Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Adaptive Backstepping Sliding Mode Control with Gaussian Networks for a Class of Nonlinear Systems with Mismatched Uncertainties

Lei Ma, Klaus Schilling and Christian Schmid

***Abstract*— This paper is concerned with the adaptive sliding-mode control of a class of nonlinear systems in nonlinear parametric-pure-feedback form with mismatched uncertainties. Backstepping design procedure is applied, which leads to a new adaptive sliding-mode control. Gaussian radial-basis-function networks are used to approximate the unknown system dynamics. A new growing scheme of the Gaussian networks is proposed. The networks start with a loose structure in order to reduce the computational effort. More nodes are added to the networks progressively in order to improve the transient behaviour. With ideal sliding mode, asymptotic stability is reached. The performance of the control scheme is illustrated by simulation studies.**

I. INTRODUCTION

The sliding-mode control (SMC) methodology [1] is widely accepted as a feasible robust control for dynamic systems. As the behaviour of the systems with sliding mode is defined by the synthesis of sliding surfaces, the systems are insensitive to model uncertainties and disturbances. One drawback of SMC is the requirement of the matching condition. As for deterministic robust control, SMC requires that the uncertainties and disturbances can be lumped into the input channel, so that they can be efficiently compensated by the control input [2][3][4]. However, the matching condition is a very strong assumption of the system structure. This seriously clams the application of the SMC method.

The backstepping method [5][6][7] is a breakthrough for adaptive nonlinear control. This provides a systematic procedure to construct a robust control Lyapunov function. The method was firstly applied to the control of dynamic systems in parametric-pure-feedback form (PPF) and in parametric-strict-feedback form (PSF), where local stabilisation was approached for the former and global stabilisation for the latter. Efforts have been made in the last decade to extend the backstepping method for control design of nonlinear dynamic systems in more general forms. Seto *et al.* [8] proposed backstepping control design for systems with a triangular structure. Yao [9] solved the control problem of systems in semi-strict feedback form where external disturbances were taken into consideration. Ge *et al.* [10] investigated the control of systems in the so-called triangular control form. Liu *et al.* [11] studied the control of interconnected MIMO systems using backstepping.

L. Ma and K. Schilling are with the Department of Computer Science VII, Universität Würzburg, D-97074 Würzburg, Germany {ma,schi}@informatik.uni-wuerzburg.de

Chr. Schmid is with the Institute of Automation and Computer Control, Ruhr-Universität Bochum, D-44801 Bochum, Germany cs@atp.rub.de

It is natural to integrate the backstepping algorithm into the design of SMC. This removes the requirement of the matching condition, and the powerful SMC technique can provide robustness of the adaptive systems. Two basic ideas are known from literature: 1) A sliding surface as a linear combination of the control errors is constructed at the final step of the backstepping. A SMC term ensures the convergence of the system states to the sliding surface, so that the control errors also converge [12][13]. 2) Sliding surfaces are constructed in each step of the backstepping, so that the convergence of the system states is progressively approached [14][15][16]. According to the latter method, consider each system state as a virtual controller for its integration, construction of sliding surfaces and sliding control term is required for every state variables.

To solve the control problem of systems not in linear-in-parameter form, neural networks have been utilised for approximation of the unknown dynamics. Knohl and Unbehauen [17][18] extended the backstepping algorithm to a wider class of systems in nonlinear parametric-strict-feedback form (NPSF). Gong and Yao [19][20] proposed neural-network based backstepping control for systems in normal form and in semi-strict feedback form, respectively, and applied these to the precision motion control of a linear motor drive system. Further results in this aspect can also be found in the work of Ge *et al.* [10][21]. Uniform ultimate boundedness of the signals is assured by using a proper updating law of the neural networks.

In this paper, adaptive SMC of systems in nonlinear parametric-pure-feedback form (NPPF) is investigated. Backstepping design is utilised to remove the problem due to mismatched uncertainties. A single sliding surface is constructed in the last step of backstepping. Compared with the method proposed in [12] and [13], the stability analysis and the control law are considerably simplified. Gaussian radial-basis-function (GRBF) networks are applied to approximate the nonlinear system dynamics, which can not be linearly parameterised. The updating law of the networks is obtained by the Lyapunov design. A new network growing scheme proposed in [22] is used to reduce the computational effort. The sliding control term is only required for the real control, which serves for the compensation of the approximation errors of the networks. With ideal sliding mode, asymptotic stability is obtained. The paper is organised as follows: in section II the problem statement is given. The backstepping design of adaptive SMC is described in section III. The stability analysis is naturally integrated into the design procedure. In section IV the feasibility of the proposed control scheme is illustrated by simulation studies. A short conclusion is given in the last section.

II. Problem Statement

The systems considered in this paper have the following nonlinear parametric-pure-feedback form (NPPF):

$$\begin{aligned} \dot{x}_i &= x_{i+1} + f_i(x_1, ..., x_{i+1}, \theta), \qquad 1 \le i \le n-1, \\ \dot{x}_n &= b(x)u + f_n(x), \end{aligned} \tag{1}$$

where $x = [x_1,\ x_2,\ ...,\ x_n]^T$ is the vector of the system states. $f_1,\ f_2,\ ...,\ f_n$ are unknown yet smooth scalar nonlinear functions. θ presents the unknown parameters in the model. The control objective is to track a desired trajectory x_d with x_1. It is assumed that all of the systems states $x_1,\ x_2,\ ...,\ x_n$ and the derivatives of the desired trajectory $\dot{x}_\mathrm{d},\ \ddot{x}_\mathrm{d}, ..., x_\mathrm{d}^{(n)}$ are available for control design.

Using GRBF networks to approximate the unknown functions, (1) can be rewritten as

$$\begin{aligned} \dot{x}_i &= x_{i+1} + W_i\phi_i(x_1, ...,\ x_{i+1}), \qquad 1 \le i \le n-1, \\ \dot{x}_n &= b(x)u + W_n\phi_n, \end{aligned} \tag{2}$$

For the the i-th GRBF network with m_i basis functions, $W_i = [w_{i,1}, ..., w_{i,m_i}]$ is a row vector of the output weights, $\phi_i = [\phi_{i,1}, ..., \phi_{i,m_i}]^T$ is a column vector of the output of the basis functions with

$$\phi_{i,j} = e^{-\frac{1}{2\sigma_{i,j}^2}\sum_{k=1}^{i+1}(x_k - \xi_{i,j}^k)^2}, \qquad 1 \le j \le m_i, \tag{3}$$

where $\sigma_{i,j}$ is the width of the (i,j)-th basis function, $\xi_{i,j}^k$ is the centre of the (i,j)-th basis function with respect to the k-th input x_k.

Using a network arranged on a regular lattice [23] at the beginning, there exists an optimal output-weight vector W_i^* and a positive scalar ε_i^0 such that

$$f_i = W_i^*\phi_i(x_1, ...,\ x_{i+1}) + \varepsilon_i, \qquad |\varepsilon_i| \le \varepsilon_i^0, \tag{4}$$

where ε_i $i = 1, ..., n$ is the approximation error of the i-th GRBF network. Define $\hat{W}_i$ as the estimation of W_i^*, and the estimation error $\tilde{W}_i = W_i^* - \hat{W}_i$, (4) becomes

$$f_i = \hat{W}_i\phi_i + \tilde{W}_i\phi_i + \varepsilon_i, \qquad |\varepsilon_i| \le \varepsilon_i^0. \tag{5}$$

By increasing the amount of neurons in the network, the upper bound of the approximation error ε_i^0 can be reduced to be arbitrarily small.

Let $W = [W_1, ..., W_n]$, and $\Phi_i = [0^T_{(1\times\sum_{j=1}^{i-1} m_j)} \vdots\ \phi_i^T \vdots\ 0^T_{(1\times\sum_{j=i+1}^{n} m_j)}]^T$, (2) is rewritten as

$$\begin{aligned} \dot{x}_i &= x_{i+1} + W\Phi_i(x_1, ...,\ x_{i+1}) + \varepsilon_i, \qquad 1 \le i \le n-1, \\ \dot{x}_n &= b(x)u + W\Phi_n + \varepsilon_n. \end{aligned} \tag{6}$$

III. Adaptive Sliding Mode Control

In this section the backstepping algorithm is applied to design a new adaptive SMC, such that the restrictive matching condition for classical SMC is removed. A sliding surface which determines the error dynamics is constructed in the final step of the backstepping. The control law is also obtained in the final step.

A. Backstepping design procedure

1) Step 0: Define the tracking error

$$z_1 = x_1 - x_\mathrm{d} \tag{7}$$

and positive constants $c_1, ..., c_n,\ g_1, ..., g_n$ to be selected.

2) Step 1: The first derivative of the control error is

$$\begin{aligned} \dot{z}_1 &= \dot{x}_1 - \dot{x}_\mathrm{d} \\ &= x_2 + f_1(x_1, x_2, \theta) - \dot{x}_\mathrm{d}. \end{aligned} \tag{8}$$

Treating x_2 as a control signal for (8), the control law $x_{\mathrm{d}2}$ for x_2 which stabilises z_1 would be

$$x_{2\mathrm{d}} = -c_1z_1 - f_1 + \dot{x}_\mathrm{d}. \tag{9}$$

Since f_1 is unknown, a GRBF network is used for approximation, such that the actual value of x_2 is

$$x_2 = -c_1z_1 - \hat{W}\Phi_1(x_1, x_2) + \dot{x}_\mathrm{d}. \tag{10}$$

Define z_2 as the difference between x_2 and $x_{2\mathrm{d}}$ as

$$z_2 = x_2 + c_1z_1 + \hat{W}\Phi_1 - \dot{x}_\mathrm{d}. \tag{11}$$

Substituting (11) into (8), $\dot{z}_1$ becomes

$$\dot{z}_1 = -c_1z_1 + z_2 + \tilde{W}\Phi_1 + \varepsilon_1. \tag{12}$$

Furthermore, let

$$\alpha_1(x_1, x_2, \hat{W}_1) = -c_1z_1 - \hat{W}\Phi_1 = -c_1z_1 - \hat{W}_1\Phi_1, \tag{13}$$

z_2 is written as

$$z_2 = x_2 - \alpha_1 - \dot{x}_\mathrm{d}. \tag{14}$$

3) Step 2: The derivative of z_2 is

$$\begin{aligned} \dot{z}_2 &= \dot{x}_2 - \dot{\alpha}_1 - \ddot{x}_\mathrm{d} \\ &= x_3 + f_2 - \frac{\partial\alpha_1}{\partial x_1}\dot{x}_1 - \frac{\partial\alpha_1}{\partial x_2}\dot{x}_2 - \frac{\partial\alpha_1}{\partial\hat{W}_1}\dot{\hat{W}}_1^T - \ddot{x}_\mathrm{d} \\ &= x_3 + W\Phi_2 + \varepsilon_2 \\ &\quad -\frac{\partial\alpha_1}{\partial x_1}(x_2 + W_1\Phi_1 + \varepsilon_1) \\ &\quad -\frac{\partial\alpha_1}{\partial x_2}(x_3 + W_2\Phi_2 + \varepsilon_2) - \frac{\partial\alpha_1}{\partial\hat{W}_1}\dot{\hat{W}}_1^T - \ddot{x}_\mathrm{d}. \end{aligned} \tag{15}$$

By ignoring $\frac{\partial\alpha_1}{\partial x_1}\varepsilon_1$ and $\frac{\partial\alpha_1}{\partial x_2}\varepsilon_2$, (15) becomes

$$\begin{aligned} \dot{z}_2 &= x_3 + \varepsilon_2 - \ddot{x}_\mathrm{d} - \frac{\partial\alpha_1}{\partial\hat{W}_1}\dot{\hat{W}}_1^T \\ &\quad + W(\Phi_2 - \sum_{i=1}^{2}\frac{\partial\alpha_1}{\partial x_i}\Phi_i) - \sum_{i=1}^{2}\frac{\partial\alpha_1}{\partial x_i}x_{i+1}. \end{aligned} \tag{16}$$

Let

$$\begin{aligned} \alpha_2 &= -c_2z_2 - z_1 + \frac{\partial\alpha_1}{\partial\hat{W}_1}\dot{\hat{W}}_1^T \\ &\quad -\hat{W}(\Phi_2 - \sum_{i=1}^{2}\frac{\partial\alpha_1}{\partial x_i}\Phi_i) + \sum_{i=1}^{2}\frac{\partial\alpha_1}{\partial x_i}x_{i+1}, \end{aligned} \tag{17}$$

the control law for x_3 to stabilise $\dot{z}_2$ would be

$$x_{3\mathrm{d}} = \alpha_2 + \ddot{x}_\mathrm{d}, \tag{18}$$

and the difference between x_3 and its desired value $x_{3\mathrm{d}}$ is

$$\begin{aligned} z_3 &= x_3 + c_2 z_2 + z_1 - \frac{\partial \alpha_1}{\partial \hat{W}_1}\dot{\hat{W}}_1^T - \ddot{x}_\mathrm{d} \\ &+ \hat{W}(\Phi_2 - \sum_{i=1}^{2}\frac{\partial \alpha_1}{\partial x_i}\Phi_i) - \sum_{i=1}^{2}\frac{\partial \alpha_1}{\partial x_i}x_{i+1}. \quad (19) \end{aligned}$$

Substituting (19) into (15), $\dot{z}_2$ becomes

$$\dot{z}_2 = -c_2 z_2 + z_3 - z_1 + \varepsilon_2 + \tilde{W}(\Phi_2 - \sum_{i=1}^{2}\frac{\partial \alpha_1}{\partial x_i}\Phi_i). \quad (20)$$

4) Step i ($1 \le i \le n-1$): The derivative of z_i is

$$\begin{aligned} \dot{z}_i &= \dot{x}_i - \dot{\alpha}_{i-1} - x_\mathrm{d}^{(i)} \\ &= x_{i+1} + \varepsilon_i - x_\mathrm{d}^{(i)} - \frac{\partial \alpha_{i-1}}{\partial \hat{W}_{i-1}}\dot{\hat{W}}_{i-1}^T \\ &+ W(\Phi_i - \sum_{j=1}^{i}\frac{\partial \alpha_{i-1}}{\partial x_j}\Phi_j) - \sum_{j=1}^{i}\frac{\partial \alpha_{i-1}}{\partial x_j}x_{j+1}. \end{aligned} \quad (21)$$

Let

$$\begin{aligned} \alpha_i &= -c_i z_i - z_{i-1} - \hat{W}(\Phi_i - \sum_{j=1}^{i}\frac{\partial \alpha_{i-1}}{\partial x_j}\Phi_j) \\ &+ \sum_{j=1}^{i}\frac{\partial \alpha_{i-1}}{\partial x_j}x_{j+1} + \frac{\partial \alpha_{i-1}}{\partial \hat{W}_{i-1}}\dot{\hat{W}}_{i-1}^T, \quad (22) \end{aligned}$$

and define z_{i+1} as

$$z_{i+1} = x_{i+1} - \alpha_i - x_d^{(i)}, \quad (23)$$

(21) becomes

$$\dot{z}_i = -c_i z_i + z_{i+1} - z_{i-1} + \varepsilon_i + \tilde{W}\beta_i, \quad (24)$$

with

$$\beta_i = \Phi_i - \sum_{j=1}^{i}\frac{\partial \alpha_{i-1}}{\partial x_j}\Phi_j. \quad (25)$$

5) Step n: Let z_n be the difference between x_n and its desired value $x_{n\mathrm{d}}$, its derivative is

$$\begin{aligned} \dot{z}_n &= b(x)u + \varepsilon_n - x_\mathrm{d}^{(n)} - \sum_{i=1}^{n-1}\frac{\partial \alpha_{n-1}}{\partial x_i}x_{i+1} \\ &+ W(\Phi_n - \sum_{i=1}^{n}\frac{\partial \alpha_{n-1}}{\partial x_i}\Phi_i) + \frac{\partial \alpha_{n-1}}{\partial \hat{W}_{n-1}}\dot{\hat{W}}_{n-1}^T. \quad (26) \end{aligned}$$

Define the sliding surface as

$$s = \sum_{i=1}^{n-1} g_i z_i + z_n, \qquad g_i \ge 0, \quad (27)$$

let

$$u^* = \sum_{i=1}^{n-1} g_i(-c_i z_i + z_{i+1} - z_{i-1}), \quad (28)$$

and

$$u_\mathrm{smc} = -\rho\frac{s}{|s|}, \qquad \rho > \sum_{i=1}^{n-1} g_i \varepsilon_i^0 + \varepsilon_n^0, \quad (29)$$

the main result of this paper is presented in the following theorem:

Theorem 1: For the dynamic system in nonlinear parametric-pure-feedback form (1), the tracking error $x_1 - x_\mathrm{d}$ converges to zero asymptotically with the control

$$\begin{aligned} u &= \frac{1}{b(x)}[x_\mathrm{d}^{(n)} + \sum_{i=1}^{n-1}\frac{\partial \alpha_{n-1}}{\partial x_i}x_{i+1} \\ &- \hat{W}(\Phi_n - \sum_{i=1}^{n}\frac{\partial \alpha_{n-1}}{\partial x_i}\Phi_i) \\ &- \frac{\partial \alpha_{n-1}}{\partial \hat{W}_{n-1}}\dot{\hat{W}}_{n-1}^T - u^* + u_\mathrm{smc}]. \quad (30) \end{aligned}$$

The stability analysis is presented in the next subsection.

B. Stability analysis

Substitute (30) into (26), $\dot{z}_n$ becomes

$$\dot{z}_n = \varepsilon_n + \tilde{W}(\Phi_n - \sum_{i=1}^{n}\frac{\partial \alpha_{n-1}}{\partial x_i}\Phi_i) - u^* + u_\mathrm{smc}. \quad (31)$$

Consider the Lyapunov function candidate

$$V = \frac{1}{2}s^2 + \frac{1}{2}\gamma^{-1}\tilde{W}\tilde{W}^T, \qquad \gamma > 0, \quad (32)$$

the derivative of this function is

$$\dot{V} = s(\sum_{i=1}^{n-1} g_i \dot{z}_i) + s\dot{z}_n + \gamma^{-1}\tilde{W}\dot{\tilde{W}}^T. \quad (33)$$

Substitute (31) into (33) and let $g_n = 1$, it follows that

$$\dot{V} = s[\sum_{i=1}^{n}(g_i\varepsilon_i + \tilde{W}\beta_i) + u_\mathrm{smc} + \gamma^{-1}\tilde{W}\dot{\tilde{W}}^T] \quad (34)$$

with

$$\beta_n = \Phi_n - \sum_{i=1}^{n}\frac{\partial \alpha_{n-1}}{\partial x_i}\Phi_i. \quad (35)$$

Let the updating law of $\hat{W}$ be

$$\dot{\hat{W}}^T = -\dot{\tilde{W}}^T = \gamma s\sum_{i=1}^{n}\beta_i, \quad (36)$$

one has

$$\begin{aligned} \dot{V} &= s[\sum_{i=1}^{n} g_i\varepsilon_i + u_\mathrm{smc}] \\ &= s\sum_{i=1}^{n} g_i\varepsilon_i - \rho|s| \\ &\le |s|(\sum_{i=1}^{n} g_i\varepsilon_i^0 - \rho). \quad (37) \end{aligned}$$

Recall that $\rho > \sum_{i=1}^{n} g_i\varepsilon_i^0$, then it is true that

$$\dot{V} < 0 \quad (38)$$

such that s and $\tilde{W}$ converge to zero asymptotically. According to (27), z_i also converges to zero for $1 \le i \le n$. Specially, it is true that $x_1 - x_\mathrm{d} \to 0$ as $t \to 0$, viz. the tracking error converges to zero asymptotically.

Remark 1: In the presented paper, the input function $b(x)$ is assumed to be known. For real control objectives, $b(x)$ is normally nonzero and bounded in order to fulfil the controllability of the system, such that (30) is appropriate. If $b(x)$ is unknown, it can be also approximated by an additional network. In this case, some projection methods are needed in order to avoid possible singularities in $b(x)$.

Remark 2: Regarding the adaptive backstepping SMC in [12] and [13], the Lyapunov function is defined as $V = \frac{1}{2}\sum_{i=1}^{n-1} z_i^2 + \frac{1}{2}s^2 + \frac{1}{2}\gamma^{-1}\tilde{\theta}\tilde{\theta}^T$, where θ is the vector of unknown parameters. With the control scheme proposed here, the redundant term of z_i already included in s is omitted, such that both the stability analysis and the control law are considerably simplified.

C. The growing Gaussian network

To reduce the computational effort, a growing scheme for the GRBF networks proposed in [22] is applied. The GRBF networks start with very few neurons. New neurons are progressively added according to the novelty of the system states. The idea of growing networks with several subgrids [24] was adopted. The centres of the neurons are arranged on regular lattices and the widths are determined by heuristic methods. The adaptation of the networks is performed only by the determination of the output weights. Therefore, the problem of adaptation remains linear in the parameters. Then one can expect a fast convergence of the adaptation. The crossings of the subgrids provide only potential positions for the new neurons. Here, a popular idea has been adopted, where the neurons whose centres are included in a hypersphere of the actual inputs will be activated. In practice, for approximation of a nonlinear function defined on a compact set, as shown in Figure 1 for a two-dimensional case, the network starts with a very loose 2×2 base grid, where only 4 neurons are arranged on the edges. The arabian numbers denote the grids. The hyperspheres in the two-dimensional case are circles of different radii. Figure 1 also shows that for the current system state "$*$" only those neurons from different subgrids are activated which lie within the corresponding hyperspheres.

It is very difficult to determine analytically a proper amount of necessary neurons. A popular idea is to add new neurons according to the tracking error [25]. This leads to the problem that unnecessary neurons might be included. When the initial condition of the system lies far from the desired trajectory and the transient period is relatively long, one would get a very large network. Though the network size can be reduced by including a pruning strategy [24] [26] to delete superfluous neurons, the computational effort might be intermittently very large.

In this paper, a time-varying measure of the reasonable error bound is defined as

$$E(t) = \begin{cases} \eta_1 e^{-\eta_2 t} & \text{for} \quad |s| > z_1^0 \\ z_1^0 & \text{for} \quad |s| \le z_1^0 \end{cases}, \qquad (39)$$

where z_1^0 is the required tracking accuracy for $x_1 - x_d$, η_1 and η_2 are design parameters to be chosen. $E(t)$ seeks to represent the available tracking accuracy during the transient period. This is based on the requirement that the tracking error should converge faster than some exponential function. New neurons are only inserted into the network when the sliding variable $s(t)$ is larger than the current error bound

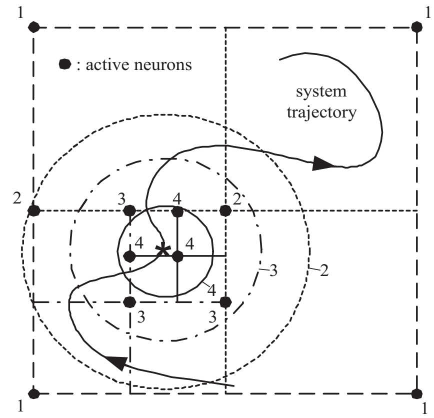

Fig. 1. Phase-plane portrait with an example of active neurons at the current system state for a lattice with three subgrids

$E(t)$. For the design, the factor η_1 can be chosen as $|s(0)|$. The exponent η_2 decides about the amount of neurons added to the network.

The introduction of new, denser "higher-order" subgrids has to satisfy the condition that the tracking error is larger than the current error bound $E(t)$. Furthermore, the time period between adding of subgrids must be long enough. In [24], 11 subgrids were used to control a SISO system. In this paper, much less subgrids are required. This is due to the use of a SMC term for the compensation of the approximation error. The neurons of "lower-order" subgrids, especially those of the base grid, can be treated as "global" approximators, which try to provide general information about the unknown function on the entire compact set. The neurons of the "higher-order" subgrids can then be treated as local approximators, which provide more details about the unknown function in a certain region.

IV. SIMULATION RESULTS

A Van der Pol system [21] is considered in the simulation:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= (1-x_1^2)x_2 - x_1 + (1+x_1^2+x_2^2)u, \end{aligned} \qquad (40)$$

with $x_1(0) = 0.5$, $x_2(0) = 0$. The control objective is to track a desired trajectory x_d with x_1. (40) is a simple example of the NPPF in (1). A GRBF network is used to approximate the nonlinear term $(1-x_1^2)x_2 - x_1$. The GRBF network has one base grid with 4 neurons and 2 subgrids with a maximum of 9 and 25 hidden neurons, respectively. The widths σ_i of the corresponding Gaussian functions of the different grids are π, $\pi/2$ and $\pi/4$, respectively. The hyperspheres have a radius of 0.5 and 0.25 for the second and the third subgrid, respectively. To avoid infinite-frequency switching of the control torques, a boundary layer with $\psi = 0.02$ is introduced so that when $|s| < \psi$ the control $u_{smc} = -(1/\psi)s$ is used. Other parameters used in the simulation studies are the error bound $z_1^0 = 0.02$, $\eta_1 = 0.5$, $\eta_2 = 1.5$, the adaptation rate $\gamma^{-1} = 5$ and the switching gain $\rho = 0.5$. The latter is quite conservative according to the approximation error of the GRBF network, yet much lower than the switching gain required for a classical SMC.

During the simulation, g_1 and c_1 are both set to 1. The desired and actual trajectories are illustrated in Fig.2. It is shown that x_1 converges to $x_\mathrm{d}(t) = \sin(t)$ very quickly. 16 neurons are activated finally, the nonlinearity $(1-x_1^2)x_2 - x_1$ and its approximation with the GRBF network are shown in Fig. 3. After a short transient period with oscillations, the GRBF network can follow the unknown nonlinearity, and after about 15 seconds, the approximation is very close to the real uncertainty. The tracking errors z_1 and z_2 and the control signal $u(t)$ are presented in Fig. 4 and Fig. 5, respectively.

Remark 3: The controller requires no information about the nonlinearities in f_i. In the second simulation example, a disturbance term $\sin(2t)x_2^3$ is added to $\dot{x}_2$ when $t > 10s$. Fig. 7 shows that the GRBF network has no difficulty to approximate the nonlinearity with the disturbance. The tracking error converges to zero as well. According to [17], however, a classical backstepping controller will probably fail dealing with disturbances.

Remark 4: As g_i, $i = 1, ..., n-1$ define the sliding surface, c_i, $i = 1, ..., n-1$ provide more freedom for determining the desired error dynamics. For the system (40), tracking errors of x_1 with g_1 taken as 1 and 3 are shown in Fig. 8.

V. CONCLUSIONS

In this paper an adaptive sliding-mode control scheme was proposed. With the application of the backstepping design procedure, the requirement on the matching condition for classical sliding-mode control was removed. Gaussian radial-basis-function networks were used to approximate the unknown system dynamics. A new growing scheme of the Gaussian networks proposed recently was applied in order to reduce the computational effort. The stability analysis and the control law are simple. The controller shows very good performance with a relatively small scale of the networks. The control system is robust to unknown nonlinearities and disturbances.

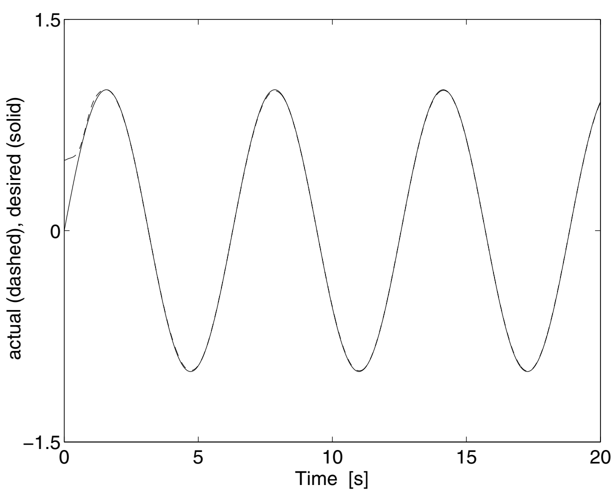

Fig. 2. Actual and desired output of x_1

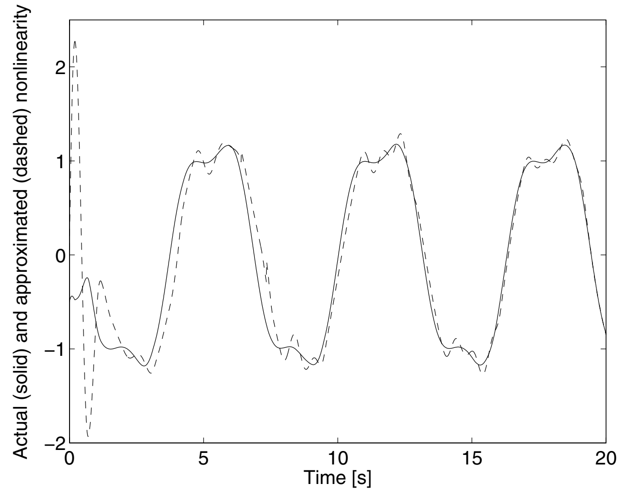

Fig. 3. The nonlinearity and the approximation

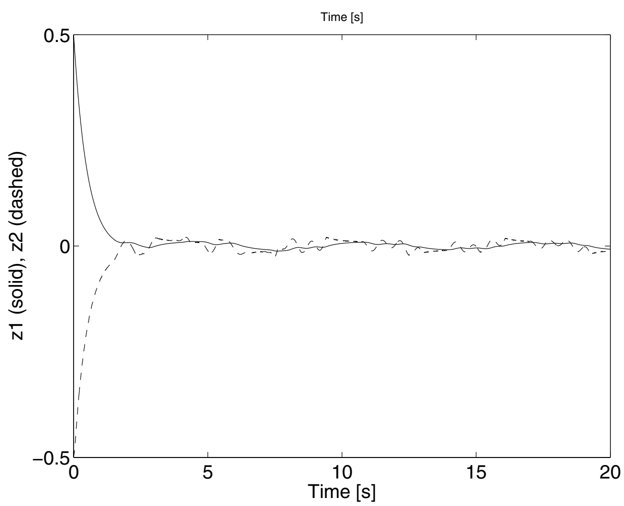

Fig. 4. Tracking error

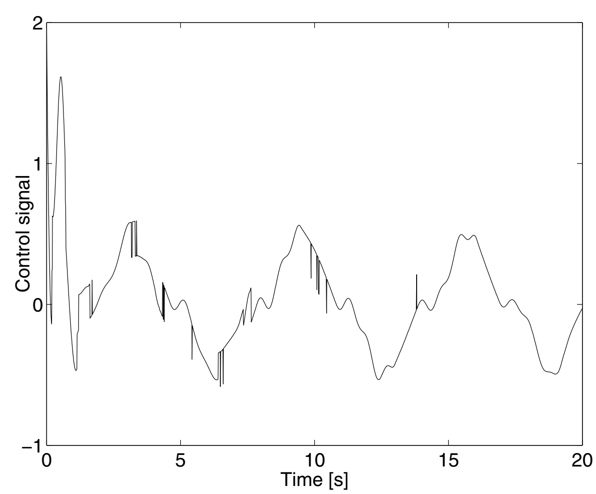

Fig. 5. Control signal

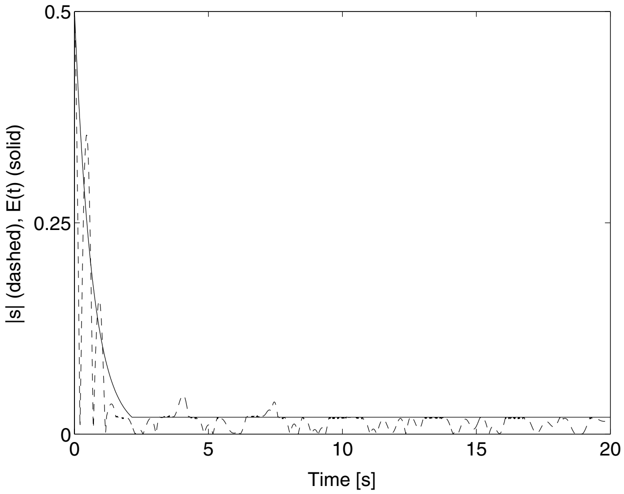

Fig. 6. $|s(t)|$ and the error bound $E(t)$

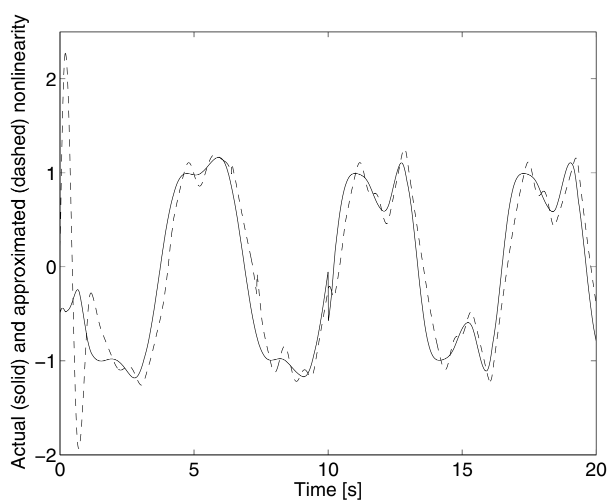

Fig. 7. The GRBF network can follow the disturbance

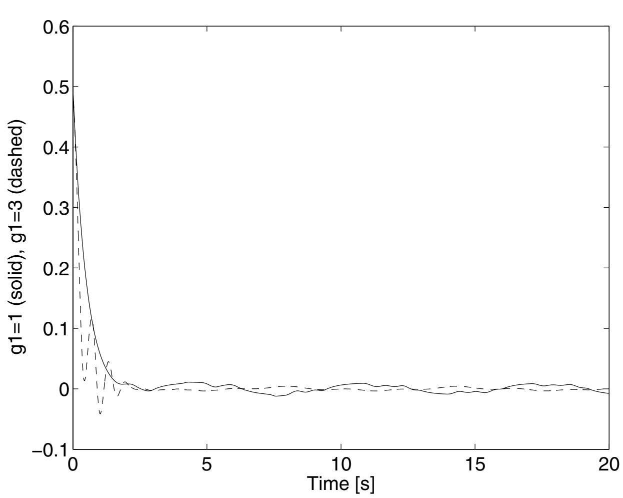

Fig. 8. Tracking error with different sliding surfaces

REFERENCES

[1] V.I. Utkin. Variable structure systems with sliding modes. *IEEE Transactions on Automatic Control*, 22:212–222, 1977.

[2] B. Drazenović. The invariance conditions in variable structure systems. *Automatica*, 5(3):287–295, 1969.

[3] S. Gutman. Uncertain dynamical systems - A lyapunov min-max approach. *IEEE Transactions on Automatic Control*, 24(3):437–443, June 1979.

[4] M.J. Corless and G. Leitmann. Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems. *IEEE Transactions on Automatic Control*, 26(5):1139–1144, October 1981.

[5] I. Kanellakopoulos, P.V. Kokotović, and A.S. Morse. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transactions on Automatic Control*, 36(11):1241–1253, 1991.

[6] R.A. Freeman and P.V. Kokotović. *Robust Nonlinear Control Design*. Birkhäuser, Boston, 1996.

[7] P. Kokotović and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37:637–662, 2001.

[8] D. Seto, M. Annaswamy, and J. Baillieul. Adaptive control of nonlinear systems with a triangular structure. *IEEE Transactions on Automatic Control*, 39(7):1411–1427, 1994.

[9] B. Yao. *Adaptive robust Control of Nonlinear Systems with Application to Control of Mechanical Systems*. PhD thesis, University of California at Berkeley, 1996.

[10] S.S. Ge, C.C. Hang, and T. Zhang. Stable adaptive control for nonlinear multivariable systems with a triangular control structure. *IEEE Transactions on Automatic Control*, 45(6):1221–1225, 2000.

[11] X.P. Liu, G.X. Gu, and K.M. Zhou. Robust stabilization of mimo nonlinear systems by backstepping. *Automatica*, 35:987–992, 1999.

[12] M. Rios-Bolivar, A.S.I. Zinober, and H. Sira-Ramirez. Dynamical adaptive sliding mode output tracking control of a class of nonlinear systems. *International Journal of Robust and Nonlinear Control*, 7:387–405, 1997.

[13] A.J. Koshkouei, R.E. Mills, and A.S.I. Zinober. Adaptive backstepping control. In X.H. Yu and J.X.Xu, editors, *Variable Structure Systems:Towards the 21st Century*, volume 274 of *Lecture Notes in control and Information Sciences*, pages 129–155. Springer, 2002.

[14] M. Won and J.K. Hedrick. Multi-surface sliding control of a class of uncertain nonlinear systems. *International Journal of Control*, 64(4):693–706, 1996.

[15] A. Stotsky, J.K. Hedrick, and P.P. Yip. The use of sliding modes to simplify the backstepping control method. In *Proc. American Control Conference*, pages 1703–1708, Albuquerque,NM, 1997.

[16] A.C. Huang and Y.C. Chen. Adaptive sliding control for single-link flexible-joint robot with mismatched uncertainties. *IEEE Transactions on Control Systems Technology*, 12(5):770–775, September 2004.

[17] T. Knohl. *Anwendung künstlicher neuronaler Netze zur nichtlinearen adaptiven Regelung*. PhD thesis, Ruhr-Universität Bochum, 2000.

[18] T. Knohl and H. Unbehauen. Annnac-extension of adaptive backstepping algorithm with artificial neural networks. *IEE Proceedings–Control Theory and Applications*, 147(2):177–183, 2000.

[19] J.Q. Gong and B. Yao. Neural network adaptive robust control of nonlinear systems in a normal form. *Asian Journal of Control*, 3(2):96–110, 2001.

[20] J.Q. Gong and B. Yao. Neural network adaptive robust control of nonlinear systems in semi-strict feedback form. *Automatica*, 37:1149–1160, 2001.

[21] T. Zhang, S.S. Ge, and C.C. Hang. Adaptive neural network control for strict-feedback nonlinear systems using backstepping design. *Automatica*, 36:1835–1846, 2000.

[22] L. Ma, K. Schilling, and Chr. Schmid. Adaptive sliding–mode control with gaussian network. Accepted by the 16th IFAC World Congress, Prague, July 2005.

[23] R.M. Sanner and J-J.E. Slotine. Gaussian networks for direct adaptive control. *IEEE Transactions on Neural Networks*, 3(6):837–863, 1992.

[24] G.P. Liu, V. Kadirkamanathan, and S.A. Billings. Variable neural networks for adaptive control of nonlinear systems. *IEEE Transactions on Systems, Man and Cybertics, Part C*, 29(1):34–43, 1999.

[25] S. Fabri and V. Kadirkamanathan. Dynamic structure neural networks for stable adaptive control of nonlinear systems. *IEEE Transactions on Neural Networks*, 7(5):1151–1167, 1996.

[26] Y. Li, N. Sundararajan, and P. Saratchandran. Neuro-controller design for nonlinear fighter aircraft maneuver using fully tuned RBF networks. *Automatica*, 37(8):1293–1301, 2001.