Hierarchical Wavelet Neural Networks

Sathyanarayan S. Rao and Ravikanth S. Pappu
Department of Electrical and Computer Engineering
Villanova University, Villanova, PA 19085.
phone: (215) 645-4971; fax: (215) 645-4436
email: pappu@vu-vlsi.vill.edu

Abstract: Neural Networks are capable of learning highly complex, nonlinear input-output mappings. This characteristic of neural networks enables them to be used in nonlinear system modelling and prediction applications. The wavelet decomposition, on the other hand. provides a method of examining a signal at multiple scales. In this paper, we draw upon the recently established connection between these two fields. A method is outlined which exploits the localized, hierarchical nature of wavelets in the learning of time series. This is achieved by having a dynamic network - one in which nodes are added to the network so as to progressively reduce the modelling error. This cascade correlation approach overcomes some of the disadvantages of a static network architecture. The learning algorithm is outlined and its performance is demonstrated using simulations.

INTRODUCTION

Time series prediction has important applications in many fields. The majority of commonly occurring time series are nonlinear in nature and hence can be approximated by neural networks. Since [1], neural networks have been increasingly employed for such tasks. Multilayer Perceptrons trained using backpropogation and Radial Basis Function Networks have been used successfully to predict nonlinear and chaotic data [2], [3].

Wavelet techniques have generated tremendous interest among the signal processing community in recent years. Wavelet decomposition involves representing arbitrary functions in terms of simpler basis functions at different scales and positions. In other words, the wavelet decomposition represents the signal as the sum of contributions of components at different scales. By its very definition, the wavelet decomposition is hierarchical in nature. For more details on wavelet theory and applications, the reader is referred to any of the number of books and articles which have appeared in the literature ([4-7]).

Wavelet neural networks represent a fruitful synthesis of ideas from neural networks and wavelet analysis. Recently the utility of wavelets in nonlinear system modelling and approximation was demonstrated in [8], [9], and [10]. In [10] the authors show that any arbitrary time series may be approximated by using wavelets

0-7803-0928-6/93$03.00 © 1993 IEEE

as activation functions in a typical 1+1/2 layer network. Our work draws upon ideas from [10] to provide an alternate method of using wavelets as activation functions. The algorithm presented in this paper is inspired by the hierarchical nature of the wavelet decomposition and cascade-correlation learning [13]. Hidden units are progressively added to the existing network to model the residual error from the previous approximation. This method overcomes the problem of selecting the number of hidden units in advance and also allows explicit control over the global approximation error.

This paper is organized as follows. The next section provides a brief introduction to wavelet networks. Then the proposed scheme is outlined and we illustrates how it overcomes some of the disadvantages of a static wavelet network. Details of the simulations are presented here. The last section summarizes the key results of this paper.

LEARNING IN WAVELET NETWORKS

In this section of the paper, we provide a concise description of the structure and learning ability of the wavelet network. The material in this section is, for the most part, is along the lines of [10].

Structure

Cybenko [11] proved the following result. If $\sigma(\ )$ is a continuous, discriminatory function, then finite sums of the form

$$f(x) = \sum_{i=1}^{N} w_i \sigma(a_i^T x + b_i) \tag{1}$$

are dense in the space of continuous functions defined on $[0,1]^n$ - where $w_i, b_i \in R$ and $a_i \in R^n$. This implies that any continuous function $f(\ )$ may be approximated by a weighted sum of $\sigma(\ )$ functions. The parameters w_i, b_i, and a_i may be determined by some optimization technique, such as backpropagation.

There is an analogous result in wavelet theory that enables arbitrary functions to be written as weighted sums of dilated and translated wavelets. This states that the sum

$$f(x) = \sum_{i=1}^{N} w_i det(D_i^{1/2}) \psi(D_i x - t_i) \tag{2}$$

is dense in $L^2(R^n)$. Here the t_i's are translation vectors and the $D_i = diag(d_i)$, where d_i's are the dilation vectors. $\psi(\ )$ is the basic wavelet whose translates and dilates form the basis for the space $L^2(R^n)$. Equation (2) immediately suggests the network structure shown in Fig.(1).
The primary advantages which wavelets have to offer over other activation

functions are:

- They guarantee the universal approximation property, i.e., (2).
- Initial values for the learning algorithm may be obtained from the continuous or discrete wavelet transform coefficients and thus enable faster convergence.
- If orthogonal wavelets are used, then adding or removing nodes from the network does not affect those weights which have already been trained. This is true since components at different scales lie in orthogonal subspaces.

These features lead to fast, localized, and hierarchical learning - in the spirit of [12].

Learning Algorithm

Learning in the network of Fig.(1) is accomplished by the stochastic gradient algorithm. We present the flow chart in Fig.(2) for a 1-D function (3), with the understanding that this procedure may be easily extended to higher dimensions. In 1-D, (2) reduces to

$$f(x) = \sum_{i=1}^{N} w_i \psi\left(\frac{x - t_i}{s_i}\right) + \bar{f} \tag{3}$$

where $\bar{f}$ is the estimated mean of $f(x)$ from the available samples and s_i is the scale parameter. The algorithm proceeds as shown in Fig.(2).

Let Θ be the vector containing all the parameters to be evaluated i.e., $[w_i, t_i, s_i, \bar{f}]$, $N_\Theta(x_k)$ be the output of the network for input x_k, and $J(\Theta, x_k, y_k)$ be the objective function to be minimized. This is defined as follows.

$$J(\Theta, x_k, y_k) = \frac{1}{2}[N_\Theta(x_k) - y_k]^2 \tag{4}$$

The gradient of $J(\Theta, x_k, y_k)$ with respect to the various parameters are computed and Θ is changed in the opposite direction of the gradient of the objective function. Further details of the training algorithm may be found in [10].

Observations

The above method prescribes an algorithm which uses wavelets as activation functions in a neural network. Admittedly, the selection of the number of hidden units (N) poses a problem. It is difficult to select a value of N directly from the data that will guarantee good approximation performance. Several statistical criteria such as the Akaike Criterion or the Minimum Description Length Principle have been suggested to determine the value of N. This problem may be rectified by approaching the problem from a cascade-correlation point of view. This is a powerful method which provides explicit control over the error and also exploits the hierarchical character of the wavelet decomposition. This is the inspiration for

the modified scheme proposed below.

PROPOSED SCHEME

As an alternative to the fixed network structure above, a dynamic network structure based on cascade correlation is proposed in this section. The principles of cascade correlation learning architecture are simple. It is a supervised learning architecture that builds a near-minimal multilayer network topology during the course of training. The key word here is "build". Initially the network contains only inputs, output units and the connections between them. The single layer of connections is trained repeatedly till there is no change in the error. The network's performance is evaluated at this stage. If the error is small enough, then we stop. Otherwise a new hidden unit is added in an attempt to reduce the error further.

Cascade correlation eliminates the need for the user to guess the size of the network and its topology in advance. A reasonably small network is built automatically. The learning is also faster than backpropagation for several reasons. First only a single layer of weights is being trained at any given time. There is no need to propagate any errors backwards as in backprop. Another reason is that this is a 'greedy" algorithm. Each node "grabs" as much of the residual error as it can. Thus, each node has a well defined role to play in the scheme of things. This is the motivation for adopting a variant of this method for the wavelet neural network.

In our method, the network begins learning by having a single "wavelon" - a combination of a translation, dilation, and wavelet lying along the same path from input to output. This wavelon is trained according to the algorithm in Fig.(2) till the error does not decrease any further. Then a new node is added and trained to model the residual error from the previous approximation. This process is repeated till the modelling error performance is satisfactory. Intuitively, it is clear that this hierarchical method of learning a time series is very well matched to the use of wavelets as activation functions. The hierarchical nature of the wavelet decomposition enables each wavelon to optimally complete its assigned task - i.e., to model the residual error from the previous approximation - before a new wavelon is added.

Simulation Results

In this section, we present two examples of wavelets based hierarchical learning. In the first example, a sinusoid is approximated by translations and dilations of a basic wavelet defined by

$$\psi(x) = -xe^{-\frac{1}{2}x^2} \tag{5}$$

Graph 1 is the approximation at the largest scale and graphs 2 through 5 are the added details. The approximation at different scales is clearly observed by

examining the y-axis of each plot. Graph 6 shows the sum of all the components at different scales along with the desired output. In this example the learning rate γ was 1. The final network had 5 wavelons and the sum of squared error (SSE) was 0.1026. Each wavelon was trained for 10 epochs of the training data. Comparable performance was achieved using a static network only after the network was trained for about 10^4 epochs.

The second example involves prediction of the logistic map. This is a quadratic map defined by the equation

$$x(t+1) = \alpha x(t)\,(1-x(t)) \tag{6}$$

This system is known to be chaotic for values of $\alpha > 3$. We use $\alpha = 4$ for which the system is guaranteed to be chaotic. The initial condition $x(0) = 0.1$. The inputs were $x(t)$ and the desired outputs were $x(t+1)$. The nonlinear map is shown in graph 7 along with the training data pairs. Daubechies' wavelets of order 2 were used as activation functions. Graphs 8 through 10 show the approximation of the training data at various scales and graph 11 shows the predicted time series. It is clear from the graphs that large structures in the input data are approximated by wavelets at large scales and detail is added at lower scales. The table below provides figures for the approximation and the prediction SSE at various levels.

	L=9 (ψ_9)	L=8 (ψ_8)	L=7 (ψ_7)	L=6 (ψ_6)	L=5 (ψ_5)
Approximation Error	0.3699	0.1766	0.1123	0.0942	0.0940
Prediction Error	0.3971	0.2409	0.1543	0.1694	0.1688

Table 1: APPROXIMATION AND PREDICTION SSE FOR THE LOGISTIC MAP

DISCUSSIONS AND SUMMARY

In summary, we have shown that a time series may be approximated by using a dynamic wavelet neural network. This method of learning a time series offers the advantage of having control over the error and eliminates the need to select the number of hidden units before the training phase begins. Simulation results have shown that the proposed algorithm is indeed well matched to the hierarchical character of wavelets. There are two issues which merit further investigation. First, it is likely that a more efficient algorithm will have a scale dependent learning rate - with large scales having a large rate and small scales having a smaller rate. In other words, the approximation of large structures is done quickly and the detail is added at a rate proportional to the scale of approximation. Another issue is the choice of the wavelet family. There are several families of wavelets available and one has to

choose one of them. It appears that continuous, orthogonal wavelets offer the most advantages. Desirable properties of wavelets for prediction applications have to be established and various families have to be compared. These ideas are deferred to future work.

REFERENCES

[1] Alan Lapedes and Robert Farber, "Non Linear Signal Processing Using Neural Networks: Prediction and System Modelling", Los Alamos National Laboratory Report LA-UR-87-2662, 1987.

[2] M. Casdagli, "Nonlinear Prediction of Chaotic Time Series", Physica D, vol. 35, pp. 335-356, 1989.

[3] M. Niranjan et. al., "A Nonlinear Model for Time Series Prediction and Signal Interpolation", Proceedings of ICASSP 1991, pp. 1713-1716.

[4] S. G. Mallat, "A Theory for Multiresolution Decomposition: The Wavelet Representation", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 2, pp. 674-693, July 1989.

[5] I. Daubechies, "Orthonormal Bases of Compactly Supported Wavelets", Communications on Pure and Applied Math., vol. 41, pp. 909-996, 1988.

[6] J. M. Combes et. al., editors, Wavelets: Time-Frequency Methods in Phase Space, Springer-Verlag, 1989.

[7] C. K. Chui, editor, Wavelets: A Tutorial in Theory and Applications, Academic Press, 1992.

[8] B. Bakshi et. al, "Wave-Net: A Multiresolution, Hierarchical Neural Network with Localized Learning", American Institute of Chemical Engineers Journal, July 1992.

[9] S. S. Rao et. al, 'Nonlinear Time Series Prediction Using Wavelet Networks', Proceeding of the World Congress on Neural Networks, 1993 (to appear).

[10] Q. Zhang et. al, "Wavelet Networks", IEEE Transactions on Neural Networks, vol. 3, pp. 889-898, November 1992.

[11] G. Cybenko, "Approximation by Superposition of a Sigmoidal Function", Mathematics of Control, Signals and Systems, vol. 2, pp. 303-314, 1989.

[12] J. Moody, "Fast Learning in Multiresolution Hierarchies", Research Report, Yale University, YALEU/DCS/RR-681, 1989.

[13] S. Fahlman et. al, "The Cascade-Correlation Learning Architecture", Technical Report CMU-CS-90-100, School of Computer Science, Carnegie Mellon University, August 1992.

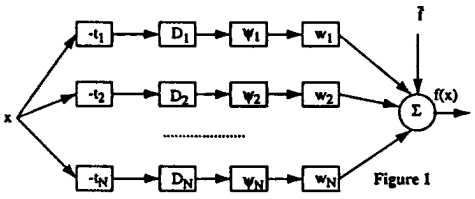


Figure 1

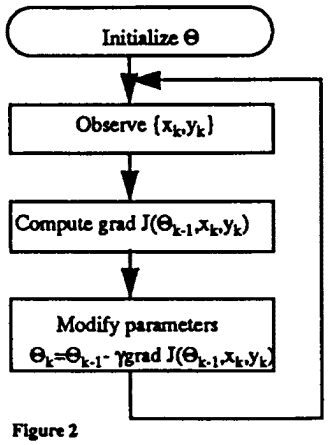
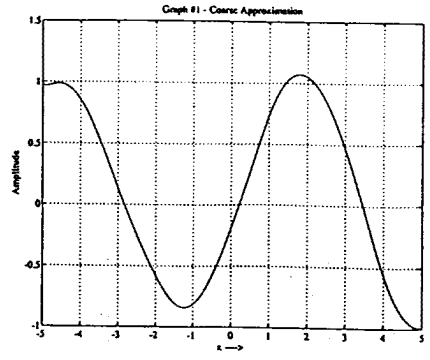
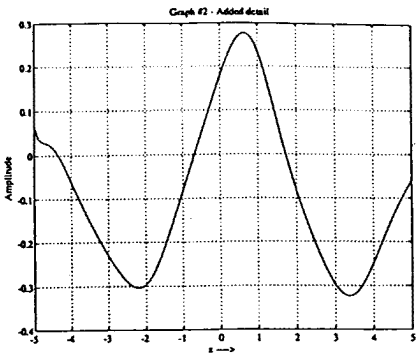
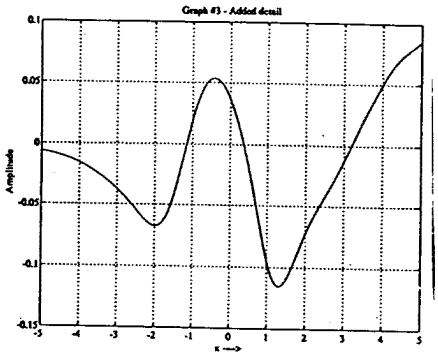
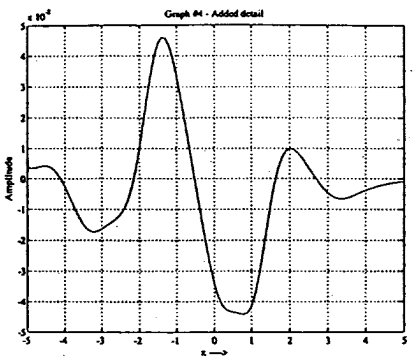


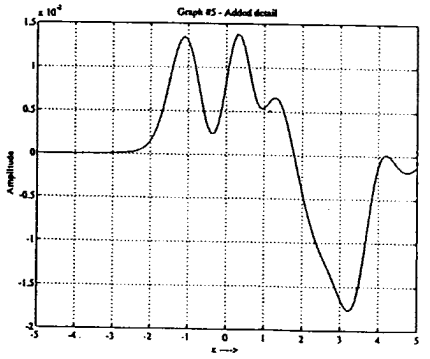
Figure 2

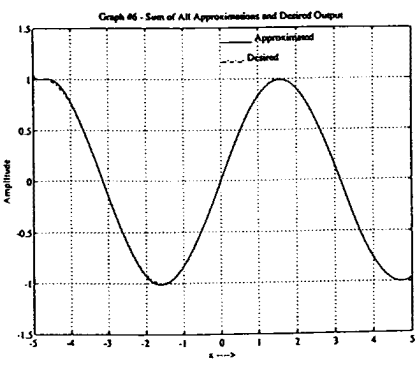

Graph #1 - Coarse Approximation
Amplitude
x ⟶

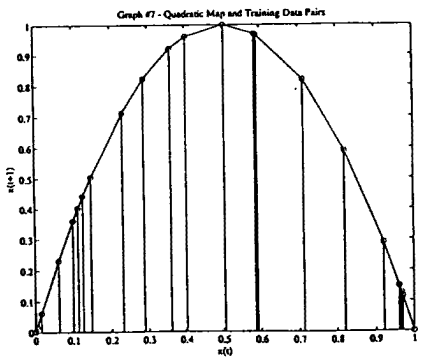

Graph #2 - Added detail
Amplitude
x ⟶

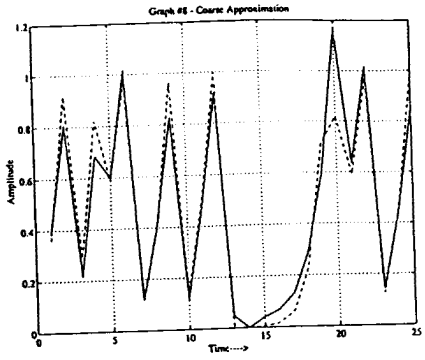

Graph #3 - Added detail
Amplitude
x ⟶

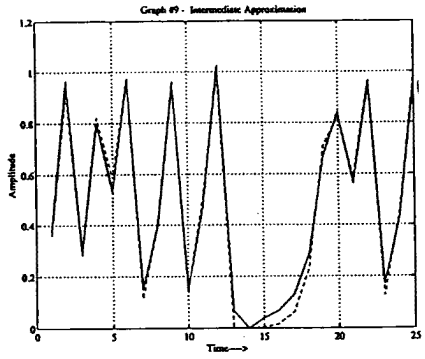

x 10^-2
Graph #4 - Added detail
Amplitude
x ⟶

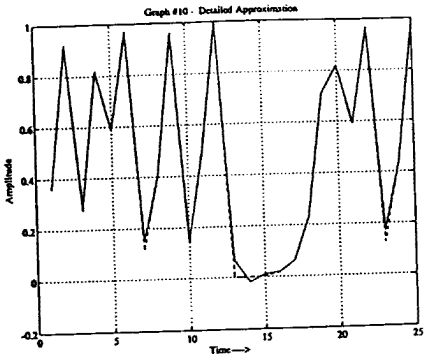

x 10^-2
Graph #5 - Added detail
Amplitude
x ⟶

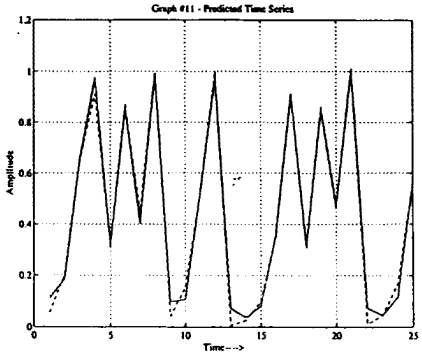

Graph #6 - Sum of All Approximations and Desired Output
Approximated
Desired
Amplitude
x ⟶











Approximation and prediction of the logistic map. Dashed lines represent true values.