A Data-Based State Feedback Control Method for a Class of Nonlinear Systems

Zhuo Wang and Derong Liu, *Fellow, IEEE*

Abstract—In this paper, a data-based state feedback control method is developed for a class of nonlinear systems. It is a real-time control method, which requires little prior knowledge about the system dynamics, and does not need to know or to build the mathematical model of the system. We apply a fast sampling technique to sample the state signal, which contains useful information of the system. The zero-order hold (ZOH) and the control switch are also used to obtain system information. The feedback gain matrix is calculated and adjusted according to these sampled data. Theoretical analysis on the convergence and simulation results demonstrate the feasibility of this data-based control method.

Index Terms—Control switch, data-based state feedback control, fast sampling technique, nonlinear systems, real-time control, sampled state data, zero-order hold.

I. INTRODUCTION

AS the development of information sciences and technologies, modern industries have undergone great changes in the past few decades. Industrial systems have become more and more complex. Many of them, for instance, chemical industrial systems, electrical power systems, and metallurgical systems, etc., have strongly coupled characteristics, uncertain boundary conditions, and adverse operating environments [1]. In these cases, engineers often encounter great difficulties in system modeling based on physical and chemical mechanisms [2], [3]. Sometimes, it is even impossible to analytically build their mathematical models. As a result, using the traditional model-based methods to deal with all the control issues in these systems would become impractical.

With the development and extensive applications of digital sensor technologies, these industrial systems are producing vast amounts of data every day, which contain important information about the system [4], [5]. Then, how to effectively utilize these measured data, both on-line and off-line, to directly control the industrial systems, would have great significance. Consequently, the establishment and development of data-based control theory and methods is an urgent issue not only in theories but also in applications [1], [4], [6].

Up until now, some typical data-based control methods have been developed, for example: iterative feedback tuning (IFT), iterative learning control (ILC), model free adaptive control (MFAC), simultaneous perturbation stochastic approximation (SPSA), and virtual reference feedback tuning (VRFT), etc. We can divide them into two broad categories: the first one needs to perform system identification and establish the approximate models with the measured data first, using approaches such as neural networks (NN) [7]–[9] [10] and support vector machines (SVM) [11]–[13], and then designs the controllers according to these approximate models; the other one could directly control the system with the measured data, where no model identification of the plant is needed, but it is subject to some kind of restrictions or prior assumptions.

The second category of the data-based control methods has some typical examples: ILC [14]–[16], MFAC [17], [18], and VRFT [19], [20]. For ILC methods, they involve two basic assumptions: the global Lipschitz condition and the strict repeatability of system dynamical behaviors and the control task. MFAC methods employ a dynamic linearization technique with the concept called pseudo-partial derivative. The dynamic behaviors of the nonlinear systems suitable for MFAC methods are assumed to have continuous partial derivatives with respect to control inputs, and satisfy some generalized Lipschitz conditions. For VRFT methods, the control schemes are restricted by some optimality criteria.

In this paper, we develop a data-based state feedback control method for a class of nonlinear systems, which have unknown mathematical models. To obtain the corresponding Jacobian matrices, we employ a fast sampling technique to measure the state signal. The control switch and ZOH are also applied for the same purpose. To ensure that the system state converges to its desired value, the feedback gain matrix is calculated and adjusted in real-time with measured data.

This method belongs to the second category of the data-based control methods. It is a real-time control method, which only needs some basic prior knowledge about the system, while requiring neither repeatability of the system behaviors, nor conditions that satisfy any optimality criteria.

II. SYSTEM DESCRIPTION AND BASIC ASSUMPTIONS

In this paper, we study a class of nonlinear systems, which can be described in the following form:

$$\dot{x}(t) = g\left(x(t), u(t)\right) \quad (t \geq 0) \tag{1}$$

where $u(t) \in D_u \subset R^m$, and $x(t) \in D_x \subset R^n$ are the input and the state of system (1), respectively, $m \leq n$, $x = 0 \in D_x$, and $u = 0 \in D_u$. The explicit expression of $g\left(x(t), u(t)\right)$ is

unknown and $g(0,0) \equiv 0$. $u(t)$ is piecewise continuous in t. The values of $u(t)$ and $x(t)$ can be measured and stored, and we can set the value of $x(0)$.

Since in modern control engineering, computer technology is often used in control systems, we sample the input and state signals of system (1) to obtain digital data for numerical computation. Suppose that the sampled input and state data have the following relationship:

$$x\left((k+1)T\right) = f\left(x(kT), u(kT)\right) \quad (k \geq 0),$$

where $T > 0$ is the sampling period, $f(x(kT), u(kT))$ is also unknown, and $f(0,0) \equiv 0$. For abbreviation, we use

$$x(k + 1) = f\left(x(k), u(k)\right) \quad (k \geq 0) \tag{2}$$

instead of the above system.

In order to control system (1), a digital-to-analog converter is needed at the input terminal to convert the digital signals back to continuous-time signals. We employ the zero-order hold (ZOH) for this purpose. Furthermore, we have the following assumption for system (2).

Assumption 1: Function $f(x(k), u(k))$ given in (2) has the following properties.
  1) It is $\mathcal{C}^1$ in both $x(k)$ and $u(k)$, while its Jacobian matrix with respect to $u(k)$ has full column rank.
  2) $\forall u(k) \in D_u \ (k \geq 0)$ and $x(0) \in D_x$,

  $$M_1 \|u(k)\| \leq \|x(k)\| \leq M_2 \|u(k)\| + M_3 \|x(0)\|,$$

  where $0 < M_1, M_2, M_3 < \infty$ and $M_1 < M_2$, and $\| \cdot \|$ represents the Euclidean norm.

For a practical control system, its dynamics is either described by a single smooth function, or by a piecewise smooth function. If there is at least one component in the control system, which has intrinsic nonlinear characteristics, such as dead zone, backlash, jumping, and amplitude limiting, etc., such that its dynamics can only be described by a piecewise smooth function, then we can consider this control system unsmooth; otherwise, we can consider it smooth. In practice, we may verify whether a system has intrinsic nonlinear components by knowledge or by experiment.

By Assumption 1 and Lagrange's mean value theorem, for any $x(k), \bar{x}(k) \in D_x$ and $u(k), \bar{u}(k) \in D_u$, where the two pairs $(x(k), u(k))$ and $(\bar{x}(k), \bar{u}(k))$ satisfy (2), respectively, we have

$$x(k + 1) - \bar{x}(k + 1) = f\left(x(k), u(k)\right) - f\left(\bar{x}(k), \bar{u}(k)\right)$$
$$= \tilde{F}(k)\left[z(k) - \bar{z}(k)\right]. \tag{3}$$

In (3), we denote

$$\begin{cases} z(k) = [x_1(k), \ldots, x_n(k), u_1(k), \ldots, u_m(k)]^T \\ \quad = \left[x^T(k), u^T(k)\right]^T \\ \bar{z}(k) = [\bar{x}_1(k), \ldots, \bar{x}_n(k), \bar{u}_1(k), \ldots, \bar{u}_m(k)]^T \\ \quad = \left[\bar{x}^T(k), \bar{u}^T(k)\right]^T \\ \tilde{F}(k) = \left[\tilde{F}_x(k), \tilde{F}_u(k)\right], \end{cases} \tag{4}$$

where

$$\begin{cases} \tilde{F}_x(k) = \left[\frac{\partial f_1}{\partial x}(k)^T, \frac{\partial f_2}{\partial x}(k)^T, \ldots, \frac{\partial f_n}{\partial x}(k)^T\right]^T \\ \frac{\partial f_i}{\partial x}(k) = \left[\frac{\partial f_i(\varphi_i(k))}{\partial x_1}, \ldots, \frac{\partial f_i(\varphi_i(k))}{\partial x_n}\right] \\ \tilde{F}_u(k) = \left[\frac{\partial f_1}{\partial u}(k)^T, \frac{\partial f_2}{\partial u}(k)^T, \ldots, \frac{\partial f_n}{\partial u}(k)^T\right]^T \\ \frac{\partial f_i}{\partial u}(k) = \left[\frac{\partial f_i(\varphi_i(k))}{\partial u_1}, \ldots, \frac{\partial f_i(\varphi_i(k))}{\partial u_m}\right] \\ \varphi_i(k) = \bar{z}(k) + \theta_i(k)\left[z(k) - \bar{z}(k)\right] \\ 0 < \theta_i(k) < 1, 1 \leq i \leq n. \end{cases} \tag{5}$$

Obviously, $\tilde{F}_x(k) \in R^{n \times n}$ and $\tilde{F}_u(k) \in R^{n \times m}$. Thus, (3) can be re-written as

$$x(k+1) - \bar{x}(k+1) = \tilde{F}_x(k)[x(k) - \bar{x}(k)] + \tilde{F}_u(k)[u(k) - \bar{u}(k)]. \tag{6}$$

Let a constant vector $x^* \in D_x$ denote the desired state, where $0 < \|x^*\| < \infty$. By Assumption 1, there exists a control input $u^* \in D_u$, which is also a constant vector, such that x^* and u^* satisfy

$$x^* = f(x^*, u^*), \quad 0 < \|u^*\| < \infty. \tag{7}$$

However, we cannot analytically obtain u^* since the mathematical model of system (2) is unknown. In this situation, our objective is to find a method for given $x(0)$ and x^*, which can adjust $u(k)$ in real time, to make $x(k) \to x^*$ (or, $\|x(k) - x^*\| \leq \varepsilon$, where $\varepsilon > 0$ is the maximum tolerable error) as $k \to \infty$.

III. DATA-BASED FEEDBACK CONTROL

In this section, we design the control law for system (2) as

$$u(k) = u^* + B(k)\left[x(k) - x^*\right] \tag{8}$$

where $B(k) \in R^{m \times n}$ is the feedback gain matrix, which will be adjusted at each time kT. Similar to (6), and by (8), we shall have

$$x(k+1) - x^* = F_x(k)\left[x(k) - x^*\right] + F_u(k)B(k)\left[x(k) - x^*\right]$$
$$= [F_x(k) + F_u(k)B(k)]\left[x(k) - x^*\right] \tag{9}$$

where

$$F_x(k) = \tilde{F}_x(k)|_{\bar{x}(k)=x^*, \ \bar{u}(k)=u^*},$$
$$F_u(k) = \tilde{F}_u(k)|_{\bar{x}(k)=x^*, \ \bar{u}(k)=u^*}.$$

Let

$$e(k) = x(k) - x^* \quad (k \geq 0) \tag{10}$$

and define

$$A(k) = F_x(k) + F_u(k)B(k). \tag{11}$$

Therefore, $A(k) \in R^{n \times n}$ and (9) becomes

$$e(k + 1) = A(k)e(k). \tag{12}$$

As a result, the problem of controlling system (2) is transformed into the problem of how to calculate and adjust $B(k)$, to make $\|A(k)\| < 1$ for all $k \geq 0$. Therefore, $\|e(k)\|$ will keep on decreasing until it is less than ε. The calculation of $B(k)$ becomes

Fig. 1. Block diagram of the data-based state feedback control using fast sampling, control switch, and ZOH.

the key to solve the present control problem, and should be completed before time kT. Otherwise, we may not adjust $B(k)$ and control the system on time.

From (11), we can see that in order to obtain $B(k)$, one needs to determine $A(k)$, $F_x(k)$ and $F_u(k)$ first. However, we cannot analytically compute them due to the unknown mathematical model. To overcome this difficulty, we need to estimate these matrices directly from the measured state data. To ensure the feasibility of this data-based method, we make the following assumption.

Assumption 2: During each $[kT, (k+1)T)$ $(k \geq 0)$, $A(k)$, $F_x(k)$, and $F_u(k)$ can be considered constant matrices.

For a nonlinear autonomous system, people used to employ a linear time-invariant system to approximate it in a neighborhood of the equilibrium point. Obviously, this approximation method can also be extended to other neighborhoods, which do not contain the equilibrium point. Assumption 2 is proposed under this consideration.

On the other hand, Assumption 2 indicates that the system is slowly time-varying. But, the system changing rate (fast or slow) is just a relative concept comparing with the value of the sampling period T. With the development of technologies, the speed of computers and the sampling frequency of samplers can be very fast. In such case, those systems, whose dynamics used to be regarded as fast, can now be controlled by using our method.

Under Assumption 2, (12) provides a piecewise linearization method to deal with system (2). It should be noted that $A(k)$, $F_x(k)$, and $F_u(k)$ have different values for different $k \geq 0$, although they are considered constant matrices during the same time interval $[kT, (k+1)T)$. Then, these matrices will be estimated by using the following data-based method.

We introduce a fast sampling technique to collect information about the system. It samples the state signal as measured data with a higher frequency during each $[kT, (k+1)T)$, and stores these measured data. The new sampling period is τ $(\tau = T/N,$ where $N > 4n$ is an integer. $4n\tau + t_c \leq T$, where t_c is the time spent on computing $A(k)$, $F_x(k)$, $F_u(k)$, and $B(k)$). It should be pointed out that the time interval between two successive adjustments of $B(k)$ is still T.

Furthermore, we put a control switch between $B(k)$ and the ZOH (see Fig. 1). This control switch uses a counter with the period of τ to control its opening and closing. When a sample arrives at the control switch at time kT, the counter begins to count the number of samples. The first sample is counted at kT. After counting $2n$ samples, turn off the control switch. Then, $B(k)$ will be disconnected from the ZOH, while the counter is still counting samples. In the mean time, the ZOH keeps $B(k)[x(kT + (2n-1)\tau) - x^*]$ until the number of counted samples becomes $(4n + 1)$ at time $(kT + 4n\tau)$. Then, turn on the control switch to make $B(k)$ connected to the ZOH again. The counter keeps working until the Nth sample is counted at time $(kT + (N-1)\tau)$, and then reset the counter to zero. We can repeat the above procedure until time $(k+1)T$.

In the above process, measure and store the values of $x(kT)$, $x(kT + \tau), \ldots, x(kT + 4n\tau)$. With these data, we can calculate and adjust the feedback gain matrix $B(k)$ in real-time.

Since τ is very small as compared with T, then by Assumption 2 and (12), we shall have

$$\begin{cases} e(kT + \tau) = A(k)e(kT) \\ e(kT + 2\tau) = A(k)e(kT + \tau) \\ \quad \vdots \\ e(kT + 2n\tau) = A(k)e(kT + (2n-1)\tau). \end{cases} \quad (13)$$

Define

$$\begin{cases} E_0(k) = [e(kT), \ldots, e(kT + (2n-1)\tau)] \\ E_\tau(k) = [e(kT + \tau), \ldots, e(kT + 2n\tau)]. \end{cases} \quad (14)$$

From (13) and (14), we can obtain

$$E_\tau(k) = A(k)E_0(k). \quad (15)$$

As introduced above, after time $(kT + (2n-1)\tau)$, the value of $B(k)[x(kT + (2n-1)\tau) - x^*]$ will be kept by the ZOH until time $(kT + 4n\tau)$. By Assumption 2, $F_x(k)$ and $F_u(k)$ are constant matrices during this time interval. In addition, $B(k)$ will not be adjusted before time $(k+1)T$. Therefore,

$$F_u(k)B(k)[x(kT + j\tau) - x^*]$$
$$= F_u(k)B(k)[x(kT + (2n-1)\tau) - x^*], \quad (16)$$

where $2n \leq j \leq 4n$. By (9) and (16), we can obtain

$$e(kT + (l+2)\tau) - e(kT + (l+1)\tau)$$
$$= [x(kT + (l+2)\tau) - x^*] - [x(kT + (l+1)\tau) - x^*]$$

$$\begin{aligned}
&= F_x(k)\left[x\left(kT+(l+1)\tau\right)-x^*\right]+F_u(k)B(k)\\
&\quad \times\left[x\left(kT+(2n-1)\tau\right)-x^*\right]-F_x(k)\left[x(kT+l\tau)-x^*\right]\\
&\quad -F_u(k)B(k)\left[x\left(kT+(2n-1)\tau\right)-x^*\right]\\
&= F_x(k)\left[x\left(kT+(l+1)\tau\right)-x(kT+l\tau)\right]
\end{aligned} \tag{17}$$

where $(2n-1)\leq l\leq(4n-2)$. Define

$$\begin{aligned}
X_0(k) &= \{[x(kT+2n\tau)-x\left(kT+(2n-1)\tau\right)],\ldots,\\
&\qquad [x\left(kT+(4n-1)\tau\right)-x\left(kT+(4n-2)\tau\right)]\}\\
X_\tau(k) &= \{[x\left(kT+(2n+1)\tau\right)-x(kT+2n\tau)],\ldots,\\
&\qquad [x(kT+4n\tau)-x\left(kT+(4n-1)\tau\right)]\}. \tag{18}
\end{aligned}$$

From (17) and (18), we have

$$X_\tau(k)=F_x(k)X_0(k). \tag{19}$$

To calculate $A(k)$ and $F_x(k)$ with $E_0(k)$ and $X_0(k)$, respectively, we make another assumption.

Assumption 3: For each $k\geq 0$, there exist two integers p_k and q_k, where $(n-1)\leq p_k,q_k<(2n-1)$. Such that

$$\begin{aligned}
&\text{Rank}\{[e(kT),e(kT+\tau),\ldots,e(kT+p_k\tau)]\}=n\\
&\text{Rank}\{[x(kT+2n\tau)-x\left(kT+(2n-1)\tau\right)],\ldots,\\
&\quad [x\left(kT+(2n+q_k)\tau\right)-x\left(kT+(2n+q_k-1)\tau\right)]\}=n. \tag{20}
\end{aligned}$$

Lemma 1 ([22, Chap. 3]): Let the matrix D be partitioned as follows

$$D=[D_1|D_2]. \tag{21}$$

We have the inequality

$$\text{Rank}[D_j]\leq\text{Rank}[D]\leq\text{Rank}[D_1]+\text{Rank}[D_2]\ (j=1,2). \tag{22}$$

Then, by Assumption 3 and Lemma 1, we can obtain

$$\forall k\geq 0,\ \text{Rank}[E_0(k)]=\text{Rank}[X_0(k)]=n. \tag{23}$$

Since

$$\begin{aligned}
\text{Rank}\left[E_0(k)E_0^T(k)\right]&=\text{Rank}[E_0(k)]\\
\text{Rank}\left[X_0(k)X_0^T(k)\right]&=\text{Rank}[X_0(k)],
\end{aligned}$$

$[E_0(k)E_0^T(k)]$ and $[X_0(k)X_0^T(k)]$ are both invertible. Then, $A(k)$ and $F_x(k)$ $(k\geq 0)$ can be computed as follows

$$\begin{aligned}
A(k)&=E_\tau(k)E_0^T(k)\left[E_0(k)E_0^T(k)\right]^{-1}\\
F_x(k)&=X_\tau(k)X_0^T(k)\left[X_0(k)X_0^T(k)\right]^{-1}. \tag{24}
\end{aligned}$$

Remark 1: When calculating $A(k)$ and $F_x(k)$, there are some points for attention.

1) From (14) and (18), we can see that each time interval $[kT,(k+1)T)$ is divided into two parts. The first part is $[kT,kT+4n\tau]$, during which the state data are measured. The second part is $(kT+4n\tau,(k+1)T)$, during which the corresponding matrices are calculated. In practice, if the computation time is adequate, then the larger portion of the first part is, the more accurately $A(k)$ and $F_x(k)$ will

be estimated. Therefore, when T is fixed, one should use a τ as large as possible.

2) Observe (16) and (17), the control switch and the ZOH are indispensable for our method. This is because we need to eliminate the effect caused by $F_u(k)$ and $B(k)$, which are contained in the feedbacks. Thus, we can compute $A(k)$ and $F_x(k)$, separately.

Since $B(k)$ is an $m\times n$ matrix, comparing m with n, there are two cases to be studied.

Case 1 m < n:

Step 1 Set Initial Values, When $k=0$, 1:

In the initial stage, we have not collected sufficient data to determine $B(0)$ and $B(1)$. Our method will set $B(0)$ and $B(1)$ before the computation starts. It is also necessary to set the initial values of $B(k)$, so as to control the system when the proper controller has not been obtained yet. The first two feedback gain matrices are set as follows

$$B(1)=B(0)=\begin{bmatrix}\beta_1 & 0 & \cdots & 0 & 0 & \cdots & 0\\ 0 & \beta_2 & \cdots & 0 & 0 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots & & \vdots\\ 0 & 0 & \cdots & \beta_m & 0 & \cdots & 0\end{bmatrix} \tag{25}$$

where β_i $(0<|\beta_i|<1,i=1,2,\ldots,m)$ are selected so that $\text{Rank}[B(1)]=\text{Rank}[B(0)]=m$.

When $k=0$, set $B(0)$ as in (25). During $[0,T)$, measure and record $x(0),x(\tau),\ldots,x(4n\tau)$. Then, calculate $A(0)$ and $F_x(0)$ by (24). Since

$$\forall k\geq 0,\ \text{Rank}\left[B(k)B^T(k)\right]=\text{Rank}[B(k)] \tag{26}$$

$[B(0)B^T(0)]$ is invertible. With $A(0)$, $B(0)$, and $F_x(0)$, we can obtain $F_u(0)$ by

$$F_u(0)=[A(0)-F_x(0)]B^T(0)\left[B(0)B^T(0)\right]^{-1}. \tag{27}$$

During $[T,2T)$, we can compute $F_u(1)$ in the same way:

$$F_u(1)=[A(1)-F_x(1)]B^T(1)\left[B(1)B^T(1)\right]^{-1} \tag{28}$$

where $A(1)$ and $F_x(1)$ are computed by (24).

Step 2 Calculate and Adjust $B(k)$ When $k\geq 2$:

It is not practical to calculate $A(k)$ and $F_x(k)$ before time kT arrives, since (14) and (18) can only be constructed after $(kT+2n\tau)$ and $(kT+4n\tau)$, respectively. However, we have to determine $B(k)$ before kT arrives and replace $B(k-1)$ with it at time kT, in order to control the system on time. To solve this problem, we make one-step predictive estimations of $A(k)$, $F_x(k)$, and $F_u(k)$. With these predictive estimations and the following assumption, we can design $B(k)$.

Assumption 4:

$$\forall k\geq 1,\ \text{Rank}[2F_u(k)-F_u(k-1)]=m. \tag{29}$$

For all $k\geq 2$, the one-step predictive estimations of $F_x(k)$ and $F_u(k)$ are

$$\begin{cases}\hat{F}_x(k)=F_x(k-1)+[F_x(k-1)-F_x(k-2)]\\ \hat{F}_u(k)=F_u(k-1)+[F_u(k-1)-F_u(k-2)].\end{cases} \tag{30}$$

$F_x(k-2)$, $F_u(k-2)$, $A(k-1)$, $F_x(k-1)$, and $F_u(k-1)$ have been computed and stored during $[(k-2)T,(k-1)T)$

and $[(k-1)T, kT)$. The control task during $[(k-1)T, kT)$ is to estimate $A(k)$, $F_x(k)$, and $F_u(k)$, and calculate $B(k)$ based on them. Since it is desired that $\|A(k)\| < 1$, we preset the desired $A(k)$ by using the previous results as follows

$$\hat{A}(k) = \frac{A(k-1) + \delta \|A(k-1)\| I_n}{(1 + \alpha + \|e(k-1)\|) \|A(k-1)\|} (k \geq 2) \quad (31)$$

where $0 \leq \delta < \alpha < 1$, and I_n is the identity matrix. We may adjust α to improve the convergence rate, and choose proper δ to make $\hat{A}(k)$ and $\hat{F}_x(k)$ satisfy

$$\text{Rank} \left[\hat{A}(k) - \hat{F}_x(k) \right] \geq m. \quad (32)$$

The choice of δ is always possible for the following reasons. Let

$$A_p(k) = \frac{A(k-1)}{(1 + \alpha + \|e(k-1)\|) \|A(k-1)\|}$$

$$A_\delta(k) = \frac{\delta I_n}{1 + \alpha + \|e(k-1)\|}.$$

Then, $\hat{A}(k) = A_p(k) + A_\delta(k)$. For given $k \geq 2$, $A_p(k)$, $A_\delta(k)$, and $\hat{F}_x(k)$ are all $n \times n$ constant matrices. There always exists a nonsingular $n \times n$ matrix P, such that

$$P^{-1} \left[A_p(k) + A_\delta(k) - \hat{F}_x(k) \right] P = \text{diag}[H_1, H_2, \ldots, H_q],$$

where

$$H_i = \begin{bmatrix} h_i & 1 & \cdots & 0 \\ 0 & h_i & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 \\ 0 & 0 & \cdots & h_i \end{bmatrix} \quad (1 \leq i \leq q),$$

is an $m_i \times m_i$ Jordan block, $h_i = \lambda_i + (\delta/(1 + \alpha + \|e(k-1)\|))$, λ_i is an eigenvalue of $[A_p(k) - \hat{F}_x(k)]$, and $\sum_{i=1}^{q} m_i = n$. Therefore, $\text{Rank}[\hat{A}(k) - \hat{F}_x(k)] = \sum_{i=1}^{q} \text{Rank}[H_i]$. In this case, if $\text{Rank}[A_p(k) - \hat{F}_x(k)] \geq m$, choose $\delta = 0$; otherwise, since $m \leq n$, there is at least one $\lambda_i = 0$, and we can always find such a $\delta > 0$, that makes $h_i > 0$, while not making other non-zero h_j $(1 \leq j \leq q, j \neq i)$ become zero. As a result, $\text{Rank}[\hat{A}(k) - \hat{F}_x(k)]$ will increase, and we can continuously try different values of δ, until (32) is satisfied.

It can be seen from (31) that

$$\left\| \hat{A}(k) \right\| = \left\| \frac{A(k-1) + \delta \|A(k-1)\| I_n}{(1 + \alpha + \|e(k-1)\|) \|A(k-1)\|} \right\|$$
$$\leq \frac{\|A(k-1)\| + \delta \|A(k-1)\|}{(1 + \alpha + \|e(k-1)\|) \|A(k-1)\|}$$
$$= \frac{1 + \delta}{1 + \alpha + \|e(k-1)\|}$$
$$< 1. \quad (33)$$

In our method, $B(k)$ will be calculated according to this $\hat{A}(k)$.

Similar to (11), the matrices $\hat{A}(k)$, $\hat{F}_x(k)$, $\hat{F}_u(k)$, and $B(k)$ have the following relationship:

$$\hat{A}(k) = \hat{F}_x(k) + \hat{F}_u(k)B(k). \quad (34)$$

By Assumption 4 and (30), $\left[\hat{F}_u^T(k)\hat{F}_u(k) \right]^{-1}$ exists, since

$$\text{Rank} \left[\hat{F}_u^T(k)\hat{F}_u(k) \right] = \text{Rank} \left[\hat{F}_u(k) \right] = m.$$

Then, $B(k)$ $(k \geq 2)$ is calculated as

$$B(k) = \left[\hat{F}_u^T(k)\hat{F}_u(k) \right]^{-1} \hat{F}_u^T(k) \left[\hat{A}(k) - \hat{F}_x(k) \right]. \quad (35)$$

On the other hand, by (32) and (34), we have

$$m \leq \text{Rank} \left[\hat{F}_u(k)B(k) \right] \leq \text{Rank} \left[B(k) \right]. \quad (36)$$

According to (35), $\text{Rank}[B(k)] \leq \text{Rank}[\hat{F}_u(k)] = m$. Therefore, $\text{Rank}[B(k)] = m$, and $[B(k)B^T(k)]$ is invertible.

After applying $B(k)$ to the system at time kT, measure and store the state data. Then, compute the real $A(k)$ and $F_x(k)$ by (24). Because $B(k)$ given in (35) will stay unchanged until time $(k+1)T$ arrives, the real $F_u(k)$ can be obtained by

$$F_u(k) = [A(k) - F_x(k)] B^T(k) \left[B(k)B^T(k) \right]^{-1}. \quad (37)$$

Store the computed $A(k)$, $F_x(k)$, and $F_u(k)$ for computing $\hat{A}(k+1)$, $\hat{F}_x(k+1)$, and $\hat{F}_u(k+1)$ during the next time interval.

Step 3 Check the Result:

At the time instant kT, check whether $x(k) = x^*$ (or, whether $\|e(k)\| \leq \varepsilon$). If $\|e(k)\| \leq \varepsilon$, keep on using the same $B(k)$ from then on; otherwise, repeat the procedure illustrated in Step 2, until this condition is satisfied.

Case 2 $m = n$:

Case 2 is similar to Case 1, while there are a few differences between them. The first different point is that, when setting the initial values in Step 1, we now set

$$B(1) = B(0) = \text{diag}[\beta_1, \beta_2, \ldots, \beta_n] \quad (38)$$

where β_i $(0 < |\beta_i| < 1, i = 1, 2, \ldots, n)$ are selected to make $\text{Rank}[B(1)] = \text{Rank}[B(0)] = n$.

The second different point is that, $B(k)$ and $\hat{F}_u(k)$ are both square matrices. Thus, we have

$$\begin{cases} B^T(k) \left[B(k)B^T(k) \right]^{-1} = B^{-1}(k) \\ \left[\hat{F}_u^T(k)\hat{F}_u(k) \right]^{-1} \hat{F}_u^T(k) = \hat{F}_u^{-1}(k). \end{cases} \quad (39)$$

When computing $B(k)$, it is a little different from (35):

$$B(k) = \hat{F}_u^{-1}(k) \left[\hat{A}(k) - \hat{F}_x(k) \right]. \quad (40)$$

Then, we can obtain the real $F_u(k)$ by

$$F_u(k) = [A(k) - F_x(k)] B^{-1}(k). \quad (41)$$

The rest of Case 2 are the same as those of Case 1, and will not be discussed here.

In summary, this data-based state feedback control method can be depicted using the block diagram shown in Fig. 1.

Remark 2: There are some points need to be noted.

1) In Assumption 2, the smaller the T is, the more accurately $A(k)$ and $F_x(k)$ will be estimated. It will be easier to stabilize the original continuous-time system. However, in order to calculate $B(k)$ before time kT, it is required that $(4n\tau + t_c) \leq T$, where t_c represents the computation time. But, τ and t_c are limited by the sampler and computer speed. Therefore, T cannot be arbitrarily small, and should be carefully selected.

2) If Assumptions 1 and 2 can be satisfied, our method is also applicable to the system, whose desired state is a slowly time-varying signal $x^*(k)$. For this case, we divide the time domain into a group of sections, making sure that during each time section, all the elements of $x^*(k)$ are monotone. The final value of $x^*(k)$ in each time section will be used not only as the desired value for the current time section, but also as the initial value for the next time section. Then, we can apply our method to control the system, making $x(k)$ track $x^*(k)$.

IV. CONVERGENCE ANALYSIS

In this section, we will study the convergence condition, under which $B(k)$ guarantees $\|A(k)\| < 1$. Define

$$\begin{cases} a(k) = \|\Delta F_x(k) - \Delta F_x(k-1)\| \\ c(k) = \|\Delta F_u(k) - \Delta F_u(k-1)\| \end{cases} \tag{42}$$

where

$$\Delta F_x(k) = F_x(k) - F_x(k-1)$$
$$\Delta F_u(k) = F_u(k) - F_u(k-1).$$

Then, we have the following theorem.

Theorem 1: If

$$a(k) + c(k)\|B(k)\| < \frac{\|e(k-1)\| + \alpha - \delta}{\|e(k-1)\| + \alpha + 1}(k \geq 2) \tag{43}$$

where α and δ are introduced in (31), then the state $x(k)$ of system (2) will converge to its desired value x^* as $k \to \infty$.

Proof: It is clear that (43) is equivalent to

$$a(k) + c(k)\|B(k)\| + \frac{1+\delta}{\|e(k-1)\| + \alpha + 1} < 1(k \geq 2). \tag{44}$$

From (11), (30), (34), and (42), we can infer that

$$\begin{aligned} &a(k) + c(k)\|B(k)\| \\ &= \|\Delta F_x(k) - \Delta F_x(k-1)\| \\ &\quad + \|\Delta F_u(k) - \Delta F_u(k-1)\| \|B(k)\| \\ &\geq \|\Delta F_x(k) - \Delta F_x(k-1)\| \\ &\quad + \|\Delta F_u(k)B(k) - \Delta F_u(k-1)B(k)\| \\ &\geq \|F_x(k) - [2F_x(k-1) - F_x(k-2)] + F_u(k)B(k) \\ &\quad - [2F_u(k-1) - F_u(k-2)]B(k)\| \\ &= \left\| [F_x(k) + F_u(k)B(k)] - \left[\hat{F}_x(k) + \hat{F}_u(k)B(k)\right] \right\| \\ &= \left\| A(k) - \hat{A}(k) \right\|. \end{aligned} \tag{45}$$

From (33), (44), and (45), we can obtain that $\forall k \geq 2$,

$$\begin{aligned} \|A(k)\| &\leq \left\| A(k) - \hat{A}(k) \right\| + \left\| \hat{A}(k) \right\| \\ &\leq a(k) + c(k)\|B(k)\| + \frac{1+\delta}{\|e(k-1)\| + \alpha + 1} \\ &< 1. \end{aligned} \tag{46}$$

Since $\|e(k+1)\| = \|A(k)e(k)\| \leq \|A(k)\|\|e(k)\|$, from (10) and (46), we can conclude that $x(k) \to x^*$ as $k \to \infty$. ∎

V. SIMULATION STUDIES

In this section, a smooth mathematical model, which represents the three-tank system (TTS) DTS200 [23], is adopted for computer simulation. The layout of this system is illustrated in Fig. 2, which consists of three cylinders: tank 1 (T_1), tank 3 (T_3) and tank 2 (T_2), having the same cross section area $S_a = 154$ cm^2. These cylinders are connected in series with each other by two cylindrical pipes, which have cross section area $S_n = 0.5$ cm^2. The nominal outflow valve with cross section area S_n is located at tank 2. The outflow liquid is collected in a reservoir, which supplies pump 1 and pump 2. Denote $h_i(t)$ ($t \geq 0$) as the liquid level (in cm) of tank T_i ($i = 1, 2, 3$), whose measurements will be carried out by difference pressure sensors. $Q_1(t)$ and $Q_2(t)$ are the supplying flow rates (in cm^3/s) of pump 1 and pump 2, respectively. Under the assumption that $h_1(t) > h_3(t) > h_2(t)$, we have the following nonlinear continuous-time model:

$$\begin{aligned} \frac{dh_1(t)}{dt} &= -\frac{1}{S_a}a_1 S_n \sqrt{2g(h_1(t) - h_3(t))} + \frac{1}{S_a}Q_1(t) \\ \frac{dh_2(t)}{dt} &= \frac{1}{S_a}a_3 S_n \sqrt{2g(h_3(t) - h_2(t))} \\ &\quad - \frac{1}{S_a}a_2 S_n \sqrt{2gh_2(t)} + \frac{1}{S_a}Q_2(t) \\ \frac{dh_3(t)}{dt} &= \frac{1}{S_a}a_1 S_n \sqrt{2g(h_1(t) - h_3(t))} \\ &\quad - \frac{1}{S_a}a_3 S_n \sqrt{2g(h_3(t) - h_2(t))} \end{aligned} \tag{47}$$

where $a_1 = 0.5$, $a_2 = 0.6$, and $a_3 = 0.5$ denote the outflow coefficients; while $Q_{1\max} = 93$ cm^3/s, $Q_{2\max} = 91$ cm^3/s and $H_{\max} = 62$ cm are the maximum values of $Q_1(t)$, $Q_2(t)$ and $h_i(t)$, respectively. Here, $g = 9.81$ m/s^2 represents the gravity acceleration.

Let $u_l(t) = Q_l(t)$ ($l = 1, 2$) and $x_i(t) = h_i(t)$ ($i = 1, 2, 3$) denote the inputs and the states of system (47), respectively. Thus, the dimensions of the input vector and the state vector are $m = 2$ and $n = 3$.

In this example, we use Euler method to approximate the sampled system (47) as follows

$$\begin{aligned} x(kT + (j+1)\tau) &= x(kT + j\tau) + \tau f_h(x(kT + j\tau)) \\ &\quad + \tau Du(kT + j\tau) \quad (0 \leq j \leq N-1), \end{aligned} \tag{48}$$

where $\tau = T/N$ is the fast sampling period in our method, and other terms in (48) are

$$f_h(x) = \begin{bmatrix} -Q_{13} \\ Q_{32} - Q_{20} \\ Q_{13} - Q_{32} \end{bmatrix}, \quad D = \frac{1}{S_a}\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{49}$$

and

$$\begin{cases} Q_{13} = \frac{1}{S_a}a_1 S_n \sqrt{2g(x_1 - x_3)} \\ Q_{32} = \frac{1}{S_a}a_3 S_n \sqrt{2g(x_3 - x_2)} \\ Q_{20} = \frac{1}{S_a}a_2 S_n \sqrt{2gx_2}. \end{cases} \tag{50}$$
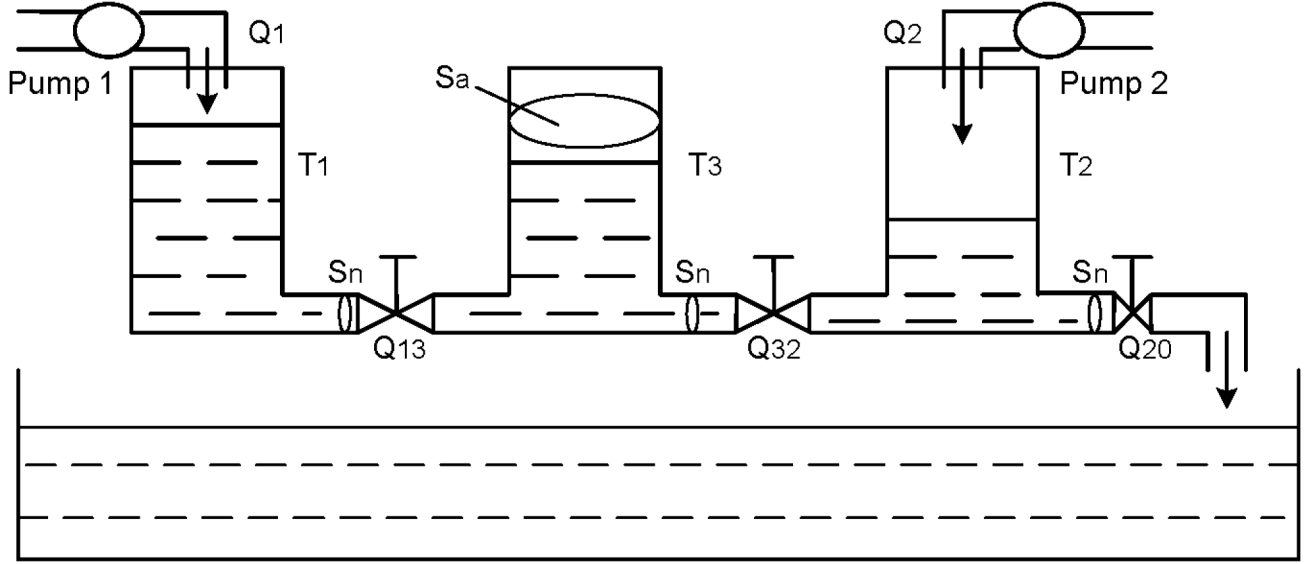
Fig. 2. The layout of the three-tank system DTS200.

TABLE I
T, τ, α AND CONVERGENCE TIME

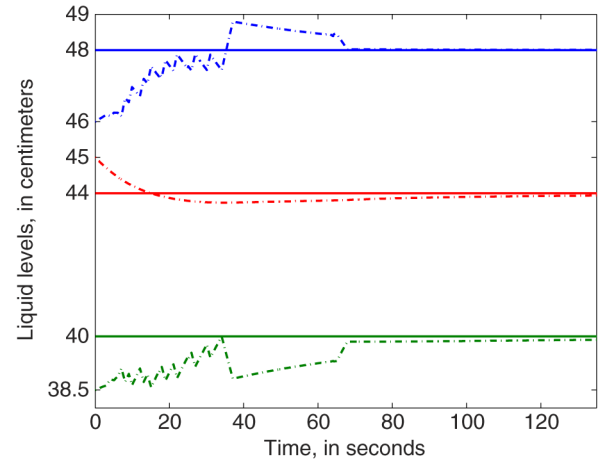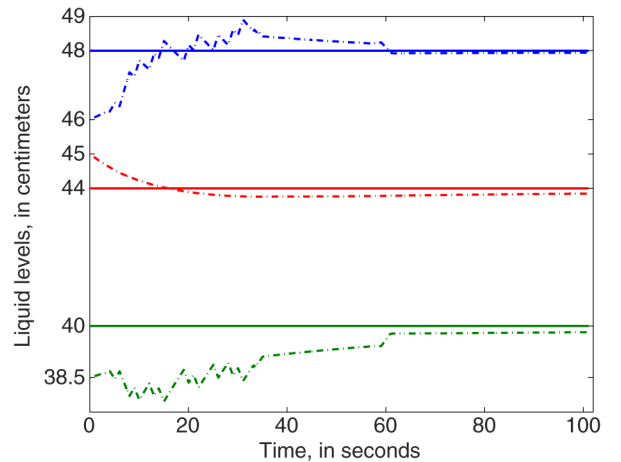| T	τ	α	convergence time
$1s$	$0.07s$	0.01	$61s$
$1s$	$0.06s$	0.01	$67s$
$1.1s$	$0.08s$	0.03	$45.1s$
$1.2s$	$0.09s$	0.02	$63.6s$

Now, we set the desired state, the initial liquid levels, and $B(0)$, $B(1)$ as

$$x^* = [48, 40, 44]^T$$
$$x(0) = [46, 38.5, 45]^T$$
$$B(1) = B(0) = \begin{bmatrix} -0.0038 & 0.00049 & 0 \\ 0.0010 & -0.00301 & 0 \end{bmatrix}. \quad (51)$$

We then design the controller with the state data, which are generated by model (48). Note that, the state data used to calculate $A(k)$, $F_x(k)$, $F_u(k)$, and $B(k)$ are not actual measurement results, but are generated by Matlab programs using model (48) and the initial values in (51). The corresponding methods to calculate the above matrices are illustrated in Section III.

The following figures illustrate the simulation results. They have the same initial conditions, but different T, τ, and α. In these figures, the horizontal axis represents the time (in seconds), while the vertical one represents the liquid levels (in cm). The solid lines are the desired liquid levels x_i^*, and the dash dot lines are the states $x_i(kT + j\tau)$ $(i = 1, 2, 3)$.

Observe the curves shown in these figures, all of the states starting from $x(0)$ converge to x^* as time goes to infinity, which illustrate the feasibility of our method. The values of T, τ, α and the corresponding convergence times are listed in Table I. We can see that, our method is applicable to control the system even with different values of T. As discussed in Remark 1, when T is fixed, a smaller τ will cause larger estimation errors. As a result, $x(k)$ will converge to x^* at a slower rate. The curves in Figs. 3 and 4 match this conclusion. It also can be seen from

Fig. 3. Simulation result when $T = 1$ s, $\tau = 0.06$ s and $\alpha = 0.01$.

Fig. 4. Simulation result when $T = 1$ s, $\tau = 0.07$ s and $\alpha = 0.01$.

these curves that, different values of T, τ, and α will cause different overshoots.
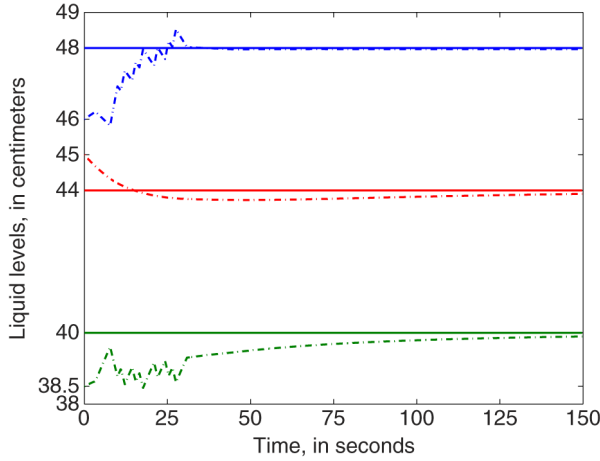
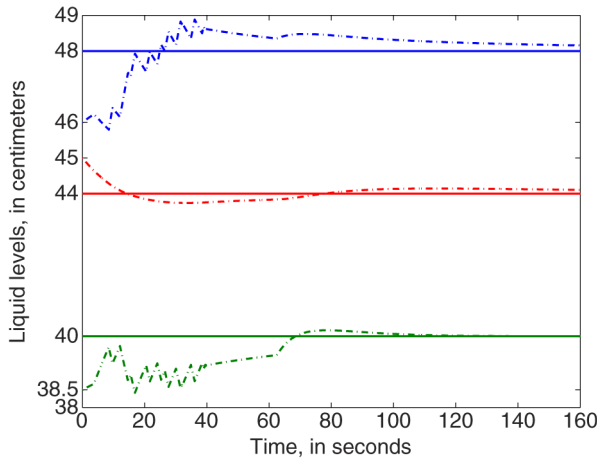Fig. 5. Simulation result when $T = 1.1$ s, $\tau = 0.08$ s and $\alpha = 0.03$.

Fig. 6. Simulation result when $T = 1.2$ s, $\tau = 0.09$ s and $\alpha = 0.02$.

These simulation results indicate that our approach can guarantee stability of the controlled system when the sampling period is sufficiently small. Thus, in order to avoid the feedback stabilization problem caused by discretization method as discussed in [24], we should use a T as small as possible.

VI. Conclusions

In this paper, we developed a data-based state feedback control method for a class of nonlinear systems, which have unknown mathematical models. The principle of this method is first to calculate the feedback gain matrix $B(k)$ in real-time, by using the measured data during each $[(k-1)T, kT)$, and then replace $B(k-1)$ with it when time kT arrives. This $B(k)$ will make the system state converge to its desired value as time goes to infinity.

There are three advantages of our method. The first advantage is that, it provides an approach to approximate the nonlinear system with a linear time-invariant model during each time interval $[kT, (k+1)T)$, which does not have higher order infinitesimals that need to be ignored. Thus, this dynamic linearization is accurate, and can be applied to nonlinear systems, which are continuous and have continuous first-order partial derivatives.

In engineering practice, there are many systems satisfying Assumption 1, for instance, liquid level control systems, pneumatic control systems, thermal control systems, and hydraulic control systems, etc. [25, Chap. 4]. According to this situation, our method may have broad application prospects.

This data-based state feedback control method employs a fast sampling technique to collect the state data. With these data, we can calculate the corresponding matrices $A(k)$, $F_x(k)$, $F_u(k)$, and $B(k)$. Therefore, the second advantage is that, we can directly control the system without modeling it.

The third advantage is that, $\hat{A}(k)$ is related to the previous results $e(k-1)$ and $A(k-1)$, and the adjustment of $B(k)$ is a process of estimating, updating and correcting. So it can be inferred that, our method has the adaptive capacity against uncertain changes of system dynamics.

Computer simulation results demonstrated the effectiveness of this control method. These results also showed the influences of the parameters T, τ, and α on the convergence rate and the overshoot.

The systems studied in this paper are deterministic, while in practical terms, the measurement noise often exists in sampled data. We will extend this data-based control method to deal with the systems with measurement noise in future research, by using least squares estimation approaches.

References

[1] T. Chai, Z. Hou, F. L. Lewis, A. Hussain, and D. Zhao, "Guest editorial: Special section on data-based control, modeling, and optimization," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2150–2153, Dec. 2011.

[2] D. Wang, "Robust data-driven modeling approach for real-time final product quality prediction in batch process operation," *IEEE Trans. Ind. Inf.*, vol. 7, no. 2, pp. 371–377, May 2011.

[3] G. Li, B. Liu, S. J. Qin, and D. Zhou, "Quality relevant data-driven modeling and monitoring of multivariate dynamic processes: The dynamic T-PLS approach," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2262–2271, Dec. 2011.

[4] D. Wang, J. Liu, and R. Srinivasan, "Data-driven soft sensor approach for quality prediction in a refining process," *IEEE Trans. Ind. Inf.*, vol. 6, no. 1, pp. 11–17, Feb. 2010.

[5] C. Caione, D. Brunelli, and L. Benini, "Distributed compressive sampling for lifetime optimization in dense wireless sensor networks," *IEEE Trans. Ind. Inf.*, vol. 8, no. 1, pp. 30–40, Feb. 2012.

[6] Z. Hou and S. Jin, "A novel data-driven control approach for a class of discrete-time nonlinear systems," *IEEE Trans. Contr. Syst. Technol.*, vol. 19, no. 6, pp. 1549–1558, Nov. 2011.

[7] A. Gomperts, A. Ukil, and F. Zurfluh, "Development and implementation of parameterized FPGA-based general purpose neural networks for online applications," *IEEE Trans. Ind. Inf.*, vol. 7, no. 1, pp. 78–89, Feb. 2011.

[8] B. Subudhi and D. Jena, "Nonlinear system identification using memetic differential evolution trained neural networks," *Neurocomput.*, vol. 74, no. 10, pp. 1696–1709, May 2011.

[9] C. Y. Yeh, W. R. Jeng, and S. J. Lee, "Data-based system modeling using a type-2 fuzzy neural network with a hybrid learning algorithm," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2296–2309, Dec. 2011.

[10] N. Dong and Z. Chen, "A novel data based control method based upon neural network and simultaneous perturbation stochastic approximation," *Nonlinear Dynamics*, vol. 67, no. 2, pp. 957–963, Jan. 2012.

[11] I. Steinwart and A. Christmann, *Support Vector Machines*. New York: Springer, 2008.

[12] S. Ertekin, L. Bottou, and C. L. Giles, "Nonconvex online support vector machines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 368–381, Feb. 2011.

[13] C. Gao, L. Jian, and S. Luo, "Modeling of the thermal state change of blast furnace hearth with support vector machines," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 1134–1145, Feb. 2012.

[14] C. T. Freeman, Z. Cai, E. Rogers, and P. L. Lewin, "Iterative learning control for multiple point-to-point tracking application," *IEEE Trans. Contr. Syst. Technol.*, vol. 19, no. 3, pp. 590–600, May 2011.
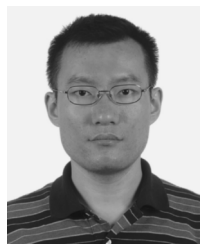
[15] S. Taniguchi, H. Kino, R. Ozawa, R. Ishibashi, M. Uemura, K. Kanaoka, and S. Kawamura, "Inverse dynamics of human passive motion based on iterative learning control," *IEEE Trans. Syst., Man Cybern., Part A: Syst. Humans*, vol. 42, no. 2, pp. 307–315, Mar. 2012.

[16] X. Ruan, Z. Z. Bien, and Q. Wang, "Convergence characteristics of proportional-type iterative learning control in the sense of Lebesgue-p norm," *IET Contr. Theory Appl.*, vol. 6, no. 5, pp. 707–714, Mar. 2012.

[17] T. Chai, Y. Zhang, H. Wang, C. Y. Su, and J. Sun, "Data-based virtual unmodeled dynamics driven multivariable nonlinear adaptive switching control," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2154–2172, Dec. 2011.

[18] Z. S. Hou and S. T. Jin, "Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 22, pp. 2173–2188, Dec. 2011.

[19] Y. Fujisaki and K. Saitou, "Virtual reference feedback tuning for H_∞ controllers," in *Proc. SICE Ann. Conf. 2011*, Tokyo, Japan, Sep. 2011, pp. 1768–1769.

[20] L. Campestrini, D. Eckhard, M. Gevers, and A. S. Bazanella, "Virtual reference feedback tuning for non-minimum phase plants," *Automatica*, vol. 47, no. 8, pp. 1778–1784, Aug. 2011.

[21] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, NJ: Princeton Univ. Press, 2012.

[22] J. E. Gentle, *Matrix Algebra: Theory, Computations, and Applications in Statistics*. New York: Springer, 2007.

[23] D. H. Zhou, X. He, Z. Wang, G. P. Liu, and Y. D. Ji, "Leakage fault diagnosis for an internet-based three-tank system: An experimental study," *IEEE Trans. Contr. Syst. Technol.*, vol. 20, no. 4, pp. 857–870, Jul. 2012.

[24] D. Nešić and A. R. Teel, "A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models," *IEEE Trans. Autom. Contr.*, vol. 49, no. 7, pp. 1103–1122, Jul. 2004.

[25] K. Ogata, *Modern Control Engineering*, 5th ed. Englewood Cliffs, NJ: Prentice Hall, 2009.

Zhuo Wang was born in Handan, Hebei, China, in 1983. He received the B.E. degree from Beihang University, Beijing, China, in 2006. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Illinois at Chicago. His Ph.D. thesis addresses data-based analysis and control for complex nonlinear dynamical systems.

He has published several journal papers in relevant research areas. He is also interested in dynamic programming, model-free adaptive control, and artificial neural networks.

Derong Liu (S'91–M'94–SM'96–F'05) received the B.S. degree in mechanical engineering from the East China Institute of Technology (now Nanjing University of Science and Technology), Nanjing, China, in 1982, the M.S. degree in automatic control theory and applications from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1987, and the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 1994.

Dr. Liu was a Product Design Engineer with China North Industries Corporation, Jilin, China, from 1982 to 1984. He was an Instructor with the Graduate University of the Chinese Academy of Sciences, Beijing, China, from 1987 to 1990. He was a Staff Fellow with General Motors Research and Development Center, Warren, MI, from 1993 to 1995. He was an Assistant Professor in the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, from 1995 to 1999. He joined the University of Illinois at Chicago in 1999, and became a Full Professor of electrical and computer engineering and of computer science in 2006. He was selected for the "100 Talents Program" by the Chinese Academy of Sciences in 2008. He has published 10 books (five research monographs and five edited volumes).

Dr. Liu was an Associate Editor of *Automatica* (2006–2009). He serves as an Associate Editor of *Neurocomputing, Neural Computing and Applications, Soft Computing, International Journal of Neural Systems, Journal of Control Science and Engineering*, and *Science in China Series F: Information Sciences*. He was General Chair for the 2012 International Conference on Brain Inspired Cognitive Systems (Shenyang, China) and the 2007 International Symposium on Neural Networks (Nanjing, China). He is an elected member of the Board of Governors of the International Neural Network Society. He was a member of the Conference Editorial Board of the IEEE Control Systems Society (1995–2000), an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: FUNDAMENTAL THEORY AND APPLICATIONS (1997—1999), the IEEE TRANSACTIONS ON SIGNAL PROCESSING (2001–2003), the IEEE TRANSACTIONS ON NEURAL NETWORKS (2004–2009), the IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE (2006–2009), and the IEEE CIRCUITS AND SYSTEMS MAGAZINE (2008–2009), and the Letters Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS (2006–2008). He was the Founding Editor of the IEEE Computational Intelligence Society's *Electronic Letter* (2004–2009). Currently, he is the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and an Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY. He was General Chair for the 2009 IEEE Conference on Service Operations, Logistics, and Informatics (Chicago, IL) and the 2008 IEEE International Conference on Networking, Sensing and Control (Sanya, China). He is the General Chair of 2014 IEEE World Congress on Computational Intelligence (Beijing, China). He was an elected AdCom member of the IEEE Computational Intelligence Society (2006–2008). He received the Michael J. Birck Fellowship from the University of Notre Dame (1990), the Harvey N. Davis Distinguished Teaching Award from Stevens Institute of Technology (1997), the Faculty Early Career Development (CAREER) Award from the National Science Foundation (1999), the University Scholar Award from University of Illinois (2006), and the Overseas Outstanding Young Scholar Award from the National Natural Science Foundation of China (2008). He is a member of Eta Kappa Nu and a Fellow of the IEEE.