

Research Article

Neural-Fuzzy Digital Strategy of Continuous-Time Nonlinear Systems Using Adaptive Prediction and Random-Local-Optimization Design

Zhi-Ren Tsai

Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan

Correspondence should be addressed to Zhi-Ren Tsai; ren@asia.edu.tw

Received 24 January 2013; Revised 18 March 2013; Accepted 21 March 2013

Academic Editor: Bo Shen

Copyright © 2013 Zhi-Ren Tsai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A tracking problem, time-delay, uncertainty and stability analysis of a predictive control system are considered. The predictive control design is based on the input and output of neural plant model (NPM), and a recursive fuzzy predictive tracker has scaling factors which limit the value zone of measured data and cause the tuned parameters to converge to obtain a robust control performance. To improve the further control performance, the proposed random-local-optimization design (RLO) for a model/controller uses offline initialization to obtain a near global optimal model/controller. Other issues are the considerations of modeling error, input-delay, sampling distortion, cost, greater flexibility, and highly reliable digital products of the model-based controller for the continuous-time (CT) nonlinear system. They are solved by a recommended two-stage control design with the first-stage (offline) RLO and second-stage (online) adaptive steps. A theorizing method is then put forward to replace the sensitivity calculation, which reduces the calculation of Jacobin matrices of the back-propagation (BP) method. Finally, the feedforward input of reference signals helps the digital fuzzy controller improve the control performance, and the technique works to control the CT systems precisely.

1. Introduction

During the past decade, many fuzzy theories [1–7] and delay analysis [8–13] have attracted great attention from both the academic and industrial communities, and there have been many successful applications. Despite this success, it has become evident that many basic and important issues remain to be further addressed. These stability analysis and systematic designs are among the most important issues for robust control systems [1], and there has been significant research on these issues (see [2, 14–16]). However, the problems of modeling error and prediction are still worth to be solved further. Therefore, the two-stage control method in this paper is proposed to suppress the modeling error to guarantee the stability of predictive control system in the presence of this modeling error.

In addition, neural-network- (NN-) based modeling has become an active research field because of its unique merits in solving complex nonlinear system identification and control problems (see [15]). Neural networks (NNs) or NARMAX/NARX neural networks [17] are composed of simple elements operating in parallel, inspired by biological nervous systems. A neural network can be trained to represent a particular function by adjusting the weights between elements. Due to discrete-time (DT) controllers (microcontrollers) being cheaper and more flexible than continuous-time (CT) controllers, the DT control problem for CT plant is worth studying. In modern control engineering, controllers are commonly implemented directly by the hardware or software of digital computers. However, one important issue has to be faced; that is, the proposed design (DT-CT design) in this paper effects a new type of application.

The study of CT control of CT time-delay systems has received considerable attention in recent years since delay is a major cause of poor performance in many important engineering systems [18, 19]. As is known, the delay control problem is an important and complex factor in the stability performance of CT nonlinear systems. In general, a delay CT time-delay control systems needs to involve the DT control problem. Moreover, the recommended adaptive NN-model-based design method has not yet been developed to adjust the parameters of a discrete-time (DT) adaptive fuzzy controller such that the original continuous-time (CT) system, with time delays and uncertainties, is uniformly ultimately bounded (UUB) stable in Section 2.

Based on the timer of the micro-controller, the effect of delay in neural system identification can be approximated by many tape-delay terms. This reduces the difficulty of delay identification. The DT NARMAX model is generally sufficient to approximate an unknown, nonlinear, dynamical, and delayed CT system by selecting an appropriate sampling time. Moreover, the measured modeling error between the model and the physical system is considered in the theorems by Lyapunov functions. In Remark 1, this kind of modeling error is used to estimate the proposed corner of sampling period (CSP) which decides this kind of sampling time to support the intersample method.

The feedforward term in [20] is derived indirectly by assuming some constraints. Due to the overfitting problem and the local optimal problem of NN, the method [20] is not suitable for real applications because of the need for lengthy convergence time. These neural techniques [20, 21] have usually been demonstrated under nonlinear control due to their powerful nonlinear modeling capability [22] and adaptability. However, they must exhibit the optimal problems of falling into the local minimum easily by using the back-propagation (BP) [23] method. To alleviate the requirements for accurate modeling of the plant, the proposed NARMAX plant and control models are trained by initially using novel offline methods with the RLO algorithm to improve this drawback. It not only guarantees the gradient decent method against the local optimal solution but also speeds up the convergence of the PSO [24].

Conventional optimization methods assume that all parameters and goals of a model are precisely known [25, 26]. However, in many practical problems incomplete and unreliable information exists [27]. Therefore, the two-stage scheme is proposed to treat this unknown system as reliability problem.

Inspired by the DT neural controller of [20] only for a DT system, a digital fuzzy control design for a CT system is proposed and an approximate inverse of the delayed plant dynamics is used to act as the NARMAX fuzzy controller. The adaptive controller and NARMAX models are easier to converge than [20, 21] by the recommended two-stage scheme. Moreover, this paper concludes with a simulation example and experimental data to demonstrate these techniques.

The remainder of this paper is organized as follows. In Section 2, the system description and its problem formulation are provided. In Section 3, robustness controllers and stability analysis in the presence of modeling errors for a series of control design algorithms are introduced. In Section 4, the effectiveness of the proposed approach is illustrated by a numerical example with three cases. Finally, the conclusions are drawn in Section 5.

2. System Description and Problem Formulation

First, the conventional PWM buck converter by using AM-OTS-DS [28] methodology is modeling to the following equivalent circuit plant:

$$\begin{bmatrix} \dot{i}_{L}(t) \\ \dot{v}_{C}(t) \end{bmatrix} = \begin{bmatrix} \frac{-1}{L} \begin{bmatrix} R_{L} + \frac{\overline{R} \cdot R_{C}}{\overline{R} + R_{C}} \end{bmatrix} & \frac{-\overline{R}}{L \cdot (\overline{R} + R_{C})} \\ \frac{\overline{R}}{C \cdot (\overline{R} + R_{C})} & \frac{-1}{C \cdot (\overline{R} + R_{C})} \end{bmatrix} \begin{bmatrix} \dot{i}_{L}(t) \\ v_{C}(t) \end{bmatrix} \\ + \begin{bmatrix} \frac{-1}{L} \begin{bmatrix} R_{M} \dot{i}_{L}(t) - V_{\text{in}} - V_{D} \end{bmatrix} \\ u(t) + \begin{bmatrix} \frac{-V_{D}}{L} \\ 0 \end{bmatrix}, \\ v_{o}(t) = \begin{bmatrix} \overline{R} \cdot R_{C} \\ \overline{\overline{R} + R_{C}} \end{bmatrix} \begin{bmatrix} \dot{i}_{L}(t) \\ v_{C}(t) \end{bmatrix}.$$
(1a)

In this paper, the robustness of the above control system is emphasized, so uncertainty Δ and delay τ are added to the original control system (1a)

$$\begin{bmatrix} \dot{i}_{L}(t) \\ \dot{v}_{C}(t) \end{bmatrix} = \begin{bmatrix} \frac{-1}{L} \begin{bmatrix} R_{L} + \frac{R \cdot R_{C}}{R + R_{C}} \end{bmatrix} & \frac{-R}{L \cdot (R + R_{C})} \\ \frac{R}{C \cdot (R + R_{C})} & \frac{-1}{C \cdot (R + R_{C})} \end{bmatrix} \begin{bmatrix} i_{L}(t) \\ v_{C}(t) \end{bmatrix}$$
$$+ \begin{bmatrix} \frac{-1}{L} \begin{bmatrix} R_{M} i_{L}(t) - V_{\text{in}} - V_{D} \end{bmatrix} \end{bmatrix} u(t) + \begin{bmatrix} \frac{-V_{D}}{L} \\ 0 \end{bmatrix},$$
$$v_{o}(t) = \begin{bmatrix} \frac{R \cdot R_{C}}{R + R_{C}} & \frac{R}{R + R_{C}} \end{bmatrix} \begin{bmatrix} i_{L}(t) \\ v_{C}(t) \end{bmatrix} + \zeta \cdot v_{o}(t - \tau),$$
(1b)

where $R = \overline{R} + \Delta$; $\overline{R} = 6$; $V_{in} = 30$ is a DC voltage source; $C = 202.5 \times 10^{-6}$; $\Delta = 0.52 \sin(2\pi \cdot t/3)$; $\zeta = 0.1$; $\tau = 10^{-2}(1 + \sin(10t))$; $L = 98.58 \times 10^{-6}$; $R_L = 48.5 \times 10^{-3}$, and $R_C = 162 \times 10^{-3}$ are the parasitic resistances of the inductor and capacitor, respectively. The element $R_M = 0.27$ is the static drain to source resistance of the power MOSFET, and $V_D = 0.82$ is the forward voltage of the power diode. u(t) is duty ratio of conventional PWM buck converter. The state $x(t) = x = [i_L(t), v_C(t)]^T$ and the output $y(t) = v_o(t)$ of this power system are defined.

The nonlinear, uncertain, hotter circuit's components, time-delay, and digital control problems of PWM buck converter CT system renders a tracking control problem difficult to analyze. A simulation system in (1a) and (1b) is built with uncertainty. In this study, it is assumed that the parameters of the circuit's components are not ideal, and the capacity of the digital controller is limited by using a lower-cost chip. Here, the sampling period $T_s = 10^{-4}$ is designed



FIGURE 1: UBSP of (1a) and (1b) is decided based on CSP.

for this power system (1a) and (1b). Hence, the delay τ is very large for this system.

Then, consider a general nonlinear system with delays described as follows:

$$P: \dot{x}(t) = f(x, u, t, \tau, \Delta), \qquad y(t) = g(x), \qquad (2)$$

where the bounded uncertainties $\Delta(t)$ create the dynamic quality of the system parameters which refer to electrical elements of the power system; the zero-order-hold control input $u(t) = u(k \cdot T_s) = u(k) = u_k$, $t = k \cdot T_s$, T_s is the appropriate sample time of DT-CT design, k is the index of the discrete result u_k of u(t) referring to the NN model of (1a) and (1b); τ is the time delay; $g(\cdot)$ is the relational function of the state x(t) and system output y(t).

Remark 1. With the understanding that normal physical systems are low-pass filter and smooth nonlinearity, the upper bound, UBSP (shown in Figure 1), of the test sampling period T_s is decided based on a test intersample method with the plot of logarithm *x*-axis for T_s and logarithm *y*-axis for $\sum_k |e_{\text{mod}}(k)|$, where $e_{\text{mod}}(k) = \hat{y}_k - y_k$. In Figure 1, we can find an obvious corner of sampling period (CSP) easily to support the intersample method.

3. Stability Analysis for Two-Stage Control Scheme

3.1. Stability Analysis with Neural-Fuzzy Design. In this paper, an adaptive prediction control structure is proposed, as shown in Figure 2, where the FRP controller C_F is designed as follows:

$$u(z(k \cdot T)) = u_k = C_F(z_k, (W_C)_k, T_s) + s \cdot C_X(e_k)$$

= $u_F + s \cdot u_X$, (3)

where the switch index

$$s = \begin{cases} 0, & \tilde{e}_{k+1} \ge 0, \\ 1, & \tilde{e}_{k+1} < 0, \end{cases}$$

$$u_k(\hat{z}) = C_F(\hat{z}_k, (W_C)_k, T_s), \qquad (4)$$

where $u_P = u_F + u_X$, $\hat{y}_{(k+1)1} = \hat{P}(u_P(k+1))$, $\hat{y}_{(k+1)2} = \hat{P}(u_F(k+1))$, $\hat{e}_{(k+1)1} = r_{k+1} - \hat{y}_{(k+1)1}$, $\hat{e}_{(k+1)2} = r_{k+1} - \hat{y}_{(k+1)2}$, $|\hat{e}_{(k+1)1}| - |\hat{e}_{(k+1)2}| = \tilde{e}_{k+1}$, as shown in Figure 2. The feedforward terms are reference signals $[r_k, r_{k-1}, \dots, r_{k-p}]$, and recursive terms are control signals $[u_{k-1}, u_{k-2}, \dots, u_{k-q}]$.

The offline training input of controller is

$$\hat{z}(k) = \hat{z}_{k} = \left[y_{k}, y_{k-1}, \dots, y_{k-p}, u_{k-1}, u_{k-2}, \dots, u_{k-q} \right],
z(k) = z_{k} = \left[r_{k}, r_{k-1}, \dots, r_{k-p}, u_{k-1}, u_{k-2}, \dots, u_{k-q} \right],$$
(5)

 $y_k = y(k)$, $r_k = r(k)$. The controller has two working phases: z(k) is the data vector of the testing phase, and $\hat{z}(k)$ is the data vector of the training phase. The fuzzy controller is as follows:

$$u_{F}(z(k \cdot T)) = u_{F}(k)$$

$$= \frac{\sum_{l=1}^{M} \overline{h}_{l} \left[\prod_{i=1}^{n} \exp\left(-\left((z_{i} - \overline{z}_{il}) / \sigma_{il}\right)^{2}\right) \right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left((z_{i} - \overline{z}_{il}) / \sigma_{il}\right)^{2}\right) \right]}$$

$$= C_{F}(r_{k}, r_{k-1}, \dots, u_{k-1}, u_{k-2}, \dots, (W_{C})_{k}, T_{s}),$$
(6a)

$$u_{F}(\hat{z}(k)) = C_{F}(y_{k}, y_{k-1}, \dots, u_{k-1}, u_{k-2}, \dots, (W_{C})_{k}, T_{s}),$$
(6b)

where feedforward terms are reference signals $[r_k, r_{k-1}, ..., r_{k-p}]$, and recursive terms are control signals $[u_{k-1}, u_{k-2}, ..., u_{k-q}]$. Assume that $Y = [y_k, y_{k-1}, ..., y_{k-p}]^T$, $U = [u_k, u_{k-1}, ..., u_{k-q}]$ are measured to train model/controller. *M* is fixed;

$$z_{i} = c_{i0}y_{k} + c_{i1}y_{k-1} + \dots + c_{ip}y_{k-p} + \overline{c}_{i1}u_{k-1} + \overline{c}_{i2}u_{k-2} + \dots + \overline{c}_{iq}u_{k-q}.$$
(7)

The tuned parameter vector of controller is as follows:

$$(W_C)_k = \left[\left(\overline{z}_{11}, \overline{z}_{12}, \dots, \overline{z}_{il}, \dots, \overline{z}_{nM} \right), \\ \left(\sigma_{11}, \sigma_{12}, \dots, \sigma_{il}, \dots, \sigma_{nM} \right), \\ \left(\overline{h}_1, \overline{h}_2, \dots, \overline{h}_l, \dots, \overline{h}_M \right), \left(c_{i0}, c_{i1}, \dots, c_{ip} \right), \\ \left(\overline{c}_{i1}, \overline{c}_{i2}, \dots, \overline{c}_{iq} \right) \right].$$

$$(8)$$

The proposed digital fuzzy controller u_k has feedforward terms and recursive structure. It can use a NARMAX neural model or inverse of the plant dynamics to aid control precision in the face of a delayed plant. Owing to the adaptive fuzzy controller, the uncertainty of the plant can be suppressed. Although the structure of the fuzzy controller is chosen, the parameter vector $(W_C)_k$ is not specified; therefore, the fuzzy



FIGURE 2: Two-stage adaptive prediction structure of DT-CT control system. (a) Two-stage block diagram and (b) online adaptive prediction

controller has not been designed. $\gamma \cdot T_s$ is the tape-delay time, γ is a positive integer. The idea of the inverse-model-based fuzzy controller is proposed by the following simplified relation:

block diagram.

If
$$y_k = \hat{P}(u_k)$$
, $u_k = \hat{P}^{-1}(r_k) = C_F(r_k)$, then $y_k = r_k$,
(9)

where $\hat{P}(\cdot)$ is the adaptive NARMAX neural model of plant; $C_F(\cdot)$ is the adaptive NARMAX fuzzy controller; r_k is the desired output. According to the idea of (9), the recursive structure $\hat{P}(\cdot)$ can be designed with tape delays as follows:

$$y_{k} \approx y_{k}$$

= $\hat{P}(\hat{y}_{k-1}, \hat{y}_{k-2}, \dots, \hat{y}_{k-n}, u_{k}, u_{k-1}, \dots, u_{k-p}, (W_{p})_{k}, T_{s}),$
(10)

but the weights and biases $(W_P)_k$ of this model are trained by the way of the feedforward structure neural network (NN) as follows:

$$y_{k} \approx \widehat{y}_{k}$$

= $\widehat{P}(y_{k-1}, y_{k-2}, \dots, y_{k-n}, u_{k}, u_{k-1}, \dots, u_{k-p}, (W_{p})_{k}, T_{s}).$
(11)

Due to the parameters of the recursive structure of the plant model, convergence may be much stricter, where n, p + 1are the amount of tape delays of \hat{y}, u , respectively. The plant output is compared with the desired output to create a system error signal $e_k = r_k - y_k$. The errors $\hat{e}_k = r_k - \hat{y}_k$ and e_k are used by the adaptation algorithm to update the parameters of \hat{P} and C_F . Next, the performance index is designed, as follows:

$$J_{k} = \frac{1}{2}e_{k}^{T}e_{k} = \frac{1}{2}(r_{k} - y_{k})^{T}(r_{k} - y_{k})$$

$$= \frac{1}{2}(y_{k} - r_{k})^{T}(y_{k} - r_{k}),$$
 (12)

is a cost function to be minimized by the proposed algorithm. Then, the BP algorithm adapts the control parameters $(W_C)_k$. That is, the change in control parameters $(\Delta W_C)_k$ is calculated as

$$(\Delta W_{C})_{k}^{T} = -\eta_{C}(k) \frac{dJ_{k}}{d(W_{C})_{k}}$$

$$= -\eta_{C}(k) \frac{d(y_{k} - r_{k})^{T}(y_{k} - r_{k})}{2 \cdot d(W_{C})_{k}}$$

$$= -\eta_{C}(k) (y_{k} - r_{k}) \frac{dy_{k}}{d(W_{C})_{k}}$$

$$= -\eta_{C}(k) (y_{k} - r_{k}) \frac{dy_{k}}{du_{k}} \frac{du_{k}}{d(W_{C})_{k}},$$
(13)

where the small positive $\eta_C(k)$ can be selected as a stable learning rate via the following theorems.

Theorem 2. If the amount of neurons of the neural model is sufficient and the appropriate sampling time T_s is selected to let $\|\overline{y}_k - y_k\| \le \overline{\varepsilon}$ and the following condition

$$0 < \eta_P(k) < \frac{2}{\left\| d\hat{y}_k / d(W_P)_k \right\|^2} \le \overline{\eta}_P \tag{14}$$

be satisfied, where \overline{y}_k is an output of the optimal model, then the trajectories \hat{y}_k of converging to plant output y_k is a uniformly ultimately bounded (UUB) approximation on the bounded error $\hat{y}_k - y_k$.

Proof of Theorems. First, consider the following ideal Lyapunov candidate [29] of the model part,

$$V_{1k} = \frac{1}{2} (\hat{y}_k - y_k)^T (\hat{y}_k - y_k)$$
$$= \frac{1}{2} \|\hat{y}_k - \overline{y}_k + \overline{y}_k - y_k\|^2$$
$$= \frac{1}{2} \|\hat{y}_k - \overline{y}_k\|^2 + \varepsilon_k = V_{2k} + \varepsilon_k,$$
(15)

where $V_{2k} = (1/2) \|\hat{y}_k - \overline{y}_k\|^2$ is an actual Lyapunov candidate of reachable and assumptive trajectory \overline{y}_k ; the bounded approximation error

$$\varepsilon_{k} = \frac{1}{2} \left\| \overline{y}_{k} - y_{k} \right\|^{2} + \left(\widehat{y}_{k} - \overline{y}_{k} \right)^{T} \left(\overline{y}_{k} - y_{k} \right)$$
(16)

and the number of neurons of the neural model are sufficient, and the appropriate sampling time T_s is selected to

let $\overline{y}_k \approx y_k$. The next task is to train the model such that V_{2k} is minimized,

$$\frac{\left(\Delta W_{P}\right)_{k}}{\eta_{P}\left(k\right)} = -\frac{dV_{2k}}{d\left(W_{P}\right)_{k}}$$

$$= -\left(\widehat{y}_{k} - \overline{y}_{k}\right) \frac{d\left(\widehat{y}_{k} - \overline{y}_{k}\right)}{d\left(W_{P}\right)_{k}}$$

$$= -\left(\widehat{y}_{k} - \overline{y}_{k}\right) \frac{d\widehat{y}_{k}}{d\left(W_{P}\right)_{k}},$$
(17)

where $(\Delta W_p)_k = W_p(k+1) - W_p(k); W_p(k) = (W_p)_k$.

Then, the following Lyapunov candidate of the controller part is designed:

$$V_{3}(k) = V_{3k} = \frac{1}{2} (\hat{y}_{k} - r_{k})^{T} (\hat{y}_{k} - r_{k}) = \frac{1}{2} \|\hat{y}_{k} - r_{k}\|^{2}, \quad (18)$$

Thus, the change in the Lyapunov function is obtained by

$$V_{3}(k+1) - V_{3}(k) = \frac{1}{2} \left(\left\| \hat{y}(k+1) - r(k+1) \right\|^{2} - \left\| \hat{y}(k) - r(k) \right\|^{2} \right).$$
⁽¹⁹⁾

Finally, the update law of the control parameters of the controller is obtained as follows:

$$\frac{\left(\Delta W_{\rm C}\right)_k^T}{\eta_{\rm C}\left(k\right)} \approx -\frac{dV_3\left(k\right)}{d\left(W_{\rm C}\right)_k} = -\left(\hat{y}_k - r_k\right) \frac{d\hat{y}_k}{d\left(W_{\rm C}\right)_k}.$$
 (20)

This study develops some convergence theorems to select appropriate stable learning rates. First, the difference of modeling error $e_P(k) = \hat{y}_k - \overline{y}_k$ can be represented by

$$e_{P}(k+1) = e_{P}(k) - \left[\frac{de_{P}(k)}{d(W_{P})_{k}}\right]^{T} \left[\eta_{P}(k) e_{P}(k) \frac{de_{P}(k)}{d(W_{P})_{k}}\right]$$
$$= e_{P}(k) \left(1 - \left[\frac{de_{P}(k)}{d(W_{P})_{k}}\right]^{T} \eta_{P}(k) \frac{de_{P}(k)}{d(W_{P})_{k}}\right)$$
$$= e_{P}(k) \left(1 - \left[\frac{d\hat{y}_{k}}{d(W_{P})_{k}}\right]^{T} \eta_{P}(k) \frac{d\hat{y}_{k}}{d(W_{P})_{k}}\right);$$
(21)

thus, the change in the Lyapunov function is obtained by

$$V_{2} (k + 1) - V_{2} (k)$$

= $\frac{1}{2} (||e_{P} (k + 1)||^{2} - ||e_{P} (k)||^{2})$

$$= \frac{1}{2} \left(\left\| e_{P}\left(k\right) \left(1 - \left[\frac{d\hat{y}_{k}}{d(W_{P})_{k}} \right]^{T} \eta_{P}\left(k\right) \frac{d\hat{y}_{k}}{d(W_{P})_{k}} \right) \right\|^{2} - \left\| e_{P}\left(k\right) \right\|^{2} \right)$$
$$= \frac{1}{2} \left\| e_{P}\left(k\right) \right\|^{2} \left[\left(1 - \left[\frac{d\hat{y}_{k}}{d(W_{P})_{k}} \right]^{T} \eta_{P}\left(k\right) \frac{d\hat{y}_{k}}{d(W_{P})_{k}} \right)^{2} - 1 \right].$$
(22)

Hence, if $-1 < (1 - [d\hat{y}_k/d(W_p)_k]^T \eta_p(k)(d\hat{y}_k/d(W_p)_k)) < 1$ and $\|\overline{y}_k - y_k\| \le \overline{\varepsilon}$, then $V_2(k+1) < V_2(k)$, that is, $V_2(k) \to 0$ or $\hat{y}_k \to \overline{y}_k$ makes the UUB approximation of this model on the bounded $\hat{y}_k - y_k$. The proof is thereby completed. \Box

Furthermore, the following theorem for the convergence of the controller is obtained by the same procedure as the above proof.

Theorem 3. If Theorem 2 in (14) is satisfied, the function $d\hat{y}_k/d(W_C)_k$ in (20) is computed to let the following condition

$$0 < \eta_C(k) < \frac{2}{\left\| d\hat{y}_k / d(W_C)_k \right\|^2} \le \overline{\eta}_C, \tag{23}$$

be satisfied. Then, the nonlinear systems (1a) and (1b) are UUB stable, and the tracking errors $e(k) = r_k - y_k$ are bounded via the controller.

Hence, the dynamic response of the system P can be controlled using C_F , as shown in Figure 2. This C_F needs the plant model \hat{P} to adjust control parameters via sensitivity function $\partial \hat{y}_k / \partial u_{k-i}$.

The digital feedback controller includes a delay block D, as shown in Figure 2. Here, the error \tilde{e}_{k+1} is used to estimate u_X , and the proposed predictor of the delayed system can let us cancel some complex computations, such as $\partial \hat{y}_k / \partial u_k \approx (\hat{y}_k - \hat{y}_{k-1})/\Delta u_k = \Delta \hat{y}_k / \Delta u_k = \Delta \hat{y}_k / u_X(k)$, of sensitivity function $\partial \hat{y}_k / \partial u_{k-i}$ in the BP algorithm. Hence, the following theorem is proposed to update the control parameters of FRP under the assumption of providing a model which applies a lower prediction error and a more correct u_X . The prediction error $\hat{e}_{k+1} = r_{k+1} - \hat{y}_{k+1}$, being bounded due to the previous $\hat{e}_k = r_k - \hat{y}_k$, is bounded at any time. Hence, the prediction error \hat{e}_{k+1} will be bounded by using Theorems 2 and 3. Furthermore, the following theorem is obtained for the convergence of the adaptive prediction controller by the same procedure as Theorem 2.

Theorem 4. If Theorem 2 in (14) is satisfied, the predictive function $du_{k+1}/d(W_C)_{k+1}$ is computed to let the following condition

$$0 < \eta_X(k+1) < \frac{2}{\|du_{k+1}/d(W_C)_{k+1}\|^2} \le \overline{\eta}_X$$
(24)

be satisfied, then the nonlinear systems (1a) and (1b) are UUB stable, and the tracking errors $e_k = r_k - y_k$ are bounded via the predictive controller $u_{k+1} = u_F(k+1) + s \cdot u_X(k+1)$.

The tracking error is $e_k = r_k - y_k$ and $e_k = [e_{1k}, e_{2k}, e_{3k}, ...]^T$, but the parameters of adaptive control C_F are updated by using the predictive offset

$$\Delta u_{k+1} = u_X (k+1) = C_X (e_k)$$

= $[K_1 (e_{1k}), K_2 (e_{2k}), K_3 (e_{3k}), \dots]^T,$ (25)

so its previous term is

$$\Delta u_{k} = u_{X}(k) = C_{X}(e_{k-1})$$

$$= \left[K_{1}(e_{1(k-1)}), K_{2}(e_{2(k-1)}), K_{3}(e_{3(k-1)}), \ldots\right]^{T},$$
(26)

of control input u_k , where $K_1(\cdot), K_2(\cdot), K_3(\cdot), \ldots$ are defined by the user, and the predictive controller is $u_{k+1} = u_F(k+1) + s \cdot u_X(k+1)$, so its previous term is $u_k = u_F(k) + s \cdot u_X(k)$.

3.2. Two-Stage Scheme in the Presence of Stability. Figure 2 shows a block-diagram of an adaptive recursive control system. The system to be controlled is labeled as the plant P, which is subject to modeling errors, uncertainties, and delays. Due to gradient-descent-based training algorithms, let the model/controller converge to some local minimums which destroy the control system stability by enlarging the modeling error. Hence, we proposed the two-stage training algorithm to reduce this modeling error as follows.

In the first stage, the measured data is used to train the global optimal NARMAX plant and the fuzzy controller by the training-data-shuffle method. This method shuffles the training data to avoid most of the local optimal solutions obtained by the offline training procedure in the next section. The measured data is divided into a training data and other testing data. This testing data is not used for training the NN. However, the final performance of the NN is decided by the testing data and the training data.

In the second stage, the global optimal NARMAX plant model and fuzzy controller is adapted. The two stages are divided into the following five steps.

Step 1. First, the reference signal $r(k) = r_k$ is designed. By the white noise of input $u(k) = u_k$ for plant, output data $y(k) = y_k$ is collected, and a training-data-shuffle method is used to shuffle the input/output pairs' data. These shuffled data are ready to train the NARMAX model/controller. Here, the following reasonable conditions need to be taken into account:

$$\max_{k} (r_k) \approx \max_{k} (y_k), \qquad \min_{k} (r_k) \approx \min_{k} (y_k),
\max_{k} (u_k) \le u_U, \qquad \min_{k} (u_k) \ge u_L$$
(27)

need be satisfied, where u_U is the upper bound of u_k , and u_L is the lower bound of u_k . According to (27), much of the excessive control effort u_k can be avoided. If (27) is satisfied, then go to Step 2.

Step 2. The feedforward structure model \hat{P} is trained/tested offline

$$\widehat{y}(k) = \widehat{P}\left(S_{u}u_{k}, S_{u}u_{k-1}, \dots, S_{u}u(k-p_{u}), S_{y}y_{k-1}, S_{y}y_{k-2}, \dots, S_{y}y(k-p_{y}), (W_{P})_{k}, T_{s}\right) \left(\frac{1}{S_{y}}\right),$$
(28)

via the shuffled input/output pairs' data. After system identification, \hat{P} is performed, and the digital fuzzy controller C_F for the CT system can be built by using this inverse NARMAX plant model \hat{P}^{-1} in the next step.

Step 3. In practice, according to the exchanged output/input pairs' data from Step 2, the offline stage to train/test the fuzzy controller can be passed through

$$u_{k} = C_{F} \left(S_{u} u_{k-1}, S_{u} u_{k-2}, \dots, S_{u} u \left(k - c_{u} \right), S_{y} y_{k}, \right.$$

$$S_{y} y_{k-1}, \dots, S_{y} y \left(k - c_{y} \right), \left(W_{C} \right)_{k}, T_{s} \left(\frac{1}{S_{u}} \right).$$
(29)

If (28) and (29) work, go to Step 4.

Step 4. Update the online weights and biases W_P of the recursive structure model \hat{P} :

$$\widehat{y}(k) = \widehat{P}\left(S_{u}u_{k}, S_{u}u_{k-1}, \dots, S_{u}u(k-p_{u}), S_{y}\widehat{y}_{k-1}, S_{y}\widehat{y}_{k-2}, \dots, S_{y}\widehat{y}(k-p_{y}), (W_{P})_{k}, T_{s}\right) \left(\frac{1}{S_{y}}\right),$$
(30)

to approximate the CT nonlinear system by using Theorem 2. Due to the adaption laws for (28) and (30), an exchange for both of them can be designed to switch into the system, as a switching in Figure 2, when (30)'s absolute approximation error is too big. If (28) and (30) work, go to Step 5.

Step 5. Adapt the digital fuzzy controller for the modeling error and tracking error by using Theorems 2 and 3. Finally, update the online parameters of the fuzzy controller C_F

$$u_{k} = C_{F} \left(S_{u} u_{k-1}, S_{u} u_{k-2}, \dots, S_{u} u \left(k - c_{u} \right), \right.$$

$$S_{y} r_{k}, S_{y} r_{k-1}, \dots, S_{y} r \left(k - c_{y} \right), \left(W_{C} \right)_{k}, T_{s} \left(\frac{1}{S_{u}} \right),$$
(31)

to minimize the tracking error, and finish the above two stages: the offline stage and online stage.

To make sure of the robustness of the control system, the convergence to the global optimal solution of parameters of the model/controller has to be guaranteed. Hence, some random initial weights and biases of the model are designed by particle swarm optimization (PSO) [24] with the parameters of the controller first. The PSO algorithm consists of the velocity

$$v_{i}(j+1) = v_{i}(j) + \gamma_{1i} \cdot (p_{i} - \overline{x}_{i}(j)) + \gamma_{2i} \cdot (G - \overline{x}_{i}(j)),$$
(32)

and position

$$\overline{x}_{i}(j+1) = \overline{x}_{i}(j) + v_{i}(j+1), \qquad (33)$$

where i = 1, 2, ..., H is the particle index; j = 1, 2, ..., N is the iteration index; v_i is the velocity of *i*th particle; \overline{x}_i is the position of *i*th particle; p_i is the best position found by *i*th particle (personal best); *G* is the best position found by the swarm (global best, best of personal best); γ_{1i} , γ_{2i} are the random numbers on the interval [0, 1] applied to the *i*th particle.

The PSO supplies random initial parameters, hence, it is an initial parameters' conductor. These initial parameters are then converged locally by the BP method and the best solution for the initial model/controller is chosen. Finally, the global optimal solution of parameters can be found every time. Hence, this idea has been named the randomlocal-optimization (RLO) algorithm. The RLO algorithm is a composite of the BP algorithm and a random initialization procedure of evaluating fitness value $1/(\Xi + 0.01)$, where $\Xi =$ $\rho \cdot \Xi_1 + (1 - \rho) \cdot \Xi_2, \rho \in [0, 1]$. The total of absolute training error Ξ_1 is obtained by BP via the training data, and Ξ_2 is the total of absolute testing error of the model/controller output via the testing data input. In this paper, offline RLO is used as a learning algorithm for the feedforward structure model (28) due to the online tuning parameters of the recursive structure of the plant model being not converged. After the offline training stage, in order to tune the online parameters of the plant model (30) recursively, $d\hat{y}_k/d(W_P)_k$ of (14) needs to be calculated as follows:

$$\frac{d\hat{y}_k}{d(W_P)_k} = \frac{\partial\hat{y}_k}{\partial(W_P)_k} + \sum_{i=0}^{p_u} \frac{\partial\hat{y}_k}{\partial u_{k-i}} \frac{du_{k-i}}{d(W_P)_k} + \sum_{i=1}^{P_y} \frac{\partial\hat{y}_k}{\partial\hat{y}_{k-i}} \frac{d\hat{y}_{k-i}}{d(W_P)_k}.$$
(34)

Similarly, in order to tune the online parameters of the controller (31) recursively, and $d\hat{y}_k/d(W_C)_k$ of (24) needs to be calculated as follows:

$$\frac{d\hat{y}_k}{d(W_C)_k} = \sum_{i=0}^{p_u} \frac{\partial\hat{y}_k}{\partial u_{k-i}} \frac{du_{k-i}}{d(W_C)_k} + \sum_{i=1}^{P_y} \frac{\partial\hat{y}_k}{\partial\hat{y}_{k-i}} \frac{d\hat{y}_{k-i}}{d(W_C)_k}, \quad (35)$$

where

$$\frac{du_k}{d(W_C)_k} = \frac{\partial u_k}{\partial (W_C)_k} + \sum_{i=1}^{c_u} \frac{\partial u_k}{\partial u_{k-i}} \frac{du_{k-i}}{d(W_C)_k}.$$
 (36)

Hence, the following algorithm adapts a NARMAX neural controller for a NARMAX neural model of plant.

Step 1. Back propagate through C_F to form $\partial u_k / \partial u_{k-i}$ and $\partial u_k / \partial (W_C)_k$ in (36). If update $du_k / d(W_C)_k$ of (36) and shift $du_{k-i}/d(W_C)_k$ down in (35), then go to Step 2.



FIGURE 3: (a) The learning curve of the summation of W_p , (b) the learning curve of the summation of W_C , (c) the tracking control performance, (d) the modeling error $\hat{y}_k - y_k = y_{Nk} - y_k$, (e) the trajectories of predictive input signal u_X , and (f) total input signal u of Case 3.



FIGURE 4: (a) The tracking control performance and (b) the parameters ϕ , *a*, *b* whose learning curves of Case 2.

Step 2. Back propagate through \widehat{P} to form $\partial \widehat{y}_k / \partial u_{k-i}$ and $\partial \widehat{y}_k / \partial \widehat{y}_{k-i}$ in (35). If update $d \widehat{y}_k / d(W_C)_k$ of (35) and shift $d \widehat{y}_{k-i} / d(W_C)_k$ down in (35), then go to Step 3.

Step 3. Compute $(\Delta W_C)_k^T = -\eta_C(k)(\hat{y}_k - r_k)(d\hat{y}_k/d(W_C)_k)$. If we update weights

$$(W_C)_{k+1} = (W_C)_k + (\Delta W_C)_k,$$

$$(W_P)_{k+1} = (W_P)_k + (\Delta W_P)_k,$$
(37)

then go to Step 1.

To clarify this method, in [20], a robust and adaptive method was used to allow learning to occur online, tuning performance as the system runs. But, [20] did not consider the prediction, modeling error, global optimal initialization of control parameters, the problem of lengthy convergence time of online control, delayed terms, uncertainties in plant, and DT-CT problems. Moreover, the choice method of initial parameters of the online controller still lacks the ability to overcome the overfitting problem of the controller. Hence, the offline stage is proposed for an RLO learning algorithm to choose the initial weights and biases of the online neural controller in the simulation example of the power plant, as shown in the following case study.

4. Cases Study

First, the conventional PWM buck converter, by using AM-OTS-DS [28] methodology, is modeled to the equivalent circuit plant. In this paper, the robustness of this control system is emphasized, so uncertainty and delay are considered in (1b). The state is $x(t) = x = [i_L(t), v_C(t)]^T$, and the output is $y(t) = v_o(t)$ in this power system.

Referring to Figure 2, and the above sections about twostage control structure, it can be seen how to model the CT plant dynamics from Figure 2(a) by considering the modeling error, and how to use the DT neural model of plant to adapt a DT fuzzy controller and predict the DT compensation of total control input u_k from Figure 2(b).

To compare with other methods, the following cases are introduced.

Case 1. This case is in [28], and its digital controller is a kind of fuzzy controller with integral term shown in (38).

Case 2. This case is in [30], and its controller is a kind of single neuron shown in (39) without multilayer structure, and this CT controller is discretized to DT controller to compare with Case 3.

Case 3. This is the control method presented here, and the proposed neural-model-based fuzzy controller is adaptive, predictive, and globally optimal.

The detail designs of the Cases 1–3 are as follows.

Case 1 is an LMI control method of original example for this power plant. The control parameters of Case 1 are solved by the LMI toolbox as follows.

 $K_{11} = K_{21} = [0.0476, 0.9348]^T$ and $K_{12} = K_{22} = -426.4969$ are referred to [28], and this T-S fuzzy controller of Case 1 is designed as

$$u_{k} = -h_{1k} \cdot \left(K_{11}x_{k} + K_{12}\sum_{k}e_{k}T_{s} \right)$$

$$-h_{2k} \cdot \left(K_{21}x_{k} + K_{22}\sum_{k}e_{k}T_{s} \right),$$
(38)

where $\sum_{k} e_k T_s$ is integral term.

Case 2 is an adaptive control method. This neural controller is designed as

$$u_k = \frac{a \cdot \left(1 - e^{-b \cdot \operatorname{net}}\right)}{1 + e^{-b \cdot \operatorname{net}}}, \quad \operatorname{net} = I - \phi, \tag{39}$$

where $I = e_k$ is a net input of single neuron.



FIGURE 5: (a)(b) Comparison of the output trajectories for Cases 1 to 3, and (b) is the detail view of (a). (c) Comparisons of $e_k = r_k - y_k$ and (d) $\sum_k |e_k|$ of the control performances for Cases 1 to 3.

And the design parameters $P_2 = 1200$ and learning rate $\eta = 1$ for this neuron of Case 2 are referred to [30].

Case 3 is also an adaptive control method, but its NARMAX fuzzy control design method with neural-modelbased prediction is very different from Case 2. The predictive controller of Case 3 is $u_X = K_1 e_k$, where $K_1 = 0.1$.

First, the sampling time of the model/controller is set for $T_s = 10^{-4}$ sec to simulate the proposed adaptive prediction control system, the initial state $x(0) = [0,0]^T$, and the reference signal $r_k = 12 \sin(30k \cdot T_s)$. Case 3 uses NN structure 5-8-1 of the NARMAX plant model, it has 5 inputs, $[u_k, u_{k-1}, u_{k-3}, \hat{y}_{k-1}, \hat{y}_{k-2}]$, 8 tansig(\cdot) neurons in the hidden layer, and 1 purelin(\cdot) neuron in the output \hat{y}_k layer.

The adaptive weights and biases, W_P , are trained as follows by selecting a suitable scaling factor $S_u = 1$, $S_y = 1/24$ of the plant model. The summation W_{PS} of adaptive plant model parameters W_P is updated as shown in Figure 3(a). The summation W_{CS} of adaptive fuzzy control parameters W_C is updated as shown in Figure 3(b). The tracking control performances $e_k = r_k - y_k$ and $\sum_k |e_k|$ of Case 3 are shown in Figures 3(c) and 5. The bounded modeling error of Case 3 is shown in Figure 3(d). The trajectory of predictive input signal u_X of Case 3 is shown in Figure 3(e). Note that this predictive control input signal is just the minor portion of total control signal. It works to control the CT system precisely. To clarify this difference between predictive control input and total control input, the figure of total control signal is shown in Figure 3(f).

And the tracking control performance of Case 2 is shown in Figures 4(a) and 4(c), and the controller of Case 2 includes the adaptive parameters ϕ , *a*, *b* whose learning curves are shown in Figure 4(b). From Figures 3(d) and 5(d), the offline training scheme of Case 3 does let the neural model be optimized globally to make the adaptive laws of Case 3 converge faster than that of Case 2. Therefore, the more precise result from Case 3 is made as shown in Figure 5.

Finally, the control performances of Cases 1–3 are compared, as shown in Figure 5. Figure 3 shows the precise fuzzy control performance of Case 3. Figure 4 shows the digital adaptive neural control performance of Case 2. Figure 5 shows the LMI control performance of Case 1. From the control results of Cases 1 and 3 shown in Figure 5(d), the proposed adaptive and predictive laws do let the proposed controllers adapt and predict themself to overcome these problems about robustness as the delay, uncertainty, and nonlinearity.

It is clear that the two-stage scheme, Case 3, for the proposed adaptive controller and predictive controller has excellent tracking performance when compared with Cases 1 and 2.

5. Conclusion

The proposed two-stage adaptive prediction control converges very fast, works highly effective, and precise. It simplifies the complex model-based adaptive control design, and works for nonlinear delayed plants with uncertainty. The proposed recursive and feedforward control scheme is partitioned into two stages that can be independently optimized. First, an offline neural model of continuous-time (CT) nonlinear power plant is made. Second, a constrained offline digital fuzzy controller is generated; then, an adaptive plant model is made, and an adaptive NARMAX prediction tracker is generated. Finally, all processes may continue concurrently, and robustness and adaptive prediction design with DT-CT problems for a power plant are solved. Although this power system is only a simulation, the control strategy can be extended to LED dimmer systems and time-delay robot systems based on visual servo, and is within my plans of future research.

Acknowledgments

The author would like to thank the National Science Council and Asia University for the support under Contracts nos. NSC-97-2218-E-468-009, NSC-98-2221-E-468-022, NSC-99-2628-E-468-023, NSC-100-2628-E-468-001, NSC-101-2221-E-468-024, 98-ASIA-02, 100-asia-35, and 101-asia-29.

References

- H. Dong, Z. Wang, D. W. C. Ho, and H. Gao, "Robust H_{co} fuzzy output-feedback control with multiple probabilistic delays and multiple missing measurements," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 4, pp. 712–725, 2010.
- [2] H. O. Wang, K. Tanaka, and M. F. Griffin, "An approach to fuzzy control of nonlinear systems: stability and design issues," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 1, pp. 14–23, 1996.
- [3] L. Wu, X. Su, P. Shi, and J. Qiu, "Model approximation for discrete-time state-delay systems in the TS fuzzy framework," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 2, pp. 366–378, 2011.
- [4] C. H. Sun, Y. T. Wang, and C. C. Chang, "Switching T-S fuzzy model-based guaranteed cost control for two-wheeled mobile robots," *International Journal of Innovative Computing*, *Information and Control*, vol. 8, pp. 3015–3028, 2012.
- [5] C. H. Sun, S. W. Lin, and Y. T. Wang, "Relaxed stabilization conditions for switching T-S fuzzy systems with practical

constraints," International Journal of Innovative Computing, Information and Control, vol. 8, pp. 4133–4145, 2012.

- [6] X. Su, P. Shi, L. Wu, and Y. D. Song, "A novel control design on discrete-time Takagi-Sugeno fuzzy systems with time-varying delays," *IEEE Trans on Fuzzy Systems*, no. 99, 2013.
- [7] X. Su, L. Wu, and P. Shi, "Sensor networks with random link failures: distributed filtering for T-S fuzzy systems," *IEEE Transactions on Industrial Informatics*, no. 99, 2012.
- [8] Z. Wang, Y. Liu, G. Wei, and X. Liu, "A note on control of a class of discrete-time stochastic systems with distributed delays and nonlinear disturbances," *Automatica*, vol. 46, no. 3, pp. 543–548, 2010.
- [9] F. Li and X. Zhang, "Delay-range-dependent robust H_∞ filtering for singular LPV systems with time variant delay," *International Journal of Innovative Computing, Information and Control*, vol. 9, pp. 339–353, 2013.
- [10] R. Yang, H. Gao, and P. Shi, "Delay-dependent robust H_{∞} control for uncertain stochastic time-delay systems," *International Journal of Robust and Nonlinear Control*, vol. 20, no. 16, pp. 1852–1865, 2010.
- [11] R. Yang, P. Shi, G.-P. Liu, and H. Gao, "Network-based feedback control for systems with mixed delays based on quantization and dropout compensation," *Automatica*, vol. 47, no. 12, pp. 2805–2809, 2011.
- [12] X. Su, P. Shi, L. Wu, and S. K. Nguang, "Induced *l*2 filtering of fuzzy stochastic systems with time-varying delays," *IEEE Transactions on Systems, Man, and Cybernetics B*, no. 99, pp. 1– 14, 2012.
- [13] L. Wu, X. Su, P. Shi, and J. Qiu, "A new approach to stability analysis and stabilization of discrete-time T-S fuzzy timevarying delay systems," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 41, no. 1, pp. 273–286, 2011.
- [14] B. S. Chen, C. S. Tseng, and H. J. Uang, "Mixed H_2/H_{∞} fuzzy output feedback control design for nonlinear dynamic systems: an LMI approach," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 3, pp. 249–265, 2000.
- [15] K. Tanaka, "An approach to stability criteria of neural-network control systems," *IEEE Transactions on Neural Networks*, vol. 7, no. 3, pp. 629–642, 1996.
- [16] X. Su, L. Wu, P. Shi, and Y. D. Song, " H_{∞} model reduction of T-S fuzzy stochastic systems," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 42, pp. 1574–1585, 2012.
- [17] H. T. Siegelmann, B. G. Horne, and C. L. Giles, "Computational capabilities of recurrent NARX neural networks," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 27, no. 2, pp. 208–215, 1997.
- [18] K. R. Lee, J. H. Kim, E. T. Jeung, and H. B. Park, "Output feedback robust H_{∞} control of uncertain fuzzy dynamic systems with time-varying delay," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 6, pp. 657–664, 2000.
- [19] Y. Y. Cao and P. M. Frank, "Analysis and synthesis of nonlinear time-delay systems via fuzzy control approach," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 2, pp. 200–211, 2000.
- [20] G. L. Plett, "Adaptive inverse control of linear and nonlinear systems using dynamic neural networks," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 360–376, 2003.
- [21] F. J. Lin, W. J. Hwang, and R. J. Wai, "A supervisory fuzzy neural network control system for tracking periodic inputs," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 1, pp. 41–52, 1999.
- [22] C. Li and K.-H. Cheng, "Recurrent neuro-fuzzy hybrid-learning approach to accurate system modeling," *Fuzzy Sets and Systems*, vol. 158, no. 2, pp. 194–212, 2007.

- [23] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Prentice-Hall, New York, NY, USA, 1997.
- [24] T. Niknam, H. D. Mojarrad, and M. Nayeripour, "A new hybrid fuzzy adaptive particle swarm optimization for nonconvex economic dispatch," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 1, pp. 189–202, 2011.
- [25] Y. C. Ho, W. C. Hsu, and C. C. Chang, "Multi-category and multi-standard project selection with fuzzy value-based time limit," *International Journal of Innovative Computing, Information and Control*, vol. 9, pp. 971–989, 2013.
- [26] C. M. Lin, C. F. Hsu, and R. G. Yeh, "Adaptive fuzzy slidingmode control system design for brushless DC motors," *International Journal of Innovative Computing, Information and Control*, vol. 9, pp. 1259–1270, 2013.
- [27] H. M. Lee, C. F. Fuh, and J. S. Su, "Fuzzy parallel system reliability analysis based on level (λ , ρ) interval-valued fuzzy numbers," *International Journal of Innovative Computing, Information and Control*, vol. 8, pp. 5703–5713, 2012.
- [28] C. Y. Huang, T-S fuzzy controller design for DC-DC power converter [M.S. thesis], Chung Yuan Christian University, Taiwan, 2002.
- [29] J. P. LaSalle, "Some extensions of Liapunov's second method," vol. 7, pp. 520–527, 1960.
- [30] W. D. Chang, R. C. Hwang, and J. G. Hsieh, "Stable direct adaptive neural controller of nonlinear systems based on single auro-tuning neuron," *Neurocomputing*, vol. 48, pp. 541–554, 2002.



The Scientific World Journal





Decision Sciences







Journal of Probability and Statistics



Hindawi Submit your manuscripts at http://www.hindawi.com



(0,1),

International Journal of Differential Equations





International Journal of Combinatorics





Mathematical Problems in Engineering



Abstract and Applied Analysis



Discrete Dynamics in Nature and Society







Function Spaces



International Journal of Stochastic Analysis



Journal of Optimization