Open-closed-loop iterative learning control for a class of nonlinear systems with random data dropouts

X Y Cheng*, H B Wang, Y L Jia and YH Dong

The Key Lab of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China

E-mail: 94180102@qq.com

Abstract: In this paper, an open-closed-loop iterative learning control (ILC) algorithm is constructed for a class of nonlinear systems subjecting to random data dropouts. The ILC algorithm is implemented by a networked control system (NCS), where only the off-line data is transmitted by network while the real-time data is delivered in the point-to-point way. Thus, there are two controllers rather than one in the control system, which makes better use of the saved and current information and thereby improves the performance achieved by open-loop control alone. During the transfer of off-line data between the nonlinear plant and the remote controller data dropout occurs randomly and the data dropout rate is modeled as a binary Bernoulli random variable. Both measurement and control data dropouts are taken into consideration simultaneously. The convergence criterion is derived based on rigorous analysis. Finally, the simulation results verify the effectiveness of the proposed method.

1. INTRODUCTION

Iterative learning control (ILC) is a control method applied to systems where the same task is executed repetitively and periodically in a finite time interval, such as a robotic manipulator in a manufacturing environment or a chemical reactor in a batch processing application. By ILC, there is no need to know the system model completely, but the previous and/or current information is used to update the control law in order to achieve a better performance as the iteration process repeats. Because of its simplicity and effectiveness, ILC has been studied extensively by experts and scholars with significant progress in both theory and application since it was proposed by Arimoto in 1984 [1]. However, among all the achievements that have been made so far, the most centralized topics are the convergence analysis [2-3] and the robustness [4-6] of the proposed ILC methods.

On the other hand, owing to the rapid development of sensing, information processing, and communication technologies, the research in the emerging area of networked control systems (NCSs) has attracted considerable attention in the research community [7]. As a result, the introduction of the communication network to control systems has made great changes on the conventional control structure gradually. Compared with the traditional point-to-point control systems, such NCSs bring some new merits such as low power consumption and cost, easy installation and maintenance, reduced system wiring, and increased system flexibility. However, during data packets transfer between the remote ILC controller and the controlled plant, data dropout is inevitable and unpredictable due to restrictions such as network failure and the limited bandwidth in network communication. It is well known that ILC is a data-based method and data dropouts are potential sources of instability and poor
Finally, the conclusion section 4, convergence condition for the output (SISO) nonlinear system

The remainder of this paper is organized as follows. The problem formulation will be used in this paper. Convergence \[25\] while Th dropouts have been proved to be convergent, because the learning control structure is basically open loop. Instead, both \[8\] and \[9\] just take the measurement data dropout into account. In \[10\], both measurement and control data dropouts are considered by Ahn and his co-workers. The framework established in \[10\] is able to cover various uncertainties, which makes it possible to deal with more general engineering problems. Bu and his co-workers concentrate on ILC algorithms for discrete linear systems subjecting to measurement dropouts in their early works \[11-14\]. Super-vector approach, which is based on a lifting technique, is widely used in their studies on account of the advantage that the two-dimensional problem of ILC can be changed into the one-dimensional multi-input multi-output problem. Notably, the stability of high order ILC schemes with data dropouts is discussed in \[12-13\]. In \[14\], H∞ iterative learning controller design has been discussed. Then they extend their studies to systems where data dropouts are modeled as stochastic variables satisfying the Bernoulli binary random distribution \[15-17\]. As super-vector approach can’t be used in the nonlinear systems, a contraction mapping approach is proposed in \[15-16\]. In \[17\], the ILC design has been investigated based on 2-D stochastic Roesser system and the condition for mean-square asymptotic stability is given by a linear matrix inequality (LMI) technique. In paper \[18\], the bad effect of data dropouts on convergence speed and stability of the system is discussed. Shen and his co-workers focus on the study of stochastic iterative learning control (SILC) which is defined as ILC for systems containing system noises, measurement noises, random packet losses and other stochastic signals \[19-22\].

The data dropout is modeled by an arbitrary stochastic sequence with bounded length requirement and the almost sure convergence analysis is given, which are the remarkable differences from the above. In addition, Liu et al. apply an averaging ILC scheme in \[23\] to overcome the random factors in nonlinear non-affine processes with random data dropouts. Wang et al. give the convergence conditions for the robust ILC method based on contraction mapping approach and a forgetting factor is introduced to obtain a better performance in \[24\]. Except for the achievements presented above, there are still many unmentioned scholars and researchers who have made prominent contributions to the study of the ILC with data dropouts.

However, all the aforementioned papers are restricted to open-loop ILC update laws. It is very possible that the tracking error can become quite large before finally converging to zero even when the applied ILC algorithm has been proved to be convergent, because the learning control structure is basically open-loop. Inspired by \[25\], a closed-loop is added to the existing open-loop ILC method for a class of nonlinear systems suffering data dropouts in this paper. In addition to eliminating the huge overshoot, faster convergence speed and better performance in stability and robustness of the system are obtained because of the introduction of the closed-loop. The main contributions of this paper different from \[25\] are stated as follows: i) Linear systems are considered in \[25\] while a class of nonlinear systems, which is more complex, are studied in this paper. ii) The condition of convergence is derived with the analytical methods of 2-D system in \[25\] while contraction mapping approach is used in this paper. iii) When data dropout occurs, the current data will be zero in \[25\], while the previous data will be used instead in this paper, which can weaken the bad effect on the system caused by data dropout in some degree.

The remainder of this paper is organized as follows. The problem formulation for a class of single-input-single-output (SISO) nonlinear systems and design of controllers are described in section 2. In section 3, the convergence condition for the SISO nonlinear system is given and the convergence analysis is provided. In section 4, the result of the SISO nonlinear system is extended to a class of multi-input-multi-output (MIMO) nonlinear systems. In section 5, a numerical example is given to verify the effectiveness of the proposed method. Finally, the conclusion is given in section 6.

2. PROBLEM FORMULATION AND DESIGN OF CONTROLLERS

2.1. Problem formulation

Consider a SISO nonlinear discrete-time system \[16\] described in the following form:

\[
\begin{align*}
    x_i(k+1) &= f(x_i(k)) + b(x_i(k))u_i(k), \\
    y_j(k) &= g(x_j(k)) + d(x_j(k))u_j(k),
\end{align*}
\]

(1)
where the subscript $i$ denotes the iteration number and $k$ denotes the time instance in an iteration process, $i = 0,1,2,\ldots, k \in \{0,1,2,\ldots,N\}$; $x_i(k) \in \mathbb{R}^1$, $u_i(k) \in \mathbb{R}^1$ and $y_i(k) \in \mathbb{R}^1$ are the state, control input and output of the system, respectively.

Some relevant assumptions of the system (1) are given as follows.

**Assumption 1:** The desired trajectory $y_d(k)$ is iteration invariant.

**Assumption 2:** For $y_d(k)$, there exist unique $u_d(k)$ and $x_d(k)$ satisfying the following equations:

\[
\begin{align*}
  x_d(k+1) &= f(x_d(k)) + b(x_d(k))u_d(k), \\
  y_d(k) &= g(x_d(k)) + d(x_d(k))u_d(k),
\end{align*}
\]

where $u_d(k)$ is the desired control input and $x_d(k)$ is the desired state.

**Assumption 3:** The resetting condition below is satisfied for every iteration.

\[x_i(0) = x_d(0),\]

where $x_d(0)$ is the initial value of the desired state and $x_i(0)$ is the initial value of the $i$th iteration.

**Assumption 4:** The nonlinear functions $f(\cdot)$, $b(\cdot)$, $g(\cdot)$, and $d(\cdot)$ are global Lipschitz in $x$, that is, for every $k \in \{0,N\}$, $i = 0,1,2,3L$, there exist positive bounded constants $k_f$, $k_b$, $k_g$ and $k_d$, satisfying the following inequalities:

\[
\begin{align*}
  |f(x_1(k)) - f(x_2(k))| &\leq k_f |x_1(k) - x_2(k)|, \\
  |b(x_1(k)) - b(x_2(k))| &\leq k_b |x_1(k) - x_2(k)|, \\
  |g(x_1(k)) - g(x_2(k))| &\leq k_g |x_1(k) - x_2(k)|, \\
  |d(x_1(k)) - d(x_2(k))| &\leq k_d |x_1(k) - x_2(k)|,
\end{align*}
\]

for any pair $(x_1(k), x_2(k))$ in $\mathbb{R} \times \mathbb{R}$, where $x_1, x_2 \in x_i(k), i = 0,1,2,3L$, $k = 0,1,2,\ldots, N$.

The structure of the NCS in this paper is illustrated in Fig. 1 and the main difference from the general NCS is that there are two controllers in the system rather than one. Thus, the control input $u_i(k)$ consists of two parts, one is $u_{ff,i}(k)$ produced by the feed-forward controller and the other is $u_{fb,i}(k)$ produced by the feed-back controller. This can be described as follows:

\[u_i(k) = u_{ff,i}(k) + u_{fb,i}(k),\]

![Fig. 1. The block diagram of the NCS](image)

It’s obvious that only the data of the feed-forward controller is interchanged by the network. However, as the local controller, the data of the feed-back controller is delivered in the point-to-point way, which makes it no sense to consider data dropouts in feed-back loop. Then, the following discussion will concentrate on the feed-forward loop. As shown in Fig. 1, there are two different types of data dropouts. The first one occurs when the
measurement output $y_i(k)$ is transmitted between the sensor and the remote controller, and the other one occurs when the control input $\hat{u}_{ff,i}(k)$ generated by the remote controller is delivered to the actuator. In order to describe this situation vividly, the communication network can be regarded as a switch that opens and closes in a random way. Suppose data successful transfer rate $\alpha$ and $\beta$ are stochastic parameters, which consist of independent and identically distributed Bernoulli random variables taking the value of 0 or 1 with

$$\text{prob}\{\alpha = 1\} = \mathcal{E}\{\alpha\} = \alpha, \quad 0 \leq \alpha \leq 1,$$
$$\text{prob}\{\beta = 1\} = \mathcal{E}\{\beta\} = \beta, \quad 0 \leq \beta \leq 1,$$

where $\alpha$ and $\beta$ are known constants and $\mathcal{E}\{\epsilon\}$ stands for the mathematical expectation of $\epsilon$. It is assumed that $\alpha$ and $\beta$ are independent of each other. When the switch is open, the current data packet is lost and the previous data stored in the Memory1/Memory2 will be used instead, otherwise, the current data will be delivered successfully and the data stored in the Memory1/Memory2 will be updated synchronously. That is, the actual value $\hat{y}_i(k)$ received by the feed-forward controller is the current output $y_i(k)$ with the probability of $\alpha$ and the previous value $y_i(k-1)$ with the probability of $1-\alpha$. Similarly, the actual input $u_{ff,i}(k)$ transmitted to the actuator is the current feed-forward controller output $\hat{u}_{ff,i}(k)$ with the probability of $\beta$ and the previous value $\hat{u}_{ff,i}(k-1)$ with the probability of $1-\beta$. These can be represented as follows:

$$\hat{y}_i(k) = \alpha y_i(k) + (1-\alpha)y_i(k-1), \quad (9)$$
$$u_{ff,i}(k) = \beta \hat{u}_{ff,i}(k) + (1-\beta)\hat{u}_{ff,i}(k-1). \quad (10)$$

2.2. Design of controllers

An open-loop P-type ILC law is adopted for the remote controller while a closed-loop P-type law is used for the local controller. The ILC laws are shown below:

$$\hat{u}_{ff,i}(k) = \hat{u}_{ff,i-1}(k) + \gamma_{ff}\hat{e}_{i-1}(k), \quad (11)$$
$$u_{fb,i}(k) = \gamma_{fb}e_i(k), \quad (12)$$

where $\gamma_{ff}$ and $\gamma_{fb}$ are the learning gain of the remote controller and the local controller, respectively. $\hat{e}_{i-1}(k) = y_d(k) - \hat{y}_{i-1}(k)$ and $e_i(k) = y_d(k) - y_i(k)$.

3. CONVERGENCE ANALYSIS

In this section, the convergence condition of the controllers for the SISO nonlinear system is given and proved.

**Theorem 1:** For system (1), assume Assumptions 1-4 hold, applying the update laws (11) and (12) to the feed-forward controller and the feed-back controller, respectively, if $\rho_{i,k} = \frac{1-(\beta_1\alpha - \gamma_{fb})d(x_i(k))}{1+\gamma_{fb}d(x_{i+1}(k))} < 1$ is satisfied for all $i$ and $k$, then $\lim_{i \to \infty} y_i(k) = y_d(k)$ is guaranteed for all $k$.

**Proof:** By introducing equations (10), (11) and (12) into equation (8) and making subtraction of the adjacent iterations, we have
\[ u_{i+1}(k) - u_i(k) = u_{g_{ij}}(k) + u_{b_i}(k) - u_{g_{ij}}(k) - u_{b_i}(k) \]
\[ = \beta(u_{g_{ij}}(k) - \hat{u}_{g_{ij}}(k)) + (1 - \beta)(\hat{u}_{g_{ij}}(k - 1)) - \hat{u}_{g_{ij}}(k - 1)) + \gamma \hat{e}_{i+1}(k) - \hat{e}_i(k) \]
\[ = \beta \gamma \hat{e}_i(k) + (1 - \beta) \gamma \hat{e}_i(k - 1) + \gamma \hat{e}_{i+1}(k) - \hat{e}_i(k) \]
\[ = \hat{e}_i(k), \]
where \( \hat{e}_i(k) \) and \( e_i(k) \) conform to the following relationship
\[ \hat{e}_i(k) = y_d(k) - \alpha y_i(k) - (1 - \alpha) y_i(k - 1) \]
\[ = \alpha e_i(k) + (1 - \alpha) e_i(k - 1). \]
Substituting (14) for \( \hat{e}_i(k) \) in (13) leads to
\[ u_{i+1}(k) = u_i(k) + \gamma \hat{e}_{i+1}(k) + (\beta \gamma \hat{e}_i(k) - \gamma \hat{e}_i(k) - 1) + (1 - \beta) \gamma \hat{e}_i(k - 1) + (1 - \beta) \gamma e_i(k - 2). \]
Note that
\[ \left\{ \begin{array}{l}
y_d(k) = g(x_d(k)) + d(x_d(k))u_d(k), \\
y_d(k) = g(x_d(k)) + d(x_d(k))u_d(k),
\end{array} \right. \]
thus
\[ e_i(k) = y_d(k) - y_i(k) \]
\[ = g(x_d(k)) + d(x_d(k))u_d(k) - g(x_i(k)) - d(x_i(k))u_i(k) \]
\[ = g(x_d(k)) - g(x_i(k)) + [d(x_d(k)) - d(x_i(k))]u_d(k) + d(x_i(k))u_i(k), \]
where \( \delta u_i(k) = u_{g_{ij}}(k) - u_i(k) \).
For simplicity in writing, define
\[ h(x_i(k)) = g(x_d(k)) - g(x_i(k)) \]
\[ + [d(x_d(k)) - d(x_i(k))]u_d(k), \]
then (17) can be rewritten as
\[ e_i(k) = h(x_i(k)) + d(x_i(k))\delta u_i(k). \]
Replacing \( e_i(k) \) for (19) in (15) leads to
\[ u_{i+1}(k) \]
\[ = u_i(k) + l_1 \left[ h(x_{i+1}(k)) + d(x_{i+1}(k)) \delta u_{i+1}(k) \right] + l_2 \left[ h(x_i(k)) + d(x_i(k)) \delta u_i(k) \right] + l_3 \left[ h(x_i(k-1)) \right] + l_4 \left[ h(x_i(k-2)) + d(x_i(k-2)) \delta u_i(k-2) \right], \]

where
\[ l_1 = \gamma_{\beta}, \quad l_2 = \beta \gamma_g \alpha - \gamma_{\beta}, \]
\[ l_3 = \beta \gamma_g (1-\alpha) + (1-\beta) \gamma_g \alpha, \]
\[ l_4 = (1-\beta) \gamma_g (1-\alpha). \]

Subtracting both sides of the equation (20) from \( u_d(k) \), we have
\[ \delta u_{i+1}(k) \]
\[ = \delta u_i(k) - l_1 \left[ h(x_{i+1}(k)) + d(x_{i+1}(k)) \delta u_{i+1}(k) \right] - l_2 \left[ h(x_i(k)) + d(x_i(k)) \delta u_i(k) \right] - l_3 \left[ h(x_i(k-1)) \right] - l_4 \left[ h(x_i(k-2)) + d(x_i(k-2)) \delta u_i(k-2) \right], \]

where \( \delta u_{i+1}(k) = u_d(k) - u_{i+1}(k) \).

After arranging and combining like terms, (21) can be rewritten as
\[ \left[ 1 + l_d(x_{i+1}(k)) \right] \delta u_{i+1}(k) \]
\[ = \left[ 1 - l_2 d(x_i(k)) \right] \delta u_i(k) - l_3 d(x_i(k-1)) \delta u_i(k-1) - l_4 d(x_i(k-2)) \delta u_i(k-2) - l_2 h(x_i(k)) - l_4 h(x_i(k-2)). \]

Considering the Lipschitz condition in Assumption 4, (18) gives
\[ \left| h(x_i(k)) \right| \leq \left| g(x_d(k)) - g(x_i(k)) \right| + \left| u_d(k) \right| \]
\[ \left| d(x_d(k)) - d(x_i(k)) \right| \leq k_g + k_d b_{ud} \left| \delta x_i(k) \right|, \]

where \( b_{ud} = \sup_{k \in [0,N]} \left| u_d(k) \right|, \delta x_i(k) = x_d(k) - x_i(k) \).

Similarly, considering the Lipschitz condition in Assumption 4 for (22) and substituting (23) for \( \left| h(x_i(k)) \right| \) result in
\[ \left| 1 + l_d(x_{i+1}(k)) \right| \left| \delta u_{i+1}(k) \right| \leq \left| 1 - l_2 d(x_i(k)) \right| \left| \delta u_i(k) \right| + \left| l_3 d(x_i(k-1)) \right| \left| \delta u_i(k-1) \right| \]
\[ + \left| l_4 d(x_i(k-2)) \right| \left| \delta u_i(k-2) \right| \]
\[ + \left| k_g + k_d b_{ud} \left| \delta x_i(k) \right| \right| \leq \left| l_1 \right| \left| \delta u_{i+1}(k) \right| + \left| l_2 d(x_i(k-1)) \right| \left| \delta u_i(k) \right| \]
\[ + \left| l_3 d(x_i(k-2)) \right| \left| \delta u_i(k-1) \right| + \left| l_4 d(x_i(k-3)) \right| \left| \delta u_i(k-2) \right| \].
Taking expectation in both sides of (24), we have
\[
\mathbb{E}\left[\left|1 + \tilde{t}_1 d(x_{i+1}(k))\right|\right] \\
\leq \mathbb{E}\left[\left|1 - \tilde{t}_2 d(x_i(k)) + \tilde{t}_3 d(x_i(k - 1))\right|\right].
\]
\[
\mathbb{E}\left[\left|\delta u_i(k)\right|\right] + \mathbb{E}\left[\left|\delta u_i(k - 1)\right|\right] + \mathbb{E}\left[\left|\delta u_i(k - 2)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k - 1)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k - 2)\right|\right]
\]
(25)
\[
+ \mathbb{E}\left[\left|\tilde{t}_1\right|\right] \mathbb{E}\left[\left|\tilde{t}_2\right|\right] \mathbb{E}\left[\left|\tilde{t}_3\right|\right] \mathbb{E}\left[\left|\tilde{t}_4\right|\right].
\]

where
\[
\tilde{t}_1 = \gamma_{fb}, \quad \tilde{t}_2 = \bar{\beta}_g (1 - \alpha) - \gamma_{fb}, \quad \tilde{t}_3 = \bar{\beta}_g (1 - \alpha) + (1 - \bar{\beta}_g) \alpha,
\]
\[
\tilde{t}_4 = (1 - \bar{\beta}_g) \gamma_{fb} (1 - \alpha).
\]

Both sides of (25) are divided by \(1 + \tilde{t}_1 d(x_{i+1}(k))\), and then we can obtain that
\[
\mathbb{E}\left[\left|\delta u_i(k)\right|\right] \\
\leq \rho_{1,i,k} \mathbb{E}\left[\left|\delta u_i(k - 1)\right|\right] + \rho_{2,i,k} \mathbb{E}\left[\left|\delta u_i(k - 2)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k - 1)\right|\right] + \mathbb{E}\left[\left|\delta x_i(k - 2)\right|\right]
\]
(26)
\[
+ \mathbb{E}\left[\left|\tilde{t}_1\right|\right] \mathbb{E}\left[\left|\tilde{t}_2\right|\right] \mathbb{E}\left[\left|\tilde{t}_3\right|\right] \mathbb{E}\left[\left|\tilde{t}_4\right|\right].
\]

where
\[
\rho_{1,i,k} = \frac{\tilde{t}_1 d(x_i(k - 1))}{1 + \tilde{t}_1 d(x_{i+1}(k))}, \quad \rho_{2,i,k} = \frac{\tilde{t}_2 d(x_i(k - 2))}{1 + \tilde{t}_1 d(x_{i+1}(k))},
\]
\[
v_{i+1,k} = \frac{k_g + k_d b_{ud} \tilde{t}_1}{1 + \tilde{t}_1 d(x_{i+1}(k))}, \quad v_{i,k} = \frac{k_g + k_d b_{ud} \tilde{t}_2}{1 + \tilde{t}_1 d(x_{i+1}(k))},
\]
\[
v_{1,k} = \frac{k_g + k_d b_{ud} \tilde{t}_3}{1 + \tilde{t}_1 d(x_{i+1}(k))}, \quad v_{2,k} = \frac{k_g + k_d b_{ud} \tilde{t}_4}{1 + \tilde{t}_1 d(x_{i+1}(k))}.
\]

It is clear that
\[
\delta x_i(k) \\
= x_i(k) - x_i(k - 1)
\]
(27)
\[
= f(x_i(k - 1)) - f(x_i(k - 1)) + \left[ b(x_i(k - 1)) - b(x_i(k - 1)) \right] u_i(k - 1)
\]
\[
+ b(x_i(k - 1)) \delta u_i(k - 1),
\]

taking absolute value of both sides of the equation (27) and applying Lipschitz condition lead to
\[
\begin{align*}
\delta x_i(k) & = f(x_i(k-1)) - f(x_i(k-1)) \\
& + b(x_i(k-1)) - b(x_i(k-1))v_i(k-1) \\
& + b(x_i(k-1))\delta u_i(k-1) \\
& \leq (k_f + k_f b_{ud}) \delta x_i(k-1) + b(x_i(k-1)) \\
& + b(x_i(k-1))\delta u_i(k-1) \\
& = k_1 \delta x_i(k-1) + k_2 \delta u_i(k-1),
\end{align*}
\]

where \( k_1 = k_f + k_f b_{ud}, k_2 = \sup_{k \in [0,\infty)} b(x_i(k)) \).

Since \( |\delta x_i(0)| = |x_i(0) - x_i(0)| = 0 \), by substituting the relation recursively for \( \delta x_i(j), j = 0,1,2,3, \ldots k \), the following expression can be obtained
\[
|\delta x_i(k)| \leq \sum_{p=1}^{k} k_{p-1}^{p-1} k_2 |\delta u_i(k-p)|.
\]

Thus, we have
\[
\begin{align*}
\mathcal{E} \left( \|\delta u_{i+1}(k)\| \right) & \leq \rho_{i,1} \mathcal{E} \left( \|\delta u_i(k)\| \right) + \rho_{i,2} \mathcal{E} \left( \|\delta u_i(k-1)\| \right) + \rho_{i,3} \\
& + \mathcal{E} \left( \|\delta u_i(k-2)\| \right) + v_{i+1,1} \sum_{p=1}^{k} K_{0,1,1} \mathcal{E} \left( \|\delta u_{i+1}(k-p)\| \right) \\
& + v_{i+1,2} \sum_{p=2}^{k} K_{1,1,1} \mathcal{E} \left( \|\delta u_{i+1}(k-p)\| \right) \\
& + v_{i+1,3} \sum_{p=3}^{k} K_{2,1,1} \mathcal{E} \left( \|\delta u_{i+1}(k-p)\| \right),
\end{align*}
\]

where
\[
K_{0,1} = K_{1,1} = k_1^{p-1} k_2, \\
K_{2,1} = k_1^{p-2} k_2, \\
K_{3,1} = k_1^{p-3} k_2.
\]

Expanding expression (30) from \( k = 0 \) to \( k = N \), we have
\[
\mathcal{E} \left( \|\delta u_{i+1}(0)\| \right) \leq \mathcal{E} \left( \|\delta u_i(0)\| \right),
\]
\[
\mathcal{E} \left( \|\delta u_{i+1}(1)\| \right) \leq \rho_{i,1} \mathcal{E} \left( \|\delta u_i(1)\| \right) + \rho_{i,2} \mathcal{E} \left( \|\delta u_i(0)\| \right) \\
+ v_{i+1,1} K_{0,1,1} \mathcal{E} \left( \|\delta u_{i+1}(0)\| \right) + v_{i,1} K_{1,1,1} \mathcal{E} \left( \|\delta u_i(0)\| \right) \\
\leq \rho_{i,1} \mathcal{E} \left( \|\delta u_i(1)\| \right) + \left( \rho_{i,1} + v_{i,1} K_{1,1,1} + \rho_{i,2} v_{i+1,1} K_{0,1,1} \right) \mathcal{E} \left( \|\delta u_i(0)\| \right) \\
= \rho_{i,1} \mathcal{E} \left( \|\delta u_i(1)\| \right) + L_{i,0} \mathcal{E} \left( \|\delta u_i(0)\| \right),
\]
\[
\varepsilon \left\| \delta u_{i+1}(2) \right\| \\
\leq \rho_{1,2} \varepsilon \left\| \delta u_i(2) \right\| + \rho_{0,12} \left\| \delta u_i(1) \right\| + \rho_{1,2} \varepsilon \left\| \delta u_i(0) \right\| \\
+ v_{i+1,2} K_{0,1} \varepsilon \left\| \delta u_{i+1}(1) \right\| + v_{i+1,2} K_{0,2} \varepsilon \left\| \delta u_{i+1}(0) \right\| \\
+ v_{1,2} K_{1,1} \varepsilon \left\| \delta u_i(1) \right\| + v_{1,2} K_{1,2} \varepsilon \left\| \delta u_i(0) \right\| \\
+ v_{1,2} K_{2,2} \varepsilon \left\| \delta u_i(0) \right\| \\
\leq \rho_{1,2} \varepsilon \left\| \delta u_i(2) \right\| + \left( \rho_{1,2} + \rho_{1,2} v_{i+1,2} K_{0,1} + v_{1,2} K_{1,1} \right) \\
\cdot \left\| \delta u_i(1) \right\| + \left( \rho_{2,2} + L_{1,0} v_{i+1,2} K_{0,1} + \rho_{2,2} v_{i+1,2} K_{0,2} \right) \left\| \delta u_i(0) \right\| \\
= \rho_{1,2} \varepsilon \left\| \delta u_i(2) \right\| + L_{2,1} \left\| \delta u_i(1) \right\| + L_{0,2} \left\| \delta u_i(0) \right\| \\
+ v_{i,1} \sum_{p=1}^{N} K_{i,p} \varepsilon \left\| \delta u_i(N-p) \right\| \\
+ v_{i,1} \sum_{p=2}^{N} K_{2,p} \varepsilon \left\| \delta u_i(N-p) \right\| \\
+ v_{1,2} \sum_{p=3}^{N} K_{3,p} \varepsilon \left\| \delta u_i(N-p) \right\| \\
\varepsilon \left\| \delta u_{i+1}(N) \right\| \\
\leq \rho_{2,2} \varepsilon \left\| \delta u_i(N) \right\| + \left( \rho_{2,2} + v_{i+1,2} K_{0,1} \rho_{i,2} + v_{1,2} K_{1,1} \right) \\
\cdot \left\| \delta u_i(N-1) \right\| + \left( \rho_{2,2} + v_{i+1,2} K_{0,1} L_{N-1,N-2} + v_{i+1,N} \right) \\
\varepsilon \left\| \delta u_i(N-2) \right\| \cdot v_{i+1,2} \sum_{p=1}^{N} K_{0,p} \varepsilon \left\| \delta u_{i+1}(N-p) \right\| \\
+ v_{i+1,N} \sum_{p=1}^{N} K_{0,p} \cdot L_{N-p,0} + v_{i,1} N_{K_{1,1}} + v_{i,2} N_{K_{2,2}} \\
+ v_{1,2} \sum_{p=3}^{N} K_{3,p} \cdot \varepsilon \left\| \delta u_i(0) \right\| \\
= \rho_{2,2} \varepsilon \left\| \delta u_i(N) \right\| + L_{N,N-1} \varepsilon \left\| \delta u_i(N-1) \right\| \\
+ L_{N,N-2} \varepsilon \left\| \delta u_i(N-2) \right\| + L_{N,0} \varepsilon \left\| \delta u_i(N-1) \right\|
\]

where \( L_{m,n} (m,n \in [0,N]) \) is a known constant.

The above expressions can be arranged in a matrix form as below

\[
\Delta U_{i+1} \leq \Phi_i \Delta U_i
\]

where

\[
\Delta U_i = \begin{bmatrix}
\varepsilon \left\| \delta u_i(0) \right\| \\
\varepsilon \left\| \delta u_i(1) \right\| \\
\varepsilon \left\| \delta u_i(2) \right\| \\
\vdots \\
\varepsilon \left\| \delta u_i(N) \right\|
\end{bmatrix}
\]
for any pair
inequalities
vector
4
difference exist
if
\lim_{i \to \infty} \epsilon \left\| \delta u_i(k) \right\| = 0 \quad \text{as} \quad i \to \infty \quad \text{for all} \quad k \in \{0,1,2,L Na\}. \quad \text{Since} \quad \Phi_i \quad \text{is a lower triangular matrix, all its eigenvalues at the} \ i-th \ \text{iteration are} \ \rho_{i,k}. \ \text{Therefore,}
\begin{align*}
\rho_{i,k} &= \left| 1 - \left( \overline{\beta}_f \beta - \gamma_f \right) \left[ \frac{d(x_i(k))}{1 + \gamma_f d(x_i+1(k))} \right] \right| < 1 \quad \text{holds for all} \quad i \ \text{and} \ \k \ \text{then} \ \lim_{i \to \infty} \epsilon \left\| \delta u_i(k) \right\| = 0. \quad \text{That is}
\lim_{i \to \infty} \epsilon \left\| y_i(k) \right\| = u_d(k), \quad \text{which, in terms of Assumption 2, implies that} \ \lim_{i \to \infty} \epsilon \left\| y_i(k) \right\| = y_d(k) \quad \text{is guaranteed for all} \quad k \in \{0,1,2,L Na\}. \quad \text{This is the end of proof.} \quad \square
\end{align*}

Remark 1. Theorem 1 is the convergence condition for the P-type ILC algorithms as shown in (11) and (12). It is obvious that this proof method is also suitable for the PD-type and PID-type ILC algorithms. The only difference existing in the convergence condition is that the adjustable parameters are different in each algorithm.

4. Extension to a class of MIMO nonlinear systems

The result of Theorem 2 can be extended to a class of MIMO nonlinear systems satisfying the following form:
\begin{align}
\begin{aligned}
x_i(k+1) &= f(x_i(k)) + b(x_i(k))u_i(k), \\
y_i(k) &= g(x_i(k)) + d(x_i(k))u_i(k),
\end{aligned}
\end{align}
(31)
where \(i\) and \(k\) denote the same meaning as in system (1). \(x_i(k) \in \mathbb{R}^l, u_i(k) \in \mathbb{R}^m\) and \(y_i(k) \in \mathbb{R}^n\) are the state vector, control input vector and output vector of the system, respectively.

Some relevant assumptions of the system (31) are given as follows.

Assumption 5: The desired trajectory \(y_d(k)\) is iteration invariant.

Assumption 6: For \(y_d(k)\), there exist unique \(u_d(k)\) and \(x_d(k)\) meeting the following equations:
\begin{align}
\begin{aligned}
x_d(k+1) &= f(x_d(k)) + b(x_d(k))u_d(k), \\
y_d(k) &= g(x_d(k)) + d(x_d(k))u_d(k),
\end{aligned}
\end{align}
(32)
where \(u_d(k)\) is the desired control input vector and \(x_d(k)\) is the desired state vector.

Assumption 7: The resetting condition below is satisfied for every iteration.
\begin{align}
x_i(0) = x_d(0),
\end{align}
(33)
where \(x_d(0)\) is the initial value of the desired state vector and \(x_i(0)\) is the initial value of the \(i\)-th iteration.

Assumption 8: The nonlinear function vectors and matrices \(f(\cdot) \in \mathbb{R}^l, b(\cdot) \in \mathbb{R}^{lm}, g(\cdot) \in \mathbb{R}^n\) and \(d(\cdot) \in \mathbb{R}^{mn}\) are global Lipschitz in \(x\), that is, there exist positive bounded constants \(k_f, k_b, k_g, k_D\), satisfying the following inequations:
\begin{align}
\|f(x_1(k)) - f(x_2(k))\| &\leq k_f \|x_1(k) - x_2(k)\|, \quad (34) \\
\|b(x_1(k)) - b(x_2(k))\| &\leq k_b \|x_1(k) - x_2(k)\|, \quad (35) \\
\|g(x_1(k)) - g(x_2(k))\| &\leq k_g \|x_1(k) - x_2(k)\|, \quad (36) \\
\|d(x_1(k)) - d(x_2(k))\| &\leq k_D \|x_1(k) - x_2(k)\|, \quad (37)
\end{align}
for any pair of \((x_1(k), x_2(k))\) in \(\mathbb{R}^l \times \mathbb{R}^l\), where \(x_1, x_2 \in x_i(k), i = 0,1,2,3L, k \in [0,N]\).
\[ \|v\|_v = \max_{1 \leq i \leq l} |m_i|, \]

where \( M \in \mathbb{R}^n \), \( m_j \) is the \( j \)th element of the vector \( M \).

\[ \|N\|_n = \max_{1 \leq i \leq m} \sum_{j=1}^{n_i} |n_{ij}|, \]

where \( N \in \mathbb{R}^{m \times n} \), \( n_j \) is the element of the matrix \( N \).

As shown in Fig.1, two types of data dropouts are considered in the MIMO nonlinear system. Suppose each component in the multivariable output or input vector of the nonlinear system suffers the same data dropout rate. Then the relationship can be expressed as follows:

\[ \hat{y}_i(k) = Ay_i(k) + (I - A)y_i(k - 1), \quad (38) \]
\[ u_{gf,i}(k) = Bu_{gf,i}(k) + (I - B)u_{gf,i}(k - 1), \quad (39) \]

where

\[
A = \begin{pmatrix} \alpha & O \\ O & \beta \end{pmatrix}, \quad B = \begin{pmatrix} \beta & O \\ O & \beta \end{pmatrix},
\]

and \( \alpha, \beta \) denote the same meaning as in the SISO system. Then (38) and (39) can be rewritten as:

\[ \hat{y}_i(k) = \alpha y_i(k) + (1 - \alpha)y_i(k - 1), \quad (40) \]
\[ u_{gf,i}(k) = \beta u_{gf,i}(k) + (1 - \beta)u_{gf,i}(k - 1), \quad (41) \]

Then the P-type ILC law for the feed-forward controller and P-type law for the feed-back controller are given as follows:

\[ \bar{u}_{gf,i}(k) = \bar{u}_{gf,i}(k - 1) + \bar{\Theta}_{gf,i}(k), \quad (42) \]
\[ u_{fb,i}(k) = \Theta_{fb,i}(k), \quad (43) \]

where

\[
\Theta = \begin{pmatrix} \gamma_{gf1} & O \\ O & \gamma_{fb1} \end{pmatrix}, \quad \bar{\Theta} = \begin{pmatrix} \gamma_{gf1} \\ \gamma_{fb1} \end{pmatrix},
\]

are the learning gain matrices of the remote controller and the local controller, respectively.

**Theorem 2:** For system (31), assume Assumptions 5-8 hold, applying the update laws (42) and (43) to the feed-forward controller and the feed-back controller, respectively, if \( \rho_i = \frac{1 - (\beta \Theta - \Theta) \Theta d(x_i(k))}{\Theta + \Theta d(x_i(k))} < 1 \) is satisfied for all \( i \) and \( k \), then \( \lim_{i \to \infty} \|y_i(k) - y_d(k)\|_v = 0 \) is guaranteed for all \( k \).

The proof process of the Theorem 2 is similar to the Theorem 1, so the proof process is omitted here.
5. NUMERICAL EXAMPLE

Consider the following SISO nonlinear system
\[
\begin{align*}
    x_i(k+1) &= \sin(x_i(k)) + (1-x_i(k))u_i(k) \\
    y_i(k) &= \cos(x_i(k)) + u_i(k)
\end{align*}
\]  

(44)

where \( k \in [0, 200] \). The desired tracking trajectory is \( y_d(k) = \sin(\pi k / 300) + \cos(\pi k / 400) \), the initial condition is \( x_i(0) = x_d(0) = 0 \).

Three different cases of data dropouts are taken into account for the SISO nonlinear system (44):

Case 1: \( \alpha = \beta = 1 \), no data dropout.

Case 2: \( \alpha = 0.9 \), 10% data dropouts.

Case 3: \( \alpha = 0.8 \), 20% data dropouts.

For further verifying the effectiveness of the open-closed-loop ILC algorithm, we also take the open-loop ILC algorithm into consideration. When the learning gain \( \gamma_f \) is chosen as 0, the open-loop ILC algorithm can be regarded as an open-loop ILC algorithm. Given to Theorem 1, the learning gain is chosen as \( \gamma_f = 5 \), \( \gamma_f = 4.5 \) in the open-closed-loop and \( \gamma_f = 0 \), \( \gamma_f = 0.9 \) in the open-loop.

It is assumed that \( d(x_i(k)) = 1 \) in system (1) and check the convergence condition in Theorem 1, we can obtain

\[
\begin{align*}
    \text{Case 1: } \rho_{i,k} &= \frac{1 - (\beta \gamma_f \alpha - \gamma_f) d(x_i(k))}{1 + \gamma_f d(x_{i+1}(k))} = 0.25 < 1, \\
    \text{Case 2: } \rho_{i,k} &= \frac{1 - (\beta \gamma_f \alpha - \gamma_f) d(x_i(k))}{1 + \gamma_f d(x_{i+1}(k))} = 0.39 < 1, \\
    \text{Case 3: } \rho_{i,k} &= \frac{1 - (\beta \gamma_f \alpha - \gamma_f) d(x_i(k))}{1 + \gamma_f d(x_{i+1}(k))} = 0.52 < 1,
\end{align*}
\]

so \( \lim_{k \to \infty} \| y_i(k) \| = y_d(k) \) is granted for all \( k \).

Through the theoretical analysis in section 3, it can be easily concluded that the smaller \( \rho_{i,k} \) is, the faster convergence speed can be obtained. That is why the convergence speed gets slower and slower as shown in Fig. 2, Fig. 4 and Fig. 6. However, the maximum tracking error eventually goes to zero as the iteration increases, which can verify the effectiveness of the proposed method. Besides, by comparing the two figures in Fig. 2, Fig. 4 and Fig. 6, respectively, it could be better understood that, under the same data dropout rate, in addition to a faster convergence speed, the tracking error is also reduced greatly in the open-closed loop compared with the open loop.

![Fig. 2. The maximum tracking error with no data dropout](image-url)
Fig. 3. The output of 20 times iteration with no data dropout

Fig. 4. The maximum tracking error with 10% data dropouts

Fig. 5. The output of 20 times iteration with 10% data dropouts

Fig. 6. The maximum tracking error with 20% data dropouts

Fig. 3, Fig. 5 and Fig. 7 describe the initialized output of the system ($i=0$), the output of 20 times iteration ($i=1, 2, \ldots, 20$) and the desired output, that is to say, there are twenty-two lines in Fig. 3, Fig. 5 and Fig. 7, respectively. However, the actual output lines will be coincident with the desired output line after some times of iteration, which is why the lines are less than twenty-two visually. It can be seen that the actual output of the system gets closer and closer to the desired output as the iteration number increases and the desired trajectory can be tracked completely after some iterations.
REFERENCES

This paper is supported by National Nature Science Foundation of China: 61473248.

6. CONCLUSION

An open-closed-loop ILC method for a class of nonlinear systems with random data dropouts is introduced in this paper. The main contribution of this paper is adding a closed-loop to the existing ILC method for nonlinear systems. Though the data dropouts have a bad effect on the convergence speed, the ILC algorithm can achieve convergence eventually as long as the packet dropout is incompletete. It has faster convergence speed, greatly reduced tracking error and better performance in stability and robustness due to the open-closed-loop ILC.

When the current data packet loses during the transmission, the previous data stored in the memory will be used instead in this paper. However, some more efficient compensation methods for data dropouts will be considered as one of our further research topics, especially when the data dropout rate is high. The other is the extension to a class of MIMO systems where each component in the multivariable output or input vector of the system suffers different data dropout rate.

7. ACKNOWLEDGEMENT

This paper is supported by National Nature Science Foundation of China: 61473248.

REFERENCES

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, “Bettering operation of robots by learning,” Journal of Robotic Systems, vol. 1, no. 2, pp. 123-140, 1984.
[2] K. L. Moore, Y.-Q. Chen, and V. Bahl, “Monotonically convergent iterative learning control for linear discrete-time systems,” Automatica, vol. 41, no. 9, pp. 1529-1537, June 2005.
[3] X.-E. Ruan and Z.-Z. Li, “Convergence characteristics of PD-type iterative learning control in discrete frequency domain,” Journal of Process Control, vol. 24, no. 12, pp. 86-94, November 2014.
[4] J. Van de Wijdeven, T. Donkers, and O. Bosgra, “Iterative learning control for uncertain systems: robust monotonic convergence analysis,” Automatica, vol. 45, no. 10, pp. 2383-2391, August 2009.
[5] D. Meng, Y. Jia, J. Du, and F. Yu, “Robust iterative learning control design for uncertain time-delay systems based on a performance index,” Control Theory & Applications, IET, vol. 4, no. 5, pp. 759-772, 2010.
[6] T. Y. Doh, J. R. Ryoo, and D. E. Chang, “Robust iterative learning controller design using the performance weighting function of feedback control systems,” International Journal of Control, Automation and Systems, vol. 12, no. 1, pp. 63-70, 2014.
[7] K.-Y. You and L.-H. Xie, “Survey of recent progress in networked control systems,” Acta Automatica Sinica, vol. 39, no. 2, pp. 101-117, February 2013.
[8] H. S. Ahn, Y.-Q. Chen, and K. L. Moore, “Intermittent iterative learning control,” Proc. of the 2006 IEEE International Symposium on Intelligent Control, pp. 832-837, Munich, Germany, 4-6 October, 2006.
[9] H. S. Ahn, K. L. Moore, and Y.-Q. Chen, “Discrete-time intermittent iterative learning controller with independent data dropouts,” Proc. of the 2008 IFAC World Congress, pp. 12442-12447, Seoul, Korea, 2008.
[10] H. S. Ahn, K. L. Moore, and Y.-Q. Chen, “Stability of Discrete-time iterative learning control with random data dropouts and delayed controlled signals in networked control systems,” In Proceedings of International Conference on Control Automation, Robotics and Vision, IEEE, pp. 757-762, Hanoi, Vietnam, 17–20 December, 2008.
[11] X.-H. Bu, Z.-S. Hou, and H.-R. Dong, “Stability analysis of iterative learning control with data dropouts,”
Proc. of the 2009 IEEE International Conference on Control and Automation, pp. 1146-1151, Christchurch, New Zealand, 9-11 December, 2009.

[12] X.-H. Bu and Z.-S. Hou, “Stability of iterative learning control with data dropouts via asynchronous dynamical system,” International Journal of Automation and Computing, vol. 8, no. 1, pp. 29-36, February 2011.

[13] X.-H. Bu, Z.-S. Hou, and F.-S. Yu, “Stability of first and high order iterative learning control with data dropouts,” International Journal of Control, Automation and Systems, vol. 9, no. 5, pp. 843-849, 2011.

[14] X.-H. Bu, Z.-S. Hou, F.-S. Yu, and F.-Z. Wang, “$H_\infty$ iterative learning controller design for a class of discrete-time systems with data dropouts,” International Journal of Systems Science, vol. 45, no. 9, pp. 1902-1912, 2014.

[15] X.-H. Bu, F.-S. Yu, Z.-S. Hou, and F.-Z. Wang, “Iterative learning control for a class of nonlinear systems with measurement dropouts,” Control Theory & Applications, vol. 29, no.11, pp. 1458-1464, November 2012.

[16] X.-H. Bu, F.-S. Yu, Z.-S. Hou, and F.-Z. Wang, “Iterative learning control for a class of nonlinear systems with random packet losses,” Nonlinear Analysis: Real World Applications, vol. 14, no. 1, pp. 567-580, 2013.

[17] X.-H. Bu, Z.-S. Hou, S.-T. Jin, and R.-H. Chi, “An iterative learning control design approach for networked control systems with data dropouts,” International Journal of Robust and Nonlinear Control, in press.

[18] X.-H. Bu, Z.-S. Hou, F.-S. Yu, and Z.-Y. Fu, “Effect analysis of data dropout on iterative learning control”, Control and Decision, vol.29, no. 3, pp. 443-448, March 2013.

[19] D. Shen and Y.-Q. Wang, “Survey on stochastic iterative learning control,” Journal of Process Control, vol. 24, no. 12, pp. 64-77, May 2014.

[20] D. Shen and Y.-Q. Wang, “Iterative learning control for networked stochastic systems with random packet losses,” International Journal of Control, pp. 1-10, 2014.

[21] D. Shen, Y. Zhang, and Y.-Q. Wang, “ILC for networked nonlinear systems with random measurement losses and unknown control direction,” The 11th IEEE International Conference on Control & Automation (ICCA), pp. 1438-1443, Taichung, Taiwan, 18-20 June, 2014.

[22] D. Shen and Y.-Q. Wang, “ILC for Networked Discrete Systems with Random Data Dropouts: A Switched System Approach,” Proceedings of the 33rd Chinese Control Conference, pp. 8670-8677, Nanjing, China, 28-30 July, 2014.

[23] C.-P. Liu, J.-X. Xu, and J. Wu, “Iterative learning control for network systems with communication delay or data dropout,” Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference, pp. 4858-4863, Shanghai, China, 16-18 December, 2009.

[24] R. Wang, X.-H. Ma, and X.-H. Bu, “Robust Iterative Learning Control for a Class of Linear Systems with Data Dropouts,” Mathematical Problems in Engineering, vol. 2013, pp. 1-9, 2013.

[25] Z. Zhao, Y. Gao, and Z.-L. Liu, “Novel open-closed-loop iterative learning control in networked control systems,” Control Theory & Applications, vol. 30, no. 10, pp. 1335-1341, October 2013. (in Chinese)