KYBERNETIKA — VOLUME 33 (1997), NUMBER 1, PAGES 2-16

A NEW METHODOLOGY FOR THE DESIGN OF ADAPTIVE CONTROLLERS USING "STATE-STRICT PASSIVITY": APPLICATION TO NEURAL NETWORK CONTROLLERS

SESH COMMURI AND FRANK L. LEWIS[1]

The notion of passivity has played an important role in extending stability results to systems based on the input-output properties of the system. This approach was also utilized to study the stability properties of interconnected passive systems. In the control of unknown nonlinear dynamical systems, however, passivity properties were studied only as an off-shoot of the resulting controller. In this paper, it is shown that a stronger form of passivity, namely *state-strict passivity*, is required to prove guaranteed tracking performance and internal stability for a class of nonlinear systems without standard observability (i. e. "persistence of excitation") conditions. It is shown that this property can be made a design objective in the design of neural network controllers for the control of unknown nonlinear systems that satisfy certain assumptions on the system structure. This yields "robust" neural network controllers that do not require persistency of excitation or the often tedious computations of the regression vector.

1. INTRODUCTION

Real-time control of nonlinear plants with unknown dynamics remains a very challenging area of research. Traditionally, the plant dynamics were first modeled and verified through off-line experimentation. The control was then designed using linear system design techniques or geometric techniques with linear analogues. Thus, feedback linearization is a first step in controls design for nonlinear systems. The disadvantage, however, is that the method is suitable only for systems described by an accurate model. The results for systems with unknown dynamics were at first limited by-and-large to *ad hoc* techniques and simulations involving assumptions such as certainty equivalence. These approaches are limited by the complexity of the model and cannot accommodate the variation of system parameters. This has resulted in the development of controllers that can learn the process dynamics, as well as, adapt to parametric changes in the system. Adaptive feedback linearization plays a very important role in the control of unknown nonlinear systems that are

[1]Research supported by NSF Grant IRI-9216545 and EPRI Grant RP8030-09.

feedback linearizable [1, 9]. Since the plant inputs and outputs are used to tune the adaptation parameters, a lot of research was directed towards the study of the convergence of adaptive algorithms based on the input-output properties of the system [1, 8, 9, 17]. In this context, *Passivity* properties of the resulting controller were used to show the convergence of the adaptation algorithm [7, 11, 12]. In this paper, it is shown that if the system and the adaptive network satisfy a stronger condition than passivity, namely *state-strict passivity*, then this guarantees the internal stability of the overall system. This is the first work to our knowledge that brings out this important relationship between the input-out properties and the internal stability of the interconnected system. This result is shown to be of fundamental importance in the development of a new methodology for designing adaptive feedback-linearizing controllers for a class of nonlinear systems. In fact, it allows the design of controllers that do not require persistence of excitation (e. g. "observability") conditions.

The remainder of the paper is organized as follows. A brief background on nonlinear dynamical systems is given in Section 2, In Section 3, results from literature on passivity are presented and sufficient conditions derived to prove the stability of the closed-loop system. In Section 4, the formulation of Section 2 is shown to satisfy the "state-strict passivity" property, and examples presented on designing adaptive controllers and controllers based on neural networks. The performance of these controllers is demonstrated through simulation examples in Section 5 and the results are summarized in Section 6.

2. BACKGROUND ON NONLINEAR DYNAMICAL SYSTEMS

Consider the multi-input multi-output system whose state-space representation is given in the Brunovský canonical form as

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
&\vdots \\
\dot{x}_{n_1} &= f_1(\underline{x}) + u_1 + d_1 \\
\dot{x}_{n_1+1} &= x_{n_1+2} \\
\dot{x}_{n_1+2} &= x_{n_1+3} \\
&\vdots \\
\dot{x}_{n_1+n_2} &= f_2(\underline{x}) + u_2 + d_2 \\
\dot{x}_{n_1+n_2+\cdots+n_{m-1}+1} &= x_{n_1+n_2+\cdots+n_{m-1}+2} \\
\dot{x}_{n_1+n_2+\cdots+n_{m-1}+2} &= x_{n_1+n_2+\cdots+n_{m-1}+3} \\
&\vdots \\
\dot{x}_n &= f_m(\underline{x}) + u_m + d_m
\end{aligned} \tag{1}$$

with the output equation given as

$$y = \begin{bmatrix} x_1 \\ x_{n_1+1} \\ \vdots \\ x_{n_1+n_2+\cdots+n_{m-1}+1} \end{bmatrix}. \tag{2}$$

It is assumed that $d = [d_1, d_2, \ldots, d_m]^T$ is an unknown disturbance with known upper bound so that $\|d\| < b_d$, $\underline{x} = [x_1, x_2, \ldots, x_n]^T \in R^n$, and $f = [f_1, f_2, \ldots, f_m]^T : R^n \to R^m$ is a smooth vector function.

Definition 1. The solution of the system (1) and (2) is said to be *uniformly ultimately bounded* (UUB) if for any compact subset $U \subset R^n$, there exists a domain of attraction $U_0 \subset U$, a constant $\varepsilon > 0$ and a number $T(\varepsilon, \underline{x}_0)$ such that for all $\underline{x}(t_0) \in U_0$, $\underline{x}(t) \in U \; \forall t$, and $\|\underline{x}(t)\| < \varepsilon$ for all $t \geq t_1 + T$.

2.1. Output tracking problem

Given the system (1) and (2), it is required to manufacture a bounded control input $u(t) = [u_1, u_2, \ldots, u_m]^T$ such that the output $y(t)$ of the system tracks a specified desired output $y_d(t) = [y_{d_1}(t), y_{d_2}(t), \ldots, y_{d_m}(t)]^T$ while ensuring that the states $\underline{x}(t)$ are bounded. It is assumed that the desired output is smooth so that derivatives of all orders exist and the desired output and all its derivatives are bounded by a known constant γ, that is

$$\left\| \begin{bmatrix} y_d \\ \dot{y}_d \\ \vdots \\ \overset{(n-1)}{y_d} \end{bmatrix} \right\| \leq \gamma, \tag{3}$$

where $\overset{(n-1)}{y_d}$ denotes the $(n-1)$-st derivative of y_d.

Define the *tracking error* as $e = y - y_d$, that is

$$\begin{aligned} e_1 &= x_1 - y_{d_1} \\ e_2 &= x_{n_1+1} - y_{d_2} \\ &\vdots \\ e_m &= x_{n_1+n_2+\cdots+n_{m-1}+1} - y_{d_m}. \end{aligned} \tag{4}$$

Then using (1), the error dynamics in (4) can be expressed as

$$\begin{aligned} \overset{(n_1)}{e_1} &= f_1(\underline{x}) + d_1 + u_1 - \overset{(n_1)}{y_{d_1}} \\ \overset{(n_2)}{e_2} &= f_2(\underline{x}) + d_2 + u_2 - \overset{(n_2)}{y_{d_2}} \\ &\vdots \\ \overset{(n_m)}{e_m} &= f_m(\underline{x}) + d_m + u_m - \overset{(n_m)}{y_{d_m}}. \end{aligned} \tag{5}$$

Define the *filtered tracking error* $r(t)$ with components

$$r_i = \overset{(n_i-1)}{e_i} + \lambda_{i,1} \overset{(n_i-2)}{e_i} + \cdots + \lambda_{i,n_i-1} e_i, \quad 1 \leq i \leq m. \tag{6}$$

Selecting now the control inputs as

$$u_i = -\hat{f}_i(\underline{x}) - K_{\nu_i} r_i - \left[\lambda_{i,n_i-1} e_i + \lambda_{i,n_i-2} \dot{e}_i + \cdots + \lambda_{i,1} \overset{(n_i-1)}{e_i} \right] + \overset{(n_i)}{y_{d_i}}, \quad \text{for } 1 \leq i \leq m \tag{7}$$

the *filtered error system* can be expressed in the form

$$\dot{r}_i \;=\; -K_{v_i} r_i + \tilde{f}_i(\underline{x}) + d_i, \quad 1 \le i \le m. \tag{8}$$

In (7), $\hat{f}_i(\underline{x})$ denotes an *estimate* of $f_i(x)$, to be subsequently provided by a Adaptive network. The *functional estimation error* is $\tilde{f}_i(\underline{x}) = f_i(\underline{x}) - \hat{f}_i(\underline{x})$. The coefficients $\lambda_{i,j}$ in (7) are selected such that

$$\lambda_{i,n_i-1} e_i + \lambda_{i,n_i-2} \dot{e}_i + \cdots + \lambda_{i,1} \overset{(n_i-1)}{e_i} = 0 \tag{9}$$

is Hurwitz, that is all the roots of (9) have negative real parts. Then, the controller designed in (6), (7) ensures that the filtered tracking error system (8) is stable. This then implies that the tracking error $e(t)$ remains bounded for all time.

In the implementation of the controller (6), (7) it is assumed that an estimate $\hat{f}(\cdot)$ of the function $f(\cdot)$ is available. This estimate is manufactured by an Adaptive Network that can approximate the nonlinear function to any desired degree of accuracy. However, for such a network to ensure small tracking error in closed-loop control, it is necessary to *learn the nonlinear function on-line*. The proposed control scheme (7) is shown in Figure 1. Note that the structure has a nonlinear adaptive inner loop plus a linear outer tracking loop.

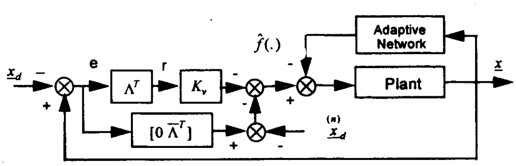

Fig. 1. Control of an unknown nonlinear system using adaptive network.

3. PASSIVITY PROPERTIES AND IMPLICATIONS FOR STABILITY

There are a number of approaches determining the stability of the adaptive control scheme (7), (8). A widely used approach would involve breaking down the system into a number of subsystems and finding a Lyapunov function for each. The gains in (7) and the adaptation scheme can then be selected to ensure stability. This approach has been utilized to control unknown systems of the form (1) and (2) using adaptive, neural, and fuzzy logic controllers. The procedure, however, is cumbersome and depends on finding a suitable Laypunov function that results in tractable adaptation laws. While these results enable exact calculation of the ultimate bounds on the states of the system, it is illuminating to study the passivity properties of the closed-loop system and its relation to the overall internal stability. Given the controller structure (7), it is natural to view the closed-loop system as an interconnection of the filtered tracking error system and the Adaptive network as shown in Figure 2.

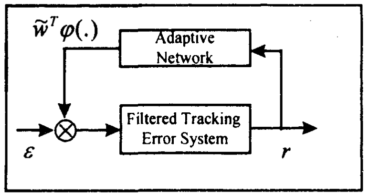

Fig. 2. Interconnected feedback structure of the adaptive controller.

In the following section, we show the relationship between the passivity properties of the interconnected system and the overall internal stability, using this to define a suitable adaptive network. To our knowledge, this is the first time that controller design has been attempted using passivity properties exclusively.

3.1. Background on passivity

The relationship between the input-output properties of a system and its stability has been extensively studied using the theory of *dissipative systems*. Here a few results from literature [3, 4, 5, 6, 13, 14] are first presented, and the results extended to derive conditions for nonlinear systems subjected to bounded disturbances. The relevance is that the Adaptive network used for control purposes herein will be constructed to have an important dissipativity property that makes them *robust* to disturbances and unmodeled dynamics.

Assumption 1. [5, 13] Let the system in (1), (2) satisfy

(i) $f(0) = y(0) = 0$.

(ii) The system is completely reachable, that is, for a given $\underline{x}_f$ and t_f there exists a $t_0 \leq t_f$ and a locally square integrable $u(t)$ such that the state can be driven from $\underline{x}(t_0) = 0$ to $\underline{x}(t_f) = \underline{x}_f$.

(iii) $\sigma(t)$ is an energy supply rate associated with this system such that

$$\sigma(t) = y^T Qy + 2y^T Su + u^T Ru \tag{10}$$

where Q, R are constant matrices with Q and R symmetric.

Definition 2. [5, 13] The system (1), (2) with supply rate (10) is said to be *dissipative* if for all locally square integrable inputs $u(t)$ and for all $t_f > t_0$

$$\int_{t_0}^{t_f} \sigma(t)\,\mathrm{d}t \geq 0 \tag{11}$$

with $\underline{x}(t_0) = 0$ and $\sigma(t)$ evaluated along the trajectory of (1), (2).

Lemma 1. [5, 13] The system (1), (2) is dissipative with respect to the supply function (10) if and only if there exist real function $\Psi(\cdot)$, $\ell(\cdot)$ and $W(\cdot)$ satisfying $\Psi(x) > 0 \; \forall \underline{x} \neq 0$, $\Psi(0) = 0$, and

$$\dot{\Psi}(\underline{x}) = \sigma(t) - [\ell(\underline{x}) + W(\underline{x})\, u]^T \; [\ell(\underline{x}) + W(\underline{x})\, u] \tag{12}$$

along the trajectories of (1), (2).

The function $\Psi(\cdot)$ is known as the storage function for the system (1), (2).

3.2. Stability properties of interconnected systems based on passivity

While dissipativity property is a convenient tool for generating Lyapunov functions for autonomous systems, it is not possible to study the internal stability of feedback systems subject to exogenous inputs without stronger conditions on the system like complete state observability. It has been observed only recently [10] that using a stronger version of passivity namely, *state-strict passivity* can overcome this limitation. In this subsection, this novel concept is first defined and its use in analyzing internal stability of interconnected systems demonstrated.

Definition 3. The system (1), (2) is *passive* if it is dissipative with respect to the supply rate (10) with $R = 0$ and $Q = 0$. A passive system is *state strict passive* if it is dissipative with respect to the supply function

$$\sigma(t) = y^t\, u - \varepsilon \underline{x}^T\, \underline{x}, \quad \varepsilon > 0 \tag{13}$$

where $\underline{x}$ is the state of the system. From Lemma 1, it is clear that any system verifying (13) with $\sigma(t) = y'u$ is passive. Under these conditions, that is $\sigma(t) = y'u$, (13) is said to be in *power form*.

Theorem 2. Consider the system of the form shown in Figure 3. Suppose the subsystems H_1 and H_2 are *state-strict passive* with respect to the supply rates

$$\begin{aligned} \sigma_1(t) &= y_1^T\, u_1 - \varepsilon_1 \|\underline{x}_1\|^2 \\ \sigma_2(t) &= y_2^T\, u_2 - \varepsilon_2 \|\underline{x}_2\|^2 \end{aligned}$$

and

$$\|y_1\| \leq \alpha \|\underline{x}_1\|, \quad \alpha > 0. \tag{14}$$

Then the feedback system is UUB for all bounded inputs $\xi(t)$.

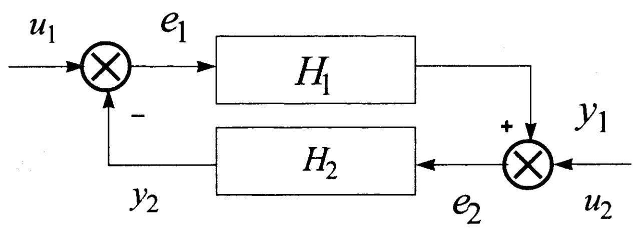

Fig. 3. Interconnection of two subsystems in feedback configuration.

Remark. No observability or persistence of excitation conditions are required on subsystem H_2.

Proof. Since H_1 and H_2 are state-strict passive, there exist storage functions $\Psi_1(x_1)$ and $\Psi_2(x_2)$ satisfying Lemma 1. Taking the Lyapunov function

$$\Psi(x_1, x_2) = \Psi_1(x_1) + \Psi_2(x_2),$$

we have

$$\dot{\Psi}(x_1, x_2) \leq \sigma_1(t) + \sigma_2(t) \leq y_1' u_1 - \varepsilon_1 \|\underline{x}_1\|^2 + y_2' u_2 - \varepsilon_2 \|\underline{x}_2\|^2.$$

Substituting (14) and using (12), (13)

$$\dot{\Psi}(x_1, x_2) \leq \alpha \|\xi\| \, \|\underline{x}_1\| - \varepsilon_1 \|\underline{x}_1\|^2 - \varepsilon_2 \|\underline{x}_2\|^2.$$

Thus, for all bounded inputs the states $\underline{x}_1$ and $\underline{x}_2$ are bounded for all time or the states of the system are UUB. □

4. DESIGN OF ADAPTIVE CONTROLLERS

In Section 2 it was shown that the proposed controller (7) could be interpreted as having two parts – a nonlinear adaptive inner loop and a linear outer tracking loop. The results of Section 3 suggest that the stability of the system (1), (2) under the controller (7) can be concluded if the system is state-strict passive and the adaptive network is designed to be state-strict passive. The design of the controller is therefore carried out in two stages. First the dynamics of the filtered error system (8) are shown to be state-strict passive and satisfy condition (14). Then, the adaptation laws for the adaptive network are chosen to make it state-strict passive. The stability of the interconnected system can then be concluded using Theorem 2.

Define $\xi(t) = \hat{f}(t) + d$. Then the filtered error system (8) can be expressed in vector notations as

$$\dot{r}(t) = -K_\nu \, r(t) + \xi(t). \tag{15}$$

Lemma 3. The dynamics (15) from $\xi(t)$ to $r(t)$ are a *state-strict passive system.*

Proof. Consider the Lyapunov function

$$V = \tfrac{1}{2} r^T r.$$

Differentiating both sides, and substituting (15) results in

$$\dot{V} = -r^T K_\nu \, r + r^T \xi. \tag{16}$$

(16) is in power-form (13) with the supply function $r^T \xi - r^T K_\nu \, r$. From Lemma 1 and Definition 3, it follows that the dynamics (15) from $\xi(t)$ to $r(t)$ are a state-strict passive system. □

In the implementation of the controller (7), the estimate of the nonlinearity $\hat{f}(\cdot)$ was assumed to be manufactured by an adaptive network to a specified degree of accuracy. Several techniques are now given for selecting the state-strict passive adaptive network.

4.1. Functional estimation using CMAC neural networks

Consider a CMAC Neural Network [2] with input $r(t)$ and output $\hat{f}(\cdot)$. The output of the neural network can be expressed as

$$\hat{f}(x) = \hat{w}^T \varphi(\underline{x}) \tag{17}$$

where $\varphi(\cdot)$ is a vector of activation functions and $\hat{w}$ is a matrix of the weights associated with each node. The activation functions $\varphi(\cdot)$ form a basis set for a class of functions. That is, there exists a set of ideal weights w such that any function belonging to this class can be expressed as $w^T \varphi(\cdot) + \varepsilon$, where ε is the function reconstruction error. The error in the function estimate is then given by

$$\tilde{f}(\underline{x}) = \tilde{w}\,\varphi(\underline{x}) + \varepsilon, \tag{18}$$

where $\tilde{w} = w - \hat{w}$ [2].

Let the weight update for the Neural Network be given by

$$\dot{\hat{w}} = -\kappa \|r\|\, F\,\hat{w} + F\,r\,\varphi^T(\underline{x}) \tag{19}$$

where F is a symmetric positive definite matrix.

Lemma 4. The weight update law (19) guarantees the neural network to be state-strict passive from input $r(t)$ to $\tilde{w}^T \varphi(\underline{x})$.

Proof. Take the nonnegative function

$$V = \mathrm{tr}(\tilde{w}^T F^{-1} \tilde{w}).$$

Differentiating both sides, and substituting (19)

$$\begin{aligned} \dot{V} &= \mathrm{tr}(\tilde{w}^T \varphi(\underline{x})\, r^T) + \mathrm{tr}(\kappa\, \tilde{w}^T \|r\|\, \hat{w}) \\ &= r^T(\tilde{w}^T \varphi(\underline{x})) + \kappa \|r\|\, \mathrm{tr}(\tilde{w}^T \hat{w}) \\ &\le r^T(\tilde{w}^T \varphi(\underline{x})) - \kappa \|r\| \left(\|\tilde{w}\|^2 - w_{\max} \|\tilde{w}\| \right), \end{aligned}$$

where $\max(w) = w_{\max}$.

Since the derivative of V is in the power form, from Lemma 1 and Definition 3 it follows that the neural network is state-strict passive from input $r(t)$ to $\tilde{w}^T \varphi(\underline{x})$. □

4.2. Functional estimation using multi–layer neural networks

The output of a multi-layer neural network can be expressed in the form

$$y_i = \sum_{j=1}^{N_2} \left[w_{ij}\, \sigma_j \left(\sum_{k=1}^{N_1} \nu_{ik}\, x_k + \theta_{\nu k} \right) + \theta_{wj} \right]; \quad i = 1, \ldots, N_3 \tag{20}$$

with $\sigma_j(\cdot)$ the activation functions, ν_{jk} the first-to-second layer interconnection weights, and w_{jk} are second-to-third layer interconnection weights. θ_i's are the threshold offsets and N_i the number of nodes in layer i [11].

Lemma 5. Let the weight tuning for the neural network in (20) be selected as

$$\begin{aligned}\dot{\hat{W}} &= F\,\hat{\sigma}\,r^T - F\,\hat{\sigma}\,\hat{V}^T\,x\,r^T - \kappa\,F\|r\|\,\hat{W},\\ \dot{\hat{V}} &= G\,x\left(\hat{\sigma}'^T\,\hat{W}\,r\right)^T - \kappa\,G\|r\|\,\hat{V},\end{aligned} \tag{21}$$

where W and V are the weight matrices augmented by the thresholds, F, G are positive-definite symmetric matrices, and κ a positive design parameter.

Then, the weight update law (21) guarantees the neural network to be state-strict passive from input $r(t)$ to $\tilde{w}^T\,\varphi(\underline{x})$.

Proof. See [11]. □

The structure of the neural network (19), (21) is obtained using a filtered error/passivity approach. This method has several advantages over conventional neural networks based on gradient laws and backpropagation algorithms. Standard backpropagation tuning can result in unbounded weights in the neural network if (a) the network cannot exactly reconstruct a certain required nonlinear function, or (b) there are bounded unknown disturbances in the system dynamics. The novel weight update laws (19) do not require a learning phase. The stability of the closed-loop system can be established without requiring strong observability conditions or persistence of excitation. The algorithm includes correction terms to the backpropagation, plus an additional robustifying signal that guarantees tracking as well as bounded weights.

4.3. Functional estimation using adaptive networks

The neural network can estimate the unknown function as the activation functions $\varphi(\cdot)$ form a basis set and it is assumed that the nonlinear function can be expressed as a weighted linear sum of the elements of the basis set to the specified degree of accuracy. For the adaptive estimation of the function however, it is required to compute a regression matrix [16]. Let the regression matrix be $W(\underline{x})$ and the parameter vector be given by Ξ. If Ξ is known then the function can be constructed exactly as

$$f(\underline{x}) = \Xi^T\,W(\underline{x}). \tag{22}$$

Since Ξ is unknown, let the estimate of Ξ be denoted by $\hat{\Xi}$. Then the functional estimate error is given by

$$\tilde{f}(\underline{x}) = \tilde{\Xi}^T\,W(\underline{x}), \tag{23}$$

where $\tilde{\Xi}$ is the error in the parameter estimates.

Lemma 6. The adaptive update law given by

$$\dot{\tilde{\Xi}} = -\dot{\hat{\Xi}} = -\Gamma\,r\,W^T(\underline{x}) + \kappa\|r\|\,\Gamma\,\hat{\Xi}, \tag{24}$$

where Γ is a positive definite matrix makes the map from r to $\tilde{\Xi}^T\,W(\underline{x})$ state-strict passive.

Proof. Take the nonnegative function

$$V = \mathrm{tr}\left(\tilde{\Xi}^T \Gamma^{-1} \tilde{\Xi}\right).$$

Differentiating both the sides, and substituting (24)

$$\begin{aligned}\dot{V} &= \mathrm{tr}\left(\tilde{\Xi}^T W(\underline{x})\, r^T\right) + \mathrm{tr}\left(\kappa\, \tilde{\Xi}^T \|r\|\, \hat{\Xi}\right) \\ &= r^T \left(\tilde{\Xi}^T W(\underline{x})\right) + \kappa \|r\|\, \mathrm{tr}\left(\tilde{\Xi}^T \hat{\Xi}\right) \\ &\le r^T \left(\tilde{\Xi}^T W(\underline{x})\right) - \kappa \|r\| \left(\|\tilde{\Xi}\|^2 - \Xi_{\max} \|\tilde{\Xi}\|\right),\end{aligned}$$

where $\max(\Xi) = \Xi_{\max}$.

From Lemma 1 and Definition 3 it follows that the adaptive network is state-strict passive from input $r(t)$ to $\tilde{\Xi}^T W(\underline{x})$. □

Lemmas 4, 5, 6 satisfy the conditions of Theorem 2, and hence the *closed-loop system is Uniformly Ultimately Bounded* when the approximating network satisfies (19), (20) or (24).

It is seen that state-strict passivity is needed to ensure boundedness of all states when the closed-loop system is subjected to bounded disturbances. The choice of the adaptation laws for the network is crucial as this guarantees boundedness of the signals without the requirement of the *persistency of excitation* (PE) [15] condition required in most adaptive control techniques.

5. SIMULATION EXAMPLE

As an example the controller proposed in Sections 3, 4 is tested on the system given by the following set of equations

$$\begin{aligned}\dot{x}_1 &= x_2 + u_1, \\ \dot{x}_2 &= -x_1 + 2e^{-(x_1^2 + x_2^2)}\, x_2 - 0.1x_2 + u_2.\end{aligned} \tag{25}$$

The system outputs are

$$\begin{aligned}y_1 &= x_1 \\ y_2 &= x_2.\end{aligned} \tag{26}$$

The control inputs u_1 and u_2 are to be selected so that y_1 tracks a square signal and y_2 tracks a sinusoidal signal of period 2 seconds.

5.1. CMAC controller

The receptive fields for the CMAC NN are selected to cover the input space $\{[-2, 2] \times [-2, 2]\}$ with knot points at intervals of 0.25 along each input dimension. The initial conditions for both states x_1 and x_2 are taken to be zero. Figures 4 and 5 show the desired and actual the MIMO system (25), (26) using the CMAC NN controller (7), (19). It is seen that although *578 weights are needed to define the output* (22), *only 8* (2×2^2) *weights are updated at any given instant.*

5.2. Multi–layer neural network controller

A three layer neural network is selected with 5 nodes in the hidden layer. A Sigmoid function is selected as activation function for each node and the weight updates are performed according to tuning law (21). Figures 6 and 7 show the desired and the actual outputs for the MIMO system given by equations (25), (26) using the neural controller (21), (22).

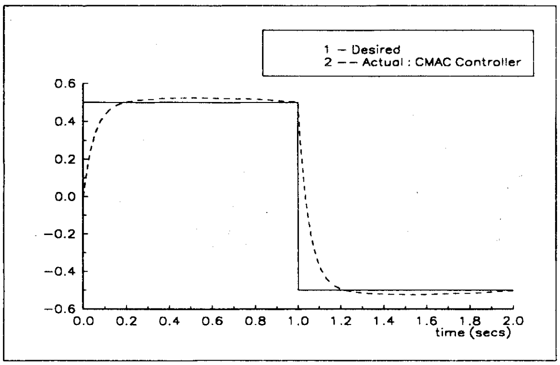

Fig. 4. Actual and desired output y_1 with CMAC controller.

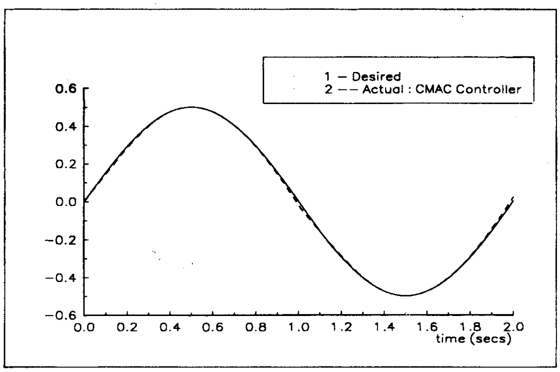

Fig. 5. Actual and desired output y_2 with CMAC controller.

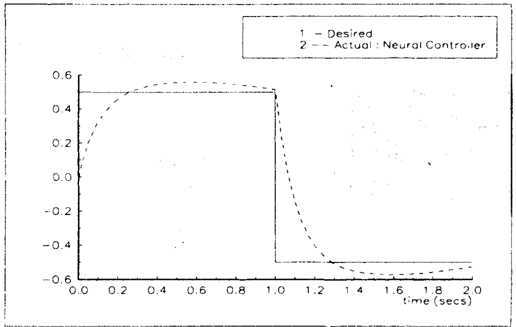

Fig. 6. Response of the system with multi-layer neural network controller – output y_1.

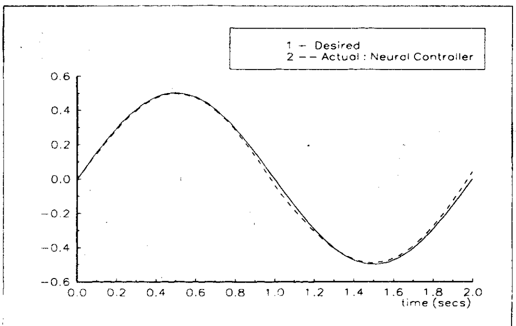

Fig. 7. Response of the system with multi-layer neural network controller – output y_2.

5.3. Adaptive controller

It is clearly seen that the performance of CMAC NN is excellent despite the fact that the dynamics are unknown. For comparison, a standard adaptive controller [16] is implemented assuming that the only unknowns are the coefficients of the terms on the right-hand-side on (25). The regression vector of the given system is $W = [\, x_1 \; x_2 \; e^{-(x_1^2 - x_2^2)} \; x_2 \,]$. The outputs for the adaptive case are shown by '–.' in Figures 8 and 9. It is to be noted that to obtain good performance, the regression vector must be exactly known. Inaccuracy in the knowledge of the unmodeled dynamics can result in inaccurate regression vector which leads to rapid degradation of the performance. To demonstrate this, the regression vector is assumed to be $W = [\, 0 \; x_2 \; e - (x_1^2 + x_2^2) \, x_2 \,]$. From the response shown in Figures 8 and 9 it is seen that this inaccurate regression vector leads to poor tracking of x_2.

5.4. Exact computed torque controller

Finally, a comparison is made to compare the performance of the CMAC controller with that of a *computed-torque* controller where all the nonlinearities are known. Figure 10 shows the output y_1 for both the cases. It can be seen that the CMAC is able to match the performance of the computed-torque controller even though the CMAC controller knows none of the dynamics *a priori*.

6. CONCLUSIONS

A new methodology for the design of stabilizing controllers for a class of unknown nonlinear systems is presented. It is shown that designing the controller in two stages significantly simplifies the overall implementation. The resultant controller has two components: an outer tracking-loop and an inner-loop consisting of an adaptive network for manufacturing the nonlinear elements in the dynamics. It is shown that choosing an adaptation law that makes the network *state-strict passive* is *sufficient* to guarantee the closed-loop stability of the overall system. The result is a "robust"

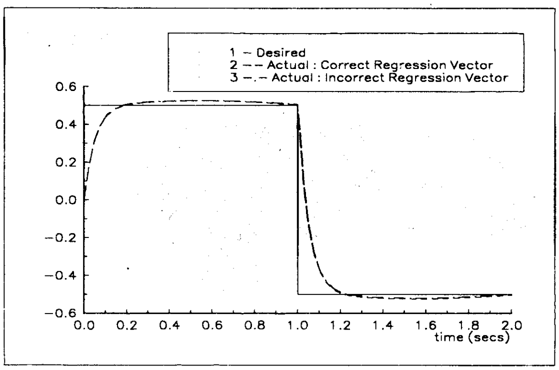

Fig. 8. Response of the system with adaptive controller – output y_1.

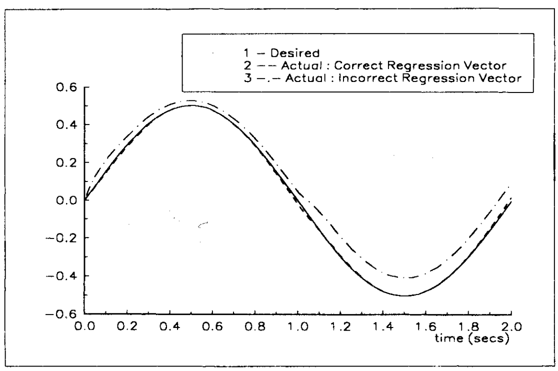

Fig. 9. Response of the system with adaptive controller – output y_2.

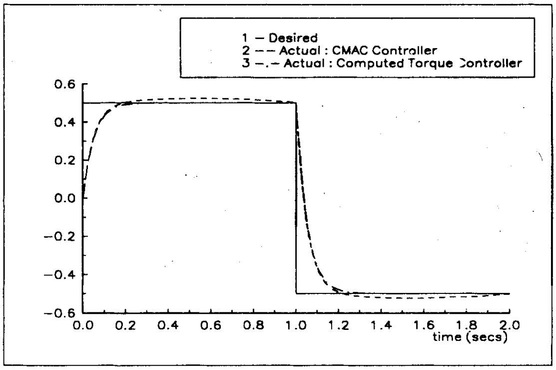

Fig. 10. Comparison of CMAC controller with a computer-torque controller.

neural network controller that does not require persistence of excitation and learns the nonlinear function on-line.

(Received February 16, 1996.)

REFERENCES

[1] C.I. Byrnes, A. Isidori and J.C. Willems: Passivity, feedback equivalence, and the global stabilization of minimum phase nonlinear systems. IEEE Trans. Automat. Control *36* (1991), 11, 1228–1240.

[2] S. Commuri and F. L. Lewis: CMAC neural networks for control of nonlinear dynamical systems: Structure, stability and passivity. Automatica, to appear.

[3] G.C. Goodwin and K.S. Sin: Adaptive Filtering, Prediction, and Control. Prentice Hall, New Jersey 1984.

[4] D. J. Hill and P. J. Moylan: The stability of nonlinear dissipative systems. IEEE Trans. Automat. Control *AC–21* (1976), 5, 708–711.

[5] D. J. Hill and P. J. Moylan: Stability results for nonlinear feedback systems. Automatica *14* (1977), 377–382.

[6] D. J. Hill and P. J. Moylan: Dissipative dynamical systems: Basic input–output and state properties. J. Franklin Inst. *309* (1990), 5, 328–357.

[7] S. Jagannathan and F. L. Lewis: Multilayer discrete–time neural net controller with guaranteed performance. IEEE Trans. Neural Networks, to appear.

[8] I. Kannelakopoulos, P.V. Kotovic and A.S. Morse: Systematic design of adaptive controllers for feedback linearizable systems. IEEE Trans. Automat. Control *36* (1991), 36, 1241–1253.

[9] P. V. Kokotovic, M. Krstic and I. Kannelakopoulos: Backstepping to passivity: Recursive design of adaptive systems. In: Proc. IEEE Conf. Decision and Control, Tuscon 1992, pp. 3276–3280.

[10] F.L. Lewis, C.T. Abdallah and D.M. Dawson: Control of Robot Manipulators. MacMillan Publishing Company, New York 1993.

[11] F.L. Lewis, K. Liu and A. Yesildirek: Neural net robot controller with guaranteed tracking performance. IEEE Trans. Neural Networks, to appear.

[12] F.L. Lewis, A. Yesildirek and K. Liu: Neural net robot controller: Structure and stability proofs. J. Intelligent and Robotic Systems, to appear.

[13] P. J. Moylan: Implications of passivity in a class of nonlinear systems. IEEE Trans. Automat. Control *AC-19* (1974), 4, 373–381.

[14] P. J. Moylan and D. J. Hill: Stability criteria for large-scale systems. IEEE Trans. Automat. Control *AC-23* (1978), 2, 143–149.

[15] K. S. Narendra and A. M. Annaswamy: A new adaptive law for robust adaptation without persistent adaptation. IEEE Trans. Automat. Control *AC-32* (1987), 2, 134–145.

[16] K. S. Narendra and A. M. Annaswamy: Stable Adaptive Systems. Prentice Hall, New Jersey 1989.

[17] D. G. Taylor, P. V. Kokotovic, R. Marino and I. Kannelakopoulos: Adaptive regulation of nonlinear systems with unmodeled dynamics. IEEE Trans. Automat. Control *34* (1989), 4, 405–412.

Sesh Comuri, CGN & Associates, Inc., 415 S. W. Washington, Peoria, IL 61614. U. S. A.

Frank L. Lewis, Automation and Robotics Research Institute, The University of Texas and Arlington, 7300, Jack Newell Blvd. S, Fort Worth, Texas 76118–7115. U. S. A.