2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Self-organizing Approximation Based Control with $\mathcal{L}_1$ Transient Performance Guarantees

Yiming Chen, Jay A. Farrell

***Abstract*— This paper considers tracking control for a n-th order system with unknown nonlinearities. A performance-dependent self-organizing approximation approach is proposed. The self-organizing approximation based controller monitors the tracking performance and adds basis elements only as needed to achieve the tracking specification. The tracking performance is guaranteed not only in steady state but also in transient with a $\mathcal{L}_1$ bound. The low-pass filter used in this control design avoids the high-frequency oscillation while ensuring transient performance. To show the effectiveness of the proposed controller, a numerical example is included.**

I. Introduction

On-line approximation based control has been considered extensively during past decades in e.g., [1, 2, 7–10, 12, 14, 18–21, 24, 25, 27]. The design and analysis of adaptive controllers, which involve on-line approximation to achieve stability and accurate trajectory tracking in the presence of unknown or partially unknown nonlinear dynamics, has been well developed.

For on-line approximation, under reasonable assumptions on the function to be approximated and the basis functions, for any given $\epsilon^* > 0$ and for a known compact set of approximation, if the approximator has a sufficiently large number of nodes, then ϵ^* approximation accuracy can be achieved by proper selection of the approximator structure and parameters [11, 22]. Thus, to satisfy ϵ^* approximation accuracy, one approach is to allocate a sufficient large number of learning elements with an appropriate network structure, at the design stage. However, this is not as straightforward as it may appear, as the appropriate number of nodes and the network structure cannot be defined for an arbitrary unknown function. Allocating too many learning elements bears the danger of over-parameterizing the approximation and may have computational and performance penalties.

An alternative approach is to define the approximator structure automatically during operation. Adaptive controllers employing self-organizing function approximators have been discussed in several articles [1, 2, 8, 10, 12, 17–21, 24, 25, 27].

In each of the articles [1, 2, 8, 12, 17–21, 24, 25], the structure self-organization is *exploration-based*: if none of the existing nodal functions is excited sufficiently, then a new node is allocated. Therefore, excess nodes may be allocated in regions of state space where they are not needed to support the control objective. Pruning methods are considered in various articles to remove unneeded nodes [2, 17–19, 24]. In articles [17, 18, 24], the self-organization is based on exploration along with other criteria, such as a threshold on the size of the error for adding new nodes; however, there is no theoretical basis for the selection of the threshold. In the method proposed in [10, 27], the structure self-organization is derived within the Lyapunov context which provides a theoretical basis for a *performance-based* self-organization. In this approach, new nodes are only added if necessary to achieve the performance objective; therefore, pruning methods are not required.

Notice that none of the papers cited above provided transient performance guarantees, which is also a challenging issue in applications of on-line approximation-based adaptive controllers. To ensure the boundedness of transient in Model Reference Adaptive Control (MRAC) scheme, a filtered version of MRAC, termed as $\mathcal{L}_1$ adaptive control, was developed recently in [3, 4, 6]. This kind of controller ensures output tracking not only in steady state but also on transient. $\mathcal{L}_1$ Neural Network Adaptive Control Architecture is proposed in [5], for the case where the number of neural nodes is predesigned and fixed during the control process.

The main contribution of the article is the development of a self-organizing approximation based controller with $\mathcal{L}_1$ transient bound. The development requires a major reformulation of the analysis in [3–5, 10, 26–28]. The final result is that the controller adjusts the on-line approximation (the number of nodes, definition of basis set, parameters) as necessary to ultimately achieve a prespecified tracking error. During the training and self-organizing transient, the method guarantees certain bounds on the tracking error. Simulation results are included to demonstrate the effectiveness of the approach.

For the analysis in this article, the following basic definitions and facts from linear system theory will be useful [13, 15, 29]: $\|\xi\|_{\mathcal{L}_\infty}$ and $\|\xi_t\|_{\mathcal{L}_\infty}$ denote the $\mathcal{L}_\infty$ norm and truncated $\mathcal{L}_\infty$ norm of a signal $\xi(t) : [0, \infty) \mapsto \mathbb{R}^n$. The $\mathcal{L}_1$ gain of a stable, proper, single-input single-output system with impulse response $h(t)$ is defined to be $\|h(t)\|_{\mathcal{L}_1}$. Let $v(t) = \zeta^\top u(t)$ where $\zeta \in \mathbb{R}^n$ is a constant vector and then the $\mathcal{L}_1$ gain from $u(t)$ to $v(t)$ is defined to be $\|\zeta^\top\|_{\mathcal{L}_1} = \max_{i=1,\ldots,n}(|\zeta_i|)$, where ζ_i is the i_{th} component of $\zeta^\top$.

II. Problem Formulation

The following SISO system dynamics [5] are considered:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + b\big(u(t) - f(x(t))\big), \\ y(t) &= c^\top x(t), \quad x(0) = x_0 = 0 \end{aligned} \tag{1}$$

Yiming Chen and Jay A. Farrell are with Department of Electrical Engineering, University of California, Riverside, CA 92521, USA `yichen, farrell@ee.ucr.edu`

978-1-4577-0079-8/11/$26.00 ©2011 AACC

where $x(t) : \mathbb{R} \mapsto \mathbb{R}^n$ is the system state vector assumed to be measured and available, $u(t) : \mathbb{R} \mapsto \mathbb{R}$ is the control signal, $b, c \in \mathbb{R}^n$ are known constant vectors, A is a known $n \times n$ matrix with (A, b) controllable, $y : \mathbb{R} \mapsto \mathbb{R}$ is the tracking output, and $f(x) : \mathbb{R}^n \mapsto \mathbb{R}$ is an unknown continuous function, with Lipschitz constant L [15].

It is assumed that the upper bound for $f(0)$ is known

$$|f(0)| \leq B. \tag{2}$$

Control Objective: The control objective is to design an adaptive controller to ensure that $y(t)$ tracks a bounded continuous reference signal $r(t)$, with known upper bound of $\|r\|_{\mathcal{L}_\infty}$. This tracking should be satisfied both in transient and steady state, while all other error signals remain bounded.

More rigorously, the control objective can be stated as design of a control signal $u(t)$ to achieve

$$\|y(t) - y_d(t)\|_{\mathcal{L}_\infty} \leq B_1, \tag{3}$$

where a design constant B_1 is specified before the control operation, and $y_d(t)$ is defined as the inverse Laplace transformation of $Y_d(s)$, where $Y_d(s) = D(s)R(s)$,

and $D(s)$ is a strictly proper stable LTI system that specifies the desired transient and steady-state performance while $R(s)$ is the Laplace transform of $r(t)$.

III. Self-Organizing Approximation-Based $\mathcal{L}_1$ Adaptive Controller

Within the control structure developed in [3], we design and analyze a self-organizing control approach that achieves the control objective specified in Section II.

Let the control signal be

$$u(t) = u_1(t) + u_2(t), \text{ where } u_1(t) = -K^\top x(t), \tag{4}$$

$u_2(t)$ is an adaptive controller to be determined later, and $K \in \mathbb{R}^n$ is a design gain that ensures $A_m = A - bK^\top$ is Hurwitz. Equivalently, that $H_o(s) = (sI - A_m)^{-1}b$ is stable. Notice that if A is Hurwitz, then we can set $K = 0$. The following dynamics can be derived with the control signal in (4):

$$\begin{aligned} \dot{x}(t) &= A_m x(t) + b\big(u_2(t) - f(x(t))\big) \\ y(t) &= c^\top x(t); \quad x(0) = x_0. \end{aligned} \tag{5}$$

For this system, the following *state predictor* is introduced ,

$$\begin{aligned} \dot{\hat{x}}(t) &= A_m \hat{x}(t) + b\big(u_2(t) - \hat{f}(x(t))\big) \\ \hat{y}(t) &= c^\top \hat{x}(t); \quad \hat{x}(0) = x_0, \end{aligned} \tag{6}$$

where $\hat{f}(x(t)) = \hat{W}^\top \phi(x)$ is an approximation of $f(x)$. To approximate the unknown function $f(x)$, in this article we define a approach to self-organize the Locally Weighted Learning (LWL). See Section IV.

Letting $\bar{r}(t) = \hat{f}\big(x(t)\big)$ and $\bar{R}(s) = \mathcal{L}[\bar{r}(t)]$, we consider the following control design for (6):

$$U_2(s) = C(s)\bar{R}(s) + k_g R(s), \tag{7}$$

where $U_2(s)$ is the Laplace transform of $u_2(t)$, $C(s)$ is a stable and strictly proper system with DC gain $C(0) = 1$ and $k_g = -1/c^\top A_m^{-1} b$.

With the control signal defined in (7), the closed-loop state predictor in (6) can be viewed as an LTI system with two inputs $\bar{r}(t)$ and $r(t)$:

$$\hat{X}(s) = \bar{G}(s)\bar{R}(s) + G(s)R(s),$$

where $\hat{X}(s)$ is the Laplace transform of $\hat{x}(t)$, $\bar{G}(s) = H_o(s)\big(C(s) - 1\big)$ and $G(s) = k_g H_o(s)$. Let $\bar{g}(t)$, $g(t)$ and $h_o(t)$ be the inverse Laplace transformations of $\bar{G}(s)$, $G(s)$ and $H_o(s)$, respectively. Notice that $\bar{G}(s)$ and $G(s)$ are strictly proper stable systems, since both $H_o(s)$ and $C(s)$ are strictly proper stable systems.

The following performance requirement (i.e. the $\mathcal{L}_1-$**gain requirement**) will ensure boundedness of the entire system and desired transient performance [5].

$$\|\bar{g}(t)\|_{\mathcal{L}_1} < \frac{1}{L}, \tag{8}$$

where L is the Lipschitz constant defined in Section II.

Define a compact operational region $\mathcal{D}_x$

$$\mathcal{D}_x = \big\{x \big| \|x\|_\infty \leq \gamma_r + \gamma_0 + \gamma_1 + \sigma\big\} \tag{9}$$

where $\sigma > 0$ is an arbitrary positive constant, and γ_r, γ_0 and γ_1 are bounds to be defined later in eqn. (16-18).

Remark 1: From eqn. (2), (9) and the Lipschitz condition, it is straightforward to get the upperbound of $f(x)$ over $\mathcal{D}_x$

$$|f(x)| \leq |f(0)| + L\|x\|_\infty \leq B_f, \; x \in \mathcal{D}_x,$$

where $B_f = B + L(\gamma_r + \gamma_1 + \gamma_0 + \sigma)$. △

IV. Locally Supported Function Approximation

This article uses *Locally Weighted Learning* (LWL) to approximate $f(x)$, i.e., construct $\hat{f}(x) = \hat{W}\phi(x)$. In this section, the details of LWL are presented. In LWL, 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

$$\hat{f}(x) = \frac{\sum_k \omega_k(x)\hat{f}_k(x)}{\sum_k \omega_k(x)} \tag{10}$$

where each ω_k is nonzero only on a set denoted by S_k (defined in eqn. (11)) over which $\hat{f}_k$ will be adapted to improve its accuracy relative to f. Herein, we only state the main characteristics that we require. Additional detail is presented in [1, 9, 10, 25, 26].

A. Weighting Functions

We define a continuous, non-negative and locally supported weighting function $\omega_k(x)$ for the k-th local approximator, with $k \in 1, \ldots, N(t)$, where $N(t)$ is the total number of local approximators at time t. Denote the support of $\omega_k(x)$ by

$$S_k = \Big\{x \in \mathcal{D}_x \mid \omega_k(x) \neq 0\Big\}. \tag{11}$$

Let $\bar{S}_k$ denote the closure of S_k. Note that $\bar{S}_k$ is compact.

An example of a compactly supported weighting function is the biquadratic kernel defined as

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

where $c_k \in \mathbb{R}^n$ is the center location of the k-th weighting function and $\mu \in \mathbb{R}$ is a constant which represents the radius of the region of support S_k. In this example, the region of support is

$$S_k = \Big\{ x \in \mathcal{D}_x \mid \|x - c_k\| < \mu \Big\}.$$

Since the approximator is self-organizing, $N(t)$ is not constant. Conditions for increasing $N(t)$ at discrete instants of time are presented in Section V-C. Since $N(t)$ is time varying, the region over which the approximator defined in eqn. (10) 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 $\bar{\omega}_k(x) = \omega_k(x)/\sum_{k=1}^{N(t)} \omega_k(x)$. The set of non-negative functions $\{\bar{\omega}_k(x)\}_{k=1}^{N(t)}$ forms a *partition of unity* on $\mathcal{A}^{N(t)}$: $\sum_{k=1}^{N(t)} \bar{\omega}_k(x) = 1$, 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. (10) to be valid for any $x \in \mathcal{D}_x$:

$$\hat{f}(x) = \begin{cases} \displaystyle\sum_{k=1}^{N(t)} \bar{\omega}_k(x)\hat{f}_k(x) & \text{if } x \in \mathcal{A}^{N(t)} \\ 0 & \text{if } x \in \mathcal{D}_x - \mathcal{A}^{N(t)}. \end{cases} \quad (12)$$

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. Optimal Local Approximators

We define

$$\hat{f}_k(x) = \Phi_k^\top \hat{\theta}_{f_k} \quad (13)$$

where Φ_k is a vector of continuous basis functions. For example, it can be selected as $\Phi_k^\top = [1, x - c_k]$ for $k \in [1, N(t)]$. For the function $f(x)$, the vector $\theta^*_{f_k}$ denotes the unknown optimal parameter estimates for $x \in \bar{S}_k$:

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

Note that $\theta^*_{f_k}$ is well defined for each k because f and $\hat{f}_k$ are smooth enough on compact $\bar{S}_k$ (from the Lipschitz continuous condition). Therefore, $f_k^* = \Phi_k^\top \theta^*_{f_k}$ 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}: $\epsilon_{f_k}(x) = f(x) - f_k^*(x)$. 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 $\epsilon^* > 0$. We make the following assumptions.

Assumption 1: The basis set Φ_k and μ are selected such that $|\epsilon_{f_k}(x)| \le \bar{\epsilon}_f$ for $x \in \bar{S}_k$ for some (unknown) positive constant $\bar{\epsilon}_f < \epsilon^*$.

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

$$f(x) = \sum_k \bar{\omega}_k(x) f_k^*(x) + \epsilon(x). \quad (14)$$

This expression defines the optimal approximation error $\epsilon(x)$ on $\mathcal{A}^{N(t)}$ which satisfies $|\epsilon(x)| \le \bar{\epsilon}_f$ [9, 27, 28]. 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 ϵ^*. The $\epsilon(x)$ term in (14) is the *inherent approximation error* of $\hat{f}(x)$ for $f(x)$.

Since we have assumed that $f(x)$ is unknown, the parameter vector $\theta^*_{f_k}$ is unknown for each k. The control law will, therefore, be written using an approximated function defined by (12) and locally on $\bar{S}_k$ by (13). The controller will be adaptive in the sense that the local parameter vectors $\hat{\theta}_{f_k}$ will be adjusted to improve the controller performance while the controller is in operation.

We further assume that a compact convex set $\Omega^{N(t)}$ is a known *priori* such that the optimal weight

$$W = [\ \theta^{*\top}_{f_1}\ \cdots\cdots\ \theta^{*\top}_{f_{N(t)}}\ ] \in \Omega^{N(t)},$$

where $\Omega^{N(t)} = \Theta_1 \times \cdots \times \Theta_{N(t)}$ and Θ_k are known compact convex sets such that $\theta_{f_k} \in \Theta_k$.

Thus, based on (12), to approximate $f(x)$, we have

$$\hat{f}(x) = \begin{cases} \hat{W}^\top \phi(x) & \text{if } x \in \mathcal{A}^{N(t)} \\ 0 & \text{if } x \in \mathcal{D}_x - \mathcal{A}^{N(t)}. \end{cases}$$

where $\hat{W} = [\ \hat{\theta}^\top_{f_1}\ \cdots\cdots\ \hat{\theta}^\top_{f_{N(t)}}\ ]^\top$ is the approximate weight and $\phi(x) = [\ \bar{\omega}_1 \Phi_1^\top\ \cdots\cdots\ \bar{\omega}_{N(t)} \Phi_{N(t)}^\top\ ]^\top$ is the function approximation basis.

The adaptive law for $\hat{W}$ is that for $x(t) \in \mathcal{A}^{N(t)}$,

$$\dot{\hat{W}}(t) = \begin{cases} \Gamma_c Proj\big(\hat{W}(t), \tilde{x}^\top(t) P b \phi(x)\big), & \text{if } x \in \mathcal{A}^{N(t)} \\ 0, & \text{otherwise}. \end{cases} \quad (15)$$

where $\hat{W}(0) = \hat{W}_0$ and $\tilde{x}(t) = \hat{x}(t) - x(t)$ is the predictor error, $\Gamma_c \in \mathbb{R}^+$ is the adaptation gain, $Proj(\cdot,\cdot)$ denotes the projection operator [16], and $P = P^\top > 0$ is the solution of the algebraic Lyapunov equation $A_m^\top P + P A_m = -Q$, where $Q > 0$ is selected by the designer.

In Self-Organizing LWL approximation, the number of local approximators, i.e. $N(t)$, is not constant but can increase automatically based on some criteria, in order to satisfy the prespecified approximation accuracy requirement adaptively. In V-C, these criteria are specified.

V. Analysis of Self-Organizing Approximation-Based $\mathcal{L}_1$ Adaptive Controller

A. Notations

Based on the notation introduced above, a set of bounds are defined as follows [5],

$$\gamma_r = \big(\|g(t)\|_{\mathcal{L}_1}\|r\|_{\mathcal{L}_\infty} + \|\bar{g}(t)\|_{\mathcal{L}_1}(B+\epsilon^*) + \|h_o(t)\|_{\mathcal{L}_1}\big)/\big(1 - L\|\bar{g}(t)\|_{\mathcal{L}_1}\big) \quad (16)$$

$$\gamma_0 = \sqrt{\frac{\bar{\lambda}(P)}{\underline{\lambda}(P)}\left(\frac{2\|Pb\|\epsilon^*}{\underline{\lambda}(Q)}\right)^2 + \frac{W_{max}}{\underline{\lambda}(P)\Gamma_c}} \quad (17)$$

$$\gamma_1 = \frac{5\epsilon^*\|\bar{g}(t)\|_{\mathcal{L}_1} + \|h_o(t)\|_{\mathcal{L}_1}\epsilon^* + \big(1 + \|q_1(t)\|_{\mathcal{L}_1}\big)\gamma_0}{1 - L\|\bar{g}(t)\|_{\mathcal{L}_1}} \quad (18)$$

where $q_1(t) \triangleq \mathcal{L}^{-1}\big[C(s) - 1\big]$ and $W_{max} \triangleq \max_{W\in\Omega^{N(t)}} 4\|W\|^2$, where W is the optimal weight for the approximation to $f(x)$.

B. Without approximation

When $x(t) \in \mathcal{D}_x - \mathcal{A}^{N(t)}$ (*i.e.*, $\hat{f} = 0$), from (5) and (6) it follows that $\tilde{x} = \hat{x} - x$ satisfies

$$\dot{\tilde{x}}(t) = A_m\tilde{x}(t) + bf(x(t)), \quad \tilde{x}(0) = 0. \quad (19)$$

Consider the Lyapunov function $V_0(t) = \tilde{x}^\top(t)P\tilde{x}(t)$, whose time derivative along solutions of (19) is

$$\dot{V}_0 = -\tilde{x}^\top(t)Q\tilde{x}(t) + 2b^\top P\tilde{x}(t)f(x). \quad (20)$$

For all x such that $|f(x)| \le \epsilon^*$, when $\tilde{x}^\top(t)Q\tilde{x}(t) > 2b^\top P\tilde{x}(t)\epsilon^*$, it will be ensured that $\dot{V}_0 \le 0$.

If V_0 increases while $\tilde{x}^\top(t)Q\tilde{x}(t) > 2b^\top P\tilde{x}(t)\epsilon^*$, then it must be true that $|f(x)| > \epsilon^*$. Therefore, the Lyapunov function V_0 provides a mechanism to detect those locations along the trajectory $x(t) \notin \mathcal{A}^{N(t)}$ where $|f(x(t))| > \epsilon^*$. This motivates the following criterion for augmenting the approximator structure.

C. Structure Adaptation

We initialize the approximation of f in (12) 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 the present operating point $x(t)$ does not activate any of the existing local approximators (i.e., $\max_{1\le k\le N(t)}(\omega_k(x)) = 0$); and
2) $\dot{V}_0(t) \ge 0$ while $\tilde{x}^\top(t)Q\tilde{x}(t) > 2b^\top P\tilde{x}(t)\epsilon^*$.

With the above criteria, $N(t)$ is non-decreasing. The structure of $\hat{f}$ in (12) and the region $\mathcal{A}^{N(t)}$ changes as $N(t)$ increases. These criteria are motivated by eqn. (20).

For $i \ge 1$, we denote the time at which the i-th local approximator is added as T_i (i.e., $N(T_i) = i$ and $\lim_{\epsilon\to 0} N(T_i - \epsilon) = i - 1$). With this notation, $N(t)$ is constant with value i for $t \in [T_i, T_{i+1})$. It is possible that for some i, the approximator has sufficient approximation capability, in which case $T_{i+1} = \infty$. The center location of the new local approximator is denoted as c_i. At $t = T_i$, when the i-th node is added, it is the case that $x(t) \notin \mathcal{S}_k$ for $k = 1, ..., N(T_i) - 1$. The center location $c_{N(T_i)} \in \mathcal{D}_x$ will be selected such that $x(T_i) \in \mathcal{S}_{N(T_i)}$ and $c_{N(T_i)} \notin \mathcal{S}_j$, $\forall j \in [1, N(T_i)]$.

D. Reference System for Self-organizing Adaptive Controller

To specify the reference system that the Self-Organizing controller in (4) and (6)-(7) tracks, consider the ideal adaptive controller in (4) and (7):

$$u_r(s) = k_g r(s) + \eta(s) - K^\top x_r(s) \quad (21)$$

where $\eta(s)$ is the filtered output of $W^\top\phi(x_r(t))$ by $C(s)$, i.e. $\eta(s) = C(s)\eta_1(s)$. $\eta_1(s)$ is the Laplace transformation of $W^\top\phi(x_r(t))$ and $x_r(s)$ is introduced to denote the Laplace transformation of the closed-loop system state with the controller (21).

For $x_r(t) \in \mathcal{D}_x$, the control law in (21) leads to the following closed-loop dynamics:

$$\begin{aligned} x_r(s) &= G(s)r(s) + \bar{G}(s)\eta_1(s) - H_o(s)\epsilon_r(s) \\ y_r(s) &= c^\top x_r(s) \end{aligned} \quad (22)$$

where $\epsilon_r(s)$ is the Laplace transformation of $\epsilon(x_r(t)) = f(x_r(t)) - W^\top\phi(x_r(t))$, and $x_r(0) = x_0$.

E. Theorem 1

With the notation and analysis above, the following theorem can be proved.

Theorem 1: The system defined in (1) with the Self-organizing $\mathcal{L}_1$ adaptive controller has the following properties:

1) The number of allocated approximators $N(t)$ is finite $\forall t \ge 0$.
2) The closed-loop system in (5) is stable with $\hat{W}$, x, $\tilde{x}$, $x_r \in \mathcal{L}_\infty$. In fact, $\|x_r\|_{\mathcal{L}_\infty} \le \gamma_r$ where γ_r is defined in (16).
3) The state-prediction error $\|\tilde{x}\| \le 2\epsilon^*\|Pb\|/\underline{\lambda}(Q)$.

Proof: The proof proceeds in an inductive fashion.

1) The controller starts at $t = 0$ with $N(0) = 0$ (i.e. $\hat{f}(x) = 0$, no approximation). By the analysis in Subsection V-B, $\dot{V}_0 = -\tilde{x}^\top Q\tilde{x} + 2b^\top P\tilde{x}f(x)$. The controller monitors $V_0(t) = \tilde{x}^\top P\tilde{x}$. If $V_0(t)$ is decreasing for all $t \ge 0$, then the proof is complete. If $\tilde{x}^\top(t)Q\tilde{x}(t) \le 2b^\top P\tilde{x}\epsilon^*$, then $\|\tilde{x}(t)\| \le 2\epsilon^*\|Pb\|/\underline{\lambda}(Q)$, which achieves the performance specification whether or not $V_0(t)$ increases. If $\tilde{x}^\top(t)Q\tilde{x}(t) > 2b^\top P\tilde{x}\epsilon^*$ and $V_0(t)$ increases for some $t = T_1$, then the Self-organizing process is initiated with $N(T_1) = 1$ and $\mathcal{A}^1 = S_1$.

2) For $i \ge 1$ and $t \ge T_i$, define $\mathcal{B}^{N(T_i)} = D_x - \mathcal{A}^{N(T_i)}$, where $\mathcal{A}^{N(t)} = \bigcup_{1\le k\le N(t)} S_k$. Define the Lyapunov function

$$V_i = \tilde{x}^\top P\tilde{x} + \tilde{W}^\top\tilde{W}/\Gamma_c.$$

Using (15), we have

$$\dot{V}_i = -\tilde{x}^\top(t)Q\tilde{x}(t) + 2b^\top P\tilde{x}(t)f(x), \ \forall x \in \mathcal{B}^{N(T_i)},$$

$$\dot{V}_i \le -\tilde{x}^\top(t)Q\tilde{x}(t) + 2|b^\top P\tilde{x}(t)|\epsilon^*,\ \forall x \in \mathcal{A}^{N(T_i)}.$$

Thus, when $\tilde{x}^\top(t)Q\tilde{x}(t) \ge 2|b^\top P\tilde{x}(t)|\epsilon^*$, $\dot{V}_i \le 0$, $\forall x \in \mathcal{A}^{N(T_i)}$. The projection function in (15) ensures that $\hat{W} \in \Omega$, $\forall x \in \mathcal{A}^{N(T_i)}$. For all $t \ni x(t) \in \mathcal{B}^{N(T_i)}$, adaptation is off; therefore, $\dot{V}_i(t) = \dot{V}_0(t)$ and the controller monitors $V_0(t)$. If $V_0(t)$ is decreasing $\forall t > T_i \ni x(t) \in \mathcal{B}^{N(T_i)}$, then $V_i(t)$ is also decreasing and the proof is complete with $T_{i+1} = \infty$. If $\tilde{x}^\top(t)Q\tilde{x}(t) \le 2b^\top P\tilde{x}\epsilon^*$, then $\|\tilde{x}(t)\| \le 2\epsilon^*\|Pb\|/\underline{\lambda}(Q)$. If $\tilde{x}^\top(t)Q\tilde{x}(t) > 2b^\top P\tilde{x}\epsilon^*$ and $V_0(t)$ increases at some $t = T_{i+1}$, then $N(T_i) = i+1$ and the approximation region expands $\mathcal{A}^{N(T_{i+1})} = \mathcal{A}^{N(T_i)} \bigcup S_{N(T_{i+1})}$. To complete the proof we now need to show that the process must terminate with $N(t)$ being finite.

In this paragraph, we prove that $N(t)$ and W_{max} are finite. For this purpose, assume that $N(t)$ tends to infinity. Then there exists a infinite sequence of center locations $\{c_i\}_{i=1}^{\infty}$ with each $c_i \in \mathcal{D}_x$. From Bolzano-Weierstrass theorem [23], any bounded infinite sequence on a compact set has a convergent subsequence. Let $\{c_{i_k}\}_{i=1}^{\infty}$ be a convergent subsequence of $\{c_i\}_{i=1}^{\infty}$, then by the definition of a convergent subsequence there exists an integer $\bar{N}$ such that for $i_k > \bar{N}$, $\|c_{i_k}\| < \mu$. But $\|c_i - c_j\| > \mu$ $(\forall i, j)$ by the structure adaptation algorithm because $c_i \notin \mathcal{S}_j$ and vice versa. Therefore, we have a contradiction which implies that $N(t)$ must remain finite. Because the approximation is self-organizing, $N(t)$ can increase during the control process, expanding the dimension of the compact set $\Omega^{N(t)}$. However, it is trivial to show that W_{max} is still finite because of the fact checked above that $N(t)$ is finite.

3) The proof of $\|x_r\|_{\mathcal{L}_\infty} \le \gamma_r$ now follows by the same logic of *Lemma* 5 in [5]. ∎

F. Boundedness and Guaranteed Transient Performance

Theorem 2:

$$\begin{aligned}\|x - x_r\|_{\mathcal{L}_\infty} &\le \gamma_1\\ \|y - y_r\|_{\mathcal{L}_\infty} &\le \|c^\top\|_{\mathcal{L}_1}\gamma_1\\ \|u - u_r\|_{\mathcal{L}_\infty} &\le \gamma_2\end{aligned}$$

where γ_1 is defined in (18), and

$$\gamma_2 = \|q_2(t)\|_{\mathcal{L}_1}\gamma_0 + 3\|c(t)\|_{\mathcal{L}_1}\epsilon^* + \big(\|c(t)\|_{\mathcal{L}_1}L + \|K^\top\|_{\mathcal{L}_1}\big)\gamma_1$$

where $q_2(t) \triangleq \mathcal{L}^{-1}\big[C(s)\frac{1}{c_o^\top H_o(s)}c_o^\top\big]$, $c(t) = \mathcal{L}^{-1}\big[C(s)\big]$, $\|c^\top\|_{\mathcal{L}_1}$ and $\|K^\top\|_{\mathcal{L}_1}$ are the $\mathcal{L}_1$ gains of constant vector $c^\top$ and $K^\top \in \mathbb{R}^n$.

Proof: The proof follows by the same logic of *Theorem 1* in [5]. ∎

Corollary 1: Given the system in (1) and the Self-Organizing adaptive controller defined via (4) and (6)-(7), we have

$$\lim_{\Gamma_c \to \infty, \epsilon^* \to 0} \big(x(t) - x_r(t)\big) = 0 \quad \forall t \le 0$$

$$\lim_{\Gamma_c \to \infty, \epsilon^* \to 0} \big(y(t) - y_r(t)\big) = 0 \quad \forall t \le 0$$

$$\lim_{\Gamma_c \to \infty, \epsilon^* \to 0} \big(u(t) - u_r(t)\big) = 0 \quad \forall t \le 0$$

From *Corollary* 2, if the adaptive gain is selected sufficiently large and the Self-Organizing approximation is accurate enough, that $x(t)$, $y(t)$ and $u(t)$ track $x_r(t)$, $y_r(t)$ and $u_r(t)$ not only asymptotically but also during the transient. Therefore, the control objective is reduced to designing K and $C(s)$ to ensure that the reference system has the desired response, $D(s)$ from $r(t)$ to y_r.

VI. Design of the Self-Organizing Controller

To design K and $C(s)$ ensuring the desired response $D(s)$, consider the following signals:

$$y_d(s) \triangleq c^\top x_d(s) \triangleq c^\top G(s)r(s) = k_g c^\top H_o(s)r(s) \quad (23)$$

where $x_d(0) = x_0$ and

$$u_d(s) \triangleq k_g r(s) - K^\top x_d(s) + C(s)\eta_7(s) \quad (24)$$

where $\eta_7(s)$ is the Laplace transformation of $f(x_d)$. Since $r(t)$ is bounded and $G(s)$ is stable, $x_d(t)$ is also bounded and we can straightforwardly derive its upperbound:

$$\|x_d\|_{\mathcal{L}_\infty} \le \|g(t)\|_{\mathcal{L}_1}\|r\|_{\mathcal{L}_\infty}.$$

It follows that $x_d(t) \in \mathcal{D}_x$ for any $t \ge 0$.

Lemma 1: Given the system in (1), the reference system in (21) and (22), and the Self-Organizing adaptive controller defined via (4), (6)-(7), subject to (8), we have

$$\begin{aligned}\|x_r - x_d\|_{\mathcal{L}_\infty} &\le \gamma_3\\ \|y_r - y_d\|_{\mathcal{L}_\infty} &\le \|c^\top\|_{\mathcal{L}_1}\gamma_3\\ \|u_r - u_d\|_{\mathcal{L}_\infty} &\le \big(\|K^\top\|_{\mathcal{L}_1} + \|c(t)\|_{\mathcal{L}_1}L\big)\gamma_3\\ &\quad + \|c(t)\|_{\mathcal{L}_1}\epsilon^*\end{aligned} \quad (25)$$

where $\gamma_3 = \|\bar{g}(t)\|_{\mathcal{L}_1}\big(B + L\gamma_r + \epsilon^*\big) + \|h_o(t)\|_{\mathcal{L}_1}\epsilon^*$.

Proof: The proof follows by the same logic of the proof of *Lemma* 6 in [5]. ∎

From Lemma 1, the condition in (8) is crucial for characterization of the transient performance. For this purpose, the control design is reduced to finding a strictly proper stable $C(s)$ and a gain K to satisfy the performance requirement in (8). It follows from (25) that for achieving that y_r tracks y_d, it is desirable to ensure $\|\bar{g}(t)\|_{\mathcal{L}_1}$ small enough [5].

From [4], we have the following lemma

Lemma 2: Let $C(s) = \omega/(s+\omega)$. For any single input n-output strictly proper stable system $H_o(s)$, $\lim_{\omega\to\infty}\|\bar{g}(t)\|_{\mathcal{L}_1} = 0$.

Lemma 2 states that if one chooses $D(s) = k_g c^\top H_o(s)$, then by increasing the bandwidth of the low-pass system $C(s)$, it is possible to render $\|\bar{g}(t)\|_{\mathcal{L}_1}$ arbitrarily small, and, hence $y_r(s) \approx y_d(s) = D(s)r(s)$.

Remark 2: Otherwise, $\|\bar{g}(t)\|_{\mathcal{L}_1}$ can also be decreased via higher order filter design method, rather than increasing the bandwidth of $C(s)$ [4]. In the following simulation, a third order filter is used for controller design. △

Corollary 2: From *Theorem* 2 and *Lemma* 1, it is straightforward to get the upperbound B_1 in eqn. (3),

$$\|y(t) - y_d(t)\|_{\mathcal{L}_\infty} \le B_1 = \|c^\top\|_{\mathcal{L}_1}(\gamma_1 + \gamma_3).$$

Corollary 2 implies that the control objective specified in section II is satisfied. Notice that B_1 can be rendered arbitrarily small by decreasing $\|\bar{g}(t)\|_{\mathcal{L}_1}$ and increasing the accuracy of function approximation.

Remark 3: Notice that zero initial condition is assumed for the convenience of analysis in this article. By accommodating the operational region $\mathcal{D}_x$ defined by (9) to ensure the property 2) in *Theorem* 1, this approach can be extended to nonzero initial conditions. Additional details can also be derived by following the discussion in *Remark* 9 [5]. $\triangle$

VII. Simulation

Consider the following nonlinear system

$$\dot{x}(t) = Ax(t) - f\big(x(t)\big) + bu(t), \quad x(0) = x_0 = [0,\ 0]^\top$$

where

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

and $x = [x_1\ x_2]^\top$ is the measured state vector, u is the control signal, and $f(x) : \mathbb{R}^n \mapsto \mathbb{R}$ is an unknown nonlinear function. The control objective is to design a Self-Organizing adaptive controller to ensure that $y = x_1(t)$ tracks any continuous $r(t)$, subject to $|r(t)| \leq 1$, both in transient and steady state with bound $B_1 = 0.138$ (see eqn. (3)).

To demonstrate that the self-organizing function approximation assigns local approximators only as needed to achieve the specification, the simulation uses

$$f(x) = \begin{cases} -0.44 \cdot tan(-1.4), & x_1 < -3.5 \\ -0.44 \cdot tan(0.4 \cdot x_1), & x_1 \in [-3.5, 3.5] \\ -0.44 \cdot tan(1.4), & x_1 > 3.5. \end{cases}$$

This $f(x)$ has Lipschitz constant $L = 1$ and $B = 0$. Note that $|f(x)| < \epsilon^* = 0.05$ for $|x_1| < 0.283$, which is shaded in Fig. 2.

We choose $K = [2,\ 3]^\top$ for eqn. (4) to make $A_m = A - bK^\top$ is Hurwitz and $\hat{W}^\top(t)$ is updated following eqn. (15). Choosing $Q = 50 \cdot I_{2\times 2}$, yields

$$P = \begin{bmatrix} 62.5 & 12.5 \\ 12.5 & 12.5 \end{bmatrix}.$$

The adaptation is initialized with $N = 0$, $\hat{W}(0) = \hat{W}_0 = 0$ and uses $\Gamma_c = 5000$.

The adaptive component of control law $u(t)$ (i.e. $u_2(t)$) is implemented following eqn. (7) with a third-order low-pass filter [4, 5].

$$C(s) = \frac{3\omega^2 s + w^3}{(s + \omega)^3}$$

and $\omega = 10$.

We define the reference input $r(t)$ as a bipolar square wave, with magnitude 3 and a 14-second period. Fig. 1 shows that the system output $y(t)$ tracks the reference input $r(t)$ and satisfies transient bound B_1, due to the size of Γ_c and the accuracy of the self-organizing approximation. Fig. 2 shows the distribution of local approximations assigned by the Self-organizing Adaptive Controller. Notice that in the middle of Fig. 2, there is a blank zone where no nodes are assigned. This blank zone includes the region where $|f(x)| < \epsilon^*$, which is unknown to the controller. Fig. 3 shows $f(x)$ and $\hat{f}(x)$ versus time. The approximation $\hat{f}$ converges rapidly toward $f(x(t))$, due to the size of Γ_c. The high variation in $\hat{f}$ versus time is due to the size of Γ_c and $x(t)$ moving through S_k. As $x(t)$ passes through the region S_k, the currently active local approximator switches. The $k-th$ approximator is adapting locally dependent on the training experience received over the past history of $x(t) \in S_k$. The control signal $u(t)$ is plotted in Fig. 4 which shows that there is no high-frequency oscillation in the controller output, due to the filter $C(s)$.

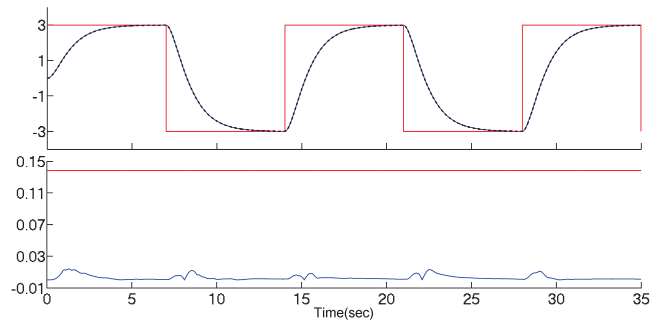

Fig. 1. Top: performance of $\mathcal{L}_1$ SO Adaptive Controller: $r(t)$(red, solid), $y(t)$(blue, solid) and $y_d(t)$(black, dashed). Bottom: tracking error versus time satisfies eqn. (3): $B1$(red), $|y - y_d|$(blue).

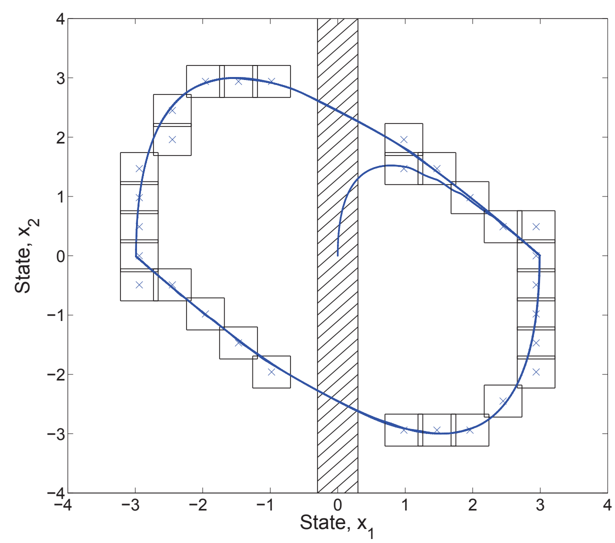

Fig. 2. Phase plane plot of x_1 versus x_2 for $t \in [0,\ 28]s$. The $\times' s$ indicate the assigned center locations and the small square around each center location represents the associated region of support.

VIII. Conclusion

In this article, a self-organizing approximation based control method, which has $\mathcal{L}_1$ transient performance guarantees, is developed to solve the output tracking problem for systems of order n with unknown nonlinearities. The low-pass filter in

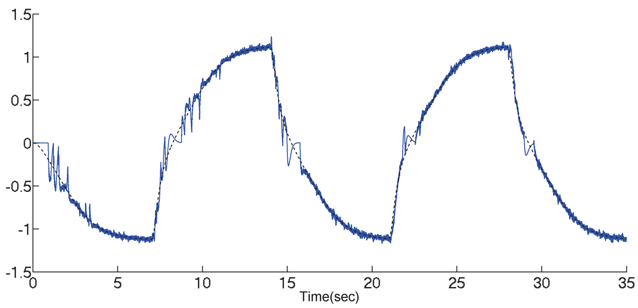

Fig. 3. Time history of $\hat{f}(x)$ and $f(x)$.

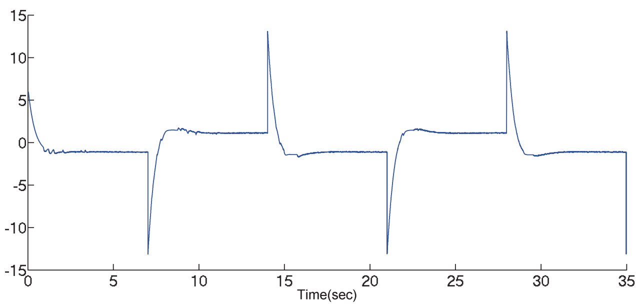

Fig. 4. Time history of $u(t)$.

this control design avoids high-frequency oscillations while ensuring transient performance. Performance based self-organizing function approximation leads to a more effective controller. Simulation results show the effectiveness of this controller.

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] 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.

[3] C. Cao and N. Hovakimyan, "Design and analysis of a novel $\mathcal{L}_1$ adaptive controller, Part I: Control signal and asymptotic stability," *Proceedings of the 2006 American Control Conference*, 2006.

[4] ——, "Design and analysis of a novel $\mathcal{L}_1$ adaptive controller, Part II: Guaranteed transient performance," *Proceedings of the 2006 American Control Conference*, 2006.

[5] ——, "Novel $\mathcal{L}_1$ neural network adaptive control architecture with guaranteed transient performance," *IEEE Transactions on Neural Networks*, 2007.

[6] ——, "Design and analysis of a novel $\mathcal{L}_1$ adaptive control architecture with guaranteed transient performance," *IEEE Transactions on Automatic Control*, 2008.

[7] 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.

[8] 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.

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

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

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

[12] J. Hu and G. Pratt, "Self-organizating CMAC neural networks and adaptive dynamic control," 1999, pp. 259–265.

[13] P. Ioannou and J. Sun, *Robust Adaptive Control*. Prentice Hall, 1996.

[14] 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.

[15] H. K. Khalil, *Nonlinear Systems*, third edition ed. Prentice Hall, 2002.

[16] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*. Wiley, 1995.

[17] J. Lian, Y. Lee, S. D. Sudhoff, and S. H. Zak, "Self-organizing radial basis function network for real-time approximation of continuous-time dynamical systems," *IEEE Trans. on Neural Networks*, vol. 19, no. 3, pp. 460–474, March 2008.

[18] ——, "Variable structure neural network based direct adaptive robust control of uncertain systems," in *Proceedings of the American Control Conference*, 2008, pp. 3402–3407.

[19] C.-M. Lin and T.-Y. Chen, "Self-organizing cmac control for a class of MIMO uncertain nonlinear systems," *Neural Networks, IEEE Transactions on*, vol. 20, no. 9, pp. 1377 –1384, 2009.

[20] 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.

[21] 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.

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

[23] H. L. Royden, *Real Analysis*. Prentice-Hall, Inc., 1988.

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

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

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

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

[28] ——, "Self-organizing approximation for controllable nonaffine systems," *Proceedings of the IEEE Decision and Control Conference*, pp. 1281–1287, 2007.

[29] K. Zhou and J. C. Doyle, *Essentials of Robust Control*. Prentice Hall, 1998.