A_1 \\
0 \\
\vdots \\
0
\end{bmatrix} \in \mathbb{R}^{pN_1 \times p},
\]

\[
\tilde{\mathbf{A}}_2 = \begin{bmatrix}
A_2 \\
0 \\
\vdots \\
0
\end{bmatrix} \in \mathbb{R}^{pN_1 \times p}.
\]

Since \(\Phi_1\) is a nonsingular matrix, premultiplying \(\Phi_1^{-1}\) on both sides of (A.6), we get

\[
\delta \mathbf{X}_k(n_i + 1) = \Phi_1^{-1} \Phi_2 \delta \mathbf{X}_k(n_i) + \Phi_1^{-1} \Phi_2 \mathbf{E}_k(n_i) + \tilde{\Phi}_1^{-1} \tilde{\mathbf{A}}_1 \delta \mathbf{x}_k(n_i, 0) + \tilde{\Phi}_1^{-1} \tilde{\mathbf{A}}_2 \delta \mathbf{x}_k(n_i + 1, 0). \tag{A.8}
\]

Similarly, (17) can also be reformulated as

\[
\mathbf{E}_{k+1}(n_i) = \varphi_1 \delta \mathbf{X}_k(n_i) + \varphi_2 \mathbf{E}_k(n_i) - \tilde{\mathbf{C}} \delta \mathbf{x}_k(n_i, 0), \tag{A.9}
\]

where

\[
\Phi_1 = \begin{bmatrix}
0 & 0 & 0 & \cdots & 0 \\
-C & 0 & 0 & \cdots & 0 \\
0 & -C & 0 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & -C & 0
\end{bmatrix} \in \mathbb{R}^{(N_1+1) \times pN_2},
\]

\[
\varphi_1 = \begin{bmatrix}
I_1 - DI^T & 0 & \cdots & 0 \\
0 & I_1 - DI^T & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & I_1 - DI^T
\end{bmatrix} \in \mathbb{R}^{(N_1+1) \times (N_1+1)},
\]

\[
\varphi_2 = \begin{bmatrix}
C \\
0 \\
0 \\
0 
\end{bmatrix} \in \mathbb{R}^{(N_1+1) \times p}.
\]

From Assumption 2, we obtain \(\delta_k(n_1, 0) = x_{k,1}(n_1, 0) - x_k(n_1, 0) = 0\) for \(\{n_1\}_{0}^{N_1-1}\). Thus, (A.8) and (A.9) can be expressed as

\[
\begin{align*}
\delta \mathbf{X}_k(n_i + 1) &= \Phi_1^{-1} \Phi_2 \delta \mathbf{X}_k(n_i) + \Phi_1^{-1} \Phi_2 \mathbf{E}_k(n_i) \\
\mathbf{E}_{k+1}(n_i) &= \varphi_1 \delta \mathbf{X}_k(n_i) + \varphi_2 \mathbf{E}_k(n_i),
\end{align*}
\]

(111)

where \(\{n_1\}_{0}^{N_1-1}\). In (111), according to Assumption 2, we deduce

\[
\delta \mathbf{X}_k(0) = \left[ \delta \mathbf{x}_k^T(0, 1) \delta \mathbf{x}_k^T(0, 2) \cdots \delta \mathbf{x}_k^T(0, N_2) \right]^T = 0,
\]

(122)

and \(E_0(n_1)\) is bounded for \(\{n_1\}_{0}^{N_1-1}\). Using Lemma 4 in [18] and Property 2, if \(\rho(I_1 - DI^T) < 1\), there is

\[
\delta \mathbf{X}_k(n_i) = \left[ \lim_{k \to \infty} \delta \mathbf{X}_k(n_i) \right] = 0, \quad \{n_1\}_{0}^{N_1-1},
\]

(133)

In addition, taking \(n_1 = N_1\) in (17), we have

\[
e_{k+1}(N_1, n_2) = -C \delta \mathbf{x}_k(N_1, n_2) + (I_1 - DI^T) e_k(N_1, n_2).
\]

(144)

From (133) and (144), if \(\delta \mathbf{x}_k(N_1, 0) = 0\), using Lemma 1, if \(\rho(I_1 - DI^T) < 1\), it yields

\[
\lim_{k \to \infty} e_k(N_1, n_2) = 0, \quad \{n_2\}_{0}^{N_2}.
\]

(155)

From (133) and (155) and the definition of \(\mathbf{E}_k(n_i)\) in (A.5), we obtain

\[
\lim_{k \to \infty} e_k(n_1, n_2) = 0, \quad \{n_1\}_{0}^{N_1}, \{n_2\}_{0}^{N_2}.
\]
Theorem 1 is completed.

**Proof of Theorem 2 by using the RSA.** For \(|n_1|_{0}^{N_1}\) and \(|n_2|_{0}^{N_2}\), define

\[
\delta \tilde{x}_k(n_1 + 1, n_2) = x_{k+1}(n_1, n_2) - x_k(n_1, n_2). \tag{A.17}
\]

From (A.17), there is

\[
\begin{align*}
\delta \tilde{x}_k(n_1 + 1, n_2 + 1) &= A_1 \delta \tilde{x}_k(n_1 + 1, n_2) + A_2 \delta \tilde{x}_k(n_1, n_2) \\
&\quad + B[\eta_{k+1}(n_1 - r_1 - 1, n_2 - r_2) \\
&\quad - \eta_k(n_1 - r_1 - 1, n_2 - r_2)],
\end{align*}
\tag{A.18}
\]

where \(|n_1|_{0}^{N_1}\) and \(|n_2|_{0}^{N_2}\). Inserting the ILC law (38) into (A.18), it generates

\[
\begin{align*}
\delta \tilde{x}_k(n_1 + 1, n_2 + 1) &= A_1 \delta \tilde{x}_k(n_1 + 1, n_2) + A_2 \delta \tilde{x}_k(n_1, n_2) \\
&\quad + A_3 \delta \tilde{x}_k(n_1, n_2 + 1) \\
&\quad + B \delta \eta_k(n_1, n_2 + 1).
\end{align*}
\tag{A.19}
\]

On the contrary, according to \(e_k(n_1, n_2) = y_r(n_1, n_2) - y_k(n_1, n_2)\) and using (A.17), we have

\[
\begin{align*}
e_{k+1}(n_1, n_2 + 1) - e_k(n_1, n_2 + 1) \\
&= y_r(n_1, n_2 + 1) - y_{k+1}(n_1, n_2 + 1) - y_r(n_1, n_2 + 1) \\
&\quad + y_k(n_1, n_2 + 1) \\
&= -C[x_{k+1}(n_1, n_2 + 1) - x_k(n_1, n_2 + 1)] \\
&= -C \delta \tilde{x}_k(n_1 + 1, n_2 + 1).
\end{align*}
\tag{A.20}
\]

where \(|n_1|_{0}^{N_1}\) and \(|n_2|_{0}^{N_2}\). Let

\[
\begin{align*}
\delta \tilde{X}_k(n_1) &= [\delta x_k^T(n_1, 1) \delta x_k^T(n_1, 2) \cdots \delta x_k^T(n_1, N_2)]^T, \tag{A.21}
\end{align*}
\]

\[
\tilde{E}_k(n_1) = [e_k^T(n_1, 1) e_k^T(n_1, 2) \cdots e_k^T(n_1, N_2)]^T. \tag{A.22}
\]

where \(|n_1|_{0}^{N_1}\). From (A.21) and (A.22), using the RSA on vertical dynamical direction \(n_2\), (A.19) can be rewritten as

\[
\begin{align*}
\Phi_1 \delta \tilde{X}_k(n_1 + 1) &= \Phi_2 \delta \tilde{X}_k(n_1) + \Phi_3 \tilde{E}_k(n_1) + \bar{A}_2 \delta \tilde{x}_k(n_1, 0) \\
&\quad + \bar{A}_1 \delta \tilde{x}_k(n_1 + 1, 0), \quad |n_1|_{0}^{N_1-1}, \tag{A.23}
\end{align*}
\]

where

\[
\begin{align*}
\Phi_1 &= \begin{bmatrix}
I_p & 0 & \cdots & 0 \\
-A_1 & I_p & \cdots & 0 \\
& \vdots & \ddots & \vdots \\
& \vdots & \ddots & 0 \\
0 & \cdots & 0 & I_p
\end{bmatrix} \\
\Phi_2 &= \begin{bmatrix}
A_1 & 0 & \cdots & 0 \\
A_2 & A_3 & \cdots & 0 \\
& \vdots & \ddots & \vdots \\
& \vdots & \ddots & 0 \\
0 & \cdots & 0 & A_2 \end{bmatrix},
\end{align*}
\tag{A.24}
\]

Premultiplying \(\Phi_1^{-1}\) on both sides of (A.23), we get

\[
\delta \tilde{X}_k(n_1 + 1) = \Phi_2 \delta \tilde{X}_k(n_1) + \Phi_2 \tilde{E}_k(n_1) \tag{A.25}
\]

Similarly, (A.20) can also be reformulated as

\[
\bar{E}_{k+1}(n_1) = \bar{\Phi}_2 \delta \tilde{x}_k(n_1 + 1) + \bar{\Phi}_2 \tilde{E}_k(n_1). \tag{A.26}
\]
where

\[
\bar{\varphi}_1 = \begin{bmatrix} -C & 0 & \cdots & 0 \\ 0 & -C & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -C \end{bmatrix} \in \mathbb{R}^{N_2 \times pN_2},
\]

\[
\bar{\varphi}_2 = \begin{bmatrix} I \_1 & 0 & \cdots & 0 \\ 0 & I \_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & I \_1 \end{bmatrix} \in \mathbb{R}^{pN_1 \times N_2}.
\]

Substituting (A.25) into (A.26), consider

\[
\begin{align*}
\delta \bar{x}_k(n_1 + 1, 0) &= \bar{\varphi}_1 \bar{\varphi}_3 \delta \bar{x}_k(n_1) + \bar{\varphi}_1 \bar{\varphi}_3 \bar{E}_k(n_1), \\
\bar{E}_k(n_1) &= \bar{\varphi}_1 \bar{\varphi}_3 \bar{\varphi}_3 \delta \bar{x}_k(n_1) + \left( \bar{\varphi}_1 \bar{\varphi}_3 \bar{\varphi}_3 + \bar{\varphi}_2 \right) \bar{E}_k(n_1),
\end{align*}
\]

(A.28)

where \(n_1 \in N_1\). Based on Assumption 2, the initial condition in (A.28) is given as

\[
\delta \bar{x}_k(1) = \left[ \delta \bar{x}_k(1, 1) \delta \bar{x}_k(1, 2) \cdots \delta \bar{x}_k(1, N_2) \right]^T = 0,
\]

(A.29)

and \(\bar{E}_0(n_1)\) is bounded for \(n_1 \in N_1\). Using Lemma 4 in [18], if

\[
\rho(\bar{\varphi}_1 \bar{\varphi}_3 \bar{\varphi}_3) \leq 1 \quad \text{(equivalently, } \rho(I \_1 - CBT) < 1, \text{ see Property 2)},
\]

there is

\[
\lim_{k \to \infty} \bar{E}_k(n_1) = 0, \quad \{n_1 \in N_1\}.
\]

(A.30)

From (A.22), it yields

\[
\lim_{k \to \infty} \varepsilon_k(n_1, n_2) = 0, \quad \{n_1 \in N_1, \{n_2 \in N_2\}.
\]

(A.31)

Theorem 2 is completed.

Data Availability

All the data have been included in the paper; therefore, there are no other data available.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant nos. 61903096, 61573385, and 61703444), Science and Technology Program of Guangzhou (Grant no. 201904010475), Professorial and Doctoral Scientific Research Foundation of Huizhou University (Grant no. 2020JBJ017), and Huizhou Science and Technology Plan Project (Grant no. 2020SD0406034).

References

[18] K. Wan and X. D. Li, "High-order internal model based iterative learning control for 2-D linear FMMI systems with


