Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

A Wavelet-Based Recurrent Fuzzy Neural Network Trained With Stochastic Optimization Algorithm

Ahmad T.AbdulSadda, PhD. Student,
Department of Applied Science,Systems Engineering,
College of Engineering and Information Technology
(EIT), University of Arkansas at Little Rock(UALR)
email:atabdulsadda@ualr.edu

Kameran Iqbal, Associate Professor,
Department of Systems Engineering, College of
Engineering and Information Technology (EIT),
University of Arkansas at Little Rock (UALR)
email:kxiqbal@ualr.edu

***Abstract*— this paper presents a Wavelet-based Recurrent Fuzzy Neural Networks (WRFNN) trained with a stochastic search-based adaptation algorithm. A WRFNN represents a recurrent network of neurons employing wavelet functions whose outputs are combined using fuzzy rules. In this paper an earlier WRFNN model proposed by Lin, and Chin, [1], is modified by application of Simultaneously Perturbed Stochastic Approximation (SPSA) method for training the network. The model includes TSK-type fuzzy implication to compute output of each layer. The SPSA algorithm was shown to be a stable global optimization technique that is applicable to WRFNN models with demonstrated computational advantages over other optimization algorithms.**

***Keywords*—neural networks, fuzzy-wavelet, simultaneous perturbation algorithm.**

I. Introduction

Recently, fuzzy neural networks have demonstrated to be successful in a variety of applications [1]–[4]. Two common types of fuzzy neural works are: Mamdani-type and TSK-type fuzzy neural networks. For Mamdani-type fuzzy neural networks [3], [4], the minimum fuzzy implication is used in fuzzy reasoning. Whereas, for TSK-type fuzzy neural networks [5], the consequence of each rule represents a function input variables. The generally adopted function is a linear combination of input variables plus a constant term. Researchers [6], have shown that compared to Mamdani-type fuzzy neural networks, a TSK-type fuzzy neural network is capable of achieving superior performance in network size and learning accuracy.

A recurrent neural network, which naturally involves dynamic elements in the form of feedback connections, and may be used as internal memory, has recently attracted great interest [7]–[9]. For example, Elman, [7], networks comprise feed forward multilayer perceptron networks with an extra set of context nodes for copying the delayed states of the hidden or output nodes back to the network input. The radial basis function recurrent networks [8] were proposed to make the network output history-sensitive. Similarly, Jin *et al.* [9] studied the approximation of continuous-time dynamic systems using dynamic recurrent neural networks (DRNN).

The simultaneously perturbed stochastic approximation (SPSA) algorithm was proposed by Spall (1988, 1992), [10], which is based on a highly efficient gradient approximation techniques (requiring only two measurements of a scalar differentiable loss function). SPSA algorithm belongs to class of iterative gradient-free algorithms, [10]-[12], that have been effectively used for multivariate nonlinear optimization of complex system when an accurate system model is not available. Under reasonably general conditions, SPSA and the standard finite-difference stochastic analysis methods achieve the same level of statistical accuracy for a given number of iterations, even though SPSA uses p times fewer measurements of the objective function at each iteration (since each gradient approximation uses only 1/p the number of function measurements) [10].

This paper discusses the application of WRFNN trained by SPSA algorithm. The paper is organized as follows. In section 2 the model of wavelet neural networks will be described. In section 3 structure of the wavelet-based recurrent fuzzy neural network model will be given. In Section 4 simultaneous perturbation method will be explained. In section 5 problem formulations will be described. Section 6, and section 7consists of illustrative example and conclusion.

II. Wavelet Bases and Wavelet Neural Networks

A set of wavelet bases is a suitable tool for effectively representing nonlinearity. These orthogonal wavelets are infinite, continuous and differentiable. The support of these wavelets is $-\infty < x < \infty$. Daubechies, [13], presented wavelet bases, which are compactly supported but not infinitely supported. Rather than proposing a three-layered feed forward neural network, Daubechies proposed a simple wavelet neural network, which exhibits a much higher ability to generalize and much shorter learning time. This study adopts the non orthogonal and compactly supported functions in the finite range as wavelet bases. All the wavelet bases are allocated over the normalized range [0, 1] on the variable space.

Neural networks employing wavelet neurons are refereed to *Wavelet Neural Networks* (WNN). The WNN are

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

characterized by weights and wavelet bases. Each linear synaptic weight of wavelet basis is adjustable by learning. Notably, the ordinary wavelet neural network model applications are often useful for normalizing the input vectors into the interval [0, 1]. For the WNN with fixed wavelets, the main problem is the selection of wavelet bases/frames. The wavelet bases have to be selected appropriately since the choice of the wavelet basis can be critical to approximation performance.

Yamakawa *et al.* [14] has proposed a novel WNN structure that includes an n input vector $\{x_1, \ldots, x_i, \ldots, x_n\}$, and p output vector $\{y_1, \ldots, y_s, \ldots, y_p\}$ $y \in R^p$ (Fig. 1). This model is obtained by replacing a sigmoid activation function with single-scaling wavelets.

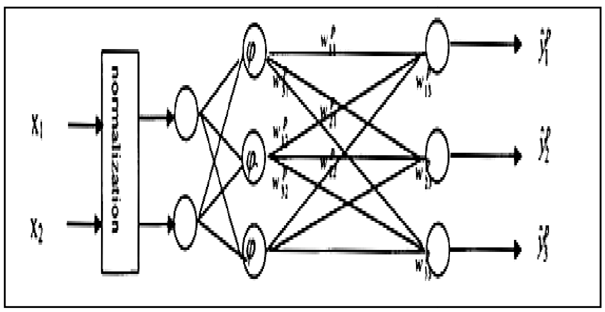

Figure 1. Schematic diagram of the WNN

III. Structure of the Wavelet-Based Recurrent Fuzzy Neural network Model

This subsection introduces the structure of the WRFNN illustrated in Fig. 2. For the TSK-type fuzzy networks [5]–[7], the consequence of each rule is a function input linguistic variable. The widely adopted function is a linear combination of input variables plus a constant term. This study adopts a nonlinear combination of input variables (i.e., WNN). Each fuzzy rule corresponds to a sub-WNN consisting of single-scaling wavelets, [1], [15].

The WRFNN model is composed of fuzzy rules that can be presented in the following general form, [1]:

R^j : If I_{1j} is A_{1j} and . . . I_{ij} is A_{ij} and . . . and I_{nj} is A_{nj}.

$$\hat{y}_j^1 = \sum_{k=1}^{M} \mathrm{w}_{\mathrm{jk}}^1 \, \emptyset_{\mathrm{a,b}} \qquad (1)$$

Then,

$$\hat{y}_j^2 = \sum_{k=1}^{M} \mathrm{w}_{\mathrm{jk}}^2 \, \emptyset_{\mathrm{a,b}} \qquad (2)$$

and,

$$Y_s = O_s^{(5)} = \frac{\sum_{j=1}^{M} I_{sj}^{(5)} I_j^{(5)}}{\sum_{j=1}^{M} I_j^{(5)}} \qquad (3)$$

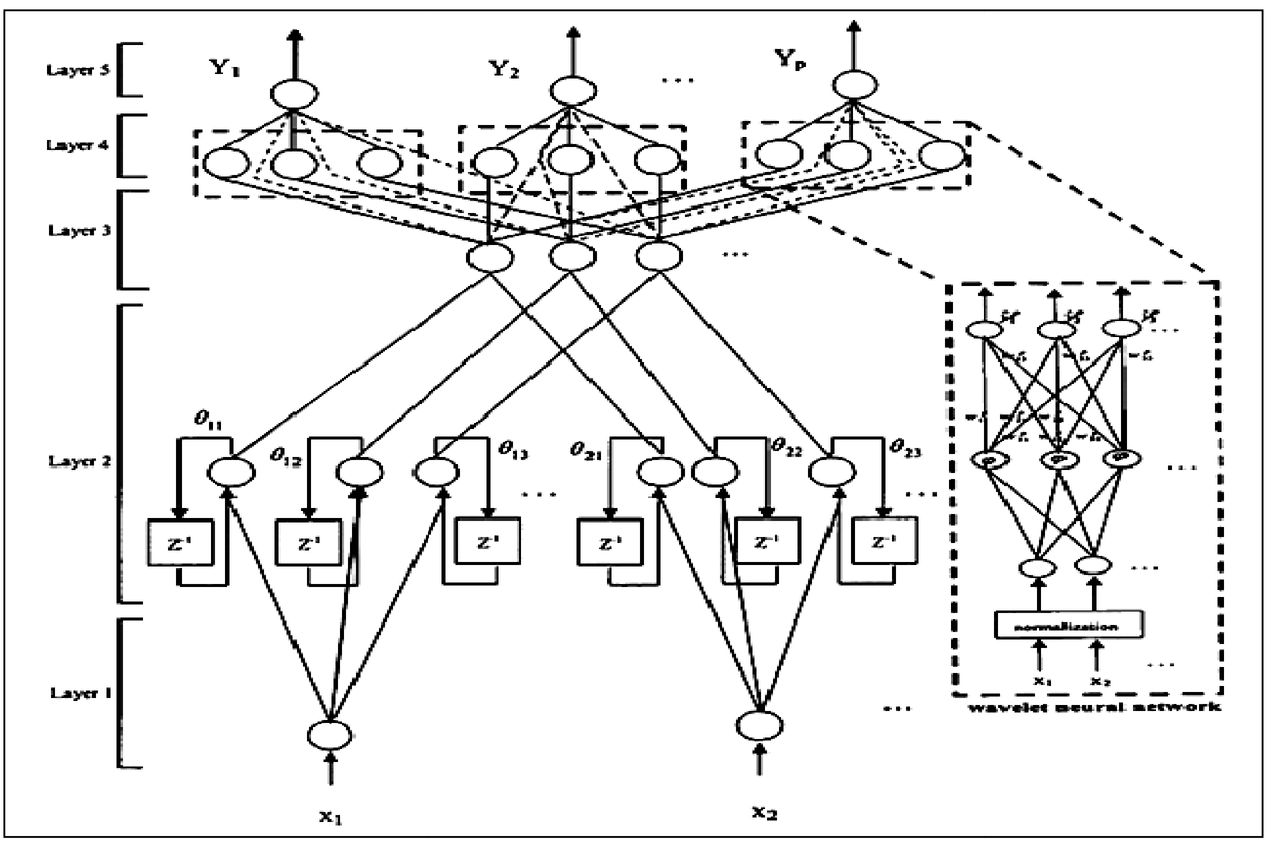

Figure 2. Schematic diagram of the WRFNN model, [1]

where R^j denotes the *jth* rule, $I_{ij} = \{I_{1j}, \ldots, I_{ij}; \ldots; I_{nj}\}$ is the network input pattern, i.e., $\{x_1, \ldots, x_i, \ldots; x_n\}$ plus the temporal term for the linguistic term of the precondition part $A_j = \{A_{1j}, \ldots, A_{ij}, \ldots, A_{nj}\}$; the local WNN model outputs $\hat{y}_j^1$ and $\hat{y}_j^2$ are calculated for outputs Y_1 and Y_2, and rule R_j, as derived in [1].

IV. Simultaneous Perturbation Method

Stochastic approximation (SA) represents an important class of stochastic search algorithms. Many well-known techniques are special cases of SA, including general-network back propagation, perturbation analysis for discrete-event systems, recursive least squares and least mean squares, and some forms of simulated annealing. Therefore, progress in general SA methodology can have a potential bearing on a wide range of practical implementations [12].

Stochastic approximation (SA) has long been applied for problems of minimizing loss functions or root finding with noisy input information. As with all stochastic search algorithms, there are adjustable algorithm coefficients that must be specified, and that can have a profound effect on algorithm performance. It is known that choosing these coefficients according to an SA analog of the deterministic Newton–Raphson algorithm provides an optimal or near-optimal form of the algorithm. However, directly determining the required Hessian matrix (or the Jacobean matrix for root finding) to achieve this algorithm form has often been difficult or impossible in practice.

The problem of minimizing a (scalar) differentiable loss function $L(\theta)$ where $\theta \in R^p$, $p \geq 1$ is considered. A typical example of would be some measure of mean-square error for the output of a process as a function of some design parameters. For many cases of practical interest, this is equivalent to finding the unique minimizing such that,[12]

$$g(\theta^*) \equiv \frac{\partial L}{\partial \theta} \quad at\ \theta = \theta^* \tag{4}$$

For the gradient-free setting, it is assumed that measurements $L(\theta)$ of, say $y(\theta)$, are available at various values of θ. These measurements may or may not include random noise. No direct measurements (either with or without noise) of $g(\theta)$ are assumed available in this setting. The recursive procedure has been used to calculate θ in general SA form:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \tag{5}$$

Where present the estimate of θ at the k^{th} iteration, $a_k > 0$ represents a$\hat{\theta}_k$ scalar gain coefficient, $\hat{g}_k(\hat{\theta}_k)$ represents an approximation of $g(\hat{\theta}_k)$ Typically, the i^{th} component of $\hat{g}_k(\hat{\theta}_k)$ (i=1,2,…,p) approximation is given by:

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{y(\hat{\theta}_k + c_k\Delta_k) - y(\hat{\theta}_k - c_k\Delta_k)}{2c_k\Delta_{ki}} \tag{6}$$

where the elements of the user specified random perturbation vector $\Delta_k = (\Delta_{k1}, \ldots, \Delta_{kp})$. are assumed to be independent and symmetrically distributed around zero; and c_k is a positive scalar (decaying to 0 for formal convergence).

V. Problem Formulation

The main set important equation was derived by Lin, and Chin [1]. These equations summarize as:

$$E(t) = \frac{1}{2p}\sum_{s=1}^{p}(Y_s(t) - Y_s^{des}(t)) \tag{7}$$

where $Y_s(t)$ is the s^{th} model output, $Y_s^{des}(t)$ is the s^{th} desired output and p denotes the number of output nodes. Let us apply SPSA algorithm to find the weight update which is used to update the laws of m_{ij} (cluster mean), σ_{ij} (variance of each incoming pattern x_i rule firing strength), w_{jk}^s (the link weight of the wave let neural network), and α_{ij}(weight of feedback recurrent connection). These updates are given in equations (8-11) below [1]:

$$w_{jk}^s(t+1) = w_{jk}^s(t) - \eta_w \frac{\partial E}{\partial w_{jk}^s} \tag{8}$$

$$mij(t+1) = mij(t) - \eta m \ \frac{\partial E}{\partial mij} \tag{9}$$

$$\sigma ij\ (t+1) = \sigma ij\ (t) - \eta\sigma \ \frac{\partial E}{\partial \sigma ij} \tag{10}$$

$$\alpha ij\ (t+1) = \alpha ij\ (t) - \eta\alpha \ \frac{\partial E}{\partial \alpha ij} \tag{11}$$

where η_w, η_m, η_σ, η_α are learning rate which are having a constant value between 0 and 1. All partial derivative will be solved depending on equation 4 (using SPSA algorithm). Under reasonably general conditions, SPSA and the standard finite-difference stochastic analysis methods achieve the same level of statistical accuracy for a given number of iterations, even though SPSA uses p times fewer measurements of the objective function at each iteration (since each gradient approximation uses only 1/p the number of function measurements)[10]-[12]. This indicates that SPSA will converge to the optimal solution within a given level of accuracy with p times fewer objective function evaluations than the standard method.

VI. Illustative Example: Identification of Nonlinear Dynamics System

We consider the following dynamics plant with time delay inputs, which has been used by Kim et al., [16]:

$$y_p(t+1) = 0.72\, y_p(t) + 0.025\, y_p(t-1)\, u(t-1) + 0.01u^2(t-2) + 0.2\, u(t-1) \tag{12}$$

In Lin, and Chin model, [1], only two input values, $y_p(t)$ and $u(t)$, were fed to the WRFNN model to determine the output $y_p(t)$. The training inputs were independent and identically distributed (*i.i.d.*) uniform sequence over [-1, 1] for about half of the training time. A single sinusoid signal was given by

1.05 sin (πt/45) for the remaining training time. The following input signal *u(t)* was used to determine the identification results:

$$u(t) = \begin{cases} \sin\left(\frac{\pi t}{25}\right) & 0 < t < 250 \\ 1.0 & 250 \le t < 500 \\ -1.0 & 500 \le t < 750 \\ 0.3\sin\left(\frac{\pi t}{25}\right) + 0.1\sin\left(\frac{\pi t}{32}\right) + 0.6\sin\left(\frac{\pi t}{25}\right) & 750 \le t < 1000 \end{cases} \quad (13)$$

Figure 3 described how the input signal looks like. In another hand, figure 4 described the ideal desired output signal how should look like.

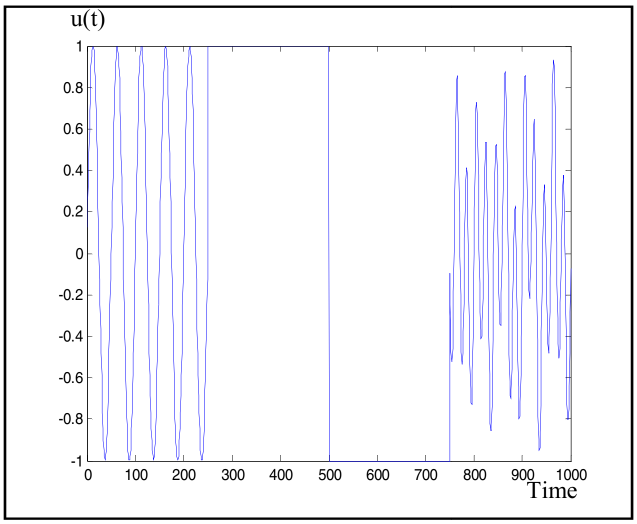

Figure 3. Input signal in equation (13)

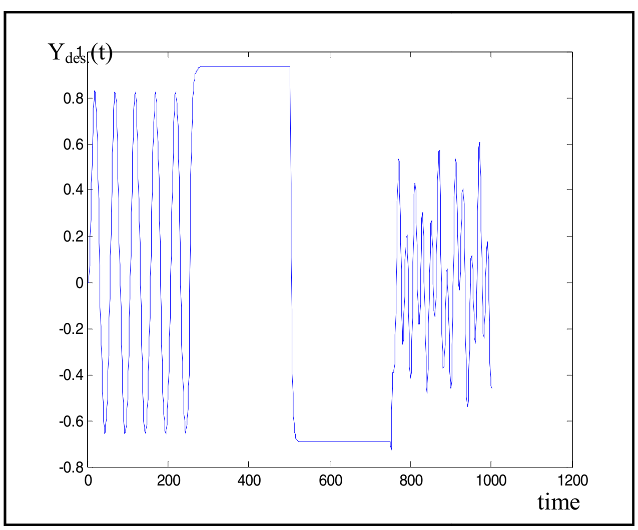

Figure 4. Desired output signal.

When applied the algorithm with five fuzzy rules, and random initial value of the forward and feedback weights with values between (-1, 1). The desired and actual output response was described in Figure 5. Figure 6 gave the relationship between the total mean square error which is represented the accuracy term parameter with the number of epochs. Finally, the cross validation verses the desired output was plotted in figure 7.

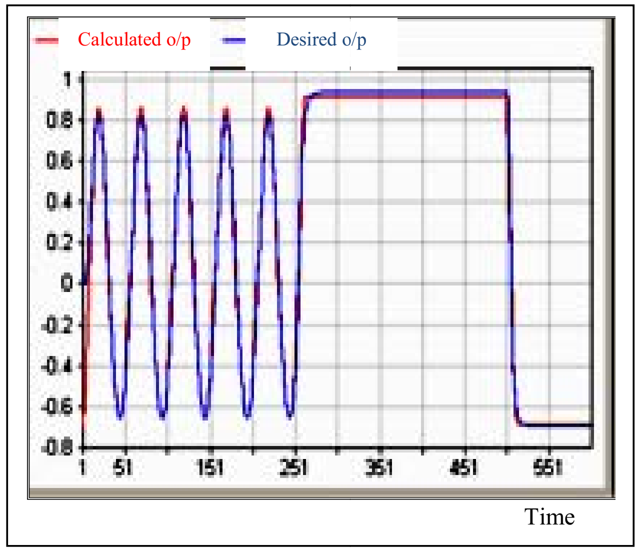

Figure 5. Desired and calculated output

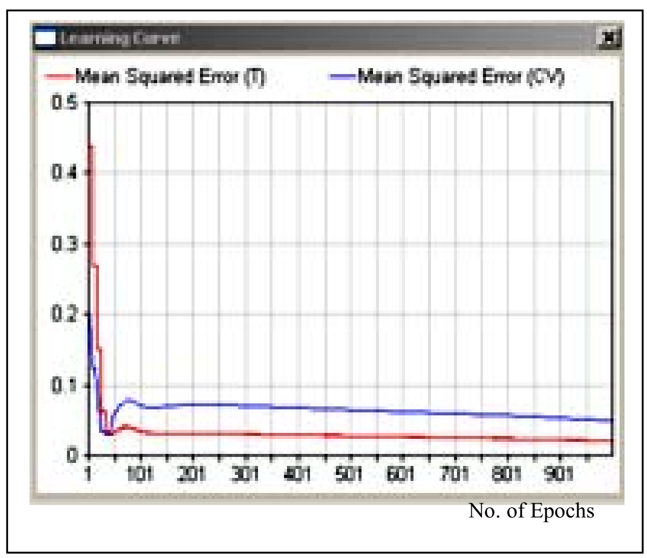

Figure 6. Mean square error vs. number of epochs

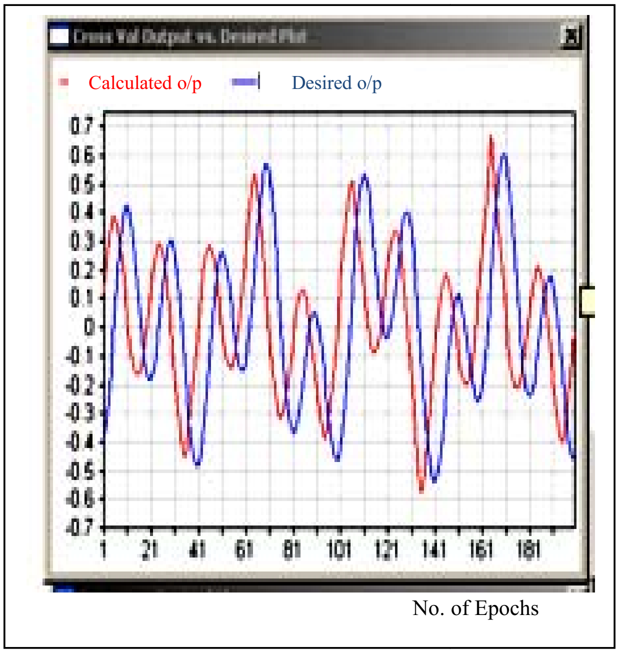

Figure 7. Cross validation output vs. desired output.

VII. CONCLUSION

This study developed the wavelet-based recurrent fuzzy network (WRFNN) by adding the simultaneous perturbation stochastic analysis (SPSA) as network learning algorithm. The algorithm was used to adjust all weights and other associated parameters of the network. The results show that the mean square error (MSE) reached a minimum in the 30^{th} epoch. Finally, the WRFNN with SPSA training algorithm overcome temporal problem in prediction and improve on the powerful representation capabilities of the WRFNN model [1]. We note that SPSA will impart the same advantage to other network structures.

REFERENCES

[1] C. J. Lin and C.C. Chin," *Prediction and Identification using Wavelet-Based Recurrent Fuzzy Neural Network. "IEEE Transaction on systems ,man, and Cybernetics, Vol.34,No.5,* pp. 2144–2154,*2004*.

[2] C. T. Lin and C. S. Lee, *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to System*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[3] G. G. Towell and J.W. Shavlik, "Extracting refined rules from knowledge-based neural networks," *Mach. Learn.*, vol. 13, pp. 71–101, 1993.

[4] C. J. Lin and C. T. Lin, "An ART-based fuzzy adaptive learning control network," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 477–496, Nov. 1997.

[5] L. X.Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 1414–1427, Nov./Dec. 1992.

[6] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 116–132, 1985.

[7] J. L. Elman, "Finding structure in time," *Cognit. Sci.*, vol. 14, pp.179–211, 1990.

[8] S. A. Billings and C. F. Fung, "Recurrent radial basis function networks for adaptive noise cancellation," *Neural Networks*, vol. 8, no. 2, pp. 273–290, 1995.

[9] L. Jin, P. N. Nikiforuk, and M. Gupta, "Approximation of discrete-time state-space trajectories usding dynamic recurrent neural networks," *IEEE Trans. Automat. Contr.*, vol. 40, pp. 1266–1270, July 1995.

[10] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Automat. Contr.*, vol.37, pp. 332–341, 1992.

[11] J. C. Spall, "Developments in Stochastica Optimization Algorithms with Gradient Approximation Based on Function Measurement" *Winter Simulation Conference*, pp. 207–214, 1994.

[12] J. N. Tsitsiklis, "Asynchronous stochastic approximation and Q-learning," *Mach. Learn.*, vol. 16, pp. 185–202, 1994.

[13] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pur. Appl. Math.*, vol. 41, 1998.

[14] T. Yamakawa, E. Uchino, and T. Samatsu, "Wavelet neural networks employing over-complete number of compactly supported nonorthogonal wavelets and their applications," in *Proc. IEEE Conf. Neural Networks*, vol. 3, 1994, pp. 1391–1396.

[15] D.W. C. Ho, P. A. Zhang, and J. Xu, "Fuzzy wavelet networks for function learning," *IEEE Trans. Fuzzy Syst.*, vol. 9, pp. 200–211, Feb. 2001.

[16] J. H. Kim, D. T. College, A. Gun, and G. Do," Fuzzy model based predicative control," in Proc, IEEE Int. Conf. Fuzzy System, vol. 1, Anchorage, Ak, May 1998, pp.405-409.