BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 70(3), 2022, Article number: e140934
DOI: 10.24425/bpasts.2022.140934

CONTROL AND INFORMATICS

Iterative learning fault-tolerant control for the networked control systems with initial state disturbance

Fu XINGJIAN* and **Zhao QIANJUN**

School of Automation, Beijing Information Science and Technology University, Beijing 100192, China

Abstract. The iterative learning fault-tolerant control strategies with non-strict repetitive initial state disturbances are studied for the linear discrete networked control systems (NCSs) and the nonlinear discrete NCSs. In order to reduce the influence of the initial state disturbance in iteration, for the linear NCSs, considering the external disturbance and actuator failure, the iterative learning fault-tolerant control strategy with impulse function is proposed. For the nonlinear NCSs, the external disturbance, packet loss and actuator failure are considered, the iterative learning fault-tolerant control strategy with random Bernoulli sequence is provided. Finally, the proposed control strategies are used for simulation research for the linear NCSs and the nonlinear NCSs. The results show that both strategies can reduce the influence of the initial state disturbance on the tracking effect, which verifies the effectiveness of the given method.

Key words: networked control systems; initial state disturbance; iterative learning; actuator failure; fault-tolerant control.

1. INTRODUCTION

With the development of control theory, computer technology and modern network communication technology, the disciplines intersect each other, and control systems become more and more complex. The controlled objects are becoming increasingly diverse. These make the NCSs develop rapidly [1–3]. The introduction of the NCSs has improved the information transmission efficiency and resource utilization rate, and the control cost has also been reduced. At the same time, the application of the NCSs breaks the space limitations for the traditional control system, which enables information exchange and joint work between links in different locations. Therefore, the remote control for the sensors, actuators and controllers can be easily realized. Based on the above advantages, NCSs have broad application prospects in many fields [4–7]. Therefore, the research on the NCSs is very meaningful.

Due to the diversity and unpredictability of network conditions, the network will also bring adverse effects and new problems to the control system. For example, network-induced time-delays, data packet loss, communication constraints, quantization errors and other faults which may be caused by the network [8–11]. These problems can not only destroy the original stability, but also affect the performance of the system. On the other hand, some components in the system, such as sensors and actuators, will also produce certain failures and affect the stability [12, 12–15]. In addition, the external disturbances will also affect system stability and performance. These failures will have a greater or lesser impact on the stability of the system. Therefore, the precise mathematical model for the controlled system is difficult to establish, which makes the research on the NCSs more challenging.

As a typical model-free control, the data-driven iterative learning control (ILC) algorithm is the most attractive method for the control systems with repetitive operation characteristics [16–20]. It does not need to understand the dynamic structure of the controlled object, and only needs to use the input and output information to design the controller, which can realize the tracking for the desired target within a limited operating interval. According to this characteristic, by using the data-driven iterative learning control to study the problems for the complex NCSs, the establishment of precise mathematical models is avoided. This simplifies the analysis and research of NCSs. Therefore, the combination of the data-driven iterative learning control and networked control systems has strong research significance.

In the data-driven iterative learning control, the controlled objects and control tasks need to meet certain strict repeatability assumptions. The strict repeatability means that the initial state is consistent with the expected initial state in iteration. But in the actual application system, due to the disturbance of uncertain factors, this harsh condition is difficult to be met. For example, the error of the sensor or positioning device will cause the deviation of the initial state. The noise may also affect the dynamic performance of the system. This limits the application for the iterative learning control in actual engineering systems. Therefore, the non-strictly repetitive systems are more suitable for actual systems [21–23]. It is an important research direction for the data-driven iterative learning control for solving the non-strict repetitive problems.

*e-mail: fxj@bistu.edu.cn

Manuscript submitted 2021-05-22, revised 2022-01-04, initially accepted for publication 2022-02-06, published in June 2022.

The initial states for the iterative learning can be divided into the following aspects. First, the initial state is exactly equal to the expected initial state, that is, the initial states are strictly repeated. Second, although the initial state is not equal to the expected initial state, the initial states of the system are fixed at the same value in iteration. Third, the initial state in iteration is an arbitrary value, and the arbitrary value changes within a certain range of the ideal value. Fourth, the initial state of the system during iteration is arbitrary and does not satisfy the change within a certain range. The fourth situation is an ideal situation, and it is also the most realistic in actual system applications.

Many scholars have studied the initial state of iterative learning [24, 25, 27–29]. In [24], for a class of nonlinear systems running repeatedly in the study of the convergence of iterative learning control, a PID fuzzy iterative learning control algorithm has been proposed in the arbitrary initial state. In [25], the effect of initial state error in the ILC system is studied. The robustness is investigated against the initial state error of the generalized ILC algorithm. In [26], a fractional-order ILC framework with initial state learning for the tracking problems of linear time-varying systems was presented. [27] presents a P-type ILC scheme with initial state learning for a class of α fractional-order nonlinear system. In [28], the paper presents a second order P-type ILC scheme with initial state learning for a class of fractional order linear distributed parameter systems. In [29], the study addresses a robust ILC scheme for non-linear discrete-time systems in which both the trail lengths and the initial state shifts could be randomly variant in iteration domain.

In the research of iterative learning control algorithm, it is a very worthy research direction to reduce the influence for initial state disturbance on the performance of control system. It has also important practical significance for the application of iterative learning control. In addition, the factors that affect the stability of the system are diverse in actual engineering. Among them, external disturbances and actuator failure are common factors. At present, although some scholars have studied the initial state disturbance problem for iterative learning control in some literature, most of the results obtained are based on the assumption that the system is not affected by external disturbances or failures, which is not general. The main contributions of this paper are as follows. (1) The initial state problem considered in this paper is that there are disturbances and actuator failures in the system. For the linear discrete systems and the nonlinear discrete systems, the corresponding iterative learning control algorithms are respectively proposed, the robust convergence analysis is performed, and the convergence conditions are given. (2) A new iterative learning control algorithm with the initial state correction item is proposed. In the linear discrete system, in order to reduce the influence of the initial state deviation in iteration, the impulse function is introduced into iterative learning control algorithm. It is proved that under the action of the algorithm controller, the system output can gradually converge to the desired trajectory. (3) In the nonlinear discrete systems, when there is data packet loss and the initial state disturbance in NCSs, the problem of the tracking desired trajectory for the system output is studied. In the data packet loss network environment, Bernoulli sequence is used to describe the data loss. The robust convergence conditions are given. As the number of iterative learning increases, the system output can gradually converge to the desired trajectory when the system has actuator failure, initial state deviation and input disturbance. Finally, for the linear systems and the nonlinear systems, simulation studies are done based on the proposed control strategy. The simulation results show that the two control strategies can reduce the influence of the initial state deviation on the tracking effect for the system, which verifies the effectiveness of the proposed method.

A simplified block diagram of the considered NCSs is shown in Fig. 1.

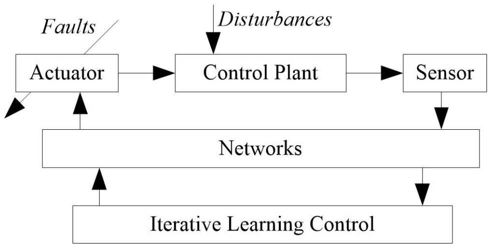

Fig. 1. A block diagram of the considered NCSs

2. ITERATIVE LEARNING FAULT-TOLERANT CONTROL FOR LINEAR DISCRETE NCSS

2.1. Iterative learning controller design

Considering the linear discrete NCSs as follows:

$$\begin{cases} x_k(t+1) = Gx_k(t) + Fu_k(t) + d_k(t), \\ y_k(t) = Hx_k(t), \end{cases} \tag{1}$$

where, $u_{k+1}(t)$ is the control input of the current iteration. $u_k(t)$ is the control input of the previous iteration. $d_k(t)$ is the input disturbance, and the change of two adjacent disturbance is bounded, and its upper bound is the constant b_d. k indicates the number of iterations. t is the time. $x_k(t)$ is the state. $y_k(t)$ is the output. G, F and H are matrices of appropriate dimensions.

Considering the actuator failure model

$$u_{k+1}^M(t) = M_a u_{k+1}(t), \tag{2}$$

where the matrix M_a is the actuator failure, $M_a = \operatorname{diag}(m_{a1}, m_{a2}, \cdots m_{an})$, and $0 \le m_{ai} \le 1$, $(i = 1, \cdots, n)$. $m_{ai} = 0$ indicates complete failure of the i-th actuator. $0 < m_{ai} < 1$ is the partial failure of the i-th actuator. When $m_{ai} = 1$ means the i-th actuator is working normally.

Then the actuator failure system is

$$\begin{cases} x_k(t+1) = Gx_k(t) + Fu_k^M(t) + d_k(t), \\ y_k(t) = Hx_k(t). \end{cases} \tag{3}$$

For system (1), the traditional D-type iterative learning control law is

$$u_{k+1}(t) = u_k(t) + P(t)\left(e_{k+1}(t) - e_k(t)\right), \tag{4}$$

where, $P(t)$ is the learning gain matrix. $e_k(t) = y_d(t) - y_k(t)$ is the system output error.

The iterative learning control law has good convergence for the system whose initial state is the expected value. However, for the arbitrary initial state in iterative learning control, the convergence speed of this control algorithm is not good. Therefore, it is necessary to consider reducing the impact of initial state disturbance during iteration, so that the system can achieve rapid convergence under any initial state conditions.

Because the system is discrete, considering that the system can eliminate the interference of different initial state values to the system at time $t = 0$, an impulse function $r(t)$ is introduced, which satisfies

$$r(t) = \begin{cases} 1 & t = 0, \\ 0 & t = 1, 2, \ldots, T. \end{cases} \tag{5}$$

Combining the excellent characteristics of the traditional D-type iterative control law, in order to make the system output quickly converge to the desired output under any initial state conditions, an iterative learning control algorithm that can reduce the influence of different initial state disturbances is proposed.

$$u_{k+1}(t) = u_k(t) + P(t)\left(e_{k+1}(t) - e_k(t)\right) + r(t)\theta_k\,, \tag{6}$$

where $r(t)\theta_k$ is the initial state correction item. θ_k is a term related to the initial value and initial error of the iteration, and θ_k can be taken as

$$\theta_k = F^{\mathrm{T}}\left(FF^{\mathrm{T}}\right)^{-1} G(x_k(0) - x_{k+1}(0)) + P(t)e_k(0). \tag{7}$$

2.2. Convergence analysis

Theorem 1. For the system (1) and iterative learning control strategy (6), if FF^{T} is invertible, and

$$\|I - HFM_aP(t)\| < 1. \tag{8}$$

Then when $k \to \infty$, in any initial state, the proposed iterative learning control strategy can make the system converge to the desired output trajectory.

Proof. According to the error, we get

$$\begin{aligned} e_{k+1}(t) - e_k(t) &= y_d(t) - y_{k+1}(t) - (y_d(t) - y_k(t)) \\ &= y_k(t) - y_{k+1}(t) = H(x_k(t) - x_{k+1}(t)). \end{aligned} \tag{9}$$

because

$$\begin{aligned} & x_k(t) = G^t x_k(0) + \sum_{s=0}^{t-1} G^{t-s-1}(Fu_k^M(s) + d_k(s)), \\ & x_k(t+1) - x_k(t) = G^t x_{k+1}(0) \\ & \quad + \sum_{s=0}^{t-1} G^{t-s-1}(Fu_{k+1}^M(s) + d_{k+1}(s)) - G^t x_k(0) \\ & \quad - \sum_{s=0}^{t-1} G^{t-s-1}(Fu_k^M(s) + d_k(s)) \\ & \quad = G^t(x_{k+1}(0) - x_k(0)) \\ & \quad + \sum_{s=0}^{t-1} G^{t-s-1}\left[F(u_{k+1}^M(s) - u_k^M(s)) + d_{k+1}(s) - d_k(s)\right]. \end{aligned} \tag{10}$$

Substituting (6) into (10), and because of $u_{k+1}^M = M_a u_{k+1}$, we can get

$$\begin{aligned} x_k(t+1) - x_k(t) &= G^t(x_{k+1}(0) - x_k(0)) \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}\big[FM_aPe_k(s+1) \\ &- FM_a(Pe_k(s) + r(t)\theta_k)\big] \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)). \end{aligned} \tag{11}$$

From the equation (5), $t = \{1, 2, \ldots, T\}$, $r(t) = 0$. So $t \geq 1$, (11) can be simplified to

$$\begin{aligned} x_k(t+1) - x_k(t) &= G^t(x_{k+1}(0) - x_k(0)) \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}FM_aPe_k(s+1) - \sum_{s=0}^{t-1} G^{t-s-1}FM_aPe_k(s) \\ &+ G^{t-1}FM_a\theta_k + \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)). \end{aligned} \tag{12}$$

Substituting (7) into (12):

$$\begin{aligned} x_k(t+1) - x_k(t) &= G^t(x_{k+1}(0) - x_k(0)) \\ &+ \sum_{s=1}^{t} G^{t-s}FM_aPe_k(s) - \sum_{s=0}^{t-1} G^{t-s-1}FM_aPe_k(s) \\ &+ G^{t-1}FM_a\left[F^T(FF^{\mathrm{T}})^{-1}G(x_k(0) - x_{k+1}(0)) + Pe_k(0)\right] \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)), \end{aligned} \tag{13}$$

and simplify the equation (13) we can get:

$$\begin{aligned} x_k(t+1) - x_k(t) &= \sum_{s=1}^{t} G^{t-s}FM_aPe_k(s) \\ &- \sum_{s=0}^{t-1} G^{t-s-1}FM_aPe_k(s) + G^{t-1}FM_aPe_k(0) \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)) \\ &= \sum_{s=1}^{t-1} G^{t-s}FM_aPe_k(s) + FM_aPe_k(t) \\ &- \sum_{s=1}^{t-1} G^{t-s-1}FM_aPe_k(s) \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)) \\ &= \sum_{s=1}^{t-1} G^{t-s-1}(G-I)FM_aPe_k(s) + FM_aPe_k(t) \\ &+ \sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s) - d_k(s)). \end{aligned} \tag{14}$$

Then

$$e_{k+1}(t) = e_k(t) - H\sum_{s=1}^{t-1} G^{t-s-1}(G-I)FM_aPe_k(s) - HFM_aPe_k(t) - H\sum_{s=0}^{t-1} G^{t-s-1}(d_{k+1}(s)-d_k(s)). \tag{15}$$

Simplify the equation (15):

$$e_{k+1}(t) = (I - HFM_aP)e_k(t) - \sum_{s=1}^{t-1} HG^{t-s-1}(G-I)FM_aPe_k(s) - \sum_{s=0}^{t-1} HG^{t-s-1}(d_{k+1}(s)-d_k(s)). \tag{16}$$

Both ends of the equation (16) are normed:

$$\|e_{k+1}(t)\| \le \|I - HFM_aP\|\,\|e_k(t)\| + hb_mfb_p\sum_{s=1}^{t-1} G^{t-s-1}(G-I)\,\|e_k(s)\| + h\sum_{s=0}^{t-1} G^{t-s-1}\,\|d_{k+1}(s)-d_k(s)\|, \tag{17}$$

where $h = \|H\|$, $f = \|F\|$, $b_p = \|P\|$, $b_m = \|M_a\|$.

Taking the mathematical expectation on both ends of the equation (17), we have

$$\varepsilon\{\|e_{k+1}(t)\|\} \le \|I - HFM_aP\|\,\varepsilon\{\|e_k(t)\|\} + hb_mfb_p\sum_{s=1}^{t-1} G^{t-s-1}(G-I)\varepsilon\{\|e_k(s)\|\} + \eta_{k+1}(t), \tag{18}$$

where $\eta_{k+1}(t) = \sum_{s=0}^{t-1} G^{t-s-1}\,\|d_{k+1}(s)-d_k(s)\|$.

Let $L_1 = \|I - HFM_aP\|$, then

$$\varepsilon\{\|e_{k+1}(t)\|\} \le L_1\varepsilon\{\|e_k(t)\|\} + hb_mfb_p\sum_{s=1}^{t-1} G^{t-s-1}(G-I)\varepsilon\{\|e_k(s)\|\} + \eta_{k+1}(t). \tag{19}$$

Expanding the equation (18) from $t = 0$ to $t = T$, we can get

$$\begin{aligned}
&\varepsilon\{\|e_{k+1}(0)\|\} \le L_1\varepsilon\{\|e_k(0)\|\} + \eta_{k+1}(0),\\
&\varepsilon\{\|e_{k+1}(1)\|\} \le L_1\varepsilon\{\|e_k(1)\|\} + hb_mfb_pG\varepsilon\{\|e_k(1)\|\} + \eta_{k+1}(1),\\
&\qquad\vdots\\
&\varepsilon\{\|e_{k+1}(t)\|\} \le L_1\varepsilon\{\|e_k(t)\|\} + hb_mfb_p\sum_{s=0}^{t} G^{t-s}\varepsilon\{\|e_k(s)\|\} + \eta_{k+1}(t).
\end{aligned}$$

The above equation can be expressed as the following matrix:

$$\varepsilon_{k+1} \le D_k\varepsilon_k + N_{k+1}, \tag{20}$$

where

$$\varepsilon_{k+1} = \begin{bmatrix} \varepsilon\{\|e_{k+1}(0)\|\} \\ \varepsilon\{\|e_{k+1}(1)\|\} \\ \vdots \\ \varepsilon\{\|e_{k+1}(t)\|\} \end{bmatrix}, \qquad \varepsilon_k = \begin{bmatrix} \varepsilon\{\|e_k(0)\|\} \\ \varepsilon\{\|e_k(1)\|\} \\ \vdots \\ \varepsilon\{\|e_k(t)\|\} \end{bmatrix},$$

$$D_k = \begin{bmatrix} L_1 & 0 & 0 & 0 \\ Ghb_mfb_p & L_1 & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ G^thb_mfb_p & \cdots & Ghb_mfb_p & L_1 \end{bmatrix},$$

$$N_{k+1} = \begin{bmatrix} \eta_{k+1}(0) \\ \eta_{k+1}(1) \\ \vdots \\ \eta_{k+1}(t) \end{bmatrix}.$$

With the number of iterations increasing $k \to \infty$, $\lim_{k\to\infty}\|d_{k+1}(t)-d_k(t)\| = 0$. Therefore, $\lim_{k\to\infty} N_{k+1} = 0$. For matrix D_k, the eigenvalue at the k-th iteration is L_1. If $\|I - HFM_aP\| < 1$ holds for all k and t, then $\lim_{k\to\infty}\varepsilon\{e_k(t)\} = 0$ holds when $t \in [0, T]$, that is, $\lim_{1\le t\le T} y_k(t) = y_d(t)$.

The Theorem 1 is proved. □

Remark 1. From the above analysis, it can be seen that the system tracking error is not affected by the initial state value in $t \in [1, T]$. When the number of iterations gradually increases, $L_1 < 1$ is satisfied, the system tracking error gradually converges to zero.

Remark 2. In Theorem 1, during iteration, the error generated by any initial state value can be eliminated at time $t = 0$, so the system output will be not affected at a later time.

Remark 3. In Theorem 1, under any initial state value conditions, after iterative learning, the system output trajectory can converge to the desired trajectory within a limited time.

Although the impulse function $r(t)$ introduced in this section can better reduce the influence for the initial value deviation in the iteration on the system performance, it is not suitable to apply this method to deal with the influence for the nonlinear system. Next, for the nonlinear discrete system, an iterative learning fault-tolerant control strategy with Bernoulli random variables is proposed. When the initial state has a deviation, the system convergence is discussed.

3. ITERATIVE LEARNING FAULT-TOLERANT CONTROL FOR NONLINEAR NCSS

3.1. Iterative learning controller design

Considering the following nonlinear discrete NCSs:

$$\begin{cases} x_k(t+1) = g(x_k(t),t) + f(x_k(t),t)u_k(t) + d_k(t), \\ y_k(t) = Bx_k(t), \end{cases} \tag{21}$$

where $t \in \{0, 1, \cdots, T\}$ is the discrete time. $x_k(t) \in R^m$ is the state. $u_k(t) \in R^n$ is the control input. $y_k(t) \in R^n$ is the output. $d_k(t)$ is the input disturbance and satisfies $\sup_{t\in[0,T]} \|d_k(t)\| \leq b_1$. $g(\cdot)$ and $f(\cdot)$ are nonlinear functions. The subscript k is the number of iterations. B is a matrix of appropriate dimensions.

Then the actuator failure system is

$$\begin{cases} x_k(t+1) = g(x_k(t),t) + f(x_k(t),t)M_a u_k(t) + d_k(t), \\ y_k(t) = Bx_k(t), \end{cases} \tag{22}$$

where M_a is the actuator fault matrix, and its definition is the same as the equation (2).

For the system (22), the control goal is to design an iterative learning fault-tolerant controller so that the control system output $y_k(t)$ converges to the desired output trajectory $y_d(t)$, that is

$$\sup_{0\leq t\leq T} |y_d(t) - y_k(t)| < \varepsilon, \tag{23}$$

where ε is a small positive number.

The output tracking error $e_k(t)$ is defined as follows:

$$e_k(t) = y_d(t) - y_k(t). \tag{24}$$

Due to network constraints, there are often two cases of data packet loss in NCSs. One is the loss of the control input signal between the controller and the actuator, and the other is the loss of the measurement signal between the controller and the sensor. In this paper, the control input signal loss is only considered in NCSs. When there is data packet loss, the Bernoulli random matrix $\gamma(t) = \mathrm{diag}[\alpha_1(t), \alpha_2(t), \cdots, \alpha_n(t)]$ is used to represent the successful probability of the data transmission, where $\alpha_i(t) (i = 1, 2, \cdots, n)$ is an independently distributed Bernoulli random variable. When $\alpha_i(t) = 0$, it means that the data transmission fails, that is, the output data at time t is lost. When $\alpha_i(t) = 1$, the output signal at time t is transmitted successfully. $\alpha_i(t)$ is a Bernoulli random variable that is independent and equally distributed, and satisfies

$$\mathrm{prob}\,\{\alpha_i(t) = 1\} = \varepsilon\,\{\alpha_i(t)\} = \overline{\alpha}, \qquad 0 \leq \overline{\alpha} \leq 1. \tag{25}$$

In the equation (25), prob$\{\cdot\}$ is the probability. $\varepsilon\{\cdot\}$ is mathematical expectation. $0 \leq \overline{\alpha} \leq 1$ represents the successful probability of the channel data transmission. Therefore, combined with the traditional iterative learning control law, the following iterative learning control algorithm is proposed:

$$u_{k+1}(t) = u_k(t) + \overline{\alpha}P(t)e_k(t). \tag{26}$$

For the nonlinear NCSs (22) with actuator failure, the control strategy (26) will be designed in the event of data packet loss. The following convergence is analyzed when there is the initial state disturbance.

3.2. Convergence analysis

In the iterative learning algorithms, it is often required that the initial state and the expected initial state are consistent in the iterative learning process. However, in the actual system, the initial state value of the system and the ideal value are often not equal, that is, the strict repeatability is not satisfied. Below, when there is a deviation between the initial state and the ideal initial state, that is, when the initial state conditions are not strictly limited, the sufficient condition for the convergence of the iterative learning algorithm is given.

Theorem 2. For the nonlinear system (22) with actuator failure, the initial state value satisfies $x_k(0) = x^0$ and $x^0 \neq x_d(0)$. If the iterative learning controller (26) is used and satisfies

$$\|I - Bf_k(t)M_a\overline{\alpha}P(t)\| < 1. \tag{27}$$

Then as the number of iterations increases, the system output gradually converges to the desired trajectory. That is, when $k \to \infty$, the system output $y_k(t)$ and the desired trajectory $y_d(t)$ meet $\lim\limits_{k\to\infty} \varepsilon\,\{y_k(t)\} = y_d(t)$.

Proof. The tracking error at the $(k+1)$-th iteration

$$\begin{aligned} e_{k+1}(t+1) &= y_d(t+1) - y_{k+1}(t+1) \\ &= y_d(t+1) - y_k(t+1) - (y_{k+1}(t+1) - y_k(t+1)) \\ &= e_k(t+1) - B(x_{k+1}(t+1) - x_k(t+1)). \end{aligned} \tag{28}$$

Simplifying the above equation (28), we can get:

$$\begin{aligned} e_{k+1}(t+1) = e_k(t+1) - B[&g_{k+1}(t) + f_{k+1}(t)M_a u_{k+1}(t) \\ &+ d_{k+1}(t) - g_k(t) - f_k(t)M_a u_k(t) + d_k(t)]. \end{aligned} \tag{29}$$

That is

$$\begin{aligned} e_{k+1}(t+1) = e_k(t+1) - B[&g_{k+1}(t) \\ -\ &g_k(t) + (f_{k+1}(t) - f_k(t))M_a u_{k+1}(t) \\ +\ &f_k(t)M_a(u_{k+1}(t) - u_k(t)) + d_{k+1}(t) - d_k(t)]. \end{aligned} \tag{30}$$

The two ends of the equation (30) are normed and the Lipschintz condition is used, then there are the constants ξ_g and ξ_f that make the following equation holds

$$\begin{aligned} \|e_{k+1}(t+1)\| \leq\ & \|I - Bf_k(t)M_a\overline{\alpha}P(t)\|\,\|e_k(t+1)\| \\ &+ \|B\| \Big[\left(\xi_g + \xi_f b_m \|u_{k+1}(t)\|\right) \|x_{k+1}(t) - x_k(t)\| \\ &+ \|d_{k+1}(t) - d_k(t)\| \Big]. \end{aligned} \tag{31}$$

From the above equation (31):

$$\begin{aligned}\|e_{k+1}(t+1)\| \leq{}& \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|\,\|e_k(t+1)\| \\ &+(a_2\xi_g+a_2\xi_f b_2 b_m)\,\|x_{k+1}(t)-x_k(t)\| \\ &+a_2\,\|d_{k+1}(t)-d_k(t)\|,\end{aligned} \tag{32}$$

where $a_2 = \|B\|$.

Next, $\|x_{k+1}(t)-x_k(t)\|$ is expressed as

$$\begin{aligned}&\|x_{k+1}(t+1)-x_k(t+1)\| \\ &\quad= \|g_{k+1}(t)+f_{k+1}(t)M_a u_{k+1}(t)+d_{k+1}(t) \\ &\qquad - g_k(t)-f_k(t)M_a u_k(t)-d_k(t)\| \\ &\quad= \|g_{k+1}(t)-g_k(t)+(f_{k+1}(t)-f_k(t))M_a u_{k+1}(t) \\ &\qquad + f_k(t)\overline{\alpha}P(t)e_k(t+1)+d_{k+1}(t)-d_k(t)\| \\ &\quad\leq (\xi_g+\xi_f b_2 b_m)\,\|x_{k+1}(t)-x_k(t)\|+\|\gamma(t)\|\,\|f_k(t)\| \\ &\qquad \times \|P(t)\|\,\|e_k(t+1)\|+\|d_{k+1}(t)-d_k(t)\|.\end{aligned} \tag{33}$$

Sorting out the above equation (33), we have

$$\begin{aligned}\|x_{k+1}(t+1)-x_k(t+1)\| \leq{}& c_1\,\|x_{k+1}(t)-x_k(t)\| \\ &+a_1 b_p b_r\,\|e_k(t+1)\|+\|d_{k+1}(t)-d_k(t)\|,\end{aligned} \tag{34}$$

where

$c_1 = \xi_g+\xi_f b_2 b_m$, $b_p = \sup\limits_{t\in[0,T-1]}\|P(t)\|$, $b_r = \sup\limits_{t\in[0,T-1]}\|\gamma(t)\|$.

From the above equation (34), we can get

$$\begin{aligned}\|x_{k+1}(t)-x_k(t)\| \leq{}& c_1^t\,\|x_{k+1}(0)-x_k(0)\| \\ &+\sum_{q=0}^{t-1} c_1^{t-q-1}\big(a_1 b_p b_r\,\|e_k(q+1)\| \\ &+\|d_{k+1}(q)-d_k(q)\|\big).\end{aligned} \tag{35}$$

Substituting (33) into (30):

$$\begin{aligned}&\|e_{k+1}(t+1)\| \leq \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|\,\|e_k(t+1)\| \\ &\qquad + a_2 c_1\,\|x_{k+1}(t)-x_k(t)\|+a_2\,\|d_{k+1}(t)-d_k(t)\| \\ &\quad\leq \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|\,\|e_k(t+1)\| \\ &\qquad + a_2 c_1[c_1^t\,\|x_{k+1}(0)-x_k(0)\| \\ &\qquad + \sum_{q=0}^{t-1} c_1^{t-q-1}(a_1 b_p b_r\,\|e_k(q+1)\| \\ &\qquad + \|d_{k+1}(t)-d_k(t)\|)]+a_2\,\|d_{k+1}(t)-d_k(t)\|.\end{aligned} \tag{36}$$

From the above equation (36), then

$$\begin{aligned}&\|e_{k+1}(t+1)\| \leq \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|\,\|e_k(t+1)\| \\ &\quad + a_2 c_1^{t+1}\,\|x_{k+1}(0)-x_k(0)\| \\ &\quad + \sum_{q=0}^{t-1} c_1^{t-q-1}(a_2 b_p b_r\,\|e_k(q+1)\| \\ &\quad + \sum_{q=0}^{t-1} c_1^{t-q-1}\,\|d_{k+1}(q)-d_k(q)\|+a_2\,\|d_{k+1}(t)-d_k(t)\|.\end{aligned} \tag{37}$$

Taking expectations on both sides of equation (37), we have

$$\begin{aligned}\varepsilon\{\|e_{k+1}(t+1)\|\} \leq{}& \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|\,\varepsilon\{\|e_k(t+1)\|\} \\ &+\sum_{q=0}^{t-1} c_1^{t-q} a_2 a_4 b_p b_r \varepsilon\{\|e_k(q+1)\|\}+\Gamma_{k+1}(t),\end{aligned} \tag{38}$$

where

$$\begin{aligned}\Gamma_{k+1}(t) ={}& a_2 c_1^{t+1}\,\|x_{k+1}(0)-x_k(0)\| \\ &+\sum_{q=0}^{t-1} c_1^{t-q} a_4\,\|d_{k+1}(q)-d_k(q)\|+a_2\,\|d_{k+1}(t)-d_k(t)\|, \\ a_4 ={}& c_1^{-1}.\end{aligned}$$

Let $L = \|I-Bf_k(t)M_a\overline{\alpha}P(t)\|$. Then the equation (38) can be written as

$$\begin{aligned}\varepsilon\{\|e_{k+1}(t+1)\|\} \leq{}& L\varepsilon\{\|e_k(t+1)\|\} \\ &+\sum_{q=0}^{t-1} c_1^{t-q} a_2 a_4 b_p b_r \varepsilon\{\|e_k(q+1)\|\}+\Gamma_{k+1}(t).\end{aligned} \tag{39}$$

Expanding the equation (39) from $t = 0$ to $t = T$, we have

$$\begin{aligned}&\varepsilon\{\|e_{k+1}(1)\|\} \leq L\varepsilon\{\|e_k(1)\|\}+\Gamma_{k+1}(0), \\ &\varepsilon\{\|e_{k+1}(2)\|\} \leq L\varepsilon\{\|e_k(2)\|\} \\ &\qquad + c_1 a_2 a_4 b_p b_r e_k(1)+\Gamma_{k+1}(1) \\ &\qquad \vdots \\ &\varepsilon\{\|e_{k+1}(t)\|\} \leq L\varepsilon\{\|e_k(t)\|\} \\ &\qquad + \sum_{q=0}^{t-1} c_1^{t-q-1} a_2 a_4 b_p b_r \varepsilon\{\|e_k(q+1)\|\}+\Gamma_{k+1}(t-1).\end{aligned}$$

The expanded items can be expressed as the following matrix:

$$\varepsilon_{k+1} \leq C_k \varepsilon_k + M_{k+1}, \tag{40}$$

where

$$\varepsilon_{k+1} = \begin{bmatrix}\varepsilon\{\|e_{k+1}(1)\|\} \\ \varepsilon\{\|e_{k+1}(2)\|\} \\ \vdots \\ \varepsilon\{\|e_{k+1}(t)\|\}\end{bmatrix}, \quad \varepsilon_k = \begin{bmatrix}\varepsilon\{\|e_k(1)\|\} \\ \varepsilon\{\|e_k(2)\|\} \\ \vdots \\ \varepsilon\{\|e_k(t)\|\}\end{bmatrix},$$

$$C_k = \begin{bmatrix} L & 0 & 0 & 0 \\ c_1 a_2 b_p b_r & L & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ c_1^{t-1} a_2 b_p b_r & \cdots & c_1 a_2 b_p b_r & L \end{bmatrix},$$

$$M_{k+1} = \begin{bmatrix} \Gamma_{k+1}(0) \\ \Gamma_{k+1}(1) \\ \vdots \\ \Gamma_{k+1}(t) \end{bmatrix}.$$

With the number of iterations increasing $k \to \infty$, $\lim_{k\to\infty} \|x_{k+1}(0) - x_k(0)\| = 0$ and $\lim_{k\to\infty} \|d_{k+1}(0) - d_k(0)\| = 0$. Therefore, $\lim_{k\to\infty} M_{k+1} = 0$. For the matrix C_k, the eigenvalue at the k-th iteration is L. If $L = \|I - Bf_k(t)M_a\overline{\alpha}P(t)\| < 1$ holds for all and t, then $\lim_{k\to\infty} \varepsilon\{e_k(t)\} = 0$ holds when $t \in [0,\mathrm{T}]$. The proof is completed. □

According to Theorem 2, for a nonlinear system with initial state disturbance, when there is a deviation between the initial state and the ideal state during iterations, the system output can still converge to the expected output.

4. SIMULATION STUDY

4.1. Numerical simulation for the linear NCSS

In order to verify the effectiveness of the proposed control strategy, let us consider the following discrete linear NCSs

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \quad y(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix}, \quad u(t) = \begin{pmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{pmatrix}.$$

The system parameters are selected as

$$G = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}, \quad F = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \quad H = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

The learning gain matrix $P = \begin{pmatrix} 0.2 & 0 \\ 0 & 0.1 \\ 0 & 0 \end{pmatrix}$.

The fault matrix is $M_a = \operatorname{diag}\{0.9,\ 0.9,\ 0.9\}$. The condition $\|I - HFM_aP\| < 1$ is satisfied. The matrix FF^{T} is invertible, $FF^{\mathrm{T}} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$, $(FF^{\mathrm{T}})^{-1} = \begin{pmatrix} \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} \end{pmatrix}$. Suppose the desired tracking trajectory $y_d(t) = \begin{pmatrix} y_{1d}(t) \\ y_{2d}(t) \end{pmatrix} = \begin{pmatrix} 0.01t(1+t) \\ 5\sin(0.2\pi t) \end{pmatrix}$ and the initial control input $u_0(t) = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^{\mathrm{T}}$. Input disturbance $d_k = \begin{bmatrix} 0.2\sin(\pi t/100) & 0.2\cos(\pi t/100) \end{bmatrix}^{\mathrm{T}}$. The initial states value in iteration for the system are generated by the random function rand().

By using the iterative learning control (4) without correction terms, the output results are shown in Fig. 2 and Fig. 3 when the initial state value in iteration is random.

The tracking curve for the iterative output $y_1(t)$ to the desired trajectory $y_{1d}(t)$ is shown in Fig. 2. When $k = 300$, the system output can better track the desired trajectory. The tracking curve for the iterative output $y_2(t)$ to the desired trajectory $y_{2d}(t)$ is shown in Fig. 3. When $k = 300$, there is still a large error between the system output and the expected output, but when $k = 600$, the system output can track the desired trajectory $y_{2d}(t)$.

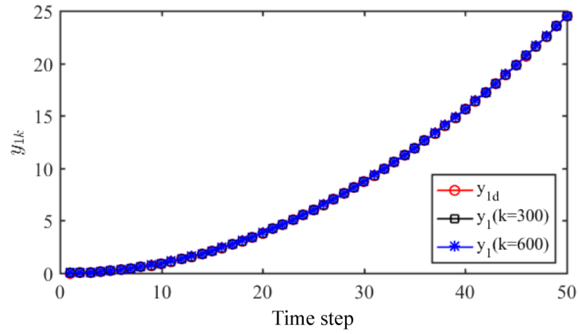

Fig. 2. y_1 tracking curves

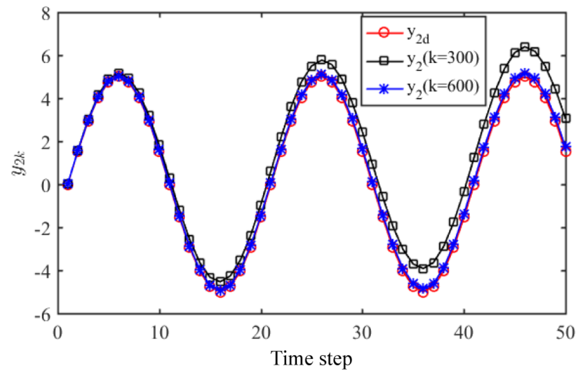

Fig. 3. y_2 tracking curves

When the proposed iterative learning control (6) with correction terms is used, the output results are shown in Fig. 4 and Fig. 5.

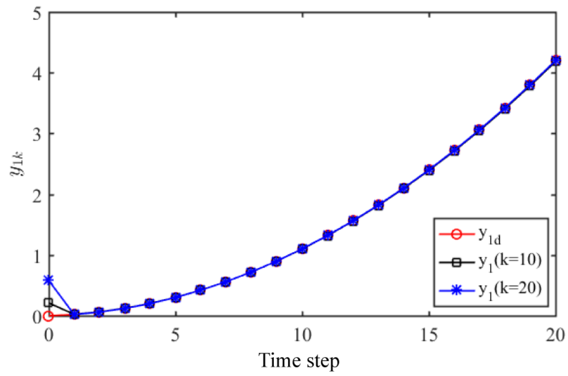

Fig. 4. y_1 tracking curves

The tracking curve of the system iterative output $y_1(t)$ and $y_2(t)$ to the desired trajectories $y_{1d}(t)$ and $y_{2d}(t)$ are shown in Fig. 4 and Fig. 5, respectively. The curves in Fig. 4 are the track-

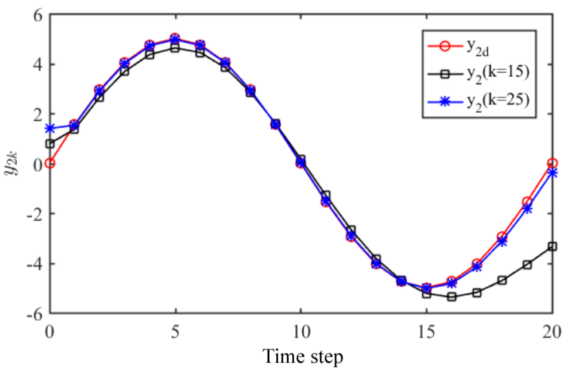

Fig. 5. y_2 tracking curves

ing results at $k = 10$ and $k = 20$. When $k = 10$, the system output can better track the desired trajectory. The tracking curves of $k = 15$ and $k = 25$ are shown in Fig. 5. When $k = 15$, the system output still has a large error, but when $k = 25$, the system output can better track the desired trajectory $y_{2d}(t)$. It can be seen from the simulation results that although the initial state is different in iteration, the system output can quickly converge to the expected output by adopting the proposed iterative learning control.

When the control strategy (6) is applied, the system output error curves in the iteration process are shown in Fig. 6 and Fig. 7. It can be seen from the figures that as the number of iterations increases, the system output error gradually decreases, and finally converges to zero.

For the system with actuator failures, in the presence of input disturbances and initial state deviations, the iterative learning control strategy proposed can make the output converge to the desired trajectory faster. The correction item has a good effect on the controlled system with any initial state value. Compared with the traditional control algorithm (4), the system converges faster when the control algorithm (6) is applied, which proves the effectiveness of the proposed control strategy.

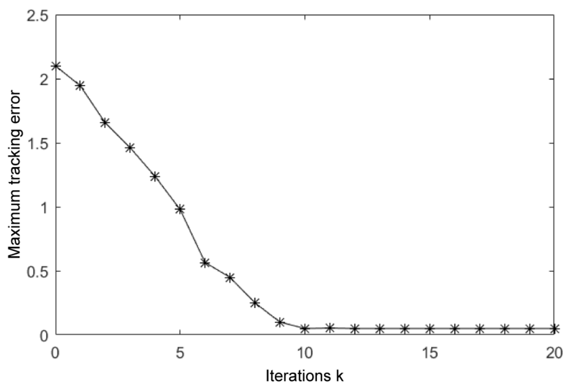

Fig. 6. The error curve of y_1 to desired y_{1d}

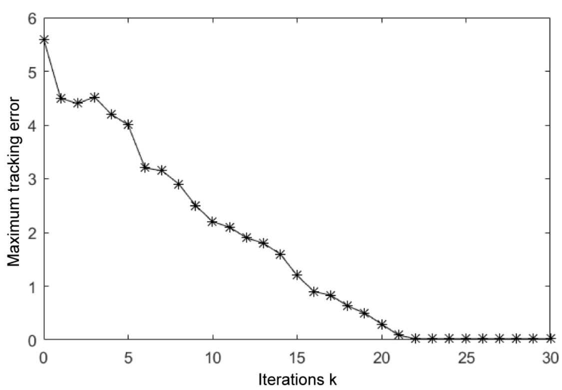

Fig. 7. The error curve of y_2 to desired y_{2d}

4.2. Numerical simulation for the nonlinear NCSS

Considering the following nonlinear discrete NCSs:

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix} = \begin{bmatrix} a_1(x) + a_2(x)u(t) + d(t) \\ x_1(t) \end{bmatrix},$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t),\ x_2(t) \end{bmatrix}^{\mathrm{T}},$$

where

$$a_1(x) = 0.1x_1(t)x_2(t),$$

$$a_2(x) = [1 + 0.1\cos(x_1(t))]^{-1}.$$

The desired tracking trajectory is $y_d(t) = \sin(\pi t/200)$ where $t \in [0, 450]$. The learning gain $P = 0.7$. The failure $Ma = 0.7 + 0.3\sin(\pi t)$. The disturbance is $d_k(t) = 0.2\sin(\pi t/100)$.

The following is to verify the effectiveness of the algorithm by numerical simulation for the data packet loss of 0%, 10%, and 20% respectively.

Under the condition that the initial state value is not equal to the expected initial value, that is, when $x_d(0) = [0, 0]^T$, $xk(0) = [1, 1]^T$, the numerical simulation is studied.

For the nonlinear NCSs with actuator failures, when there is output data packet loss, input disturbance and iteration initial value deviation, after applying the control strategy (24), it can be seen from Fig. 8 to Fig. 10 that the system output can track the expectation trajectory.

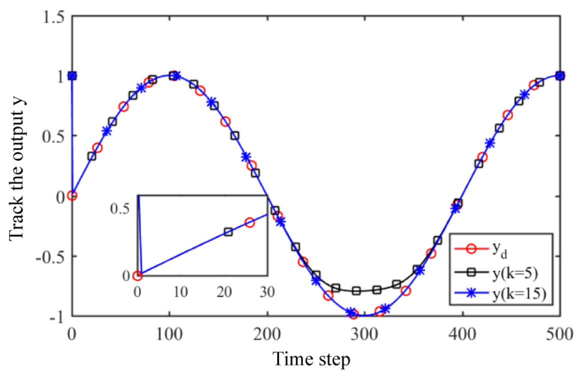

Fig. 8. The output curves without data loss

The output error curves in the system iteration are shown in Figs. 11 to 13. When there is no data packet loss, as the number of iterations increases, the system output gradually approaches

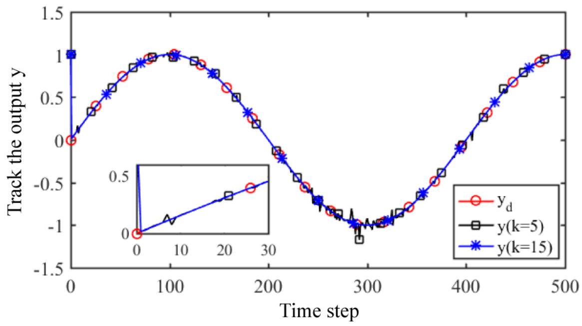

Fig. 9. The output curves with data loss 10%

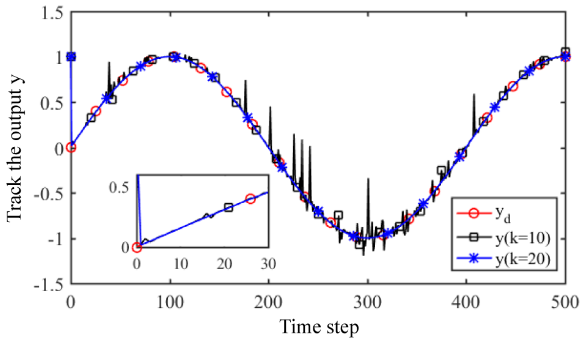

Fig. 10. The output curves with data loss 20%

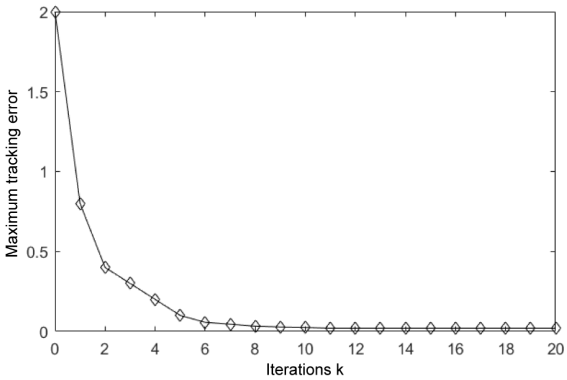

Fig. 11. Tracking error curve without data loss

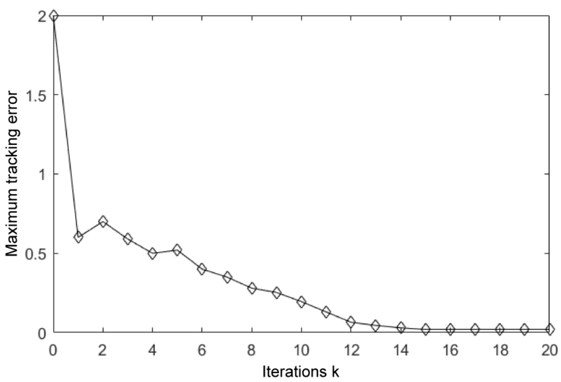

Fig. 12. Tracking error curve with data loss 10%

the expected value. When the number of iterations is $k = 11$, the system output can track the desired trajectory, and the system output error is close to zero. When the data packet loss rate is 10% and the number of iterations is $k = 15$, the system output error converges to zero. When the data packet loss rate is 20% and the number of iterations is $k = 19$, the system output can track the desired trajectory, and the output error can converge to zero. Through simulation, it can be known that as the data loss rate increases, the convergence speed of the tracking error also slows down. However, as the number of iterations increases, the error can still be gradually reduced, and finally it can converge to zero.

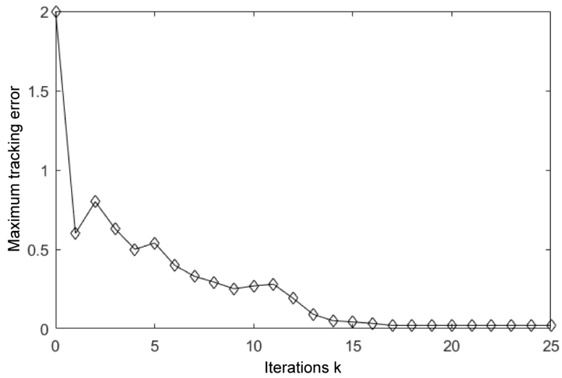

Fig. 13. Tracking error curve with data loss 20%

4.3. Application simulation for the nonlinear NCSS

For the two-degree-of-freedom motion robot considered in this section [30], the front wheels are driven wheels, and the rear wheels are driving wheels. When the motion robot model is constructed, the friction between the tire and the ground is ignored. The motion model is

$$\begin{bmatrix} \dot{x}_a(t) \\ \dot{y}_a(t) \\ \dot{\theta}_a(t) \end{bmatrix} = \begin{bmatrix} \cos\theta_a(t) & 0 \\ \sin\theta_a(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_a(t) \\ \omega_a(t) \end{bmatrix}, \tag{41}$$

where, $x_a(t)$ is the abscissa. $y_a(t)$ is the ordinate. $\theta_a(t)$ is the attitude angle. $v_a(t)$ is the linear velocity. $\omega_a(t)$ is the angular velocity.

The above model (41) is changed as follows:

$$\begin{cases} x(t) = \begin{bmatrix} x_a(t) & y_a(t) & \theta_a(t) \end{bmatrix}^{\mathrm{T}}, \\ u(t) = \begin{bmatrix} v_a(t) & \omega_a(t) \end{bmatrix}^{\mathrm{T}} \end{cases} \tag{42}$$

The motion model can be written as

$$\dot{x}(t) = f(x(t))u(t), \tag{43}$$

where

$$f(x(t)) = \begin{bmatrix} \cos\theta_a(t) & 0 \\ \sin\theta_a(t) & 0 \\ 0 & 1 \end{bmatrix}. \tag{44}$$

Because the movement of the robot system is repetitive, we can get

$$\dot{x}_k(t) = f(x_k(t))u_k(t). \tag{45}$$

Since $f(x_k(t))$ satisfies the Lipschintz condition and is also bounded. The motion model meets the convergence requirements of iterative learning control.

Discretization of the motion model, then

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ \theta(k) \end{bmatrix} + T_s \begin{bmatrix} \cos\theta(k) & 0 \\ \sin\theta(k) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(k) \\ \omega(k) \end{bmatrix}, \tag{46}$$

where T_s is the sampling time.

In this section, the proposed iterative learning control strategy is applied to the control for the two-degree-of-freedom motion robot. The mobile robot is divided into linear motion and curved motion for the simulation research.

1. Linear motion of the robot

The expected linear trajectory equation is selected as follows:

$$\begin{cases} x_d(t) = t, \\ y_d(t) = 2t, \\ \theta_d(t) = \arctan 2. \end{cases} \tag{47}$$

The initial expected value is

$$\begin{bmatrix} x_k(0) & y_k(0) & \theta_k(t) \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & \arctan 2 \end{bmatrix}^{\mathrm{T}}. \tag{48}$$

The initial control input is $u_0(t) = 0$.

When there is no data loss, the linear path tracking simulations for the robot are shown in Figs. 14 to 16. It can be seen from the simulation that as the number of iterations increases, the system output trajectory gradually tracks the desired trajectory.

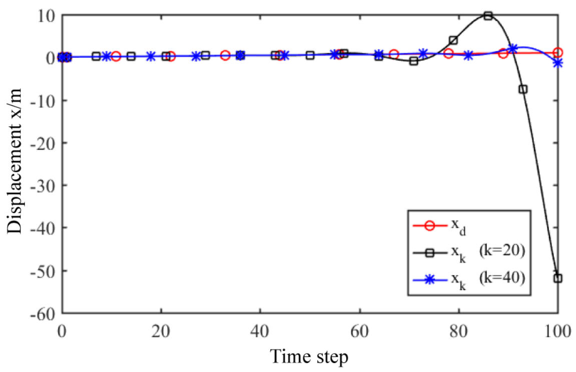

Fig. 14. Tracking curves of x

The root mean square error curve for the motion robot tracking during the iteration process is shown in Fig. 17. When the number of iterations is $k = 40$, the path tracking error tends to zero.

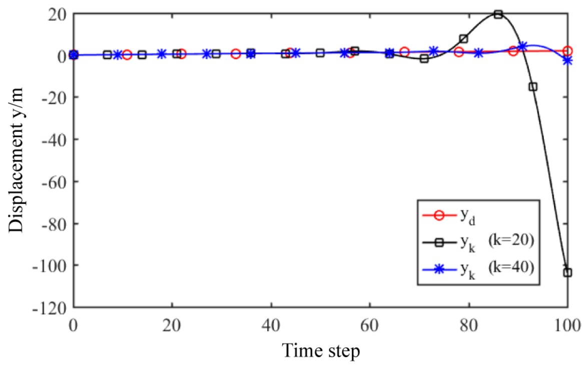

Fig. 15. Tracking curves of y

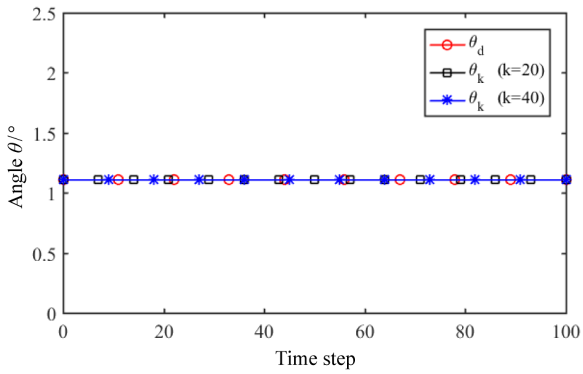

Fig. 16. Tracking curves of θ

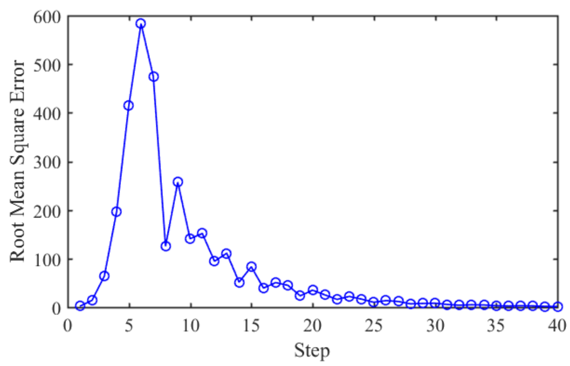

Fig. 17. Root mean square error curve

When the data packet loss rate is 40%, the simulation is shown in Figs. 18 to 20. As the number of iterations increases, the system output trajectory gradually follows the desired trajectory.

The root mean square error curve for the motion robot tracking is shown in Fig. 21. When the number of iterations is $k = 80$, the path tracking error tends to zero. This proves that the control strategy is feasible and effective.

Through the above simulation, it can be known that the data loss has an impact on the system control performance. The higher the data loss rate, the slower the convergence speed of the algorithm. However, under the action of the given controller,

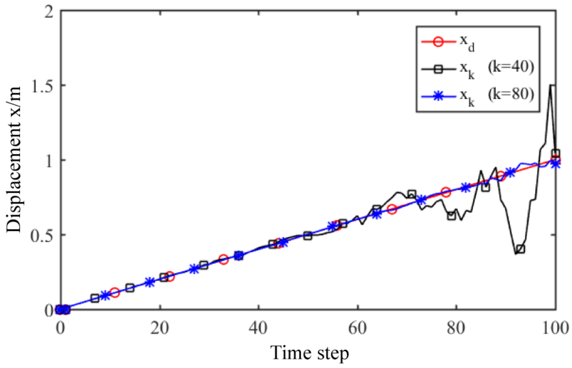

Fig. 18. Tracking curves of x

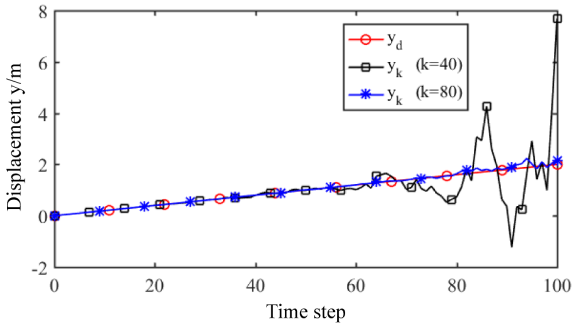

Fig. 19. Tracking curves of y

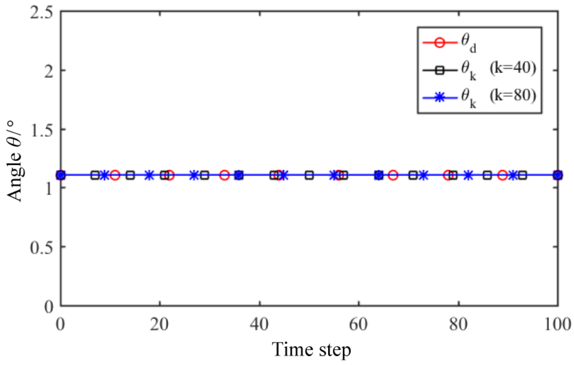

Fig. 20. Tracking curves of θ

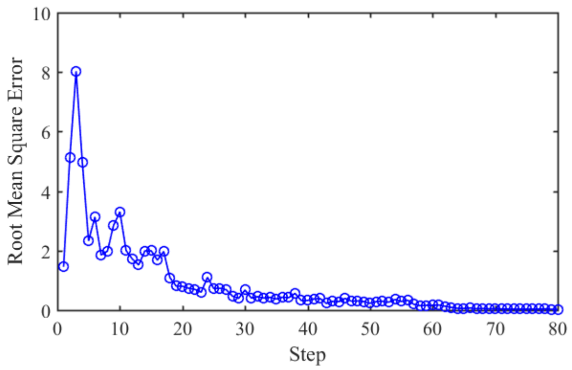

Fig. 21. Root mean square error curve

as the number of iterations increases, trajectory tracking can finally be realized, that is, the motion robot can run according to the desired linear trajectory.

2. Curve motion of the robot
The desired curve trajectory equation is selected as follows:

$$\begin{cases} x_d(t) = 2\cos(\pi t) - 2, \\ y_d(t) = \sin(\pi t), \\ \theta_d(t) = \pi t + \dfrac{\pi}{2}. \end{cases} \tag{49}$$

The initial expectation is

$$\begin{bmatrix} x_k(0) & y_k(0) & \theta_k(t) \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} 0 & 0 & \dfrac{\pi}{2} \end{bmatrix}^{\mathrm{T}}. \tag{50}$$

The initial control input $u_0(t) = 0$.

When the data packet loss rate is 10%, the path tracking curves of the motion robot are shown in Figs. 22 to 24. As the number of iterations increases, the system output trajectory gradually follows the desired trajectory.

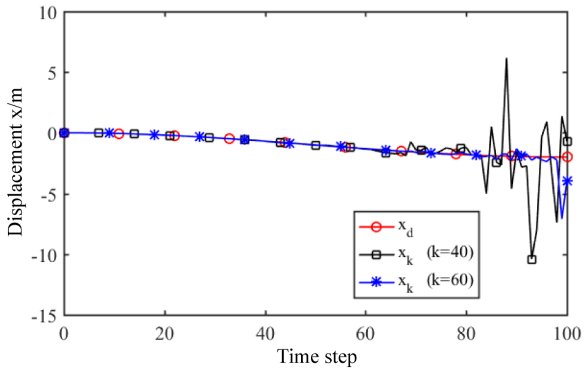

Fig. 22. Tracking curves of x

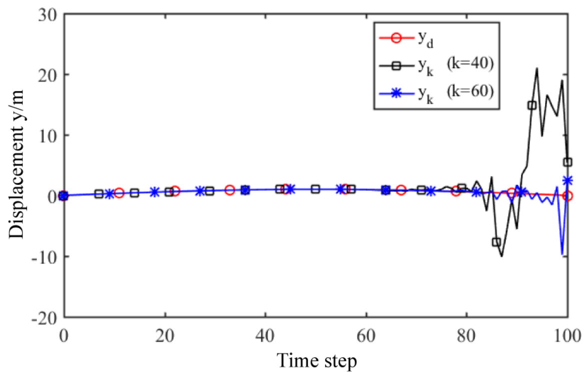

Fig. 23. Tracking curves of y

The root mean square error curve during the iteration for the motion robot tracking is shown in Fig. 25. When the number of iterations is $k = 70$, the path tracking error tends to zero. The control strategy is verified to be effective.

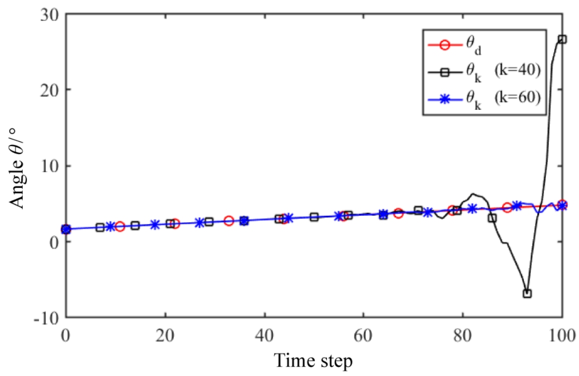

Fig. 24. Tracking curves of θ

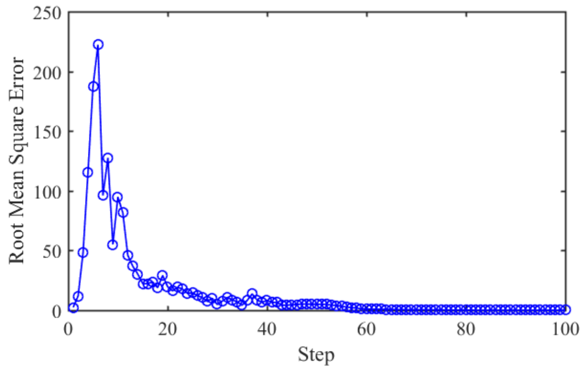

Fig. 25. Root mean square error curve

When the data packet loss rate is 40%, the simulation curves are shown in Figs. 26 to 28. As the number of iterations increases, the system output trajectory gradually follows the desired trajectory.

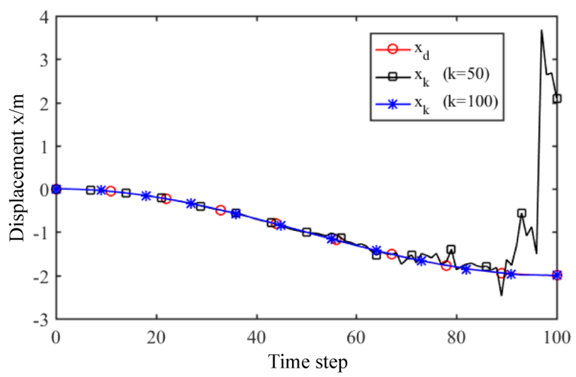

Fig. 26. Tracking curves of x

The root mean square error curve in the iteration for the motion robot tracking is shown in Fig. 29. When the number of iterations is $k = 90$, the path tracking error approaches to zero. It can be seen from the simulation results that under the action of the designed control strategy, as the number of iterations increases, the robot can achieve the task of trajectory tracking, that is, it can follow the desired curve trajectory.

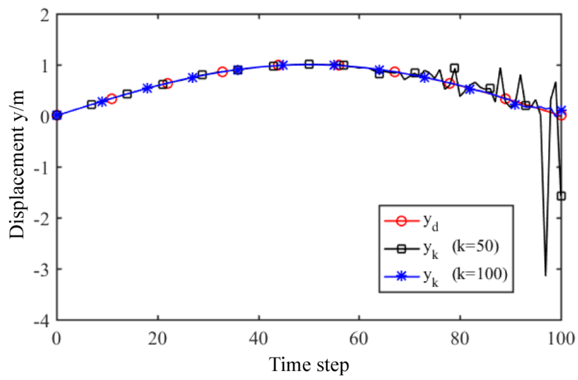

Fig. 27. Tracking curves of y

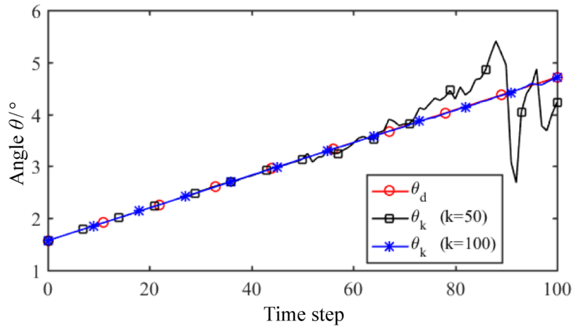

Fig. 28. Tracking curves of θ

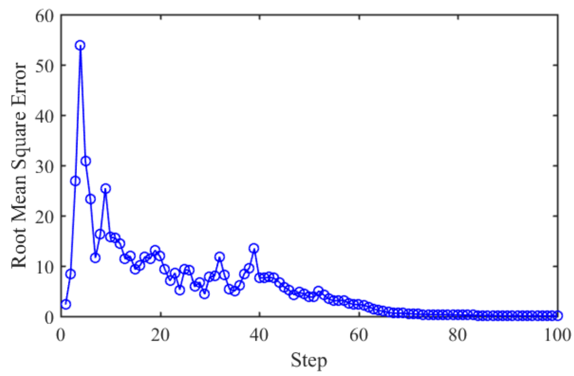

Fig. 29. Root mean square error curve

Based on the above simulation research, in the NCSs, as the data packet loss rate increases, the convergence speed of the tracking error becomes slower. However, as the number of iterations increases, the system tracking error can gradually decrease until it converges to zero.

5. CONCLUSIONS

In this paper, for the NCSs with the external disturbances and actuator failures, the initial state deviation influences for the linear discrete systems and the nonlinear discrete systems are studied respectively. The corresponding iterative learning fault-tolerant control is proposed. Finally, by using the proposed control strategy, the linear system and the nonlinear system are simulated respectively to verify the effectiveness of the proposed method.

The discrete linear network system and discrete nonlinear network system are considered in this paper, and the impact of sample time on the performance of the control system is not studied. In the considered discrete system, only the impact of initial state disturbance, external disturbances, actuator failures, and data packet loss failures on system performance are considered, and the network time-delay problem is not considered. Certainly, in NCSs, the sample time and delays will definitely affect system performance and need to be studied next.

ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China under Grant 61973041.

REFERENCES

[1] G. Saeid, S.A. Akbar, and N.S. Vahid, "Event-triggered robust model predictive control for Lipschitz nonlinear networked control systems subject to communication delays," *Trans. Inst. Meas. Control*, vol. 43, no. 5, pp. 1126–1142, 2021, doi: 10.1177/0142331220969058.

[2] M. M. Azimi, A. A. Afzalian and R, Ghaderi, "Decentralized stabilization of a class of large scale networked control systems based on modified event-triggered scheme," *Int. J. Dyn. Control*, vol. 9, no. 1, pp. 149–159, 2021, doi: 10.1007/s40435-020-00649-4.

[3] T. Zhongda. "Networked control system time-delay compensation based on PI-based dynamic matrix control," *At-Automatisierungstechnik*, vol. 69, no. 1, pp. 41–51, 2021, doi: 10.1515/auto-2020-0020.

[4] L. Yicai and L. Bin, "Research on the Control of Networked Switched Systems with Delay and Packet Dropout," *Control Eng. China*, vol. 25, no. 8, pp. 1482–1489, 2018, doi: 10.14107/j.cnki.kzgc.160777.

[5] Y. Li, B. Zhang, and X. Xu, "Robust control for permanent magnet in-wheel motor in electric vehicles using adaptive fuzzy neural network with inverse system decoupling," *Trans. Can. Soc. Mech. Eng.*, vol. 42, no. 3, pp. 286–297, 2018, doi: 10.1139/tcsme-2018-0027.

[6] A. Gunathillake, H. Huang, and A.V. Savkin, "Sensor-Network-Based navigation of a mobile robot for extremum seeking using a topology map," *IEEE Trans. Ind. Inf.*, vol. 15, no. 7, pp. 3962–3972, 2019, doi: 10.1109/TII.2019.2893345.

[7] H. Zhang *et al.*, "New Results on Stability and Stabilization of Networked Control Systems with Short Time-Varying Delay," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2772–2781, 2017, doi: 10.1109/TCYB.2015. 2489563.

[8] R.A. Kumar and K. Srinivasan, "State estimation for a networked control system with packet delay, packet dropouts and uncertain observation in S-E and C-A channels," *Optim. Control. Appl. Methods*, vol. 41, no. 6, pp. 2094–2114, 2020, doi: 10.1002/ oca.2614.

[9] B. Koo, W. Kwon, and S. Lee, "Integral-based event-triggered PD control for systems with network-induced delay using a quadratic generalised free-weighting matrix inequality," *IET Control Theory Appl.*, vol. 11, no. 18, pp. 3261–3268, 2017, doi: 10.1049/iet-cta.2017.0473.

[10] S. Yuanbo *et al.*, "Robust Mixed H_2/H_∞Control for An Uncertain Wireless Sensor Network Systems with Time Delay and Packet Loss," *Int. J. Control Autom. Syst.*, vol. 19, no. 1, pp. 88–100, 2020, doi: 10.1007/s12555-018-0508-9.

[11] Y. Wang and Y. Zhang, "Stability Analysis of Network Control System with Time Delay and Packet Dropout," *Comput. Eng.*, vol. 41, no. 10, pp. 111–116, 2015, doi: 10.3969/j.issn.1000-3428.2015.10.021.

[12] P. Witczak *et al.*, "A robust predictive actuator fault-tolerant control scheme for Takagi-Sugeno fuzzy systems," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 63, no. 4, pp. 977–987, 2015, doi: 10.1515/bpasts-2015-0111.

[13] G. Min, J. Shun, and P. Feng, "Random fault detection for networked control systems with packet losses," *J. Nanjing Univ. Sci. Technol.*, vol. 42, no. 3, pp. 292–299, 2018, doi: 10.14177/j.cnki.32-1397n.2018.42.03.006.

[14] F. Fang *et al.*, "Fault tolerant sampled-data H∞ control for networked control systems with probabilistic time-varying delay," *Inf. Sci.*, vol. 544, no. 8, pp. 395–414, 2021, doi: 10.1016/j.ins.2020.08.063.

[15] L. Hongfei *et al.*, "Integral-based event-triggered fault estimation and impulsive fault-tolerant control for networked control systems applied to underwater vehicles," *Neurocomputing*, vol. 442, no. 2, pp. 36–47, 2021, doi: 10.1016/j.neucom.2021.02.035.

[16] X. Bu, F. Yu, Z. Hou, and F. Wang, "Iterative learning control for a class of nonlinear systems with measurement dropouts," *Control Theory Appl.*, vol. 29, no. 11, pp. 1458–1464, 2012, doi: 10.1016/j.nonrwa.2012.07.017.

[17] J. Yang, M. Dou, and D. Zhao, "Iterative sliding mode observer for sensorless control of five-phase permanent magnet synchronous motor," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 65, no. 6, pp. 845–857, 2017, doi: 10.1515/bpasts-2017-0092.

[18] D. Shen, "Iterative learning control using faded measurements without system information: a gradient estimation approach," *Int. J. Syst. Sci.*, vol. 51, no. 14, pp. 2675–2689, 2020, doi: 10.1080/00207721.2020.1799258.

[19] T. Hongfeng *et al*., "Robust point-to-point iterative learning control with trial-varying initial conditions," *IET Control Theory Appl.*, vol. 14, no. 19, pp. 3344–3350, 2020, doi: 10.1049/iet-cta.2020.0557.

[20] L. Shaozhe *et al.*, "An Experience Transfer Approach for the Initial Data of Iterative Learning Control," *Appl. Sci.s*, vol. 11, no. 4, pp. 1631–1631, 2021, doi: 10.3390/app11041631.

[21] D. Meng and K.L. Moore, "Convergence of iterative learning control for SISO nonrepetitive systems subject to iteration-dependent uncertainties," *Automatica*, vol. 79, pp. 167–177, 2017, doi: 10.1016/j.automatica.2017.02.009.

[22] D. Meng and K.L. Moore. "Robust Iterative Learning Control for Nonrepetitive Uncertain Systems," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 907–913, 2017, doi: 10.1109/TAC.2016.2560961.

[23] J. Ling *et al*., "A Master-slave Cross-coupled Iterative Learning Control for Repetitive Tracking of Nonlinear Contours in Multi-axis Precision Motion Systems," *Acta Autom. Sin.*, vol. 43, no. 12, pp. 2127–2140, 2017, doi: 10.16383/j.aas.2017.c160725.

[24] X. Hao and D. Wang. "Fuzzy PID Iterative learning control for a class of nonlinear systems with arbitrary initial value," in *Proceedings of the 7th International Conference on Education, Management, Computer and Medicine (EMCM)*, 2016, pp. 304–311, doi: 10.2991/emcm-16.2017.60.

[25] K.-H. Park, "A study on the robustness of a PID-type iterative learning controller against initial state error," *Int. J. Syst. Sci.*, vol. 30, no. 1, pp. 49–59, 2012, doi: 10.1080/002077299292669.

[26] Y. Zhao *et al.*, "Fractional-order iterative learning control with initial state learning design," *Nonlinear Dyn.*, vol. 90, pp. 1257–1268, 2017, doi: 10.1007/s11071-017-3724-6.

[27] Y.H. Lan, "Iterative learning control with initial state learning for fractional order nonlinear systems," *Comput. Math. Appl.*, vol. 64, no. 10, pp. 3210–3216, 2012, doi: 10.1016/j.camwa.2012.03.086.

[28] Y.H Lan and Z.M. Cui, "ILC with Initial State Learning for Fractional Order Linear Distributed Parameter Systems," *Algorithms*, vol. 11, no. 6, pp. 85–96, 2018, doi: 10.3390/a11060085.

[29] Y.S. Wei and X.D. Li, "Robust higher-order ILC for non-linear discrete-time systems with varying trail lengths and random initial state shifts," *IET Control Theory Appl.*, vol. 11, no. 15, pp. 2440–2447, 2017, doi: 10.1049/iet-cta.2017.0008.

[30] W. Xia, X. Xu, T. Lu, and P. Ma, "Two Degrees of Freedom Vehicle TrajectoryTracking Based on Iterative Control," *J. Luoyang Normal Univ.*, vol. 34, no. 5, pp. 37–39, 2015, doi: 10.16594/j.cnki.41-1302/g4.2015.05.029.