ADAPTIVE ITERATIVE LEARNING CONTROL OF ROBOTIC SYSTEMS USING BACKSTEPPING DESIGN

Ying-Chung Wang, Chiang-Ju Chien and Chi-Nan Chuang
Department of Electronic Engineering, Huafan University, Shihding, New Taipei City, Taiwan
E-mail: ycwang@cc.hfu.edu.tw; cjc@cc.hfu.edu.tw; cnchuang@cc.hfu.edu.tw

ICETI 2012-I1058_SCI
No. 13-CSME-45, E.I.C. Accession 3503

ABSTRACT

In this paper, a backstepping adaptive iterative learning control (AILC) is proposed for robotic systems with repetitive tasks. The AILC is designed to approximate unknown certainty equivalent controller. Finally, we apply a Lyapunov like analysis to show that all adjustable parameters and the internal signals remain bounded for all iterations.

Keywords: robotic systems; backstepping; adaptive iterative learning control; fuzzy neural network.

COMMANDE AUTODIDACTE ADAPTIVE ET ITÉRATIVE POUR UN SYSTÈME ROBOTIQUE UTILISANT LE CONCEPT DE RÉTROGRADATION

RÉSUMÉ

Dans cet article, nous proposons une commande autodidacte adaptive et itérative par rétrogradation pour des systèmes robotiques effectuant des tâches répétitives. La commande est conçue pour une approche approximative de l'équivalent d'un contrôleur des incertitudes. Finalement, nous appliquons une analyse de type Lyapunov pour montrer que tous les paramètres et les signaux demeurent liés pour toutes les itérations.

Mots-clés : systèmes robotiques ; rétrogradation ; commande autodidacte adaptive et itérative.

1. INTRODUCTION

Taking advantage of the fact that robot manipulators are generally used in repetitive tasks, several iterative learning control (ILC) schemes have been proposed for robot manipulators in the past two decades. The ILC is basically an effective non-model based learning approach in dealing with repetitive control tasks [1, 2]. In order to relax the restrict Lipschitz condition on the plant's nonlinearity, the Lyapunov-like approach without using contraction mapping theory is applied in adaptive iterative learning control (AILC) [3, 4]. In [3], the AILC strategy was proposed for repetitive control of robotic systems. The main structure of this AILC scheme including a switching-type robust AILC and an adaptive law combines time-domain and iteration-domain adaptation is proposed so that the projection mechanism or bounds on the unknown control parameters is not required in the adaptive law. Different from the classical PD-type ILC schemes, the PD-type AILC scheme proposed in [4] which is a combination of a feedback PD-type ILC and a feedforward learning component. Therefore, it can be used to control of robotic systems to a wide class of task space. On the other hand, the backstepping design approach provides a systematic procedure and recursive design methodology for nonlinear systems [5]. However, the drawback of backstepping design approach is that the nonlinear functions of the systems are necessarily known. Recently, a robust backstepping control of a class of nonlinear pure-feedback systems using fuzzy system was proposed in [6]. The fuzzy systems were used to learn the behavior of the unknown plant dynamics so that the approximation errors can be efficiently approximated by the robust compensators. In [7], a neural network based stable decentralized adaptive backstepping tracking controller is designed for a class of large-scale systems with mismatched interconnections. The neural networks in the proposed scheme can be used to approximate the unknown interconnections dependent on all reference signals. On the other hand, the backstepping approach has been used to design the ILC in [8], the input gain function of the systems is assumed to unity and some restrictive assumptions are necessarily in technical analysis. In [9], an AILC approach was proposed for a class of hybrid parametric nonlinear systems using backstepping method. The proposed approach consists of a differential-deference type update law and a learning control law to deal with the non-uniform trajectory tracking problem. In order to relax the restrictions on the tracking trajectory in the traditional ILC, a sufficient condition to ensure the tracking error to converge to zero in the finite interval is given by constructing a composite energy function. In [10], the Lyapunov stability analysis and backstepping control technology are used to design a robust AILC for a class of second order nonlinear systems with identical initial condition and alignment condition. The unknown parameters will be estimated in time-domain and the disturbance is inhibited by the proposed robust control.

In this paper, combining the advantages of AILC deign concept, backstepping design technique and fuzzy neural network function approximation property, a backstepping-like procedure is introduced to design a backstepping AILC for uncertain robotic systems. The FNN is used to be a fuzzy neural learning component for compensation of the unknown certainty equivalent controller. Using this direct scheme, only one FNN is required to design the ILC. In addition, a robust iterative learning control component is designed to compensate the uncertainties. In order to guarantee the finiteness of control parameters, control input, the convergence of the tracking error and the boundedness of all internal signals for all the time interval during each iteration, we apply a Lyapunov-like analysis to guarantee the stability and convergence of the closed-loop learning system. We also show 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. Finally, a simulation example is used to verify stability, convergence and the control performance of the proposed backstepping AILC systems.

This paper is organized as follows. In Section 2, the error model and controller design are derived. Analysis of closed-loop stability and learning performance is studied extensively in Section 3. Section 4 gives an example of computer simulation to demonstrate the effectiveness of the proposed learning controller. Finally a conclusion is made in Section 5.

2. PROBLEM FORMULATION

2.1. Robotic Systems and Control Objective

In this paper, we consider an uncertain robotic system with n rigid bodies which can perform a given task repeatedly over a finite time interval as follows:

$$D(q^j(t))\ddot{q}^j(t)+B(q^j(t),\dot{q}^j(t))\dot{q}^j(t)+F(q^j(t),\dot{q}^j(t))=u^j(t), \tag{1}$$

where $j \in \{0,1,\ldots,\infty\}$ denotes the index of iteration number and $t \in [0,T]$ denotes the time index. The signals $q^j(t),\dot{q}^j(t),\ddot{q}^j(t) \in R^{n\times 1}$ are respectively the generalized joint position, joint velocity and joint acceleration vectors. $D(q^j(t)) \in R^{n\times n}$ is the unknown inertia matrix, $B(q^j(t),\dot{q}^j(t)) \in R^{n\times n}$ is the unknown centripetal plus Coriolis force vector, $F(q^j(t),\dot{q}^j(t)) \in R^{n\times 1}$ is the unknown gravitational plus frictional forces and $u^j(t) \in R^n$ is the joint torque vector. Given the specified desired joint position, velocity, acceleration trajectories $q_d(t)$, $\dot{q}_d(t)$, $\ddot{q}_d(t)$, $\forall t \in [0,T]$, the control objective is to design a backstepping adaptive iterative learning controller $u^j(t)$ such that $\|q^j(t)-q_d(t)\|$ and $\|\dot{q}^j(t)-\dot{q}_d(t)\|$ will converge to zero $\forall t \in [0,T]$ when iteration number j is large enough. 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.

(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

$$0 < m_1 I \leq D(q^j(t)) \leq m_2 I, \tag{2}$$

where $m_1, m_2 > 0$ 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,

$$z^{\mathrm{T}}(\dot{D}(q^j(t)) - 2B(q^j(t),\dot{q}^j(t)))z = 0 \tag{3}$$

for all $z \in R^{n\times 1}$ and $z \neq 0$.

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

(A4) Let the initial joint position, velocity, acceleration at each iteration are the same, i.e., $q^j(0) = q_d(0)$ and $\dot{q}^j(0) = \dot{q}_d(0)$.

Now, in order to illustrate the idea of the learning controller, we use the following four steps to explain the design approach in the next subsection.

2.2. Derivations of Error Model and Controller

Step 1

At first, let the output error $e_1^j(t)$ as

$$e_1^j(t) = q^j(t) - q_d(t). \tag{4}$$

Then we derive the time derivative of $e_1^j(t)$ as

$$\dot{e}_1^j(t) = \dot{q}^j(t) - \dot{q}_d(t). \tag{5}$$

Let us define the following stabilizing function as

$$\alpha^j(t) = -\lambda_1 e_1^j(t) + \dot{q}_d(t), \tag{6}$$

where $\lambda_1 > 0$ is a positive constant. Define the error function as

$$e_2^j(t) = \dot{q}^j(t) - \alpha^j(t). \tag{7}$$

Then the first Lyapunov-like positive function is chosen as

$$V_1^j(t) = \frac{1}{2} e_1^j(t)^{\mathrm{T}} e_1^j(t) \tag{8}$$

and compute its derivative with respective to time t as

$$\begin{aligned} \dot{V}_1^j(t) &= e_1^j(t)^{\mathrm{T}} \dot{e}_1^j(t) \\ &= e_1^j(t)^{\mathrm{T}} e_2^j(t) - \lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t). \end{aligned} \tag{9}$$

Step 2
Let us compute the derivative of $e_2^j(t)$ with respective to time t as follows:

$$\begin{aligned} \dot{e}_2^j(t) &= \ddot{q}^j(t) - \dot{\alpha}^j(t) \\ &= -D^{-1}(q^j(t))[B(q^j(t)), \dot{q}^j(t))\dot{q}^j(t) + F(q^j(t), \dot{q}^j(t))] \\ &\quad + D^{-1}(q^j(t))u^j(t) - \ddot{q}_d(t) + \lambda_1 \dot{e}_1^j(t). \end{aligned} \tag{10}$$

In order to design the backstepping control system, we define the second Lyapunov-like positive function as

$$V_2^j(t) = V_1^j(t) + \frac{1}{2} e_2^j(t)^{\mathrm{T}} D(q^j(t)) e_2^j(t). \tag{11}$$

Using Eq. (9), we compute the derivative of $V_2^j(t)$ with respective to time t as

$$\begin{aligned} \dot{V}_2^j(t) &= \dot{V}_1^j(t) + \frac{d}{dt}\left(\frac{1}{2} e_2^j(t)^{\mathrm{T}} D(q^j(t)) e_2^j(t)\right) \\ &= -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) \\ &\quad + e_2^j(t)^{\mathrm{T}} D(q^j(t)) \Big\{ \lambda_1 \dot{e}_1^j(t) - \ddot{q}_d(t) - D^{-1}(q^j(t))[B(q^j(t)), \dot{q}^j(t))\dot{q}^j(t) \\ &\quad + F(q^j(t), \dot{q}^j(t)] + D^{-1}(q^j(t)) B(q^j(t), \dot{q}^j(t)) e_2^j(t) + D^{-1}(q^j(t)) e_1^j(t) \\ &\quad + D^{-1}(q^j(t)) u^j(t) \Big\}. \end{aligned} \tag{12}$$

If $D(q^j(t))$, $B(q^j(t), \dot{q}^j(t))$ and $F(q^j(t), \dot{q}^j(t))$ are known, we can define the following certainty equivalent controller

$$\begin{aligned} u_*^j(t) &= B(q^j(t), \dot{q}^j(t))\dot{q}^j(t) + F(q^j(t), \dot{q}^j(t)) - B(q^j(t), \dot{q}^j(t)) e_2^j(t) - e_1^j(t) \\ &\quad + D(q^j(t))[\ddot{q}_d(t) - \lambda_1 \dot{e}_1^j(t)] - \frac{\lambda_2}{m_1} e_2^j(t) \end{aligned} \tag{13}$$

with a positive constant λ_2. Let $u^j(t) = u_*^j(t)$, Eq. (12) becomes

$$\dot{V}_2^j(t) = -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t). \tag{14}$$

Since $D(q^j(t))$, $B(q^j(t), \dot{q}^j(t))$ and $F(q^j(t), \dot{q}^j(t))$ of the robotic system are in general unknown or only partially known, the result of Eq. (14) cannot be achieved. However, using the results of Eq. (13) and (14), Eq. (12) can actually be rewritten as

$$\dot{V}_2^j(t) \leq -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) + e_2^j(t)^{\mathrm{T}}(u^j(t) - u_*^j(t)). \tag{15}$$

Step 3

In order to overcome the unknown certainty equivalent controller $u_*^j(t)$, we adopt the FNN $O^{(4)}(q^j(t), \dot{q}^j(t), W^j(t))$ (for detailed, please see in [11]) which is an universal approximator can be further written in a matrix form as follows:

$$O^{(4)}(q^j(t), \dot{q}^j(t), W^j(t)) = W^j(t)^{\mathrm{T}} O^{(3)}(q^j(t), \dot{q}^j(t)), \tag{16}$$

where $W^j(t) \in \Re^{M \times n}$ is the consequent parameter matrix in output layer and $O^{(3)^j}(t) \equiv O^{(3)}(q^j(t), \dot{q}^j(t)) \in \Re^{M \times 1}$ is the fuzzy basis function vector in rule layer, M is the numbers of rule nodes in rule layer. It is well known that the FNN Eq. (16) can uniformly approximate real continuous nonlinear function vector $u_*^j(t)$ on a compact set $A_c \subset R^{n \times 1}$. An important aspect of the above approximation property is that there exists optimal weight matrix W^* such that the function approximation error between the optimal $O^{(4)}(q^j(t), \dot{q}^j(t), W^*)$ and vector $u_*^j(t)$ can be bounded by a prescribed constant θ^* on the compact set A_c. More precisely, if we let $u_*^j(t) = O^{(4)}(q^j(t), \dot{q}^j(t), W^*) + \varepsilon(q^j(t), \dot{q}^j(t))$, then the approximation errors will satisfy $|\varepsilon(q^j(t), \dot{q}^j(t))| \leq \theta^*$, $\forall q^j(t), \dot{q}^j(t) \subset A_c$. Based on the FNN given in Eq. (16), we propose the backstepping AILC $u^j(t)$ as follows:

$$u^j(t) = W^j(t)^{\mathrm{T}} O^{(3)^j}(t) - \mathrm{sgn}(e_2^j(t))\theta^j(t). \tag{17}$$

It is noted that in addition to the network parameter matrix $W^j(t)$, there is an another control parameters $\theta^j(t)$ in the backstepping AILC Eq. (17). In addition, we design $-\mathrm{sgn}(e_2^j(t))\theta^j(t)$ in Eq. (17) by using the concept of a sliding-mode like control force in order to compensate for the network approximation error. However, the optimal weight matrix $W^*$of the FNN and the optimal parameter θ^* for minimum approximation error are generally unknown. In order to see how the adaptive learning system can guarantee both time domain stability and iteration domain convergence, we define the parameter errors as $\tilde{W}^j(t) = W^j(t) - W^*$ and $\tilde{\theta}^j(t) = \theta^j(t) - \theta^*$. Then, substituting Eq. (17) into Eq. (12), we have

$$\dot{V}_2^j(t) \leq -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) + e_2^j(t)^{\mathrm{T}} \tilde{W}^j(t)^{\mathrm{T}} O^{(3)^j}(t) - |e_2^j(t)|\tilde{\theta}^j(t). \tag{18}$$

Step 4

A set of stable adaptive laws for the estimated value $W^j(t)$ and $\theta^j(t)$ are necessary to update the parameters as follows:

$$(1-\gamma_1)\dot{W}^j(t) = -\gamma_1 W^j(t) + \gamma_1 W^{j-1}(t) - \beta_1 O^{(3)^j}(t) e_2^j(t)^{\mathrm{T}}, \tag{19}$$

$$(1-\gamma_2)\dot{\theta}^j(t) = -\gamma_2 \theta^j(t) + \gamma_2 \theta^{j-1}(t) + \beta_2 |e_2^j(t)|, \tag{20}$$

where $W^j(0) = W^{j-1}(T), \theta^j(0) = \theta^{j-1}(T)$ for $j \geq 1$, and $0 < \gamma_1, \gamma_2 < 1$, $\beta_1, \beta_2 > 0$. In this adaptive law, γ_1, γ_2 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. In Fig. 1, we show the block diagram of the proposed backstepping based AILC. For this block diagram, signal generator is used to implement $e_2^j(t)$ by Eqs. (6) and (7). The adaptation law is realized according to Eqs. (19) and (20). The proposed AILC is obviously feasible.

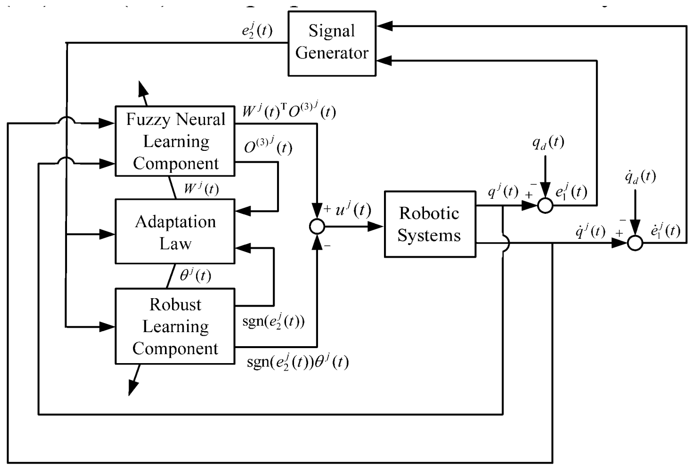

Fig. 1. The block diagram of the backstepping based AILC.

3. ANALYSIS OF ERROR CONVERGENCE AND LEARNING PERFORMANCE

Fact 1

The proposed AILC system guarantees that all the control parameters and internal signals are bounded in the first iteration. First, choose a Lyapunov-like function

$$V_a^j(t) = V_2^j(t) + \frac{(1-\gamma_1)}{2\beta_1} tr\{\tilde{W}^j(t)^{\mathrm{T}}\tilde{W}^j(t)\} + \frac{(1-\gamma_2)}{2\beta_2}(\theta^j(t))^2. \tag{21}$$

By using the similar technique in [12], we compute its derivative with respective to time t along Eqs. (18), (19) and (20) as follows:

$$\begin{aligned}
\dot{V}_a^j(t) &= \dot{V}_2^j(t) + \frac{(1-\gamma_1)}{\beta_1} tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\dot{\tilde{W}}^j(t)\right\} + \frac{(1-\gamma_2)}{\beta_2}\tilde{\theta}^j(t)\dot{\tilde{\theta}}^j(t) \\
&\leq -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) + e_2^j(t)^{\mathrm{T}}\tilde{W}^j(t)^{\mathrm{T}}O^{(3)^j}(t) - \left|e_2^j(t)\right|\tilde{\theta}^j(t) \\
&\quad + \frac{1}{\beta_1} tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\left[(1-\gamma_1)\dot{\hat{W}}^j(t)\right]\right\} + \frac{1}{\beta_2}\tilde{\theta}^j(t)\left[(1-\gamma_2)\dot{\hat{\theta}}^j(t)\right] \\
&\leq -\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) - \frac{\gamma_1}{2\beta_1} tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\tilde{W}^j(t)\right\} - \frac{\gamma_2}{2\beta_2}\left(\tilde{\theta}^j(t)\right)^2 \\
&\quad - \frac{\gamma_1}{2\beta_1} tr\left\{\left(\tilde{W}^j(t) - \tilde{W}^{j-1}(t)\right)^{\mathrm{T}}\left(\tilde{W}^j(t) - \tilde{W}^{j-1}(t)\right)\right\} - \frac{\gamma_2}{2\beta_2}\left(\tilde{\theta}^j(t) - \tilde{\theta}^{j-1}(t)\right)^2 \\
&\quad + \frac{\gamma_1}{2\beta_1} tr\left\{\tilde{W}^{j-1}(t)^{\mathrm{T}}\tilde{W}^{j-1}(t)\right\} + \frac{\gamma_2}{2\beta_2}\left(\tilde{\theta}^{j-1}(t)\right)^2 \\
&\leq \frac{\gamma_1}{2\beta_1} tr\left\{\tilde{W}^{j-1}(t)^{\mathrm{T}}\tilde{W}^{j-1}(t)\right\} + \frac{\gamma_2}{2\beta_2}\left(\tilde{\theta}^{j-1}(t)\right)^2 \equiv V_b^{j-1}(t),
\end{aligned} \tag{22}$$

where we use the properties of $tr\{\tilde{W}^j(t)^{\mathrm{T}}O^{(3)^j}(t)e_2^j(t)^{\mathrm{T}}\} = e_2^j(t)^{\mathrm{T}}\tilde{W}^j(t)^{\mathrm{T}}O^{(3)^j}(t)$, $-\gamma_1 W^j(t) + \gamma_1 W^{j-1}(t) = -\gamma_1\tilde{W}^j(t) + \gamma_1\tilde{W}^{j-1}(t)$ and $-\gamma_2\theta^j(t) + \gamma_2\theta^{j-1}(t) = -\gamma_2\tilde{\theta}^j(t) + \gamma_2\tilde{\theta}^{j-1}(t)$. Since $\tilde{W}^0(t) = W^0(t) - W^* = W^0 - W^* \equiv \bar{W}^0$ and $\tilde{\theta}^0(t) = \theta^0(t) - \theta^* = \theta^0 - \theta^* \equiv \bar{\theta}^0$ are bounded for all $t \in [0,T]$. So if we let $j = 1$, we can rewrite Eq. (22) as follows:

$$\dot{V}_a^1(t) \le V_b^0(t) \equiv \frac{\gamma_1}{2\beta_1} tr\{\bar{W}^{0\mathrm{T}}\bar{W}^0\} + \frac{\gamma_2}{2\beta_2}(\bar{\theta}^0)^2. \tag{23}$$

It is noted that the initial value $V_a^1(0)$ is bounded since $V_2^1(0)=0$, $\tilde{W}^1(0)=W^1(0)-W^*=W^0(T)-W^*=\bar{W}^0$, $\tilde{\theta}^1(0)=\theta^1(0)-\theta^*=\theta^0(T)-\theta^*=\bar{\theta}^0$. Together with the result of Eq. (23), it implies that $V_a^1(t)$, $V_2^1(t)$, $\tilde{W}^1(t)$, $\tilde{\theta}^1(t)\in L_{\infty e}[0,T]$ and hence, $u^1(t)$ (by (17)), $\dot{W}^1(t)$ (by (19)), $\dot{\theta}^1(t)$ (by (20)) $\in L_{\infty e}[0,T]$.

Fact 2
The proposed AILC system guarantees that $\tilde{W}^j(T)$ and $\tilde{\theta}^j(T)$ are bounded for all $j=1$. First, define a positive functions $V^j(T)$ as

$$\begin{aligned} V^j(T) \;&=\; \int_0^T \left[\frac{\gamma_1}{2\beta_1} tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\tilde{W}^j(t)\right\} + \frac{\gamma_2}{2\beta_2}(\tilde{\theta}^j(t))^2\right] dt \\ &\quad + \frac{(1-\gamma_1)}{2\beta_1} tr\left\{\tilde{W}^j(T)^{\mathrm{T}}\tilde{W}^j(T)\right\} + \frac{(1-\gamma_2)}{2\beta_2}(\tilde{\theta}^j(T))^2. \end{aligned} \tag{24}$$

Using the technique of integration by parts, we have

$$\frac{(1-\gamma_1)}{2\beta_1} tr\left\{\tilde{W}^j(T)^{\mathrm{T}}\tilde{W}^j(T)\right\} = \frac{(1-\gamma_1)}{\beta_1}\int_0^T tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\dot{\tilde{W}}^j(t)\right\} dt + \frac{(1-\gamma_1)}{2\beta_1} tr\left\{\tilde{W}^j(0)^{\mathrm{T}}\tilde{W}^j(0)\right\}$$

$$\frac{(1-\gamma_2)}{2\beta_2}\left(\tilde{\theta}^j(T)\right)^2 = \frac{(1-\gamma_2)}{\beta_2}\int_0^T \tilde{\theta}^j(t)\dot{\tilde{\theta}}^j(t)dt + \frac{(1-\gamma_2)}{2\beta_2}\left(\tilde{\theta}^j(0)\right)^2.$$

By using the facts of $\tilde{W}^j(0)=\tilde{W}^{j-1}(T)$, $\tilde{\theta}^j(0)=\tilde{\theta}^{j-1}(T)$ and the integration of both sides in Eq. (18) from 0 to T, we can further derive the difference between $V^j(T)$ and $V^{j-1}(T)$ by using a similar technique as in [12]:

$$\begin{aligned} &V^j(T)-V^{j-1}(T) \\ &= \int_0^T \left[\frac{\gamma_1}{2\beta_1}\left(tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\tilde{W}^j(t)\right\} - tr\left\{\tilde{W}^{j-1}(t)^{\mathrm{T}}\tilde{W}^{j-1}(t)\right\}\right) + \frac{\gamma_2}{2\beta_2}\left((\tilde{\theta}^j(t))^2-(\tilde{\theta}^{j-1}(t))^2\right)\right] dt \\ &\quad + \frac{(1-\gamma_1)}{\beta_1}\int_0^T tr\left\{\tilde{W}^j(t)^{\mathrm{T}}\dot{\tilde{W}}^j(t)\right\} dt + \frac{(1-\gamma_1)}{2\beta_1} tr\left\{\tilde{W}^j(0)^{\mathrm{T}}\tilde{W}^j(0)\right\} \\ &\quad - \frac{(1-\gamma_1)}{2\beta_1} tr\left\{\tilde{W}^{j-1}(T)^{\mathrm{T}}\tilde{W}^{j-1}(T)\right\} + \frac{(1-\gamma_2)}{\beta_2}\int_0^T \tilde{\theta}^j(t)\dot{\tilde{\theta}}^j(t)dt \\ &\quad + \frac{(1-\gamma_2)}{2\beta_2}\left(\tilde{\theta}^j(0)\right)^2 - \frac{(1-\gamma_2)}{2\beta_2}\left(\tilde{\theta}^{j-1}(T)\right)^2 \\ &\le \int_0^T \left[-\lambda_1 e_1^j(t)^{\mathrm{T}}e_1^j(t) - \lambda_2 e_2^j(t)^{\mathrm{T}}e_2^j(t)\right] dt - V_2^j(T), \end{aligned} \tag{25}$$

where we use the property of $V_2^j(0)=0$. Since $V^1(T)$ is bounded by Fact 1, and is positive and monotonically decreasing, $V^j(T)$ is bounded for all $j\ge 1$ and will converge as approaches infinity to some limit value $V(T)$ (independent of j). The boundedness of $V^j(T)$ also ensures the boundedness of $W^j(T), \theta^j(T)\in L_{\infty e}[0,T]$ for all $j\ge 1$. The boundedness of $W^j(T)$ and $\theta^j(T)$ (or equivalently the boundedness of $W^j(0)$ and $\theta^j(0)$ for all $j\ge 1$ is shown in Fact 2, the convergence of $e_1^j(t), e_2^j(t)$ and boundedness of all internal signals for all $j \geq 1$ are now established in the following lemma.

Lemma 1. The proposed AILC system ensures that all adjustable control parameters and internal signals $e_1^j(t), e_2^j(t), W^j(t), \theta^j(t), \dot{e}_1^j(t), \dot{e}_2^j(t), \dot{W}^j(t), \dot{\theta}^j(t), u^j(t) \in L_{\infty e}[0,T]$ for all $j \geq 1$.

Proof. Integrating Eq. (22) from 0 to t, we have

$$V_a^j(t) \leq V_a^j(0) + \int_0^t V_b^{j-1}(t')dt' \leq V_a^j(0) + \int_0^T V_b^{j-1}(t)dt. \tag{26}$$

Since $V^j(T)$, defined in Eq. (24), is bounded $\forall j \geq 1$ according to Fact 2, we conclude that $\int_0^T V_b^{j-1}(t)dt$ is bounded $\forall j \geq 1$. Furthermore, the initial value $V_a^j 0)$ of $V_a^j(t)$ is also bounded for all $j \geq 1$ due to Fact 2. This implies from Eq. (26) that $V_a^j(t)$ and hence, $e_1^j(t), e_2^j(t), \tilde{W}^j(t), \tilde{\theta}^j(t) \in L_{\infty e}[0,T]$. Using the same argument as in Fact 1, it can be easily shown that $e_1^j(t), e_2^j(t), W^j(t), \theta^j(t), \dot{e}_1^j(t), \dot{e}_2^j(t), \dot{W}^j(t), \dot{\theta}^j(t), u^j(t) \in L_{\infty e}[0,T]$ all $j \geq 1$.

Theorem 1. The proposed AILC system guarantees that the tracking performance and system stability will satisfy the following results:

(t1) $\lim_{j\to\infty} \lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) = 0$ and $\lim_{j\to\infty} \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) = 0$, for all $t \in [0,T]$.

(t2) $\lim_{j\to\infty} \left|q^j(t) - q_d(t)\right| = 0$ and $\lim_{j\to\infty} \left|\dot{q}^j(t) - \dot{q}_d(t)\right| = 0$, for all $t \in [0,T]$.

Proof. (t1) According to Lemma 1, we have $e_1^j(t), e_2^j(t) \in L_{\infty e}[0,T]$ and $\dot{e}_1^j(t), \dot{e}_2^j(t) \in L_{\infty e}[0,T]$ for all $j \geq 1$. This implies that $e_1^j(t)$ and $e_2^j(t)$ are uniformly continuous over $[0,T]$ for all $j \geq 1$. On the other hand, we have

$$\int_0^T [\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) + \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t)]dt \leq V^{j-1}(T) - V^j(T) \leq V^1(T) \tag{27}$$

for all $j \geq 1$ by using Eq. (28). Equation (30) implies that

$$\lim_{j\to\infty} \int_0^T \left[\lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) + \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t)\right] dt = 0. \tag{28}$$

We can ensure by using Barbalat's lemma [5] that $\lim_{j\to\infty} \lambda_1 e_1^j(t)^{\mathrm{T}} e_1^j(t) = 0$ and $\lim_{j\to\infty} \lambda_2 e_2^j(t)^{\mathrm{T}} e_2^j(t) = 0$, for all $t \in [0,T]$.

(t2) Since $\lim_{j\to\infty} e_1^j(t) = 0$ and $\lim_{j\to\infty} e_2^j(t) = 0$, for all $t \in [0,T]$, we have $\lim_{j\to\infty} \alpha^j(t) = \dot{q}_d(t)$. Therefore, $\lim_{j\to\infty} \left|q^j(t) - q_d(t)\right| = 0$ and $\lim_{j\to\infty} \left|\dot{q}^j(t) - \dot{q}_d(t)\right| = 0$, for all $t \in [0,T]$. This proves (t2) of Theorem 1.

4. SIMULATION EXAMPLE

In this section, a computer simulation is conducted to show the learning effectiveness of the proposed AILC. First, we consider a two-link planar robotic system [11] as follows:

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

where $D_{11}^j = m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos(q_2^j(t))) + I_1 + I_2$, $D_{12}^j = D_{21}^j = m_2 l_1 l_{c2}\cos(q_2^j(t)) + m_2 l_{c2}^2 + I_2$, $D_{22}^j = m_2 l_{c2}^2 + I_2, h = m_2 l_1 l_{c2}\sin(q_2^j(t))$. Here m_i, I_i, l_i and l_{c_i} represent mass, inertia, length of link i, and the

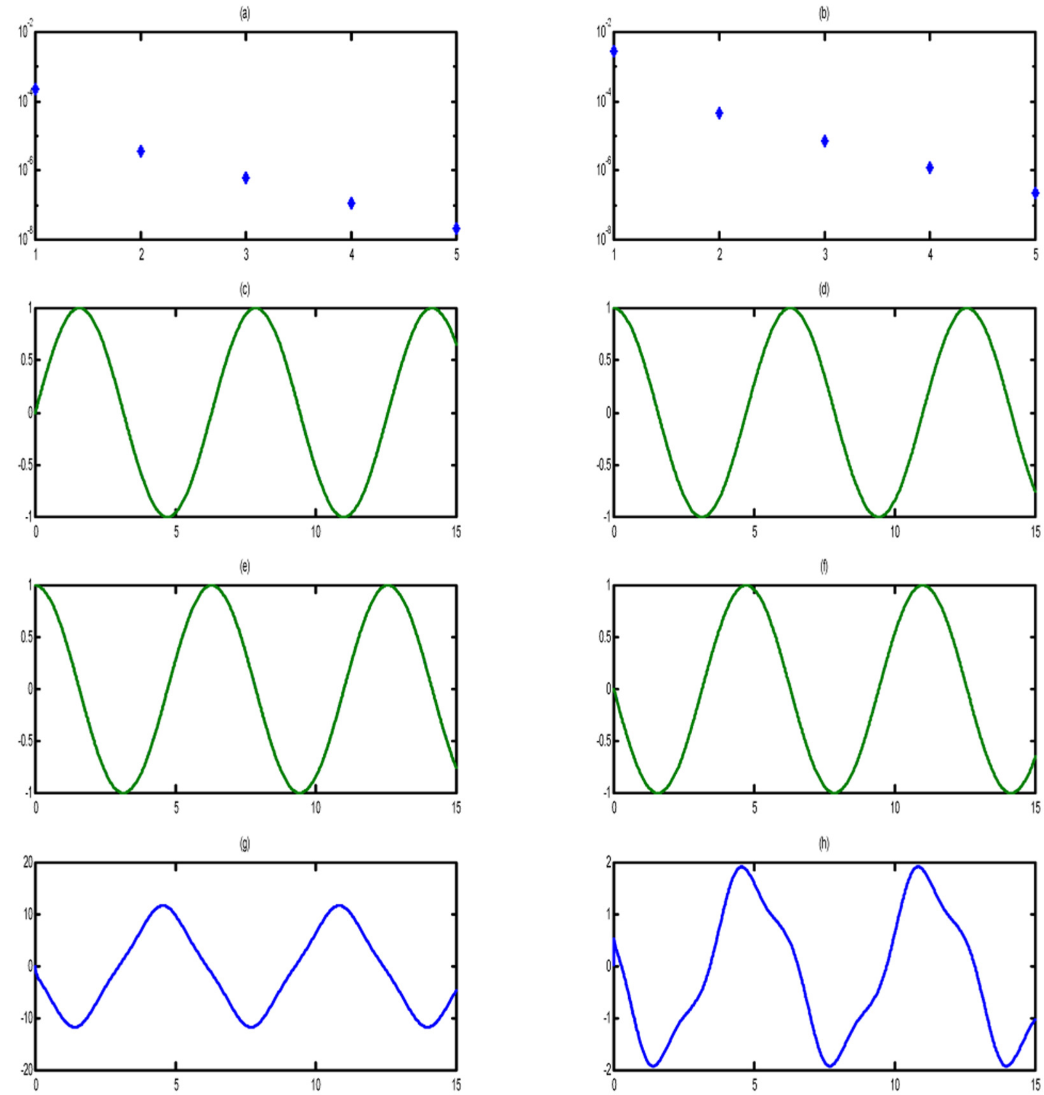

Fig. 2. (a) $\sup_{t\in[0,15]}|e_1^j(t)|$ versus j; (b) $\sup_{t\in[0,15]}|e_2^j(t)|$ versus j; (c) $q_1^5(t)$ (solid line) and $q_{d_1}(t)$ (dotted line) versus time t; (d) $q_2^5(t)$ (solid line) and $q_{d_2}(t)$(dotted line); (e)$\dot{q}_1^5(t)$ (solid line) and $\dot{q}_{d_1}(t)$ (dotted line) versus time t; (f) $\dot{q}_2^5(t)$ (solid line) and $\dot{q}_{d_2}(t)$ (dotted line); (g) $u_1^5(t)$versus time t; (h) $u_2^5(t)$ versus time t.

distance from the previous joint to the center of mass of link i, respectively. In this simulation example, we set m_1 = 10 kg, m_2 = 5 kg, l_1 = 1 m, l_2 = 0.5 cm, l_{c1} = 0.5 m, l_{c2} = 0.25 m, $I_1 = 0.83kg-m^2$, I_2 = 0.3 kg-m^2. The control objective is to let $q^j(t) = [q_1^j(t), q_2^j(t)]^{\mathrm{T}}$ track the desired trajectory $q_d(t) = [q_{d1}(t), q_{d2}(t)]^{\mathrm{T}} = [\sin(t), \cos(t)]^{\mathrm{T}}$ over a finite time interval $[0, 15]$. The design steps are summarized as follows:

(D1) Define the output error $e_1^j(t) = q^j(t) - q_d(t)$, stabilizing function $\alpha^j(t) = -\lambda_1 e_1^j(t) + \dot{q}_d(t)$ where $\lambda_1 = 10 > 0$, and the error function $e_2^j(t) = \dot{q}^j(t) - \alpha^j(t)$, respectively.

(D2) Construct the membership functions for $[q^j(t), \dot{q}^j(t)]^{\mathrm{T}}$. Then, solve the basis function vector $O^{(3)}(q^j(t), \dot{q}^j(t))$. The FNN is given as $W^j(t)^{\mathrm{T}} O^{(3)}(q^j(t), \dot{q}^j(t))$. In addition, we choose the centers as $m^0(t) = m^0 = [m_1^0, m_2^0, m_3^0, m_4^0]$ with $m_i^0 = [m_{i1}^0, m_{i2}^0, m_{i3}^0] = [-1.5, 0, 1.5]$, $i = 1, 2, 3, 4$ and variances as

$\sigma^0(t) = \sigma^0 = [\sigma_1^0, \sigma_2^0, \sigma_3^0, \sigma_4^0]$, $\sigma_i^0 = [\sigma_{i1}^0, \sigma_{i2}^0, \sigma_{i3}^0] = [3,3,3]$, $i = 1,2,3,4$, respectively. In addition, we set the control parameter $\theta^0(t) = \theta^0 = 5$ at the first iteration for all $t \in [0,15]$. It is noted that the initial values of the consequent parameters $W^0(t) = W^0$ can be roughly estimated if the the certainty equivalent controller $u_*^j(t)$ of the robotic system is partially known. However, we often arbitrarily choose these initial parameters.

(D3) Design the controller $u^j(t) = W^j(t)^{\mathrm{T}} O^{(3)^j}(t) - \mathrm{sgn}(e_2^j(t))\theta^j(t)$.

(D4) Finally, the adaptation algorithms Eqs. (19) and (20) are adopted with the weighing gains $\gamma_1 = \gamma_2 = 0.5$ and the learning gains $\beta_1 = \beta_2 = 10$.

To study the effect of learning performances, the supremum values of $|e_1^j(t)|$ and $|e_2^j(t)|$ versus iteration j are shown in Figs. 2a and 1b, respectively. In this simulation, it is clear to show the asymptotic convergences proved in the technical result (t1) of Theorem 1. Since the nice tracking performances of both joint position vector and joint velocity vector at the fifth iteration are achieved, we investigate the relations between joint position vector $q^j(t)$, joint velocity vector $\dot{q}^j(t)$ and desired joint position vector $q_d(t)$, desired joint velocity vector $\dot{q}_d(t)$. The nice tracking performance of both joint position and joint velocity at the 5th iteration are plotted in Figs. 2c, 2d, 2e, and 2f, respectively. Finally, the bounded learned control forces $u_1^5(t)$ and $u_2^5(t)$ are given in Figs. 2g and Fig. 2h, respectively.

5. CONCLUSIONS

For a repetitive control task of robotic system, a backstepping AILC is proposed in this paper. The back-stepping like procedure is first introduced to design the main structure of the AILC. In order to guarantee the boundedness of internal signals, we design the AILC with two components including a fuzzy neural network (FNN) used to approximate unknown certainty equivalent controller and a robust learning term used to compensate for uncertainty from the network approximation error. The adaptive laws combining time domain and iteration domain adaptation for the FNN parameters and control parameters are proposed to ensure the stability and convergence of the learning system. Finally, we apply a Lyapunov like analysis to show that the adjustable parameters as well as internal signals remain bounded and the tracking error will asymptotically converge to zero as iteration goes to infinity.

ACKNOWLEDGEMENT

This work is supported by National Science Council, R.O.C., under Grants NSC 100-2221-E-211-008, NSC102-2221-E-211-011 and NSC102-2221-E-211-003.

REFERENCES

1. Ahn., H.S., Chen, Y. and Moore, K.L., "Iterative learning control: Brief survey and categorization", *IEEE Transactions on Systems, Man, and Cybernetics, Part C – Applications and Reviews*, Vol. 37, No. 6, pp. 1099–1121, 2007.
2. Wang, Y., Gaoc, F. and Doyle, F.J., "Survey on iterative learning control, repetitive control, and run-to-run control", *Journal of Process Control*, Vol. 19, No. 10, pp. 1589–1600, 2009.
3. Chien, C.J. and Tayebi, A., "Further results on adaptive iterative learning control of robot manipulators", *Automatica*, Vol. 44, No. 3, pp. 830–837, 2008.
4. Ngo, T., Wang, Y., Mai, T.L., Ge, J., Nguyen, M.H. and Wei, S.N., "An adaptive iterative learning control for robot manipulator in task space", *International Journal of Computers, Communications and Control*, Vol. 7, No. 3, pp. 510–521, 2012.
5. Khalil, H.K., *Nonlinear Systems*, Prentice Hall, NJ, 2000.

6. Zou, A.M., Hou, Z.G. and Tan, M., "Adaptive control of a class of nonlinear pure-feedback systems using fuzzy backstepping approach", *IEEE Transactions on Fuzzy Systems*, Vol. 16, No. 4, pp. 886–897, 2008.
7. Chen, W. and Li, J., "Globally decentralized adaptive backstepping neural network tracking control for unknown nonlinear interconnected systems", *Asian Journal of Control*, Vol. 12, No. 1, pp. 96–102, 2010.
8. Tayebi, A., Polushin, I.G. and Chien, C.J., "Cascaded iterative learning control for a class of uncertain time-varying nonlinear systems", in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006.
9. Li, J.M., Sun, Y.P. and Liu, Y., "Hybrid adaptive iterative learning control of non-uniform trajectory tracking", *Control Theory & Application*, Vol. 25, No. 1, pp. 100–104, 2008.
10. Liu, P., Guo, Y., Hu, Y., Xiang, Z.R., "Robust adaptive iterative learning control for a class of second order uncertain nonlinear systems", in *Proceedings of the 27th Chinese Control Conference*, 2008.
11. Wang, Y.C. and Chien, C.J., "A fuzzy neural network direct adaptive iterative learning controller for robot manipulators", in *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2008.
12. Wang, Y.C. and Chien, C.J., "Repetitive tracking control of nonlinear systems using reinforcement fuzzy-neural adaptive iterative learning controller", *Applied Mathematics & Information Sciences*, Vol. 6, No. 3, pp. 473–481, 2012.