2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

Tracking Control of Nonaffine Systems: A Self-organizing Approximation Approach

Wenjie Dong, Yuanyuan Zhao, Jay A. Farrell
Department of Electrical Engineering
University of California, Riverside, CA 92521

***Abstract*— This paper considers the tracking control of a nonaffine system. A performance-dependent self-organizing approximation approach is proposed. The designer specifies a positive tracking error criteria. The self-organizing approximation based controller then monitors the tracking performance and adds basis elements only as needed to achieve the tracking specification.**

***Index Terms*— self-organizing approximation based control, adaptive nonlinear control, locally weighted learning.**

I. Introduction

In the past decades, on-line approximation based control has been considered extensively in e.g., [1–7, 9–12, 17–19, 22, 23]. The design and analysis of adaptive controllers involving on-line approximation to achieve stability and accurate trajectory tracking in the presence of unknown or partially unknown nonlinear dynamics have been well developed.

In general, on-line approximation based controllers cannot achieve an exact modeling of unknown nonlinearities, *inherent approximation errors* could arise even if optimal approximator parameters were selected. Under reasonable assumption on the basis function and the function to be approximated, for any given $\epsilon > 0$, if the network approximator has a sufficiently large number of nodes, then ϵ approximation accuracy can be achieved by proper selection of the approximator parameters [8, 15]. Thus, to meet ϵ approximation accuracy, one approach is to allocate a sufficient large number of learning parameters. However, allocating too many learning parameters bears the danger of over-parameterizing the approximation. This may have computation and performance penalties. The other approach is to choose suitable approximators. With these motivations, nonlinear adaptive control with function approximation employing automatic structure adaptation has been discussed in a few articles [1, 3, 5, 6, 13, 14, 19, 20, 22, 23]. Articles [3, 19] used wavelet networks and adapted the structure of the network in response to the evaluation of the magnitude of the output weights by "hard-thresholding". Smoothly interpolated linear models were considered in [5].

In [1, 20], local approximators within localized receptive field were defined and the on-line approximation was tuned in a local region without affecting the approximation accuracy previously achieved in other regions. Therefore, the function approximation structure is able to retain approximation accuracy as a function of the operating point. As in [5], the structure adaptation is based on exploration: if none of the existing basis functions is excited, then a new node is allocated. These articles also use gradient descent to adjust the distance metric of each local approximator so that each receptive field is tuned according to the local curvature properties of the unknown function. In [5, 13, 14], linearly parameterized locally model were used, which is a special case of the Receptive Field Weighted Regression (RFWR) approach. No stability results are given in [1, 20]. The common shortcoming of the existing approaches in [3, 5, 13, 14, 19] is that (i) they only address the stability analysis for the state and the approximator parameters, not the change in the number of basis functions, and (ii) the structure adaptation algorithms are defined by the trajectory, not by the performance. New approximator nodes are added when the current state is sufficiently far from all existing receptive field centers, whether or not additional approximation accuracy is required. Recent articles [6, 23] developed an approach where the approximation structure is adjusted during system operation, based on the observed trajectory tracking performance. A self-organized state estimation approach is developed in [22] and organizes the approximator structure based on estimation error. Article [6, 22, 23] focus on models that are affine in the control variable.

In this article we consider the tracking control of a n-th order nonaffine system. Our goal is to design a self-organizing on-line approximation based controller to achieve a prespecified tracking accuracy, without using high-gain control nor large magnitude switching. Towards this end, we first propose a sliding mode control such that the state of the system comes into the operational region, then we propose an adaptive controller in the operational region based on the self-organizing idea in [23] and approximation ideas in [7, 16]. It is shown that our proposed controller can achieve the prespecified tracking performance and requires less computation during control. The contribution of the article is that we extend the self-organizing based control method to control high-order nonaffine system.

II. Problem Statement

Consider single-input single-output (SISO), input-state feedback linearizable systems of the form

$$\dot{x}_i = x_{i+1}, \qquad 1 \le i \le n-1 \tag{1}$$
$$\dot{x}_n = h(x, u) \tag{2}$$

where $x = [x_1, \cdots, x_n]^\top \in R^n$ is the state vector and $u \in R$ is the control signal. The function $h(x, u)$ represents nonlinear effects that are unknown at the design stage. The function h is assumed to be differentiable.

Given a desired trajectory $x_d(t)$ with derivatives $x_d^{(i)}(t)$, $i = 1, \ldots, n$, each of which is available and bounded $\forall t \geq 0$. For convenient, we denote $x_c(t) = [x_{1c}, x_{2c}, \ldots, x_{nc}]^\top = [x_d, x_d^{(1)}, \ldots, x_d^{(n-1)}]^\top$.

Control Problem: Design the control signal u to steer $x_1(t)$ to track the desired trajectory $x_d(t)$ and to achieve boundedness for the states x_i for $i = 2, \ldots, n$.

To make the problem tractable, we make the following assumption.

Assumption 1: For any $x \in \mathcal{R}^n$ and $u \in \mathcal{R}$,

$$\epsilon_0(x) < \frac{\partial h(x, u)}{\partial u} < 2c(x) \quad (3)$$

$$|h(x, 0)| \leq b(x) \quad (4)$$

where $\epsilon_0(x)$, $c(x)$, and $b(x)$ are known positive functions.

Remark 1: In Assumption 1, the first inequality in (3) makes sure that the system is controllable at any time. The second inequality in (3) makes sure that $f(x)$ defined in eqn. (17) is unique and continuous (see Lemma 2). The bound $b(x)$ in (4) is used to make sure that the tracking error comes into and stays in a bounded operational region which is chosen by the designer.

Remark 2: The tracking control problem could be solved by

$$u = -\frac{1}{\epsilon_0}\left(Ke + (|\Lambda| + b(x)) sign(e)\right) \quad (5)$$

with e and Λ defined in (7) and (11), respectively; or by

$$u = -\frac{1}{\epsilon_0}\left(Ke + (|\Lambda| + b(x)) sat(e/\varepsilon)\right) \quad (6)$$

where ε is a positive constant. The control of (5) achieves finite time convergence of e to zero and asymptotic convergence of $x(t)$ to $x_c(t)$, but requires control with magnitude $|\Lambda| + b(x)$ switching at very high rates. The control of (6) achieves finite time convergence to $|e| \leq \varepsilon$ and $\|x - x_c\| < \gamma\varepsilon$ where γ is a constant determined by the choice of L in (7). The control of (6) still requires control with magnitude $|\Lambda| + b(x)$ and with the region of the sat function the effective gain is $(\|\Lambda\| + b)/\varepsilon$ which can be quite large when the desired accuracy ε is small. The magnitude of the switching term in the u from (6) as a function of e is shown as the dotted line in Fig. 1. In this paper we propose a new approach. We introduce a new parameter $\sigma > 0$ which is finite but can be significantly larger than ε. The large magnitude sliding mode term will be designed (see eqn. (12)) to ensure that $|e(t)| \leq \sigma$ is achieved in finite time and maintained for all future times. When $|e(t)| \leq \sigma$ a self-organizing approximation based controller (see the next section) is designed to ultimately achieve $|e| \leq \epsilon$. Due to the inclusion of the self-organizing approximator, the control is able to achieve this same tracking accuracy using a switching term of magnitude ε which is small in comparison to the switching term in control law (5). Fig. 1 shows as a solid line a sketch of the magnitude of the switching term from the proposed approach of (12).

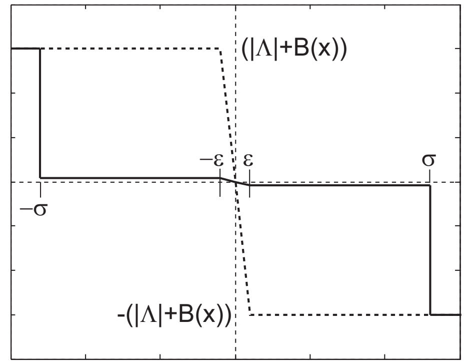

Fig. 1. Sketch of the magnitude of the switching term in the control laws of eqns. (6) (dotted) and (12) (solid).

III. Tracking Errors and Basic Control Structure

Throughout the article the tracking error components are defined as

$$\tilde{x}_i = x_i - x_{ic}, \quad 1 \leq i \leq n$$

where $\tilde{x}$ is the tracking error vector defined as $\tilde{x} = x - x_c = [\tilde{x}_1, \ldots, \tilde{x}_n]^\top$. Note that

$$\dot{\tilde{x}}_i = \tilde{x}_{i+1}, \quad 1 \leq i \leq n-1.$$

Let

$$e(t) = L^\top \tilde{x}(t) \quad (7)$$

where $L = [l_1, l_2, \ldots, l_{n-1}, 1]^\top$ is a constant vector. Note that $L^\top \tilde{x} = 0$ defines an $(n-1)$-dimensional hyperplane in $\Re^n$. The absolute value of $e(t)$ represents the distance of $\tilde{x}(t)$ from this hyperplane. On the hyperplane $e(t) = 0$, the dynamics of $\tilde{x}_1$ are defined by

$$\begin{aligned}
\tilde{x}_n + l_{n-1}\tilde{x}_{n-1} + \ldots + l_3\tilde{x}_3 + l_2\tilde{x}_2 + l_1\tilde{x}_1 &= 0 \\
\dot{\tilde{x}}_{n-1} + l_{n-1}\dot{\tilde{x}}_{n-2} + \ldots + l_3\dot{\tilde{x}}_2 + l_2\dot{\tilde{x}}_1 + l_1\tilde{x}_1 &= 0 \\
&\vdots \\
(s^{n-1} + l_{n-1}s^{n-2} + \ldots + l_3 s^2 + l_2 s + l_1)\tilde{x}_1 &= 0
\end{aligned}$$

where s is the Laplace variable; therefore, L is be selected so that $(s^{n-1} + l_{n-1}s^{n-2} + \ldots + l_3 s^2 + l_2 s + l_1) = 0$ is a Hurwitz polynomial. In this case, the transfer function

$$\frac{\tilde{x}_1(s)}{e(s)} = \frac{1}{s^{n-1} + l_{n-1}s^{n-2} + \ldots + l_3 s^2 + l_2 s + l_1}$$

is Bounded-Input-Bounded-Output (BIBO) stable. If $e(t)$ can be shown to be bounded for all $t \geq 0$, then each $\tilde{x}_i, 1 \leq i \leq n$ is bounded.

To allow the bounds to be easily expressed, we choose L such that

$$\begin{aligned}
e(t) &= \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x}_1 \\
&= [\lambda^{n-1}, C_{n-1}^1 \lambda^{n-2}, \ldots, C_{n-1}^{n-2}\lambda, 1]\tilde{x}
\end{aligned}$$

for some constant $\lambda > 0$. This implies that the vector L in (7) is defined as $l_i = C_{n-1}^{i-1}\lambda^{n-i}, \quad 1 \le i \le n$, where

$$C_{n-1}^{i-1} = \frac{(n-1)!}{(n-i)!(i-1)!}$$

is the binomial coefficient. The transfer functions to $\tilde{x}_i$ from e are

$$\begin{aligned} \frac{\tilde{x}_i(s)}{e(s)} &= \frac{\tilde{x}_1(s)}{e(s)} s^{i-1} = \frac{s^{i-1}}{(s+\lambda)^{n-1}} \\ &= \frac{1}{(s+\lambda)^{n-i}} \cdot \left(1 - \frac{\lambda}{s+\lambda}\right)^{i-1}, \ 1 \le i \le n. \end{aligned}$$

The advantage of defining $e(t)$ in this manner is that if there exists a constant $\mu_e > 0$ such that the magnitude of e is bounded as $|e(t)| \le \mu_e, \ \forall t \ge 0$, then the tracking errors are asymptotically bounded by

$$|\tilde{x}_i(t)| \le 2^{i-1}\lambda^{i-n}\mu_e, \ 1 \le i \le n, \tag{8}$$

which yields

$$\|\tilde{x}(t)\|_2 \quad \le \quad \|\lambda_v\|_2 \mu_e \text{ as } t \to \infty \tag{9}$$

with $\lambda_v^\top = [\lambda^{1-n}, 2\lambda^{2-n}, \ldots, 2^{n-2}\lambda, 2^{n-1}]$ and $\|\cdot\|_2$ being the 2-norm of a vector. See page 279-280 of [21] for additional detail.

The self-organizing on-line approximation based controller developed in the subsequent sections is designed to maintain stability and to achieve a tracking accuracy of $|e(t)| < \mu_e$ with μ_e prespecified at the design stage. If L is selected as in the previous paragraph, then $|e(t)| < \mu_e$ ensures that $|\tilde{x}_1| < \frac{1}{\lambda^{n-1}}\mu_e \doteq \mu_x$ as $t \to \infty$. Let $\mathcal{D}^n$ denote the region in $\mathcal{R}^n$ such that $|\tilde{x}_i| \le 2^{i-1}\lambda^{i-n}\sigma$ for $1 \le i \le n$. It is obvious that $\mathcal{D}^n$ is bounded since x_c and σ are bounded. Noting (8), if $|e| \le \sigma$, then $x \in \mathcal{D}^n$. We call the region $\mathcal{D}^n$ *the operational region*. It can be adjusted by the choice of σ.

With the definition of e in (7),

$$\begin{aligned} \dot{e} &= \Lambda + h(x,u) \\ &= \Lambda + cu + (h(x,u) - cu) \end{aligned} \tag{10}$$

where c is the bounded function in eqn. (3), and

$$\Lambda = \lambda^{n-1}\tilde{x}_2 + C_{n-1}^1\lambda^{n-2}\tilde{x}_3 + \cdots + C_{n-1}^{n-2}\lambda\tilde{x}_n - x_d^{(n)}. \tag{11}$$

We choose the control law

$$u = \begin{cases} \frac{1}{c}\left(-Ke - \Lambda - u_{ad} - \epsilon_f sat\left(\frac{e}{\epsilon_f}\right)\right), & |e| \le \sigma \\ -\frac{1}{\epsilon_0}[Ke + (|\Lambda| + b(x))sign(e)], & |e| > \sigma \end{cases} \tag{12}$$

where constant $K > 0$, $\sigma(> \mu_e)$ is a positive constant, $\epsilon_f(> 0)$ will be defined later,

$$sat(y) = \begin{cases} sign(y), & \text{if } |y| > 1 \\ y, & \text{otherwise} \end{cases}$$

and u_{ad} is defined later in (29).

Remark 3: If $|e| > \sigma$, the controller is a sliding mode control. If $|e| \le \sigma$, the controller is self-organizing in a manner that will be defined in the following section. We choose the control law (12) to be a switching controller because we want to use a high gain control to steer the state of the system to the bounded operational region $\mathcal{D}^n$. Within this operational region an adaptive learning controller is in charge of the control. Hence, in the operational region the control does not use large magnitude high gain switching even though the system model is unknown (see Fig. 1 and Remark 4 for more details).

Lemma 1: If $|e| > \sigma$, with the control in (12), the tracking error e exponentially decreases.

Proof: By the intermediate value theorem, there exists $\beta \in [0, u]$ such that eqn. (10) can be written as

$$\dot{e} \quad = \quad \Lambda + h(x,0) + \frac{\partial h(x,\beta)}{\partial u}u. \tag{13}$$

For $|e| > \sigma$, select the Lyapunov function

$$V = \frac{1}{2}e^2. \tag{14}$$

Differentiating V along the solution of (13) with the control (12) yields

$$\begin{aligned} \dot{V} &= e[\Lambda + h(x,0)] \\ &\quad -\frac{e}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u}[Ke + |\Lambda|sign(e) + b\ sign(e)] \\ &\le -\frac{Ke^2}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} - \left[\frac{1}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} - 1\right]|e|(|\Lambda| + b) \\ &\le -\frac{Ke^2}{\epsilon_0}\frac{\partial h(x,\beta)}{\partial u} \le -Ke^2. \end{aligned} \tag{15}$$

Therefore, e exponentially decreases if $|e| > \sigma$. ■

Due to the exponential decrease proved in Lemma 1, from any initial condition $e(0)$, the condition $|e(t)| \le \sigma$ is achieved in finite time. Once the condition $|e(t)| \le \sigma$ is achieved, it will be maintained for all future times. In the following, we mainly discuss the case that $|e| \le \sigma$.

If $|e| \le \sigma$, substituting (12) into (10) yields

$$\dot{e} = -Ke + (\Delta(x,u) - u_{ad}) - \epsilon_f sat\left(\frac{e}{\epsilon_f}\right) \tag{16}$$

where

$$\Delta(x,u) = h(x,u) - cu.$$

If

$$\Delta(x,u) - u_{ad} = 0,$$

it is easy to prove that e converges to zero. Lemma 2 shows that under suitable assumptions there indeed exists a unique $u_{ad}^* = f(x)$ such that

$$\Delta(x,u) - f(x) = 0. \tag{17}$$

Lemma 2: Under Assumption 1, there exits a unique continuous $f(x)$ such that $f(x)$ satisfies (17) for all $x \in \mathcal{D}^n$.

Proof: Noting Assumption 1, Lemma 2 can be proved by following the proof of Lemma 1 in [2, 16]. ■

By Lemma 2, function f exists. However, we cannot obtain the explicit expression of f because $h(x,u)$ is unknown.

In the sequel, we apply a self-organized locally weighted learning algorithm (LWL) to develop a basis for and an approximation to $f(x)$.

IV. LWL Algorithm and Structure Adaptation

In LWL [6,22], the approximation to $f(x)$ at a point x is formed from the normalized weighted average of local approximators $\hat{f}_k(x)$ such that

$$\hat{f}(x) = \frac{\sum_k \omega_k(x)\hat{f}_k(x)}{\sum_k \omega_k(x)} \tag{18}$$

where each ω_k is nonzero only on a set denoted by S_k (defined below in eqn. (19)) over which $\hat{f}_k$ will be adapted to improve its accuracy relative to f.

A. Weighting Functions

We define a continuous, non-negative and locally supported weighting function $\omega_k(x)$ for the k-th local approximator. Denote the support of $\omega_k(x)$ by

$$S_k = \Big\{ x \in \mathcal{D}^n \mid \omega_k(x) \neq 0 \Big\}. \tag{19}$$

Let $\bar{S}_k$ denote the closure of S_k. Note that $\bar{S}_k$ is a compact set. An example of a weighting function satisfying the above conditions is the biquadratic kernel defined as

$$\omega_k(x) = \begin{cases} \left(1 - \left(\frac{||x-c_k||}{\mu_k}\right)^2\right)^2, & \text{if } ||x-c_k|| < \mu_k \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

where c_k is the center location of the k-th weighting function and μ_k is a constant which represents the radius of the region of support. In this example, the region of support is

$$S_k = \Big\{ x \in \mathcal{D}^n \mid \|x - c_k\| < \mu_k \Big\}. \tag{21}$$

Since the approximator is self-organizing, the number of local approximators $N(t)$ is not constant. Conditions for increasing N at discrete instants of time are presented in Section IV-C. Since N is time varying, the region over which the approximator defined in eqn. (18) can have a nonzero value is also time varying. This region is defined as

$$\mathcal{A}^{N(t)} = \bigcup_{1 \le k \le N(t)} S_k.$$

When $x(t) \in \mathcal{A}^{N(t)}$, there exists at least one k such that $\omega_k(x) \neq 0$. The normalized weighting functions are defined as

$$\bar{\omega}_k(x) = \frac{\omega_k(x)}{\sum_{k=1}^{N(t)} \omega_k(x)}.$$

The set of non-negative functions $\{\bar{\omega}_k(x)\}_{k=1}^{N(t)}$ forms a *partition of unity* on $\mathcal{A}^{N(t)}$:

$$\sum_{k=1}^{N(t)} \bar{\omega}_k(x) = 1, \quad \text{for all } x \in \mathcal{A}^{N(t)}.$$

Note that the support of $\omega_k(x)$ is exactly the same as the support of $\bar{\omega}_k(x)$.

When $x(t) \notin \mathcal{A}^{N(t)}$, all $\omega_k(x)$ for $1 \le k \le N(t)$ are zero. Therefore, to complete the approximator definition of eqn. (18) to be valid for any $x \in \Re^n$:

$$\hat{f}(x) = \begin{cases} \sum_{k=1}^{N(t)} \bar{\omega}_k(x)\hat{f}_k(x) & \text{if } x \in \mathcal{A}^{N(t)} \\ 0 & \text{if } x \in \Re^n - \mathcal{A}^{N(t)}. \end{cases} \tag{22}$$

In the reminder of this section, we will only consider the case when $x(t) \in \mathcal{A}^{N(t)}$ to give all definitions for the LWL algorithm.

B. Local Approximators

We define

$$\hat{f}_k(x) = \Phi_k^T \hat{\theta}_{f_k} \tag{23}$$

where Φ_k is a prespecified vector of continuous basis functions. For the function $f(x)$, the vector $\theta^*_{f_k}$ denotes the unknown optimal parameter estimates for $x \in \bar{S}_k$:

$$\theta^*_{f_k} = \arg\min_{\hat{\theta}_{f_k}} \left(\int_{\bar{S}_k} \omega_k(x) \left| f(x) - \hat{f}_k(x) \right|^2 dx \right). \tag{24}$$

Note that $\theta^*_{f_k}$ is well defined for each k because f and $\hat{f}_k$ are smooth on compact $\bar{S}_k$. Therefore,

$$f^*_k = \Phi_k^\top \theta^*_k$$

will be referred to as the optimal local approximator to f on $\bar{S}_k$.

Let the optimal approximation error to f on $\bar{S}_k$ be denoted as ϵ_{f_k}:

$$\epsilon_{f_k}(x) \quad = \quad f(x) - f^*_k(x). \tag{25}$$

Since in subsequent expressions ϵ_{f_k} only appears as a product with $\omega_k(x)$, the value of $\epsilon_{f_k}(x)$ is immaterial outside $\bar{S}_k$. In order for ϵ_{f_k} to be defined everywhere, let

$$\epsilon_{f_k}(x) = \begin{cases} f(x) - f^*_k(x), & \text{on } \bar{S}_k, \\ 0, & \text{otherwise.} \end{cases}$$

The controller will use a known design constant $\epsilon_f > 0$. We make the following assumption.

Assumption 2: The basis set Φ_k is sufficiently rich and μ_k is sufficiently small such that $|\epsilon_{f_k}(x)| \le \bar{\epsilon}_f$ for $x \in \bar{S}_k$ for some (unknown) positive constant $\bar{\epsilon}_f < \epsilon_f$.

For a linear basis set $[1, x - c_k]$ for $k \in [1, N]$ and the region of supports are chosen as (21), this assumption is satisfied if $|f''(x)| < \frac{\epsilon}{2\mu_k^2}$. Note that the boundedness of $\max_{x \in \bar{S}_k}(|\epsilon_{f_k}(x)|)$ comes from the fact that $|\epsilon_{f_k}|$ is continuous on compact $\bar{S}_k$.

For any $x \in \mathcal{A}^{N(t)}$, $f(x)$ can be represented as the weighted sum of the local optimal approximators:

$$f(x) \quad = \quad \sum_k \bar{\omega}_k(x) f^*_k(x) + \delta_f(x). \tag{26}$$

This expression defines the optimal approximation error $\delta_f(x)$ on $\mathcal{A}^{N(t)}$ which satisfies $|\delta_f(x)| \leq \bar{\epsilon}_f$, since

$$\begin{aligned} |\delta_f| &= \left| f(x) - \sum_k \bar{\omega}_k(x) f_k^*(x) \right| \\ &= \left| \sum_k \bar{\omega}_k(x)(f(x) - f_k^*(x)) \right| \\ &\leq \sum_k \bar{\omega}_k(x) |\epsilon_{f_k}(x)| \qquad (27) \\ |\delta_f| &\leq \max_k(|\epsilon_{f_k}|) \sum_k \bar{\omega}_k(x) = \bar{\epsilon}_f. \qquad (28) \end{aligned}$$

Therefore, if each local optimal model $f_k(x)$ has accuracy $\bar{\epsilon}_f$ on $\bar{S}_k$, then the global accuracy of $\sum_k \bar{\omega}_k(x) f_k(x)$ on $\mathcal{A}^{N(t)}$ also achieves at least accuracy $\bar{\epsilon}_f$. The δ_f term in (26) is the *inherent approximation error* of $\hat{f}(x)$ for $f(x)$.

For the adaptive portion of the control law we choose

$$u_{ad} = \hat{f}. \qquad (29)$$

To obtain $\hat{f}$, we need to estimate $\hat{\theta}$. For $x \in \mathcal{A}^i$, we choose the adaptive laws

$$\dot{\theta}_{f_k} = \begin{cases} \Gamma_{f_k} \bar{\omega}_k e \Phi_k & \text{if } |e| > \mu_e \\ 0 & \text{otherwise} \end{cases} \qquad (30)$$

The parameter adaptation will turn off when either $x \notin \mathcal{A}^i$ or $|e| \leq \mu_e$.

C. Structure Adaptation

We initialize the estimation of f in (22) by $\hat{f}$ with no local approximators, i.e., $N(0) = 0$; therefore, the set $\mathcal{A}^0$ is initially empty. We define the following criteria for adding a new local approximator to the approximation structure. A local approximator $\hat{f}_k$ is added and $N(t)$ is increased by one:

1) if $|e| \leq \sigma$ and the present operating point $x(t)$ does not activate any of the existing local approximators (i.e., $\max_{1 \leq k \leq N(t)}(\omega_k(x)) = 0$); and
2) if $|e| \leq \sigma$ and the function $e\dot{e} \geq 0$ while $|e(t)| > \mu_e$.

With the above criterions, $N(t)$ monotonically increases. $\mathcal{A}^{N(t)}$ changes as $N(t)$ increases. Therefore, the structure of $\hat{f}$ in (22) changes as $N(t)$ increases.

V. Self-organizing Controller and Stability Analysis

For the controller described in Sections III-IV, we have the following result.

Theorem 1: The system described by eqn. (1–2) with control law

$$u = \begin{cases} \frac{1}{c}\left(-Ke - \Lambda - \hat{f} - \epsilon_f sat\left(\frac{e}{\epsilon_f}\right)\right), & |e| \leq \sigma \\ -\frac{1}{\epsilon_0}[Ke + (|\Lambda| + b(x)) sign(e)], & |e| > \sigma \end{cases} \qquad (31)$$

using the self-organizing function approximation (22) with updated laws (30) and structure adaptive criterion in Subsection IV-C has the following properties:

1) $\tilde{x}$, e, $\tilde{\theta}_{f_k}$, θ_{f_k}, $N(t) \in \mathcal{L}_\infty$;
2) $e(t) = L^\top \tilde{x}$ is ultimately bounded by $|e(t)| \leq \mu_e$;
3) each $\tilde{x}_i$ is ultimately bounded by $|\tilde{x}_i| \leq 2^{i-1}\lambda^{i-n}\mu_e$, for $i = 1, \ldots, n$, with λ being a constant selected for designing the L vector. ■

The proof of Theorem 1 is omitted due to space limitation.

Remark 4: In Theorem 1, the proposed control law consists of two parts. If $|e| > \sigma$, the control law is a sliding mode control. The magnitude of the control depends on the bound function $b(x)$. If $|e| \leq \sigma$, the control law is a self-organizing control. The self-organizing control learns the unknown system and improves the tracking performance gradually based on the structure adaptation scheme. In the self-organizing control the magnitude of the self-organizing control does not depend on the bound function $b(x)$. Fig 1. 1 shows a sketch of the control magnitude with the control law in Theorem 1. In the control, several control parameters play important roles. Constant σ determines the size of the operational region $\mathcal{D}^n$ which the self-organizing control applies. If σ is large, the size of the region $\mathcal{D}^n$ will be large. Constant $\mu_e(< \sigma)$ determines the tracking error as time tends to infinity. If μ_e is small the tracking error will be small. During the control, μ_k determine the size of the locally learning region. If μ_k are small, ϵ_f can be chosen small. But small μ_k means that there will be large number of locally learning regions. In practical control, the control parameters should be chosen according to the tradeoff of different factors.

VI. Conclusion

This paper considers tracking control for nonaffine systems. In the operational region a self-organizing controller is proposed with the aid of a lemma from [2, 16] and the self-organizing approach proposed in [23]. The proposed controller has the ability to adjust the structure of approximators and will make the tracking error smaller than a given positive constant. The approach can be directly extended to case where $\dot{x}_n = f(x) + g(x)u + h(x, u)$ where $f(x)$ and $g(x)$ are known, and $h(x, u)$ represents model error.

Acknowledgements: This material is based upon work supported by the National Science Foundation under Grant No. ECCS-0701791. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] C. G. Atkeson, A. W. Moore, and S. Schaal, "Locally weighted learning for control," *Artifical Intelligence Review*, vol. 11, pp. 75–113, 1996.

[2] A. J. Calise, N. Hovakimyan, and M. Idan, "Adaptive output feedback control of nonlinear systems using neural networks," *Automatica*, vol. 37, no. 1, pp. 1201–1211, 2001.

[3] M. Cannon and J.-J. Slotine, "Space-frequency localized basis function networks for nonlinear system estimation and control," *Neurocomputing*, vol. 9, pp. 293–342, 1995.

[4] F.-C. Chen and H. K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," *IEEE Trans. on Automatic Control*, vol. 40, no. 5, pp. 791–801, 1995.

[5] J. Y. Choi and J. A. Farrell, "Nonlinear adaptive control using networks of piecewise linear approximators," *IEEE Trans. on Neural Networks*, vol. 11, no. 2, pp. 390–401, 2000.

[6] J. A. Farrell and Y. Zhao, "Self-organizing approximation based control," *Proc. of American Control Conference*, pp. 3378–3384, 2006.

[7] S. S. Ge and J. Zhang, "Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback," *IEEE Trans. on Neural Networks*, vol. 14, no. 4, pp. 900–918, 2003.

[8] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1989.

[9] J.-P. Jiang and L. Praly, "Design of robust adaptive controllers for nonlinear systems with dynamic uncertainties," *Automatica*, vol. 34, no. 7, pp. 825–840, 1998.

[10] E. Lavretsky and N. Hovakimyann, "Adaptive dynamic inversion for nonaffine-in-control systems via time-scale separation: part ii," *Proceedings of america control conference*, p. 35483553, 2005.

[11] F. L. Lewis, K. Liu, and Y. A, "Neual net robot control with guaranteed tracking performance," *IEEE Trans. on Neural Networks*, vol. 6, no. 3, pp. 703–715, 1995.

[12] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Trans. on Neural Networks*, vol. 7, no. 2, pp. 388–399, 1996.

[13] J. Nakanishi, J. A. Farrell, and S. Schaal, "Composite adaptive control with locally weighted statistical learning," *Neural Networks*, vol. 18, no. 1, pp. 71–90, 2005.

[14] J. Nakanishi, J. A. Farrell, and S. Schall, "A locally weighted learning composite adaptive controller with structure adaptation," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 882–889, 2002.

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

[16] J.-H. Park, S.-H. Huh, S.-H. Kim, S.-J. Seo, and G.-T. Park, "Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks," *IEEE Trans. on Neural Networks*, vol. 16, no. 2, pp. 414–422, 2005.

[17] M. Polycarpou, "Stable adaptive neural control scheme for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 447–451, 1996.

[18] M. Polycarpou and M. Mears, "Stable adaptive tracking of uncertian systems using nonlinearly parameterized on-line approximators," *International Journal of Control*, vol. 70, no. 3, pp. 363–384, 1998.

[19] R. M. Sanner and J. J. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. on Neural Networks*, vol. 3, pp. 837–863, 1992.

[20] S. Schaal and C. G. Atkeson, "Constructive incremental learning from only local information," *Neural Computation*, vol. 10, no. 8, pp. 2047–2084, 1998.

[21] J. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1991.

[22] Y. Zhao and J. A. Farrell, "Performance-based self-organizing approximation for scalar state estimation and control," *Proc. of American Control Conference*, 2007.

[23] ——, "Self-organizing approximation-based control for higher order systems," *IEEE Transactions on Neural Networks*, vol. 18, no. 4, pp. 1220–1231, 2007.