

# Neural Network State Learning Based Adaptive Terminal ILC for Tracking Iteration-varying Target Points

Yu Liu<sup>1</sup>   Rong-Hu Chi<sup>1</sup>   Zhong-Sheng Hou<sup>2</sup>

<sup>1</sup>School of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao 266042, China

<sup>2</sup>Advanced Control Systems Lab, School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing 100044, China

**Abstract:** Terminal iterative learning control (TILC) is developed to reduce the error between system output and a fixed desired point at the terminal end of operation interval over iterations under strictly identical initial conditions. In this work, the initial states are not required to be identical further but can be varying from iteration to iteration. In addition, the desired terminal point is not fixed any more but is allowed to change run-to-run. Consequently, a new adaptive TILC is proposed with a neural network initial state learning mechanism to achieve the learning objective over iterations. The neural network is used to approximate the effect of iteration-varying initial states on the terminal output and the neural network weights are identified iteratively along the iteration axis. A dead-zone scheme is developed such that both learning and adaptation are performed only if the terminal tracking error is outside a designated error bound. It is shown that the proposed approach is able to track run-varying terminal desired points fast with a specified tracking accuracy beyond the initial state variance.

**Keywords:** Adaptive terminal iterative learning control, neural network, initial state learning, iteration-varying terminal desired points, initial state variance.

## 1 Introduction

Iterative learning control (ILC) is to update the control sequence by errors in the previous trials for repetitive operation systems<sup>[1]</sup>. It can achieve perfect tracking performance with system uncertainties and has been successfully applied to repetitive systems<sup>[2–7]</sup>.

Note that the ILC approach requires the measurement of I/O signals over the entire tracking trajectory, which may not be accessible in some real industrial cases. For example, in the control of an oven's temperature for wafer deposition thickness control, only the end-point error is measurable.

To overcome this problem, terminal iterative learning control (TILC) was introduced to use the terminal point only at the end of every run<sup>[8]</sup>. Now, TILC has become a new research direction of ILC with many applications, including the thickness control of rapid thermal processing chemical vapor deposition<sup>[8]</sup>, the plastic sheet surface temperature control via thermoforming machines<sup>[9]</sup>, the final product quality control of batch to batch process<sup>[10]</sup>, the control of multi-joint hand-arm robots<sup>[11]</sup>, the station stop control of a train<sup>[12]</sup>, etc.

The majority of TILC schemes in literature focus on P-type learning law<sup>[8–12]</sup> and normal optimal learning law<sup>[13]</sup>,

and require good knowledge about system model to obtain learning gains. For example, the optimal learning law calculates control gains using the exactly known system matrices to solve a Riccati equation<sup>[13]</sup>. In addition, the learning gains are given beforehand and remain fixed over iterations, which is not adaptive or robust to the system uncertainties and exogenous disturbances. Consequently, a bad control performance might be expected with fixed learning gains in the presence of severe uncertainties.

Note that all existing TILCs are limited to track iteration-invariant fixed target point and require the initial states to be identical<sup>[8–13]</sup>. This makes the control schemes quite depend on the reference, i.e., once the tracking target point changes, the controller cannot track it immediately and has to learn again. What is more, the constraint on initial state also hinders the extensive applications of TILC in real industry.

Actually, it is very often that the industrial processes are carried out alternatively with different products and different terminal target points. For example, in the widely agile manufacturing, product objective changes dynamically with customers' demands. Further, the nonzero errors always exist in the initial states because of the exogenous disturbance, system uncertainties, etc.

Recently, several adaptive ILC approaches have been proposed to relax the requirement of identical conditions on the initial states and target trajectories<sup>[14–18]</sup>. However, these methods are obtained based on the certainty equivalency principle and thus depends on the intermediate I/O measurements at all sampling time instants. Apparently, the traditional adaptive ILC approaches cannot be directly applied to the terminal control task, where only the terminal

Regular Paper  
Special Issue on Latest Advances in ILC/RLC Theory and Applications

Manuscript received May 26, 2014; accepted October 28, 2014

This work was supported by National Natural Science Foundation of China (Nos. 61374102, 61433002 and 61120106009) and High Education Science & Technology Fund Planning Project of Shandong Province of China (No. J14LN30).

Recommended by Associate Editor Min Wu

© Institute of Automation, Chinese Academy of Science and Springer-Verlag Berlin Heidelberg 2015

measurement is available for learning rather than a whole output trajectory<sup>[14–18]</sup>.

It is known that neural network has been proved to be efficient in function approximation and parameter estimation<sup>[19]</sup>. In this paper, a new neural network initial state learning based adaptive terminal ILC (NNISL-ATILC) is proposed to address the terminal tracking tasks with random initial states and iteration-varying target end-points.

First, the relationship among terminal outputs, initial states, and control inputs is formulated for a linear multiple-input multiple-output (MIMO) discrete system. Then, the effect of initial state on the terminal output is approximated by an radial basis function (RBF) neural network and an iterative updating law is designed to estimate the neural network weighting factors iteratively, where a dead-zone scheme is designed such that both learning and adaptation are performed only if the terminal tracking error is outside a designated error bound. Compared with the existing terminal ILC approaches, the new proposed NNISL-ATILC can perform well when the initial states and the target terminal points are varying with iterations. The convergence is shown with rigorous analysis. Simulation results verify the effectiveness of the proposed approach further.

The rest of this paper is organized as follows. Section 2 presents the problem formulation of the TILC with iteration-varying initial states and iteration-varying terminal reference points. Section 3 designs an adaptive terminal ILC approach with a neural network initial state learning mechanism. Section 4 shows the stability and convergence of the proposed approach with rigorous analysis. Section 5 provides simulation results to illustrate the effectiveness of the proposed approach. Finally, some conclusions are given in Section 6.

## 2 Problem formulation

Consider a linear MIMO discrete-time system as

$$\begin{cases} x_k(t+1) = Ax_k(t) + Bu_k \\ y_k(t) = Cx_k(t) \end{cases} \quad (1)$$

where  $t = 0, 1, \dots, N$  is the sampling index and  $k$  is the iteration index. All the system matrices  $A \in \mathbf{R}^{p \times p}$ ,  $B \in \mathbf{R}^{p \times m}$  and  $C \in \mathbf{R}^{n \times p}$  are unknown,  $x_k(t) \in \mathbf{R}^p$  is the state vector,  $y_k(t) \in \mathbf{R}^n$  is the output vector, and  $u_k \in \mathbf{R}^m$  is the control vector which keeps constant at all sampling times during one iteration.

From linear control theory, the output of the system at the  $N$ -th time instant (terminal point) in the  $k$ -th iteration is given as

$$y_k(N) = CA^N x_k(0) + C \sum_{t=0}^{N-1} A^{N-t-1} Bu_k \quad (2)$$

which can be rewritten into the general form as

$$y_k(N) = f(x_k(0)) + B^* u_k \quad (3)$$

where  $f(x_k(0)) = CA^N x_k(0)$  and  $B^* = C \sum_{t=0}^{N-1} A^{N-t-1} B$ .

To approximate the effect of the initial states to the terminal output, i.e.,  $f(x_k(0))$ , an RBF neural network is introduced, i.e.,

$$\hat{f}(x_k(0)) = W^T \Phi(x_k(0)) \quad (4)$$

where  $W^T \in \mathbf{R}^{n \times L}$  is an unknown ideal weight matrix,  $L$  denotes the number of neurons in the hidden layer. Function  $\hat{f}(\cdot)$  depends on the structure of the neural network and the number of neurons.  $\Phi(x_k(0)) \in \mathbf{R}^L$  is the output of the neural network's hidden layer.

Therefore, the system terminal output can be approximated by the above RBF network as

$$y_k^{NN}(N) = \hat{f}(x_k(0), W) + B^* u_k = W^T \Phi(x_k(0)) + B^* u_k. \quad (5)$$

Let  $\Theta = [W^T, B^*]$ ,  $\Psi_k = [\Phi(x_k(0))^T, u_k^T]^T$ . Then, (5) becomes

$$y_k^{NN}(N) = \Theta \Psi_k. \quad (6)$$

To restrict our discussion, some assumptions are considered.

**Assumption 1.** The system is both controllable and observable,  $n \leq m$ , and  $B^* \in \mathbf{R}^{n \times m}$  is full rank.

**Assumption 2.** Given a positive constant  $\varepsilon$  and a compact set  $S$ , there exists a coefficient  $W$  such that  $\hat{f}(\cdot)$  approximates the continuous function  $f(\cdot)$  with accuracy  $\varepsilon$  over  $S$ , i.e.,  $\forall y \in S$  and  $\forall k, \exists W$  such that

$$\begin{aligned} \max \|\hat{f}(x_k(0), W) - f(x_k(0))\| &= \\ \max \|W^T \Phi(x_k(0)) - f(x_k(0))\| &\leq \varepsilon. \end{aligned} \quad (7)$$

**Assumption 3.** The initial state  $x_k(0)$  is accessible in every iteration.

**Remark 1.** Assumption 1 is common in control theory. Assumption 2 is always satisfied if both the structure of the network and the number of neurons are properly specified.

**Remark 2.** Assumption 3 is often used in ILC and TILC for repetitive operation systems, in which the initial state will be reset before the starting of every iteration. And the reset value is fixed or follows some fixed pattern in practice. For example, in the thickness control for the wafer fabrication, the initial states including the temperature of the oven and initial thickness are quite standard. So it is reasonable to assume that the initial state is known in ILC and TILC.

**Remark 3.** Note that for a linear system, the effect of the initial states to the terminal output may be a nonlinear function about the initial state due to the measure noises and disturbances, although the expression is a linear form as in (2). Therefore, it is necessary to use an RBF neural network to approximate it as in (4).

In this paper, the desired terminal reference point, defined as  $y_{d,k}(N)$ , is iteration-varying. This is a major relaxation of the common assumption of identical terminal target point required in the existing TILC approaches<sup>[8–13]</sup>.

The control objective is to refine the input sequence by learning from the iteration-varying terminal target points directly such that the TILC system achieves a perfect control performance for iteration-different tracking tasks.

### 3 Controller design

According to the feedback equivalent principle, the proposed terminal learning control law at the  $k$ -th iteration is

$$u_k = (\hat{B}_k^*)^{-1} [y_{d,k}(N) - \hat{W}_k^T \Phi(x_k(0))] \quad (8)$$

where  $\hat{W}_k$  and  $\hat{B}_k^*$  denote the estimates of  $W$  and  $B^*$  at the  $k$ -th iteration, respectively.

The control input  $u_k$  is applied to both the plant and the neural network model. The unknown  $B^*$  and  $W$  are updated iteratively according to the estimation error.

According to (5), the estimated plant output is

$$\hat{y}_k(N) = \hat{f}(x_k(0), \hat{W}_k) + \hat{B}_k^* u_k = \hat{W}_k^T \Phi(x_k(0)) + \hat{B}_k^* u_k. \quad (9)$$

Define the estimate error as

$$\hat{e}_k(N) = \hat{y}_k(N) - y_k(N) \quad (10)$$

where  $\hat{e}_k(N) = [\hat{e}_{1,k}(N), \hat{e}_{2,k}(N), \dots, \hat{e}_{n,k}(N)]^T$ ,  $\hat{e}_{i,k}(N)$ ,  $i = 1, 2, \dots, n$ , is the  $i$ -th element of  $\hat{e}_k(N)$ . To overcome the model uncertainties in the stability analysis, we are going to employ a dead-zone algorithm for updating the weights. The dead-zone function is defined as

$$D(\hat{e}_{i,k}(N)) = \begin{cases} 0, & \text{if } |\hat{e}_{i,k}(N)| \leq d_0 \\ e_i - d_0, & \text{if } \hat{e}_{i,k}(N) > d_0 \\ e_i + d_0, & \text{if } \hat{e}_{i,k}(N) < -d_0 \end{cases} \quad (11)$$

Let  $\hat{\Theta}_k = [\hat{W}_k^T, \hat{B}_k^*]$ ,  $\hat{\theta}_{i,k}$  denote the  $i$ -th row vector of  $\hat{\Theta}_k$ ,  $\hat{\omega}_{i,k}^T$  and  $\hat{b}_{i,k}^*$  denote the  $i$ -th row vectors of  $\hat{W}_k^T$  and  $\hat{B}_k^*$ , respectively. Then,  $\hat{\theta}_{i,k} = [\hat{\omega}_{i,k}^T, \hat{b}_{i,k}^*]$ ,  $i = 1, 2, \dots, n$ . Hence, the estimated model (9) becomes

$$\hat{y}_{i,k}(N) = \hat{\omega}_{i,k}^T \Phi(x_k(0)) + \hat{b}_{i,k}^* u_k = \hat{\theta}_{i,k} \Psi_k. \quad (12)$$

Now, the neural network weight updating law with dead-zone is proposed as

$$\hat{\theta}_{i,k+1} = \hat{\theta}_{i,k} - \frac{\Psi_k^T}{1 + \Psi_k^T \Psi_k} D(\hat{e}_{i,k}(N)), i = 1, \dots, n \quad (13)$$

where the initial values of  $\hat{\theta}_{i,0}$  are bounded.

### 4 Convergence analysis

The validity of the above proposed neural network initial state learning based adaptive TILC scheme (8)–(13) is verified by the following theorem.

**Theorem 1.** For system (1) under Assumptions 1–3, the learning control law (8) and the parameter updating

law (13) together with the dead-zone algorithm (11) guarantee that:

1)  $\hat{\Theta}_k$  is bounded and the parameter estimate error  $\|\hat{\Theta}_k - \Theta\|$  is monotonically non-increasing in the iteration domain.

2) The terminal tracking error between the plant output and the desired terminal point,  $e_k(N) = y_{d,k}(N) - y_k(N)$ , converges to a ball of radius  $d_0$  centered at the origin as  $k$  approaches to infinity.

**Proof.** The proof of Theorem 1 has two parts as follows.

**Part 1.** The boundedness of  $\hat{\Theta}_k$ .

Let  $\omega_i^T$  denote the  $i$ -th row vector of  $W^T$ , and  $b_i^*$  denote the  $i$ -th row vector of  $B^*$ ,  $i = 1, 2, \dots, n$ . Then, (5) can be rewritten as

$$y_{i,k}^{NN}(N) = \hat{f}_i(x_k(0), \omega_i) + b_i^* u_k = \omega_i^T \Phi(x_k(0)) + b_i^* u_k = \theta_i \Psi_k \quad (14)$$

where  $\theta_i = [\omega_i^T, b_i^*]$ .

According to (14) and Assumption 2, we have

$$y_{i,k}(N) = \hat{f}_i(x_k(0), \omega_i) + b_i^* u_k + [f_i(x_k(0)) - \hat{f}_i(x_k(0), \omega_i)] = \hat{f}_i(x_k(0), \omega_i) + b_i^* u_k + O(\varepsilon). \quad (15)$$

In terms of (10), (14) and (15), the estimation error between the neural network output and the plant output is

$$\begin{aligned} \hat{e}_{i,k}(N) &= \hat{y}_{i,k}(N) - y_{i,k}(N) = \hat{f}_i(x_k(0), \hat{\omega}_{i,k}) - \hat{f}_i(x_k(0), \omega_i) + (\hat{b}_{i,k}^* - b_i^*) u_k + O(\varepsilon) = \\ &= [\hat{\omega}_{i,k} - \omega_i]^T \Phi(x_k(0)) + \tilde{b}_{i,k}^* u_k + O(\varepsilon) = \\ &= \tilde{\omega}_{i,k}^T \Phi(x_k(0)) + \tilde{b}_{i,k}^* u_k + O(\varepsilon) = \tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) \end{aligned} \quad (16)$$

where  $\tilde{\omega}_{i,k} = \hat{\omega}_{i,k} - \omega_i$ ,  $\tilde{b}_{i,k}^* = \hat{b}_{i,k}^* - b_i^*$  and  $\tilde{\theta}_{i,k} = \hat{\theta}_{i,k} - \theta_i$ .

Assume that  $\varepsilon$  is small enough such that

$$|O(\varepsilon)| \leq M < d_0 \quad (17)$$

where  $d_0$  is defined in (11).

Using the definition of the dead-zone function (11), we can easily verify the following claims:

Case 1. If  $|\hat{e}_{i,k}(N)| \leq d_0$ , then  $D(\hat{e}_{i,k}(N)) = 0$ .

Case 2. If  $\hat{e}_{i,k}(N) > d_0$ , i.e.,  $\tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) > d_0$ , then

$$\tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) > 0.$$

Since  $O(\varepsilon) \leq M < d_0$ ,

$$\begin{aligned} D(\hat{e}_{i,k}(N)) &= \tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) - d_0 > \\ \tilde{\theta}_{i,k} \Psi_k + d_0 - d_0 &= \tilde{\theta}_{i,k} \Psi_k. \end{aligned}$$

Case 3. If  $\hat{e}_{i,k}(N) < -d_0$ , i.e.,  $\tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) < -d_0$ , then

$$\tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) < 0.$$

Since  $O(\varepsilon) \leq M < d_0$ ,

$$D(\hat{e}_{i,k}(N)) = \tilde{\theta}_{i,k} \Psi_k + O(\varepsilon) + d_0 < \tilde{\theta}_{i,k} \Psi_k - d_0 + d_0 = \tilde{\theta}_{i,k} \Psi_k.$$

Thus, from the the above three cases,  $D(\hat{e}_{i,k}(N))$  can be represented as

$$D(\hat{e}_{i,k}(N)) = \alpha_{i,k} \tilde{\theta}_{i,k} \Psi_k \quad (18)$$

where  $0 \leq \alpha_{i,k} < 1$ .

Substituting (18) into the updating rule (13) yields

$$\hat{\theta}_{i,k+1} = \hat{\theta}_{i,k} - \frac{\alpha_{i,k} \tilde{\theta}_{i,k} \Psi_k}{1 + \Psi_k^T \Psi_k} \Psi_k^T. \quad (19)$$

By subtracting  $\theta_i$  from both sides of (19), it becomes

$$\tilde{\theta}_{i,k+1} = \tilde{\theta}_{i,k} - \frac{\alpha_{i,k} \tilde{\theta}_{i,k} \Psi_k}{1 + \Psi_k^T \Psi_k} \Psi_k^T. \quad (20)$$

Define a Lyapunov function  $V_k = \|\tilde{\theta}_{i,k}\|^2$ . Then,

$$\Delta V_{k+1} = V_{k+1} - V_k = \tilde{\theta}_{i,k+1}^T \tilde{\theta}_{i,k+1} - \tilde{\theta}_{i,k}^T \tilde{\theta}_{i,k}. \quad (21)$$

According to (20), it is easy to get from (21) that

$$\begin{aligned} \Delta V_{k+1} &= -2\alpha_{i,k} \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} + \alpha_{i,k}^2 \frac{\|\Psi_k\|^2 (\tilde{\theta}_{i,k} \Psi_k)^2}{(1 + \|\Psi_k\|^2)^2} \leq \\ &= -2\alpha_{i,k} \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} + \alpha_{i,k}^2 \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} \leq \\ &= -\alpha_{i,k}^2 \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} - 2(\alpha_{i,k} - \alpha_{i,k}^2) \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} \leq \\ &= -\alpha_{i,k}^2 \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} \leq 0. \end{aligned} \quad (22)$$

As a direct result, (22) shows the non-increasing property of  $\tilde{\theta}_{i,k}$  and the boundedness of the estimate value of neural network weights  $\hat{\theta}_{i,k}$  for all  $k$ . Thus, conclusion 1) of Theorem 1 is obtained directly.

**Part 2.** Convergence property of the tracking error  
Summing up both sides of (22) from 0 to  $k$  yields

$$V_{k+1} \leq V_0 - \sum_{j=0}^k \alpha_{i,j}^2 \frac{(\tilde{\theta}_{i,j} \Psi_j)^2}{1 + \|\Psi_j\|^2}. \quad (23)$$

Since  $V_0$  is bounded, and  $V_{k+1}$  is non-increasing and non-negative, in terms of (23), we obtain that

$$\lim_{k \rightarrow \infty} \alpha_{i,k}^2 \frac{(\tilde{\theta}_{i,k} \Psi_k)^2}{1 + \|\Psi_k\|^2} = 0. \quad (24)$$

It implies that

$$\lim_{k \rightarrow \infty} \alpha_{i,k} \tilde{\theta}_{i,k} \Psi_k = 0. \quad (25)$$

According to (18), we can obtain

$$\lim_{k \rightarrow \infty} D(\hat{e}_{i,k}(N)) = 0. \quad (26)$$

Hence,

$$|\hat{e}_{i,k}(N)| = |\hat{y}_{i,k}(N) - y_{i,k}(N)| < d_0, \quad \text{as } k \rightarrow \infty. \quad (27)$$

Substituting the control law (8) into (9) yields

$$\hat{y}_k(N) = \hat{W}_k^T \Phi(x_k(0)) + y_{d,k}(N) - \hat{W}_k^T \Phi(x_k(0)) = y_{d,k}(N). \quad (28)$$

Thereafter, the terminal tracking error is

$$\begin{aligned} \|e_k(N)\| &= \|y_{d,k}(N) - y_k(N)\| = \\ &= \|\hat{y}_k(N) - y_k(N)\| = \|\hat{e}_k(N)\|. \end{aligned} \quad (29)$$

In terms of (27),  $\lim_{k \rightarrow \infty} \hat{e}_{i,k}(N) < d_0$ . Then, the conclusion 2) of Theorem 1 is proved.  $\square$

## 5 Illustrative examples

In order to illustrate the effectiveness of the proposed NNISL-ATILC scheme, two examples are considered in simulation study.

**Example 1.** Consider the discrete-time single-input single-output (SISO) system as

$$\begin{aligned} x_k(t+1) &= \begin{pmatrix} 0.5 & 0.035 & 0.025 \\ 0.0255 & 0.6 & -0.99 \\ 0.75 & 0.03 & 0.025 \end{pmatrix} x_k(t) + \\ &\quad \begin{pmatrix} 0.2 & 0.2 & 0.0 \end{pmatrix}^T u_k \\ y_k(t) &= \begin{pmatrix} 1.0 & 0.0 & 1.0 \end{pmatrix} x_k(t) \end{aligned} \quad (30)$$

where the system operates on the interval  $[0, 20]$ .

In the simulation, the initial states are varying randomly over  $[0, 0.1]$  with iterations, as shown in Fig. 1. The tracking reference terminal points  $y_{d,k}(N) = 2 + 0.1 \sin(\frac{\pi k}{10})$ , which is varying iteratively and is shown in Fig. 2.

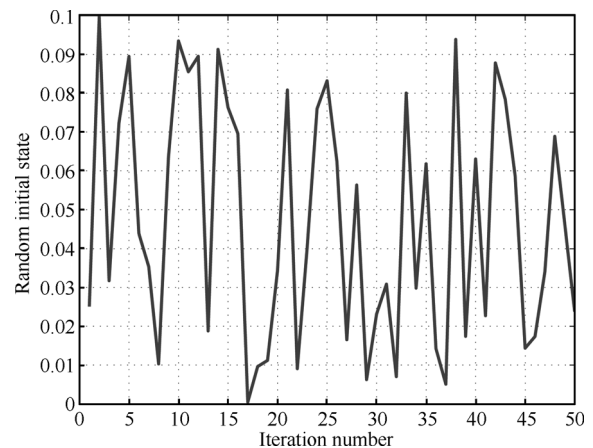


Fig. 1 The random initial states with iterations in Example 1

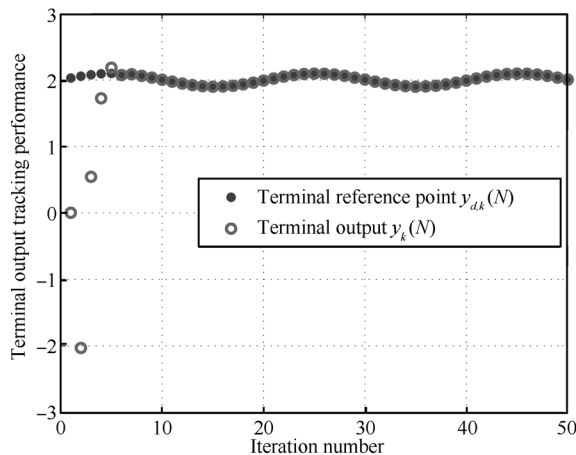


Fig. 2 Terminal output tracking performance with iterations in Example 1

By applying the proposed NNISL-ATILC (8)–(13) with  $d_0 = 0.001$ ,  $\hat{\Theta}_0 = \begin{bmatrix} 5 & 5 & 5 & 0.5 \end{bmatrix}$  and  $u_0 = 0$ , the terminal output profile and the terminal tracking error are given in Figs. 2 and 3, respectively. It can be seen that although the initial states and terminal reference points are varying iteratively along the iteration learning axis, the fast traceability and convergence of the proposed method can still be guaranteed.

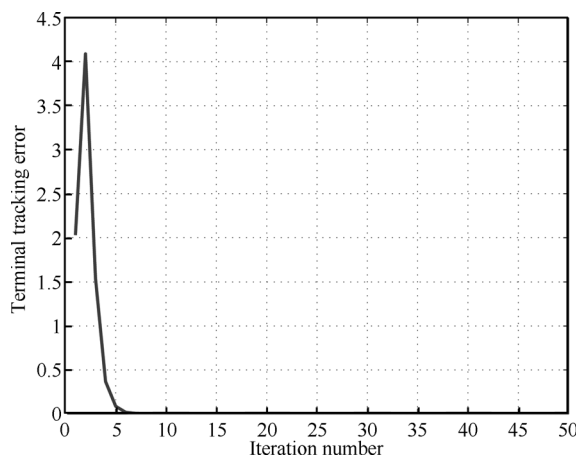


Fig. 3 Terminal tracking error with iterations in Example 1

**Example 2.** Consider an MIMO discrete-time system as

$$\begin{aligned}
 x_k(t+1) &= \begin{pmatrix} 0.5 & 0.035 & 0.025 \\ 0.0255 & 0.6 & -0.99 \\ 0.75 & 0.03 & 0.025 \end{pmatrix} x_k(t) + \\
 &\quad \begin{pmatrix} 0.2 & 0.03 & 0.025 \\ 0.2 & 0.2 & 0 \end{pmatrix}^T u_k \\
 y_k(t) &= \begin{pmatrix} 1.0 & 0.0 & 1.0 \\ 0 & 2.0 & 1.0 \end{pmatrix} x_k(t). \quad (31)
 \end{aligned}$$

The system operates on time interval  $[0, 20]$  for every iteration.

The initial state  $x_k(0)$  is also varying randomly with iterations over the interval of  $[0, 0.1]$ , as shown in Fig. 4.

In the simulation, the RBF neuron network works with 3 hidden neurons. The reference points in different iterations are shown in Fig. 5. They are also presented mathematically as

$$\begin{cases} y_{d1,k}(N) = 2 + 0.5\sin(\frac{\pi k}{20}) \\ y_{d2,k}(N) = 1 + 0.1\sin(\frac{\pi k}{20}). \end{cases} \quad (32)$$

By applying the proposed NNISL-ATILC (8)–(13) with  $d_0 = 0.001$ ,  $\hat{\Theta}_0 = \begin{bmatrix} 8 & 8 & 8 & 0.5 & 0.5 \\ 4 & 4 & 4 & -0.5 & 0.5 \end{bmatrix}$  and  $u_0 = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ , the simulation results are shown in Figs. 5–8, respectively. It is clear that the proposed approach is well to be used for discrete-time MIMO systems to track run-varying reference with initial state variance. As shown in Figs. 5 and 6, for each output, the proposed approach converges quite fast and the performance keeps good even the reference changes between iterations. From Figs. 7 and 8, it is shown that the fast convergence of the terminal errors in the iteration domain is evident for the proposed approach.

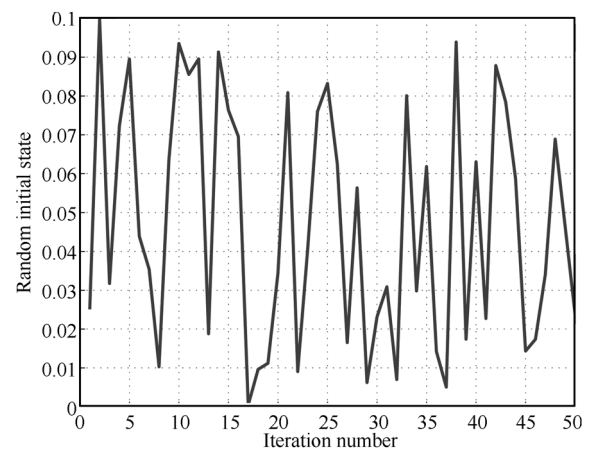


Fig. 4 Random-varying profile of the initial states in Example 2

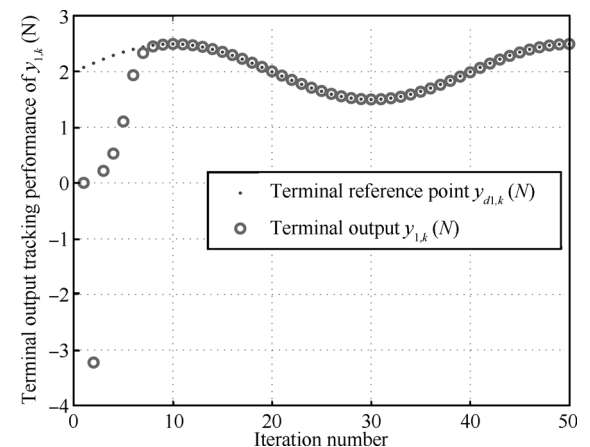


Fig. 5 Terminal output tracking performance of  $y_{1,k}$  in Example 2

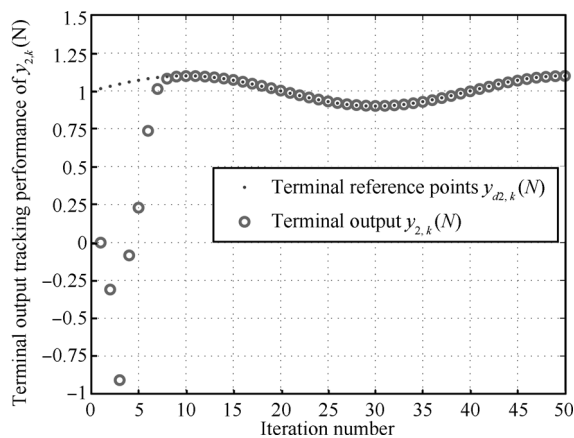


Fig. 6 Terminal output tracking performance of  $y_{2,k}$  in Example 2

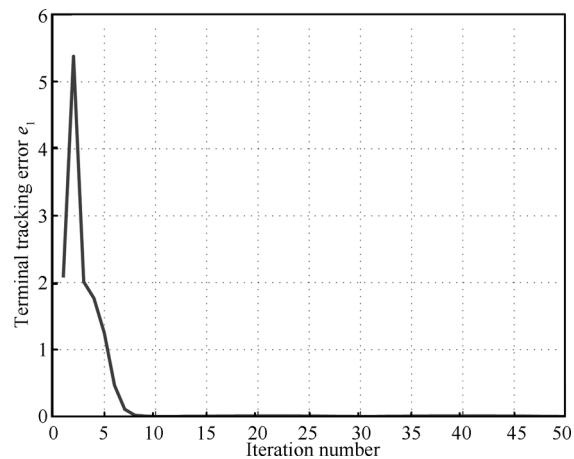


Fig. 7 Terminal tracking error  $e_1$  in Example 2

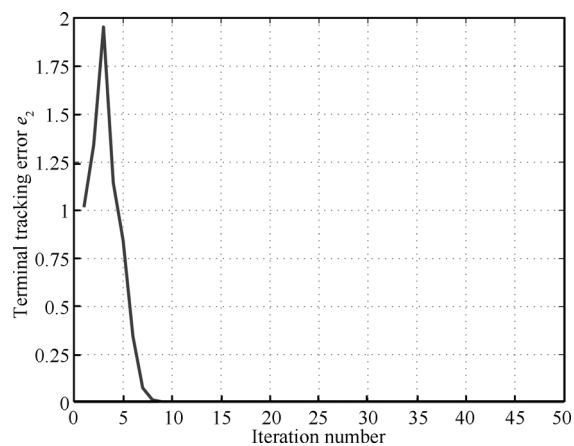


Fig. 8 Terminal tracking error  $e_2$  in Example 2

## 6 Conclusions

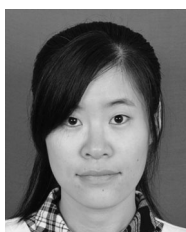
In this paper, a neural network initial state learning based adaptive terminal ILC is proposed to track iteration-varying terminal reference points with random initial state variances. The neural network is used to approximate the effect of iteration-varying initial states to the terminal output. The dead-zone scheme is designed to enhance the ro-

bustness of the control system and to achieve arbitrarily specified tracking accuracy. The proposed approach can converge very fast and has outstanding tracking performance even the initial state and reference terminal reference point are changed iteratively.

## References

- [1] S. Arimoto, S. Kawamura, F. Miyazaki. Bettering operation of robots by learning. *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.
- [2] X. D. Yu, Z. H. Xiong, D. X. Huang, Y. H. Jiang. Model-based iterative learning control for batch processes using generalized hinging hyperplanes. *Industrial & Engineering Chemistry Research*, vol. 52, no. 4, pp. 1627–1634, 2013.
- [3] D. Li, J. M. Li. Adaptive iterative learning control for nonlinearly parameterized systems with unknown time-varying delay and unknown control direction. *International Journal of Automation and Computing*, vol. 9, no. 6, pp. 578–586, 2012.
- [4] D. Y. Meng, Y. M. Jia, J. P. Du, F. S. Yu. Data-driven control for relative degree systems via iterative learning. *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2213–2225, 2011.
- [5] T. Liu, Y. Q. Wang. A synthetic approach for robust constrained iterative learning control of piecewise affine batch processes. *Automatica*, vol. 48, no. 11, pp. 2762–2775, 2012.
- [6] Y. Q. Wang, F. R. Gao, F. J. Doyle III. Survey on iterative learning control, repetitive control, and run-to-run control. *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.
- [7] J. X. Xu. A survey on iterative learning control for nonlinear systems. *International Journal of Control*, vol. 84, no. 7, pp. 1275–1294, 2011.
- [8] J. X. Xu, Y. Q. Chen, L. T. Heng, S. Yamamoto. Terminal iterative learning control with an application to RTPCVD thickness control. *Automatica*, vol. 35, no. 9, pp. 1535–1542, 1999.
- [9] G. Gauthier, B. Boulet. Terminal iterative learning control design with singular value decomposition decoupling for thermoforming ovens. In *Proceedings of the American Control Conference*, IEEE, St. Louis, MO, USA, pp. 1640–1645, 2009.
- [10] Z. H. Xiong, J. Zhang. A batch-to-batch iterative optimal control strategy based on recurrent neural network models. *Journal of Process Control*, vol. 15, no. 1, pp. 11–21, 2005.
- [11] S. Arimoto, M. Sekimoto, S. Kawamura. Iterative learning of specified motions in task-space for redundant multi-joint hand-arm robots. In *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, Roma, Italy, pp. 2867–2873, 2007.
- [12] Z. S. Hou, Y. Wang, C. K. Yin, T. Tang. Terminal iterative learning control based station stop control of a train. *International Journal of Control*, vol. 84, no. 7, pp. 1263–1274, 2011.
- [13] T. D. Son, H. S. Ahn. Terminal iterative learning control with multiple intermediate pass points. In *Proceedings of the American Control Conference*, IEEE, San Francisco, CA, USA, pp. 3651–3656, 2011.

- [14] J. X. Xu, J. Xu. On iterative learning from different tracking tasks in the presence of time-varying uncertainties. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 1, pp. 589–597, 2004.
- [15] C. J. Chien. A combined adaptive law for fuzzy iterative learning control of nonlinear systems with varying control tasks. *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 1, pp. 40–51, 2008.
- [16] R. H. Chi, Z. S. Hou, J. X. Xu. Adaptive ILC for a class of discrete-time systems with iteration-varying trajectory and random initial condition. *Automatica*, vol. 44, no. 8, pp. 2207–2213, 2008.
- [17] C. K. Yin, J. X. Xu, Z. S. Hou. An ILC scheme for a class of nonlinear continuous-time systems with time-iteration-varying parameters subject to second-order internal model. *Asian Journal of Control*, vol. 13, no. 1, pp. 126–135, 2011.
- [18] R. H. Chi, Z. S. Hou, S. T. Jin, D. W. Wang. Discrete-time adaptive ILC for non-parametric uncertain nonlinear systems with iteration-varying trajectory and random initial condition. *Asian Journal of Control*, vol. 15, no. 2, pp. 562–570, 2013.
- [19] J. Sjöberg, Q. H. Zhang, L. Ljung, A. Benveniste, B. Delyon, G. Pierre-Yves, H. Hjalmarsson, A. Juditsky. Nonlinear black-box modeling in system identification: A unified overview. *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.



**Yu Liu** received the bachelor's degree from Qilu University of Technology, China in 2013. She is currently a postgraduate with Qingdao University of Science and Technology, China.

Her research interests include iterative learning control and terminal iterative learning control.

E-mail: liuyu0113@yeah.net

ORCID iD: 0000-0001-7939-0-6308

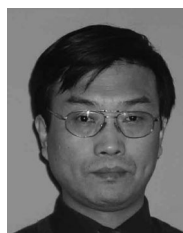


**Rong-Hu Chi** received the B.Sc. and M.Sc. degrees from Northwest A&F University, Xi'an, China in 1998 and 2003, respectively, and the Ph.D. degree from Beijing Jiaotong University, China in 2007. He is now a visiting scholar with Nanyang Technological University, Singapore. In 2007, he joined Qingdao University of Science and Technology, China. He is currently an associate professor in the School of Automation and Electronic Engineering, China.

His research interests include iterative learning control, adaptive control, data-driven control, model-free adaptive control, and intelligent transportation systems.

E-mail: ronghu\_chi@hotmail.com (Corresponding author)

ORCID iD: 0000-0002-1325-7863



**Zhong-Sheng Hou** received the B.Sc. and M.Sc. degrees from Jilin University of Technology, China in 1983 and 1988, respectively, and the Ph.D. degree from Northeastern University, Shenyang, China in 1994. He was a postdoctoral fellow with the Harbin Institute of Technology, China from 1995 to 1997, and a visiting scholar with Yale University, USA from 2002 to 2003. In 1997, he joined the Beijing Jiaotong University, China.

He is currently a full professor and the founding director of the Advanced Control Systems Lab and the head of Department of Automatic Control in School of Electronic and Information Engineering. Up to now, he has over 80 peer-reviewed journal papers published and 100 papers in prestigious conference proceedings. He is the author of two monographs *Nonparametric Model and its Adaptive Control Theory* and *Model Free Adaptive Control Theory and Applications*. He has served as a committee member for over 40 international and Chinese conferences, and as an associate editor and guest editor for a number of international journals.

His research interests include data-driven control, model free adaptive control, learning control, and intelligent transportation systems.

E-mail: zhshhou@bjtu.edu.cn