Paper

A Local Linear Adaptive Wavelet Neural Network

Student Member Ting Wang (Chiba University)
Member Yasuo Sugai (Chiba University)

Wavelet neural networks employing wavelets as the activation functions recently have been researched as an alternative approach to the traditional neural networks with sigmoidal activation functions. In this paper, we proposed a new type of wavelet neural network by introducing local linear models, which are used in some neuro-fuzzy systems, as powerful weights instead of straightforward weights employed in the previous wavelet neural networks. The proposed network is called the local linear adaptive wavelet neural network. Its effectiveness is examined by the network performances on function approximation and chaotic time series prediction problems. In these experiments, the proposed local linear adaptive wavelet neural network performed well and compared favorably to the previous wavelet neural network.

Keywords:
adaptive wavelet neural network, local linear model, function approximation, chaotic time series prediction

1. Introduction

Developing models from observed data, or learning maps between input and output spaces, is a fundamental problem in the areas such as dynamical system control, signal processing, system identification and many other fields. Multilayer perceptron neural network (MLP) with the sigmoidal functions as the activation functions has been established as a general nonlinear fitting tool [(1)~(3)]. On the other hand, instead of the sigmoidal functions, a number of locally active nonlinear basis functions such as polynomial basis functions, Gaussian radial basis functions, spline basis functions and wavelet basis functions have also been studied as activation functions. The corresponding basis function networks, such as radial basis network (RBF) and wavelet neural network (WNN), have been developed as powerful nonlinear fitting methods [(4)~(7)].

Among the basis function networks, wavelet neural networks employing wavelets as activation functions have been paid a special attention because in terms of the wavelet transformation theory, the wavelet representation of a function can reveal properties of the function in localized regions of both the time space and frequency space [(8)(9)]. Zhang and Benveniste first introduced wavelets to neural networks [(7)]. Pati and Krishnaprasad demonstrated that it is possible to construct a theoretical description of feedforward neural networks in terms of wavelet decompositions [(10)]. Employing the orthonormal scaling functions (in the theory of wavelet transformation, scaling functions are the functions used to generate wavelet functions) as the activation functions, the wavelet network developed by Jun Zhang et.al can provide a unique and efficient representation of a given function [(11)]. Yamakawa et.al. proposed a wavelet neural network in which compact supported non-orthogonal wavelets are used [(12)]. The adaptive wavelet neural network is developed by Kadambe and Srinivasan to let the network more flexible [(13)]. All of these wavelet neural networks have identical architecture, though the wavelets used as activation functions are different. In this paper, we introduce local linear models (LLM) whose good performances have been shown in some neuro-fuzzy systems, as powerful weights to an adaptive wavelet neural network (AWNN). The local linear models connect the hidden layer to the output layer instead of the previous straightforward weights. Therefore this type of wavelet neural network, called a local linear adaptive wavelet neural network (LLAWNN), is proposed as an alternative approach to nonlinear mapping problems.

The paper is organized as follows. The proposed network is introduced in section 2. The experiments performed on function approximation and chaotic time series prediction problems are described in section 3. Finally, conclusions are derived in the last section.

2. Local Linear Adaptive Wavelet Neural Network

After a brief review of some basic concepts about wavelet neural networks and local linear models, our proposed local linear adaptive wavelet neural network is introduced in this section.

2.1 Adaptive Wavelet Neural Network In terms of wavelet transformation theory, wavelets in the following form

$$\Psi = \{\Psi_i = |\boldsymbol{a}_i|^{-\frac{1}{2}}\psi(\frac{\boldsymbol{x}-\boldsymbol{b}_i}{\boldsymbol{a}_i}) : \boldsymbol{a}_i, \boldsymbol{b}_i \in \boldsymbol{R}, i \in Z\} \quad (1)$$

$$\boldsymbol{x} = (x_1, \cdots, x_N)$$

$$\boldsymbol{a}_i = (a_{i1}, \cdots, a_{iN})$$

$$\boldsymbol{b}_i = (b_{i1}, \cdots, b_{iN})$$

is a family of functions generated from one single function $\psi(\boldsymbol{x})$ by the operation of dilations and translations. $\psi(\boldsymbol{x})$, which is localized in both the time space and frequency space, is usually called a mother wavelet and the parameters $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are called the scalar parameter and translation parameter, respectively [8].

According to the previous researches of WNN (7), (10)~(12), the output of a wavelet neural network is given by

$$f(\boldsymbol{x}) = \sum_{i=1}^{M} w_i \Psi_i(\boldsymbol{x}) = \sum_{i=1}^{M} w_i |\boldsymbol{a}_i|^{-\frac{1}{2}} \psi(\frac{\boldsymbol{x}-\boldsymbol{b}_i}{\boldsymbol{a}_i}) \cdots\cdots\cdots\cdots (2)$$

where the activation functions of the hidden layer units are the wavelets Ψ and w_i is the straightforward weight connecting the i-th hidden layer unit to the output layer unit. The number of the hidden layer units is M.

Usually, each component a_{ij}, b_{ij} of the parameters $\boldsymbol{a}$ and $\boldsymbol{b}$ is in the following discretization form:

$$\{(a_{ij}, b_{ij}) = (a_0^{-m}, nb_0 a_0^{-m}) : m \in Z, n \in Z\} \cdot (3)$$

with $a_0 = 2$ and $b_0 = 1$, typically [10]. Given a set of training data, if m and n are predetermined by some analyses of the data set and the localization of ψ, the only adjustable parameters of the wavelet neural network (2) are therefore the output layer weights w_i.

On the other hand, if the discretization form (3) is not considered, not only the connection weights w_i, but also the scalar parameters $\boldsymbol{a}_i$ and the translation parameters $\boldsymbol{b}_i$ of the network (2) can be determined by some training algorithms based on the given training data set. Therefore the wavelet neural network (2) becomes an adaptive wavelet neural network [13]. Because the localization of a hidden layer unit corresponds to the parameters $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$, "adaptive" means the hidden layer units can adapt their receptive fields to the distribution of the input vectors during the training process. In this study, this kind of adaptive wavelet neural network is considered.

2.2 Local Linear Adaptive Wavelet Neural Network Due to the network architecture and the localized validity of the wavelets, the adaptive wavelet neural network (2) can be viewed as a kind of standard basis function network. The output of a standard basis function network given as :

$$y = \sum_{i=1}^{M} w_i \Phi_i(\boldsymbol{x}) \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (4)$$

is a weighted linear combination of many locally active non-linear basis functions $\Phi_i (i = 1, \cdots, M)$, where w_i is the associated weight with Φ_i [14]. In the adaptive wavelet neural network (2), it is obvious that the wavelets Ψ are the corresponding locally active non-linear basis functions.

It is well known that an intrinsic feature of the basis function networks is the localized activation of the hidden layer units, so that the connection weights associated with the units can be viewed as locally accurate piecewise constant models whose validity for a given input is indicated by the activation functions [15]. Compared to MLP, this local capacity provides some advantages such as the learning efficiency and the structure transparency. On the other hand, due to the crudeness of the local approximation(piecewise constant models are integrated by their associated localized basis functions), a large number of basis function units have to be employed to approximate a given system. As reported in the previous research (11), a shortcoming of the wavelet neural network also shared by the RBF network, is that for higher dimensional problems, many hidden layer units are needed.

In order to take advantage of the local capacity of the wavelet basis functions without having too many hidden units, here we propose an alternative type of wavelet neural network. Given that the input vector $\boldsymbol{x}$ is N dimensional and the number of the hidden layer units is M, the network output of the k-th unit in the output layer is given as :

$$y_k = \sum_{i=1}^{M} (w_{i,0} + w_{i,1}x_1 + \cdots + w_{i,N}x_N)\Psi_i(\boldsymbol{x}) \quad (5)$$
$$\boldsymbol{x} = (x_1, \cdots, x_N)$$

where, instead of the straightforward weight w_i (piecewise constant model) in (2), the following linear model

$$u_i = w_{i,0} + w_{i,1}x_1 + \cdots + w_{i,N}x_N \cdots\cdots\cdots (6)$$

is introduced as a powerful representation of weights. Because the activities of the linear models $u_i (i = 1, \cdots, M)$ are determined by the associated locally active wavelet functions $\Psi_i (i = 1, \cdots, M)$, u_i is locally valid so that it is called local linear model. The idea of introducing local linear models to wavelet neural network is inspired by the researches of the local linear neuro-fuzzy system [16] [17]. We call the proposed network (5) the local linear adaptive wavelet neural network.

By replacing the straightforward weights with the local linear models, the proposed network can be viewed as an extension of the previous wavelet neural network. In other words, the proposed network degenerates to a previous wavelet neural network if

$$w_{i,0} \neq 0 \quad \text{and} \quad w_{i,1} = w_{i,2} = \cdots = w_{i,N} = 0. (7)$$

Since the basis functions associated with the local linear models could cover larger region of the input space than with the straightforward weights [16] [17], fewer hidden layer units are required than before. For complex or high dimensional problems, the previous network possesses only a single parameter (the straightforward weight w_i) for each unit, but a large number of units is required. In contrast, though the proposed network possesses more than one parameter (parameters $w_{i,j}$'s in the local linear model u_i) for each unit, the required number of units is already significantly reduced.

A. Network Structure

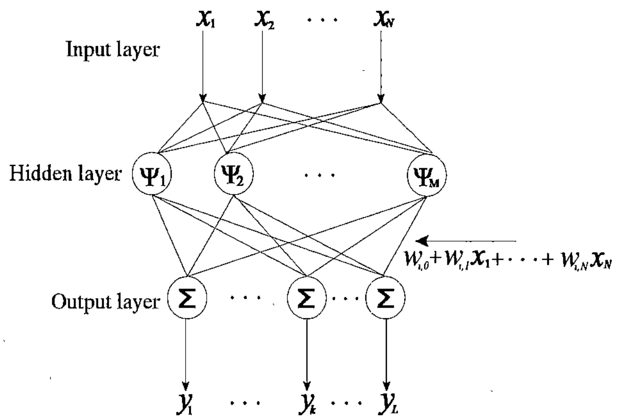


Fig. 1. Network structure

The structure of the proposed network (5) is shown in Fig.1. It has a feedforward structure consists of a single hidden layer. The activation functions of the hidden layer units are the wavelets (2) and the weights connecting the hidden layer units to the output layer units are the local linear models (6). The working of the proposed network can be viewed as to decompose the complex, nonlinear system into a set of locally active submodels, then smoothly integrate those submodels by their associated wavelet basis functions. It means that this structure has the advantages inherent to the local nature of the wavelet basis functions while, by employing the more powerful local models (6) associated with those locally basis functions, it is not requiring as many basis functions as before to achieve the desired accuracy.

Considering the network architecture and the localized validity of the wavelet basis functions Ψ, we refer the proposed network (5) to a kind of local model network. Local model networks described in the form :

$$y=\sum_{i=1}^{M} f_i(\boldsymbol{W},\boldsymbol{x})\rho_i(\boldsymbol{x}) \quad \cdots\cdots (8)$$

are networks composed of locally accurate models $f_i(i=1,\cdots,M)$, where the outputs are interpolated by locally active basis functions ρ_i [15][18]. Obviously, in the proposed local linear adaptive wavelet neural network (5), the local active basis functions used in previous local model networks, such as Gaussian basis functions, are replaced by the wavelet basis functions localized in both the time space and frequency space and f_i associated with each basis function is the local linear model (6).

B. Training Algorithm

Similar to the traditional neural networks and wavelet neural networks [20], training of the proposed local linear wavelet neural network can also be performed by employing the iterative gradient-descent method.

Given a set of training data T_P,

$$T_P=\{(\boldsymbol{x}_i,f(\boldsymbol{x}_i)),i=1,\cdots,P\}, \quad \cdots\cdots (9)$$

where $\boldsymbol{x}_i$, $f(\boldsymbol{x}_i)$ are the input vector and the corresponding ouput, the objective function to be minimized is

$$E(\boldsymbol{\theta})=\frac{1}{2}\sum_{p=1}^{P}(f_p-Y_p)^2, \quad \cdots\cdots (10)$$

where vector $\boldsymbol{\theta}$ is the collection of all of the parameters $\boldsymbol{w}_i$, $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ in (5) that should be adjusted. f_p and Y_p in Eq.(10) are the teaching signal and the output of the proposed network (5) of the p-th training example, respectively.

Therefore, $\boldsymbol{\theta}$ is updated from step t to the next step as follows :

$$\boldsymbol{\theta}(t+1)=\boldsymbol{\theta}(t)+\Delta\boldsymbol{\theta}(t), \quad \cdots\cdots (11)$$

$$\Delta\boldsymbol{\theta}(t)=-\eta\left.\frac{\partial E}{\partial\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}(t)}+\mu\Delta\boldsymbol{\theta}(t-1), \quad \cdots\cdots (12)$$

where the learning rate η scales the stepsize and μ is the momentum term added to make the training process more stable.

The partial derivative of the objective function with respect to $\boldsymbol{\theta}$ is

$$\frac{\partial E}{\partial\boldsymbol{\theta}}=-\sum_{p=1}^{P}(f_p-Y_p)\frac{\partial Y_p}{\partial\boldsymbol{\theta}}. \quad \cdots\cdots (13)$$

Therefore, explicit formulae for the partial derivatives of the objective function with respect to $w_{i,j}$ of u_i, $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are given as follows :

$$\frac{\partial E}{\partial w_{i,0}}=-\sum_{p=1}^{P}(f_p-Y_p)\Psi_i(\boldsymbol{x}_p) \quad \cdots\cdots (14)$$

$$\frac{\partial E}{\partial w_{i,j}}=-\sum_{p=1}^{P}(f_p-Y_p)x_{jp}\Psi_i(\boldsymbol{x}_p) \quad \cdots\cdots (15)$$

where, x_{jp} is the value of the j-th component of the input vector $\boldsymbol{x}$ for the p-th training example.

$$\frac{\partial E}{\partial\boldsymbol{a}_i}=-\sum_{p=1}^{P}(f_p-Y_p)u_i\frac{\partial\Psi_i(\boldsymbol{x}_p)}{\partial\boldsymbol{a}_i} \quad \cdots\cdots (16)$$

$$\frac{\partial E}{\partial\boldsymbol{b}_i}=-\sum_{p=1}^{P}(f_p-Y_p)u_i\frac{\partial\Psi_i(\boldsymbol{x}_p)}{\partial\boldsymbol{b}_i} \quad \cdots\cdots (17)$$

3. Experimental Results

The proposed network is examined by its performances on function approximation and chaotic time series prediction problems. All experiments are carried out on a Pentium 500-MHz PC with the C programming language. Before discussing the experimental results, the following notes are given first, which are employed in all of the experiments.

A. Wavelets Used as Basis Functions

As mentioned above, wavelets are generated from a mother wavelet $\psi(\boldsymbol{x})$. Here, for a problem with N inputs, the mother wavelet $\psi(\boldsymbol{x})$, is given as follows :

$$\psi(\boldsymbol{x}) = \prod_{n=1}^{N} \psi(x_n) \cdots\cdots\cdots\cdots \quad (18)$$

this is the most frequently chosen scheme to generate a multidimensional wavelet function by the tensor product of a one dimensional wavelet function. The basic one dimensional mother wavelet $\psi(x)$ we used is described as :

$$\psi(x) = -x \exp\Big(-\frac{x^2}{2}\Big) \cdots\cdots\cdots\cdots \quad (19)$$

Therefore, according to (2), wavelets used as basis functions, with N inputs, can be generated from (18) as

$$\Psi_i(\boldsymbol{x}) = \prod_{n=1}^{N} |a_{in}|^{-\frac{1}{2}} \psi\Big(\frac{x_n - b_{in}}{a_{in}}\Big). \cdots\cdots \quad (20)$$

B. Testing Criterion

To evaluate the performance of the proposed network, a testing criterion is needed. Here, Root Mean Square Error (RMSE) described in the following form is employed :

$$RMSE = \sqrt{\frac{1}{PT}\sum_{j=1}^{PT}(f_k(\boldsymbol{x}_j) - y_k(\boldsymbol{x}_j))^2} \cdots \quad (21)$$

where f_k and y_k are the desired value and the network output of the kth unit in the output layer, respectively. PT is the total number of the testing data.

3.1 Function Approximation Function approximation is a fundamental problem in many fields such as system estimation, signal processing, control, etc. It deals with the problem of learning a mapping between an input and an output space from a set of examples of input-output pairs. In this section, the performance of the proposed network (5) on this problem is shown by employing two multivariable functions as experimental examples.

3.1.1 Example 1 The first example is given by

$$f(x_1, x_2) = \frac{\sin(\pi x_1)\cos(\pi x_2) + 1.0}{2} \cdots\cdots \quad (22)$$
$$(x_1, x_2 \in [-1, 1])$$

The approximations of this example with a traditional sigmoidal neural network and a wavelet neural network are reported [19].

Over the domain $[-1,1] \times [-1,1]$, we generated a set of 49 data points, equally spaced on a 7×7 grid, as the training data set. The testing data set used to calculate the RMSE is a set of 400 data points which are equally spaced on a 20×20 grid. Experimental result of the proposed local linear adaptive wavelet neural network is shown in Tbl.1. The number of parameters means the number of adjustable parameters of which the values should be determined by training. In the case of the proposed network (5), all of the scalar parameter a_{in} and translation parameter b_{in} of the employed wavelets (20) and the weights w_{in} in the local linear models (6) are adjustable. a_{in} and b_{in} are initialized as

$$\begin{cases} a_{in} = 0.2(\beta_n - \alpha_n) \\ b_{in} = 0.5(\beta_n + \alpha_n) \end{cases} \cdots\cdots\cdots \quad (23)$$

to take advantage of the locally active property of the wavelets according to the previous study [20], where $[\alpha_n, \beta_n]$ denotes the domain of the nth component of the input vectors. On the other hand, each w_{in} is initialized to a small random value within $[-0.1, 0.1]$.

For comparison, we also give results of the sigmoidal neural network and the adaptive wavelet neural network. The sigmoidal function defined as

$$\sigma(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \cdots\cdots\cdots\cdots \quad (24)$$

is used as the activation function in the sigmoidal neural network. The number of the hidden units is set to 10 and all of the adjustable parameters, the connection weights and the thresholds, are initialized to small random values within $[-0.1, 0.1]$ [19]. The adaptive wavelet neural network is the above described network (2) with the same wavelets (20) of the proposed network as basis functions, where the initializations of the scalar and the translation parameters are also performed as (23). The initial value of each straightforward weight w_i is also a small random value within $[-0.1, 0.1]$.

Table 1. Approximation results of function (22)

	network structure	number of parameters	computation time (sec)	*RMSE*
LLAWNN	2-4-1	28	57.3	0.0158
AWNN	2-8-1	40	103.8	0.0165
NN	2-10-1	41	113.2	0.0432

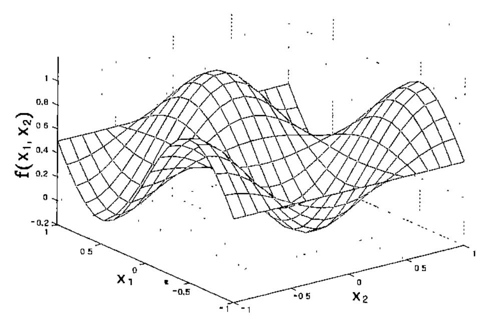


Fig. 2. Original function of example 1

As mentioned above, the number of parameters in Tbl.1 is the number of adjustable parameters of which the values should be determined by training. In the case of LLAWNN and AWNN, it is the sum of the number of the scalar parameters, the translation parameters and the connetion weights. In the case of the sigmoidal neural network, it is the sum of the number of the thresholds and the connection weights. Given that the input vector $\boldsymbol{x}$ is N dimensional and the number of the hidden layer units is M, the number of parameters of the proposed network, the previous wavelet neural network and the sigmoidal neural network is $(3N+1)M$, $(2N+1)M$ and $(N+2)M+1$, respectively.

For each network, the training was stopped when the value of the objective function (10) was no longer changing and the result of RMSE calculated with 400 points of testing data is the average of 10 experiments with different initial values of the adjustable parameters. The graph of the function Eq.(22) is shown in Fig.2. The network outputs of 400 points of testing data are shown in Fig.3~5. The corresponding RMSE of each figure is 0.0156 (Fig.3), 0.0166 (Fig.4) and 0.0422 (Fig.5), which is the typical one of the 10 experiments for each network. These experimental results showed that in terms of employing the local linear models (6) as powerful weights associated with the locally active basis functions instead of the straightforward weights w_i, the proposed network can learn the input-output mapping with a smaller number of basis functions with sufficient accuracy.

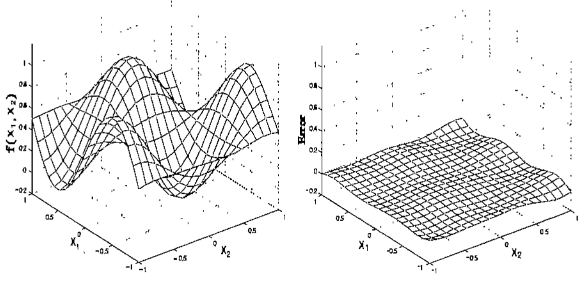

Network output surface Error surface

Fig. 3. Result of the proposed network

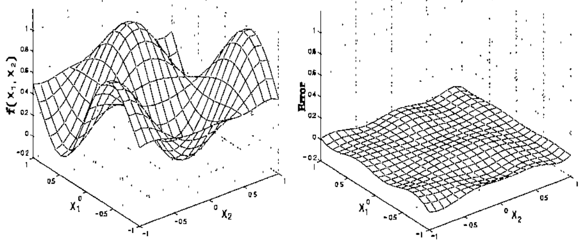

Network output surface Error surface

Fig. 4. Result of the previous wavelet neural network

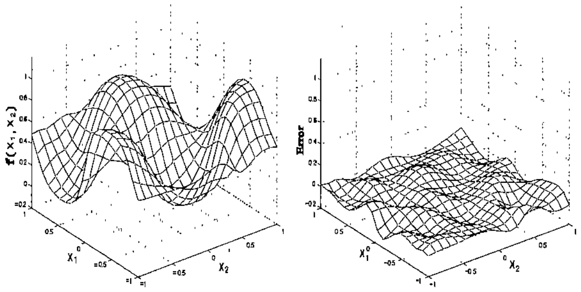

Network output surface Error surface

Fig. 5. Result of sigmoidal neural network

3.1.2 Example 2

The second example given by

$$f(x_1, x_2) = \cos(2\pi x_1)\cos(2\pi x_2)\exp^{-(x_1^2+x_2^2)} \quad (25)$$
$$(x_1, x_2 \in [-1, 1])$$

is difficult to be approximated by a sigmoidal neural network due to its complication [21].

Over the domain $[-1,1]\times[-1,1]$, we generated a set of 169 data points, equally spaced on a 13×13 grid, as the training data set. The testing data set used to calculate the RMSE is a set of 400 data points which are equally spaced on a 20×20 grid. We show the results with the proposed network (5) and the previous adaptive wavelet neural network (2) in Tbl.2. Details about the initializations of adjustable parameters are the same as the above example. The graph of the function Eq.(25) is shown in Fig.6. With the corresponding RMSE values of 0.0181 and 0.0210, the network outputs of 400 points of testing data are illustrated in Fig.7 and Fig.8. These experimental results also show the proposed network performed well on this function approximation problem and compared favorably to the previous wavelet neural network.

Table 2. Approximation results of function (25)

	network structure	number of parameters	computation time (sec)	*RMSE*
LLAWNN	2-16-1	112	646.2	0.0183
AWNN	2-29-1	145	1032.5	0.0201

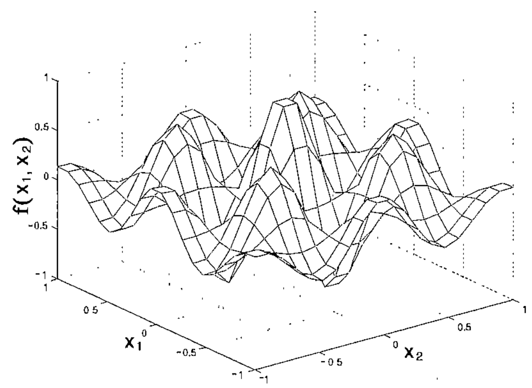
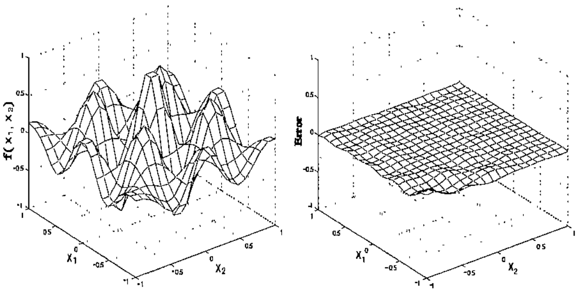
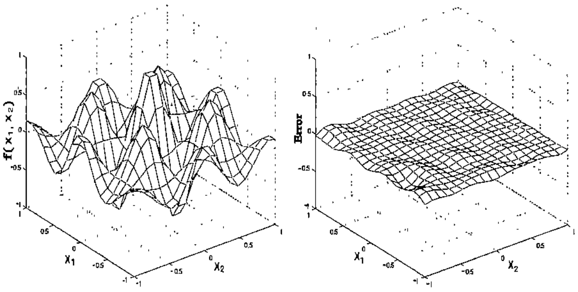


Fig. 6. Original function of example 2



Network output surface Error surface

Fig. 7. Result of the proposed network



Network output surface Error surface

Fig. 8. Result of the previous wavelet neural network

3.2 Chaotic Time Series Prediction The problem of time series prediction can be formulated as : given a time series $y(k)(k = 1, 2, \cdots)$, employing

$$y(k-m), y(k-m+1), \cdots, y(k) \quad \cdots\cdots\cdots\cdots (26)$$

to determine $y(k+l)$, where m and l are fixed positive integers. Because applications of time series prediction can be found in various domains such as signal processing, inventory and production control, economic planning and lots of other fields, it is a very important practical problem. Chaotic time series are generated from deterministic nonlinear systems. Because they are so complicated that appear to be "random" time series, they are not easy to be predicted. In this section, the effectiveness of the proposed network on this prediction problem is shown. Two well known chaotic time series are employed as examples in this research.

A. Mackey-Glass series

The Mackey-Glass series is defined by the following differential equation [22].

$$\frac{dy(t)}{dt} = \frac{ay(t-\tau)}{1+y(t-\tau)^b} - cy(t) \quad \cdots\cdots\cdots\cdots (27)$$

with $a = 0.2$, $b = 10$, and $c = 0.1$. This unperturbed system has an inherent delay time τ. It is known that when $\tau > 17$, the above equation (27) exhibits chaotic behavior and one remarkable feature of this system is that the dimension of its strange attractor increases monotonically with τ. Varying the parameter τ, strange attractors with an arbitrarily large dimension can be generated. In our experiment, we choose $\tau = 30$ to generate a Mackey-Glass chaotic time series. The data set of 1200 points of this chaotic time series which generated using $y(0) = 1.0$ and $y(t-\tau) = 0 (0 \leq t < \tau)$ is shown in Fig.9.

With different m, l in (26), the input and output vector of the Mackey-Glass series prediction problem are different. Here, $m = 6$ and $l = 1$ are employed [23], therefore the input vector is $\boldsymbol{x} = \{y(t-6), \cdots, y(t-1), y(t)\}$ and the output vector is $\boldsymbol{y} = \{y(t+1)\}$. The first 500 points of the series were used as the training data, and the further 500 points were used as the testing data.

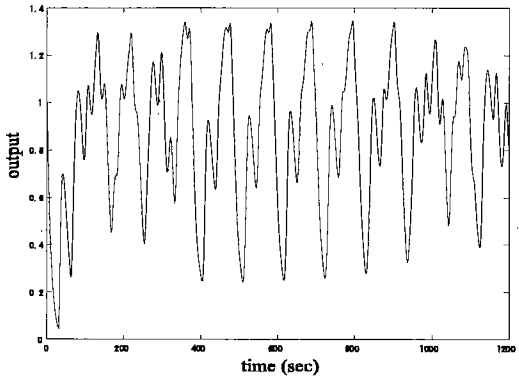


Fig. 9. Mackey-Glass chaotic time series

B. Henon Attractor

Henon attractor is a strange attractor discovered by Michel Henon [24]. It is given by :

$$\begin{cases} x_{n+1} &= 1 - ax_n^2 + y_n \\ y_{n+1} &= bx_n \end{cases} \quad \cdots\cdots\cdots\cdots (28)$$

where $a = 1.4$ and $b = 0.3$. With the initial value $x_0 = y_0 = 0$, 500 points of series x and y are generated. The $x - y$ phase portrait of the Henon attractor is shown in Fig.10.

For the Henon attractor prediction problem, m, l are set to $m = 1$ and $l = 1$, therefore the input vector is $\{x_n, x_{n-1}\}$ and the output vector is $\{x_{n+1}\}$. The experiments on this problem are performed employing the first 250 points of the x series as the training data, and the further 250 points as the testing data.

C. Results

The results of employing the proposed network to the Mackey-Glass series and the Henon attractor prediction problems are summarized in Tbl.3, where the values of RMSE were calculated by (21) with the testing data of each problem. The results of the previous wavelet neural network with straightforward weights are also presented in Tbl.3. For both the Mackey-Glass series and the Henon attractor prediction, it can be seen that both the wavelet neural network with local linear models as powerful weights and with simple weights could predict the chaotic time series with sufficient accuracy. However, compared to the wavelet neural network with

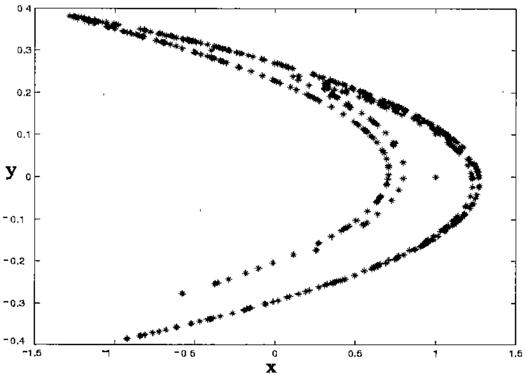
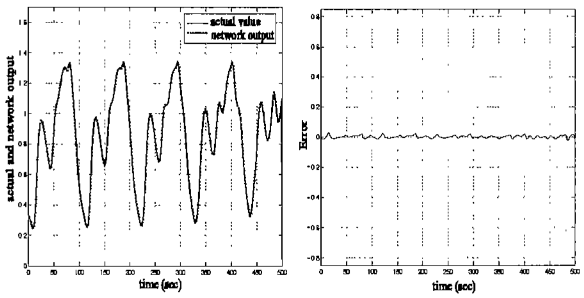


Fig. 10. Henon attractor

straightforward weights, the proposed network could achieve a similar accuracy with a smaller size due to the introduction of the local linear models. The prediction performances are illustrated in Fig.11 and Fig.12, where the predicted and actual values of the testing data are demonstrated in a same figure and the plotted prediction error is the difference between the predicted and actual values.

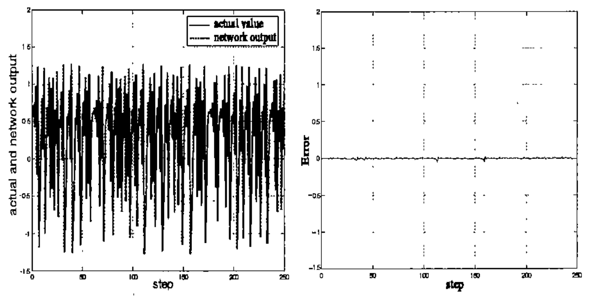
Table 3. Results of chaotic time series prediction

time series	network	structure	number of parameters	computation time (sec)	$RMSE$
Mackey Glass	LLAWNN	6-2-1	38	672	0.009
	AWNN	6-5-1	65	1558.4	0.010
Henon attractor	LLAWNN	2-2-1	14	71	0.008
	AWNN	2-8-1	40	204.3	0.011



Prediction output Prediction error

Fig. 11. Result of Mackey-Glass series



Prediction ouput Prediction error

Fig. 12. Result of Henon attractor

4. Conclusion

In this paper, we proposed a wavelet-based neural network which is called the local linear adaptive wavelet neural network. The basic idea is to replace the straightforward weights of previous wavelet neural networks by introducing the local linear models as powerful weights. This was inspired by the studies of some neuro-fuzzy systems. Furthermore, from the point of view of the network architecture, it can also be considered that the proposed local linear adaptive wavelet neural network is developed as a new type of local model network. The local active basis functions used in previous local model networks, such as Gaussian basis functions, are replaced by wavelet basis functions that localized in both the time space and frequency space. The working of the proposed network can be viewed as to decompose the complex, nonlinear system into a set of locally active submodels, then smoothly integrate those submodels by their associated wavelet basis functions. Experimental results obtained in the function approximation and chaotic time series prediction problems indicated that the proposed network could achieve sufficiently good performances with a smaller number of wavelet basis functions compared to the previous wavelet neural network.

For future work, it would be interested in the optimization of the network architecture, efficient parameter training algorithms and practical applications of the proposed network.

(Manuscript received May 23, 2001, revised September 7, 2001)

References

(1) G. Gybenko, "Approximation by superposition of a sigmoidal function", Mathematics of Control, Signals and Systems, Vol.2, pp.303-314, 1989
(2) S. Chen and S.A. Billings, "Nonlinear system identification using neural networks", Int. Journal of Control, Vol.51, No.6, pp.1191-1214, 1990
(3) M.J. Willis et al., "Artificial neural network in process estimation and control", Automatica, Vol.28, No.6, pp.1181-1187, 1992
(4) G.P. Liu, V. Kadirkamanathan and S.A. Billings, "On-line identification of nonlinear systems using Volterra polynomial basis function neural networks", Journal of Neural Networks, Vol.11, pp.1645-1657, 1998
(5) J. Moody and C. Darken, "Fast learning in networks of locally-tuned processing units", Neural Computation, Vol.1, pp.281-294, 1989
(6) K. Hlavackova and R. Neruda, "Radial basis function networks", Neural Network World, vol.1, pp.93-101, 1993
(7) Q. Zhang and A. Benveniste, "Wavelet networks", IEEE Trans. on Neural Networks, Vol.3, No.6, pp.889-898, 1992
(8) Charles K. Chui, An introduction to wavelets, Academic Press, 1992
(9) I. Daubechies, Ten lectures on wavelets, SIAM Press, 1992
(10) Y.C. Pati and P.S. Krishnaprasad, "Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations", IEEE.Trans. on Neural Networks, Vol.4, No.1, pp.77-85, 1993
(11) J. Zhang, G.G. Walter, Y.B. Miao and W.N.W. Lee, "Wavelet neural networks for function learning", IEEE Trans. on Signal Processing, Vol.4, No.6, pp.1485-1497, 1995

(12) T. Yamakawa, E. Uchino and T. Samatsu, "Wavelet neural networks employing over-complete number of compactly supported non-orthogonal wavelets and their applications", Proc. the 1994 IEEE Int. Conf. Neural Networks, Orland, Florida, pp.1391-1396, 1994

(13) S. Kadambe and P. Srinivasan, "Applications of adaptive wavelets for speech", Optical Engineering, Vol.33, No.7, pp.2204-2211, 1994

(14) T. Poggio, F. Girosi, "Networks for approximation and learning", Proceedings on the IEEE, Vol.78, No.9, 1990

(15) R. Murray-Smith, "A local model network approach to nonlinear modeling", Ph.D. Thesis, University of Strathclyde, UK., 1994

(16) B. Foss, T.A. Johansen, "On local and fuzzy modeling", Proc. 3rd Int. Industrial Fuzzy Control and Intelligent Systems, Houston, Texas, 1993

(17) M. Fischer, O. Nelles and R. Isermann, "Adaptive predictive control of a heat exchanger based on a fuzzy model", Control Engineering Practice, Vol.6, pp.259-269, 1998

(18) T.A. Johansen and B.A. Foss, "Nonlinear local model representation for adaptive systems", Proceeding of the Int. Conf. on Intelligent Control and Instrumentation, Vol.22, pp.677-682, 1992

(19) T. Wang and Y. Sugai, "A wavelet neural network for the approximation of nonlinear multivariable function", The Trans. on the Institute of Electrical Engineers C, Vol.120-C, No.2, pp.185-193, 2000

(20) Y. Oussar, I. Rivals, L. Personnaz and G. Dreyfus, "Training wavelet networks for nonlinear dynamic input-output modeling", Neurocomputing, Vol.20, pp.173-188, 1998

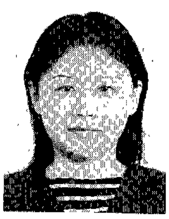
(21) C.H. Chong and J.Y. Choi, "Piecewise interpolation capabilities for function approximation", IEEE Tran. on Neural Networks, Vol.5, No.6, pp.936-944, 1994

(22) M.C. Mackey, L. Glass, "Oscillation and chaos in physiological control systems", Science 197, pp.287-289, 1997

(23) L.P. Maguire, B. Roche, T.M. McGinnity and L.J. McDaid, "Predicting a chaotic time series using a fuzzy neural network", Information Science, Vol.112, pp.125-136, 1998

(24) M. Henon, "A two-dimensional mapping with a strange attractor", Commun.Math.Phys, vol.50, pp.69-77, 1976

Ting Wang (Student Member) received the M.S. degree in electronic engineering from the Dalian University of Technology, P.R.Chian, in 1996. Presently, she is a doctoral student of the Graduate School of Science and Technology, Chiba University. Her research interests include neural networks and applications of wavelet transform theory.



Yasuo Sugai (Member) received the D.Eng. degree from the Tokyo Institute of Technology in 1985. He is an associate professor, department of urban environmental systems, faculty of engineering, Chiba university. His research areas are the neural networks, the emergent computation, the optimization engineering, and their applications. He is a member of IEICE, IPSJ and SICE.