Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A Fuzzy Neural Network Direct Adaptive Iterative Learning Controller for Robotic Systems ⋆

Y.-C. Wang * **C.-J. Chien** ** **D.-T. Lee** ***

* *Institute of Information Science, Academia Sinica, 115, Nankang, Taipei, Taiwan.*
** *Corresponding Author, Department of Electronic Engineering, Huafan University, 223, Shihding, Taipei County, Taiwan. (e-mail : cjc@huafan.hfu.edu.tw)*
*** *Institute of Information Science, Academia Sinica, 115, Nankang, Taipei, Taiwan.*

Abstract: This paper studies the iterative learning control of robotic systems with repetitive tasks. A fuzzy neural network is applied to design a direct adaptive iterative learning controller. The fuzzy neural network is introduced for compensation of the unknown certainty equivalent controller. A new adaptive law using mixed time-domain and iteration-domain adaptation is developed. It is shown that the finiteness of control parameters and control input can be guaranteed for all the time interval during each iteration without using parameter projection.

1. INTRODUCTION

In the early works [1]-[3], classical PD and PID linear controllers were widely used in robotics applications to asymptotically stabilize the joint positions of rigid robot manipulators at a given set-point. Owing to the physical property that the robot parameters enter linearly in the Lagrange equation, adaptive control strategies [4, 5] have been derived for trajectory tracking. On the other hand, taking advantage of the fact that robot manipulators are generally used in repetitive tasks, several iterative learning control (ILC) schemes have been presented for repetitive control of robot manipulators in the past two decades. The ILC approach iteratively tunes the control input in order to enhance the tracking accuracy from operation to operation for systems executing repetitive tasks. Initially, ILC algorithms for robot manipulators were developed based on the contraction mapping theory and required a certain *a priori* knowledge of robot dynamics [6]-[10]. Recently, another type of ILC algorithms, namely adaptive iterative learning control (AILC), has been deployed during the last decade (see, for instance, [11]-[15]) for robot manipulators. The main feature of AILC is to iteratively estimate the uncertain parameters, which are in turn used to generate the current control input. Instead of using contraction mapping theory, the Lyapunov-like approach is applied to analyze the stability and convergence so that the restrict Lipschitz condition can be relaxed.

However, most of the adaptive learning controllers are designed based on the fact that the robot nonlinearities are linearly parameterizable. This often leads to the problem of over-parametrization. To this end, fuzzy system or neural network based controllers [16]-[22] have become an effective approach for adaptive control of nonlinear systems if the nonlinearity can not be linearly parameterizable. Recently, the controller design based on fuzzy system or neural network was applied to iterative learning control of uncertain robotic systems or nonlinear dynamic systems [23]-[25]. Since the fuzzy system and neural network are employed to model the nonlinearities, these schemes [23]-[25] can be considered as an indirect AILC similar to that defined in [18].

As we know, both fuzzy system and neural network are to mimic human-like knowledge processing capability. To obtain the advantages of both, such as the low-level learning and computational power of neural network and the high-level human-like thinking and reasoning of fuzzy inference system, fuzzy neural network (FNN) has become a popular research topic in a variety of applications. In this paper, we apply the FNN to design a direct AILC (DAILC) for uncertain robotic systems. The FNN is introduced for compensation of the unknown certainty equivalent controller. Using this direct scheme, only one fuzzy neural network is required to design the iterative learning controller. It is well known that the whole control parameter profiles in the previous iteration must be stored for an adaptive iterative learning controller. A complex control structure implies that it requires more control parameters and a large system memory. In this work, the proposed FNN-DAILC uses only one FNN and possesses simpler structure, especially compared with the related works [23]-[25].

Because of the iteration based control problem, the adaptive learning laws for the estimation of the unknown parameters are mostly designed in the iteration-domain. In general, projection or deadzone mechanisms are necessary to construct the iteration-domain based adaptive laws in order to guarantee the tracking error convergence as well as the boundedness of all internal signals. In [14] both time-domain and iteration-domain adaptations were used.

⋆ This work is supported by the National Science Council, R.O.C., under Grant NSC96-2221-E-211-019

A time-domain adaptive law estimates the robot parameters so that the upper bounds on these parameters are not necessary. However, the iteration-domain learning law which learns the desired input and disturbances still needs the upper bound and the projection mechanism. In this paper, a new adaptive law using mixed time-domain and iteration-domain adaptation is developed in this paper to relax the requirement of projection mechanism. It will become a pure time-domain learning law or iteration-domain learning law if a weighting gain is suitably chosen. In other words, the proposed adaptive law is a general formulation and extension for the existing results [11]-[15]. We show that the finiteness of control parameters and control input can be guaranteed for all the time interval during each iteration without using parameter projection. This implies that the upper bounds on the desired unknown control parameters are not necessary. A rigorous proof via the technique of Lyapunov-like analysis is given to guarantee the stability and convergence of the closed-loop learning system. It is shown that all adjustable parameters as well as internal signals are bounded in time domain for each iteration. Furthermore, the position and velocity tracking error will asymptotically converge to zero in iteration domain if iteration number is large enough.

2. DESIGN OF FUZZY NEURAL NETWORK DIRECT ADAPTIVE ITERATIVE LEARNING CONTROLLER

In this paper, we consider an uncertain robot system with n rigid bodies which can perform a given task repeatedly over a finite time interval $[0,T]$ as follows:

$$\begin{aligned} D(q^j(t))\ddot{q}^j(t) + B(q^j(t),\dot{q}^j(t))\dot{q}^j(t) \\ +f(q^j(t),\dot{q}^j(t)) + d^j(t) = u^j(t) \end{aligned} \quad (1)$$

where $j \in \mathcal{Z}_+$ denotes the index of iteration number and $t \in [0,T]$ denotes the time index. The signals $q^j(t), \dot{q}^j(t), \ddot{q}^j(t) \in R^n$ are the unknown generalized joint position, joint velocity and joint acceleration vectors, respectively. $D(q^j(t)) \in R^{n\times n}$ is the unknown inertia matrix, $B(q^j(t),\dot{q}^j(t)) \in R^{n\times n}$ is the centripetal plus Coriolis force vector, $f(q^j(t),\dot{q}^j(t)) \in R^n$ is the unknown gravitational plus frictional forces, $d^j(t) \in R^n$ is an unknown disturbance vector, and $u^j(t) \in R^n$ is the joint torque vector. Given the specified desired joint position, velocity, acceleration trajectories $q_d(t)$, $\dot{q}_d(t)$ and $\ddot{q}_d(t)$ $\forall t \in [0,T]$, the control objective is to design a fuzzy-neural direct adaptive iterative learning controller $u^j(t)$ such that when iteration number j is large enough, $\|q^j(t) - q_d(t)\|$ and $\|\dot{q}^j(t) - \dot{q}_d(t)\|$ will converge to some small positive error tolerance bounds $\forall t \in [0,T]$ even there exists some bounded non-repeatable disturbance $d^j(t)$ and initial resetting errors. Here the initial resetting errors mean that $q_d(0) \neq q^j(0)$ or $\dot{q}_d(0) \neq \dot{q}^j(0)$ for all $j \geq 1$. In order to achieve the above control objective, some assumptions on the uncertain robot system and desired trajectories are given as follows:

(A1) The nonlinear functions $D(q^j(t))$, $B(q^j(t),\dot{q}^j(t))$ and $f(q^j(t),\dot{q}^j(t))$ are bounded if $q^j(t)$ and $\dot{q}^j(t)$ are bounded. In addition, the disturbance vector $d^j(t)$ is also bounded.

(A2) The symmetric inertia matrix $D(q^j(t))$ is assumed to be positive definite and bounded for all $t \in [0,T]$ and iteration $j \geq 1$ as $0 < \lambda_1 I \leq D(q^j(t)) \leq \lambda_2 I$ where $\lambda_1, \lambda_2 > 0$ and I is an $n \times n$ identity matrix. The matrix $\dot{D}(q^j(t)) - 2B(q^j(t),\dot{q}^j(t))$ is assumed to be skew-symmetric, that is, $x^\top(\dot{D}(q^j(t)) - 2B(q^j(t),\dot{q}^j(t)))x = 0$ for all $x \in R^n$ and $x \neq 0$.

(A3) The desired joint position, velocity, acceleration trajectories $q_d(t)$, $\dot{q}_d(t)$ and $\ddot{q}_d(t)$, $t \in [0,T]$ are bounded and contained in the compact set $\mathcal{A}_c$.

(A4) Let the initial resetting errors $e_1^j(t), e_2^j(t)$ be defined as $e_1^j(t) = q^j(t) - q_d(t), e_2^j(t) = \dot{q}^j(t) - \dot{q}_d(t)$. The initial resetting errors at each iteration are not necessarily zero, small and fixed, but assumed to be bounded.

Now, in order to illustrate the idea of the learning controller, we use the following three steps to explain the design approach [1].

• **Step 1**. We first design a control function s^j as a linear combination of the tracking errors, i.e.,

$$s^j = e_2^j + \lambda e_1^j = \dot{e}^j + \lambda e^j \quad (2)$$

where $e^j \equiv q^j - q_d$, λ is a diagonal positive define matrix. It is clear that if the learning controller can drive $s^j(t)$ to zero for all $t \in [0,T]$, then the tracking errors will also asymptotically converge to zero for all $t \in [0,T]$. However, there exist initial resetting errors such that $s^j(0) \neq 0$. In order to overcome the uncertainty from the bounded initial resetting errors, let ε^j be the known constant satisfying $\|s^j(0)\| = \|\dot{e}^j(0) + \lambda e^j(0)\| \equiv \varepsilon^j$ by assumption (A4), and introduce the following error function s_ϕ^j as

$$s_\phi^j = s^j - \phi^j \mathbf{sat}\left(\frac{s^j}{\phi^j}\right),\ \phi^j = \varepsilon^j e^{-kt},\ k > 0 \quad (3)$$

where $\mathbf{sat}(\frac{s^j}{\phi^j}) = \left[\mathbf{sat}(\frac{s_1^j}{\phi^j}), \cdots, \mathbf{sat}(\frac{s_n^j}{\phi^j})\right]^\top$ and each element of the saturation function is defined as

$$\mathbf{sat}\left(\frac{s_i^j}{\phi^j}\right) = \begin{cases} 1 & \text{if } s_i^j > \phi^j \\ \dfrac{s_i^j}{\phi^j} & \text{if } |s_i^j| \leq \phi^j \\ -1 & \text{if } s_i^j < -\phi^j \end{cases}$$

Note that ϕ^j is the width of boundary layer, and it is designed to decrease along time axis with the initial condition chosen as $\phi^j(0) = \varepsilon^j$ for jth iteration and $0 < \varepsilon^j e^{-kT} \leq \phi^j(t) \leq \varepsilon^j, \forall t \in [0,T], j \geq 1$ [24]. Now s_ϕ^j will play the main role in our controller design since it can be easily shown that $s_\phi^j(0) = 0$ and $s_\phi^j(t)^\top \mathbf{sat}(\frac{s^j(t)}{\phi^j(t)}) = |s_\phi^j(t)|$ for all $j \geq 1$. Let k be suitably large such that $\phi^j(t)$ can be as small as possible $\forall t \in [0,T]$. If we can show that $\lim_{j\to\infty} s_\phi^j(t) = 0, \forall t \in [0,T]$, then we have $\lim_{j\to\infty}\|s^j(t)\| \leq \phi^\infty(t)$ according to (3), which implies that the control objective will be achieved. To find the approach for the design of the proposed fuzzy-neural direct adaptive iterative learning controller, we derive the time derivative of $\frac{1}{2}s_\phi^{j\top} D(q^j) s_\phi^j$ as follows:

$$\frac{d}{dt}\left(\frac{1}{2}s_\phi^{j\top} D(q^j) s_\phi^j\right) = s_\phi^{j\top} D(q^j)\dot{s}_\phi^j + \frac{1}{2}s_\phi^{j\top}\dot{D}(q^j)s_\phi^j$$

[1] The argument t will now be omitted if it does not lead to any confusion.

$$
\begin{aligned}
&= s_\phi^{j\top} D(q^j)\left(\dot{s}_\phi^j + D^{-1}(q^j)B(q^j,\dot{q}^j)s_\phi^j\right)\\
&= s_\phi^{j\top} D(q^j)\left(\dot{s}^j - \dot{\phi}^j \mathbf{sgn}(s_\phi^j) + D^{-1}(q^j)B(q^j,\dot{q}^j)s_\phi^j\right)\\
&= s_\phi^{j\top} D(q^j)\Big\{\lambda\dot{e}^j - \ddot{q}_d - D^{-1}(q^j)\Big(B(q^j,\dot{q}^j)\dot{q}^j\\
&\quad + f(q^j,\dot{q}^j) + d^j\Big) + D^{-1}(q^j)B(q^j,\dot{q}^j)s_\phi^j\\
&\quad + D^{-1}(q^j)u^j - \dot{\phi}^j\mathbf{sgn}(s_\phi^j)\Big\} \qquad (4)
\end{aligned}
$$

where **sgn** is the notation for sign function. Suppose that the nonlinear functions $D(q^j)$, $B(q^j,\dot{q}^j)$, $f(q^j,\dot{q}^j)$ and disturbance d^j [26] are completely known, we can define the certainty equivalent controller as

$$
\begin{aligned}
u_\star^j &= B(q^j,\dot{q}^j)\dot{q}^j + f(q^j,\dot{q}^j) + d^j + B(q^j,\dot{q}^j)s_\phi^j\\
&\quad + D(q^j)\left(\ddot{q}_d - \lambda\dot{e}^j - ks^j\right) \qquad (5)
\end{aligned}
$$

with the positive constant k the same as that in (3). Let $u^j = u_\star^j$, equation (4) becomes

$$
\begin{aligned}
&\frac{d}{dt}\Big(\frac{1}{2}s_\phi^{j\top}D(q^j)s_\phi^j\Big) = s_\phi^{j\top}D(q^j)\left(-ks^j - \dot{\phi}^j\mathbf{sgn}(s_\phi^j)\right)\\
&= s_\phi^{j\top}D(q^j)\left\{-ks_\phi^j - k\phi^j\mathbf{sat}\Big(\frac{s^j}{\phi^j}\Big) - \dot{\phi}^j\mathbf{sgn}(s_\phi^j)\right\}\\
&= -s_\phi^{j\top}D(q^j)ks_\phi^j - s_\phi^{j\top}D(q^j)\big(\dot{\phi}^j + k\phi^j\big)\mathbf{sgn}(s_\phi^j)\\
&= -ks_\phi^{j\top}D(q^j)s_\phi^j \qquad (6)
\end{aligned}
$$

Since $D(q^j)$, $B(q^j,\dot{q}^j)$, $f(q^j,\dot{q}^j)$ and d^j of the robot system are in general unknown or only partially known, the result of (6) can not be achieved. However, using the results of (5) and (6), equation (4) can actually be rewritten as

$$
\frac{d}{dt}\Big(\frac{1}{2}s_\phi^{j\top}D(q^j)s_\phi^j\Big) = -ks_\phi^{j\top}D(q^j)s_\phi^j + s_\phi^{j\top}\left(u^j - u_\star^j\right) \qquad (7)
$$

• **Step 2**. The fuzzy neural network (FNN) is now applied to compensate for the unknown certainty equivalent controller $u_\star^j(t)$. For this FNN, let $O^{(4)}$, $O^{(3)}$ and W denote the network output (output of layer 4), firing strength of layer 3 (output of layer 3) and network weight between layer 3 and layer 4, respectively (for detailed, please see [27]). In this paper, the FNN will take the form of

$$
O^{(4)}(q^j(t),\dot{q}^j(t),W^j(t)) = W^j(t)^\top O^{(3)}(q^j(t),\dot{q}^j(t)) \qquad (8)
$$

where $W^j \in R^{M\times n}$ with M being the numbers of rule nodes and $O^{(3)^j} = O^{(3)}(q^j,\dot{q}^j) = [O_1^{(3)}(q^j,\dot{q}^j), \cdots, O_M^{(3)}(q^j,\dot{q}^j)]^\top$, with elements $O_\ell^{(3)}(q^j,\dot{q}^j)$, $\ell = 1,\cdots,M$ being determined by the selected membership functions. Note that $0 < O_\ell^{(3)}(q^j,\dot{q}^j) \leq 1$. It is well known that the FNN (8) can uniformly approximate real continuous nonlinear function vector $u_\star^j(t)$ on a compact set $\mathcal{A}_c \subset \mathcal{R}^{n\times 1}$ [18]. An important aspect of the above approximation property is that there exist optimal weights W^* such that the function approximation errors between the optimal $O^{(4)}(q^j,\dot{q}^j,W^*)$ and vector $u_\star^j$ can be bounded by prescribed constant θ^* on the compact set $\mathcal{A}_c$. More precisely, if we let $u_\star^j = O^{(4)}(q^j,\dot{q}^j,W^*) + \epsilon(q^j,\dot{q}^j)$, then the approximation errors will satisfy $\|\epsilon^j\| = \|\epsilon(q^j,\dot{q}^j)\| \leq \theta^*$, $\forall q^j,\dot{q}^j \in \mathcal{A}_c$. For simplicity, the FNN in the proposed DAILC only updates the consequent parameters. Such a concept is very similar in some works [22, 26]. The proposed FNN based DAILC is now designed as

$$
\begin{aligned}
u^j &= O^{(4)}(q^j,\dot{q}^j,W^j) - \mathbf{sat}\Big(\frac{s^j}{\phi^j}\Big)\theta^j\\
&= W^{j\top}O^{(3)^j} - \mathbf{sat}\Big(\frac{s^j}{\phi^j}\Big)\theta^j \qquad (9)
\end{aligned}
$$

where $W^j \in R^{M\times n}$, and $\theta^j \in R$ are the control parameters to be tuned via some suitable adaptive laws. If we substitute (9) into (7), we will have

$$
\begin{aligned}
&\frac{d}{dt}\Big(\frac{1}{2}s_\phi^{j\top}D(q^j)s_\phi^j\Big)\\
&= -ks_\phi^{j\top}D(q^j)s_\phi^j + s_\phi^{j\top}\left\{\widetilde{W}^{j\top}O^{(3)^j} - \mathbf{sat}\Big(\frac{s^j}{\phi^j}\Big)\theta^j + \epsilon^j\right\}\\
&\leq -s_\phi^{j\top}Ls_\phi^j + s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j} - |s_\phi^j|\widetilde{\theta}^j \qquad (10)
\end{aligned}
$$

where $L \equiv k\lambda_1 I$ is a symmetric positive define matrix, $\widetilde{W}^j = W^j - W^*$ and $\widetilde{\theta}^j = \theta^j - \theta^*$,

• **Step 3** : The adaptive laws combining time domain and iteration domain adaptation without deadzone or bounds of unknown parameters are proposed as follows :

$$
(1-\alpha_1)\dot{W}^j = -\alpha_1 W^j + \alpha_1 W^{j-1} - \beta_1 O^{(3)^j}s_\phi^{j\top} \qquad (11)
$$

$$
(1-\alpha_2)\dot{\theta}^j = -\alpha_2\theta^j + \alpha_2\theta^{j-1} + \beta_2|s_\phi^j| \qquad (12)
$$

with $W^j(0) = W^{j-1}(T)$, $\theta^j(0) = \theta^{j-1}(T)$ for $j \geq 1$, and $0 < \alpha_1,\alpha_2 < 1$, $\beta_1,\beta_2 > 0$. In this adaptive law, α_1, α_1 and β_1, β_2 are defined as the weighting gains and adaptation gains, respectively. For the first iteration, we set $W^0(t) = W^0$ and $\theta^0(t) = \theta^0$ to be any constant value. It is noted that (11) and (12) will be reduced to pure time-domain adaptation laws if α_1=α_2=0, or pure iteration-domain adaptation laws if α_1=α_2=1. Also it is obviously that $-\alpha_1 W^j + \alpha_1 W^{j-1} = -\alpha_1\widetilde{W}^j + \alpha_1\widetilde{W}^{j-1}$ and $-\alpha_2\theta^j + \alpha_2\theta^{j-1} = -\alpha_2\widetilde{\theta}^j + \alpha_2\widetilde{\theta}^{j-1}$.

3. ANALYSIS OF STABILITY AND CONVERGENCE

Lemma 1 : Consider the uncertain robot system (1) which satisfies assumptions (A1)–(A4). The proposed FNN-DAILC (9) and adaptation laws (11) and (12) will ensure that all the internal signals at first iteration are bounded, i.e., e^1, s_ϕ^1, s^1, W^1, θ^1, u^1, $\dot{s}^1$, $\dot{W}^1$, $\dot{\theta}^1 \in L_{\infty e}[0,T]$.

Proof : Let us choose a Lyapunov function as

$$
\begin{aligned}
V_a^j &= \frac{1}{2}s_\phi^{j\top}D(q^j)s_\phi^j + \frac{(1-\alpha_1)}{2\beta_1}tr\left\{\widetilde{W}^{j\top}\widetilde{W}^j\right\}\\
&\quad + \frac{(1-\alpha_2)}{2\beta_2}(\widetilde{\theta}^j)^2.
\end{aligned}
$$

Using the fact of (10) and parameter adaptation laws (11) and (12), and after some simple manipulations, we can compute its derivative with respective to time t along (2), (11) and (12) as follows :

$$
\begin{aligned}
&\dot{V}_a^j\\
&\leq -s_\phi^{j\top}Ls_\phi^j + s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j} - |s_\phi^j|\widetilde{\theta}^j
\end{aligned}
$$

$$
\begin{aligned}
&+\frac{1}{\beta_1}tr\Big\{\widetilde{W}^{j\top}\Big[(1-\alpha_1)\dot{\widetilde{W}}^j\Big]\Big\}+\frac{1}{\beta_2}\widetilde{\theta}^j\Big[(1-\alpha_2)\dot{\widetilde{\theta}}^j\Big]\\
&=-s_\phi^{j\top}Ls_\phi^j+s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}-|s_\phi^j|\widetilde{\theta}^j\\
&\quad+\frac{1}{\beta_1}tr\Big\{\widetilde{W}^{j\top}\Big[-\alpha_1W^j+\alpha_1W^{j-1}-\beta_1O^{(3)^j}s_\phi^{j\top}\Big]\Big\}\\
&\quad+\frac{1}{\beta_2}\widetilde{\theta}^j\Big[-\alpha_2\theta^j+\alpha_2\theta^{j-1}+\beta_2|s_\phi^j|\Big]\\
&=-s_\phi^{j\top}Ls_\phi^j-\frac{\alpha_1}{\beta_1}tr\left\{\widetilde{W}^{j\top}\widetilde{W}^j\right\}+\frac{\alpha_1}{\beta_1}tr\left\{\widetilde{W}^{j\top}\widetilde{W}^{j-1}\right\}\\
&\quad-\frac{\alpha_2}{\beta_2}(\widetilde{\theta}^j)^2+\frac{\alpha_2}{\beta_2}\widetilde{\theta}^j\widetilde{\theta}^{j-1}
\end{aligned}\tag{13}
$$

where we use the property of $tr\{\widetilde{W}^{j\top}O^{(3)^j}(t)s_\phi^{j\top}\}=s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}$. Note that $\widetilde{W}^0=W^0-W^*\equiv\overline{W}^0$ and $\widetilde{\theta}^0=\theta^0-\theta^*\equiv\overline{\theta}^0$ are bounded for all $t\in[0,T]$. So if we let $j=1$, we can rewrite (13) as follows,

$$
\begin{aligned}
&\dot{V}_a^1\\
&\le-s_\phi^{1\top}Ls_\phi^1-\frac{\alpha_1}{\beta_1}tr\left\{\widetilde{W}^{1\top}\widetilde{W}^1\right\}+\frac{\alpha_1}{\beta_1}tr\left\{\widetilde{W}^{1\top}\overline{W}^0\right\}\\
&\quad-\frac{\alpha_2}{\beta_2}(\widetilde{\theta}^1)^2+\frac{\alpha_2}{\beta_2}\widetilde{\theta}^1\overline{\theta}^0\\
&=-s_\phi^{1\top}Ls_\phi^1-\frac{\alpha_1}{2\beta_1}tr\Big\{\widetilde{W}^{1\top}\widetilde{W}^1\Big\}-\frac{\alpha_2}{2\beta_2}(\widetilde{\theta}^1)^2\\
&\quad-\frac{\alpha_1}{2\beta_1}tr\Big\{\Big(\widetilde{W}^1-\overline{W}^0\Big)^\top\Big(\widetilde{W}^1-\overline{W}^0\Big)\Big\}\\
&\quad-\frac{\alpha_2}{2\beta_2}(\widetilde{\theta}^1-\overline{\theta}^0)^2+\frac{\alpha_1}{2\beta_1}tr\left\{\overline{W}^{0\top}\overline{W}^0\right\}+\frac{\alpha_2}{2\beta_2}(\overline{\theta}^0)^2\\
&\le-\lambda V_a^1(t)+\overline{\lambda}^0
\end{aligned}\tag{14}
$$

where $\lambda=\min\{2k\lambda_1,\frac{\alpha_1}{1-\alpha_1},\frac{\alpha_2}{1-\alpha_2}\}$, $\overline{\lambda}^0=\frac{\alpha_1}{2\beta_1}tr\{\overline{W}^{0\top}\overline{W}^0\}+\frac{\alpha_2}{2\beta_2}(\overline{\theta}^0)^2$. Note that the initial value $V_a^1(0)$ is bounded since $s_\phi^1(0)=0$, $\widetilde{W}^1(0)=W^1(0)-W^*=W^0(T)-W^*=\overline{W}^0$, and $\widetilde{\theta}^1(0)=\theta^1(0)-\theta=\theta^0(T)-\theta=\overline{\theta}^0$. Together with the result of (14), it readily implies $V_a^1, s_\phi^1, \widetilde{W}^1, \widetilde{\theta}^1\in L_{\infty e}[0,T]$ and hence, s^1 (by (3)), u^1 (by (9)), $\dot{s}^1$ (by (2)), $\dot{W}^1$ (by 11)), $\dot{\theta}^1$ (by 12)) $\in L_{\infty e}[0,T]$. Q.E.D.

Lemma 2 : Consider the system set-up in lemma 1, the proposed FNN-DAILC ensures that

(L1) $\lim_{j\to\infty}tr\{\widetilde{W}^{j\top}(T)\widetilde{W}^j(T)\}=tr\{\widetilde{W}_T^\top\widetilde{W}_T\}$, and $\lim_{j\to\infty}(\widetilde{\theta}^j(T))^2=(\widetilde{\theta}_T)^2$ for some constant matrix $\widetilde{W}_T^\top\widetilde{W}_T$ and constant $(\widetilde{\theta}_T)^2$.

(L2) $tr\{\widetilde{W}^{j\top}(T)\widetilde{W}^j(T)\}$, $(\widetilde{\theta}^j(T))^2$, $(s_\phi^j(T))^2$ are bounded for all $j\ge1$.

(L3) $\lim_{j\to\infty}\int_0^T s_\phi^{j\top}(t)Ls_\phi^j(t)dt=0$, and $\lim_{j\to\infty}s_\phi^{j\top}(T)D(q^j(T))s_\phi^j(T)=0$.

Proof : Define a positive function $V^j(T)$ as

$$
\begin{aligned}
V^j(T)=&\int_0^T\left[\frac{\alpha_1}{2\beta_1}tr\Big\{\widetilde{W}^{j\top}\widetilde{W}^j\Big\}+\frac{\alpha_2}{2\beta_2}(\widetilde{\theta}^j)^2\right]dt\\
&+\frac{1-\alpha_1}{2\beta_1}tr\Big\{\widetilde{W}^{j\top}(T)\widetilde{W}^j(T)\Big\}+\frac{1-\alpha_2}{2\beta_2}(\widetilde{\theta}^j(T))^2,
\end{aligned}
$$

the difference between $V^j(T)$ and $V^{j-1}(T)$ can be derived by using integration by parts as follows :

$$
\begin{aligned}
&V^j(T)-V^{j-1}(T)\\
&=\int_0^T\Big[\frac{\alpha_1}{2\beta_1}\Big(tr\Big\{\widetilde{W}^{j\top}\widetilde{W}^j\Big\}-tr\Big\{\widetilde{W}^{j-1\top}\widetilde{W}^{j-1}\Big\}\Big)\\
&\quad+\frac{\alpha_2}{2\beta_2}\Big((\widetilde{\theta}^j)^2-(\widetilde{\theta}^{j-1})^2\Big)\Big]dt\\
&\quad+\frac{(1-\alpha_1)}{\beta_1}\int_0^T tr\Big\{\widetilde{W}^{j\top}\dot{\widetilde{W}}^j\Big\}dt\\
&\quad+\frac{(1-\alpha_1)}{2\beta_1}tr\left\{\widetilde{W}^{j\top}(0)\widetilde{W}^j(0)\right\}\\
&\quad-\frac{(1-\alpha_1)}{2\beta_1}tr\left\{\widetilde{W}^{j-1\top}(T)\widetilde{W}^{j-1}(T)\right\}\\
&\quad+\frac{(1-\alpha_2)}{\beta_2}\int_0^T\widetilde{\theta}^j(t)\dot{\widetilde{\theta}}^j(t)dt\\
&\quad+\frac{(1-\alpha_2)}{2\beta_2}(\widetilde{\theta}^j(0))^2-\frac{(1-\alpha_2)}{2\beta_2}(\widetilde{\theta}^{j-1}(T))^2\\
&=\int_0^T\Big[-\frac{\alpha_1}{2\beta_1}tr\Big\{\Big(\widetilde{W}^j-\widetilde{W}^{j-1}\Big)^\top\Big(\widetilde{W}^j-\widetilde{W}^{j-1}\Big)\Big\}\\
&\qquad-\frac{\alpha_2}{2\beta_2}\Big(\widetilde{\theta}^j-\widetilde{\theta}^{j-1}\Big)^2\Big]dt\\
&\quad+\int_0^T\Big[-s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}+|s_\phi^j|\widetilde{\theta}^j\Big]dt\\
&\le\int_0^T\Big[-s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}+|s_\phi^j|\widetilde{\theta}^j\Big]dt
\end{aligned}\tag{15}
$$

If we define $V_b^j=\frac{1}{2}s_\phi^{j\top}D(q^j)s_\phi^j$, we can easily derive the following inequality by similar argument in lemma 1 that

$$
\dot{V}_b^j\le-s_\phi^{j\top}Ls_\phi^j+s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}-|s_\phi^j|\widetilde{\theta}^j\tag{16}
$$

Integrating both side of (16) from 0 to T gives

$$
\int_0^T\Big[-s_\phi^{j\top}\widetilde{W}^{j\top}O^{(3)^j}+|s_\phi^j|\widetilde{\theta}^j\Big]dt\le\int_0^T-s_\phi^{j\top}Ls_\phi^jdt-V_b^j(T)\tag{17}
$$

where we use the fact of $V_b^j(0)=\frac{1}{2}s_\phi^{j\top}(0)D(q^j(0))s_\phi^j(0)=0$. Substituting (17) into (15), it yields

$$
\begin{aligned}
&V^j(T)-V^{j-1}(T)\\
&\le\int_0^T-s_\phi^{j\top}Ls_\phi^jdt-\frac{1}{2}s_\phi^{j\top}(T)D(q^j(T))s_\phi^j(T)
\end{aligned}\tag{18}
$$

Since $V^1(T)$ is bounded by lemma 1, and $V^j(T)$ is positive and monotonically decreasing, $V^j(T)$ is bounded for all $j\ge1$ and will converge as j approaches infinity to some limit value $V(T)$ (independent of j). This shows (L1) of lemma 2. On the other hand, (18) implies

$$\int_0^T s_\phi^{j\top} L s_\phi^j dt \leq V^{j-1}(T) - V^j(T) \leq V^1(T) \tag{19}$$

$$\frac{1}{2} s_\phi^{j\top}(T) D(q^j(t)) s_\phi^j(T) \leq V^{j-1}(T) - V^j(T) \leq V^1(T) \tag{20}$$

for all $j \geq 1$. The boundedness of $\frac{1}{2} s_\phi^{j\top}(T) D(q^j(T)) s_\phi^j(T)$ for all iterations is then established from (20). Therefore, this further implies that $(s_\phi^j(T))^2$ is bounded for all iterations, and hence (L2) of lemma 2 follows. Finally, as $\lim_{j\to\infty} V^{j-1}(T) - V^j(T) = 0$, (L3) of lemma 2 is achieved from (19) and (20). Q.E.D.

Using the boundedness of $\widetilde{W}^j(T)$, and $\widetilde{\theta}^j(T)$ (or equivalently the boundedness of $\widetilde{W}^j(0)$ and $\widetilde{\theta}^j(0)$) for all $j \geq 1$ as shown in (L2) of lemma 2, the convergence of s_ϕ^j and boundedness of all internal signals for all $j \geq 1$ are now established in the following theorem.

Theorem 1 : Consider the system set-up in lemma 1. The proposed FNN-DAILC guarantees the tracking performance and system stability as follows :

(T1) $s_\phi^j, s^j, W^j, \theta^j, u^j, \dot{s}^j, \dot{W}^j, \dot{\theta}^j \in L_{\infty e}[0,T]$, for all $j \geq 1$.

(T2) $\lim_{j\to\infty} s_\phi^{j\top} L s_\phi^j = s_\phi^{\infty\top} L s_\phi^\infty = 0$, for all $t \in [0,T]$.

(T3) $\lim_{j\to\infty} |s^j| = |s^\infty| \leq \phi^\infty = e^{-kt}\varepsilon^\infty$, for all $t \in [0,T]$.

Proof :

(T1) Since $s_\phi^1, \widetilde{W}^1, \widetilde{\theta}^1 \in L_{\infty e}[0,T]$ as shown in lemma 1, if we assume s_ϕ^{j-1}, $\widetilde{W}^{j-1}$, $\widetilde{\theta}^{j-1} \in L_{\infty e}[0,T]$, then derivative of Lyapunov function V_a^j in (13) can be rewritten as

$$\begin{aligned} \dot{V}_a^j \leq & -s_\phi^{j\top} L s_\phi^j \\ & - \frac{\alpha_1}{\beta_1} tr\left\{\widetilde{W}^{j\top}\widetilde{W}^j\right\} + \frac{\alpha_1}{\beta_1} tr\left\{\widetilde{W}^{j\top}\overline{W}^{j-1}\right\} \\ & - \frac{\alpha_2}{\beta_2}(\widetilde{\theta}^j)^2 + \frac{\alpha_2}{\beta_2}\widetilde{\theta}^j\overline{\theta}^{j-1} \\ \leq & -\lambda V_a^j + \overline{\lambda}^{j-1} \end{aligned} \tag{21}$$

where $\overline{\lambda}^{j-1} = \frac{\alpha_1}{2\beta_1} tr\{\overline{W}^{j-1\top}\overline{W}^{j-1}\} + \frac{\alpha_2}{2\beta_2}(\overline{\theta}^{j-1})^2$ and $\overline{W}^{j-1}$ and $\overline{\theta}^{j-1}$ are the upper bounds on $|\widetilde{W}^{j-1}|$ and $|\widetilde{\theta}^{j-1}|$, respectively. Since the initial condition of the Lyapunov function V_a^j is bounded for all $j \geq 1$ due to (L2) of lemma 2, we conclude from (21) that $s_\phi^j, \widetilde{W}^j$, $\widetilde{\theta}^j$ and $s^j, u^j, \dot{s}^j, \dot{W}^j, \dot{\theta}^j \in L_{\infty e}[0,T]$. Hence, (T1) of Theorem 1 is achieved by using mathematical induction.

(T2) By using V_b^j in lemma 2 and (T1) in this theorem, we have $\dot{V}_b^j$ and $V_b^j \in L_{\infty e}[0,T]$. On other hand, $\lim_{j\to\infty}\int_0^T s_\phi^{j\top} L s_\phi^j dt$=0, or $\lim_{j\to\infty}\int_0^T V_b^j dt$ =0 due to (L3) of lemma 2. We can finally conclude that $\lim_{j\to\infty} s_\phi^{j\top} L s_\phi^j$=$\lim_{j\to\infty} V_b^j$=0, for all $t \in [0,T]$ by using similar argument of Barbalat's lemma [28].

(T3) Since $\lim_{j\to\infty} s_\phi^j = s_\phi^\infty = 0$ for all $t \in [0,T]$, the boundedness of s^j at each iteration over $[0,T]$ can be concluded from equation (3) because ϕ^∞ is always bounded and s^∞ converges to $s^\infty = \phi^\infty \mathbf{sat}\left(\frac{s^\infty}{\phi^\infty}\right)$, so that $|s^\infty| \leq \phi^\infty = e^{-kt}\varepsilon^\infty$ for all $t \in [0,T]$. Q.E.D.

4. A SIMULATION EXAMPLE

The dynamic equation of the two-link planar robotic system [5] is given as follows,

$$\begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1^j \\ \ddot{q}_2^j \end{bmatrix} + \begin{bmatrix} -h\dot{q}_2^j & -h(\dot{q}_1^j + \dot{q}_2^j) \\ h\dot{q}_1^j & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1^j \\ \dot{q}_2^j \end{bmatrix} + \begin{bmatrix} d_1^j \\ d_2^j \end{bmatrix} = \begin{bmatrix} u_1^j \\ u_2^j \end{bmatrix} \tag{22}$$

where $D_{11} = m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos(q_2^j)) + I_1 + I_2$, $D_{12} = D_{21} = m_2 l_1 l_{c2}\cos(q_2^j) + m_2 l_{c2}^2 + I_2$, $H_{22} = m_2 l_{c2}^2 + I_2$, $h = m_2 l_1 l_{c2}\sin(q_2^j)$. Here m_i, I_i, l_i and l_{c_i} represent mass, inertia, length of link i, and the distance from the previous joint to the center of mass of link i, respectively. In this simulation, the physical parameters are specified as m_1 = 10kg, m_2 = 5kg, l_1 = 1m, l_2 = 0.5m, l_{c1} = 0.5m, l_{c2} = 0.25m, I_1 = 0.83kg-m^2 and I_2 = 0.3kg-m^2. The control objective is to let $q^j = [q_1^j, q_2^j]^\top$ track the desired trajectory $q_d = [q_{d1}, q_{d2}]^\top = [\sin(3t), \cos(3t)]^\top$ as close as possible over a finite time interval $[0, 15]$. We assume that the disturbances take the form of $d^j = [d_1^j, d_2^j]^\top = [m_1^j \sin(w_1^j t), m_2^j \sin(w_2^j t)]$ and the values of m_1^j, m_2^j, w_1^j and w_2^j are varying. The proposed FNN-DAILC in (9) is applied with the design parameters $k = 1$, and the diagonal positive define matrix $\lambda = diag[2,2]$. In this simulation, we set the weighing gains $\alpha_1 = \alpha_2 = 0.5$, i.e., the weighting between time-domain adaptation and iteration-domain adaptation is equal. Then we investigate the effect of the learning gains by choosing $\beta_1 = \beta_2 = 500$ and $\beta_1 = \beta_2 = 5000$, respectively. In order to show the robustness to the varying initial resetting errors, we assume that the initial joint position and velocity take the arbitrary values for the first 5 iterations. To study the effects of the proposed FNN-DAILC, we show some learning performances in Figure 1 and Figure 2, respectively.

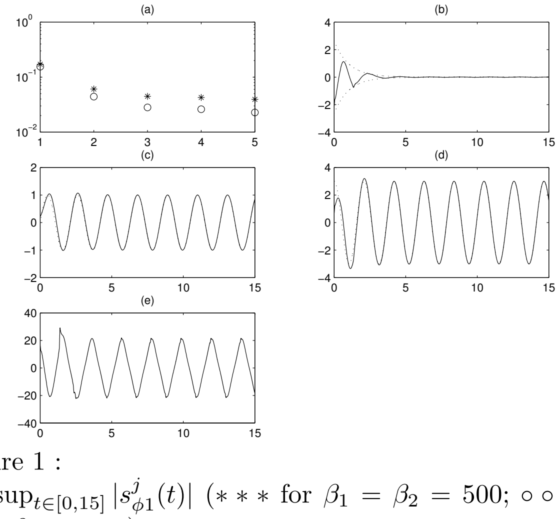

Figure 1 :
(a) $\sup_{t\in[0,15]} |s_{\phi 1}^j(t)|$ ($***$ for $\beta_1 = \beta_2 = 500$; $\circ\circ\circ$ for $\beta_1 = \beta_2 = 5000$) versus iteration j ;
(b) $s_1^5(t)$ (solid line) and $\pm\phi^5(t)$ (dotted lines) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;

(c) $q_1^5(t)$ (solid line) and q_{d1} (dotted line) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;
(d) $\dot{q}_1^5(t)$ (solid line) and $\dot{q}_{d1}$ (dotted line) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;
(e) $u_1^5(t)$ versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;

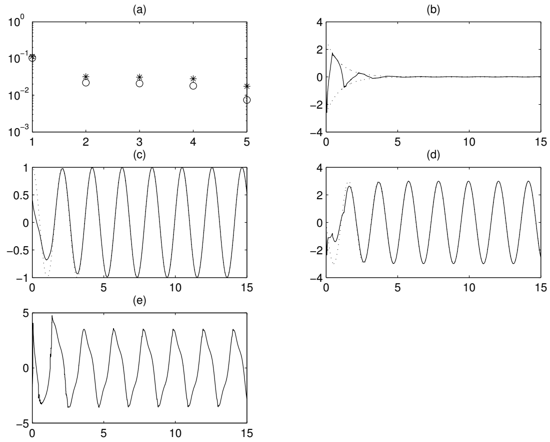

Figure 2 :
(a) $\sup_{t\in[0,15]} |s_{\phi 2}^j(t)|$ ($***$ for $\beta_1 = \beta_2 = 500$; $\circ\circ\circ$ for $\beta_1 = \beta_2 = 5000$) versus iteration j ;
(b) $s_2^5(t)$ (solid line) and $\pm\phi^5(t)$ (dotted lines) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;
(c) $q_2^5(t)$ (solid line) and q_{d2} (dotted line) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;
(d) $\dot{q}_2^5(t)$ (solid line) and $\dot{q}_{d2}$ (dotted line) versus time t, $\beta_1 = \beta_2 = 5000, k = 1$;
(e) $u_2^5(t)$ versus time t, $\beta_1 = \beta_2 = 5000, k = 1$.

5. CONCLUSION

For a repetitive control task of robotic system, a fuzzy neural network based adaptive iterative learning controller is proposed in this work. Since a direct scheme is applied to design the learning control structure, the fuzzy neural network is used to play a role of compensation for unknown desired certainty equivalent controller. The optimal weights of the fuzzy neural network are tuned by a new adaptive law which combines the adaptation along time domain and iteration domain. Rigorous analysis is presented to guarantee the signal stability in time domain and error convergence in iteration domain. Simulation results demonstrate the effectiveness of the proposed FNN-DAILC for robotic systems with repeatable control tasks.

REFERENCES

[1] M. Takegaki and S. Arimoto, "A new feedback method for dynamic control of manipulators," *Journal of Dynamic Systems, Measurement and Control*, vol. 103, pp. 119-125, 1981.

[2] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, New York: Wiley, 1989.

[3] S. Arimoto, *Control Theory of Non-linear Mechanical Systems*, Oxford, UK: Oxford Science Publications, 1996.

[4] P. Tomei, "Adaptive PD controller for robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 565-570, 1991.

[5] J. J. Slotine and W. Li, *Applied Nonlinear Control*, Englewood Cliffs, NJ: Prentice-Hall, 1991.

[6] P. Bondi, G. Casalino, and L. Gambardella, "On the iterative learning control theory for robotic manipulators," *IEEE Journal of Robotics and Automation*, vol. 4, no. 1, pp. 14-22, 1988.

[7] S. Kawamura, F. Miyazaki, and S. Arimoto, "Realization of robot motion based on a learning method," *IEEE Transactions on Systems, Man and Cybernetics, Part-B*, vol. 18, no. 1, pp. 126-134, 1988.

[8] R. Horowitz, "Learning control of robot manipulators," *ASME Journal of Dynamic Systems Measurements and Control*, vol. 115, pp. 402-411, 1993.

[9] D. Wang, Y. C. Son and C. C. Cheah, "Robust motion and force control of constrained manipulators by learning," *Automatica*, vol. 31, no. 2, pp. 257-262, 1995.

[10] M. Norrlöf, "An adaptive iterative learning control algorithm with experiments on an industrial robot," *IEEE Transtractions on Robotics and Automation*, vol. 18, no. 2, pp. 245-251, 2002.

[11] B.H. Park, T.Y. Kuc and J.S. Lee, "Adaptive learning of uncertain robotic systems," *Int. J. Contr.*, vol. 65, no. 5, pp. 725-744, 1996.

[12] T.Y. Kuc and W.G. Han, "An adaptive PID learning control of robot manipulators," *Automatica*, vol. 36, pp. 717-725, 2000.

[13] J.X. Xu, and B. Wiswanathan, "Adaptive robust iterative learning control with dead zone scheme," *Automatica*, vol. 36, pp. 91-99, 2000.

[14] J.Y. Choi and J.S. Lee, "Adaptive iterative learning control of uncertain robotic systems," *IEE Proc. D, Control Theory Application*, vol. 147, no. 2, pp. 217-223, 2000.

[15] A. Tayebi, "Adaptive iterative learning control for robot manipulators," *Automatica*, vol. 40, pp. 1195-1203, 2004.

[16] M. Sanner and J.J.E. Slotine, "Gaussin networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, pp. 837–863, 1992.

[17] G.A. Rovithakis and M.A. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 24, no. 3, pp. 400-412, 1994.

[18] L.X. Wang, *Adaptive Fuzzy Systems and Control*, Englewood Cliffs, NJ : Prentice Hall, pp. 140–154, 1994.

[19] F.C. Chen and H.K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," *IEEE Trans. Automat. Contr.*, vol. 40, no. 5, pp. 791–801, 1995.

[20] J.T. Sponner and K.M. Passino, "Stable adaptive control using fuzzy systems and neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 3, pp. 339–359, 1996.

[21] H. Han, C.Y. Su and Y. Stepanenko, "Adaptive control of a class of nonlinear systems with nonlinearly parameterized fuzzy approximators," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 2, pp. 315–323, 2001.

[22] C.H. Wang, T.C. Lin, T.T. Lee, and H.L. Liu, "Adaptive hybrid intelligent control for uncertain nonlinear dynamical systems," *IEEE Trans. on Systems, Man, and Cybernetics, Part-B*, vol. 32, no. 5, pp. 583–597, 2002.

[23] W.G. Seo, B.H. Park and J.S. Lee, "Adaptive fuzzy learning control for a class of nonlinear dynamic systems," *International Journal of Intelligent Systems*, vol. 15, pp. 1157–1175, 2000.

[24] C.J. Chien, C.T. Hsu and C.Y. Yao, "Fuzzy system based adaptive iterative learning control for nonlinear plants with initial state errors," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 5, pp. 724-732, 2004.

[25] C.J. Chien and L.C. Fu, "An iterative learning control of nonlinear systems using neural network design," *Asian Jornal of Control*, vol. 4, no. 1, pp. 21-29, 2002.

[26] F. J. Lin, R. J. Wai and C. M. Hong, "Hybrid control using recurrent fuzzy neural network for tracking periodic inputs," *IEEE Trans. Neural Networks*, vol. 12, no. 1, pp. 68–90, 2001.

[27] Y.C. Chen and C.C. Teng, "A model reference control structure using a fuzzy neural network," *Fuzzy Sets and Systems,* vol. 73, pp. 291-312, 1995.

[28] P.A. Ioannou and J. Sun, *Robust Adaptive Control*, Englewood Cliffs, NJ, Prentice-Hall, 1996.