A Novel Design of Iterative Learning Control with Pure Feedforward Structure

CHI Ronghu[1], LIN Na[1], HOU Zhongsheng[2], ZHANG Ruikun[3]

1. School of Automation & Electronic Engineering, Qingdao University of Science & Technology
Qingdao 266042 (E-mail: ronghu_chi@hotmail.com)

2. Advanced Control Systems Lab, School of Electronics & Information Engineering, Beijing Jiaotong University
Beijing 100044 (E-mail: zhshhou@bjtu.edu.cn)

3. School of Mathematics and Physics, Qingdao University of Science & Technology
Qingdao 266042 (e-mail: rkzhang@qust.edu.cn)

Abstract: A new design framework of feedforward ILC is developed for nonlinear systems in this work. By supposing that there exists a desired nonlinear controller, two dynamical linearization methods are proposed for the nonlinear controller. And then, a CFDL-ILC and a PFDL-ILC are presented. Comparatively, the PFDL-ILC is similar to the higher order ILC schemes that more errors of previous iterations are used. The proposed approaches are data-driven and no process model is required for the controller design and analysis. The availability of the proposed approaches is further confirmed by simulation results.

Key Words: Iterative learning control, Nonlinear systems, Data-driven control

1 Introduction

Iterative learning control (ILC) [1-3] is most suitable for repetitive dynamics operating over a fixed time interval. In genral, there are three major structures of ILC system: Pure feedforward (PFF) ILC, Pure feedback (PFB) ILC, and feedback – feedforward (FBFF) ILC.

The original proposed ILC is a pure feedforward algorithm, where the controlled plant is supposed stable over the finite time interval and no system dynamics of the current iteration is considered. The idea of this structure is very straightforward, i.e., learning from experience. It is guaranteed that all the tracking errors over the entire finite time interval are converging to zero as the learning number approaches to infinity.

The pure feedforward PID-type ILC has some advantages such as simple structure, easy to implement, no on-line computing burden, and so on. However, there are also some limitations hindering its further applications in practice.

First, although there is a convergence theorem pointing out a scope about the proper learning gains of the PFF PID-ILC, the learning gains are difficult to be selected if there is no enough knowledge known exactly about the controlled system. Second, the learning gains are generally fixed and invariant in the controller operation when they are selected properly. If there is a large disturbance or uncertainty exposed on the controlled plant, the PFF PID-ILC with the previous fixed learning gain may become unsatisfied. Third, the convergence of PFF PID-ILC is guaranteed under the framework of λ norm and requires λ is large enough, which leads to poor tracking performance/even divergence may occur in some time.

In fact, there always exists a generic ideal controller producing the desired control input such that the system output tracks the reference trajectory exactly. The controller is unknown and may be nonlinear or linear. The key issue is how to find such an ideal controller. Recently, model-free adaptive control (MFAC) [4-6] is proposed for nonlinear systems. And the key innovation of MFAC is its dynamical linearization. In [7], the dynamical linearization is also applied to linearize an ideal nonlinear controller. And thus a controller dynamic linearization based MFAC scheme is proposed.

This paper aims to present a general design framework of pure feedforward ILC, as well as its higher order algorithm, for a class of nonlinear system directly. Two dynamical linearization methods for the nonlinear systems and the general controller are proposed. The unknown learning gains in the controller are varying with both time and iterations, and can be updated by the designed estimation algorithms. A higher order algorithm is also proposed by a different dynamical linearization of the desired nonlinear controller. Simulation study shows the efficiency and application of the proposed method.

The rest of this paper is organized as follows. Section 2 presents problem formulation. In Section 3, a new CFDL based ILC controller is designed. Section 4 extends the result to a PFDL based ILC controller. Section 5 provides a simulation study. Finally, some conclusions are given in Section 6.

2 Problem Formulation

Consider a nonlinear discrete-time system,

$$y_k(t+1) = f\big(y_k(t), y_k(t-1), \cdots, y_k(t-n_y), u_k(t), u_k(t-1), \cdots, u_k(t-n_u)\big) \quad (1)$$

*This work was supported by National Science Foundation of China (61374102, 61503209, and 61433002), and the High School Science & Technology Fund Planning Project of Shandong Province of China (J14LN30).

978-1-4673-9714-8/16/$31.00 ©2016 IEEE

where $y_k(t)\in R$ is the system output; $u_k(t)\in R$ is the control input; $f(\cdot)$ is an unknown nonlinear function, which is continuously differentiable; $t\in\{0,1,\cdots,N\}$ denotes sampling time index with N being an integer; k indicates the system repetition number; n_y and n_u are two unknown positive integers.

By solving equation (1), the system output is obtained as,

$$\begin{aligned}
y_k(1) &= f\big(y_k(0),u_k(0)\big) = g_0\big(y_k(0),u_k(0)\big)\\
y_k(2) &= f\big(y_k(1),y_k(0),u_k(1),u_k(0)\big)\\
&= f\big(g_0(y_k(0),u_k(0)),y_k(0),u_k(1),u_k(0)\big)\\
&= g_1\big(y_k(0),u_k(0),u_k(1)\big)\\
y_k(3) &= f\big(y_k(2),y_k(1),y_k(0),u_k(2),u_k(1),u_k(0)\big)\\
&= f\big(g_1(y_k(0),u_k(0),u_k(1)),g_0(y_k(0),u_k(0)),\\
&\qquad y_k(0),u_k(2),u_k(1),u_k(0)\big)\\
&= g_2\big(y_k(0),u_k(0),u_k(1),u_k(2)\big)\\
&\ \ \vdots\\
y_k(t+1) &= f\big(y_k(t),y_k(t-1),\cdots,y_k(t-n_y),\\
&\qquad u_k(t),u_k(t-1),\cdots,u_k(t-n_u)\big)\\
&= f\big(g_{t-1}(y_k(0),u_k(0),\cdots,u_k(t-1)),\\
&\qquad g_{t-2}(y_k(0),u_k(0),\cdots,u_k(t-2)),\cdots,\\
&\qquad g_0(y_k(0),u_k(0)),y_k(0),\\
&\qquad u_k(t),u_k(t-1),\cdots,u_k(0)\big)\\
&= g_t\big(y_k(0),u_k(0),u_k(1),\cdots,u_k(t)\big)
\end{aligned}\tag{2}$$

where $g_t(\cdot)$, $t\in\{0,\cdots,N\}$, is a proper nonlinear function, which is continuously differentiable.

Define a supervector $\boldsymbol{u}_k(t)$ as follows,

$$\boldsymbol{u}_k(t)=[u_k(0),u_k(1),\cdots,u_k(t)]^T\in R^{t+1}.$$

Then, (2) becomes

$$y_k(t+1)=g_t\big(y_k(0),\boldsymbol{u}_k^T(t)\big)\tag{3}$$

In this work, two assumptions are made as stated below.

Assumption 1. The initial state $y_k(0)$ is identical for all iterations, that is $y_k(0)=y_0$, $\forall k$.

Assumption 2. Nonlinear function $g_t(\cdot)$ is globally Lipschitz, that is,

$$\|g_t(x_1,\boldsymbol{u}_1)-g_t(x_2,\boldsymbol{u}_2)\|\le L_x|x_1-x_2|+L_u\|\boldsymbol{u}_1-\boldsymbol{u}_2\|$$

where $L_x<\infty$ and $L_u<\infty$ are two positive Lipschitz constants.

Making a difference between $y_k(t+1)$ and $y_{k-1}(t+1)$ gives,

$$\begin{aligned}
\Delta y_k(t+1) &= y_k(t+1)-y_{k-1}(t+1)\\
&= g_t\big(y_k(0),\boldsymbol{u}_k^T(t)\big)-g_t\big(y_{k-1}(0),\boldsymbol{u}_{k-1}^T(t)\big)
\end{aligned}\tag{4}$$

By virtue of the Differential Mean Value Theorem, one obtains,

$$\begin{aligned}
\Delta y_k(t+1) &= \frac{\partial g_t^*}{\partial y_k(0)}\big(y_k(0)-y_{k-1}(0)\big)\\
&\quad+\frac{\partial g_t^*}{\partial \boldsymbol{u}_k^T(t)}\big(\boldsymbol{u}_k(t)-\boldsymbol{u}_{k-1}(t)\big)
\end{aligned}\tag{5}$$

where $\dfrac{\partial g_t^*}{\partial y_k(0)}$ is the optimal partial derivative values of g with respect to $y_k(0)$ in the interval $[y_k(0),y_{k-1}(0)]$; and $\dfrac{\partial g_t^*}{\partial \boldsymbol{u}_k^T(t)}=\left[\dfrac{\partial g_t^*}{\partial u_k(0)}\ \ \dfrac{\partial g_t^*}{\partial u_k(1)}\ \ \cdots\ \ \dfrac{\partial g_t^*}{\partial u_k(t)}\right]$ denotes the optimal partial derivative values of g to $\boldsymbol{u}_k(t)$ in the interval $[\boldsymbol{u}_k(t),\boldsymbol{u}_{k-1}(t)]$.

From Assumption 1, we can rewrite Eq. (5) as,

$$\Delta y_k(t+1)=\frac{\partial g_t^*}{\partial \boldsymbol{u}_k^T(t)}\big(\boldsymbol{u}_k(t)-\boldsymbol{u}_{k-1}(t)\big)=\boldsymbol{\varphi}_k(t)\Delta\boldsymbol{u}_k(t)\tag{6}$$

where $\boldsymbol{\varphi}_k(t)=[\phi_k(0)\ \ \phi_k(1)\ \ \cdots\ \ \phi_k(t)]=\dfrac{\partial g_t^*}{\partial \boldsymbol{u}_k^T(t)}$ and $\Delta\boldsymbol{u}_k=\boldsymbol{u}_k-\boldsymbol{u}_{k-1}$; where $\phi_k(i)$ is the (i+1)-th element of $\boldsymbol{\varphi}_k(t)$, $i\in\{0,1,\cdots,t\}$. According to A2, we know that $\sup_k \max_{t\in[0,\cdots,N]}\|\boldsymbol{\varphi}_k(t)\|\le L_u$ is bounded.

Then the estimation algorithm is as follows,

$$\begin{aligned}
\hat{\boldsymbol{\varphi}}_k(t) &= \hat{\boldsymbol{\varphi}}_{k-1}(t)\\
&\quad+\frac{\eta\big(\Delta y_{k-1}(t+1)-\hat{\boldsymbol{\varphi}}(t)_{k-1}\Delta\boldsymbol{u}_{k-1}(t)\big)\Delta\boldsymbol{u}_{k-1}^T(t)}{\mu+\|\Delta\boldsymbol{u}_{k-1}(t)\|^2}
\end{aligned}\tag{7}$$

3 Controller Design with CFDL Scheme

For nonlinear system (1), suppose that we have an ideal nonlinear ILC controller with pure feedforward structure, shown as follows,

$$\begin{aligned}
u_k(t)=C\big(&u_{k-1}(t),u_{k-2}(t),\cdots,u_{k-n_u}(t),\\
&e_k(t+1),e_{k-1}(t+1),\cdots,e_{k-n_e}(t+1)\big)
\end{aligned}\tag{8}$$

where $C(\cdot)$ is an unknown nonlinear function, n_u and n_e are unknown positive integers.

Assumption 3. The partial derivatives of $C(\cdot)$ with respect to $e_k(\cdot)$ are continuous.

Assumption 4. $C(\cdot)$ is generalized Lipschitz, that is, $|\Delta u_k(t)|\le b_1|\Delta e_k(t+1)|$ is satisfied for any time instant t and repetition number k and $|\Delta e_k(t+1)|\ne 0$. where $\Delta e_k(t+1)=e_k(t+1)-e_{k-1}(t+1)$, $\Delta u_k(t)=u_k(t)-u_{k-1}(t)$, and b_1 is a positive constant.

Theorem 1 Consider the nonlinear idea controller (8) under assumption 3 and 4, if $|\Delta e_k(t+1)|\ne 0$, then we have a parameter $\theta_k(t)\in\Re$, such that Eq.(8) can be transformed into an equivalent linear controller as follows,

$$\Delta u_k(t)=\theta_k(t)\Delta e_k(t+1)\tag{9}$$

with $|\theta_k(t)|\le b_1$.

Proof: From (8), it has

$$
\begin{aligned}
\Delta u_k(t) &= u_k(t) - u_{k-1}(t) \\
&= C\big(u_{k-1}(t), u_{k-2}(t), \cdots, u_{k-n_u}(t), \\
&\quad e_k(t+1), e_{k-1}(t+1), \cdots, e_{k-n_e}(t+1)\big) \\
&\quad - C\big(u_{k-1}(t), u_{k-2}(t), \cdots, u_{k-n_u}(t), \\
&\quad e_{k-1}(t+1), e_{k-1}(t+1), \cdots, e_{k-n_e}(t+1)\big) \\
&\quad + C\big(u_{k-1}(t), u_{k-2}(t), \cdots, u_{k-n_u}(t), \\
&\quad e_{k-1}(t+1), e_{k-1}(t+1), \cdots, e_{k-n_e}(t+1)\big) \\
&\quad - C\big(u_{k-2}(t), u_{k-3}(t), \cdots, u_{k-n_u-1}(t), \\
&\quad e_{k-1}(t+1), e_{k-2}(t+1), \cdots, e_{k-n_e-1}(t+1)\big)
\end{aligned} \tag{10}
$$

In terms of the Differential Mean Value Theorem, (10) can be rewritten follows

$$\Delta u_k(t) = \frac{\partial C^*}{\partial e_k(t+1)} \Delta e_k(t+1) + \zeta_k(t) \tag{11}$$

where $\frac{\partial C^*}{\partial e_k(t+1)}$ is the optimal partial derivative value of $C(\cdot)$ with respect to tracking error $e_k(t+1)$ at a certain mean point within the interval $[e_k(t+1), e_{k-1}(t+1)]$, and

$$
\begin{aligned}
\zeta_k(t) = C\big(&u_{k-1}(t), u_{k-2}(t), \cdots, u_{k-n_u}(t), \\
&e_{k-1}(t+1), e_{k-1}(t+1), \cdots, e_{k-n_e}(t+1)\big) \\
- C\big(&u_{k-2}(t), u_{k-3}(t), \cdots, u_{k-n_u-1}(t), \\
&e_{k-1}(t+1), e_{k-2}(t+1), \cdots, e_{k-n_e-1}(t+1)\big)
\end{aligned} .
$$

Given a scalar equation with $h_k(t)$ as a variable,

$$\zeta_k(t) = h_k(t) \Delta e_k(t+1) \tag{12}$$

Since the condition of $|\Delta e_k(t+1)| \neq 0$ holds, (12) has one solution $h_k^*(t)$ for each fixed k.

Let

$$\theta_k(t) = h_k^*(t) + \frac{\partial C^*}{\partial e_k(t+1)} \tag{13}$$

then we have $\Delta u_k(t) = \theta_k(t) \Delta e_k(t+1)$ according to (11). Obviously, $|\theta_k(t)| \leq b_1$ holds by virtue of Assumption 4.

Since our purpose is to find a proper input $u_k(t)$ such that $e_k(t+1) = 0$, then the above control law becomes,

$$\Delta u_k(t) = -\theta_k(t) e_{k-1}(t+1) \tag{14}$$

where the unknown parameter $\theta_k(t)$ is estimated by an iterative updating law.

Define a criterion function as follows,

$$J(\theta_k(t)) = |y_d(t+1) - y_k(t+1)|^2 + \lambda_1 \left|\theta_k(t) - \hat{\theta}_{k-1}(t)\right|^2 \tag{15}$$

where $\lambda_1 > 0$ denotes a weighting factor.

According to (6), we have

$$y_k(t+1) = y_{k-1}(t+1) + \boldsymbol{\varphi}_k(t) \Delta \boldsymbol{u}_k(t) \tag{16}$$

Substituting (16) into (15), yields,

$$
\begin{aligned}
J(\theta_k(t)) = &\left| y_d(t+1) - y_{k-1}(t+1) - \sum_{i=0}^{t-1} \phi_k(i) \Delta u_k(i) - \phi_k(t) \Delta u_k(t) \right|^2 \\
&+ \lambda_1 \left|\theta_k(t) - \hat{\theta}_{k-1}(t)\right|^2
\end{aligned} \tag{17}
$$

where $u_k(i)$ are the (i+1)-th element of $\boldsymbol{u}_k(t)$; $\Delta u_k(i) = u_k(i) - u_{k-1}(i)$, $i = 0, 1, \cdots, t$.

Substituting (14) into (17), yields,

$$
\begin{aligned}
J(\theta_k(t)) = &\left| e_{k-1}(t+1) - \sum_{i=0}^{t-1} \phi_k(i) \Delta u_k(i) + \phi_k(t) \theta_k(t) e_{k-1}(t) \right|^2 \\
&+ \lambda_1 \left|\theta_k(t) - \hat{\theta}_{k-1}(t)\right|^2
\end{aligned} \tag{18}
$$

Let us differentiate (18) with respect to $\theta_k(t)$ and set it to zero, then we have

$$
\begin{aligned}
\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) &- \frac{\rho_1 \phi_k(t) (e_{k-1}(t+1))^2}{\lambda_1 + |\phi_k(t)|^2 |e_{k-1}(t+1)|^2} \\
&+ \frac{\rho_1 \phi_k(t) \sum_{i=0}^{t-1} \phi_k(i) \Delta u_k(i) e_{k-1}(t+1)}{\lambda_1 + |\phi_k(t)|^2 |e_{k-1}(t+1)|^2} \\
&- \frac{\rho_1 (\phi_k(t))^2 \hat{\theta}_{k-1}(t) (e_{k-1}(t+1))^2}{\lambda_1 + |\phi_k(t)|^2 |e_{k-1}(t+1)|^2}
\end{aligned} \tag{19}
$$

where $\rho_1 > 0$, denotes the step-size constant. The unknown parameter $\phi_k(t)$ is estimated by the updating law proposed in [8].

So, the proposed CFDL-ILC approach is constructed as follows:

$$
\begin{aligned}
\hat{\boldsymbol{\varphi}}_k(t) = &\hat{\boldsymbol{\varphi}}_{k-1}(t) \\
&+ \frac{\eta(\Delta y_{k-1}(t+1) - \hat{\boldsymbol{\varphi}}(t)_{k-1} \Delta \boldsymbol{u}_{k-1}(t)) \Delta \boldsymbol{u}_{k-1}^T(t)}{\mu + \|\Delta \boldsymbol{u}_{k-1}(t)\|^2}
\end{aligned} \tag{20}
$$

$$
\begin{aligned}
&\hat{\boldsymbol{\varphi}}_k(t) = \hat{\boldsymbol{\varphi}}_0(t), \text{ if } \operatorname{sgn}(\hat{\phi}_k(i)) \neq \operatorname{sgn}(\hat{\phi}_0(i)) \\
&\text{or } \|\hat{\boldsymbol{\varphi}}_k(t)\| \leq \varepsilon, \ i = 0, 1, \cdots, t
\end{aligned} \tag{21}
$$

$$
\begin{aligned}
\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) &- \frac{\rho_1 \hat{\phi}_k(t) (e_{k-1}(t+1))^2}{\lambda_1 + |\hat{\phi}_k(t)|^2 |e_{k-1}(t+1)|^2} \\
&+ \frac{\rho_1 \hat{\phi}_k(t) \sum_{i=0}^{t-1} \hat{\phi}_k(i) \Delta u_k(i) e_{k-1}(t+1)}{\lambda_1 + |\hat{\phi}_k(t)|^2 |e_{k-1}(t+1)|^2} \\
&- \frac{\rho_1 (\hat{\phi}_k(t))^2 \hat{\theta}_{k-1}(t) (e_{k-1}(t+1))^2}{\lambda_1 + |\hat{\phi}_k(t)|^2 |e_{k-1}(t+1)|^2}
\end{aligned} \tag{22}
$$

$$\hat{\theta}_k(t) = -b_1, \text{ if } \hat{\theta}_k(t) < -b_1 \text{ or } \hat{\theta}_k(t) > -\bar{b}_1 \tag{23}$$

$$\Delta u_k(t) = -\hat{\theta}_k(t) e_{k-1}(t+1) \tag{24}$$

where (21) and (23) are two reset algorithm, which are added to enhance the tracking performance of $\hat{\boldsymbol{\varphi}}_k(t)$ to the iteration-time-varying parameter $\boldsymbol{\varphi}_k(t)$ and $\hat{\theta}_k(t)$ to the iteration-time-varying parameter $\theta_k(t)$; ε, b_1 and $\bar{b}_1$ are three small positive scalar.

4 Controller Design with PFDL Scheme

For nonlinear system (1), we can obtain the ILC law as follows

$$\Delta u_k(t)=\boldsymbol{\theta}_k(t)\Delta \boldsymbol{e}_k(t+1) \quad (25)$$

where $\boldsymbol{\theta}_k(t)\in R^{1\times l_e}$ is a parameter vector, and $\Delta \boldsymbol{e}_k(t+1)=\left[\Delta e_k(t+1) \quad \Delta e_{k-1}(t+1) \quad \cdots \quad \Delta e_{k-l_e+1}(t+1)\right]^T$.

When By virtue of Remark 1,

$$\Delta \boldsymbol{e}_k(t+1)=\left[-e_{k-1}(t+1) \quad \Delta e_{k-1}(t+1) \quad \cdots \quad \Delta e_{k-l_e+1}(t+1)\right]^T \quad (26)$$

Consider a different cost function as follows,

$$J(\boldsymbol{\theta}_k(t))=\left|y_d(t+1)-y_k(t+1)\right|^2+\lambda_2\left\|\boldsymbol{\theta}_k(t)-\hat{\boldsymbol{\theta}}_{k-1}(t)\right\|^2 \quad (27)$$

where $\lambda_2>0$ denotes a weighting factor.

Substituting (16) into (27), yields,

$$J(\boldsymbol{\theta}_k(t))=\left|y_d(t+1)-y_{k-1}(t+1)-\sum_{i=0}^{t-1}\phi_k(i)\Delta u_k(i)-\phi_k(t)\Delta u_k(t)\right|^2 + \lambda_2\left\|\boldsymbol{\theta}_k(t)-\hat{\boldsymbol{\theta}}_{k-1}(t)\right\|^2 \quad (28)$$

Substituting (25) into (28), yields,

$$J(\boldsymbol{\theta}_k(t))=\left|e_{k-1}(t+1)-\sum_{i=0}^{t-1}\phi_k(i)\Delta u_k(i)-\phi_k(t)\boldsymbol{\theta}_k(t)\Delta \boldsymbol{e}_k(t)\right|^2 + \lambda_2\left\|\boldsymbol{\theta}_k(t)-\hat{\boldsymbol{\theta}}_{k-1}(t)\right\|^2 \quad (29)$$

Let us differentiate (29) with respect to $\boldsymbol{\theta}_k(t)$ and set it to zero, then we have

$$\begin{aligned}\hat{\boldsymbol{\theta}}_k(t)&=\hat{\boldsymbol{\theta}}_{k-1}(t)\\&+\frac{\rho_2\phi_k(t)\Delta \boldsymbol{e}_k^T(t+1)e_{k-1}(t+1)}{\lambda_2+\left|\phi_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\\&-\frac{\rho_2\phi_k(t)\sum_{i=0}^{t-1}\phi_k(i)\Delta u_k(i)\Delta \boldsymbol{e}_k^T(t+1)}{\lambda_2+\left|\phi_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\\&-\frac{\rho_2\phi_k(t)^2\Delta \boldsymbol{e}_k^T(t+1)\hat{\boldsymbol{\theta}}_{k-1}(t)\Delta \boldsymbol{e}_k(t+1)}{\lambda_2+\left|\phi_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\end{aligned} \quad (30)$$

where $\rho_2>0$, denotes the step-size constant.

So, the proposed PFDL-ILC approach is constructed as follows:

$$\hat{\boldsymbol{\varphi}}_k(t)=\hat{\boldsymbol{\varphi}}_{k-1}(t)+\frac{\eta\left(\Delta y_{k-1}(t+1)-\hat{\boldsymbol{\varphi}}(t)_{k-1}\Delta \boldsymbol{u}_{k-1}(t)\right)\Delta \boldsymbol{u}_{k-1}^T(t)}{\mu+\left\|\Delta \boldsymbol{u}_{k-1}(t)\right\|^2} \quad (31)$$

$$\hat{\boldsymbol{\varphi}}_k(t)=\hat{\boldsymbol{\varphi}}_0(t),\ \text{if}\ \operatorname{sgn}\left(\hat{\phi}_k(i)\right)\neq\operatorname{sgn}\left(\hat{\phi}_0(i)\right)\ \text{or}\ \left\|\hat{\boldsymbol{\varphi}}_k(t)\right\|\le\varepsilon,\ i=0,1,\cdots,t \quad (32)$$

$$\begin{aligned}\hat{\boldsymbol{\theta}}_k(t)&=\hat{\boldsymbol{\theta}}_{k-1}(t)\\&+\frac{\rho_2\hat{\phi}_k(t)\Delta \boldsymbol{e}_k^T(t+1)e_{k-1}(t+1)}{\lambda_2+\left|\hat{\phi}_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\\&-\frac{\rho_2\hat{\phi}_k(t)\sum_{i=0}^{t-1}\phi_k(i)\Delta u_k(i)\Delta \boldsymbol{e}_k^T(t+1)}{\lambda_2+\left|\hat{\phi}_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\\&-\frac{\rho_2\hat{\phi}_k(t)^2\Delta \boldsymbol{e}_k^T(t+1)\hat{\boldsymbol{\theta}}_{k-1}(t)\Delta \boldsymbol{e}_k(t+1)}{\lambda_2+\left|\hat{\phi}_k(t)\right|^2\Delta \boldsymbol{e}_k^T(t+1)\Delta \boldsymbol{e}_k(t+1)}\end{aligned} \quad (33)$$

$$\left\|\hat{\boldsymbol{\theta}}_k(t)\right\|=b_1,\ \text{if}\ \left\|\hat{\boldsymbol{\theta}}_k(t)\right\|<b_1\ \text{or}\ \left\|\hat{\boldsymbol{\theta}}_k(t)\right\|>\bar{b}_1 \quad (34)$$

$$\Delta u_k(t)=\hat{\boldsymbol{\theta}}_k(t)\Delta \boldsymbol{e}_k(t+1) \quad (35)$$

where ε, b_1 and $\bar{b}_1$ are three small positive scalar.

5 Simulations

Consider a nonlinear batch reactor as follows

$$\begin{cases}\dot{x}_1=-k_1\exp\left(\dfrac{-E_1}{uT_{ref}}\right)x_1^2\\ \dot{x}_2=-k_1\exp\left(\dfrac{-E_1}{uT_{ref}}\right)x_1^2-k_2\exp\left(\dfrac{-E_2}{uT_{ref}}\right)x_2\\ y=x_2\end{cases}$$

where x_1, x_2 represent the dimensionless concentrations of the raw material and the product; $u=T/T_{ref}$; T_{ref} is the reference temperature; T is the reactor temperature and $298\text{K}\le T\le 398\text{K}$; the operation interval is 1.0 hour. In the simulation, the sampling time $h=0.1hour$.

The values of k_1, k_2, E_1, E_2, and T_{ref} are given as $k_1=4.0\times10^3$, $k_2=6.2\times10^5$, $E_1=2.5\times10^3\text{K}$, $E_2=5.0\times10^3\text{K}$, and $T_{ref}=348\text{K}$.

The desired product reference trajectory is $y_d(t)=0.61\sin(\pi t/20)$, $t\in\{0,1,\cdots,10\}$.

The initial states are selected as $x_1(0)=1$ and $x_2(0)=0$.

In the simulation, both the CFDL-ILC and the PFDL-ILC approaches are applied with the same controller parameters: $\rho_1=\rho_2=0.001$, $\eta=0.005$, $\lambda_1=\lambda_2=0.001$, $\mu=0.1$. The simulation results are shown in Fig. 1. The blue line is the profile of tracking error with iterations using CFDL-ILC, and the red and black lines are obtained by using PFDL-ILC with *le*=2 and *le*=3, respectively.

It is seen from the figure 1 that both the CFDL-ILC and the PFDL-ILC are able to guarantee the convergence of tracking error. However, by using more information of previous errors, the PFDL-ILC achieves a better performance than the CFDL-ILC.

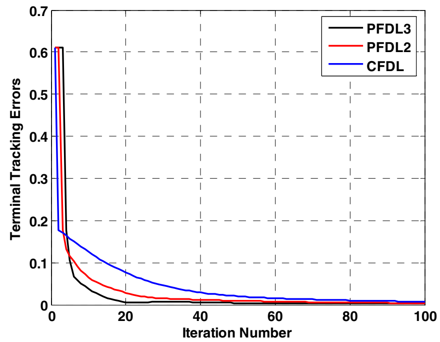

Fig. 1. The profile of tracking errors

6 Conclusions

A general design framework of ILC with pure feedforward structure is proposed in this work. The results are directly derived from nonlinear systems. By different dynamical linearization method for the desired nonlinear controller, a CFDL- and a PFDL- based ILC approaches are designed. The latter is similar to the higher order ILC scheme. Simulation results are provided to show the effectiveness of the proposed approaches.

References

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

[2] J.-X. Xu and Y. Tan, Linear and Nonlinear Iterative Learning Control, Springer-Verlag Berlin Heidelberg, 2003.

[3] D. A. Bristow, M. Tharayil, and A. G. Alleyne, A survey of iterative learning control, IEEE Control Syst. Mag., 2006, 26, (3), pp. 96 – 114

[4] Z. S. Hou and S. T. Jin, Model Free Adaptive Control: Theory and Applications. CRC press, 2013.

[5] Z. S. Hou, The parameter identification, adaptive control and model free learning adaptive control for nonlinear systems. PhD dissertation. Shenyang, China: Northeastern University, 1994.

[6] Z. S. Hou and S. T. Jin, A novel data-driven control approach for a class of discrete-time nonlinear systems. IEEE Transactions on Control Systems Technology, vol. 19, no. 6, pp. 1549-1558, 2011.

[7] Z. S. Hou and Y. M. Zhu, Controller-dynamic-linearization-based model free adaptive control for discrete-time nonlinear systems, IEEE Transactions on Industrial Informatics, vol. 9, no. 4, pp. 2301-2309, 2013.

[8] R. H. Chi, Z. S. Hou, B. Huang, S. T. Jin, A unified data-driven design framework of optimality-based general iterative learning control, Computers & Chemical Engineering, vol.77, pp. 10-23, 2015.