Proceedings of the 1996 IEEE
International Conference on Robotics and Automation
Minneapolis, Minnesota - April 1996

OUTPUT FEEDBACK CONTROL OF RIGID ROBOTS USING DYNAMIC NEURAL NETWORKS

Young H. Kim and Frank L. Lewis
Automation and Robotics Research Institute, The University of Texas at Arlington
7300 Jack Newell Blvd. South, Fort Worth, TX 76118-7115
e-mail. ykim@arri.uta.edu, flewis@controls.uta.edu

ABSTRACT

A robust neural network (NN) output feedback scheme is proposed for the motion control of rigid robots. A dynamic NN observer is presented to estimate the joint speeds. The stability of a closed-loop system composed of a robot, a NN observer, and a NN controller is proven. The NN weights in both the observer and the controller are tuned on-line, with no off-line learning phase required. Most importantly, we can guarantee the boundness of the estimated velocities, the position tracking errors, and the NN weights. Also no exact knowledge of the robot dynamics is required so that the NN controller is model-free and so applicable to any type of rigid robot. When compared with adaptive-type controllers, we do not require persistent excitation conditions, linearity in the unknown system parameters, or the tedious computation of a regression matrix. Thus the new NN approach represents an improvement over adaptive techniques.

1. Introduction

For full state feedback control, measurements of both joint positions and joint velocities are required. The joint position measurements can be obtained by means of encoders, which can give very accurate measurements of joint displacements. Conversely, joint velocity measurements are obtained by means of tachometers, which are often contaminated by noise. Therefore it is important to achieve satisfactory control performance by using only joint position measurements.

In order to eliminate the need for tachometers and numerical differentiation, one option is to use a joint velocity observer. The concept of the observer was tailored to estimate manipulator joint velocities by Nicosia and Tomei [16]. That nonlinear observer was inserted in the feedback loop ensuring local asymptotic stability. Canudas de Wit and Fixot [3] proposed an adaptive observer-controller that uses sliding observers. Zhu *et al.* [23] presented an algorithm combining variable structure control and observation for treating the parameter uncertainties. The observers with Lipschitz condition were not recommended for direct use in a closed-loop controller [18]. In [2], the passivity concept was utilized to develop the combined controller-observer system for robot motion control. Recently, a reduced-order adaptive velocity observer has been combined with an adaptive controller for robot trajectory control in [5]. But the adaptation law for the observer contains an observer velocity error, therefore an approximation of the observer must be done for implementation.

NNs have been used for approximation of nonlinear systems, for classification of signals, and for associative memory. For control engineers, the approximation capability of NN is usually used for system identification, or identification based control. However there is very little about the use of NN in direct closed-loop controllers with observers that yield guaranteed performance.

In this paper, NNs are used for closed-loop output feedback control. Only the inertia matrix is assumed known; the Coriolis terms, gravity terms, and friction terms are assumed unknown. First, a dynamic NN observer is proposed to estimate the unknown velocity. A feedforward NN with weight dynamics is inserted in the feedback path to capture the nonlinear characteristics of the observer system. This architecture formulates a dynamic NN system [15]. The estimated velocity is used as an input to a second static NN that functions as a feedback controller. We will show that all the signals in the closed-loop system are bounded under some suitable conditions on the controller and the observer gains.

Research Supported by NSF grant ECS-9521673

2. Preliminaries

We define the norm of a vector $x \in R^n$,and a matrix $A \in R^{m \times n}$ as

$$\|x\| = \sqrt{x^T x}\,,\ \|A\| = \sqrt{\lambda_{max}(A^T A)} \tag{2.1}$$

with $\lambda_{max}(\cdot)$ the maximum eigenvalue. For any matrix $A(x)=A^T(x)>0$, A_m and A_M denote the minimum and maximum eigenvalues of $A(x)$, respectively.

Given $A=[a_{ij}]$ and $B \in R^{m \times n}$, the Frobenius norm is defined by

$$\|A\|_F^2 = tr(A^T A) = \sum_{ij} a_{ij}^2 \tag{2.2}$$

with $tr(\cdot)$ the trace. The associated inner product is $\langle A,B \rangle_F = tr(A^T B)$.

2.1 Neural Networks

A 'two-layer' feedforward NN in Fig. 1 has two layers of adjustable weights. Given an input vector $x \in R^n$ the NN output vector y is given by the formula

$$y_i = \sum_{j=1}^{N_h} \left[w_{ij} \sigma\left(\sum_{k=1}^{n} v_{jk} x_k + \theta_{vj} \right) + \theta_{wi} \right]; \quad i = 1, \ldots, m \tag{2.3}$$

where $\sigma(\cdot)$ are the activation functions and N_h is the number of hidden-layer neurons. The first-layer interconnection weights are denoted by v_{jk} and the second-layer weights by w_{ij}. The threshold offsets are denoted by θ_{vj}, θ_{wi}. Many different activation functions $\sigma(\cdot)$ are in common use, including sigmoid, hyperbolic tangent, gaussian, etc.

By collecting all the NN weights v_{jk}, w_{ij} into matrices of weights V^T, W^T one can write the NN equation in terms of vectors as

$$y = W^T \sigma(V^T x) \tag{2.4}$$

with the vector of activation functions defined by $\sigma(z)=[\sigma(z_1) \ldots \sigma(z_n)]^T$ for a vector $z \in R^n$. The thresholds are included as the first columns of the weight matrices. Any tuning of W and V then includes tuning of the thresholds as well. To accomplish this, a '1' must be appended as the first element of the vectors x and $\sigma(V^T x)$.

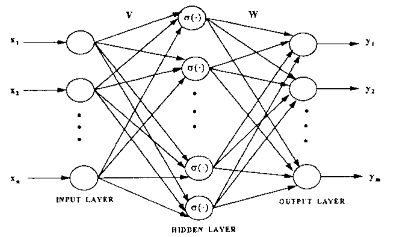

Fig.1: Multilayer Feedforward Neural Network

The main property of a NN we will be concerned with for control purposes is the *function approximation property* [4][7]. Let $f(x)$ be a smooth function from R^n to R^m. Then, it can be shown that, as long as x is restricted to a compact set S of R^n, for some sufficiently large number of hidden layer neurons N_h, there exist weights and thresholds such that any continuous function in a compact set can be represented as

$$f(x) = W^T\sigma(V^T x) + \varepsilon. \tag{2.5}$$

The value of ε is called the *NN functional approximation error.* For any choice of a positive number ε_N, one can find a NN such that $\varepsilon < \varepsilon_N\ \forall x \in S$.

For control purposes, the *ideal* approximating NN weights exist for a specified value of ε_N. Then, an estimate of $f(x)$ can be given by

$$\hat{f}(x) = \hat{W}^T\sigma(\hat{V}^T x) \tag{2.6}$$

where $\hat{W}$, $\hat{V}$ are estimates of the ideal NN weights that are provided by some on-line weight tuning algorithms.

The NN in the remainder of the paper is considered with the first layer weight V fixed. This makes the NN linear in the parameters. It is proven that linearity in the unknown parameters has the so called best-approximation property [6]. This means that given a function f, there always exists a choice of parameters that approximate f better than all other possible choices [17].

Select $V=I$ so that the NN output is

$$y(x) = W^T\sigma(x). \tag{2.7}$$

Then, for suitable NN approximation properties, some conditions (e.g., [20]) must be satisfied by $\sigma(x)$.

Definition 2.1: Let S be a compact simply connected set of R^n, and $\sigma(x): S \to R^{Nh}$, be integrable and bounded. Then $\sigma(x)$ is said to provide a basis for $C^m(S)$ if

1) A constant function on S can be expressed as (2.7) for finite N_h.
2) The functional range of NN (2.7) is dense in $C^m(S)$ for countable N_h.

The issue of selecting the hidden layer units N_h for a given $S \subset R^n$ and ε_N is a topic of current research. It is emphasized that a basis $\sigma(x)$ is not difficult to find. Specifically, the radial basis functions, for instance, provide a universal basis for all smooth nonlinear functions [21]. Sigmoid basis function examples can be found in [9] and [17].

Therefore, there exist constant weights W so that the nonlinear function to be approximated can be represented as

$$f(x) = W^T\sigma(x) + \varepsilon(x) \tag{2.8}$$

where $\|\varepsilon(x)\| < \varepsilon_N(x)$ with the bounding function $\varepsilon_N(x) \in C^1(S)$ known.

2.2 Stability of Systems

Consider the nonlinear system

$$\dot{x} = f(x,u,t),\ y = h(x,t) \tag{2.9}$$

with state $x(t) \in R^n$. We say the solution is *uniformly ultimately bounded* (*UUB*) if there exists a compact set $U \subset R^n$ such that for all $x(t_0)=x_0 \in U$, there exists an $\varepsilon>0$ and a number $T(\varepsilon, x_o)$ such that $\|x(t)\| < \varepsilon$ for all $t \ge t_0+T$ [11].

2.3 Robot Arm Dynamics and Its Properties

The dynamics of an n-link robot maninpulator may be expressed in the Lagrange form [11]

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau \tag{2.10}$$

with $q(t) \in R^n$ the joint variable vector, $M(q)$ the inertia matrix, $V_m(q,\dot{q})$ the coriolis/centripetal matrix, $G(q)$ the gravity vector, $F(\dot{q})$ the friction, and $\tau(t)$ the control input torque.

Given a desired arm trajectory $q_d(t) \in R^n$ the tracking error is

$$e(t) = q_d(t) - q(t) \tag{2.11}$$

and the filtered tracking error is defined as

$$r = \dot{e} + \Lambda e \tag{2.12}$$

where $\Lambda = \Lambda^T > 0$. The arm dynamics may be written as

$$M\dot{r} = -V_m r - \tau + f \tag{2.13}$$

where the nonlinear function is

$$f(x) = M(q)(\ddot{q}_d + \Lambda\dot{e}) + V_m(q,\dot{q})(\dot{q}_d + \Lambda e) + G(q) + F(\dot{q}) \tag{2.14}$$

and, for instance, $x = [e^T,\ \dot{e}^T,\ q_d^T,\ \dot{q}_d^T,\ \ddot{q}_d^T]^T$.

Define now a control input torque as

$$\tau_o = \hat{f} + K_v r \tag{2.15}$$

with $\hat{f}(x)$ an estimate of $f(x)$ and a gain matrix $K_v = K_v^T > 0$. The closed-loop system becomes

$$M\dot{r} = -(K_v + V_m)r + \tilde{f}. \tag{2.16}$$

The following properties of the robot dynamics are required for the subsequent development [11].

Property 1: $M(q)$ is a positive definite symmetric matrix defined by $M_m \le \|M(q)\| \le M_M$ with M_m, $M_M > 0$ known constants.

Property 2: The matrix $\dot{M} - 2V_m$ is skew symmetric [22].

Property 3: $V_m(q,\dot{q})$ is bounded by $v_b(q)\|\dot{q}\|$.

Property 4: The matrix of Christoffel symbols, $V_m(q,\dot{q})$ satisfies

$$V_m(q,x)y = V_m(q,y)x \qquad \forall x,y \in R^n \tag{2.17}$$

where a proper definition of Christoffel coefficients can be found in [11] and [22].

3. Output Feedback Controller Design with a Dynamic NN Observer

In this section, an nonlinear adaptive observer is proposed using the function approximation property of NN. The overall structure of the observer formulates a dynamic NN, as shown in Fig.2. Then, we combine the dynamic NN observer with a static NN that functions as a controller.

3.1 NN Observer Structure and Error Dynamics

We assume the joint displacements q as output variables of the robot system (2.10) and define

$$h(x_1,\hat{x}_2) = V_m(x_1,\hat{x}_2)\hat{x}_2 + F_v\hat{x}_2 + G(x_1). \tag{3.1}$$

It will be assumed that the inertia matrix $M(q)$ is known. It is usual to have uncertainty in the Coriolis terms $V_m(x_1,\hat{x}_2)$, which are difficult to compute, and the viscous friction terms F_v, which may have a complicated terms. Therefore, $h(x_1,\hat{x}_2)$ will be assumed unknown. We introduce the state vectors $x_1 = q$, $x_2 = \dot{q}$, and they will be referred to interchangeably .The state $\hat{x}$ denotes the estimate of the actual state x.

The $h(x_1,\hat{x}_2)$ can be represented by a NN for some constant "ideal" weights W and some number of input basis functions, i.e.

$$h(x_1,\hat{x}_2) = W_o^T\sigma_o(\hat{x}) + \varepsilon_o(\hat{x}). \tag{3.2}$$

Subscript "c" and "o" denotes the controller and the observer, respectively. The NN functional estimate of (3.2) is given by

$$\hat{h}(x_1,\hat{x}_2) = \hat{W}_o^T\sigma_o(\hat{x}) \tag{3.3}$$

with $\hat{W}_o$ the current values of the NN weights as provided by the tuning algorithm. With W_o the ideal weights required in (3.2), define the weight estimation error as

$$\tilde{W}_o = W_o - \hat{W}_o. \tag{3.4}$$

Assume the ideal weights are bounded by known values so that

$$\|W\|_{o,F} \le W_{o,M}. \tag{3.5}$$

The manipulator dynamics (2.10) can be written as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= M^{-1}(x_1)[\tau - V_m(x_1,x_2)x_2 - F_v x_2 - G(x_1)]. \end{aligned} \tag{3.6}$$

Defining the output observation error as $\tilde{x}_1 = x_1 - \hat{x}_1$ and $\tilde{x}_2 = x_2 - \hat{x}_2$, the proposed observer is described by

$$\begin{aligned} \dot{\hat{x}}_1 &= \hat{x}_2 + k_D\tilde{x}_1 \\ \dot{\hat{z}}_2 &= M^{-1}(x_1)[\tau - \hat{W}_o\sigma_o(\hat{x}) + K_P\tilde{x}_1 + v_o(t)] \\ \hat{x}_2 &= \hat{z}_2 + k_{P2}\tilde{x}_1 \end{aligned} \tag{3.7}$$

where k_{P2} and $k_D>0$, $K_P=K_P^T>0$, $\hat{W}_o^T\sigma_o(\hat{x})$ is given in (3.3), and $v_o(t)$ is a function to be defined later that provides robustness in the face of NN functional reconstruction error and the bounded disturbances. The proposed observer formulates a dynamic NN system with a feed-forward NN inserted in the feedback path, of the form considered in [14]; it is shown in Fig.2

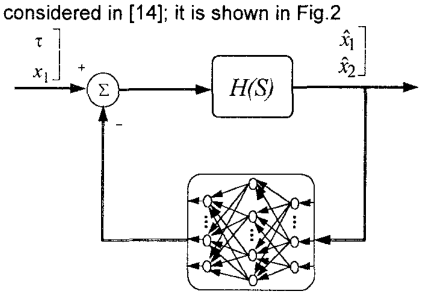

Fig.2: Representation of a Dynamic NN

The observer error dynamics is obtained by subtracting (3.7) from (3.6)

$$\begin{aligned} \dot{\tilde{x}}_1 &= \tilde{x}_2 - k_D\tilde{x}_1 \\ \dot{\tilde{x}}_2 &= M^{-1}[-(V_m + F_v)x_2 - G(x_1) - \hat{W}_o\sigma_o(\hat{x}) - K_P\tilde{x}_1 - \varepsilon_o - v_o(t)] - k_{P2}\dot{\tilde{x}}_1. \end{aligned} \tag{3.8}$$

Adding and subtracting the $h(x_1,\hat{x}_2)$ and then applying (3.3) yields

$$\begin{aligned} \dot{\tilde{x}}_1 &= \tilde{x}_2 - k_D\tilde{x}_1 \\ \dot{\tilde{x}}_2 &= M^{-1}[-(V_m + F_v)x_2 - G(x_1) - \tilde{W}_o\sigma_o(\hat{x}) + h - K_P\tilde{x}_1 - \varepsilon_o - v_o(t)] - k_{P2}\dot{\tilde{x}}_1. \end{aligned} \tag{3.9}$$

Owing to the property of (2.17) and using the relation $\hat{x}_2 = x_2 - \tilde{x}_2$, (3.1) can be written as

$$h(x_1,\hat{x}_2) = V_m(x_1,x_2)\hat{x}_2 - V_m(x_1,\hat{x}_2)\tilde{x}_2 + F_v\hat{x}_2 + G(x_1). \tag{3.10}$$

Substituting (3.10) into (3.9) yields the error dynamics

$$\begin{aligned} M\dot{\tilde{x}}_2 = &-K_P\tilde{x}_1 - V_m\tilde{x}_2 - V_m(x_1,\hat{x}_2)\tilde{x}_2 - \tilde{W}_o^T\sigma_o(\hat{x}) \\ &- F_v\tilde{x}_2 - k_{P2}M\tilde{x}_1 - \varepsilon_o - v_o(t). \end{aligned} \tag{3.11}$$

The proposed NN output feedback control structure with a dynamic NN observer is shown in Fig. 3.

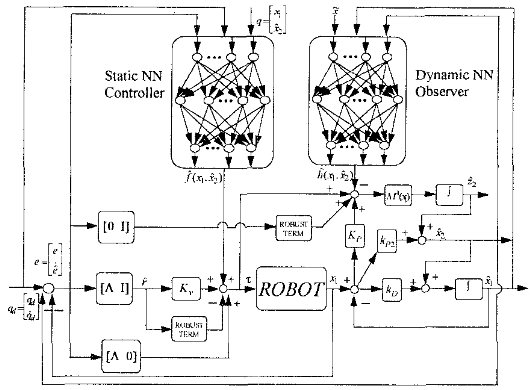

Fig. 3: The NN Output Feedback Control Structure

3.2 Controller Structure and Error Dynamics

The nonlinear robot function (2.14) can be represented by

$$f(x) = W_c^T\sigma_c(x) + \varepsilon_c(x) \tag{3.12}$$

where, for instance, x can be found in (2.14). Define the NN functional estimate of (2.14)

$$\hat{f}(\hat{x}) = \hat{W}_c^T\sigma_c(\hat{x}) \tag{3.13}$$

with $\hat{W}_c$ the current values of the controller NN weights as provided by the tuning algorithm and, for example, $\hat{x} = [e^T\ \dot{\hat{e}}^T\ q_d^T\ \dot{q}_d^T\ \ddot{q}_d^T]^T$ with $\dot{\hat{e}} = \dot{q}_d - \hat{x}_2$. The same arguments as (3.4) and (3.5) will be employed for the controller.

Define the estimated filtered error as

$$\hat{r} = \dot{\hat{e}} + \Lambda e = r + \tilde{x}_2. \tag{3.14}$$

Now, select the control input with robustifying terms as

$$\tau = \hat{W}_c^T\sigma_c(\hat{x}) + K_v\hat{r} + \Lambda e - v_c(t). \tag{3.15}$$

Substituting (3.15) into (2.13) and using (3.14), the closed-loop error dynamics become

$$M\dot{r} = -[V_m + K_v]\hat{r} + V_m\tilde{x}_2 + M\dot{\tilde{x}}_2 - \Lambda e + W_c^T\sigma_c(x) - \hat{W}_c^T\sigma_c(\hat{x}) + \varepsilon_c + v_c(t). \tag{3.16}$$

The following design steps culminate in finding tuning algorithms and determining expressions for $v(t)$, such that the time derivative of a certain Lyapunov function is negative semidefinite.

The input layer output error with sigmoid activation for a given x is defined as

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(x) - \sigma(\hat{x}). \tag{3.17}$$

The Taylor series expansion for a given x is

$$\sigma(x) = \sigma(\hat{x})\big|_{x_2=\hat{x}_2} + \dot{\sigma}(\hat{x})\big|_{x_2=\hat{x}_2}\tilde{x}_2 + O(\tilde{x}_2)^2 \tag{3.18}$$

with $\dot{\sigma}(\hat{z}) = d\sigma(z)/\sigma(z)\big|_{z=\hat{z}}$ and $O(z)^2$ denoting higher order terms in the Taylor series. Denoting $\dot{\hat{\sigma}} = \dot{\sigma}(\hat{x})\big|_{x_2=\hat{x}_2}$, we have

$$\tilde{\sigma} = \dot{\hat{\sigma}}\tilde{x}_2 + O(\tilde{x}_2)^2. \tag{3.19}$$

Fact 1: For sigmoid and tanh activation functions, the higher-order terms in the Taylor series are bounded by

$$\left\|O(\tilde{x}_2)^2\right\| \le c_1 + c_2\left\|\tilde{x}_2\right\| \qquad c_i>0, \text{ computable.} \tag{3.20}$$

Now adding and subtracting $W_c^T\hat{\sigma}_c$ from (3.16) yield

$$M\dot{\hat{r}} = -[V_m + K_v]\hat{r} + V_m\tilde{x}_2 + M\dot{\tilde{x}}_2 - \Lambda e + \tilde{W}_c^T\hat{\sigma}_c + W_c^T\tilde{\sigma}_c + \varepsilon_c + v_c \tag{3.21}$$

with $\hat{\sigma}$ and $\tilde{\sigma}$ defined in (3.17). Adding and subtracting $\hat{W}_c^T\tilde{\sigma}_c$ yields

$$M\dot{\hat{r}} = -[V_m + K_v]\hat{r} + V_m\tilde{x}_2 + M\dot{\tilde{x}}_2 - \Lambda e + \tilde{W}_c^T\hat{\sigma}_c + \hat{W}_c^T\tilde{\sigma}_c + \tilde{W}_c^T\tilde{\sigma}_c + \varepsilon_c + v_c \,. \tag{3.22}$$

According to the Taylor series approximation in (3.19) for $\tilde{\sigma}_c$, the closed-loop error system is

$$M\dot{\hat{r}} = -[V_m + K_v]\hat{r} + V_m\tilde{x}_2 + M\dot{\tilde{x}}_2 - \Lambda e_c + v_c$$
$$+\tilde{W}_c^T\hat{\sigma}_c + \hat{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2 + w_1 + \varepsilon_c. \tag{3.23}$$

where the disturbance terms are

$$w_1(t) = \tilde{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2 + W_c^T O(\tilde{x}_2)^2. \tag{3.24}$$

Manipulating the disturbance terms and using the expression (3.14), we have the final error system for the proof of the combined observer-controller system

$$M\dot{\hat{r}} = -[V_m + K_v]\hat{r} + V_m\tilde{x}_2 + M\dot{\tilde{x}}_2 - \Lambda e + \tilde{W}_c^T(\hat{\sigma}_c - \dot{\hat{\sigma}}_c\hat{x}_2)$$
$$+\hat{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2 + \varepsilon_c + w(t) + v_c(t) \tag{3.25}$$

where the disturbance terms are given by

$$w(t) = \tilde{W}_c^T\dot{\hat{\sigma}}_c\dot{q}_d - \tilde{W}_c^T\dot{\hat{\sigma}}_c\dot{e} + W_c^T O(x_1, \tilde{x}_2)^2. \tag{3.26}$$

The closed-loop error dynamics (3.25) includes the observer error dynamics, whence comes the interaction between the observer and the controller.

Fact 2: The disturbance term $w(t)$ in (3.26) is bounded by

$$\|w(t)\| \le c_3 + c_4\|\tilde{W}_c\|_F + c_5\|\tilde{W}_c\|_F\|\hat{r}\| + c_6\|\tilde{x}_2\| + c_7\|\tilde{W}_c\|_F\|\tilde{x}_2\| \tag{3.27}$$

where c_i are computable positive constants.

3.3 Weight Updates for Guaranteed Tracking Performance

For the design of the observer-controller system, it seems natural to take a Lyapunov function candidate that consists of a combination of the Lyapunov functions for the observer and the controller system. Using this technique we obtain the following theorem that shows how to tune the NN weights to guarantee stability.

Assume that the observer and the controller gains are chosen such that

$$K_{p,m} > (8\beta + 16k_{P2}k_D M_M + \gamma k_D)/(k_D - 8) \tag{3.28a}$$
$$k_{P2} > (8k_z + \Lambda_M + \gamma + c_8 - 32F_{v,m})/(32M_m - (k_D+8)M_M) + v_b Q_d \tag{3.28b}$$
$$K_{v,m} > (K_{P,M} + k_{P2}k_D M_M + k_z + \Lambda_M + 32k_{P2}M_M)/32 \tag{3.28c}$$
$$\Lambda_m^2 > 8\Lambda_M(k_z + 1) \tag{3.28d}$$
$$\alpha > \gamma(k_D + 32)/4 \tag{3.28e}$$

with γ and c_8 defined in (A.8) and β defined as $K_{P,M}$-$K_{P,m}$.

Theorem 1: Suppose that $\|\dot{q}_d(t)\| \le Q_d$, for any $t \ge 0$. Under the condition (3.28), consider the combined system of the observer (3.7) and the controller (3.15) with the robustifying terms

$$v_o(t) = k_z\hat{e} \tag{3.29}$$

$$v_c(t) = -k_{z1}\frac{\hat{r}}{\|\hat{r}\|} - k_{z2}(W_M + \|\hat{W}_c\|_F)\frac{\hat{r}}{\|\hat{r}\|} - k_{z3}(W_M + \|\hat{W}_c\|_F)\hat{r} \tag{3.30}$$

where $k_{z1} \ge c_3$, $k_{z2} \ge c_4$ and $k_{z3} \ge c_5$. Let weight tunings for the NN observer and the controller be provided by

$$\dot{\hat{W}}_o = -k_D F_o\sigma_o(\hat{x})\tilde{x}_1^T - F_o\sigma_o(\hat{x})\hat{r}^T - \kappa_o F_o\|\tilde{x}_1\|\hat{W}_o - \alpha\hat{W}_o \tag{3.31}$$

$$\dot{\hat{W}}_c = F_c\sigma_c(\hat{x})\hat{r}^T - F_c\dot{\sigma}_c(\hat{x})\hat{x}_2\hat{r}^T - \kappa_c F_c\|\hat{r}\|\hat{W}_c \tag{3.32}$$

with any constant matrices $F=F^T>0$, $\kappa>0$ a design parameter, and $\alpha>0$ given in (3.28). Then, the combined system is *locally uniformly ultimately bounded* with region of attraction for $d = [\tilde{x}_1^T,\ \tilde{x}_2^T,\ \tilde{W}_o^T,\ e^T,\ \hat{r}^T,\ \tilde{W}_c^T]^T$ given by

$$S = \left\{ d(0) \in R^{4n+2p} : \|d(0)\| < \sqrt{\frac{P_m}{P_M}} min\{\delta_2/\zeta_1,\ \delta_5/\zeta_2\} \right\} \tag{3.33}$$

with δ_2, δ_5 defined in (A.10), $P = diag[K_P,\ M(q),\ F_o^{-1},\ \Lambda,\ M(q),\ F_c^{-1}]$, $\zeta_1 = 1.25v_b + 1.25v_b\Lambda_M + 0.25(c_7 + \gamma')$, $\zeta_2 = v_b + v_b\Lambda_M + c_7 + \gamma'$.

Proof: See Appendix A.

Remarks:

1) No preliminary off-line tuning for the NN weights is needed; the NN observer-controller converges to small tracking errors and bounded weight errors on-line in real time.
2) Initializing the NN weights is easy. They are simply initialized to zero. The region of attraction can be enlarged by both the observer robustifying gain k_z and either the observer gain k_{P2} or the controller gain $K_{v,m}$ depending on (3.33).
3) Using the concept of the extension to the Lyapunov theory [14], it is shown that $\dot{L}$ is negative if $\|\tilde{x}_1\|, \|\tilde{x}_2\|, \|\tilde{W}_o\|$ and $\|\hat{r}\|$ are above some specific bounds (B.12), but the region is also upper bounded by the attractive region (3.33). So we can conclude that the error system is *locally uniformly ultimately bounded.*
4) The second terms in (3.31) and (3.32) ensures the interaction between the observer and the controller. The third terms are corresponding to *e*-modification [14] and the last term in (3.31) to σ-modification [8].

4. Simulation Results

To verify the theoretical analysis, we considered the problem of the trajectory tracking by using the output feedback control law (3.15) along with the observer (3.7). A planar two-link arm appears in Fig. 4; the dynamics are given in [11] with $\ell_1=\ell_2=1m$, $m_1=1.0$, and $m_2=2.3Kg$.

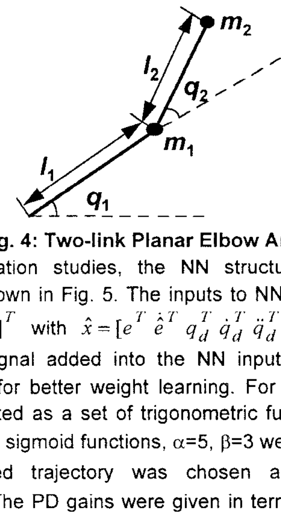

Fig. 4: Two-link Planar Elbow Arm

For simulation studies, the NN structure and sigmoid functions are shown in Fig. 5. The inputs to NN were chosen as $X = [\hat{x}^T\ \varphi(\hat{x})^T]^T$ with $\hat{x} = [e^T\ \hat{e}^T\ q_d^T\ \dot{q}_d^T\ \ddot{q}_d^T]^T$ and $\varphi(\hat{x})$ a preprocessed signal added into the NN input to increase the signal richness for better weight learning. For example, in [12] $\varphi(\hat{x})$ is suggested as a set of trigonometric function for robotic systems. For the sigmoid functions, α=5, β=3 were used.

The desired trajectory was chosen as $q_d(t)$=[*0.4sin(t) 0.4cos(t)*](rad). The PD gains were given in terms of K_v=*diag*[20] and Λ=*diag*[15]. The observer gains were set to K_P=*diag*[2000] and k_D=20. The initial conditions of all the states were equal to zero, so that a non-zero error affected the performance of the NN observer.The simulation was performed over 20 seconds to show the long-term behavior of the closed-loop system. The performance of the controller appears in Fig. 6, showing that all signals in the closed-loop system are bounded.

To verify the robustness of our algorithm, we dropped some terms in *M(q)*, keeping the same parameters in the above case. The performance still shows the same characteristics, as shown in Fig. 7. But note that the torque for each joint changes to compensate the variation in *M(q)* in the observer. This illustrates the interaction between the observer and the controller through the simultaneous weight tuning.

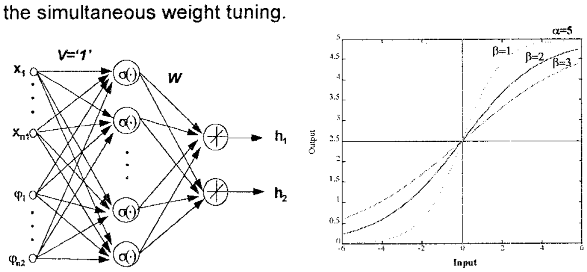

Fig. 5: The NN Structure and Sigmoid Function [$\alpha/(1+exp(-x/\beta))$]

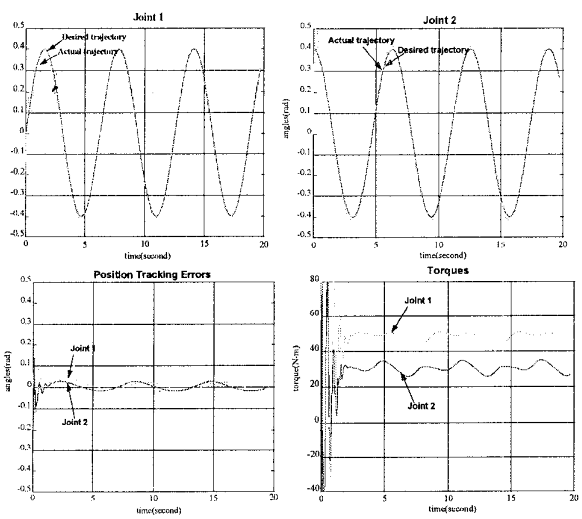

Fig. 6: Performance of Output Feedback Controller with Exact Inertia Matrix

5. Conclusion

We have presented a robust neural network output feedback control scheme that includes a novel dynamic NN observer for the motion control of the rigid robots. The method does not require the robot dynamics to be exactly known, hence the same NN observer-controller can be applied to any type of rigid robots without any modification. Only the inertia matrix, *M(q)* must be computed. A key point in developing an intelligent control system is the reusability of the low-level controllers, i.e., the same controller works even if the behavior of the system has changed. This is the case of the controller reported in this paper. Compared with adaptive controllers, no linearity in the unknown system parameters is needed and no persistent excitation condition is required. We do not require any off-line "training or learning phase". All error signals in the closed-loop system are guaranteed to be bounded.

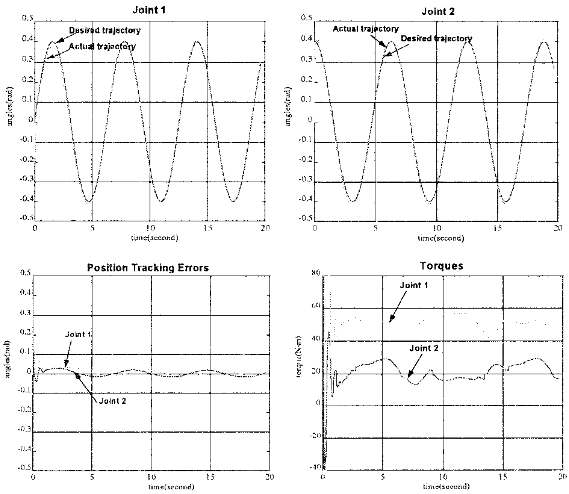

Fig. 7: Performance of Output Feedback Controller with Inexact Inertia Matrix

APPENDIX A: *proof of theorem 1*

Define the Lyapunov function candidate

$$L(d,t)=\frac{1}{2}d(t)^T Pd(t) \tag{A.1}$$

with $d=[\tilde{x}_1^T,\ \tilde{x}_2^T,\ \tilde{W}_o^T,\ e^T,\ \hat{r}^T,\ \tilde{W}_c^T]^T$ and $P=diag[K_P,\ M(q),\ F_o^{-1},\ \Lambda,\ M(q),\ F_c^{-1}]$. It follows that

$$\frac{1}{2}P_m\|d(t)\|^2 \le L(d,t) \le \frac{1}{2}P_M\|d(t)\|^2. \tag{A.2}$$

Differentiating (A.1) along (3.11) and (3.25), then applying *property 2* gives

$$\begin{aligned}\dot{L}=&-k_D\tilde{x}_1{}^T K_P\tilde{x}_1+\tilde{x}_2{}^T\{-V_m(x_1,\hat{x}_2)\tilde{x}_2-F_v\tilde{x}_2-k_{P2}M\dot{\tilde{x}}_1\\&-\varepsilon_o-v_o(t)\}-\dot{\tilde{x}}_1\tilde{W}_o^T\hat{\sigma}_o+\hat{r}^T\{-K_v\hat{r}-V_m(x_1,\hat{x}_2)\tilde{x}_2\\&-F_v\tilde{x}_2+\hat{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2-K_p\tilde{x}_1-k_{P2}M\dot{\tilde{x}}_1-\Lambda e-\varepsilon_o-v_o(t)\\&+\varepsilon_c+w(t)+v_c(t)\}+\dot{e}^T\Lambda e+tr\{\tilde{W}_o^T(-k_D\hat{\sigma}_o\tilde{x}_1^T-\hat{\sigma}_o\hat{r}\\&+F_o^{-1}\dot{\tilde{W}}_o)\}+tr\{\tilde{W}_c^T((\hat{\sigma}_c-\dot{\hat{\sigma}}_c\hat{x}_2)\hat{r}+F_c^{-1}\dot{\tilde{W}}_c\}.\end{aligned} \tag{A.3}$$

As $\dot{e}=\hat{r}-\tilde{x}_2-\Lambda e$ and $\dot{\tilde{x}}_1=\tilde{x}_2-k_D\tilde{x}_1$, (A.3) becomes

$$\begin{aligned}\dot{L}=&-k_D\tilde{x}_1{}^T K_P\tilde{x}_1-\tilde{x}_2{}^T F_v\tilde{x}_2-k_{P2}\tilde{x}_2{}^T M\tilde{x}_2-\Lambda e\Lambda e-\hat{r}^T K_v\hat{r}\\&+\tilde{x}_2{}^T\{-V_m(x_1,\hat{x}_2)\tilde{x}_2+k_{P2}k_D M\tilde{x}_1-\Lambda e-\varepsilon_o-v_o(t)\}-\tilde{x}_2\tilde{W}_o^T\hat{\sigma}_o\\&+k_D\tilde{x}_1\tilde{W}_o^T\hat{\sigma}_o+\hat{r}^T\{-V_m(x_1,\hat{x}_2)\tilde{x}_2-F_v\tilde{x}_2+\hat{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2-K_p\tilde{x}_1\\&-k_{P2}M\tilde{x}_2+k_{P2}k_D M\tilde{x}_1-\varepsilon_o-v_o(t)+\varepsilon_c+w(t)+v_c(t)\}\\&+tr\{\tilde{W}_o^T(-k_D\hat{\sigma}_o\tilde{x}_1^T-\hat{\sigma}_o\hat{r}+F_o^{-1}\dot{\tilde{W}}_o)\}+tr\{\tilde{W}_c^T((\hat{\sigma}_c-\dot{\hat{\sigma}}_c\hat{x}_2)\hat{r}+F_c^{-1}\dot{\tilde{W}}_c\}.\end{aligned} \tag{A.4}$$

Substituting (3.29) - (3.32) into (A.4) yields

$$\begin{aligned}\dot{L}\le&-k_D K_{P,m}\|\tilde{x}_1\|^2-(F_{v,m}+k_{P2}M_m)\|\tilde{x}_2\|^2-\Lambda_m^2\|e\|^2-K_{v,m}\|\hat{r}\|^2-\alpha\|\tilde{W}_o\|_F^2\\&+\tilde{x}_2{}^T\{-V_m(x_1,\hat{x}_2)\tilde{x}_2+k_{P2}k_D M\tilde{x}_1-\Lambda e-\varepsilon_o-k_z\dot{\hat{e}}\}-\tilde{x}_2\tilde{W}_o^T\hat{\sigma}_o+k_D\tilde{x}_1\tilde{W}_o^T\hat{\sigma}_o\\&+\hat{r}^T\{-V_m(x_1,\hat{x}_2)\tilde{x}_2-F_v\tilde{x}_2+\hat{W}_c^T\dot{\hat{\sigma}}_c\tilde{x}_2-K_p\tilde{x}_1-k_{P2}M\tilde{x}_2+k_{P2}k_D M\tilde{x}_1\\&-\varepsilon_o-k_z\dot{\hat{e}}+\varepsilon_c+c_6\|\tilde{x}_2\|+c_7\|\tilde{x}_2\|\|\tilde{W}_c\|_F\}+\kappa_o\|\tilde{x}_1\|\|\tilde{W}_o\|_F(W_{o,M}-\|\tilde{W}_o\|_F)\\&+\alpha W_{o,M}\|\tilde{W}_o\|_F+\kappa_c\|\hat{r}\|\|\tilde{W}_c\|_F(W_{c,M}-\|\tilde{W}_c\|_F)\end{aligned} \tag{A.5}$$

where the *fact2* (3.27) and the inequality have been used

$$tr(\tilde{W}^T(W-\hat{W}))\le\|\tilde{W}\|_F(W_M-\|\tilde{W}\|_F). \tag{A.6}$$

As $\hat{x}_2 = \dot{q}_d - \hat{r} + \Lambda e$ and *Property 4* (2.17), we have

$$V_m(x_1,\hat{x}_2)\tilde{x}_2 = V_m(x_1,\dot{q}_d)\tilde{x}_2 - V_m(x_1,\hat{r})\tilde{x}_2 + V_m(x_1,\Lambda e)\tilde{x}_2. \tag{A.7}$$

Using (A.7) and $\dot{\hat{e}} = \hat{r} - \Lambda e = \dot{e} + \tilde{x}_2$ yields

$$\begin{aligned}\dot{L} \le & -k_D K_{P,m}\|\tilde{x}_1\|^2 - \{F_{v,m} + k_{P2}M_m - v_b(Q_d + \|\hat{r}\| + \Lambda_M\|e\|)\}\|\tilde{x}_2\|^2 - \alpha\|\tilde{W}_o\|_F^2 \\ & - \Lambda_m^2\|e\|^2 - (K_{v,m} + k_z)\|\hat{r}\|^2 + k_{P2}k_D M_M\|\tilde{x}_1\|\|\tilde{x}_2\| + K_{p,M}\|\hat{r}\|\|\tilde{x}_1\| \\ & + k_{P2}k_D M_M)\|\hat{r}\|\|\tilde{x}_1\| + k_{P2}M_M\|\hat{r}\|\|\tilde{x}_2\| + k_z\|\hat{r}\|\|\tilde{x}_2\| + \gamma k_D\|\tilde{x}_1\|\|\tilde{W}_o\|_F \\ & + k_z\Lambda_M\|\hat{r}\|\|e\| + \Lambda_M\|\tilde{x}_2\|\|e\| + \gamma\|\tilde{x}_2\|\|\tilde{W}_o\|_F + \|\hat{r}\|\|\tilde{x}_2\|\{v_b\|\hat{r}\| + v_b\Lambda_M\|e\| \\ & + (\gamma' + c_7)\|\tilde{W}_c\|_F + c_8\} + \varepsilon_{o,N}\|\tilde{x}_2\| + (\varepsilon_{o,N} + \varepsilon_{c,N})\|\hat{r}\| + \kappa_o\|\tilde{x}_1\|\|\tilde{W}_o\|_F \\ & (W_{o,M} - \|\tilde{W}_o\|_F) + \alpha W_{o,M}\|\tilde{W}_o\|_F + \kappa_c\|\hat{r}\|\|\tilde{W}_c\|_F(W_{c,M} - \|\tilde{W}_c\|_F).\end{aligned} \tag{A.8}$$

The inequality $\|\hat{W}\|_F \le W_M + \|\tilde{W}\|_F$ has been used with $c_8 = c_6 + v_b Q_d + F_{v,m} + \gamma' W_{c,M}$ and γ , γ' bounded value of $\hat{\sigma}$,$\dot{\hat{\sigma}}$ respectively. For real numbers *a* and *b*, the inequality (A.9) exists.

$$ab \le a^2/32 + 8b^2. \tag{A.9}$$

Thus, we get the following result after manipulating (A.8)

$$\begin{aligned}\dot{L} \le & -\delta_1\|\tilde{x}_1\|^2 - (\delta_2 - f_1(\|d\|))\|\tilde{x}_2\|^2 - \delta_3\|\tilde{W}_o\|_F^2 - (\delta_5 - f_2(\|d\|))\|\hat{r}\|^2 \\ & - \delta_4\|e\|^2 + \varepsilon_{o,N}\|\tilde{x}_2\| + (\varepsilon_{o,N} + \varepsilon_{c,N})\|\hat{r}\| + \alpha W_{o,M}\|\tilde{W}_o\|_F + \kappa_o\|\tilde{x}_1\|\|\tilde{W}_o\|_F \\ & (W_{o,M} - \|\tilde{W}_o\|_F) + \kappa_c\|\hat{r}\|\|\tilde{W}_c\|_F(W_{c,M} - \|\tilde{W}_c\|_F)\end{aligned} \tag{A.10}$$

with $\delta_1 = k_D K_{p,m} - 8(K_{P,M} + 2k_{P2}k_D M_M) - \gamma k_D$,

$\delta_2 = F_{v,m} + k_{P2}M_m - v_b Q_d - \{k_{P2}k_D M_M + 8(k_{P2}M_M + k_z + c_8) + \Lambda_M - \gamma\}/32$,

$\delta_4 = \Lambda_m^2 - 8\Lambda_M(k_z + 1)$, $\delta_3 = \alpha - \gamma k_D/4 - 8\gamma$,

$\delta_5 = K_{v,m} - k_{P2}M_M - c_8 - (K_{p,M} + k_{P2}k_D M_M + k_z\Lambda_M)/32$,

$f_1(\|d\|) = \{5v_b\|\hat{r}\| + 5v_b\Lambda_M\|e\| + (c_7 + \gamma')\|\tilde{W}_c\|_F\}/4$,

$f_2(\|d\|) = v_b\|r\| + v_b\Lambda_M\|e\| + (c_7 + \gamma')\|\tilde{W}_c\|_F$.

It is easy to see that (3.28) and (3.33) ensures δ_1, $\delta_2 - f_1(\|d\|)$, δ_3, δ_4 and $\delta_5 - f_2(\|d\|)$ to be positive. Arranging the terms in (A.10), $\dot{L}$ becomes

$$\begin{aligned}\dot{L} \le & -\|\tilde{x}_1\|(\delta_1\|\tilde{x}_1\| - \kappa_o W_{o,M}\|\tilde{W}_o\|_F + \kappa_o\|\tilde{W}_o\|_F^2) - \|\tilde{x}_2\|(\xi_1\|\tilde{x}_2\| - \varepsilon_{o,N}) - \delta_4\|e\|^2 \\ & - \|\tilde{W}_o\|_F(\delta_3\|\tilde{W}_o\|_F - \alpha W_{o,M}) - \|\hat{r}\|\{\xi_2\|\hat{r}\| - \kappa_c W_{c,M}\|\tilde{W}_c\|_F + \kappa_c\|\tilde{W}_c\|_F^2 - (\varepsilon_{o,N} + \varepsilon_{c,N})\}\end{aligned} \tag{A.11}$$

with $\xi_1 = \delta_2 - f_1(\|d\|)$ and $\xi_2 = \delta_5 - f_2(\|d\|)$.

Completing the square terms in (A.11) yields that the right side of (A.10) is negative if besides (3.28) and (3.33), the conditions (A.12) are satisfied.

$$\begin{bmatrix} \|\tilde{x}_1\| \\ \|\tilde{x}_2\| \\ \|\tilde{W}_o\|_F \\ \|\hat{r}\| \end{bmatrix} \ge \begin{bmatrix} \kappa_o W_M^2/4\delta_1 \\ \varepsilon_{o,N}/\xi_1 \\ \alpha W_{o,M}/\delta_3 \\ \{\kappa_c W_{c,M}^2/4 + (\varepsilon_{c,N} + \varepsilon_{o,N})\}/\xi_2 \end{bmatrix} \tag{A.12}$$

The region of attraction is given by (3.33). From (A.2) we can see that $L(d,t)$ is a positive definite decrescent function. Since $\dot{L}(d,t)$ is negative semi-definite for all *d* satisfying (3.28), (3.33) and (A.12), we can conclude that the system is *locally uniformly ultimately bounded*.

REFERENCES

[1] A. R. Barron, "Universal approximation bounds for superposition of a sigmoidal function ," *IEEE Tran. Information Theory,* vol. 39, no. 3, pp. 930-945, 1993.

[2] H. Berghuis and H. Nijmeijer, "A passivity approach to controller-observer design for robots," *IEEE Trans. Robotics. Automat.,* vol. 9, no. 6, pp. 740-754, 1993.

[3] C.Canudas de Wit and N.Fixot, "Adaptive control of robot manipulators via velocity estimated feedback," *IEEE Trans. Automat. Control*, vol. 37, no. pp. 1234-1237, 1992.

[4] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Controls Signals and Systems*, vol. 2, no. 4, pp. 303-314, 1989.

[5] M. Erlic and W.-S. Lu, "A reduced order adaptive velocity observer for manipulator control," *IEEE Trans. Robotics. Automat.,* vol. 11, no. 2, pp. 293-303, 1995.

[6] F. Girosi and T. Poggio, "Networks and the best approximation property," *Biological Cybernetics*, vol. 63, pp. 169-176, 1990.

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

[8] P. A. Ioannou and P. V. Kokotovic, "Instability analysis and the improvement of robustness of adaptive control," *Automatica,* vol. 20, no. 5, pp. 583-594, 1984.

[9] L. K. Jones, "Constructive approximations for neural networks by sigmoidal functions," *Proceedings of the IEEE,* vol. 78, no. 9, pp. 1586-1589, 1990.

[10] E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P. A. Ioannou, "High-order neural network structures for identification of dynamical systems," *IEEE Trans. Neural Network*, vol. 6, no. 2, pp. 422-431, 1995.

[11] F. L. Lewis, C. T. Abdallah and D. M. Dawson, *Control of Robot Manipulators*, MacMillan, New York, 1993.

[12] F. L. Lewis, A. Yesildirek and K. Liu, "Neural net robot controller: structure and stability proofs," *Proc. 31st IEEE Conference on Decision and Control*, pp. 2785-2791, San Antonio, December 1993.

[13] F. L. Lewis, A. Yesildirek and K. Liu, "Neural net robot controller with guaranteed tracking performance," *IEEE Trans.Neural Network*, vol. 6, no. 3, pp. 703-715, 1995.

[14] K. S. Narendra and A.M. Annaswamy, "A new adaptive law for robust adaptation without persistent excitation," *IEEE Trans. Automat. Control*, vol. 32, no.2, pp. 134-145, 1987.

[15] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 4-27, 1990.

[16] S. Nicosia and P.Tomei, "Robot control by using only position measurements," *IEEE Trans. Automat. Cont.,* vol. 35, pp. 1058-1061, 1990.

[17] T. Poggi and F. Girosi, "Networks for approximation and learning," *Proceedings of the IEEE,* vol. 78, no. 9, pp. 1481-1497, 1990.

[18] R. Rajamani and J. K. Hedrick, "Adaptive observers for active automotive suspensions: Theory an experiment," *IEEE Trans. Cont. Syst. Tech.* vol. 3, no. 1, pp. 86-93, 1995.

[19] G. A. Rovithakis and M. A. Christoudoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. Syst. Man, Cybern.,* vol. 24, no. 3, pp. 400-412, 1994.

[20] N. Sadegh, "Nonlinear identification and control via neural networks," *Control systems with Inexact dynamic Models*, DSC-vol.33, ASME Winter Annual Meeting, 1991.

[21] R. M. Sanner and J. J. Slotine, "Stable adaptive control and recursive identification using radial gaussian networks," *Proc. 31th IEEE Conference on Decision and Control*, pp. 2785-2791, San Antonio, December 1991.

[22] J.E.Slotine and W.Li, "On the adaptive control of robot manipulators," *Int.J.Robotics Res.*, vol. 6, no.3, pp. 49-59,1987.

[23] W. Zhu, H. Chen, and Z. Zhang, "A variable structure robot control algorithm with an observer," vol. 8, no. 4, pp. 486-492, 1992.