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

Neural Network Control of a Class of Nonlinear Discrete Time Systems with Asymptotic Stability Guarantees[1]

Balaje T. Thumati and S. Jagannathan

Abstract— In this paper, a single and multi-layer neural network (NN) controllers are developed for a class of nonlinear discrete time systems. Under a mild assumption on the system uncertainties, which include unmodeled dynamics and bounded disturbances, by using novel weight update laws and a robust term, local asymptotic stability of the closed-loop system is guaranteed in contrast with all other NN controllers where a uniform ultimate boundedness is normally shown. Simulation results are presented to show the effectiveness of the controller design.

I. Introduction

Signifcant research has been performed in the past decade in the area of neural network (NN) control for nonlinear system. The NNs became popular [1] due to their function approximation capabilities, which are utilized to learn uncertainties. Due to a functional reconstruction errors with a NN [2], typically the controller designs render a uniformly ultimately boundedness result since this reconstruction error is assumed to be upper bounded by a known constant [2-4].

The NN controller designs were first introduced for continuous-time systems [2-3] and later extended to control nonlinear discrete-time systems [4-5]. Development of stable controllers for discrete-time systems is rather difficult since the first difference of a Lyapunov function candidate is not linear with respect to its states in contrast to a first derivative of a Lyapunov candidate for continuous-time systems. All these controllers relax the persistence of excitation condition on the input signals. The NN controller designs were then extended to a more general class of nonlinear systems with state and output feedback [2] and for nonlinear discrete-time systems [5].

Recently, a robust integral of the sign of the error (RISE) feedback method is used in conjunction with a NN to show semi global asymptotic tracking of continuous-time nonlinear systems [6]. To ensure asymptotic performance of the NN controller, an attempt has been made in [7] for a class of continuous and discrete-time nonlinear systems by using a sector boundedness assumption on the uncertainties [7]. A single layer NN is utilized in the controller design.

By contrast, in this paper, however, we develop a suite of NN controllers for a class of nonlinear discrete-time systems that guarantee local asymptotic stability under a mild assumption on the uncertainties [6-9]. Note, unlike other NN controller designs [1, 4-5], the proposed design is guaranteed to render local asymptotic stability. The proposed controllers utilize the filter tracking error notion and a robust term. This new robust term is a function of the NN weights. Initially, a linearly parameterized NN is utilized in the controller design and later extended to multilayer NNs. The stability is shown using the Lyapunov theory. Finally, a simulation example is utilized to illustrate the performance of the proposed NN controllers.

II. Background

A. Neural Networks

A general nonlinear continuous function $f(x) \in C^k(s)$ which maps $f: S \to \Re^k$, where S is a simply-connected set of $\Re^n$ and $C^k(s)$ is the space where f is continuous can be written as

$$f(x) = w^T \sigma(V^T x) + \varepsilon_1(k) \tag{1}$$

where V and W represent input-to the hidden layer and hidden-to-the output layer weights respectively and $\varepsilon_1(k)$ is a neural net functional reconstruction error vector such that $\|\varepsilon_1\| \le \varepsilon_{1_N}$ for all $x \in \Re^n$. Additionally the activation functions $\sigma(.)$ are bounded, measurable, non-decreasing functions from the real numbers onto [0, 1] which include for instance sigmoid etc. We define the output of a NN as

$$\hat{y}(k) = \hat{w}^T(k)\varphi(x(k)) \tag{2}$$

where $\hat{w}(k)$ is the actual weight matrix and $\varphi(x(k))$ is the activation function which is selected as a basis function [2] in order to guarantee the function approximation. Apart from the single layer, a given continuous function $f(.)$, could be written using a three layer NN as [3]

$$f(x(k)) = w_3^T \varphi_3(w_2^T \varphi_2(w_1^T \varphi_1(x(k)))) + \varepsilon_1(k) \tag{3}$$

where w_1, w_2, and w_3 are the ideal weights. Additionally, the ideal weights are considered to be bounded $\|W_1\| \le W_{1\max}$, $\|W_2\| \le W_{2\max}$ and $\|W_3\| \le W_{3\max}$ and $\varphi_1(.)$, $\varphi_2(.)$ and $\varphi_3(.)$ are the activation functions of the first, second and third layer of the NN respectively. Next we define the output of a three-layer NN as

---

[1] Balaje T. Thumati (e-mail: bttr74@mst.edu) and S. Jagannathan (e-mail: sarangap@mst.edu) are with the Department of Electrical and Computer Engineering, Missouri University of Science and Technology (formerly University of Missouri–Rolla), Rolla, MO 65409, USA. Research supported in part by NSF I/UCRC on Intelligent Maintenance Systems Award and Intelligent Systems Center.

$$\hat{y}(k) = \hat{W}_3^T(k)\hat{\varphi}_3(\hat{W}_2^T(k)\hat{\varphi}_2(\hat{W}_1^T(k)\hat{\varphi}_1(x(k)))) \tag{4}$$

where $\hat{W}_3(k), \hat{W}_2(k), \hat{W}_1(k)$ are the actual NN weights of the third, second and first layer respectively and $\hat{\varphi}_1(x(k))$ represent the activation function vector of the input layer. Then $\hat{\varphi}_2(\hat{W}_1^T(k)\hat{\varphi}_1(x(k)))$, $\hat{\varphi}_3(\hat{W}_2^T(k)\hat{\varphi}_2(\hat{W}_1^T(k)\hat{\varphi}_1(x(k))))$ denote the hidden layer and output layers activation function respectively at the k^{th} instant. For a multilayer function approximation, the activation function vector need not be a basis function [2, 6]. Next the class of nonlinear discrete-time system to be considered in this paper is introduced.

B. Dynamics of the mnth-Order MIMO System

Consider a *mn*th-order multi-input-multi-output (MIMO) discrete time nonlinear system given by

$$\begin{aligned} x_1(k+1) &= x_2(k) \\ &\vdots \\ x_{n-1}(k+1) &= x_n(k) \\ x_n(k+1) &= f(x(k)) + u(k) + d(k) \end{aligned} \tag{5}$$

where $x(k) = [x_1(k), ..., x_n(k)]^T$ with $x_i(k) \in \Re^m, i = 1, ..., n$ $u(k) \in \Re^m$ is the input vector, and $d(k)$ denotes a disturbance vector at k^{th} instant with $\|d(k)\| \le d_M$ a known constant. Given a desired trajectory $x_{nd}(k)$ and its delayed values, we define the tracking error as

$$e_n(k) = x_n(k) - x_{nd}(k)$$

Define the filtered tracking error $r(k) \in \Re^m$ [1], as

$$r(k) = e_n(k) + \lambda_{c_1} e_{n-1}(k) + ... + \lambda_{c_{n-1}} e_1(k) \tag{6}$$

where $e_{n-1}(k), ..., e_1(k)$ are the delayed values of the error $e_n(k)$, and $\lambda_{c_1}, ..., \lambda_{c_{n-1}}$ are constant matrices selected so that $\left|z^{n-1} + \lambda_{c_1} z^{n-2} + ... + \lambda_{c_{n-1}}\right|$ is within a unit disc. Subsequently, (6) can be written as

$$r(k+1) = e_n(k+1) + \lambda_{c_1} e_{n-1}(k+1) + ... + \lambda_{c_{n-1}} e_1(k+1) \tag{7}$$

By substituting (5) in (7), we get

$$r(k+1) = f(x(k)) - x_{nd}(k+1) + \lambda_{c_1} e_n(k) + ... + \lambda_{c_{n-1}} e_2(k) + u(k) + d(k) \tag{8}$$

Now define the control input $u(k)$ as

$$u(k) = x_{nd}(k+1) - \hat{f}(x(k)) + k_v r(k) - v(k) - \lambda_{c_1} e_n(k) - .... - \lambda_{c_{n-1}} e_2(k) \tag{9}$$

where k_v is a user selectable diagonal matrix, $v(k)$ is the robust term vector which is defined later and $\hat{f}(x(k))$ is an estimate of $f(x(k))$. The closed-loop error system becomes

$$r(k+1) = k_v r(k) + \tilde{f}(x(k)) + v(k) + d(k) \tag{10}$$

where $\tilde{f}(x(k)) = f(x(k)) - \hat{f}(x(k))$ is the functional estimation error. In the next section, we propose the NN weight update law and the robust term. Additionally, the stability of the proposed NN controller will be demonstrated.

III. NN Controller Design

In this section, we propose a single and a three-layer NN based controllers by using a novel weight update law by relaxing the persistency of excitation condition and certainty equivalence principle [2]. Initially, we consider a single layer NN then a multilayer NN.

A. Single layer network

By considering constant ideal weights W, the nonlinear function in (5) could be written as

$$f(x) = W^T \varphi(x(k)) + \varepsilon_1(k)$$

where the target weight matrix is assumed bounded such that $\|W\| \le W_{\max}$. Define the NN functional estimate as

$$\hat{f}(x) = \hat{W}^T(k)\varphi(x(k)) \tag{11}$$

and the weight estimation error as

$$\tilde{W}(k) = W - \hat{W}(k) \tag{12}$$

Thus the control input (9) is given by

$$u(k) = x_{nd}(k+1) - \hat{W}^T(k)\varphi(x(k)) + k_v r(k) - v(k) - \lambda_{c_1} e_n(k) - ... - \lambda_{c_{n-1}} e_2(k)$$

Substituting the above equation in (8) results in the following closed-loop filtered error dynamics as

$$r(k+1) = k_v r(k) - v(k) + \Psi_1(k) + d(k) + \varepsilon_1(k) \tag{13}$$

where $\Psi_1(k) = \tilde{W}^T(k)\varphi(x(k))$. Define the robust term as $v(k) = \dfrac{\hat{W}^T(k)B_1}{B_1^T\hat{W}(k)\hat{W}^T(k)B_1 + c_c}$, where $B_1 \in \Re^{l\times 1}$ is a constant vector and $c_c > 0$ is a constant. The purpose of the robust term is to improve the stability of the controller as explained later in this paper. Substitution of the robust term into (13) renders

$$r(k+1) = k_v r(k) - \frac{\hat{W}^T(k)B_1}{B_1^T\hat{W}(k)\hat{W}^T(k)B_1 + c_c} + \Psi_1(k) + \varepsilon(k) \tag{14}$$

where $\varepsilon(k) = d(k) + \varepsilon_1(k)$. Adding and subtracting $\dfrac{W^T B_1 - C_1}{B_1^T\hat{W}(k)\hat{W}^T(k)B_1 + c_c}$ in (14), where $C_1 \in \Re^{n\times 1}$ is a constant vector. The filtered tracking error dynamics becomes

$$r(k+1) = k_v r(k) + \Psi_2(k) + \Psi_1(k) + \varepsilon(k) - \frac{\left(W^T B_1 - C_1\right)}{B_1^T\hat{W}(k)\hat{W}^T(k)B_1 + c_c} \tag{15}$$

where $\Psi_2(k) = \dfrac{\left(\tilde{W}^T(k)B_1 - C_1\right)}{B_1^T\hat{W}(k)\hat{W}^T(k)B_1 + c_c}$. Next, the following lemma on the modeling uncertainty and bounded disturbances is introduced before proceeding further.

Lemma 1: The term (ε) comprising of the approximation errors , ε_1, and bounded disturbance, $d(k)$ is assumed to be upper bounded by a smooth nonlinear function of filter tracking error and the NN weights [6-9] as

$$(4+5\alpha\varphi^T(k)\varphi(k))\varepsilon^T(k)\varepsilon(k) \le d_0 + d_1\|r(k)\|^2 + d_2\|\tilde{w}(k)\|^2 + d_3\|r(k)\|\|\tilde{w}(k)\| \qquad (16)$$

where d_0, d_1, d_2, and d_3 are computable positive constants.

Proof: Using some standard norm inequalities, the fact that $\varphi(.)$ vector is bounded by constants for RBF, sigmoid, and tanh, it is easy to show that the reconstruction error is a function of the filtered tracking and weight estimation errors.

Remark 1: Similar relationship is stated by a number of researchers [6-9] in continuous and discrete-time. This assumption is mild in comparison with the assumption that the functional reconstruction error and disturbances are bounded above by a known constant.

Next in the theorem, it will be shown that the proposed control law renders an asymptotically stable system.

Theorem 3.1: Let $x_{nd}(k)$ be the desired trajectory, and the initial conditions be bounded in a compact set *S*. Consider bounded uncertainties and the control law (9) be applied to the system. Let the NN weight update law be provided by

$$\hat{w}(k+1) = \hat{w}(k) + \alpha\ \varphi(k)r^T(k+1) \qquad (17)$$

where $\alpha > 0$ is the learning rate. Then, the filter tracking error $r(k)$ and the NN weight estimation errors, $\tilde{w}(k)$ are locally asymptotically stable.

Proof: Consider a Lyapunov function candidate as

$$V = r^T(k)r(k) + \frac{1}{\alpha}tr[\tilde{w}^T(k)\tilde{w}(k)]$$

The first difference is given by

$$\Delta V = \underbrace{r^T(k+1)r(k+1) - r^T(k)r(k)}_{\Delta V_1} + \underbrace{\frac{1}{\alpha}tr[\tilde{w}^T(k+1)\tilde{w}(k+1) - \tilde{w}^T(k)\tilde{w}(k)]}_{\Delta V_2} \qquad (18)$$

Substitute the filter tracking error (15) in ΔV_1 of (18), and after performing some mathematical manipulations, we get

$$\Delta V_1 = r^T(k)k_v^Tk_vr(k) + 2r^T(k)k_v^T\Psi_1(k) + 2r^T(k)k_v^T\Psi_2(k) + 2r^T(k)k_v^T\varepsilon(k)$$
$$-2\frac{r^T(k)k_v^T(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c} + \Psi_1^T(k)\Psi_1(k) - 2\frac{\Psi_1^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}$$
$$+2\Psi_1^T(k)\Psi_2(k) + 2\Psi_1^T(k)\varepsilon(k) + \Psi_2^T(k)\Psi_2(k) - \frac{2\Psi_2^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}$$
$$-\frac{2\varepsilon^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c} + \frac{(w^TB_1 - C_1)^T(w^TB_1 - C_1)}{(B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c)^2} + 2\Psi_2^T(k)\varepsilon(k)$$
$$+\varepsilon^T(k)\varepsilon(k) - r^T(k)r(k) \qquad (19)$$

Next the weight update law (17) is substituted in the second term, ΔV_2 of (18), and using (12), to get

$$\Delta V_2 = \frac{1}{\alpha}tr[(\Delta\tilde{w}(k) + \tilde{w}(k))^T.(\Delta\tilde{w}(k) + \tilde{w}(k)) - \tilde{w}^T(k)\tilde{w}(k)]$$

$$\Delta V_2 = \frac{1}{\alpha}tr[\Delta\tilde{w}^T(k).\Delta\tilde{w}(k) + 2\Delta\tilde{w}^T(k).\tilde{w}(k)]$$
$$= \frac{1}{\alpha}tr[\alpha^2(r(k+1)\varphi^T)(\varphi r^T(k+1)) - 2\alpha r(k+1)\varphi^T\tilde{w}(k)]$$
$$= \frac{1}{\alpha}tr[\alpha^2(k_vr(k) + \Psi_1(k) + \Psi_2(k) + \varepsilon(k) - \frac{(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c})\varphi^T$$
$$\varphi(k_vr(k) + \Psi_1(k) + \Psi_2(k) + \varepsilon(k) - \frac{(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c})^T$$
$$-2\alpha\tilde{w}^T(k)\varphi(k)(k_vr(k) + \Psi_1(k) + \Psi_2(k) + \varepsilon(k) - \frac{(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c})]$$

Applying Cauchy-Schwarz inequality $((a_1 + a_2 + ... + a_n)^T.(a_1 + a_2 + ... + a_n) \le n.(a_1^Ta_1 + a_2^Ta_2 + ... + a_n^Ta_n))$ for the first term in the above equation, and applying the trace operator (given a vector $x \in \Re^n$, $tr(xx^T) = x^Tx$) to obtain

$$\le 5\alpha\varphi^T\varphi(r^T(k)k_v^Tk_vr(k) + \Psi_1^T(k)\Psi_1(k) + \Psi_2^T(k)\Psi_2(k) + \varepsilon^T(k)\varepsilon(k))$$
$$+5\alpha\varphi^T\varphi\frac{(w^TB_1 - C_1)^T(w^TB_1 - C_1)}{(B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c)^2}$$
$$-2\Psi_1^T(k)(k_ve(k) + \Psi_1(k) + \Psi_2(k) + \varepsilon(k) - \frac{(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}) \qquad (20)$$

Next, the overall first difference of the Lyapunov function candidate, $\Delta V = \Delta V_1 + \Delta V_2$, can be obtained from (19) and (20) as

$$\Delta V \le r^T(k)k_v^Tk_vr(k) + 2r^T(k)k_v^T\Psi_1(k) + 2r^T(k)k_v^T\Psi_2(k) + 2r^T(k)k_v^T\varepsilon(k)$$
$$-2\frac{r^T(k)k_v^T(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c} + \Psi_1^T(k)\Psi_1(k) - 2\frac{\Psi_1^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}$$
$$+2\Psi_1^T(k)\Psi_2(k) + 2\Psi_1^T(k)\varepsilon(k) + \Psi_2^T(k)\Psi_2(k) - \frac{2\Psi_2^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}$$
$$-\frac{2\varepsilon^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c} + \frac{(w^TB_1 - C_1)^T(w^TB_1 - C_1)}{(B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c)^2} + 2\Psi_2^T(k)\varepsilon(k)$$
$$+\varepsilon^T(k)\varepsilon(k) - r^T(k)r(k)$$
$$+5\alpha\varphi^T\varphi r^T(k)k_v^Tk_vr(k) + 5\alpha\varphi^T\varphi\Psi_1^T(k)\Psi_1(k) + 5\alpha\varphi^T\varphi\Psi_2^T(k)\Psi_2(k)$$
$$+5\alpha\varphi^T\varphi\frac{(w^TB_1 - C_1)^T(w^TB_1 - C_1)}{(B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c)^2} + 5\alpha\varphi^T\varphi\varepsilon^T(k)\varepsilon(k)$$
$$-2\Psi_1^T(k)k_vr(k) - 2\Psi_1^T(k)\Psi_1(k) - 2\Psi_1^T(k)\Psi_2(k)$$
$$-2\Psi_1^T(k)\varepsilon(k) - \frac{2\Psi_1^T(k)(w^TB_1 - C_1)}{B_1^T\hat{w}(k)\hat{w}^T(k)B_1 + c_c}$$

After some mathematical manipulation and using Lemma 1, the first difference is rewritten as

$$\Delta V \le 4r^T(k)k_v^T k_v r(k) - r^T(k)r(k) - \Psi_1^T(k)\Psi_1(k)$$
$$+5\alpha\varphi^T\varphi r^T(k)k_v^T k_v r(k) + 5\alpha\varphi^T\varphi\Psi_1^T(k)\Psi_1(k)$$
$$+d_0 + d_1\|r(k)\|^2 + d_2\|\tilde{w}(k)\|^2$$
$$+d_4\|r(k)\|^2 + d_4\|\tilde{w}(k)\|^2 + (4+5\alpha\varphi^T\varphi)(B_1^T\tilde{w}(k)\tilde{w}^T(k)B_1 + C_1^T C_1)$$
$$+(4+5\alpha\varphi^T\varphi)(B_1^T ww^T B_1 - 2B_1^T wC_1 + C_1^T C_1)$$

Taking $d_4 = d_3/2$, and $C_{1\min} = \dfrac{(d_0/(4+5\alpha\varphi_{\max}^2)) + B_{1\max}^2 w_{\max}^2 + 2C_{1\max}^2}{2w_{\min}B_{1\min}}$, applying the Frobenius norm, the first difference can be expressed in compact form as

$$\Delta V \le -(1 - 4k_{v\max}^2 - 5\alpha\varphi_{\max}^2 k_{v\max}^2 - d_1 - d_4)\|r(k)\|^2$$
$$-(\varphi_{\min}^2 - 5\alpha\varphi_{\max}^4 - 4B_{1\max}^2 - 5\alpha\varphi_{\max}^2 B_{1\max}^2 - d_2 - d_4)\|\tilde{w}(k)\|^2 \quad (21)$$

Hence $\Delta V < 0$ provided the gains are taken as

$$k_{v\max} \le \sqrt{\frac{1-d_1-d_4}{(4+5\alpha\varphi_{\max}^2)}},\ B_{1\max}^2 = \frac{\alpha}{(4+5\alpha\varphi_{\max}^2)},$$

$\varphi_{\min} \ge \sqrt{5\alpha\varphi_{\max}^4 + \alpha + d_2 + d_4}$, and $0 < \alpha << 1$. □

As long as the gains are selected above, $\Delta V < 0$ in (21), which shows stability in the sense of Lyapunov. Hence $r(k)$ and $\tilde{w}(k)$ are bounded, provided if $r(k_0)$ and $\tilde{w}(k_0)$ are bounded in the compact set *S*. Summing both sides of (21) to show that both $r(k)$ and converges $\tilde{w}(k)$ approaches to zero asymptotically.

Next, we extend the above results for a three-layer NN controller.

B. Three-Layer NN controller

Here consider a three-layer NN, by using (4), the NN output of a nonlinear function in (5) could be written as

$$\hat{f}(x) = \hat{w}_3^T(k)\hat{\varphi}_3(\hat{w}_2^T(k)\hat{\varphi}_2(\hat{w}_1^T(k)\hat{\varphi}_1(x(k)))) \quad (22)$$

Define the weight estimation errors as

$\tilde{W}_1(k) = W_1 - \hat{W}_1(k)$, $\tilde{W}_2(k) = W_2 - \hat{W}_2(k)$ and $\tilde{W}_3(k) = W_3 - \hat{W}_3(k)$.

Next the following fact can be stated.

Fact 3.1.1: The activation functions are bounded by known positive values so that $\|\hat{\varphi}_1(k)\| \le \varphi_{1\max}$, $\|\hat{\varphi}_2(k)\| \le \varphi_{2\max}$ and $\|\hat{\varphi}_3(k)\| \le \varphi_{3\max}$. Define activation function vector error as

$\tilde{\varphi}_1(k) = \varphi_1 - \hat{\varphi}_1(k)$, $\tilde{\varphi}_2(k) = \varphi_2 - \hat{\varphi}_2(k)$ and $\tilde{\varphi}_3(k) = \varphi_3 - \hat{\varphi}_3(k)$.

Thus by using (22) in the control input (9), we get

$$u(k) = x_{nd}(k+1) - \hat{W}_3^T(k)\hat{\varphi}_3(\hat{W}_2^T(k)\hat{\varphi}_2(\hat{W}_1^T(k)\hat{\varphi}_1(x(k)))) + k_v r(k) - v(k)$$
$$-\lambda_{c_1}e_n(k) - \dots\dots\dots - \lambda_{c_{n-1}}e_2(k) \quad (23)$$

Then, the closed-loop filtered error dynamics become

$$r(k+1) = k_v r(k) - v(k) + \Psi_1(k) + d(k) + \varepsilon_1(k) + W_3^T\tilde{\varphi}_3(x(k)) \quad (24)$$

where $\Psi_1(k) = \tilde{W}_3^T(k)\hat{\varphi}_3(x(k))$, and the robust term for this control design is given by $v(k) = \dfrac{\hat{W}_3^T(k)B_v}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v + c_m}$, where $c_m > 0$ is a constant and B_v is an appropriate dimensioned constant vector, to be defined later. Hence (24) would be modified to

$$r(k+1) = k_v r(k) - \frac{\hat{W}_3^T(k)B_v}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v + c_m} + \Psi_1(k) + d(k) + \varepsilon_1(k) + W_3^T\tilde{\varphi}_3(x(k))$$

Next by adding and subtracting $\dfrac{W_3^T B_v - C_v}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v + c_m}$ in the above equation, where C_v is an appropriate dimensioned constant vector, to get

$$r(k+1) = k_v r(k) + \Psi_2(k) + \Psi_1(k) + \varepsilon(k) - \frac{(W_3^T B_v - C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v + c_m} \quad (25)$$

where

$$\Psi_2(k) = \frac{\tilde{W}_3^T(k)B_v - C_v}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v + c_m},\ \varepsilon(k) = d(k) + \varepsilon_1(k) + W_3^T\tilde{\phi}_3(x(k)).$$

The following theorem guarantees asymptotic stability of the closed-loop system using the proposed control law in (23).

Theorem 3.2: Let $x_{nd}(k)$ be the desired trajectory, and the initial conditions be bounded in a compact set *S*. Considering bounded uncertainties and the control law proposed in (23), where the three layer NN is tuned online using the following weight update laws for the input and hidden layers as

$$\hat{w}_1(k+1) = \hat{w}_1(k) - \alpha_1\ \hat{\varphi}_1(k)[\hat{y}_1(k) + B_1 k_v r(k)]^T \quad (26)$$

$$\hat{w}_2(k+1) = \hat{w}_2(k) - \alpha_2\ \hat{\varphi}_2(k)[\hat{y}_2(k) + B_2 k_v r(k)]^T \quad (27)$$

with $\hat{y}_i(k) = \hat{w}_i^T(k)\ \hat{\varphi}_i(k)$ and $\|B_i\| \le \kappa_i$, $i = 1, 2$. Take the weight update law for the output-layer as

$$\hat{w}_3(k+1) = \hat{w}_3(k) + \alpha_3\ \hat{\varphi}_3(k)r^T(k+1) \quad (28)$$

where $\alpha_i > 0, \forall i = 1, 2, 3$, denotes the learning rate or adaptation gains. Then, the filter tracking error $r(k)$ is locally asymptotically stable, while the NN weight estimation errors $\tilde{w}_1(k)$, $\tilde{w}_2(k)$ and $\tilde{w}_3(k)$ are bounded.

Proof: Consider a Lyapunov candidate as

$$V = r^T(k)r(k) + \frac{1}{\alpha_1}tr[\tilde{w}_1^T(k)\tilde{w}_1(k)] + \frac{1}{\alpha_2}tr[\tilde{w}_2^T(k)\tilde{w}_2(k)] + \frac{1}{\alpha_3}tr[\tilde{w}_3^T(k)\tilde{w}_3(k)]$$

The first difference is given by

$$\Delta V = \underbrace{r^T(k+1)r(k+1) - r^T(k)r(k)}_{\Delta V_1} + \underbrace{\frac{1}{\alpha_1}tr[\tilde{w}_1^T(k+1)\tilde{w}_1(k+1) - \tilde{w}_1^T(k)\tilde{w}_1(k)]}_{\Delta V_2}$$

$$+\underbrace{\frac{1}{\alpha_2}tr[\tilde{w}_2^T(k+1)\tilde{w}_2(k+1) - \tilde{w}_2^T(k)\tilde{w}_2(k)]}_{\Delta V_3}$$

$$+\underbrace{\frac{1}{\alpha_3} tr[\tilde{w}_3^T(k+1)\tilde{w}_3(k+1)-\tilde{w}_3^T(k)\tilde{w}_3(k)]}_{\Delta V_4} \tag{29}$$

Substituting (25) to (28) in (29), collecting terms together and completing square yields

$$\begin{aligned}\Delta V \le\ & r^T(k)k_v^T k_v r(k)+2r^T(k)k_v^T\Psi_1(k)+2r^T(k)k_v^T\Psi_2(k)+2r^T(k)k_v^T\varepsilon(k)\\
&-2\frac{r^T(k)k_v^T(W_3^T B_v-C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m}+\Psi_1^T(k)\Psi_1(k)-2\frac{\Psi_1^T(k)(W_3^T B_v-C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m}\\
&+2\Psi_1^T(k)\Psi_2(k)+2\Psi_1^T(k)\varepsilon(k)+\Psi_2^T(k)\Psi_2(k)-\frac{2\Psi_2^T(k)(W_3^T B_v-C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m}\\
&-\frac{2\varepsilon^T(k)(W_3^T B_v-C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m}+\frac{(W_3^T B_v-C_v)^T(W_3^T B_v-C_v)}{\left(B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m\right)^2}+2\Psi_2^T(k)\varepsilon(k)\\
&+\varepsilon^T(k)\varepsilon(k)-r^T(k)r(k)\\
&-(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))\left\|\tilde{W}_1^T(k)\hat{\varphi}_1(k)-\frac{(1-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}{(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}\right.\\
&\left.\times(W_2^T\hat{\varphi}_2(k)+B_2k_v r(k))\right\|^2+\frac{W_{2\max}^2\left\|\hat{\varphi}_2(k)\right\|^2}{(2-\alpha_2\left\|\hat{\varphi}_2(k)\right\|^2)}\\
&+\frac{k_{v\max}^2\left\|r(k)\right\|^2\kappa_1^2}{(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}+\frac{2k_{v\max}\left\|r(k)\right\|\kappa_1\varphi_{1\max}W_{1\max}}{(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}\\
&-(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))\left\|\tilde{W}_2^T(k)\hat{\varphi}_2(k)-\frac{(1-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}{(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}\right.\\
&\left.\times(W_2^T\hat{\varphi}_2(k)+B_2k_v r(k))\right\|^2+\frac{W_{2\max}^2\left\|\hat{\varphi}_2(k)\right\|^2}{(2-\alpha_2\left\|\hat{\varphi}_2(k)\right\|^2)}\\
&+\frac{k_{v\max}^2\left\|r(k)\right\|^2\kappa_2^2}{(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}+\frac{2k_{v\max}\left\|r(k)\right\|\kappa_2\varphi_{2\max}W_{2\max}}{(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}\\
&-2r^T(k)k_v^T\Psi_1(k)-2\Psi_1^T(k)\Psi_1(k)-2\Psi_1^T(k)\Psi_2(k)-2\Psi_1^T(k)\varepsilon(k)\\
&+2\frac{\Psi_1^T(k)(W_3^T B_v-C_v)}{B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m}+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3 r^T(k)k_v^T k_v r(k)+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3\Psi_1^T(k)\Psi_1(k)\\
&+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3\Psi_2^T(k)\Psi_2(k)+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3\varepsilon^T(k)\varepsilon(k)\\
&+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3\frac{(W_3^T B_v-C_v)^T(W_3^T B_v-C_v)}{\left(B_v^T\hat{W}_3(k)\hat{W}_3^T(k)B_v+c_m\right)^2}\end{aligned} \tag{30}$$

Next, the following lemma is introduced.

Lemma 2: Using Lemma 1, the term (ε), and the ideal weights of the NN are assumed to be bounded above by a smooth nonlinear function of filter tracking error and the NN weights [6-9] as

$$\frac{\sum_{i=1}^{2} W_{i_{\max}}^2(\left\|\hat{\varphi}_i(k)\right\|+\kappa_i)\left\|\hat{\varphi}_i(k)\right\|}{(2-\alpha_i\left\|\hat{\varphi}_i(k)\right\|^2)^2}+(5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3+4)\varepsilon^T(k)\varepsilon(k)$$

$$\le \beta_0+\beta_1\left\|r(k)\right\|^2+\beta_2\left\|\tilde{w}_3(k)\right\|^2+\beta_3\left\|r(k)\right\|\left\|\tilde{w}_3(k)\right\| \tag{31}$$

where $\beta_0, \beta_1, \beta_2$, and β_3 are computable positive constants.

Proof: Similar to Lemma 1, using some standard norm inequalities, the fact that $\varphi_1(.), \varphi_2(.)$, and $\varphi_3(.)$ vectors are bounded by constants for RBF, sigmoid, and tanh; and the reconstruction error is a function of the filtered tracking error and the weight estimation errors.

Using Lemma 2, taking the Frobenious norm of (30) and taking $C_{v_{\min}}=\dfrac{B_{1_{\max}}^2 W_{3_{\max}}^2+3C_{1_{\max}}^2+(\beta_0/(4+5\alpha_3\hat{\varphi}_3^T\hat{\varphi}_3))}{2B_{1_{\min}}W_{3_{\min}}}$, we get

$$\begin{aligned}\Delta V \le\ & -(1-4k_{v_{\max}}^2-\gamma k_{v_{\max}}^2-\beta_1-\beta_4)\left\|r(k)\right\|^2\\
&-(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))\left\|\tilde{W}_1^T(k)\hat{\varphi}_1(k)-\frac{(1-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}{(2-\alpha_1\hat{\varphi}_1^T(k)\hat{\varphi}_1(k))}\times(W_1^T\hat{\varphi}_1(k)+B_1k_v r(k))\right\|^2\\
&-(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))\left\|\tilde{W}_2^T(k)\hat{\varphi}_2(k)-\frac{(1-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}{(2-\alpha_2\hat{\varphi}_2^T(k)\hat{\varphi}_2(k))}\times(W_2^T\hat{\varphi}_2(k)+B_2k_v r(k))\right\|^2\\
&-(1-5\alpha_3\left\|\hat{\varphi}_3(k)\right\|^2-\alpha_3-\beta_2-\beta_4)\left\|\hat{\varphi}_3(k)\right\|^2\left\|\tilde{w}_3(k)\right\|^2\end{aligned} \tag{32}$$

where $\beta_4=\beta_3/2$, and $\gamma=\dfrac{\sum_{i=1}^{2}\kappa_i\left\|\hat{\varphi}_i(k)\right\|+\kappa_i^2}{(2-\alpha_i\left\|\hat{\varphi}_i(k)\right\|^2)}+5\alpha_3\left\|\hat{\varphi}_3(k)\right\|^2$. Then $\Delta V\le 0$ in (32) provided the following gains are selected

$k_{v_{\max}}\le\sqrt{\dfrac{1-\beta_1-\beta_4}{4+\gamma}}$, $B_{1_{\max}}^2=\dfrac{\alpha_3\left\|\hat{\varphi}_3(k)\right\|^2}{(8+10\alpha_3\left\|\hat{\varphi}_3(k)\right\|^2)}$, $\alpha_3=\dfrac{1-\beta_2-\beta_4}{1+5\left\|\hat{\varphi}_3(k)\right\|^2}$, and $\beta_2+\beta_4\le 1$.

Then the first difference, $\Delta V\le 0$ in (32), which shows stability in the sense of Lyapunov provided the gains are selected above. Hence $r(k)$, $\tilde{w}_3(k)$, $\tilde{w}_2(k)$, and $\tilde{w}_1(k)$ are bounded, provided if $r(k_0)$, $\tilde{w}_3(k_0)$, $\tilde{w}_2(k_0)$, and $\tilde{w}_1(k_0)$ are bounded in the compact set *S*. Additionally by using [5], we could show that the tracking error $\left\|r(k)\right\|\to 0$ as $k\to\infty$. Hence $r(k)$ converges asymptotically.

In the next section, simulation results are introduced.

IV. SIMULATION RESULTS

Consider the following nonlinear discrete-time system [5]

$$\begin{aligned}X_1(k+1)&=\Delta t.(X_2(k))+X_1(k)\\
X_2(k+1)&=\Delta t.(F(X_1,X_2))+U+X_2(k)+D(k)\end{aligned} \tag{33}$$

where $X_1(k) = [x_1(k), x_2(k)]^T$, $X_2(k) = [x_3(k), x_4(k)]^T$, $U(k) = [u_1(k), u_2(k)]^T$ and the nonlinear function $F(X_1, X_2)$ is given by $F(X_1, X_2) = [M(X_1)]^{-1} G(X_1, X_2)$, where

$[M(X_1)] =$

$$\begin{bmatrix} (b_1 + b_2)a_1^2 + b_2 a_2^2 + 2b_2 a_1 a_2 \cos(x_2(k)) & b_2 a_2^2 + b_2 a_1 a_2 \cos(x_2(k)) \\ b_2 a_2^2 + b_2 a_1 a_2 \cos(x_2(k)) & b_2 a_2^2 \end{bmatrix}$$

and

$$G(X_1, X_2) = \begin{bmatrix} G_1(k) \\ b_2 a_1 a_2 x_1^2(k) \sin(x_2(k)) + 9.8 b_2 a_2 \cos(x_1(k) + x_2(k)) \end{bmatrix}$$ with

$$G_1(k) = -b_2 a_1 a_2 (2x_3(k)x_4(k) + x_4^2(k)) \sin(x_2(k)) + 9.8(b_1 + b_2)a_1 \cos(x_1(k))$$
$$+9.8 b_2 a_2 \cos(x_1(k) + x_2(k)) .$$

Also, $D(k)$ is the disturbance vector, which is given by $D(k) = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ for $0 < k\Delta t \le 20 \sec$, else $D(k) = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$. The system in (33) is sampled at $\Delta t = 10\text{msec}$. The desired trajectories are given as $\sin\left(\frac{2\pi k \Delta t}{25}\right)$ and $\cos\left(\frac{2\pi k \Delta t}{25}\right)$. Additionally, the parameters of the nonlinear system are taken as $a_1 = a_2 = 1$, $b_1 = 2$ and $b_2 = 1$. The initial conditions of the nonlinear system are chosen to be $X_1 = [1.5, 1]^T$ and $X_2 = [-1.5, -1]^T$. Also, the controller gains are chosen to be $k_v = diag\{0.2, 0.2\}$ and $\lambda_{c_1} = diag\{0.9, 0.9\}$.

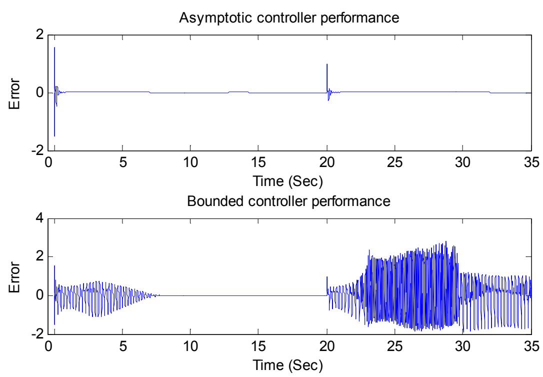

Figure 1: Tracking error for the reference trajectory 1 by the asymptotic and the bounded NN controllers.

The tracking errors for both the reference trajectories are shown in Figs. 1 and 2. The proposed asymptotic controller (NN term + robust term) is compared against another bounded NN controller presented in [4] without a robust term. From the Figs. 1 and 2, it is evident that the proposed asymptotic controller achieves a highly satisfactory tracking performance even in the presence of disturbance when compared to a NN controller that renders uniformly ultimately bounded result. On the other hand, a standard PD controller renders unstable results due to the size of the disturbance and therefore it is not suitable.

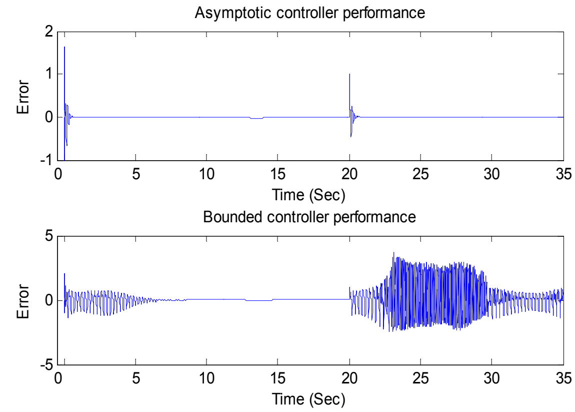

Figure 2: Tracking error for the reference trajectory 2 by the asymptotic and the bounded NN controllers.

V. Conclusions

In this paper, a suite of NN controllers were developed for nonlinear discrete time systems. By using a novel robust term and based on mild assumption on the NN approximation errors and in the presence of bounded disturbances, the asymptotic tracking is demonstrated through Lyapunov analysis. Simulation studies verify the theoretical conjectures.

References

[1] S. Jagannathan and F. L. Lewis, "Discrete-time neural net controller for a class of nonlinear dynamical systems", *IEEE Trans. on Automatic Control*, vol. 41, no. 11, pp. 1693–1699, 1996.

[2] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Trans. on Neural Networks*, vol. 7, pp. 387–398, Apr. 1996.

[3] S. S. Ge, C. C. Hang, and T. Zhang, "Adaptive neural network control of nonlinear systems by state and output feedback", *IEEE Trans. on System, Man, Cybernetics- Part B*, vol. 29, no. 6, pp. 818-828, 1999.

[4] S. Jagannathan and F. L. Lewis, "Multilayer discrete-time neural-net controller with guaranteed performance", *IEEE Trans. on Neural Networks*, vol. 7, no.1, pp. 107-130, 1996.

[5] S. Jagannathan, *Neural network control of nonlinear discrete-time systems*, CRC Press, April 2006.

[6] P. M. Patre, W. MacKunis, K. Kaiser, and W. E. Dixon, " Asymptotic tracking for uncertain dynamics systems via a multilayer NN feed forward and RISE feedback control structure", *Proc. of the American Control conference*, New York City, NY, USA, pp. 5989-5994, 2007.

[7] B. Xian, D. M. Dawson, M. S. de Queiroz, and J. Chen, " A continuous asymptotic tracking control strategy for uncertain nonlinear systems", *IEEE Trans. on Automatic Control*, vol. 49, no. 7, pp. 1206-1211, 2004.

[8] T. Hayakawa, W. M. Haddad, and N. Hovakimyan, "Neural network adaptive control for a class of nonlinear uncertain dynamical systems with asymptotic stability guarantees", *IEEE Trans. on Neural Networks*, vol. 19, no. 1, pp. 80-89, 2008.

[9] C. M. Kwan, D. M. Dawson, and F. L. Lewis," Robust adaptive control of robots using neural network: global tracking stability", *Proc. of the 34th conference on Decision and Control*, New Orleans, LA, pp. 1846-1851, 1995.