NEURAL CONTROL SYSTEMS

Campos J.
Montavista Software Inc., USA

Lewis F.L.
University of Texas at Arlington, USA

Keywords: neural networks, backstepping neural network, dynamic neural network, output feedback control, functional link neural network, neuron, hidden layer, robot control

Contents

Mathematical Representation

Thorough this article, the following notation has been adopted to differentiate among scalars, vectors, and matrices:

x is a scalar

$\bar{x}$ is a vector

$\underline{x}$ is a matrix

Summary

A framework for intelligent control using neural networks is presented. Neural networks make use of a universal approximation property that allows them to be used in feedback control of unknown systems without a requirement for linearity in the system parameters.

This article shows that neural networks provide model-free learning controllers for a class of nonlinear systems, in the sense that not even a structural or parametrized model of the system dynamics is needed. Another remarkable issue in the use of neural networks is the reusability of the low-level controller, since the system dynamics are completely unknown to the controller. The same controller works even if the behavior or structure of the system changes. Several feedback control topologies and weight tuning that guarantee closed-loop stability and bounded weights are given. Applications to robot manipulator control, active vehicle suspension, and industrial system control are set out.

1. Introduction

Intelligent control has received a great deal of interest in the past years due to the desire to emulate the functioning of the human body, particularly the brain's learning process and the execution of bodily actions. The closer that any designed system's behavior conforms to human learning/adaptation capabilities the more efficient that system will become, and the less human operator intervention will be needed. Neural networks are among the structures used in this approach to achieving "intelligence" in systems.

Neural networks (NN) have achieved great success in classification, pattern recognition, and other open-loop applications in digital signal processing and elsewhere. There has been plenty of research into the use of neural networks for control applications, and they are considered 'universal model-free controllers' in the sense that a mathematical model of the controlled plant is not required. Neural networks try to mimic the functions of biological processes, in order to learn about their environment and account for it to improve overall performance.

We must distinguish between two main classes of neural network for control applications: *open-loop identification* and *closed-loop feedback control*. The former class resembles signal processing and classification, so most of the techniques and algorithms appropriate to these fields will still apply. On the contrary, in the feedback control applications, the NN reside inside the control loop, which requires special care to ensure the tracking error and NN weights remain bounded in the closed-loop system. This article concentrates on the latter class.

There is a considerable literature on NN for feedback control of unknown systems. However, it was not until the 1990s that repeatable design algorithms and stability proofs became available, thus guaranteeing performance. Most of the initial approaches required some off-learning phase (that is, training off-line) to tune the NN weights, using measurements of the system inputs and outputs, in order to guaranteed stability. This represented a problem for industrial and mechanical systems that usually required immediate control, besides the fact that "untrained" perturbation in the system was likely to affect handling performance and capability. Furthermore, in the early application stages of direct control back-propagation weight tuning was completely dependent on the unknown system, and/or satisfied its own differential equations, making them very difficult to compute.

In this article we present a comprehensive approach to the design and analysis of neural network controllers. The control structures discussed are multiloop controllers with outer tracking proportional-derivative (PD) loops, containing NN in some of the loops. The algorithms presented are of repeatable design, and guarantee the system performance by including small tracking errors and bounded NN weights. It is shown that the NN controllers require additional structures as uncertainty about the controlled system itself increases.

The NN controllers are adaptive learning systems but do not rely on the usual assumptions made in adaptive control theory, such as linearity in their parameters and availability of a known regression matrix. This is primarily due to the NN universal function approximation property. NN controllers may be called 'nonparametric controllers' in that they are not parametrized in terms of system parameters. When designed correctly, the NN controller does not call for assumptions about persistence of excitation or certainty equivalence.

The article begins by discussing multilayer nonlinear and linear in networks of parameters. Tracking controllers for robot manipulators are presented for networks with tuning algorithms that guarantee closed-loop stability and for those with bounded weights. Some extensions are also discussed, including NN force control, actuator dynamics nonlinearity compensation, and output-feedback control.

2. Neural Network Structures and Properties

Neural networks (NN) can be used in two classes of applications in system theory: signal processing/classification and control. There are two classes of control applications: open-loop identification and closed-loop feedback control. Identification applications resemble signal processing/classification, so the same open-loop algorithms may often be used. On the other hand, in closed-loop feedback applications the NN is inside the control loop, so special steps must be taken to ensure that the NN weights remain bounded during the control run.

2.1. Static Feedforward Neural Networks

A feedforward NN is shown in Figure 1.

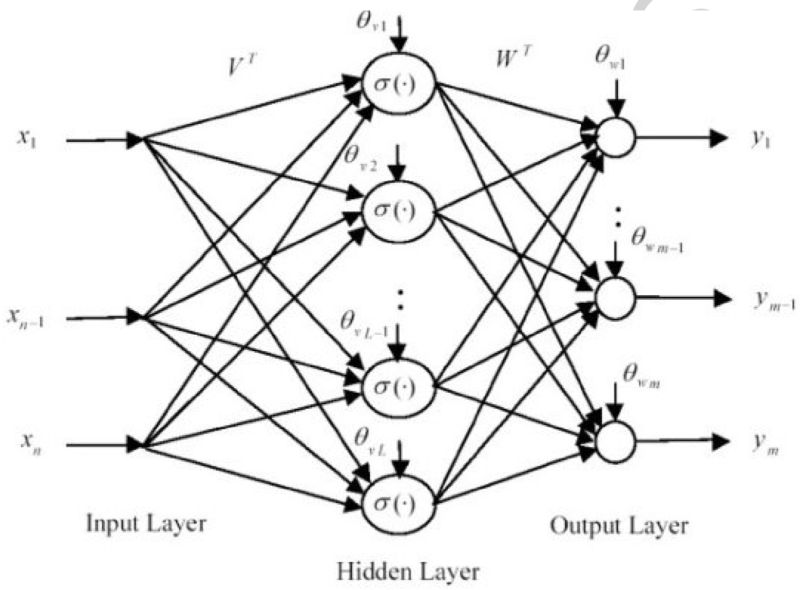


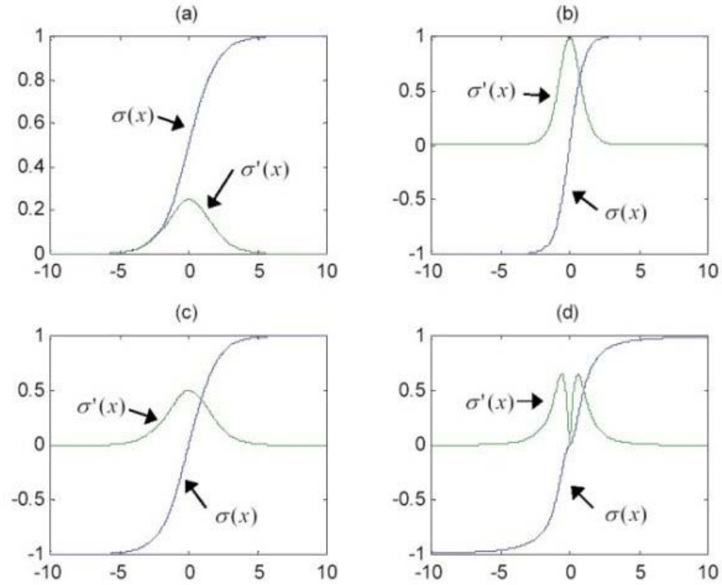
Figure 1. Two-layer feedforward neural network

This NN has two layers of adjustable weights, and is here termed a *two-layer net*. The NN output $\bar{y}$ is a vector with m components that are determined in terms of the n components of the input vector $\bar{x}$ by the formula

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

where $\sigma(\cdot)$ are the activation functions and L is the number of hidden-layer neurons. The first- to second-layer interconnection weights are denoted by v_{jk}, and the second- to third-layer interconnection weights by w_{ij}. The threshold offsets are denoted by θ_{vj}, θ_{wi}.

Many different activation functions $\sigma(\cdot)$ are in common use. In this work, it is required that $\sigma(\cdot)$ is sufficiently smooth that at least its first derivative exists: suitable choices are shown in Figure 2.



$$(a) \quad \sigma(x) = \frac{1}{1+e^{-x}} \qquad (b) \quad \sigma(x) = \tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

$$(c) \quad \sigma(x) = \frac{1-e^{-x}}{1+e^{-x}} \qquad (b) \quad \sigma(x) = \frac{x^2}{1+x^2}\mathrm{sgn}(x)$$

Figure 2. Common neural network activation functions and derivatives

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

$$\bar{y} = \underline{W}^T \bar{\sigma}(\underline{V}^T \bar{x}). \qquad (2)$$

The thresholds are included as the first columns of the weight matrices. Any tuning of $\underline{V}$ and $\underline{W}$ then includes tuning of the thresholds as well. To accomplish this, the vectors $\bar{x}$ and $\bar{\sigma}(\underline{V}^T \bar{x})$ need to be augmented by placing a '1' as their first element (e.g., $\bar{x} \equiv [1 \quad x_1 \quad x_2 \quad \cdots \quad x_n]^T$). In this equation, in order to represent Eq. (1) we have sufficient generality if $\sigma(\cdot)$ is taken as diagonal function from $\Re^L$ to $\Re^L$: that is, $\bar{\sigma}(\bar{z}) = diag\{\bar{\sigma}(z_i)\}$ for a vector $\bar{z} = [z_1 \quad z_2 \quad \cdots \quad z_L]^T \in \Re^L$.

2.2. Universal Function Approximation Property

Neural networks (NN) satisfy many important properties. A major concern for feedback control purposes is the universal function approximation property. Let $\bar{f}(\bar{x})$ be a general smooth function from $\Re^n$ to $\Re^m$. From this it can be shown—as long as $\bar{x}$ is restricted to a compact set $S \in \Re^n$—that weights and thresholds exist, such that

$$\bar{f}(\bar{x}) = \underline{W}^T \bar{\sigma}(\underline{V}^T \bar{x}) + \bar{\varepsilon} \tag{3}$$

for a number of hidden layer neurons L. This holds for a large class of activation functions. This equation indicates that a NN can approximate any smooth function on a compact set. The value $\bar{\varepsilon}$ is called the *NN functional approximation error*, and generally decreases as the net size L increases. In fact, for any choice of a positive number ε_N we can find a feedforward NN such that $\|\bar{\varepsilon}\| < \varepsilon_N$ for all $x \in S$. This means that a NN can be selected to approximate $\bar{f}(\bar{x})$ to any desired accuracy ε_N.

The ideal NN weights in matrices $\underline{V}, \underline{W}$ that are needed to best approximate a given nonlinear function $\bar{f}(\bar{x})$ are difficult to determine. In fact, they may not even be unique. However, all that must be known for control purposes is that some ideal approximate NN weights exist for a specified value for ε_N. Then, an estimate of $\bar{f}(\bar{x})$ can be given by

$$\hat{\bar{f}}(\bar{x}) = \hat{\underline{W}}^T \bar{\sigma}(\hat{\underline{V}}^T \bar{x}) \tag{4}$$

where $\hat{\underline{W}}$ and $\hat{\underline{V}}$ are estimates of the ideal NN weights that are provided by certain on-line weight-tuning algorithms, which will be detailed subsequently.

The assumption that there exist ideal weights, such that the approximation property holds, resembles various similar assumptions in adaptive control, including Erzberger's assumptions and parameter linearity (see *Adaptive Control*). The very important difference in the case of NN is that the approximation property always holds: in adaptive control such assumptions often do not hold in practice, and so they imply restrictions on the form of the systems that can be controlled.

2.3. Weight-Tuning Algorithms

In order for the NN to learn and adapt to its environment, the weights should be continuously updated on-line. Many types of NN weight-tuning algorithm are used, usually based on some sort of gradient algorithm. Tuning algorithms may be given either in continuous time or in discrete time, where the weights are updated only at discrete time points. Discrete-time tuning is useful in digital control applications of NNs. A common weight-tuning algorithm is the gradient algorithm based on the propagation error, where the NN is trained to match specified exemplar pairs $(\bar{x}_d, \bar{y}_d)$, with $\bar{x}_d$ the ideal NN input that yields the desired NN output $\bar{y}_d$. The discrete-time version of the backpropagation algorithm for the two-layer NN is given by

$$\hat{\underline{W}}_{k+1} = \hat{\underline{W}}_k + \underline{F} \cdot \bar{\sigma}\left(\hat{\underline{V}}_k^T \bar{x}_d\right) \bar{E}_k^T \tag{5}$$

$$\hat{\underline{V}}_{k+1} = \hat{\underline{V}}_k + \underline{G} \cdot \bar{x}_d \left(\hat{\underline{\sigma}}_k^{'\,T} \hat{\underline{W}}_k \bar{E}_k\right)^T$$

where k is the discrete-time index and $\underline{F}$, $\underline{G}$ are positive, definite-design parameter matrices governing the speed of convergence of the algorithm. The hidden-layer output gradient or Jacobian may be explicitly computed: for the sigmoid activation functions, for instance, it is

$$\underline{\hat{\sigma}}' = diag\left\{\underline{\hat{V}}^T \bar{x}_d\right\} \cdot \left[\underline{I} - diag\left\{\bar{\sigma}\left(\underline{\hat{V}}^T \bar{x}_d\right)\right\}\right] \quad (6)$$

where $diag\{\bar{v}\}$ denotes a diagonal matrix whose diagonal elements are the components of the vector $\bar{v}$. The error $\overline{E}_k$ that is backpropagated is selected as the desired NN output minus the actual NN output $\overline{E}_k = \bar{y}_d - \bar{y}_k$. Backpropagation tuning is accomplished off-line and requires specified training data pairs $(\bar{x}_d, \bar{y}_d)$; thus it amounts to a supervised training scheme.

The continuous-time version of the backpropagation algorithm for the two-layer NN is given by

$$\dot{\underline{\hat{W}}} = \underline{F} \cdot \bar{\sigma}\left(\underline{\hat{V}}^T \bar{x}_d\right)\overline{E}^T \quad (7)$$

$$\dot{\underline{\hat{V}}} = \underline{G}\bar{x}_d\left(\underline{\hat{\sigma}}'^T \underline{\hat{W}}\overline{E}\right)^T$$

The Hebbian algorithm is a simplified NN weight-tuning scheme, a continuous-time version of which is

$$\dot{\underline{\hat{W}}} = \underline{F} \cdot \bar{\sigma}\left(\underline{\hat{V}}^T \bar{x}\right)\overline{E}^T \quad (8)$$

$$\dot{\underline{\hat{V}}} = \underline{G} \cdot \bar{x}\left(\bar{\sigma}(\underline{\hat{V}})^T \bar{x}\right)^T$$

Thus, in Hebbian tuning, no Jacobian needs to be computed. Instead the weights in each layer are updated on the basis of the outer product of the input and output signals of that layer.

2.4. Functional-Link Basis Neural Network

The NN can be considered with the first layer of weights $\underline{V}$ and thresholds fixed, and with only the second-layer weights $\underline{W}$ tuned.

Select $\underline{V} = \underline{I}$ so that the NN output is

$$y_i = \sum_{j=1}^{L}\left[w_{ij}\sigma(\bar{x}) + \theta_{wi}\right]; i = 1, \ldots, m \quad (9)$$

or, in matrix form,

$$\bar{y}(\bar{x}) = \underline{W}^T \bar{\sigma}(\bar{x}) \tag{10}$$

Now, $\bar{\sigma}(\bar{x})$ is not diagonal but is a general function from $\Re^n$ to $\Re^L$, making it a *functional-link neural net* (FLNN). In this case, the NN approximation property does not generally hold. However, a one-layer NN can still approximate functions as long as the activation functions $\bar{\sigma}(\bar{x})$ are selected as a basis. This makes the NN linear in the parameters: this case has been treated for the radial basis functions, using a projection algorithm for weight tuning and for discrete-time systems. It is proven that linearity in the unknown parameters has the so-called 'best-approximation property': for a given function f, there always exists a parameter that approximates f better than all other possible choices.

Following this, to ensure suitable NN approximation properties some conditions must be satisfied by the activation function $\sigma(x)$.

Definition 1: Let S be a compact, simply connected set of $\Re^n$, and let $\varphi(x): S \to \Re^L$ be integrable and bounded. Then $\varphi(x)$ is said to provide a basis for $C^m(S)$ if:

- a constant function on S can be expressed as (10) for finite L, and
- the functional range of neural network (10) is dense in $C^m(S)$ for countable L.

It was shown by Barron that the neural network approximation error $\varepsilon(x)$ for one-layer NN is fundamentally bounded below by a term of the order $(1/n)^{2/d}$, where n is the number of fixed basis functions and d is the dimension of the input to the NN. This does not limit the tracking performance in our controller, because of the control system structure selected.

It is not straightforward to pick a basis $\varphi(x)$. CMAC, RBF, and other structured NN approaches allow one to choose a basis by partitioning the compact set S. This can be a tedious process, however. If one selects

$$\bar{y}(\bar{x}) = \underline{W}^T \bar{\sigma}(\underline{V}^T \bar{x}) \tag{11}$$

with $\bar{\sigma}(\bar{x}) = 1/(1+e^{\alpha \bar{x}})$, for example, as the sigmoid, then it can be shown that $\underline{\sigma}(\underline{V}^T \bar{x})$ is a basis if $\underline{V}$ is selected randomly. Once selected, $\underline{V}$ is fixed and only $\underline{W}$ is tuned. Then, the only design parameter in constructing the one-layer NN is the number of hidden layer neurons N_h. A larger N_h results in a smaller $\bar{\varepsilon}(\bar{x})$.

2.5. Gaussian or Radial Basis Function Networks

The selection of a suitable set of activation functions is considerably simplified in various kinds of structured nonlinear networks, including radial basis functions. A NN activation function is given as

$$\sigma(x) = e^{-(x-\mu)^2/2p} \qquad (12)$$

when x is a scalar with mean μ and variance p. These are called *Gaussian* or *radial basis functions* (RBF). A RBF NN can also be written as Eq. (2), but this function has an advantage over the usual sigmoid NN in that the n-dimensional Gaussian is well understood from probability theory, Kalman filtering, and elsewhere, so that n-dimensional RBFs are easy to conceptualize.

The j-th activation function can be written as

$$\sigma_j(\overline{x}) = e^{-\frac{1}{2}(\overline{x}-\overline{\mu}_j)^T \underline{P}_j^{-1}(\overline{x}-\overline{\mu}_j)} \qquad (13)$$

with $\overline{x}$, $\overline{\mu}_j \in \Re^n$. Let the vector of activation functions be defined as $\overline{\sigma}(\overline{x}) \equiv [\sigma_1(\overline{x}) \quad \sigma_2(\overline{x}) \quad \cdots \quad \sigma_L(\overline{x})]^T$. If the covariance matrix is diagonal so that $\underline{P}_j = diag\{p_{jk}\}$, then Eq. (13) becomes separable, and may be decomposed into its component parts as

$$\sigma_j(\overline{x}) = e^{-\frac{1}{2}\sum_{k=1}^{n} -(x_k-\mu_{jk})^2/p_{jk}} = \prod_{k=1}^{n} e^{-\frac{1}{2}(x_k-\mu_{jk})^2/p_{jk}} \qquad (14)$$

where x_k , μ_{jk} are the k-th components of $\overline{x}$, $\overline{\mu}_j$. Thus, the n-dimensional activation functions are the product of n scalar functions. This allows us to visualize the hidden-layer neurons as having n-dimensional activation functions, as in Figure 1.

It can be seen that Eq. (14) is of the form taken by the activation functions in Eq. (1), but with thresholds that are more general. The first-layer thresholds of the RBF NN are n-dimensional vectors corresponding to the mean values of the Gaussian functions, which serve to shift the functions in the $\Re^n$ plane. The first-layer weights in $\underline{V}^T$ are scaling factors that served to scale the width of variance of the Gaussians. These are both usually selected in designing the RBF NN and left fixed; only the output-layer weights $\underline{W}^T$ are generally tuned. Therefore, the RBF NN is a special sort of FLNN shown in Eq. (10).

Figure 3 shows separable Gaussians for the case $\overline{x} \in \Re^2$. In this figure, all the variances p_{jk} are identical, and the mean values μ_{jk} are chosen in a special way that spaces the activation functions at the node points of a 2D grid. To form an RBF NN that approximates functions over the region $\{-1 < x_1 \le 1, -1 < x_2 \le 1\}$, here we have selected $L = 5x5 = 25$ hidden-layer neurons, corresponding to five cells along x_1 and five along x_2. Nine of these neurons have 2D Gaussian activation functions, while those along the boundary require the illustrated ‘one-sided’ activation functions.

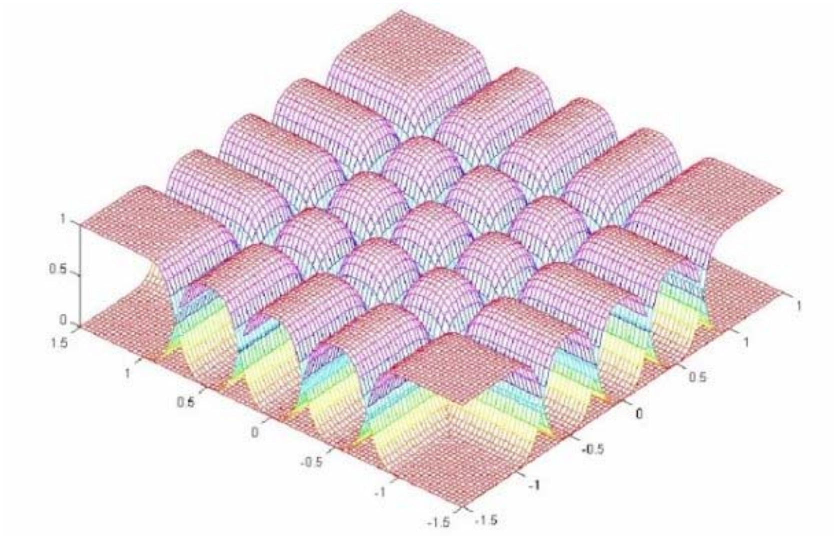


Figure 3. Two-dimensional separable Gaussian functions for an RBF NN

The importance of RBF NNs is that they show how to select the activation functions and the number of hidden-layer neurons for specific NN applications, including function approximation, while also providing insights on the information stored in the NN.

\-
\-
\-

TO ACCESS ALL THE **65 PAGES** OF THIS CHAPTER,
Click here

Bibliography

Alleyne A. and Hedrick J.K. (1992). Nonlinear control of a quarter car active suspension. *Proceedings of the American Control Conference, 1992, Chicago, IL.* pp. 21–25. Madison, Wis.: Omnipress. [This paper examines a quarter-car suspension system with a hydraulic actuator, acting under the effects of coulomb friction. Nonlinear control laws are developed and compared with both a passive system and a system acting under linear control. An observer-based nonlinear controller and an adaptive nonlinear controller are also presented.]

Alleyne A. and Hedrick J.K. (1995). Nonlinear adaptive control of active suspensions. *IEEE Transactions on Control Systems Technology* **3**(1), 94–101. [Presents a nonlinear "sliding" control law applied to an electro-hydraulic suspension system. A modified adaptation scheme that enables the identification of parameters whose values change with regions of the state space is presented.]

Astrom K.J. and Wittenmark B. (1995). *Adaptive Control*, 590 pp. Addison Wesley. [An introduction to and overview of the theoretical and practical aspects of adaptive control.]

Barron A.R. (1993). Universal approximation bounds for superposition of a sigmoidal function. *IEEE Transactions on Information Theory* **39**(3), 930–945. [Presents the approximation properties of a class of artificial neural networks. It is shown that feedforward networks with one layer of sigmoidal nonlinearities achieve integrated squared errors in the order O(1/n), where n is the number of nodes.]

Campos J., Lewis F.L., Davis L., and Ikenaga S. (2000). Backstepping-based fuzzy logic control of active vehicle suspensions systems. *Proceedings of the American Control Conference, 2000, Chicago, IL,* pp. 4030–4035. Madison, Wis.: Omnipress. [Presents a scheme for the active control of vehicle suspension systems using the two-degrees-of-freedom or quarter-car model.]

Chen F.C. and Khalil H.K. (1995). Adaptive control of nonlinear discrete-time systems using neural networks. *IEEE Transactions on Automatic Control* **450**(5), 791–801. [Presents a self-tuning adaptive control approach for an unknown feedback-linearizable discrete-time system represented by a input–output model. A layered neural network is used to model the unknown nonlinear functions.]

Craig J.J. (1993). *Adaptive Control of Robot Manipulators*. 450 pp. Reading, Mass.: Addison-Wesley. [Introduces the science and engineering of mechanical manipulation, and provides a complete overview of the fundamental skills required by the mechanics and control of manipulators.]

Ge S.S., Lee T.H., and Harris C.J. (1999). *Adaptive Neural Network Control of Robotic Manipulators*, 396 pp. World Scientific Press. [Dedicated to issues concerning adaptive control of robots based on neural networks. The text has been carefully tailored to (i) give a comprehensive study of robot dynamics, (ii) present structured network models for robots, and (iii) provide systematic approaches for neural-network-based adaptive controller design for rigid robots, flexible joint robots, and robots in constraint motion.]

Girosi F. and Poggio T. (1990). Networks and the best-approximation property. *Biological Cybernetics* **63**, 169–176. [Proves that neural networks derived from regularization theory and including radial basis functions can approximate continuous functions arbitrarily well.]

Grundelius M. and Angelli D. (1996). Adaptive control of systems with backlash acting on the input. *Proceedings of the 35th Conference on Decision and Control, 1996, Kobe, Japan.* pp. 4689–4694. Madison, Wis.: Omnipress. [Two different adaptive controllers for systems with backlash acting on the input are discussed, one in continuous time and the other in discrete time. Both are based on a backlash inverse and a linear controller. The unknown plant parameters and the backlash width are estimated.]

Han C.W. and Zhong Y.X. (1997). Robust adaptive control of time-varying systems with unknown backlash nonlinearity. *Proceedings of the American Control Conference, 1997, Albuquerque, NM.* pp. 763–767. Kobe, Japan. [Presents a robust adaptive controller for time-varying systems with an unknown backlash nonlinearity at the plant input. The controller design is based on the pole assignment strategy.]

Haykin S. (1994). *Neural Networks*, 500 pp. New York: IEEE Press and Macmillan. [Offers the most comprehensive treatment available of neural networks from an engineering perspective.]

Igelnik B. and Pao Y-H. (1995). Stochastic choice of basis functions in adaptive function approximation and the functional-link net. *IEEE Transactions on Neural Networks* **6**(6), 1320–1329. [Presents a theoretical justification for the random vector version of the functional-link (RVFL) net, based on a general approach to adaptive function approximation.]

Ioannou P.A. and Datta A. (1991). Robust adaptive control: a unified approach. *Proceedings of the IEEE* **790**(12), 1736–1768. [Presents a complete tutorial on the entire field of robust adaptive control.]

Jagannathan S. and Lewis F.L. (1996). Discrete-time control of a class of nonlinear dynamical systems. *International Journal of Intelligent Control and Systems* **1**(3), 297–326. [Presents a family of two-layer discrete-time neural network controllers for the control of a class of mnth-order multi-input and multi-output (MIMO) dynamical system, so that linearity in the parameters holds. No initial learning phase is needed, so the control action is immediate.]

Kanellakopoulos I., Kokotovic P.V., and Morse A.S. (1991). Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transactions on Automatic Control* **36**, 1241–1253. [Outlines a procedure for designing adaptive regulation and tracking schemes for a class of feedback-linearizable nonlinear system.]

Kim Y.H. and Lewis F.L. (1998). *High-Level Feedback Control with Neural Networks*, 228 pp. World Scientific Press. [Bridges the gap between feedback control and artificial intelligence. Provides design techniques for "high-level" neural-network feedback-control topologies that contain servo-level feedback-control loops, as well as AI decisions and training at the higher levels. Several advanced feedback topologies containing neural networks are presented, including "dynamic output feedback," "reinforcement learning," and "optimal design", as well as a "fuzzy logic reinforcement" controller.]

Kosko B. (1992). *Neural Networks and Fuzzy Systems*, 449 pp. NJ: Prentice Hall. [Compares fuzzy and neural approaches to control problems.]

Krstic M., Kanellakopoulos I., and Kokotovic P.V. (1995). *Nonlinear and Adaptive Control Design*, 576 pp. New York: Wiley. [This innovative book breaks new ground in nonlinear and adaptive control design for systems with uncertainties. Introducing the recursive backstepping methodology, it shows—for the first time—how uncertain systems with severe nonlinearities can be controlled successfully with this powerful new design tool.]

Lewis F.L., Abdallah C.T., and Dawson D.M. (1993). *Control of Robot Manipulators*, 424 pp. New York: Macmillan. [Provides a thorough and up-to-date examination of control techniques for robot manipulators. It enables readers to develop an understanding of a wide variety of robot control algorithms, including design and computer simulation techniques, and covers computed-torque, robust control, adoptive control, force control, and advanced topics.]

Lewis F.L., Campos J., and Selmic R. (2002). *Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities*. 244 pp. Philadelphia, PA: Society for Industrial and Applied Mathematics. [This book brings neural network and fuzzy logic together with dynamical systems for application in industrial systems with actuator nonlinearities.]

Lewis F.L., Jagannathan S., and Yesildirek A. (1999). *Neural Network Control of Robot Manipulators and Nonlinear Systems*, 442 pp. Philadelphia, PA: Taylor and Francis. [A graduate text providing an account of neural network (NN) controllers for robotics and nonlinear systems, offering a general and streamlined design procedure for NN controllers. Gives design techniques and stability proofs for NN controllers for robot arms, practical robotic systems with high frequency vibratory modes, force control, and a general class of nonlinear systems.]

Lewis F.L., Yesildirek A., and Liu K. (1995). Neural net robot controller with guaranteed tracking performance. *IEEE Transactions on Neural Networks* **6**(3), 703–715. [Presents a two-layer neural net controller for a general serial-link robot arm.]

Lewis F.L., Yesildirek A., and Liu K. (1996). Multi-layer neural-net robot controller with guaranteed tracking performance. *IEEE Transactions on Neural Networks* **7(**2), 388–399. [Presents a multi-layer neural net controller for a general serial-link robot arm.]

Lin J.-S, and Kanellakopoulos I. (1997). Nonlinear design of active suspensions. *IEEE Control Systems* **17**(3), 45–59. [Develops a new nonlinear backstepping design for the control of active suspension systems that improves the inherent tradeoff between ride quality and suspension travel. It uses a nonlinear filter whose effective bandwidth depends on the magnitude of the suspension travel.]

Narendra K.S. and Annaswamy A.M. (1990). A new adaptive law for robust adaptation without persistent excitation. *IEEE Transactions on Automatic Control* **32**(2), 4–27. [Presents an adaptive law where the output error plays a dual role in the adjustment of the control parameter vector for the robust control of plants with unknown parameters.]

Narendra K.S. and Parthasarathy K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks* **1**(1), 4–27. [Demonstrates that neural networks can be used effectively in the identification and control of nonlinear dynamical systems. It uses multilayer and recurrent networks that are interconnected in novel configurations.]

Poggio T. and Girosi F. (1990). Networks for approximation and learning. *Proceedings of the IEEE* **78**(9), 1481–1497. [Addresses the problem of approximating nonlinear mapping by using a class of three-layer networks called “regularization networks.”]

Recker D.A., Kokotovic P.V., Rhode D., and Winkelman J. (1991). Adaptive nonlinear control of systems containing a deadzone. *Proceedings of IEEE Conference on Decision and Control, 1991, Brighton, UK*, pp. 2111–2115. New Jersey: IEEE Press. [This paper presents a Lyapunov-based adaptation scheme that updates the parameters of the specific region of the unknown deadzone nonlinearity in which the system is operating. The scheme uses switching logic to turn the adaptation on and off as the state of the system proceeds from one region of the deadzone to another.]

Rovithakis G.A. and Christoudoulou M.A. (1994). Adaptive control of unknown plants using dynamical neural networks. *IEEE Transactions Systems, Man and Cybernetics* **24**(3), 400–412. [Presents a strategy that uses a dynamical neural network to perform identification, and then a dynamic state feedback to control the unknown system appropriately.]

Sadegh N. (1992). A perceptron network for functional identification and control of nonlinear systems. *IEEE Transactions on Neural Networks* **4**, 1823–1836. [Develops a novel discrete-time control strategy that employs a perceptron neural network (PNN) for direct on-line estimation of the required feedforward control input. The developed controller can be applied to both continuous-time and discrete-time plants.]

Sanner R.M. and Slotine J.-J.E. (1991). Stable adaptive control and recursive identification using radial Gaussian networks. *Proceedings of the IEEE Conference on Decision and Control, 1991, San Antonio, TX*, pp. 2785–2791. New Jersey: IEEE Press. [Uses a modified adaptive control algorithm to permit incorporation of additional prior knowledge of the system dynamics, allowing the neurocontroller to operate in parallel with conventional fixed or adaptive controllers. A Gaussian network construction that can be used in recursive identification algorithms with similar guarantees of stability and convergence is also shown.]

Sanner R.M. and Slotine J-J.E. (1992). Gaussian networks for direct adaptive control. *IEEE Transactions on Neural Networks* **3**, 837–863. [Proposes and evaluates a direct adaptive control architecture for a class of continuous-time nonlinear dynamic system for which an explicit linear parameterization of the uncertainty in the dynamics is either unknown or impossible. The architecture uses a network of Gaussian radial basis functions to compensate adaptively for the plant nonlinearities.]

Selmic R. and Lewis F.L. (1999). Backlash compensation in nonlinear systems using dynamic inversion by neural networks. *Proceedings of the Conference in Control and Automation, 1999. Kona, Hawaii*, pp. 1163–1168. Madison, Wis.: Omnipress. [Presents a dynamic inversion compensation scheme for backlash. The compensator uses the backstepping technique with neural networks for inverting the backlash nonlinearity in the feedforward path.]

Selmic R. and Lewis F.L. (2000). Deadzone compensation in motion control systems using neural networks. *IEEE Transactions on Automatic Control* **45**(4), 602–613. [Presents a compensation scheme for general nonlinear actuator deadzones with unknown width. The compensation uses two neural networks, one to estimate the unknown deadzone and another to provide adaptive compensation in the feedforward path.]

Slotine J.J.E. and Li W. (1988). Adaptive manipulator control: a case study. *IEEE Transactions on Automatic Control* **33**(11), 995–1003. [Presents experimental results demonstrating that the proposed adaptive controller enjoys essentially the robustness of unmodeled dynamics as a PD controller, yet achieves much better accuracy than either PD or computer-torque control schemes.]

Slotine J.J.E. and Li W. (1991). *Applied Nonlinear Control*, 461 pp. NJ: Prentice-Hall. [An introduction to nonlinear control theory.]

Song Y.D., Mitchell T.L., and Lai H.Y. (1994). Control of a class of nonlinear uncertain systems via compensated inverse dynamics approach. *IEEE Transactions on Automatic Control* **39**(9), 1866–1871. [Addresses the tracking control problem of a class of nonlinear systems via inverse dynamic compensation.]

Tao G. (1996). Adaptive backlash compensation for robot control. *Proceedings of the International Federation of Automatic Control (IFAC) World Congress, 1996*, pp. 307–312. San Francisco, Calif.: Elsevier. [Presents an adaptive backlash inverse control approach for robot manipulators with unknown actuator backlash.]

Tao G. and Kokotovic P.V. (1995). Continuous-time adaptive control of systems with unknown backlash. *IEEE Transactions on Automatic Control* **40**(6), 1083–1087. [Addresses backlash compensation by developing an adaptive backlash inverse scheme and applying it to feedback control of a known linear plant with unknown backlash as its input.]

Tao G. and Kokotovic P.V. (1996). *Adaptive Control of Systems with Actuator and Sensor Nonlinearities*, 294 pp. New York: Wiley. [Presents adaptive control schemes for systems with actuator and sensor nonlinearities. The nonlinearities are approximated by piecewise linear characteristics.]

Biographical Sketches

Dr. Javier Campos was born in Maracaibo, Venezuela in 1967. He received his bachelor's degree in electrical engineering from La Universidad del Zulia, Maracaibo, Venezuela in 1991. He received a M.S. degree from The University of Texas at Arlington (UTA), USA, in 1996. He received his Ph.D. degree from The University of Texas at Arlington in 2000. From 1991 to 1994 he worked for Maraven Sociedad Anonima (S.A.), an oil company subsidiary of Petroleos de Venezuela Sociedad Anonima (PDVSA), where he worked mainly as a software engineer in charge of the maintenance and improvement of the Supervisory Control and Data Acquisition System (SCADA) system for electrical power plants and substations. In 1992 he received the Antonio Jose de Sucre Award, a Venezuelan Government scholarship awarded to selected individuals throughout the country, to pursue graduate studies in the USA.

Javier Campos currently works for Montavista Software, Inc. as a Linux consultant engineer, developing and supporting the Linux operating system for embedded systems with emphasis on the x86, powerpc, arm and mips processor architectures. During 1996–1999, he worked as a consultant for small companies, mainly under Army Small Business Innovative Research (SBIR) projects for development and implementation of intelligent control algorithms for flexible mode damping, inertia stabilization, active suspension control, and satellite tracking. His current research interests are in nonlinear and adaptive control, digital control, neural networks, fuzzy logic, and active suspension systems. He is reviewer for several Institute of Electrical and Electronics Engineers (IEEE) Conferences and Journals, including *Conference on Decision and Control* (CDC), *Transactions on Automatic Control*, *Transactions on Systems, Man, and Cybernetics*, and *Automatica*.

He received the second prize at the IEEE Fort Worth Section Graduate Paper Contest in 1999. He also received the Automation and Robotics Research Institute (ARRI) Best Student Paper Award in 1998, and the ARRI Invention Award in 2000. He is currently a member of the IEEE and an associate member of the Sigma Xi research society. He has a patent application pending for the project entitled "Backlash Compensation Using Neural Networks."

Dr. F.L. Lewis was born in Würzburg, Germany, subsequently studying in Chile and at Gordonstoun School in Scotland. He obtained a Bachelor's Degree in Physics/Electrical Engineering and a Master's in Electrical Engineering Degree at Rice University in 1971. He spent six years in the US Navy, serving as Navigator aboard the frigate USS Trippe (FF-1075), and Executive Officer and Acting Commanding Officer aboard USS Salinan (ATF-161). In 1977 he received the degree of Master of Science in Aeronautical Engineering from the University of West Florida. In 1981 he obtained his Ph.D. at the Georgia Institute of Technology in Atlanta, where he was employed as a professor from 1981 to 1990 and is currently an Adjunct Professor. He is a Professor of Electrical Engineering at The University of Texas at Arlington, where he was awarded the Moncrief-O'Donnell Endowed Chair in 1990 at the Automation and Robotics Research Institute.

Dr. Lewis has studied the geometric, analytic, and structural properties of dynamical systems and feedback control automation. His current interests include robotics, intelligent control, neural and fuzzy systems, nonlinear systems, and manufacturing process control. He is the author/co-author of two US patents, 124 journal papers, twenty book chapters and encyclopedia articles, 210 refereed conference papers, and eight books: *Optimal Control*, *Optimal Estimation*, *Applied Optimal Control and Estimation*, *Aircraft Control and Simulation*, *Control of Robot Manipulators*, *Neural Network Control*, *High-Level Feedback Control with Neural Networks* and the IEEE reprint volume *Robot Control*. Dr. Lewis is a registered Professional Engineer in the State of Texas and has been selected to the Editorial Boards of the *International Journal of Control*, *Neural Computing and Applications*, and *International Journal on Intelligent Control Systems*. He is the recipient of a National Science Foundation (NSF) Research Initiation Grant and has been continuously funded by NSF since 1982. Since 1991 he has received $1.8 million in funding from NSF and upwards of $1 million in SBIR/industry/state funding. He has received a Fulbright Research Award, the American Society of Engineering Education F.E. Terman Award, three Sigma Xi Research Awards, the UTA Halliburton Engineering Research Award, the UTA University-Wide Distinguished Research Award, the ARRI Patent Award, various Best Paper awards, the IEEE Control Systems Society Best Chapter Award (as Founding Chairman), and the National Sigma Xi Award for Outstanding Chapter (as President). He was selected as Engineer of the Year in 1994 by the Fort Worth IEEE Section, and is a Fellow of the IEEE. He was appointed to the National Academy of Engineering (NAE) Committee on Space Station in 1995 and to the IEEE Control Systems Society Board of Governors in 1996. In 1998 he was selected as an IEEE Control Systems Society Distinguished Lecturer. He is a Founding Member of the Board of Governors of the Mediterranean Control Association.