2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Adaptive Self-constructing Radial-Basis-Function Neural Control for MIMO Uncertain Nonlinear Systems with Unknown Disturbances

Ning Wang, Bijun Dai, Yancheng Liu and Min Han

***Abstract*— In this paper, an adaptive self-constructing RBF neural control (AS-RBFNC) scheme for trajectory tracking of MIMO uncertain nonlinear systems with unknown time-varying disturbances is proposed. System uncertainties and unknown dynamics can be exactly identified online by a self-constructing RBF neural network (SC-RBFNN) which is implemented by employing dynamically constructive hidden nodes according to the structure learning criteria including hidden node generating and pruning. The globally asymptotical stability of the entire AS-RBFNC control system is derived from Lyapunov approach.**

I. Introduction

TRADITIONAL nonlinear controllers are essentially model-based approaches which would inevitably rely on partially or fully known model dynamics since the equivalent control is directly derived from nonlinearity cancellation or dominance while an additional robustness term is designed to attenuate residual errors. In addition to uncertain dynamics, unknown external disturbances are actually of greater importance for tracking control of nonlinear systems.

Alternatively, the approximation-based approaches via fuzzy logic systems (FLS) [1]–[3], neural networks (NN) [4], [5] and fuzzy neural networks (FNN) [6]–[8], *etc.*, do not require parametric or functional certainty. Due to model uncertainties and unknown disturbances imposed on nonlinear systems, the approximation-based control methods are highly desired to realize online adaptation and robustness to unknown dynamics. In spite of the various achievements, the learning ability of the aforementioned adaptive approximation-based control schemes with only output weights being updated is actually limited by the predefined regressors since the weight adaptation is merely required to guarantee stability of closed-loop system rather than uncertainty identification. In this context, the FNN can enhance the learning capability of FIS by incorporating the NN topology [6]–[8], which allows all free parameters to be adaptively updated according to performance criteria [9]–[12]. Note that adaptive laws only consider parameter learning without structure update, i.e., the number of fuzzy rules or hidden nodes must be determined *a priori*, although the resulting performance is acceptable due to the truth that convergence of the tracking error does not necessarily imply convergence (or even robustness) of the estimated parameters [13]. It implies that the approximation accuracy would be much poorer if inadequate fuzzy rules, i.e. too many or too few, are predefined.

N. Wang and Y. C. Liu are with the Marine Engineering College, Dalian Maritime University, Dalian 116026, P. R. China (e-mail: n.wang.dmu.cn@gmail.com). B. J. Dai is with Dalian Electric Traction R&D Center, China CNR Corporation Limited, Dalian 116022, P. R. China (e-mail: daibijun@chinacnr.com). M. Han is with Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, P. R. China (email: minhan@dlut.edu.cn).

This work is supported by the National Natural Science Foundation of P. R. China (under Grants 51009017, 51379002 and 61074096), Applied Basic Research Funds from Ministry of Transport of P. R. China (under Grant 2012-329-225-060), China Postdoctoral Science Foundation (under Grant 2012M520629), Program for Liaoning Excellent Talents in University (under Grant LJQ2013055), Fundamental Research Funds for the Central Universities of P. R. China (under Grants 2009QN025, 2011JC002 and 3132013025), and Program for Liaoning Province Science and Technology Plan (under Grant 2010219014).

To circumvent the foregoing problem, the self-organizing FNN (SOFNN) with structure and parameter updated simultaneously have been proposed in [14]–[20] and references therein, which can automatically generate fuzzy rules or hidden nodes in addition to parameter update.

In this context, an adaptive self-constructing RBF neural control (AS-RBFNC) scheme for trajectory tracking of MIMO nonlinear systems in the presence of system uncertainties and unknown time-varying disturbances is proposed in this paper. In the AS-RBFNC, system uncertainties and unknown dynamics can be identified online by a self-constructing RBF neural network (SC-RBFNN) which is implemented by employing dynamically constructive hidden nodes according to the structure learning criteria including hidden node generating and pruning.

II. Problem Formulation

Consider a class of nth-order multivariable nonlinear systems as follows:

$$\mathbf{x}^{(n)} = \mathbf{f}(\mathbf{z}) + \mathbf{G}(\mathbf{z})\mathbf{u} + \mathbf{d}(\mathbf{z}, t) \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \cdots, x_m]^T \in R^m$ is a part of the state vector $\mathbf{z} = [\mathbf{x}^T, \dot{\mathbf{x}}^T, \cdots, (\mathbf{x}^{(n-1)})^T]^T \in R^r, r = mn, \mathbf{f}(\mathbf{z}) \in R^m$ and $\mathbf{G}(\mathbf{z}) \in R^{m\times m}$ are unknown smooth functions, $\mathbf{u} \in R^m$ is the control input, and $\mathbf{d}(\mathbf{z}, t) \in R^m$ is an unknown bounded function representing system uncertainty and external disturbance in the system, satisfying $\|\mathbf{d}(\mathbf{z}, t)\| \le \varepsilon_d$ for every $\mathbf{z} \in R^r$ and all $t \ge 0$. To ensure the controllability of the system, the input matrix $\mathbf{G}(\mathbf{z})$ needs to be invertible for all $\mathbf{z}$ in a particular compact region $U_z \subset R^r$ in which $g_l, g_u > 0$ exists, such that $g_l \le \|\mathbf{G}(\mathbf{z})\| \le g_u < \infty$. Given the desired trajectory $\mathbf{z}_d = [\mathbf{x}_d^T, \dot{\mathbf{x}}_d^T, \cdots, (\mathbf{x}_d^{(n-1)})^T]^T$, the

978-1-4799-1484-5/14/$31.00

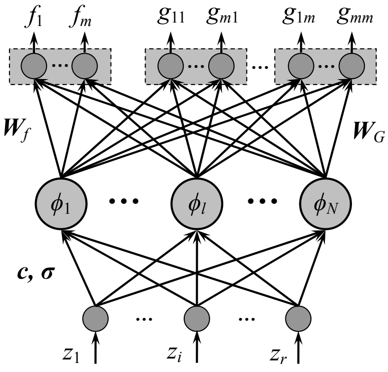

Fig. 1. Archtecture of the SC-RBFNN.

tracking error and sliding surface are defined as follows:

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_d \tag{2}$$

$$\mathbf{s} = \mathbf{e}^{(n-1)} + \mathbf{\Lambda}_n \mathbf{e}^{(n-2)} + \cdots + \mathbf{\Lambda}_1 \int_0^t \mathbf{e}(\tau) d\tau \tag{3}$$

where $\mathbf{\Lambda}_k = diag(\lambda_{k,1}, \lambda_{k,2}, \cdots, \lambda_{k,m}), \lambda_{k,i} > 0, k = 1, 2, \cdots, n, i = 1, 2, \cdots, m$.

In this context, combining (1) with (3), we have

$$\dot{\mathbf{s}} = \mathbf{f}(\mathbf{z}) + \mathbf{G}(\mathbf{z})\mathbf{u} + \mathbf{d}(\mathbf{z}, t) - \mathbf{x}_d^{(n)} + \mathbf{\Lambda}\underline{\mathbf{e}} \tag{4}$$

where $\underline{\mathbf{e}} = \mathbf{z} - \mathbf{z}_d = [\mathbf{e}^T, \dot{\mathbf{e}}^T, \cdots, (\mathbf{e}^{(n-1)})^T]^T$ and $\mathbf{\Lambda} = [\mathbf{\Lambda}_1, \mathbf{\Lambda}_2, \cdots, \mathbf{\Lambda}_n] \in R^{m \times r}$.

If $\mathbf{f}(\mathbf{z})$, $\mathbf{G}(\mathbf{z})$ and $\mathbf{d}(\mathbf{z}, t)$ are known, the ideal control law $\mathbf{u}^*$ can be designed according to the well-known feedback linearization method as follows:

$$\mathbf{u}^* = -\mathbf{G}^{-1}(\mathbf{z}) \left(\mathbf{K}\mathbf{s} + \mathbf{f}_d(\mathbf{z}) - \mathbf{x}_d^{(n)} + \mathbf{\Lambda}\underline{\mathbf{e}} \right) \tag{5}$$

where $\mathbf{f}_d(\mathbf{z}) = \mathbf{f}(\mathbf{z}) + \mathbf{d}(\mathbf{z}, t)$, and $\mathbf{K} \in R^{m \times m}$ is a symmetric positive-definite matrix. It implies that tracking errors converge exponentially to zero in a precisely modeled system without disturbances. However, it is impossible to implement this ideal controller in practical applications since dynamics $\mathbf{f}(\mathbf{z})$ and $\mathbf{G}(\mathbf{z})$ in the presence of disturbance $\mathbf{d}(\mathbf{z}, t)$ are perturbed or unknown. In this context, our objective will propose an adaptive self-constructing radial-basis-function neural control (AS-RBFNC) scheme for tracking the MIMO nonlinear system (1) with unknown dynamics and disturbances.

III. Self-Constructing Radial-Basis-Function Neural Network

A. Architecture of SC-RBFNN

Since dynamics $\mathbf{f}$, $\mathbf{G}$ and $\mathbf{d}$ are actually unknown, a self-constructing radial-basis-function neural network (SC-RBFNN) shown in Fig. 1 is proposed to identify online the lumped unknown dynamics $\mathbf{f}_d(\mathbf{z})$ and $\mathbf{G}(\mathbf{z}) = [\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_m]$ as follows:

$$\mathbf{f}_{\mathrm{NN}}(\mathbf{z}) = [f_1, f_2, \cdots, f_m]^T = \mathbf{W}_f^T \mathbf{\Phi}(\mathbf{z}; \mathbf{c}, \boldsymbol{\sigma}) \tag{6}$$

$$\mathbf{g}_{j,\mathrm{NN}}(\mathbf{z}) = [g_{1j}, g_{2j}, \cdots, g_{mj}]^T = \mathbf{W}_{g_j}^T \mathbf{\Phi}(\mathbf{z}; \mathbf{c}, \boldsymbol{\sigma}) \tag{7}$$

where $\mathbf{f}_{\mathrm{NN}}, \mathbf{g}_{j,\mathrm{NN}} : U_z \subset R^r \to R^m$, $j = 1, 2, \cdots, m$, $\mathbf{W}_f = [\boldsymbol{\omega}_{f,1}, \boldsymbol{\omega}_{f,2}, \cdots, \boldsymbol{\omega}_{f,m}] \in R^{N \times m}$, $\boldsymbol{\omega}_{f,j} = [w_{f,j}^1, w_{f,j}^2, \cdots, w_{f,j}^N]^T$, $\mathbf{W}_{g_j} = [\boldsymbol{\omega}_{g_j,1}, \boldsymbol{\omega}_{g_j,2}, \cdots, \boldsymbol{\omega}_{g_j,m}] \in R^{N \times m}$, $\boldsymbol{\omega}_{g_i,j} = [w_{g_i,j}^1, w_{g_i,j}^2, \cdots, w_{g_i,j}^N]^T$, $\mathbf{c} = [\mathbf{c}_1^T, \mathbf{c}_2^T, \cdots, \mathbf{c}_N^T]^T \in R^{rN}$, $\boldsymbol{\sigma} = [\boldsymbol{\sigma}_1^T, \boldsymbol{\sigma}_2^T, \cdots, \boldsymbol{\sigma}_N^T]^T \in R^{rN}$ and $\mathbf{\Phi} = [\phi_1, \phi_2, \cdots, \phi_N]^T$. Here, the activation function ϕ_l is defined by

$$\begin{aligned} \phi_l &= \exp\left(-\sum_{i=1}^{r} \frac{(z_i - c_i^l)^2}{(\sigma_i^l)^2} \right) \\ &= \exp\left(-(\mathbf{z} - \mathbf{c}_l)^T \mathbf{\Sigma}_l^{-2} (\mathbf{z} - \mathbf{c}_l) \right) \end{aligned} \tag{8}$$

where $\mathbf{\Sigma}_l = diag\left(\sigma_1^l, \cdots, \sigma_r^l\right)$, and $\mathbf{c}_l = [c_1^l, c_2^l, \cdots, c_r^l]^T$ and $\boldsymbol{\sigma}_l = [\sigma_1^l, \sigma_2^l, \cdots, \sigma_r^l]^T$ are the center and width vectors, respectively.

Without loss of generality, assume that there exists an optimal RBFNN with N^* hidden nodes using $\mathbf{c}^*$ and $\boldsymbol{\sigma}^*$ that can identify the nonlinear function $\mathbf{f}_d(\mathbf{z})$ and $\mathbf{G}(\mathbf{z})$ with the minimal functional approximation errors (MFAEs), i.e.,

$$\begin{aligned} \mathbf{f}_d(\mathbf{z}) &= \mathbf{f}_{\mathrm{NN}}^*(\mathbf{z}; \mathbf{W}_f^*, \mathbf{c}^*, \boldsymbol{\sigma}^*) + \boldsymbol{\varepsilon}_f(\mathbf{z}, N^*) \\ &= (\mathbf{W}_f^*)^T \mathbf{\Phi}(\mathbf{z}; \mathbf{c}^*, \boldsymbol{\sigma}^*) + \boldsymbol{\varepsilon}_f^*(\mathbf{z}) \end{aligned} \tag{9}$$

$$\begin{aligned} \mathbf{g}_j(\mathbf{z}) &= \mathbf{g}_{j,\mathrm{NN}}^*(\mathbf{z}; \mathbf{W}_{g_j}^*, \mathbf{c}^*, \boldsymbol{\sigma}^*) + \boldsymbol{\varepsilon}_{g_j}(\mathbf{z}, N^*) \\ &= (\mathbf{W}_{g_j}^*)^T \mathbf{\Phi}(\mathbf{z}; \mathbf{c}^*, \boldsymbol{\sigma}^*) + \boldsymbol{\varepsilon}_{g_j}^*(\mathbf{z}) \end{aligned} \tag{10}$$

where, $\boldsymbol{\varepsilon}_f^*(\mathbf{z}), \boldsymbol{\varepsilon}_{g_j}^*(\mathbf{z}) \in R^m$ are the MFAEs and satisfy $\|\boldsymbol{\varepsilon}_f^*(\mathbf{z})\| \le \bar{\varepsilon}_f$ and $\|\boldsymbol{\varepsilon}_{g_j}^*(\mathbf{z})\| \le \bar{\varepsilon}_{g_j}$, and the optimal output weights $\mathbf{W}_f^*$ and $\mathbf{W}_{g_j}^*$ are derived from

$$\mathbf{W}_f^* = \arg\min_{\mathbf{W}_f} \left(\max_{\mathbf{z} \in U_z} \|\mathbf{f}_{\mathrm{NN}}(\mathbf{z}; \mathbf{W}_f, \mathbf{c}^*, \boldsymbol{\sigma}^*) - \mathbf{f}_d(\mathbf{z})\| \right)$$

$$\mathbf{W}_{g_j}^* = \arg\min_{\mathbf{W}_{g_j}} \left(\max_{\mathbf{z} \in U_z} \|\mathbf{g}_{j,\mathrm{NN}}(\mathbf{z}; \mathbf{W}_{g_j}, \mathbf{c}^*, \boldsymbol{\sigma}^*) - \mathbf{g}_j(\mathbf{z})\| \right)$$

B. Self-constructing Scheme

Actually, each hidden node in the RBFNN represents a local receptive domain. Accordingly, finding the suitable number of hidden nodes online is to optimize the input space partition such that the current input can be accommodated well by at least one RBF hidden node. It is yet unexpected that too many hidden nodes are activated simultaneously with low firing strength, and thereby resulting in redundant hidden nodes. In this context, an SC-RBFNN with self-organizing structure corresponding to current input $\mathbf{z}(t)$ is proposed.

To be specific, the SC-RBFNN begins with no any hidden nodes, i.e., $\mathbf{c}(0) = \emptyset$, $\boldsymbol{\sigma}(0) = \emptyset$, $\mathbf{W}_f(0) = \emptyset$, $\mathbf{W}_{g_j}(0) = \emptyset$, $N(0) = 0$. According to the novelty of current observation $\mathbf{z}(t)$ to the existing RBF hidden nodes, the self-constructing scheme decides to generate new hidden nodes or to prune redundant ones in the whole structure learning process. Without loss of generality, consider the input $\mathbf{z}(t)$ at time instant t, i.e., $\mathbf{c}(t-1) = [\mathbf{c}_1^T, \mathbf{c}_2^T, \cdots, \mathbf{c}_{N(t-1)}^T]^T$ and $\boldsymbol{\sigma}(t-1) = [\boldsymbol{\sigma}_1^T, \boldsymbol{\sigma}_2^T, \cdots, \boldsymbol{\sigma}_{N(t-1)}^T]^T$.

Calculate the generalized distance between the current input $\mathbf{z}(t)$ and the existing RBF hidden nodes as follows:

$$\begin{aligned} d^l = \big(\mathbf{z}(t) - \mathbf{c}_l(t-1)\big)^T \mathbf{\Sigma}_l^{-2}(t-1) \big(\mathbf{z}(t) - \mathbf{c}_l(t-1)\big), \\ l = 1, 2, \cdots, N(t-1) \end{aligned} \tag{11}$$

where $\mathbf{c}_l(t-1) = [c_1^{l(t-1)}, \cdots, c_r^{l(t-1)}]^T$ and $\mathbf{\Sigma}_l(t-1) = diag(\sigma_1^{l(t-1)}, \cdots, \sigma_r^{l(t-1)})$.

1) Hidden Node Generating: Find the nearest hidden node

$$d_{\min} = \min_{l=1,2,\cdots,N(t-1)} d^l. \tag{12}$$

If

$$d_{\min} > d_{\min}^{\text{th}} \tag{13}$$

there does not exist any hidden node representing the current input. It needs to generate a new RBF unit according to the current input partition, i.e.,

$$\mathbf{c}_{N(t)} = \mathbf{z}(t),\ \boldsymbol{\sigma}_{N(t)} = \boldsymbol{\sigma}_{\text{init}},\ \mathbf{w}_{f,N(t)} = \mathbf{0},\ \mathbf{w}_{g_j,N(t)} = \mathbf{0},$$
$$N(t) = N(t-1) + 1 \tag{14}$$

where $\mathbf{w}_{f,l} = [w_{f,1}^l, w_{f,2}^l, \cdots, w_{f,m}^l]$ and $\mathbf{w}_{g_j,l} = [w_{g_j,1}^l, w_{g_j,2}^l, \cdots, w_{g_j,m}^l]$.

Here, $d_{\min}^{\text{th}}$ is the user-defined threshold and can be easily chosen as $d_{\min}^{\text{th}} = \ln(1/\epsilon), 0 < \epsilon \leq 1$, such that the dominant firing strength is no less than ϵ, and $\boldsymbol{\sigma}_{\text{init}}$ is the initial width for the newly generated RBF node. Otherwise, no any new hidden node is recruited, i.e., $N(t) = N(t-1)$.

2) Hidden Node Pruning: In sequential, find the redundant RBF hidden nodes as follows:

$$J_r = \{l^\circ\},\ d^{l^\circ} > d_0. \tag{15}$$

If

$$J_r \neq \emptyset \tag{16}$$

it needs to prune redundant hidden nodes, i.e.,

$$\mathbf{c}_{l^\circ} = \emptyset,\ \boldsymbol{\sigma}_{l^\circ} = \emptyset,\ \mathbf{w}_{f,l^\circ} = \emptyset,\ \mathbf{w}_{g_j,l^\circ} = \emptyset,$$
$$N(t) = N(t) - |J_r|,\ l^\circ \in J_r \tag{17}$$

Here, d_0 is the user-defined threshold under which the fuzzy rule is considered to be inactive, and thereby being simply chosen as $d_0 = \ln(1/\varsigma), 0 < \varsigma \leq 1$.

IV. Adaptive Self-constructing Radial-Basis-Function Neural Control

By employing the proposed SC-RBFNN approximation in (6) and (7), an adaptive self-constructing radial-basis-function neural control (AS-RBFNC) scheme is designed for tracking the MIMO nonlinear system (1) as follows:

$$\mathbf{u}_{\text{NN}} = -\widehat{\mathbf{G}}^{-1}(\mathbf{z}) \left(\mathbf{Ks} + \widehat{\mathbf{f}}_{\text{NN}}(\mathbf{z}) - \mathbf{x}_d^{(n)} + \mathbf{\Lambda} \underline{\mathbf{e}} \right) \tag{18}$$

where $\widehat{\mathbf{f}}_{\text{NN}}(\mathbf{z}; \widehat{\mathbf{W}}_f, \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}})$ and $\widehat{\mathbf{G}}_{\text{NN}}(\mathbf{z}; \widehat{\mathbf{W}}_G, \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}})$ are the SC-RBFNN-based approximations which are parameterized by parameter estimates $\widehat{\mathbf{W}}_f, \widehat{\mathbf{W}}_G, \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}}$ and derive from the adaptive laws in the following subsections.

A. SC-RBFNN Approximation

In the AS-RBFNC (18), the unknown dynamics $\mathbf{f}_d(\mathbf{z})$ and $\mathbf{G}(\mathbf{z})$ are identified online by the proposed SC-RBFNN with adaptive parameters in addition to dynamic structure according to (9) and (10) with $N(t)$ fuzzy rules as follows:

$$\mathbf{f}_d(\mathbf{z}) = \widehat{\mathbf{W}}_f^T \mathbf{\Phi}(\mathbf{z}; \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}}) + \widehat{\boldsymbol{\varepsilon}}_f(\mathbf{z}) \tag{19}$$
$$\mathbf{g}_j(\mathbf{z}) = \widehat{\mathbf{W}}_{g_j}^T \mathbf{\Phi}(\mathbf{z}; \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}}) + \widehat{\boldsymbol{\varepsilon}}_{g_j}(\mathbf{z}) \tag{20}$$

where $\widehat{\mathbf{W}}_f, \widehat{\mathbf{W}}_{g_j} \in R^{N \times m}, \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}} \in R^{rN}$ are estimated parameters, and $\widehat{\boldsymbol{\varepsilon}}_f$ and $\widehat{\boldsymbol{\varepsilon}}_{g_j}$ are actual approximation errors determined by

$$\begin{aligned}\widehat{\boldsymbol{\varepsilon}}_f &= \mathbf{f}_d(\mathbf{z}) - \widehat{\mathbf{f}}_{\text{NN}}(\mathbf{z}) = \mathbf{f}_{\text{NN}}^*(\mathbf{z}) - \widehat{\mathbf{f}}_{\text{NN}}(\mathbf{z}) + \boldsymbol{\varepsilon}_f^*(\mathbf{z}) \\ &= (\mathbf{W}_f^*)^T \mathbf{\Phi}^* - \widehat{\mathbf{W}}_f^T \widehat{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_f^* \end{aligned} \tag{21}$$
$$\begin{aligned}\widehat{\boldsymbol{\varepsilon}}_{g_j} &= \mathbf{g}_j(\mathbf{z}) - \widehat{\mathbf{g}}_{j,\text{NN}}(\mathbf{z}) = \mathbf{g}_{j,\text{NN}}^*(\mathbf{z}) - \widehat{\mathbf{g}}_{j,\text{NN}}(\mathbf{z}) + \boldsymbol{\varepsilon}_{g_j}^*(\mathbf{z}) \\ &= (\mathbf{W}_{g_j}^*)^T \mathbf{\Phi}^* - \widehat{\mathbf{W}}_{g_j}^T \widehat{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_{g_j}^* \end{aligned} \tag{22}$$

where $\mathbf{\Phi}^* = \mathbf{\Phi}(\mathbf{z}; \mathbf{c}^*, \boldsymbol{\sigma}^*) \in R^{N^*}$ and $\widehat{\mathbf{\Phi}} = \mathbf{\Phi}(\mathbf{z}; \widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}}) \in R^N$. Without loss of generality, assume $N^* \geq N$ and $\mathbf{\Phi}^* = [\mathbf{\Phi}_1^*, \mathbf{0}], \mathbf{\Phi}_1^* \in R^N$, i.e. $(\mathbf{W}_f^*)^T \mathbf{\Phi}^* = (\mathbf{W}_{f1}^*)^T \mathbf{\Phi}_1^*$ and $(\mathbf{W}_{g_j}^*)^T \mathbf{\Phi}^* = (\mathbf{W}_{g_j 1}^*)^T \mathbf{\Phi}_1^*$, where $\mathbf{W}_{f1}^* \in R^{N \times m}, \mathbf{W}_{g_j 1}^* \in R^{N \times mm}, \mathbf{\Phi}_1^* = \mathbf{\Phi}(\mathbf{z}; \mathbf{c}_1^*, \boldsymbol{\sigma}_1^*) \in R^N$. From (21) and (22),

$$\widehat{\boldsymbol{\varepsilon}}_f = \widetilde{\mathbf{W}}_f^T \left(\widehat{\mathbf{\Phi}} + \widetilde{\mathbf{\Phi}} \right) + \widehat{\mathbf{W}}_f^T \widetilde{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_f^* \tag{23}$$
$$\widehat{\boldsymbol{\varepsilon}}_{g_j} = \widetilde{\mathbf{W}}_{g_j}^T \left(\widehat{\mathbf{\Phi}} + \widetilde{\mathbf{\Phi}} + \widehat{\mathbf{W}}_{g_j}^T \widetilde{\mathbf{\Phi}} \right) + \boldsymbol{\varepsilon}_{g_j}^* \tag{24}$$

where $\widetilde{\mathbf{W}}_f = \mathbf{W}_{f1}^* - \widehat{\mathbf{W}}_f$, $\widetilde{\mathbf{W}}_{g_j} = \mathbf{W}_{g_j 1}^* - \widehat{\mathbf{W}}_{g_j}$ and $\widetilde{\mathbf{\Phi}} = \mathbf{\Phi}_1^* - \widehat{\mathbf{\Phi}}$ are the output weight errors and regressor error, respectively. By applying the Taylor series expansion of $\mathbf{\Phi}(.)$ to $(\widehat{\mathbf{c}}, \widehat{\boldsymbol{\sigma}})$ in (23) and (24), we have

$$\begin{aligned}\widehat{\boldsymbol{\varepsilon}}_f &= \widetilde{\mathbf{W}}_f^T \widehat{\mathbf{\Phi}} + \widehat{\mathbf{W}}_f^T \widetilde{\mathbf{\Phi}} + \widetilde{\mathbf{W}}_f^T \widetilde{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_f^* \\ &= \widetilde{\mathbf{W}}_f^T \widehat{\mathbf{\Phi}} + \widehat{\mathbf{W}}_f^T \left(\mathbf{\Phi}_{\mathbf{c}}' \widetilde{\mathbf{c}} + \mathbf{\Phi}_{\boldsymbol{\sigma}}' \widetilde{\boldsymbol{\sigma}} + \mathbf{h}(\mathbf{z}; \widetilde{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}}) \right) + \widetilde{\mathbf{W}}_f^T \widetilde{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_f^* \\ &= \widetilde{\mathbf{W}}_f^T \widehat{\mathbf{\Phi}} + \widehat{\mathbf{W}}_f^T \mathbf{\Phi}_{\mathbf{c}}' \widetilde{\mathbf{c}} + \widehat{\mathbf{W}}_f^T \mathbf{\Phi}_{\boldsymbol{\sigma}}' \widetilde{\boldsymbol{\sigma}} + \boldsymbol{\varepsilon}_f \end{aligned} \tag{25}$$
$$\begin{aligned}\widehat{\boldsymbol{\varepsilon}}_{g_j} &= \widetilde{\mathbf{W}}_{g_j}^T \widehat{\mathbf{\Phi}} + \widehat{\mathbf{W}}_{g_j}^T \widetilde{\mathbf{\Phi}} + \widetilde{\mathbf{W}}_{g_j}^T \widetilde{\mathbf{\Phi}} + \boldsymbol{\varepsilon}_{g_j}^* \\ &= \widetilde{\mathbf{W}}_{g_j}^T \widehat{\mathbf{\Phi}} + \widehat{\mathbf{W}}_{g_j}^T \mathbf{\Phi}_{\mathbf{c}}' \widetilde{\mathbf{c}} + \widehat{\mathbf{W}}_{g_j}^T \mathbf{\Phi}_{\boldsymbol{\sigma}}' \widetilde{\boldsymbol{\sigma}} + \boldsymbol{\varepsilon}_{g_j} \end{aligned} \tag{26}$$

where $\widetilde{\mathbf{c}} = \mathbf{c}_1^* - \widehat{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_1^* - \widehat{\boldsymbol{\sigma}}$, $\mathbf{h}(\mathbf{z}; \widetilde{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}})$ is the high order term of $\widetilde{\mathbf{c}}$ and $\widetilde{\boldsymbol{\sigma}}$, $\mathbf{\Phi}_{\mathbf{c}}'$ and $\mathbf{\Phi}_{\boldsymbol{\sigma}}'$ are Jacobian matrices derived from

$$\mathbf{\Phi}_{\mathbf{c}}' = \frac{\partial \mathbf{\Phi}}{\partial \mathbf{c}} \Big|_{\substack{\mathbf{c}=\widehat{\mathbf{c}} \\ \boldsymbol{\sigma}=\widehat{\boldsymbol{\sigma}}}} = diag\left(\boldsymbol{\phi}_{\mathbf{c}_1}^T, \cdots, \boldsymbol{\phi}_{\mathbf{c}_N}^T \right) \in R^{N \times rN} \tag{27}$$
$$\mathbf{\Phi}_{\boldsymbol{\sigma}}' = \frac{\partial \mathbf{\Phi}}{\partial \boldsymbol{\sigma}} \Big|_{\substack{\mathbf{c}=\widehat{\mathbf{c}} \\ \boldsymbol{\sigma}=\widehat{\boldsymbol{\sigma}}}} = diag\left(\boldsymbol{\phi}_{\boldsymbol{\sigma}_1}^T, \cdots, \boldsymbol{\phi}_{\boldsymbol{\sigma}_N}^T \right) \in R^{N \times rN} \tag{28}$$

Here, $\boldsymbol{\phi}_{\mathbf{c}_l}^T = \frac{\partial \phi_l}{\partial \mathbf{c}_l^T} = \left[\frac{\partial \phi_l}{\partial c_1^l}, \cdots, \frac{\partial \phi_l}{\partial c_m^l} \right]$ and $\boldsymbol{\phi}_{\boldsymbol{\sigma}_l}^T = \frac{\partial \phi_l}{\partial \boldsymbol{\sigma}_l^T} = \left[\frac{\partial \phi_l}{\partial \sigma_1^l}, \cdots, \frac{\partial \phi_l}{\partial \sigma_m^l} \right]$. From (8), we get

$$\boldsymbol{\phi}_{\mathbf{c}_l}^T = \left[\phi_1^{c,l}, \cdots, \phi_m^{c,l} \right] = 2\widehat{\phi}_l \left[\frac{z_1 - \widehat{c}_1^l}{(\widehat{\sigma}_1^l)^2}, \cdots, \frac{z_m - \widehat{c}_m^l}{(\widehat{\sigma}_m^l)^2} \right]$$
$$\boldsymbol{\phi}_{\boldsymbol{\sigma}_l}^T = \left[\phi_1^{\sigma,l}, \cdots, \phi_m^{\sigma,l} \right] = 2\widehat{\phi}_l \left[\frac{(z_1 - \widehat{c}_1^l)^2}{(\widehat{\sigma}_1^l)^3}, \cdots, \frac{(z_m - \widehat{c}_m^l)^2}{(\widehat{\sigma}_m^l)^3} \right]$$

and $\boldsymbol{\varepsilon}_f, \boldsymbol{\varepsilon}_{g_j}$ are the residual approximation errors given by

$$\boldsymbol{\varepsilon}_f = \widehat{\mathbf{W}}_f^T \mathbf{h}(\mathbf{z}; \widetilde{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}}) + \widetilde{\mathbf{W}}_f^T \widetilde{\boldsymbol{\Phi}} + \boldsymbol{\varepsilon}_f^* \quad (29)$$

$$\boldsymbol{\varepsilon}_{g_j} = \widehat{\mathbf{W}}_{g_j}^T \mathbf{h}(\mathbf{z}; \widetilde{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}}) + \widetilde{\mathbf{W}}_{g_j}^T \widetilde{\boldsymbol{\Phi}} + \boldsymbol{\varepsilon}_{g_j}^* \quad (30)$$

B. Adaptive Laws

Choose the adaptive laws as follows:

$$\dot{\widehat{\boldsymbol{\omega}}}_{f,j} = \eta_f s_j \widehat{\boldsymbol{\Phi}} \quad (31)$$

$$\dot{\widehat{\boldsymbol{\omega}}}_{g_j,k} = \eta_{g_j} u_j s_k \widehat{\boldsymbol{\Phi}} \quad (32)$$

$$\dot{\widehat{\mathbf{c}}} = \eta_c (\boldsymbol{\Phi}'_{\mathbf{c}})^T \left(\widehat{\mathbf{W}}_f - \sum_{j=1}^{m} u_j \widehat{\mathbf{W}}_{g_j} \right) \mathbf{s} \quad (33)$$

$$\dot{\widehat{\boldsymbol{\sigma}}} = \eta_\sigma (\boldsymbol{\Phi}'_{\boldsymbol{\sigma}})^T \left(\widehat{\mathbf{W}}_f - \sum_{j=1}^{m} u_j \widehat{\mathbf{W}}_{g_j} \right) \mathbf{s} \quad (34)$$

where $\eta_f, \eta_{g_j}, \eta_c, \eta_\sigma > 0$ are gains and design the parameter matrix $\mathbf{K}$ in (18) as

$$\mathbf{K} = \frac{\mathbf{I} + \boldsymbol{\Delta}^{-2}}{2} \quad (35)$$

where $\boldsymbol{\Delta} = diag\,(\delta_1, \delta_2, \cdots, \delta_m)\,, \delta_j > 0.$

C. Stability Analysis

Theorem 1: Consider the MIMO nonlinear system (1) with the proposed AS-RBFNC scheme using (18), and the adaptive laws for parameter updates using (31)-(34), where the online approximation $\widehat{\mathbf{f}}_{\mathrm{NN}}$ and $\widehat{\mathbf{g}}_{j,\mathrm{NN}}, j = 1, 2, \cdots, m$ are realized by the SC-RBFNN (19) and (20), respectively. Then, the tracking errors $\underline{\mathbf{e}}(t)$ are globally asymptotical stable.

Proof: Substituting the control law (18) into (1) yields

$$\begin{aligned}\dot{\mathbf{s}} =& -\mathbf{K}\mathbf{s} + \widehat{\boldsymbol{\varepsilon}}_f - \widehat{\boldsymbol{\varepsilon}}_G \mathbf{u} \\ =& -\mathbf{K}\mathbf{s} + \widetilde{\mathbf{W}}_f^T \widehat{\boldsymbol{\Phi}} + \widehat{\mathbf{W}}_f^T \boldsymbol{\Phi}'_{\mathbf{c}} \widetilde{\mathbf{c}} + \widehat{\mathbf{W}}_f^T \boldsymbol{\Phi}'_{\boldsymbol{\sigma}} \widetilde{\boldsymbol{\sigma}} \\ & - \sum_{j=1}^{m} u_j \left(\widetilde{\mathbf{W}}_{g_j}^T \widehat{\boldsymbol{\Phi}} + \widehat{\mathbf{W}}_{g_j}^T \boldsymbol{\Phi}'_{\mathbf{c}} \widetilde{\mathbf{c}} + \widehat{\mathbf{W}}_{g_j}^T \boldsymbol{\Phi}'_{\boldsymbol{\sigma}} \widetilde{\boldsymbol{\sigma}} \right) + \boldsymbol{\varepsilon} \quad (36)\end{aligned}$$

where $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_f - \sum_{j=1}^{m} u_j \boldsymbol{\varepsilon}_{g_j}$ and $\widehat{\boldsymbol{\varepsilon}}_G = [\widehat{\boldsymbol{\varepsilon}}_{g_1}, \widehat{\boldsymbol{\varepsilon}}_{g_2}, \cdots, \widehat{\boldsymbol{\varepsilon}}_{g_m}]$.

Consider the following Lyapunov function,

$$V\big(\mathbf{s}, \widetilde{\mathbf{W}}_f, \widetilde{\mathbf{W}}_{g_j}, \widetilde{\mathbf{c}}, \widetilde{\boldsymbol{\sigma}}; t\big) = \frac{1}{2} \times \left[\mathbf{s}^T \mathbf{s} + \frac{\mathrm{tr}\big(\widetilde{\mathbf{W}}_f^T \widetilde{\mathbf{W}}_f\big)}{\eta_f} + \sum_{j=1}^{m} \frac{\mathrm{tr}\big(\widetilde{\mathbf{W}}_{g_j}^T \widetilde{\mathbf{W}}_{g_j}\big)}{\eta_{g_j}} + \frac{\widetilde{\mathbf{c}}^T \widetilde{\mathbf{c}}}{\eta_c} + \frac{\widetilde{\boldsymbol{\sigma}}^T \widetilde{\boldsymbol{\sigma}}}{\eta_\sigma} \right]$$

Differentiating V with respect to time t and using (36), we have

$$\begin{aligned}\dot{V} =& \mathbf{s}^T \dot{\mathbf{s}} + \frac{\mathrm{tr}\big(\widetilde{\mathbf{W}}_f^T \dot{\widetilde{\mathbf{W}}}_f\big)}{\eta_f} + \sum_{j=1}^{m} \frac{\mathrm{tr}\big(\widetilde{\mathbf{W}}_{g_j}^T \dot{\widetilde{\mathbf{W}}}_{g_j}\big)}{\eta_{g_j}} + \frac{\widetilde{\mathbf{c}}^T \dot{\widetilde{\mathbf{c}}}}{\eta_c} + \frac{\widetilde{\boldsymbol{\sigma}}^T \dot{\widetilde{\boldsymbol{\sigma}}}}{\eta_\sigma} \\ =& -\mathbf{s}^T \mathbf{K}\mathbf{s} + \mathbf{s}^T \boldsymbol{\varepsilon} + \mathbf{s}^T \widetilde{\mathbf{W}}_f^T \widehat{\boldsymbol{\Phi}} + \mathbf{s}^T \sum_{j=1}^{m} u_j \widetilde{\mathbf{W}}_{g_j}^T \widehat{\boldsymbol{\Phi}} + \mathbf{s}^T \widehat{\mathbf{W}}_f^T \boldsymbol{\Phi}'_{\mathbf{c}} \widetilde{\mathbf{c}} \\ & + \mathbf{s}^T \widehat{\mathbf{W}}_f^T \boldsymbol{\Phi}'_{\boldsymbol{\sigma}} \widetilde{\boldsymbol{\sigma}} - \sum_{j=1}^{m} u_j \mathbf{s}^T \big(\widehat{\mathbf{W}}_{g_j}^T \boldsymbol{\Phi}'_{\mathbf{c}} \widetilde{\mathbf{c}} + \widehat{\mathbf{W}}_{g_j}^T \boldsymbol{\Phi}'_{\boldsymbol{\sigma}} \widetilde{\boldsymbol{\sigma}} \big) - \eta_c^{-1} \widetilde{\mathbf{c}}^T \dot{\widehat{\mathbf{c}}} \\ & - \eta_f^{-1} \mathrm{tr}\big(\widetilde{\mathbf{W}}_f^T \dot{\widehat{\mathbf{W}}}_f\big) - \sum_{j=1}^{m} \eta_{g_j}^{-1} \mathrm{tr}\big(\widetilde{\mathbf{W}}_{g_j}^T \dot{\widehat{\mathbf{W}}}_{g_j}\big) - \eta_\sigma^{-1} \widetilde{\boldsymbol{\sigma}}^T \dot{\widehat{\boldsymbol{\sigma}}} \\ =& \sum_{j=1}^{m} \widetilde{\boldsymbol{\omega}}_{f,j}^T \big(s_j \widehat{\boldsymbol{\Phi}} - \eta_f^{-1} \dot{\widehat{\boldsymbol{\omega}}}_{f,j} \big) + \sum_{j=1}^{m} \sum_{k=1}^{m} \widetilde{\boldsymbol{\omega}}_{g_j,k}^T \big(u_j s_k \widehat{\boldsymbol{\Phi}} - \eta_{g_j}^{-1} \dot{\widehat{\boldsymbol{\omega}}}_{g_j,k} \big) \\ & + \widetilde{\mathbf{c}}^T \left((\boldsymbol{\Phi}'_{\mathbf{c}})^T \widehat{\mathbf{W}}_f \mathbf{s} - \sum_{j=1}^{m} u_j (\boldsymbol{\Phi}'_{\mathbf{c}})^T \widehat{\mathbf{W}}_{g_j} \mathbf{s} - \eta_c^{-1} \dot{\widehat{\mathbf{c}}} \right) \\ & + \widetilde{\boldsymbol{\sigma}}^T \left((\boldsymbol{\Phi}'_{\boldsymbol{\sigma}})^T \widehat{\mathbf{W}}_f \mathbf{s} - \sum_{j=1}^{m} u_j (\boldsymbol{\Phi}'_{\boldsymbol{\sigma}})^T \widehat{\mathbf{W}}_{g_j} \mathbf{s} - \eta_\sigma^{-1} \dot{\widehat{\boldsymbol{\sigma}}} \right) \\ & - \mathbf{s}^T \mathbf{K}\mathbf{s} + \mathbf{s}^T \boldsymbol{\varepsilon}\end{aligned}$$

From (31) - (34) and (35), we further have

$$\begin{aligned}\dot{V} =& -\frac{1}{2} \mathbf{s}^T \left(\mathbf{I} + \boldsymbol{\Delta}^{-2} \right) \mathbf{s} + \mathbf{s}^T \boldsymbol{\varepsilon} \\ =& -\frac{1}{2} \left(\mathbf{s}^T \mathbf{s} + \left(\boldsymbol{\Delta}^{-1} \mathbf{s} - \boldsymbol{\Delta} \boldsymbol{\varepsilon} \right)^T \left(\boldsymbol{\Delta}^{-1} \mathbf{s} - \boldsymbol{\Delta} \boldsymbol{\varepsilon} \right) - \boldsymbol{\varepsilon}^T \boldsymbol{\Delta}^2 \boldsymbol{\varepsilon} \right) \\ \leq& -\frac{\mathbf{s}^T \mathbf{s}}{2} + \frac{\delta^2 \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon}}{2} \quad (37)\end{aligned}$$

where $\delta = \max\,(\delta_1, \delta_2, \cdots, \delta_m)$.

Assume $\boldsymbol{\varepsilon} \in L_2[0, T), \forall\, T \in [0, \infty)$, i.e. $\int_0^T \|\boldsymbol{\varepsilon}\|^2 dt < \infty$, we have

$$\frac{1}{2} \int_0^T \|\mathbf{s}\|^2 dt \leq V(0) + \frac{1}{2} \delta^2 \int_0^T \|\boldsymbol{\varepsilon}\|^2 dt \quad (38)$$

Since $V(0)$ is finite, we have $\mathbf{s} \in L_2$. By the Barbalat's lemma, $\lim_{t\to\infty} \mathbf{s}(t) = \mathbf{0}$. From (3), $\lim_{t\to\infty} \mathbf{e}(t) = \mathbf{0}$. It further implies that $\underline{\mathbf{e}}(t) = [\mathbf{e}^T, \dot{\mathbf{e}}^T, \cdots, (\mathbf{e}^{(n-1)})^T]^T \to \mathbf{0}$ while $t \to \infty$. This concludes the proof. ■

V. Simulation Studies

In order to demonstrate the effectiveness of the proposed AS-RBFNC scheme, we conduct simulation studies on a two-link robot manipulator governed by

$$\begin{aligned}\ddot{\mathbf{x}} =& -\mathbf{M}^{-1}(\mathbf{x}) \left[\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}) + \breve{\mathbf{G}}(\mathbf{x}) \right] + \mathbf{M}^{-1}(\mathbf{x})(\mathbf{u} + \mathbf{u}_d) \\ \equiv& \mathbf{f}(\mathbf{z}) + \mathbf{G}(\mathbf{z})\mathbf{u} + \mathbf{d}(\mathbf{z}, t) \quad (39)\end{aligned}$$

with

$$\mathbf{M}(\mathbf{x}) = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix},$$

$$\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}) = (m_2 l_1 l_{c_2} \sin x_2) \begin{bmatrix} -\dot{x}_2 & -(\dot{x}_1 + \dot{x}_2) \\ \dot{x}_1 & 0 \end{bmatrix},$$

$$\breve{\mathbf{G}}(\mathbf{x}) = \begin{bmatrix} m_1 g l_{c_1} \cos x_1 + m_2 g (l_1 \cos x_1 + l_{c_2} \cos(x_1 + x_2)) \\ m_2 g l_{c_2} \cos(x_1 + x_2) \end{bmatrix},$$

$m_{11} = m_1 l_{c_1}^2 + m_2(l_1^2 + l_{c_2}^2 + 2 l_1 l_{c_2} \cos x_2)$, $m_{12} = m_{21} = m_2(l_{c_2}^2 + l_1 l_{c_2} \cos x_2)$, $m_{22} = m_2 l_{c_2}^2$, where $\mathbf{x} = [x_1, x_2]^T$

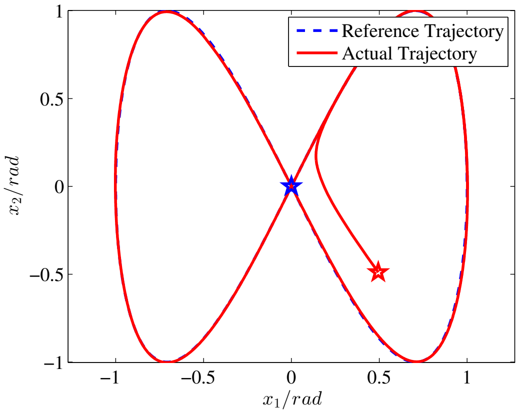

Fig. 2. Reference and actual trajectories.

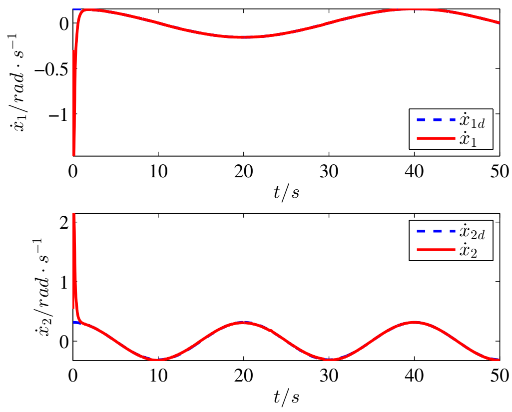

Fig. 4. Desired and actual states $\dot{x}_1$ and $\dot{x}_2$.

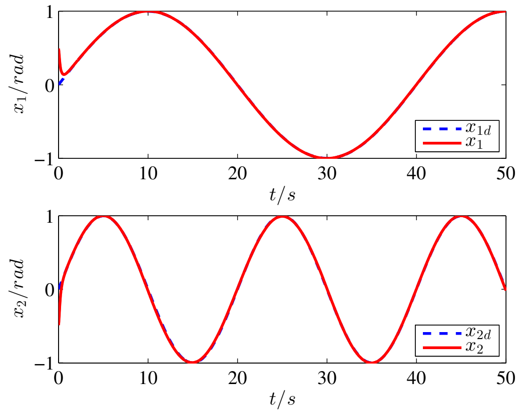

Fig. 3. Desired and actual states x_1 and x_2.

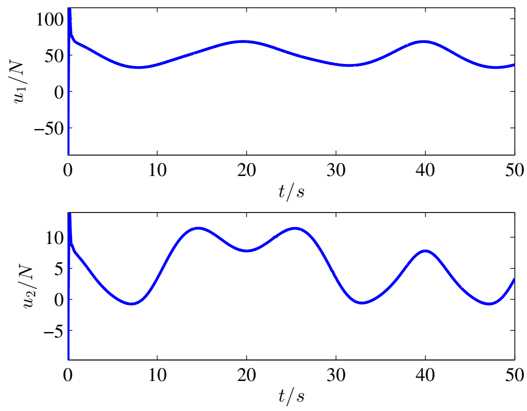

Fig. 5. Control forces u_1 and u_2.

is the joint angular vector, $\mathbf{z} = [\mathbf{x}^T, \dot{\mathbf{x}}^T]^T$ is the state vector, $\mathbf{M}(\mathbf{x}), \mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}), \breve{\mathbf{G}}(\mathbf{x})$ are the matrices of inertia, centripetal and Coriolis forces, and gravitational forces, respectively, $\mathbf{u}_d$ is the disturbance vector, m_1, m_2, l_1, l_2 and l_{c_1}, l_{c_2} are the masses, lengths and center-of-gravity lengths of the two robot links. The parameters of the two-link manipulator are as follows: $m_1 = 4$kg, $m_2 = 2$kg, $l_1 = 1$m, $l_2 = 0.5$m, $l_{c_1} = 0.5$m, $l_{c_2} = 0.25$m.

In this section, our objective is to track exactly the smooth trajectory $\mathbf{x}_d(t)$ given by

$$\mathbf{x}_d(t) = \begin{bmatrix} \sin(0.05\pi t) \\ \sin(0.1\pi t) \end{bmatrix} \tag{40}$$

The unknown external disturbances $\mathbf{u}_d(t)$ are assumed to be governed by

$$\mathbf{u}_d(t) = \begin{bmatrix} 2\sin(0.1\pi t) \\ 2\cos(0.2\pi t) \end{bmatrix} \tag{41}$$

and the initial conditions are set as $\mathbf{z}(0) = [0.5, -0.5, 0, 0]^T$.

Accordingly, the design parameters of the proposed AS-RBFNC are chosen as follows: $\mathbf{K} = diag\,(10, 10)$, $\mathbf{\Lambda}_1 = diag(1, 1)$, $\mathbf{\Lambda}_1 = diag(3, 3)$, $\epsilon = 0.8, \varsigma = 0.1$, $\eta_f = 100, \eta_{g_j} = 10, \eta_c = 5, \eta_\sigma = 5$, $\boldsymbol{\sigma}_{\text{init}} = [2, 2, 2, 2]^T$.

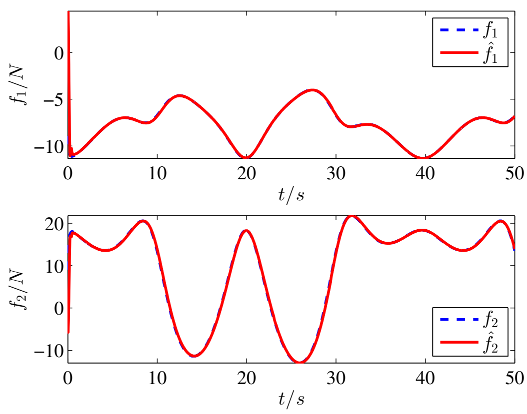

Fig. 6. Nonlinear dynamics f_1, f_2 and approximation $\widehat{f}_1$, $\widehat{f}_2$.

The actual and reference trajectories in the phase space are shown in Fig. 2, from which we can see that the proposed AS-RBFNC control system can track the desired trajectory with high accuracy. The states $\mathbf{x}$ and $\dot{\mathbf{x}}$ together with their desired targets are shown in Fig. 3 and Fig. 4, respectively, from which we can see that the actual states are able to track the desired ones with rapid transient responses and

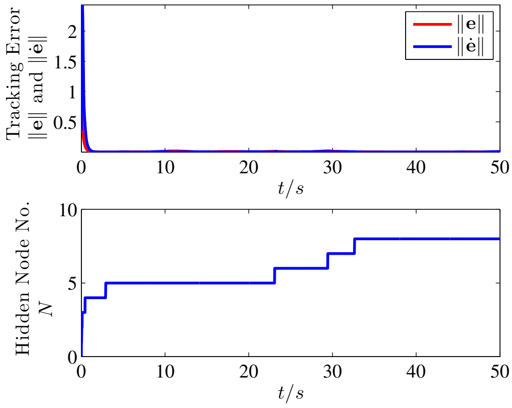

Fig. 7. Online tracking errors and corresponding fuzzy rule numbers.

high steady-state accuracy. The corresponding control forces and torque $\mathbf{u} = [u_1, u_2]^T$ from the AS-RBFNC are shown in Fig. 5, which shows that the smooth control actions dynamically vary with the unknown nonlinear dynamics $\mathbf{f}$ strongly disturbed by the external signal $\mathbf{d}(t)$. The remarkable control performance of the AS-RBFNC actually results from the online approximation ability of the SC-RBFNN which is shown in Fig. 6. Moveover, the online tracking errors $\|\mathbf{e}\|, \|\dot{\mathbf{e}}\|$ and hidden node number $N(t)$ are shown Fig. 7, which shows that the SC-RBFNN with compact RBF hidden nodes guarantees convergent tracking errors and high approximation accuracy simultaneously.

VI. Conclusions

In this paper, we have proposed an adaptive self-constructing RBF neural control (AS-RBFNC) scheme for trajectory tracking of MIMO nonlinear systems in the p-resence of system uncertainties and unknown time-varying disturbances. In the AS-RBFNC, system uncertainties and unknown dynamics can be identified online by a self-constructing RBF neural network (SC-RBFNN) which is implemented by employing dynamically constructive hidden nodes according to the structure learning criteria including hidden node generating and pruning. Moreover, it has been proven that the tracking errors of the AS-RBFNC control system are globally asymptotical stable. Finally, simulation studies are conducted on a two-link robot manipulator. The results demonstrate that the AS-RBFNC achieves remarkably superior performance of both trajectory tracking and online approximation.

References

[1] S. C. Tong and H. X. Li, "Fuzzy adaptive sliding-mode control for MIMO nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 3, pp. 354–360, 2003.

[2] T. S. Li, S. C. Tong and G. Feng, "A novel robust adaptive-fuzzy-tracking control for a class of nonlinear multi-input/multi-output systems," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 1, pp. 150–160, 2010.

[3] Y. J. Liu, W. Wang, S. C. Tong and Y. S. Liu, "Robust adaptive tracking control for nonlinear systems based on bounds of fuzzy approximation parameters," *IEEE Trans. Syst. Man Cybern. A, Syst. Humans*, vol. 40, no. 1, pp. 170–184, 2010.

[4] S. S. Ge and C. Wang, "Direct adaptive NN control of a class of nonlinear systems," *IEEE Trans. Neural Netw.*, vol. 13, no. 1, pp. 214–221, 2002.

[5] C. Wang and D. J. Hill, "Learning from neural control," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 130–146, 2006.

[6] W. Y. Wang, M. L. Chan, C. C. J. Hsu and T. T. Lee, "H_∞ tracking-based sliding mode control for uncertain nonlinear systems via an adaptive fuzzy-neural approach," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 32, no. 4, pp. 483–492, 2002.

[7] Z. Liu and C. W. Li, "Fuzzy neural networks quadratic stabilization output feedback control for biped robots via H_∞ approach," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 33, no. 1, pp. 67–84, 2003.

[8] F. C. Sun, H. X. Li and H. P. Liu, "Neuron-fuzzy dynamic-inversion-based adaptive control for robotic manipulators - discrete time case," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1342–1351, 2007.

[9] C. M. Lin and C. F. Hsu, "Supervisory recurrent fuzzy neural network control of wing rock for slender delta wings," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 5, pp. 733–742, 2004.

[10] W. Yu and X. Li, "Fuzzy identification using fuzzy neural networks with stable learning algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 3, pp. 411–420, 2004.

[11] W. Y. Wang, Y. H. Chien, Y. G. Leu and T. T. Lee, "Adaptive T-S fuzzy-neural modeling and control for general MIMO unkown noaffine systems using prjection update laws," *Automatica*, vol. 46, no. 5, pp. 852–863, 2010.

[12] Y. P. Pan, M. J. Er, D. P. Huang and Q. R. Wang, "Adaptive fuzzy control with guaranteed convergence of optimal approximation error," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 5, pp. 807–818, 2011.

[13] J. A. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*. Hoboken NJ: Wiley, 2006.

[14] N. Wang, M. J. Er, and X.-Y. Meng, "A fast and accurate online self-organizing scheme for parsimonious fuzzy neural networks," *Neurocomputing*, vol. 72, no. 16, pp. 3818–3829, 2009.

[15] N. Wang, M. J. Er, X.-Y. Meng, and X. Li, "An online self-organizing scheme for parsimonious and accurate fuzzy neural networks," *Int. J. Neural Syst.*, vol. 20, no. 5, pp. 389–405, 2010.

[16] N. Wang, "A generalized ellipsoidal basis function based online self-constructing fuzzy neural network," *Neural Process. Lett.*, vol. 34, no. 1, pp. 13–37, 2011.

[17] Y. Gao and M. J. Er, "Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 4, pp. 462–477, 2003.

[18] C. F. Hsu, "Self-organizing adaptive fuzzy neural control for a class of nonlinear systems," *IEEE Trans. Neural Netw.*, vol. 18, no. 4, pp. 1232–1241, 2007.

[19] C. S. Chen, "Dynamic structure neural-fuzzy networks for robust adaptive control of robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 55, no. 9, pp. 3402–3414, 2008.

[20] C. S. Chen, "Robust self-organizing neural-fuzzy control with uncertainty observer for MIMO nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 694–706, 2011.