

# Variable Structure Neural Network Based Direct Adaptive Robust Control of Uncertain Systems

Jianming Lian, Yonggon Lee, Scott D. Sudhoff and Stanislaw H. Żak

**Abstract**—Direct adaptive robust state feedback and output feedback controllers are proposed for the output tracking control of a class of uncertain systems with disturbance. The proposed controllers employ self-organizing raised-cosine radial basis function networks, which are capable of determining their structures dynamically, to approximate unknown system dynamics. Radial basis functions of the network can be added or removed on-line in order to ensure the desired tracking accuracy and the computational efficiency simultaneously. The closed-loop systems are characterized by the guaranteed transient response and the final tracking accuracy. The performance of the proposed output feedback controller is illustrated by numerical simulations.

## I. INTRODUCTION

In the controller design, a mathematical model of the dynamical system to be controlled plays an important role. However, it is often infeasible to derive a quality model because of unknown system dynamics and disturbance. Thus, adaptive robust controllers (ARC), which are consisted of adaptive components and robustifying components, have been proposed to deal with uncertain systems as in [1]–[4]. The adaptive component has the learning mechanism that adjusts the controller's parameters automatically by adaptation laws in order to compensate for the effect of uncertainties, while the robustifying component ensures guaranteed controller performance in the presence of compensation error and disturbance. Most of the adaptive robust control strategies have been developed requiring the availability of the system states. In practical applications, this is not always the case. To overcome this problem, output feedback controllers that employ high-gain observers in state feedback implementation have been developed as in [5]. The advantage of using high-gain observers is that we can formulate the control problem in a standard singular perturbation format and then the singular perturbation theory can be applied to the closed-loop system stability analysis. Moreover, the performance of the output feedback controllers utilizing high-gain observers would asymptotically approach the performance of the state feedback controllers [6].

Adaptive components of ARC often involve different types of function approximators such as fuzzy logic systems and neural networks to approximate unknown system dynamics.

This work was supported by the Office of Naval Research Grant N00014-02-1-0623.

Jianming Lian, Scott D. Sudhoff and Stanislaw H. Żak are with the School of Electrical and Computer Engineering, Purdue University, IN 47907, USA. Yonggon Lee is with the Weapons and Systems Engineering Department, United States Naval Academy, Annapolis, MD 21402, USA. [jlilian@purdue.edu](mailto:jlilian@purdue.edu), [yilee@usna.edu](mailto:yilee@usna.edu), [sudhoff@ecn.purdue.edu](mailto:sudhoff@ecn.purdue.edu), [zak@purdue.edu](mailto:zak@purdue.edu).

In particular, Seshagiri and Khalil [5] investigated the use of fixed-structure radial basis function (RBF) networks for function approximation in the adaptive robust output feedback controller design. However, fixed-structure RBF networks, which require off-line network structure determination, is not suitable for on-line operations. In [3], multilayer neural network (MLNN) based adaptive robust control strategies were proposed. Although it is not required to obtain a basis function set off-line for MLNN, it is still necessary to pre-determine the number of hidden neurons. Moreover, compared to MLNNs, RBF networks are characterized by simpler structure, faster computation time and superior adaptive performance. Recently, variable structure RBF networks, whose structures can vary over time have been proposed for on-line function approximation in [7]–[9]. Variable structure RBF networks preserve the advantages of RBF networks and, at the same time, overcome the limitation of fixed-structure RBF networks.

In this paper, novel direct adaptive robust state feedback and output feedback controllers are proposed, where the output feedback controller is constructed by incorporating a high-gain tracking error observer into the state feedback implementation. We employ the self-organizing RBF networks, described in [9], for function approximation, and use the raised-cosine RBF (RCRBF) instead of the commonly used Gaussian RBF (GRBF). Although the GRBF possesses the property of universal approximation, its unbounded support usually results in high computational cost for the network's training and output evaluation. The RCRBF, on the other hand, has the compact support which can significantly reduce computations [9], [10]. For the closed-loop systems driven by the direct adaptive robust controllers, the transient performance and the final tracking accuracy can be guaranteed and specified by the controllers' design parameters.

## II. SYSTEM DESCRIPTION AND PROBLEM STATEMENT

We consider a class of uncertain systems modeled by

$$y^{(n)} = f(y, \dots, y^{(n-1)}) + g(y, \dots, y^{(n-1)})u + d, \quad (1)$$

where  $u \in \mathbb{R}$  is the control input,  $y \in \mathbb{R}$  is the system output,  $d$  models the disturbance,  $f$  and  $g$  are unknown continuous functions with  $g$  bounded away from zero. Without loss of generality, we assume that  $g(x)$  is strictly positive, that is,  $0 < \underline{g} \leq g(x)$ , where  $\underline{g}$  is the lower bound of  $g(x)$ . We also assume that  $|d| \leq d_o$ . Let  $x = [y \ \dots \ y^{(n-1)}]^\top$ . We can

represent the system (1) in a canonical controllable form

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{b}(f(\mathbf{x}) + g(\mathbf{x})u + d), \\ y &= \mathbf{c}\mathbf{x} = [1 \ 0 \ \cdots \ 0] \mathbf{x},\end{aligned}$$

where  $(\mathbf{A}, \mathbf{b})$  is the canonical controllable pair that represents chains of  $n$  integrators.

The control objective is to develop a tracking control strategy that forces the system output  $y$  to track a given signal  $y_d$  that has bounded derivatives up to the  $n$ -th order, that is,  $y_d^{(n)} \in \Omega_y$  with  $\Omega_y$  a compact subset of  $\mathbb{R}$ . We define the desired system state vector  $\mathbf{x}_d$  as  $\mathbf{x}_d = [y_d \ \cdots \ y_d^{(n-1)}]^\top \in \Omega_{x_d}$ , where  $\Omega_{x_d}$  is a compact subset of  $\mathbb{R}^n$ . Let  $e = y - y_d$  denote the output tracking error and let  $\mathbf{e} = \mathbf{x} - \mathbf{x}_d$  denote the system tracking error. The tracking error dynamics can be modeled as

$$\begin{aligned}\dot{\mathbf{e}} &= \mathbf{A}\mathbf{e} + \mathbf{b}(y^{(n)} - y_d^{(n)}) \\ &= \mathbf{A}\mathbf{e} + \mathbf{b}(f(\mathbf{x}) + g(\mathbf{x})u - y_d^{(n)} + d).\end{aligned}\quad (2)$$

To proceed, we define several compact sets that will be used later. Let  $\Omega_{e_0}$  be a compact set of all possible initial tracking errors and let  $c_{e_0} = \max_{\mathbf{e} \in \Omega_{e_0}} \frac{1}{2} \mathbf{e}^\top \mathbf{P}_m \mathbf{e}$ . Choose  $c_e > c_{e_0}$  and define  $\Omega_e = \{\mathbf{e} : \frac{1}{2} \mathbf{e}^\top \mathbf{P}_m \mathbf{e} \leq c_e\}$ . Then we define  $\Omega_x$  as  $\Omega_x = \{\mathbf{x} : \mathbf{x} = \mathbf{e} + \mathbf{x}_d, \mathbf{e} \in \Omega_e, \mathbf{x}_d \in \Omega_{x_d}\}$ . It will be shown later that if  $\mathbf{e}(t_0) \in \Omega_{e_0}$ , then  $\mathbf{e}(t) \in \Omega_e$  and  $\mathbf{x}(t) \in \Omega_x$  for  $t \geq t_0$ .

Consider the following controller,

$$u_a = \frac{1}{\hat{g}(\mathbf{x})} \left( -\hat{f}(\mathbf{x}) + y_d^{(n)} - \mathbf{k}\mathbf{e} \right), \quad (3)$$

where  $\hat{f}(\mathbf{x})$  and  $\hat{g}(\mathbf{x})$  are approximations of  $f(\mathbf{x})$  and  $g(\mathbf{x})$ , respectively, and  $\mathbf{k}$  is selected such that  $\mathbf{A}_m = \mathbf{A} - \mathbf{b}\mathbf{k}$  is Hurwitz. The controller  $u_a$  in (3) consists of a feedforward term  $-\hat{f}(\mathbf{x}) + y_d^{(n)}$  for model compensation and a linear feedback term  $-\mathbf{k}\mathbf{e}$  for stabilization. Substituting (3) into (2), we obtain

$$\dot{\mathbf{e}} = \mathbf{A}_m \mathbf{e} + \mathbf{b}\tilde{d}, \quad (4)$$

where  $\tilde{d} = (f(\mathbf{x}) - \hat{f}(\mathbf{x})) + (g(\mathbf{x}) - \hat{g}(\mathbf{x}))u_a + d$ . It follows from (4) that if we only apply  $u_a$ , the tracking error does not converge to zero if  $\tilde{d}$  is present. Therefore, a robustifying component  $u_s$  is needed to ensure the tracking performance in the presence of approximation errors and disturbance. The structure of the robustifying component will be given in Section IV. In the following, we first briefly introduce the self-organizing RBF network proposed in [9], which will be employed herein to approximate  $f(\mathbf{x})$  and  $g(\mathbf{x})$  over the compact set  $\Omega_x$ .

### III. SELF-ORGANIZING RAISED-COSINE RBF NETWORK

The self-organizing RBF networks used to approximate  $f(\mathbf{x})$  and  $g(\mathbf{x})$  have similar structures. In Fig. 1, we show the structure of the network used to approximate  $f(\mathbf{x})$ . This network consists of  $n$  input neurons,  $M_f$  hidden neurons, and one output neuron. The number of hidden neurons  $M_f$

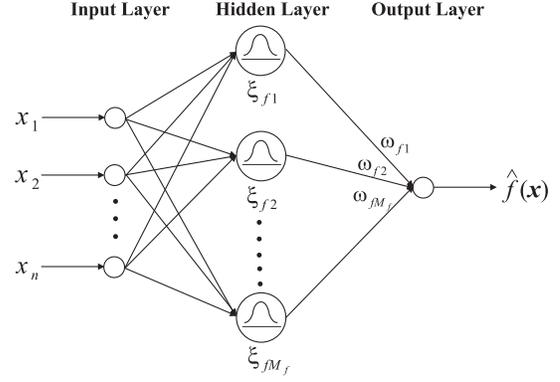


Fig. 1. Radial basis function network for the approximation of  $f(\mathbf{x})$ .

can either increase or decrease over time. For a given input  $\mathbf{x} = [x_1 \ \cdots \ x_n]^\top$ , the output of the network is

$$\begin{aligned}\hat{f}(\mathbf{x}) &= \sum_{j=1}^{M_f} \omega_{fj} \xi_{fj}(\mathbf{x}; \mathbf{c}_{(j)}, \boldsymbol{\delta}_{(j)}) \\ &= \sum_{j=1}^{M_f} \omega_{fj} \prod_{i=1}^n \psi \left( \frac{|x_i - c_{i(j)}|}{\delta_{i(j)}} \right),\end{aligned}\quad (5)$$

where  $\omega_{fj}$  is the adjustable weight from the  $j$ -th hidden neuron to the output neuron. In the following, we use the notation,  $\xi_{fj}(\mathbf{x}) = \xi_{fj}(\mathbf{x}; \mathbf{c}_{(j)}, \boldsymbol{\delta}_{(j)})$ , to denote the radial basis function located at the  $j$ -th hidden neuron. The vector  $\mathbf{c}_{(j)} = [c_{1(j)} \ \cdots \ c_{n(j)}]$  is the center of  $\xi_{fj}(\mathbf{x})$ , the parameter  $\delta_{i(j)}$ ,  $i = 1, \dots, n$ , is the radius or the width of  $\xi_{fj}(\mathbf{x})$  in the  $i$ -th coordinate. Finally,  $\psi : [0, \infty) \rightarrow \mathbb{R}^+$  is the activation function, which characterizes the shape of the RBF, where  $\mathbb{R}^+$  is the set of non-negative real numbers. Thus, we have  $\hat{f}(\mathbf{x}) = \boldsymbol{\omega}_f^\top \boldsymbol{\xi}_f(\mathbf{x})$ , where  $\boldsymbol{\omega}_f = [\omega_{f1} \ \cdots \ \omega_{fM_f}]^\top$  and  $\boldsymbol{\xi}_f(\mathbf{x}) = [\xi_{f1}(\mathbf{x}) \ \cdots \ \xi_{fM_f}(\mathbf{x})]^\top$ . We employ the raised-cosine RBF (RCRBF) whose activation function  $\psi$  is

$$\psi = \begin{cases} \frac{1}{2} \left( 1 + \cos \left( \frac{\pi(x_i - c_{i(j)})}{\delta_{i(j)}} \right) \right) & \text{if } |x_i - c_{i(j)}| \leq \delta_{i(j)} \\ 0 & \text{if } |x_i - c_{i(j)}| > \delta_{i(j)}. \end{cases}$$

The advantage of the RCRBF over the GRBF is the property of the compact support associated with the raised-cosine RBF. The compact support of the RCRBF enables fast and efficient training and output evaluation of the network. This feature becomes especially important when the center grid becomes finer and finer and the dimension of the network input becomes higher and higher [9]. The self-organizing RBF network proposed in [9] will be capable of determining the parameters  $M_f$ ,  $\mathbf{c}_{(j)}$  and  $\boldsymbol{\delta}_{(j)}$  by itself. A brief description of this self-organizing RBF network is given in the following subsections.

#### A. Center Grid Concept

Let  $f(\mathbf{x}) : \Omega_x \rightarrow \mathbb{R}$  be the unknown continuous function to be approximated, where  $\Omega_x \subset \mathbb{R}^n$  is a compact set. We

assume that  $\Omega_x$  can be represented as

$$\begin{aligned}\Omega_x &= \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u\} \\ &= \{\mathbf{x} \in \mathbb{R}^n : x_{li} \leq x_i \leq x_{ui}, 1 \leq i \leq n\},\end{aligned}$$

where the  $n$ -dimensional vectors  $\mathbf{x}_l$  and  $\mathbf{x}_u$  denote lower and upper bounds of  $\mathbf{x}$ , respectively. We next cover  $\Omega_x$  with an initial  $n$ -dimensional grid,  $\{x_{l1}, x_{u1}\} \times \cdots \times \{x_{ln}, x_{un}\}$ . The above boundary nodes are fixed throughout the operation of the direct adaptive robust controller. However, additional nodes within this initial grid can be added, or removed, as the system evolves in time. We propose an algorithm for adding and removing subsequent grid nodes at the potential locations. Each grid node corresponds to the center of one RBF. That is, RBFs cannot be placed arbitrarily inside  $\Omega_x$  but only at the grid nodes. The potential grid nodes are determined coordinate-wise. In each coordinate, the potential grid nodes of the first layer are the two fixed boundary nodes. The second layer has only one potential grid node in the middle of the boundary nodes. Then the potential grid nodes of the subsequent layers are in the middle of the adjacent potential grid nodes of all the previous layers.

### B. Adding RBFs

As the system trajectory evolves in time, the output tracking error  $e$  is measured. If the magnitude of  $e$  exceeds a predetermined threshold  $e_{max}$ , and if the period between the two adding operations is greater than the minimum response time  $T_a$ , where  $e_{max}$  and  $T_a$  are design parameters, two neighboring centers, in the sense of the  $\infty$ -norm, to the current input,  $\mathbf{c}_{(nearest)}$  and  $\mathbf{c}_{(nearer)}$ , among the existing RBFs are determined. That is, each component of  $\mathbf{c}_{(nearest)}$  and  $\mathbf{c}_{(nearer)}$  is determined coordinate-wise. Next, new RBFs are added if necessary. This adding operation is performed sequentially, one coordinate at a time, for all  $n$  coordinates. We now describe in detail the RBF adding operation for one coordinate. First, the distance  $d_i$  between  $x_i$  and  $c_{i(nearrest)}$  is determined. If this distance is smaller than  $\frac{1}{4}$  of the distance between  $c_{i(nearrest)}$  and  $c_{i(nearer)}$ , we do not add any grid nodes in the  $i$ -th coordinate. Otherwise, if the distance  $d_i$  exceeds the prescribed threshold  $d_{i(threshold)}$ , where  $d_{i(threshold)}$  is a design parameter that specifies the minimum grid distance in the  $i$ -th coordinate, a new grid node with its  $i$ -th coordinate equal to half of the sum of  $c_{i(nearrest)}$  and  $c_{i(nearer)}$  is added in this coordinate. Thus, we add  $M_{f1} \times \cdots \times M_{f(i-1)} \times M_{f(i+1)} \times \cdots \times M_{fn}$  new RBFs with initialized weights to the existing RBFs, where  $M_{fi}$  is the number of existing grid nodes in the  $i$ -th coordinate.

### C. Removing RBFs

The RBF removing operation is also implemented sequentially for all  $n$  coordinates. First, the output tracking error  $e$  is measured, and the nearest center  $\mathbf{c}_{(nearest)}$ , in the sense of the  $\infty$ -norm, to the current input is found from the existing RBFs. Then in the  $i$ -th coordinate, the grid node with its  $i$ -th coordinate equal to  $c_{i(nearrest)}$  is determined. Several conditions must be satisfied before this grid node is removed. First, the magnitude of  $e$  must be smaller than  $\tau e_{max}$ , where

$\tau \in (0, 1]$  is a design parameter. Second,  $c_{i(nearrest)}$  should not be equal to the lower bound  $x_{li}$  or the upper bound  $x_{ui}$  of the  $i$ -th coordinate. Third, this grid node has been present in the network for long enough time so that the first condition has been satisfied for the time duration  $T_d$ , where  $T_d$  is a design parameter. Finally, this grid node must be in the higher than or in the same layer as the highest layer of the two neighboring grid nodes in the  $i$ -th coordinate.

## IV. STATE FEEDBACK CONTROLLER DEVELOPMENT

The proposed direct adaptive robust state feedback controller (DARSFC) has the form

$$u = \frac{1}{\hat{g}(\mathbf{x})} \left( -\hat{f}(\mathbf{x}) + y_d^{(n)} - \mathbf{k}e \right) + u_s, \quad (6)$$

where  $\hat{f}(\mathbf{x}) = \boldsymbol{\omega}_f^\top \boldsymbol{\xi}_f(\mathbf{x})$ ,  $\hat{g}(\mathbf{x}) = \boldsymbol{\omega}_g^\top \boldsymbol{\xi}_g(\mathbf{x})$  and  $u_s$  will be defined later. For practical implementation, we constrain  $\boldsymbol{\omega}_f$  and  $\boldsymbol{\omega}_g$ , respectively, to reside in compact sets  $\Omega_f$  and  $\Omega_g$ , respectively, where

$$\begin{aligned}\Omega_f &= \{\boldsymbol{\omega}_f : \underline{\omega}_f \leq \omega_{fj} \leq \bar{\omega}_f, 1 \leq j \leq M_f\}, \\ \Omega_g &= \{\boldsymbol{\omega}_g : 0 < \underline{\omega}_g \leq \omega_{gj} \leq \bar{\omega}_g, 1 \leq j \leq M_g\},\end{aligned}$$

where  $\underline{\omega}_f, \bar{\omega}_f, \underline{\omega}_g$  and  $\bar{\omega}_g$  are design parameters and fixed for different  $\boldsymbol{\omega}_f$  and  $\boldsymbol{\omega}_g$  when the structures of the SORBFNs change. Let  $\boldsymbol{\omega}_f^*$  and  $\boldsymbol{\omega}_g^*$  be ‘‘optimal’’ constant weight vectors of the SORBFNs with respect to a given structures such that

$$\begin{aligned}\boldsymbol{\omega}_f^* &= \operatorname{argmin}_{\boldsymbol{\omega}_f \in \Omega_f} \max_{\mathbf{x} \in \Omega_x} |f(\mathbf{x}) - \boldsymbol{\omega}_f^\top \boldsymbol{\xi}_f(\mathbf{x})|, \\ \boldsymbol{\omega}_g^* &= \operatorname{argmin}_{\boldsymbol{\omega}_g \in \Omega_g} \max_{\mathbf{x} \in \Omega_x} |g(\mathbf{x}) - \boldsymbol{\omega}_g^\top \boldsymbol{\xi}_g(\mathbf{x})|.\end{aligned}$$

It is obvious that  $\boldsymbol{\omega}_f^*$  and  $\boldsymbol{\omega}_g^*$  vary when the structures of the self-organizing RCRBF networks change. We assume that

$$\begin{aligned}\max \left( \max_{\mathbf{x} \in \Omega_x} |f(\mathbf{x}) - \boldsymbol{\omega}_f^{*\top} \boldsymbol{\xi}_f(\mathbf{x})| \right) &\leq d_f, \\ \max \left( \max_{\mathbf{x} \in \Omega_x} |g(\mathbf{x}) - \boldsymbol{\omega}_g^{*\top} \boldsymbol{\xi}_g(\mathbf{x})| \right) &\leq d_g,\end{aligned}$$

where  $\max(\bullet)$  denotes the maximum of  $\bullet$  taken over all the structures of the self-organizing RCRBF networks. In the following, the same outer maximization is used. Let  $\phi_f = \boldsymbol{\omega}_f - \boldsymbol{\omega}_f^*$  and  $\phi_g = \boldsymbol{\omega}_g - \boldsymbol{\omega}_g^*$ . Define

$$\begin{aligned}c_f &= \max \left( \max_{\boldsymbol{\omega}_f, \boldsymbol{\omega}_f^* \in \Omega_f} \frac{1}{2\eta_f} \phi_f^\top \phi_f \right), \\ c_g &= \max \left( \max_{\boldsymbol{\omega}_g, \boldsymbol{\omega}_g^* \in \Omega_g} \frac{1}{2\eta_g} \phi_g^\top \phi_g \right),\end{aligned}$$

where  $\eta_f$  and  $\eta_g$  are positive design parameters. Let  $\sigma = \mathbf{b}^\top \mathbf{P}_m e$ , where  $\mathbf{P}_m$  is the solution to the continuous Lyapunov matrix equation  $\mathbf{A}_m^\top \mathbf{P}_m + \mathbf{P}_m \mathbf{A}_m = -2\mathbf{Q}_m$  for  $\mathbf{Q}_m = \mathbf{Q}_m^\top > 0$ . We employ the following weight vector adaptation laws

$$\dot{\boldsymbol{\omega}}_f = \operatorname{Proj}_{\Omega_f} (\eta_f \sigma \boldsymbol{\xi}_f(\mathbf{x})), \quad (7)$$

$$\dot{\boldsymbol{\omega}}_g = \operatorname{Proj}_{\Omega_g} (\eta_g \sigma \boldsymbol{\xi}_g(\mathbf{x}) u_a), \quad (8)$$

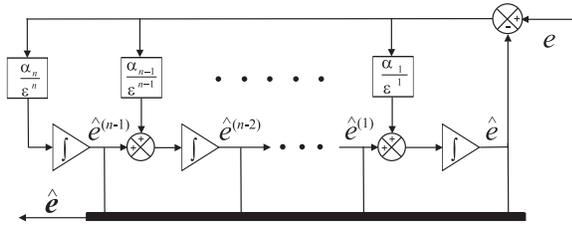


Fig. 2. High-gain observer.

where  $\text{Proj}_{\omega}(\bullet)$  denotes  $\text{Proj}_{\omega_i}(\bullet_i)$  for  $i = 1, \dots, M$  and

$$\text{Proj}_{\omega_i}(\bullet_i) = \begin{cases} 0 & \text{if } \omega_i = \underline{\omega} \text{ and } \bullet_i < 0 \\ 0 & \text{if } \omega_i = \bar{\omega} \text{ and } \bullet_i > 0 \\ \bullet_i & \text{otherwise,} \end{cases}$$

is a discontinuous projection operator proposed in [11]. The above adaptation laws guarantee that  $\omega_f(t) \in \Omega_f$  and  $\omega_g(t) \in \Omega_g$  for  $t \geq t_0$  if  $\omega_f(t_0) \in \Omega_f$  and  $\omega_g(t_0) \in \Omega_g$ .

*Theorem 1:* Consider the system (1) driven by the proposed DARSFC (6) with (7), (8) and

$$u_s = -\frac{1}{g} k_s \text{sat} \left( \frac{\sigma}{\mu} \right), \quad (9)$$

where  $k_s = (d_f + d_g|u_a| + d_o)$  and  $\text{sat}(\bullet)$  is the saturation function with  $\mu > 0$ .

(i) For  $t \geq t_0$ , we have

$$\frac{1}{2} e(t)^\top P_m e(t) \leq \exp(-2\mu_1(t - t_0)) c_{e_0} + \left( c_f + c_g + \frac{k_s}{8\mu_1} \mu \right), \quad (10)$$

where  $\mu_1 = \lambda_{\min}(Q_m)/\lambda_{\max}(P_m)$ . Furthermore, if we choose large  $\eta_f$  and  $\eta_g$  and small  $\mu$  such that  $c_f + c_g + k_s\mu/8\mu_1 \leq c_e - c_{e_0}$ , we have  $e(t) \in \Omega_e$ , that is,  $x(t) \in \Omega_x$  for  $t \geq t_0$ .

(ii) If  $d = 0$ ,  $f(x) = \omega_f^\top \xi_f(x)$  and  $g(x) = \omega_g^\top \xi_g(x)$ , then  $e(t)$ ,  $\phi_f(t)$  and  $\phi_g(t)$  are bounded, and asymptotically tracking is achieved, that is,  $\lim_{t \rightarrow \infty} e(t) = 0$ .

*Proof:* See [12]  $\blacksquare$

## V. OUTPUT FEEDBACK CONTROLLER CONSTRUCTION

The DARSFC presented in the previous section requires the availability of the system states. However, it is often in practice that only system outputs are available. Thus, it is desirable to develop a direct adaptive robust output feedback controller (DAROFc) architecture. To achieve this, we apply the following high-gain tracking error observer [5], [6],

$$\dot{\hat{e}} = A\hat{e} + l(e - c\hat{e}). \quad (11)$$

The observer gain  $l$  is chosen as  $l = [\alpha_1/\varepsilon \ \dots \ \alpha_n/\varepsilon^n]^\top$ , where  $\varepsilon$  is a design parameter such that  $0 < \varepsilon < 1$  and  $\alpha_i$ ,  $i = 1, \dots, n$ , are selected so that the roots of the polynomial equation,  $s^n + \alpha_1 s^{n-1} + \dots + \alpha_{n-1} s + \alpha_n = 0$ , have negative real parts. In Fig. 2, we show the structure of the above high-gain observer. In order to eliminate the peaking phenomena that accompany the above high-gain

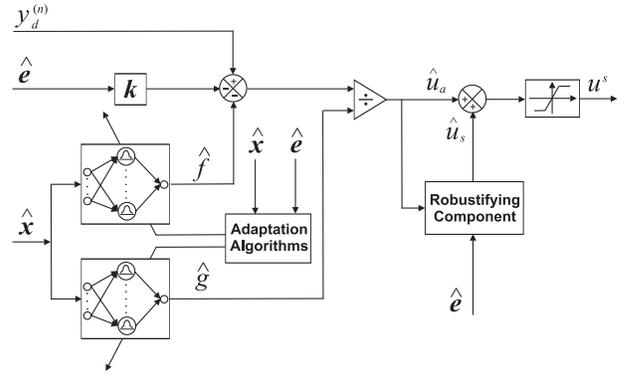


Fig. 3. Diagram of the output feedback controller.

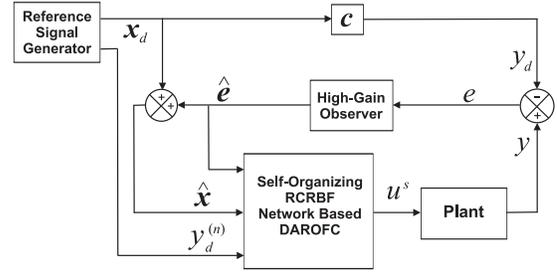


Fig. 4. The closed-loop system driven by the output feedback controller.

tracking error observer [13], we introduce the saturation of the control input. Define  $\Omega_{\hat{e}} = \{e : \frac{1}{2} e^\top P_m e \leq c_{\hat{e}}\}$ , where  $c_{\hat{e}} > c_e$ . Let  $S \geq \max(\max |u(\hat{e}, x_d, y_d^{(n)}, \omega_f, \omega_g)|)$ , where  $u$  is defined in (6) and the inner maximization is taken over  $\hat{e} \in \Omega_{\hat{e}}$ ,  $x_d \in \Omega_{x_d}$ ,  $y_d^{(n)} \in \Omega_{y_d}$ ,  $\omega_f \in \Omega_f$  and  $\omega_g \in \Omega_g$ . Then the the DAROFc takes the following form

$$u^s = S \text{sat} \left( \frac{\hat{u}_a + \hat{u}_s}{S} \right), \quad (12)$$

where

$$\hat{u}_a = \frac{1}{\hat{g}(\hat{x})} \left( -\hat{f}(\hat{x}) + y_d^{(n)} - k\hat{e} \right),$$

$$\hat{u}_s = -\frac{1}{\hat{g}} \hat{k}_s \text{sat} \left( \frac{\hat{\sigma}}{\mu} \right),$$

with  $\hat{k}_s = (d_f + d_g|\hat{u}_a| + d_o)$  and  $\hat{\sigma} = b^\top P_m \hat{e}$ . The adaptation laws for the weight vectors  $\omega_f$  and  $\omega_g$  become

$$\dot{\omega}_f = \text{Proj}_{\omega_f} (\eta_f \hat{\sigma} \xi_f(\hat{x})), \quad (13)$$

$$\dot{\omega}_g = \text{Proj}_{\omega_g} (\eta_g \hat{\sigma} \xi_g(\hat{x}) \hat{u}_a). \quad (14)$$

A block diagram of this self-organizing RCRBF network based robust adaptive controller is shown in Fig. 3. A block diagram of the closed-loop system is given in Fig. 4.

In order to facilitate the stability analysis of the closed-loop system, we cast the control problem into a standard singular perturbation form. Let  $\zeta = [\zeta_1 \ \dots \ \zeta_n]^\top$ , where

$$\zeta_i = \frac{e^{(i-1)} - \hat{e}^{(i-1)}}{\varepsilon^{n-i}}, \quad i = 1, \dots, n. \quad (15)$$

It follows from (15) that  $e - \hat{e} = D(\varepsilon)\zeta$ , where  $D(\varepsilon) = \text{diag}[\varepsilon^{n-1} \ \varepsilon^{n-2} \ \dots \ 1]$ . Note that the induced Euclidian

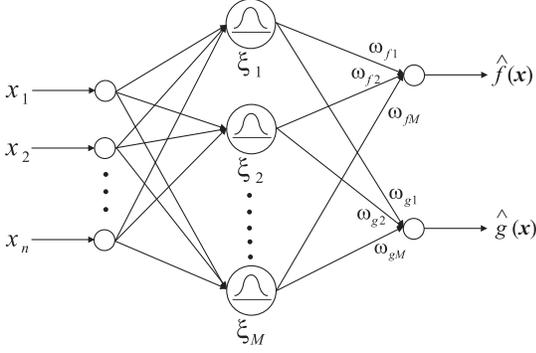


Fig. 5. RCRBF network used to approximate  $f(\mathbf{x})$  and  $g(\mathbf{x})$ .

norm of  $\mathbf{D}(\varepsilon)$  is 1, that is,  $\|\mathbf{D}(\varepsilon)\| = 1$ . It follows from (2) and (11) that

$$\varepsilon \dot{\zeta} = \mathbf{A}_c \zeta + \varepsilon \mathbf{b} \left( f(\mathbf{x}) + g(\mathbf{x})u^s - y_d^{(n)} + d \right), \quad (16)$$

where  $\mathbf{A}_c = \varepsilon \mathbf{D}(\varepsilon)^{-1}(\mathbf{A} - \mathbf{l}c)\mathbf{D}(\varepsilon)$  is Hurwitz. Applying the method in [14], we can prove the following result.

*Proposition 1:* There exist a constant  $\varepsilon_1^*$  ( $0 < \varepsilon_1^* < 1$ ), and a finite time  $T_1$  such that for  $\varepsilon < \varepsilon_1^*$ ,  $\|\zeta(t)\| \leq \beta\varepsilon$  for some  $\beta > 0$  and  $t \in [t_0 + T_1, t_0 + T_3)$ , where  $t_0 + T_3$  is the time when the trajectory of the tracking error  $e(t)$  leaves the set  $\Omega_e$  for the first time.

*Theorem 2:* Consider the system (1) driven by the proposed DAROFC (12) with (13), (14) and the high-gain observer (11). Let  $c_{e_1} = \frac{1}{2}e(t_0 + T_1)^\top \mathbf{P}_m e(t_0 + T_1)$ . If we choose large  $\eta_f$  and  $\eta_g$  and small  $\mu$  such that  $c_f + c_g + \hat{k}_s \mu / 8\mu_1 < c_e - c_{e_1}$ , there exists a constant  $\varepsilon^*$  ( $0 < \varepsilon^* < 1$ ) such that for  $\varepsilon < \varepsilon^*$ , we have

$$\begin{aligned} \frac{1}{2}e(t)^\top \mathbf{P}_m e(t) &\leq \exp(-2\mu_1(t - t_0 - T_1))c_{e_1} \\ &+ \left( c_f + c_g + \frac{\hat{k}_s}{8\mu_1}\mu + r\varepsilon \right) \end{aligned} \quad (17)$$

for some  $r > 0$  and  $t \geq t_0 + T_1$ , and  $e(t) \in \Omega_e$ , that is,  $\mathbf{x}(t) \in \Omega_x$  for  $t \geq t_0$ .

*Proof:* See [12] ■

It can be clearly seen from (10) and (17) that as  $\varepsilon$  approaches zero, the performance of the DAROFC approaches that of the DARSFC.

## VI. EXAMPLES

In this section, a benchmark problem from the literature is used to illustrate the features of the proposed DAROFC. In Example 1, the controller performance is tested with the white noise disturbance. In Example 2, the desired output signal changes during the operation in order to demonstrate the advantage of using self-organizing RBF network. For both examples, one self-organizing RCRBF network with two outputs is used to approximate  $f$  and  $g$ , whose structure is shown in Fig. 5.

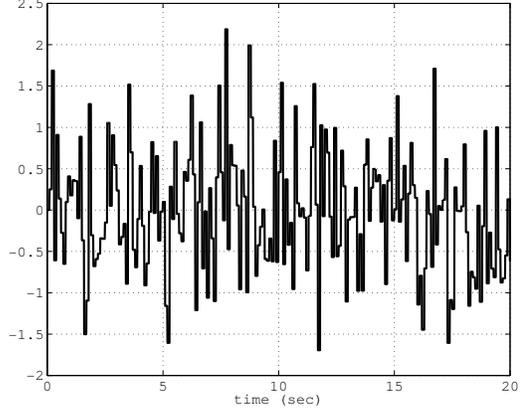


Fig. 6. Disturbance  $d$  in Example 1.

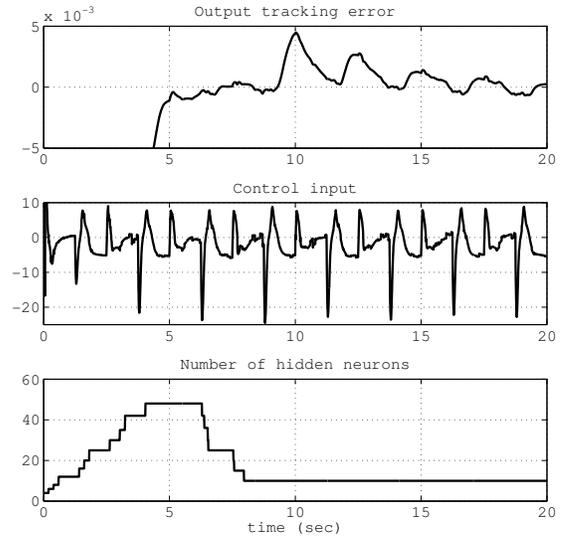


Fig. 7. DAROFC performance in Example 1.

*Example 1:* The nonlinear plant model used in this example is given by

$$\ddot{y} = 4 \frac{\sin(4\pi y)}{\pi y} \left( \frac{\sin(\pi \dot{y})}{\pi \dot{y}} \right)^2 + (2 + \sin(3\pi(y - 0.5)))u + d,$$

which, if  $d = 0$ , is the same plant model used as a testbed for proposed controllers as in [1], [5]. The disturbance  $d$  is selected to be band-limited white noise generated using SIMULINK (ver. 6.6) with noise power 0.05, sample time 0.1 and seed value 23341. A plot of this disturbance signal versus time is shown in Figure 6.

The reference signal is the same as in [5], which is the output of a low-pass filter with the transfer function  $(1 + 0.1s)^{-3}$ , driven by a unity amplitude square wave input with frequency of 0.4 Hz and a time average of 0.5. Thus, the grid boundaries for  $y$  and  $\dot{y}$ , respectively, are selected to be  $[-1.5 \ 1.5]$  and  $[-3.5 \ 3.5]$ , that is,  $\mathbf{x}_l = [-1.5 \ -3.5]^\top$  and  $\mathbf{x}_u = [1.5 \ 3.5]^\top$ . The rest of the network's parameters are

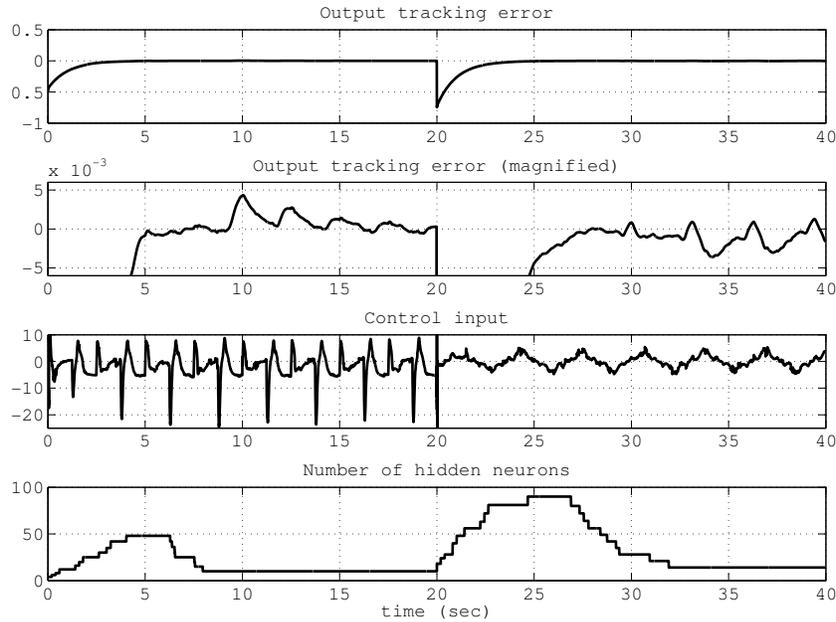


Fig. 8. DAROFC performance with changing reference signals during its operation in Example 2.

$\mathbf{d}_{threshold} = [0.2 \ 0.3]$ ,  $e_{max} = 0.005$ ,  $T_a = 0.2$ ,  $\tau = 0.5$ ,  $T_d = 1.0$ ,  $\bar{\omega}_f = 25$ ,  $\underline{\omega}_f = -25$ ,  $\bar{\omega}_g = 5$ ,  $\underline{\omega}_g = 0.1$  and  $\eta_f = \eta_g = 1000$ . The controller's parameters are  $\mathbf{k} = [1 \ 2]$ ,  $\mathbf{Q}_m = 0.5\mathbf{I}_2$ ,  $d_f = 5$ ,  $d_g = 2$ ,  $d_o = 3$ ,  $\mu = 0.01$  and  $S = 50$ . The observer's parameters are  $\varepsilon = 0.001$ ,  $\alpha_1 = 10$  and  $\alpha_2 = 25$ . The initial conditions are  $y(0) = -0.5$  and  $\dot{y}(0) = 2.0$ . The controller performance in the presence of disturbance is shown in Fig. 7.

*Example 2:* In order to demonstrate the advantages of the self-organizing RBF network in the proposed controller architecture, a different reference signal,  $y_d(t) = \sin(2t)$ , is applied at 20 second. In such a case, we still have  $\Omega_{xd} \subset \Omega_x$  for the new reference signal. It can be seen from Fig. 8 that the self-organizing RCRBF network based DAROFC still performs very well even when the reference signal changes. There is no need to adjust the network's or the controller's parameters off-line for the new reference signal. The self-organizing RBF network determines its structure dynamically as the output tracking error trajectory evolves with time.

## VII. CONCLUSIONS

Novel direct adaptive robust controllers have been proposed for the output tracking control of a class of systems with unknown system dynamics and disturbance. The presented techniques incorporate self-organizing raised-cosine RBF networks that can determine their structures on-line automatically. The structure of the network varies, as the output tracking error trajectory evolves, in order to ensure the tracking accuracy and, at the same time, the computational efficiency. Simulation results illustrate the effectiveness of the proposed direct adaptive robust output feedback controller.

## REFERENCES

- [1] R. M. Sanner and J. J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Netw.*, vol. 3, no. 6, pp. 837–863, Nov. 1992.
- [2] L.-X. Wang, "Stable adaptive fuzzy control of nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp. 146–155, May 1993.
- [3] A. Yesildirek and F. L. Lewis, "Feedback linearization using neural networks," *Automatica*, vol. 31, no. 11, pp. 1659–1664, 1995.
- [4] Y. Lee and S. H. Žak, "Uniformly ultimately bounded fuzzy adaptive tracking controllers for uncertain systems," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 6, pp. 797–811, Dec. 2004.
- [5] S. Seshagiri and H. K. Khalil, "Output feedback control of nonlinear systems using RBF neural networks," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 69–79, Jan. 2000.
- [6] H. K. Khalil, "Adaptive output feedback control of nonlinear systems represented by input–output models," *IEEE Trans. Autom. Control*, vol. 41, no. 2, pp. 177–188, Feb. 1996.
- [7] G. P. Liu, V. Kadiramanathan, and S. A. Billings, "Variable neural networks for adaptive control of nonlinear systems," *IEEE Trans. Syst., Man, Cybern. B*, vol. 39, no. 1, pp. 34–43, Feb. 1999.
- [8] Y. Lee, S. Hui, E. Zivi, and S. H. Žak, "Variable neural adaptive robust controllers for uncertain systems," *Int. J. Adapt. Control Signal Process.*, 2008.
- [9] J. Lian, Y. Lee, S. D. Sudhoff, and S. H. Žak, "Self-organizing radial basis function network for real-time approximation of continuous-time dynamical systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 3, pp. 460–474, Mar. 2008.
- [10] R. J. Schilling, J. J. Carroll, and A. F. Al-Ajlouni, "Approximation of nonlinear systems with radial basis function neural network," *IEEE Trans. Neural Netw.*, vol. 12, no. 1, pp. 1–15, Jan. 2001.
- [11] B. Yao, "Integrated direct/indirect adaptive robust control of SISO nonlinear systems in semi-strict feedback form," in *Proc. American Control Conference*, Denver, CO, 2003, pp. 3020–3025.
- [12] J. Lian, Y. Lee, S. D. Sudhoff, and S. H. Žak, "Variable neural direct adaptive robust control of uncertain systems," *IEEE Trans. Autom. Control*, revised and re-submitted for publication, 2008.
- [13] F. Esfandiari and H. K. Khalil, "Output feedback stabilization of fully linearizable systems," *Int. J. Contr.*, vol. 56, no. 5, pp. 1007–1037, 1992.
- [14] N. A. Mahmoud and H. K. Khalil, "Asymptotic regulation of minimum phase nonlinear systems using output feedback," *IEEE Trans. Autom. Control*, vol. 41, no. 10, pp. 1402–1412, Oct. 1996.