2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Asymptotically Stable Adaptive Critic Design for Uncertain Nonlinear Systems

Jianguo Yao, Xue Liu, *Member, IEEE*, Xiaoping Zhu

Abstract—**Recently, Adaptive critic design (ACD) has been applied to controller design extensively. It is a powerful approach to cope with the model nonlinearity and uncertainties. Existing ACD-based controllers have been proven as uniformly ultimately bounded (UUB). However, UUB only makes the tracking error converge to a certain bounded region. Although we can minimize the bounded region by increasing the number of the hidden nodes of the neural networks in the ACD, the computation cost of the ACD-based controller increases. In many engineering applications, we prefer the asymptotical stability which can ensure the tracking error converges to zero. In this paper, we propose a novel asymptotically stable ACD-based controller for a class of uncertain nonlinear systems. This controller firstly uses the feedback linearization to improve the system dynamic characteristics, and then combines ACD and variable structure control to achieve the asymptotical stability under large model uncertainties. An empirical study is conducted on a 2-link manipulator to validate the new controller design approach. Results show that the nonlinear system using the proposed controller can achieve asymptotical stability and good dynamic response characteristics when large model uncertainties exist.**

I. Introduction

Adaptive critic design (ACD) schemes, both in continuous and discrete time forms, have been developed in the past few years [1][2][3]. ACD originates from the combination of Dynamic Programming and Reinforcement Learning [4], which is used to solve nonlinear optimal control problems online or offline without the need of accurate plant models. In the ACD architecture, there are two networks termed the *critic network* and the *action network* [5]. The *critic network* approximates the cost-to-go function describing the performance of the system; and the *action network* gives optimal action by minimizing the output of critic network. In both of the critic network and the action network, neural networks are employed to approximate the cost-to-go function and the action inputs respectively.

The stability of the ACD is studied in [4][6][7]. In exiting literature, ACD-based controllers have been proven as uniformly ultimately bounded (UUB) [8][9][10][11]. Although UUB is useful in the bounded error control cases, it is not enough for precise tracking applications. For example, when an airplane's attitude is disturbed from its nominal position by the wind, we not only expect the airplane to maintain its attitude in a certain range which is determined by the magnitude of the disturbance, i.e., UUB, but also require that the attitude gradually goes back to the original value [12]. This type of engineering requirement is referred to as asymptotic stability. In the controller design based on ACD, although the bounded error region can be minimized by increasing the number of the hidden nodes in the neural networks, computing the control law consumes more time.

In this paper, a novel asymptotically stable ACD-based controller is developed for uncertain nonlinear systems. In previous research on controller designs, many models have been setup in many engineering applications. However, these models are usually discarded in many ACD-based controller designs [13],[14]. Although these models may not be as accurate as they are required, they are useful for improving the system dynamic characteristics in the design. In our approach, the plant is approximately linearized employing feedback linearization method using the known nominal model. Actually, the nominal model is given with model uncertainty because the accurate plant usually is not available. To make the controller robust, ACD is employed to approximate the unknown model uncertainty. The ACD-based controller proposed in this paper is different from the original formulation. In our architecture, the critic network is not used to approximate the cost-to-go function. The output of the critic network is an adaptive control item to adjust the action network. Similar schemes are proposed in [8][9][11]. However, these schemes are only proven to hold the UUB property. To ensure the system as asymptotically stable, variable structure control (VSC) is employed in our scheme to compensate the approximated error in the action network. The updating of all the controller parameters is performed online, without the offline learning required in the traditional ACD schemes.

The reminder of the paper is organized as follows: Section II gives the problem statement. Section III introduces the background knowledge on the neural networks and Lyapunov stability theorems. Section IV presents a general controller structure employing feedback linearization. Section V gives the detailed description of the asymptotically stable controller design based on the combination of ACD and VSC. Section VI presents the asymptotical stability proof of the proposed controller. Section VII conducts a simulation using the asymptotically stable ACD controller for a 2-link manipulator and

This work was supported in part by an NSERC Discovery Grant #341823-07 and a National Study-Abroad Scholarship of P. R. China under Grant No. [2007] 3020.

Jianguo Yao is a Ph.D student at the School of Astronautics, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, P.R.CHINA. He is also a Visiting Ph.D student at the School of Computer Science, McGill University, Montreal, QC H3A2A7 Canada, (e-mail: jianguo@cs.mcgill.ca)

Xue Liu is an assistant professor at the School of Computer Science, McGill University, Montreal, QC H3A2A7 Canada, (phone: 514-398-1819; fax: 514-398-3883; e-mail: xueliu@cs.mcgill.ca).

Xiaoping Zhu is a professor at the UAV Research and Development Center, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, P.R.CHINA.

978-1-4244-4524-0/09/$25.00 ©2009 AACC

summarizes the results. The conclusion is given in Section VIII.

II. Problem Statement

In this paper, we consider a class of nonlinear systems given by the following general formulation

$$\begin{aligned} \dot{x}_1 &= x_2 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= f(x) + g(x)u(t) + d(t) \\ y &= x_1 \end{aligned} \quad , \tag{1}$$

where $x = [x_1 \quad x_2 \quad \cdots \quad x_n]^T$ is the system state vector, $u(t)$ is the control input, and $y(t)$ is the output of the system. $d(t)$ is the unknown disturbance with a known upper bound so that $d(t) < d_M$. It is assumed that the nonlinear function $f(x)$ and $g(x)$ are smooth and they consist of the nominal part and the uncertain part as $f(x) = \hat{f}(x) + \Delta f(x)$ and $g(x) = \hat{g}(x) + \Delta g(x)$ respectively, where $\hat{f}(x)$ and $\hat{g}(x)$ are the nominal parts which are known, $\Delta f(x)$ and $\Delta g(x)$ are the uncertain parts which are unknown.

In many studies on adaptive critic design (ACD), the filtered tracking error is defined as [8][11]:

$$r(t) = e^{(n-1)}(t) + \lambda_{n-2} e^{(n-2)}(t) + \cdots + \lambda_1 e^{(1)}(t) + e(t) , \tag{2}$$

where $e(t) = x_d(t) - x_1(t)$ is the tracking error, $x_d(t)$ is the desired trajectory. $e^{(n-1)}(t), \cdots, e^{(1)}(t)$ are the derivatives of the tracking error $e(t)$. $\lambda_{n-2}, \lambda_{n-3}, \cdots, \lambda_1$ are the design parameters so that $s^{n-1} + \lambda_{n-2} s^{n-2} + \cdots + \lambda_1$ is Hurwitz. This means that when $r(t)$ converges to 0 and then $e(t)$ also converges to 0.

Note that the filtered tracking error is actually a sliding surface in variable structure control (VSC). According to the theorem of VSC, the VSC control law makes sure the states of the system reach the sliding surface. When the system is constrained by VSC to stay on the sliding surface, the system dynamics are governed by $r(t) = 0$. To maintain the system at $r(t) = 0$, the following action input is adopted in the exiting literature [9]

$$u(t) = g_N^{-1}(x)[-K_v r - f_N(x) + v(t)] , \tag{3}$$

where $g(x)$ is assumed as a known function, K_v is the gain matrix which is positive definite. $f_N(x)$ is the approximate matrix of the nonlinear function $f(x)$ by neural network, and $v(t)$ is a robustifying vector.

Note that it is assumed there is no prior knowledge of the function $f(x)$ in (3). Actually, the nominal part $\hat{f}(x)$ of $f(x)$ is known in many engineering applications. To make full use of the prior plant model to improve the dynamic characteristics, the feedback linearization is conducted firstly using the nominal plant model consisting of $\hat{f}(x)$ and $\hat{g}(x)$ in our approach.

The stability of the existing ACD-based controllers are proven to be uniformly ultimately bounded (UUB) [8][9][10][11]. This means that the filtered tracking error is bounded and converges to a certain residual set which is determined by the approximate errors in the neural networks. Although we can minimize the residual set by increasing the number of hidden nodes of the neural networks in ACD, it results in the increment of the computing cost. However when the number of the hidden nodes is small, the system performance may deteriorate. In the engineering practice of the controller design, asymptotical stability is preferred which means the tracking error converges to 0 asymptotically. In the following sections, an asymptotically stable ACD-based controller will be studied.

III. Background Knowledge

The objective of this section is to present two aspects of knowledge. The first one is the neural networks, which is used to approximate the model uncertainty in this paper. The other is Lyapunov stability theorem, which is employed to analyze the stability of our controller design.

A. Neural Networks

In this paper, we apply two-layer feedforward neural networks to approximate the model uncertainty. The neural networks consist of a hidden layer and an output layer. They are described as

$$y_N = W^T \sigma(V^T \bar{x}) , \tag{4}$$

where $\bar{x} = [1 \quad x^T]^T$ is the augmenting vector, y_N is the output of the neural networks. V is the weight matrix in the hidden layer. W is the weight matrix in the output layer. $\sigma(.)$ is the activation function, which is chosen as $\sigma(.) = 1/[1 + \exp(.)]$ in this paper.

Neural networks are extensively used to approximate a general function $f(x)$, where $f(x)$ is a smooth linear or a nonlinear function. According to the Stone-Weirstrass theorem, the approximation equation of neural networks can be given in [15],[16], as

$$f(x) = W^T \sigma(V^T \bar{x}) + \varepsilon , \tag{5}$$

where ε is a neural network reconstruction error. Note that for all the constant vector ε_N, we can construct a neural network so that $\|\varepsilon\| < \varepsilon_N$. That means ε can be decreased by increasing the number of hidden nodes in the neural networks.

B. Lyapunov Stability

Lemma 1: Uniformly Ultimately Bounded (UUB) [17]

Let $L(x)$ be a Lyapunov function of a continuous-time system that satisfies the following properties

$$\begin{aligned} \gamma_1(\|x\|) \le L(x) \le \gamma_2(\|x\|) \\ \dot{L}(x) \le \gamma_3(\|x\|) + \gamma_3(\eta) \end{aligned} , \tag{6}$$

where η is a positive constant , γ_1 and γ_2 are continuous strictly increasing functions, and γ_3 is a continuous, non-decreasing function, then if

$$\dot{L}(x) < 0, \textit{for } \|x(t)\| > \eta , \tag{7}$$

Then $x(t)$ is uniformly ultimately bounded. In addition, if $x(0)=0$, $x(t)$ is uniformly bounded.

Lemma 2: Lyapunov's theorem for local stability [12]

If, in a ball B_{R_0}, there exists a function $L(x): R^n \to R$, where $L(x)$ is called a Lyapunov function candidate, such that:

a) $L(x)>0$ with equality if and only if $x = 0$, that means positive definite;

b) $\dot{L}(x)<0$, that means the derivative of the $L(x)$ is negative definite.

Then the system is asymptotically stable in the sense of Lyapunov.

There are two steps in applying the above theorems to analyze the stability of a nonlinear system. The first step is to choose an appropriate positive Lyapunov function; and the second step is to determine its derivative along the path of the nonlinear systems.

IV. THE CONTROLLER USING FEEDBACK LINEARIZATION

In this section, the control law is deduced based on feedback linearization using the nominal plant model. The UUB stability proof of the feedback linearization based controller is also given in this section.

Using the Equation (1) and (2), the dynamics of the filtered tracking error can be rewritten as

$$\begin{aligned}\dot{r} &= \hat{f}(x)+\hat{g}(x)u - x_d^{(n)} + \lambda_{n-2}e^{(n-1)}(t) \\ &+\cdots+\lambda_1 e^{(2)}(t)+\dot{e}(t)+\Delta f(x)+\Delta g(x)u+d(t)\end{aligned} \quad (8)$$

where $\hat{f}(x)$ and $\hat{g}(x)$ are the nominal functions.

We use feedback linearization to get the control input $u(t)$ as follows

$$\begin{aligned}u(t) &= \hat{g}^{-1}(x)[-K_v r - \hat{f}(x) + x_d^{(n)} \\ &-\lambda_{n-2}e^{(n-1)}(t)-\cdots-\lambda_1 e^{(2)}(t)-\dot{e}(t)]\end{aligned}. \quad (9)$$

Substituting (9) into (8), the closed-loop system becomes

$$\dot{r} = -K_v r + p(x,u) + d(t), \quad (10)$$

where $p(x,u)=\Delta f(x)+\Delta g(x)u$ is the model uncertainty which is derived from the inaccuracy of the nominal model.

To evaluate the stability of the closed-loop system, Lyapunov function is defined as

$$L = \frac{1}{2} r^T r > 0. \quad (11)$$

The derivative of the Lyapunov function is

$$\begin{aligned}\dot{L} &= -r^T K_v r + r^T[p(x,u)+d(t)] \\ &< -K_v^{\min}\|r\|^2 + \|r\|(\Delta f_M + \Delta g_M u + d_M)\end{aligned}, \quad \textbf{(12)}$$

where Δf_M and Δg_M are the upper bounds so that $\|\Delta f(x)\| < \Delta f_M$ and $\|\Delta g(x)\| < \Delta g_M$. $K_v^{\min}$ is the smallest singular value of the gain matrix K_v.

Equation (12) implies that $\dot{L}<0$ provided that $-K_v^{\min}\|r\|^2 + \|r\|(\Delta f_M + \Delta g_M u + d_M) < 0$. This further implies that we can get $\dot{L}<0$ as long as

$$\|r\| > \frac{\Delta f_M + \Delta g_M u + d_M}{K_v^{\min}}. \quad (13)$$

According to the *Lemma 1*, the closed-loop system employing the control law (9) is UUB. However, we prefer that the filtered the tracking error asymptotically converge to 0. In the next section, we will discuss how to make the closed-loop system asymptotically stable using ACD and VSC.

V. THE ASYMPTOTICALLY STABLE ADAPTIVE CRITIC CONTROLLER ARCHITECTURE

In this section, a new asymptotically stable controller is proposed. This controller is obtained by combining ACD and VSC. To analyze the stability of the closed-loop system, a Lyapunov function is designed, and the proof of the asymptotical stability is given in the next section.

A. The Architecture of the Proposed Controller Design

Fig. 1 shows the proposed controller architecture. The control law has three entities. The first is the feedback linearization control item, which is determined by the nominal model and the gain matrix K_v. The second is the ACD item, which is the output of the action network. In the updating law of ACD, the weight matrix in the critic network is tuned by itself and the weight matrix in the action network is updated by the critic network. The third is the VSC item, which is adjusted by the critic network.

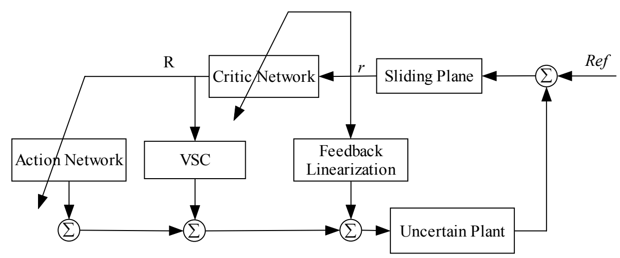

Fig. 1 The architecture of the proposed controller design

B. The Critic Network

The critic network is defined as a neural network

$$R = W_1^T \sigma(V_1^T r) + \rho, \quad (14)$$

where ρ is an adaptive parameter. W_1 is the weight matrix in the critic network, V_1 is randomized in the initialization which is constant in the control process.

The weight matrix in the critic network is updated by

$$\dot{W}_1 = -\Gamma_1 R \sigma(V_1^T r)^T, \quad (15)$$

where Γ_1 is a diagonal positive definitive matrix can be chosen by the designer.

The updating law of the parameter ρ is defined as

$$\dot{\rho} = W_1^T \sigma'(V_1^T r) V_1^T K_v r, \quad (16)$$

where $\sigma'(\cdot)$ is the derivate of the function $\sigma(\cdot)$

C. The Action Network

To make the control law robust to the model uncertainty $\Delta f(x)$ and $\Delta g(x)$, the action network is applied to approximate $p(x,u)$ using neural networks. In the approximation architecture, the action network has the following formulation

$$p(s) = W_2^T \sigma(V_2^T s) + \varepsilon(s), \quad (17)$$

where $s = [1 \;\; x^T \;\; u^T]^T$, W_2 is the ideal weight to approximate the uncertainty. V_2 is randomized in the initialization and it is constant in the control process. $\varepsilon(s)$ is the neural network reconstruction error which is bounded by $\|\varepsilon_1(x)\| < \varepsilon_{N1}$. It was shown in [18] that the bigger the number of hidden layer nodes, the smaller the neural network reconstruction error.

The output of the action network is defined to approximate the model uncertainty $p(s)$ as

$$\hat{p}(s) = \hat{W}_2^T \sigma(V_2^T s), \tag{18}$$

where $\hat{p}(s)$ is the current approximated value of the model uncertainty $p(s)$. $\hat{W}_2$ is the current estimation of the real weight W_2.

The weight matrix in the action network is updated by

$$\dot{\hat{W}}_2 = \Gamma_2^{-1}\sigma(V_2^T s)r^T + \Gamma_2^{-1}\sigma(V_2^T s)R^T \hat{W}_1^T \sigma'(V_1^T r)V_1^T, \tag{19}$$

where Γ_2 is a diagonal positive definitive matrix chosen by the designer.

D. The New Controller Law Formulation

The new control law is defined as

$$\begin{aligned} u(t) = \hat{g}^{-1}(x)[&-K_v r - \hat{p}(x,u) + v - \hat{f}(x) + x_d^{(n)} \\ &-\lambda_{n-2} e^{(n-1)}(t) - \cdots - \lambda_1 e^{(2)}(t) - \dot{e}(t)] \end{aligned}, \tag{20}$$

where v is the VSC item or a robustifying vector. It is defined as

$$v_1(i) = \begin{cases} < -\zeta(i), f(i) > 0 \\ > \zeta(i), f(i) < 0 \end{cases}, f = r^T + R^T \hat{W}_1^T \sigma'(V_1^T r)V_1^T, \tag{21}$$

where $\zeta = \varepsilon_{N1} + d_M$ is the upper bound of $\varepsilon_1 + d(t)$.

Using the control law (20), the closed-loop system is formulated as

$$\dot{r} = -K_v r + v + \tilde{p}(s) + d(t), \tag{22}$$

where $\tilde{p}(s)$ is the current estimation error of $p(s)$. It is also defined as

$$\tilde{p}(s) = \tilde{W}_2^T \sigma(V_2^T s) + \varepsilon(s), \tag{23}$$

where $\tilde{W}_2 = W_2 - \hat{W}_2$ is the current estimation error of the weight matrix in the action network.

VI. The Stability Analysis

In this section, we make a study on the stability of the proposed controller in Section V. According to the Lyapunov stability theorem, the proof of asymptotical stability is given.

Before crafting the Lyapunov function, the following assumptions are given

$$\|W_1\| \le W_{1m}, \; \|W_2\| \le W_{2m}, \tag{24}$$

where W_{1m}, W_{2m} are the bounds on the weight matrices W_1, W_2.

In [8] [9] [11], the variables $r, R, \tilde{W}_1, \tilde{W}_2$ are selected to construct the Lyapunov function. Note that the goal of the controller is that the filtered tracking error r and the output R of the critic network are stable, so r and R are selected in the construction of the Lyapunov function; in addition, the action network is used to approximate the model uncertainty, and the objective of the estimation is to make the error of the weight $\tilde{W}_2$ converge to 0, so $\tilde{W}_2$ is selected to construct the Lyapunov function. In contrast, the critic network is not used to approximate a function as in tradition ACD design: it is used as an adaptive item to adjust the action network and VSC. So the approximation error $\tilde{W}_1$ in the critic network does not exist. Actually, the convergence of R denotes that both of the weight matrix W_1 and adaptive parameter ρ in the critic network are convergent. Based on the above analysis, we choose Lyapunov function constructed by the $r, R, \tilde{W}_2$, which are different from the formulation in the former literatures [8][9][11]. The Lyapunov function is defined as

$$L = \frac{1}{2} r^T r + \frac{1}{2} R^T R + \frac{1}{2} tr(\tilde{W}_2^T \Gamma_2 \tilde{W}_2), \tag{25}$$

where $tr(\cdot)$ is the operator of the matrix trace.

Proof: Calculating the derivative of L

$$\dot{L} = r^T \dot{r} + R^T \dot{R} + tr(\tilde{W}_2^T \Gamma_2 \dot{\tilde{W}}_2). \tag{26}$$

Substituting (14) and (22) into (26), we can get

$$\begin{aligned} \dot{L} = &-r^T K_v r + r^T[\varepsilon_1 + d(t) + v] + r^T \tilde{W}_2^T \sigma(V_2^T s) \\ &+ R^T \dot{\hat{W}}_1^T \sigma(V_1^T r) + R^T \hat{W}_1^T \sigma'(V_1^T r) V_1^T K_v r \\ &+ R^T \hat{W}_1^T \sigma'(V_1^T r) V_1^T [\varepsilon_1 + d(t) + v]) \\ &+ R^T \hat{W}_1^T \sigma'(V_1^T r) V_1^T \tilde{W}_2^T \sigma(V_2^T s) + R^T \dot{\rho} + tr(\tilde{W}_2^T \Gamma_2 \dot{\tilde{W}}_2) \end{aligned}. \tag{27}$$

Using $\dot{\tilde{W}}_2 = -\dot{\hat{W}}_2$ and the property $tr(ab) = tr(ba)$,

$$\begin{aligned} \dot{L} = &-r^T K_v r + [r^T + R^T \hat{W}_1^T \sigma'(V_1^T r) V_1^T][\varepsilon_1 + d(t) + v] \\ &+ tr[\tilde{W}_2^T \sigma(V_2^T s) r^T] + tr(\dot{\hat{W}}_1^T \sigma(V_1^T r) R^T) \\ &+ R^T \hat{W}_1^T \sigma'(V_1^T r) V_1^T K_v r \\ &+ tr[\tilde{W}_2^T \sigma(V_2^T s) R^T \hat{W}_1^T \sigma'(V_1^T r)) V_1^T] \\ &+ R^T \dot{\rho} - tr(\tilde{W}_2^T \Gamma_2 \dot{\hat{W}}_2) \end{aligned}. \tag{28}$$

Simplifying the above function

$$\begin{aligned} \dot{L} = &-r^T K_v r + [r^T + R^T W_1^T \sigma'(V_1^T r) V_1^T][\varepsilon_1 + d(t) + v] \\ &+ tr\{\tilde{W}_2^T [\sigma(V_2^T s) r^T + \sigma(V_2^T s) R^T W_1^T \sigma'(V_1^T r) V_1^T - \Gamma_2 \dot{\hat{W}}_2]. \\ &+ tr\{\dot{W}_1^T \sigma(V_1^T r) R^T\} + R^T(\dot{\rho} + W_1^T \sigma'(V_1^T r) V_1^T K_v r) \end{aligned} \tag{29}$$

Substituting (15) (16) and (19) into (29),

$$\begin{aligned} \dot{L} = &-r^T K_v r - \Gamma_1 \|\sigma(V_1^T r) R^T\| \\ &+ [r^T + R^T W_1^T \sigma'(V_1^T r) V_1^T][\varepsilon_1 + d(t) + v] \end{aligned}. \tag{30}$$

Substituting VSC item (21) into (30),

$$\dot{L} < -r^T K_v r - \Gamma_1 \|\sigma(V_1^T r) R^T\| < 0. \tag{31}$$

According to the *Lemma 2*, the system employing the control law (20) is asymptotically stable. It implies that r, R and $\tilde{W}_2$ converge to 0 asymptotically.

VII. Numerical Study

We conducted empirical studies to evaluate the proposed adaptive controller based on the combination of feedback linearization, ACD, and VSC applied to a 2-link manipulator. The results are summarized in this section.

A. The 2-link manipulator Problem

The model of the 2-link manipulator used in this simulation is the same as the one in [8].

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x) + u + d(t)\end{aligned}, \qquad (32)$$

where $x_1 = [q_1 \quad q_2]^T$, $x_2 = [\dot{q}_1 \quad \dot{q}_2]^T$,the control input is $u = M^{-1}(q)\tau$ and The disturbance is $d(t) = M^{-1}(q)\tau_d$, τ_d is the external disturbance. The nonlinear function $f(x)$ is give by

$$f(x) = -M^{-1}(q)[C(q,\dot{q}) + F(q,\dot{q}) + G(q)] . \qquad (33)$$

The parameters in the 2-link manipulator model are given by

$$M(q) = \begin{bmatrix} (m_1+m_2)a_1^2 + m_2a_2^2 + 2m_2a_1a_2\cos(q_2) & m_2a_2^2 + m_2a_1a_2\cos(q_2) \\ m_2a_2^2 + m_2a_1a_2\cos(q_2) & m_2a_2^2 \end{bmatrix}, \qquad (34)$$

$$C(q,\dot{q}) = \begin{bmatrix} -m_2a_1a_2(2\dot{q}_1\dot{q}_2 + \dot{q}_2^2)\sin(q_2) \\ m_2a_1a_2\dot{q}_1^2\sin(q_2) \end{bmatrix}, \qquad (35)$$

$$G(q) = \begin{bmatrix} (m_1+m_2)ga_1\cos q_1 + m_2ga_2\cos(q_1+q_2) \\ m_2ga_2\cos(q_1+q_2) \end{bmatrix}, \qquad (36)$$

where $a_1 = a_2 = 1$ m are the manipulator arm lengths, $g = 9.8$ m/s^2 is the acceleration due to gravity, $m_1 = m_2 = 1$ kg are the joint masses.

B. Simulation Setup

The simulation is conducted in Matlab. In our simulation, the nominal plant model parameters are set to be $a_1 = a_2 = 1.4$ m and $m_1 = m_2 = 2$ kg . The reference command is set to be a constant vector $x_{1c} = [0.3 \quad 0.3]^T$ (rad). The noise of the plant is set as one combining a constant vector and a white noise, as

$$\tau_d(t) = \begin{bmatrix} 0.01 \\ 0.01 \end{bmatrix} + d_n(t) , \qquad (37)$$

where $d_n(t)$ is the white noise with the variance 0.1.

The parameters of the controller is summarized as follows

$$K_v = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}, \lambda_1 = \begin{bmatrix} 0.295 & 0 \\ 0 & 0.28 \end{bmatrix}, \qquad (38)$$

$$v = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \Gamma_1 = 0.09,\ \Gamma_2 = 0.06 . \qquad (39)$$

C. Simulation Results

The simulation results are presented in Fig. 2-Fig. 5. In these figures, the complete controller follows the proposed controller design in Equation (20) (Method 1); the controller without model compensation means that the nominal model is not available (Method 2). In this situation, we set the nominal model as $\hat{f}(x) = O$, $\hat{g}(x) = I$, where O is zero matrix and I is identity matrix; the controller without VSC compensation means that VSC item v in Equation (20) is O (Method 3).

Fig. 2 shows the time response of the complete controller when the number of the hidden nodes in neural networks is 20. Fig. 3 shows the time response of the controller without model compensation. Comparing the two figures, we observe that, although both controllers make the tracking errors converge to 0, the complete controller has faster response speed than the controller without model compensation.

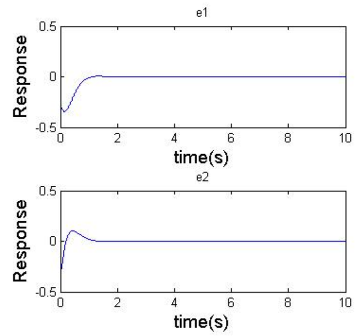

Fig. 2 The time response of the complete controller ($N_h = 20$)

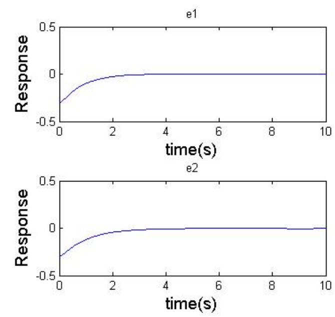

Fig. 3 The time response of the controller without model compensation ($N_h = 20$)

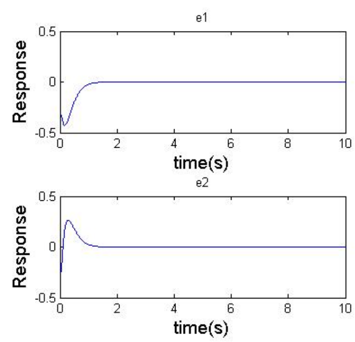

Fig. 4 The time response of the controller without VSC compensation ($N_h = 20$)

Fig. 4 and Fig. 5 show the time response of the controller without the VSC compensation when the number of the hidden nodes in the neural networks is 20 and 2 respectively. As we know in Section III, the more the hidden nodes in the neural networks, the smaller the reconstruction error. Comparing the two figures, we get that, when the reconstruction error is small, the controller performs well without the VSC compensation. However, if the reconstruction error is big, the controller without the VSC compensation is unable to make the tracking error converge to 0. Fig. 6 shows the time response of the controller when the number of the hidden nodes is 2. It implies that, although the reconstruction error in the neural networks is big, the controller with the VSC compensation can deal with large reconstruction error and make the tracking error converge to 0. That is to say, the system using the complete controller is also asymptotically stable when the number of the hidden nodes is small.

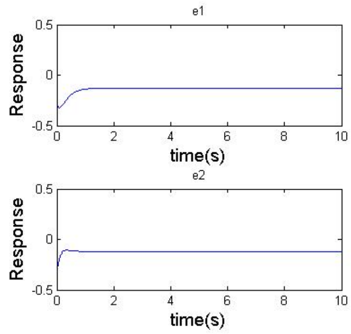

Fig. 5 The time response of the controller without the VSC compensation ($N_h = 2$)

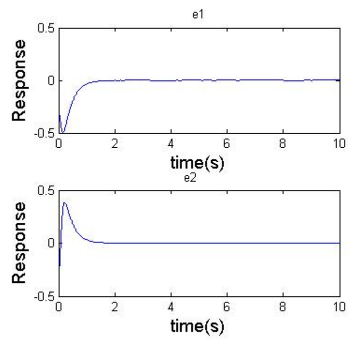

Fig. 6 The time response of the complete controller ($N_h = 2$)

Using the aggregate of squared errors between the reference and the actual response on each output over the duration of the experiment, we can compare the aggregate performance of these controllers. The smaller the aggregate error, the better the control performance. Table 1 summarizes the results.

Table 1. The aggregate errors with different controllers and different number of the hidden nodes

Method	1(N_h=20)	2(N_h=20)	3(N_h=20)	3(N_h=2)	1(N_h=2)
e_1	0.0035	0.0124	0.0244	0.1465	0.0036
e_2	0.0045	0.0166	0.0378	0.1876	0.0049

VIII. CONCLUSION

In this paper, we discuss the controller design using Adaptive Critic Design (ACD) for uncertain nonlinear systems. Existing ACD-based controllers were proven to be uniformly ultimately bounded. Although this is useful in the bounded error control, it is not enough for many precise tracking applications where asymptotical stability is preferred. To solve this problem, we developed a novel asymptotically stable ACD-based controller.

In our controller design, the feedback linearization is conducted based on the known nominal plant model to improve the system dynamic characteristics firstly; then we employ ACD to approximate unknown model uncertainties in order to make the controller robust; and lastly variable structure control item is employed to make the controller asymptotically stable. To evaluate the validity of the proposed controller, an empirical study is conducted on a 2-link manipulator. Results show the proposed controller can ensure good dynamic response characteristics and asymptotical stability under large model uncertainties.

REFERENCE

[1] R. Enns and J. Si, "Helicopter trimming and tracking control using direct neural dynamic programming," *IEEE Transactions on Parallel and Distributed Systems*, vol. 14, no. 4, pp. 929-939, 2003.

[2] G.K. Venayagamoorthy, R.G. Harley, and D.C. Wunsch, "Comparison of heuristic dynamic programming and dual heuristic programming adaptive critics for neurocontrol of a turbogenerator," *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 764-773, 2002.

[3] T. Hanselmann, L. Noakes, and A. Zaknich, "Continuous-Time Adaptive Critics," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 631-647, 2007.

[4] J. Si, A.G. Barto, W.B. Powell, and D.W. II, *Handbook of learning and approximate dynamic programming*: Wiley & Sons, Inc. Publications 2004.

[5] D.V. Prokhorov and D.C. Wunsch, II, "Adaptive critic designs," *IEEE Transactions on Parallel and Distributed Systems*, vol. 8, no. 5, pp. 997-1007, 1997.

[6] D.V. Prokhorov and D.C. Wunsch. "Convergence of critic-based training," in *proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, pp. 3057-3060 vol.4, 1997.

[7] L. Xin, L. Xin, and S.N. Balakrishnan. "Convergence analysis of adaptive critic based optimal control," in *proceedings of the American Control Conference, 2000. Proceedings of the 2000*, vol. 3, pp. 1929-1933 vol.3, 2000.

[8] O. Kuljaca and F.L. Lewis, "Adaptive critic design using non-linear network structures," *International Journal of Adaptive Control and Signal Processing*, vol. 17, no. 6, pp. 431-445, 2003.

[9] F.L. Lewis, J. Campos, and R. Selmic. "On adaptive critic architectures in feedback control," in *proceedings of the the 38th IEEE Conference on Decision and Control*, vol. 2, pp. 1677-1684, 1999.

[10] H. Pingan and S. Jagannathan, "Reinforcement Learning Neural-Network-Based Controller for Nonlinear Discrete-Time Systems With Input Constraints," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, vol. 37, no. 2, pp. 425-436, 2007.

[11] H. Pingan and S. Jagannathan. "Adaptive critic neural network-based controller for nonlinear systems with input constraints," in *proceedings of the Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 6, pp. 5709-5714 Vol.6, 2003.

[12] J.-J.E. Slotine and W. Li, *Applied Nonlinear Control*. NJ: Prentice Hall 1991.

[13] S. Lin and G. Manimaran. "Double-Loop Feedback-Based Scheduling Approach for Distributed Real-Time Systems," in *proceedings of the High Performance Computing (HiPC)*, pp. 268-278, 2003.

[14] P. Jung-Wook, R.G. Harley, and G.K. Venayagamoorthy, "Adaptive-critic-based optimal neurocontrol for synchronous generators in a power system using MLP/RBF neural networks," *Industry Applications, IEEE Transactions on*, vol. 39, no. 5, pp. 1529-1540, 2003.

[15] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems (MCSS)*, vol. 2, no. 4, pp. 303-314, 1989.

[16] J. Park and I.W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, no. 2, pp. 246-257, 1991.

[17] F.L.Lewis, C.T.Abdallah, and D.M.Dawson, *Control of Robot Manipulators*. New York: Macmillan 1993.

[18] B. Igelnik, B. Igelnik, and P. Yoh-Han, "Stochastic choice of basis functions in adaptive function approximation and the functional-link net," *IEEE Transactions on Neural Networks*, vol. 6, no. 6, pp. 1320-1329, 1995.