Non uniform noisy data training using Wavelet neural network based on sampling theory

EHSAN HOSSAINI ASL, MEHDI SHAHBAZIAN, KARIM SALAHSHOOR
Department of Automation and Instrumentation
Petroleum University of technology
South khosro, Sattarkhan Street, Tehran, Iran
IRAN
e.hossaini.asl@gmail.com, shahbazian_m@yahoo.com , salahshoor@yahoo.com

Abstract: - Global convergence and overfitting are the main problem in neural network training. One of the new methods to overcome these problems is sampling theory that is applied in training of wavelet neural network. In this paper this new method is improved for training of wavelet neural network in non uniform and noisy data. The improvements include suggesting a method for finding the appropriate feedback matrix, addition of early stopping and wavelet thresholding to training procedure. Two experiments are conducted for one and two dimensional function. The results establish a satisfied performance of this algorithm in reduction of generalization error, reduction the complexity of wavelet neural network and mainly avoiding overfitting.

Key-Words: wavelet neural network; sampling theory; overfitting; early stopping; feedback matrix; wavelet thresholding; non uniform data

1 Introduction

Neural networks are proven to be a powerful tool for modeling nonlinear systems using numerical data [19,20]. The generalization capability is an ultimate criterion for measuring the validity of identified models. Overfitting is one of the most important problems in neural network learning. Complexity of model and noise contents in training data are two major sources of this problem.

Wavelet neural networks which use wavelets as basis function are found to have various interesting properties including fast training and good generalization performance [2]. Various methods have been proposed for structure selection and training of wavelet neural networks [1,2]. Recently, training wavelet neural network based on sampling theory has been proposed by Zhang [1]. This new algorithm is based on the limited band of wavelet networks, in which the input weights are determined by the sampling period or the frequency band of the target function (if available). This approach has been shown to have global convergence and avoid overfitting for non-noisy equi-spaced samples.

In many practical situations, a finite number of samples of the target function are known and there is not a priori information about the frequency contents and frequency band of the target function. Without using information about target signal (or noise), relying on sampling period for finding the input weights of the neural network may results in a complex structure and serious overfitting. To overcome this problem, a suitable model selection approach for complexity control and preventing overfitting should be used.

In case of noisy data, wavelet thresholding and early stopping are two helpful techniques for suppressing the overfitting by preventing the noise to be trained in the wavelet neural network. In wavelet thresholding technique, various methods are presented by donoho and silverman for denoising of uniform data [6,7] and non uniform data [8,9,10,11,21] in case of white and colored noise. In the wavelet thresholding techniques, the wavelet basis functions with coefficients smaller than specified thresholds will be eliminated because they essentially represent noise. This approach has been proven to be a powerful method in wavelet domain denoising. In the early stopping technique, which is a general approach in neural network modeling, the training data are divided into several sets for training of networks and validation of generalization capability. In this technique, before achieving a minimum training error, the training course is stopped at certain iteration. The stopping iteration is decided by cross validation.

Using only the sampling period for determination of the input weights of a wavelet neural network may results in a very large number of basis functions and overfitting. In this article, we have shown that using wavelet thresholding and or early stopping in conjunction with the sampling information of the given data will results in a less complex network with better noise removal.

This article is divided into four sections. Following this introduction, section 2 briefly reviews the theory of wavelet networks and its training based on the sampling theory. In section 3, new approaches developed for improvement of the wavelet neural network training based on sampling theory are presented. This section includes two parts. The first part explains a new method for constructing the appropriate feedback matrix and the second part explains developments in the training wavelet network. In section 4 simulation results for both one and two-dimensional target functions are presented.

2 Wavelet network and sampling theory

2.1 Review of wavelet neural network

In neural network learning, in order to take the full advantage of orthonormality of basis function, with localized learning, we need a set of basis functions which are local and orthogonal. Wavelets are new family of localized basis functions that have found many applications in large areas of science and engineering [2,3]. Wavelets are universal approximator which can be used to approximate any arbitrary multidimensional nonlinear function. They have many powerful mathematical properties such as orthonormality, locality in time and frequency domains, different degrees of smoothness, fast implementations, and effective compact support.

Wavelets are usually introduced in a multiresolution framework developed by Mallat [3]. We focus on the wavelet networks constructed from a multiresolution analysis (MRA) [3]. Consider a function $f(x)$ in $L^2(\mathbb{R})$, where $L^2(\mathbb{R})$ denotes the vector space of all measurable, square integrable one dimensional functions. In addition, assume V_j be the vector space containing all possible approximations of $f(x)$ at the resolution m. Then, the ladder of spaces $V_j\,, j\epsilon\mathbb{Z}$ represents the successive resolution levels for $f(x)$. The properties of these spaces are as follows:

1.
(Nested) $V_j \subseteq V_{j+1}\,, \forall j \in \mathbb{Z}$ (1)

2.
$f(x) \in V_j \;\Leftrightarrow f(x-k) \in V_j\,, \forall (j,k) \in \mathbb{Z}^2$ (2)

3.
(Density) $close_{L^2}\left(\bigcup_{j\in\mathbb{Z}} V_j\right) = L^2(\mathbb{R})$ (3)

4.
(Separation) $\bigcap_{j\in\mathbb{Z}} V_j = \{0\}$ (4)

5.
(Scaling) the function $f(x)$ belongs to V_j if and only if the function $f(2^{-j}x)$ belongs to V_0 (5)

6.
(Basis) There exists a function $\phi \in L^2$ (called a scaling function or a father wavelet), with $\phi_{j,k} = 2^{j/2}\phi(2^j x - k)$, such that $\{\phi_{0,k}; k \in \mathbb{Z}\}$ is a basis for V_0. (6)

The function ϕ is called a scaling function of the multiresolution analysis (MRA). A family of scaling functions of the MRA is expressed as:

$\phi_{j,k}(x) = 2^{j/2}\phi\left(2^j x - k\right),\; j,k \in \mathbb{Z}$ (7)

Where 2^j and k correspond to the dilation and translation factors of the scaling function respectively while $2^{j/2}$ is an energy normalisation factor.

Let W_j be the orthogonal complement of V_j to V_{j+1} ($V_j \oplus W_j = V_{j+1}$). Then the orthonormal basis functions corresponding to W_j's named wavelets and denoted by $\psi_{j,k}$'s can be easily obtained from $\phi_{j,k}$'s[3]. A family of wavelets may be represented as:

$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)\,,\; j,k \in \mathbb{Z}$ (8)

with 2^j,k and $2^{j/2}$ being the dilation, translation, and normalisation factor of the wavelets, respectively. Next $L^2(\mathbb{R})$ can be expressed as:

$L^2(\mathbb{R}) = \bigcup_{j\in\mathbb{Z}} V_j = \cdots W_{-1} \oplus W_0 \oplus W_1 \ldots = \oplus_{j\in\mathbb{Z}} W_j$ (9)

Where $W_j \perp W_m$ for $j \neq m$.

Fig.1 illustrates the relation between V_j and W_j spaces in MRA:

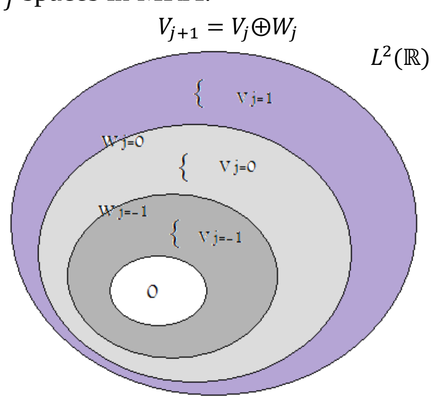

Fig.1 Embedded spaces V_j's for multiresolution representation of $L^2(\mathbb{R})$

Equation (9) indicates that the wavelet basis generates decomposition of the L^2 space. It shows that any L^2 function is uniformly approximated using a wavelet series:

$f(x) = \sum_{j=-\infty}^{j=+\infty} \sum_{k=-\infty}^{k=+\infty} d_{j,k}\psi_{j,k}(x)$ (10)

If we start from the approximation of the function at resolution j=0, then:

$f(x) = f_0(x) + \sum_{j=0}^{j=+\infty} \sum_{k=-\infty}^{k=+\infty} d_{j,k}\psi_{j,k}(x)$ (11)

Where

$f_0(x) = \sum_{k=-\infty}^{k=+\infty} a_{0,k}\phi_{0,k}(x)$ (12)

We can conclude that any function $f(x) \in L^2$ can be written as a unique linear combination of wavelets of different resolutions. This means that $(x) = \cdots + g_{-1}(x) + g_0(x) + g_1(x) + \cdots$, where $g_j(x) \in W_j$ is unique. Since $V_j = W_j + W_{j-1} + \cdots$ and spaces V_j can be generated by the scaling function $\phi(x) \in L^2$, there exists

$f_{ne}(x) = \sum_{k=-\infty}^{\infty} c_{j,k}\phi(2^j x - k) = \sum_{k=-\infty}^{\infty} c_{j,k}\phi_{j,k}$ (13)

Such that $\|f(x) - f_{ne}(x)\| \longrightarrow 0$ when j→∞. In fact formula (13) is just the presentation of wavelet networks with three layers. In an impact interval of interest, formula (13) can be written ass:

$f_{ne}(x) = \sum_{k=I_0}^{I_1} c_{j,k}\phi_{j,k}(x)$ (14)

Where $\phi_{j,k} = \phi(2^j x - k)$. A wavelet network is realized by taking $c_{j,k}$'s as the output weights, 2^j's as the input weights and $\phi(x-k)$ as the activation function.

Variety of approaches have been proposed for determining wavelet network parameters such as input weights 2^j and also output weights $c_{j,k}$'s. Here we use the approach based on sampling theory proposed by Zhang[1] for specifying appropriate resolution j.

2.2 Sampling theory

Since we use the sampling theory for the training of wavelet network, we briefly introduce some aspects of the sampling theory. For more discussions, we refer the reader to [17,18]. An analog signal can be simply discretized by recording its sample values $\{f(nT)\}_{n\in\mathbb{Z}}$ at interval T. An approximation of f(x) at any $x \in \mathbb{R}$ may be recovered by interpolating these samples.

If the samples x_i are taken in constant T period, then the target function is represented as:

$f(nT) = f(nT)\delta(x - nT)$ (15)

The Fourier transform of the discrete signal obtained by sampling f at intervals T is:

$\hat{f}_d(w) = \frac{1}{T}\sum_{k=-\infty}^{+\infty} \hat{f}(w - \frac{2k\pi}{T})$ (16)

If the support of $\hat{f}$ is included in $[-\pi/T\,,\pi/T]$ then

$f(x) = \sum_{k=-\infty}^{+\infty} f(kT)((x - kT)/T)$ (17)

On the other hand, the frequency band of wavelet network that described in previous section is obtained as follows:

$\int_{-\infty}^{\infty} \left|\tilde{f}_{ne}(w)\right|^2 dw \leq \int_{-2^j b_w}^{2^j b_w} \left|\tilde{f}_{ne}(w)\right|^2 dw + 2^{-j}\varepsilon \sum_k \left|c_{j,k}\right|^2$ (18)

So the energy of wavelet network is concentrated well in the following frequency band:

$[-2^j b_w, 2^j b_w]$ (19)

The parameter b_w only depends on scaling function. Formula (19) means that the frequency band of wavelet network can be controlled by input weights.

Suppose $\{kT, f_s(kT)\}$'s are training data with $\sum_k |f_s(kT)|^2 < +\infty$ then, by the sampling theorem, there exists a unique function $f(x)$ to interpolate all training data in the Paley–Wiener space P_T. On the other hand, a wavelet network is a function in $L^2(\mathbb{R})$ so a wavelet network represents a function in P_T if its Fourier transform has a support included in $[-\pi/T, \pi/T]$.

This means that the network $f_{ne}(x)$whose Fourier transform has a support in $[-\pi/T, \pi/T]$ is complex enough to recover a band-limited function. Since the regularity of function is related to the asymptotic decay of Fourier transform, a band-limited function is always ‘‘smoother’’ than the other functions. This means that the training results that are obtained when the frequency band of wavelet network is limited in the interval $[-\pi/T, \pi/T]$ are always more regular than the results that are obtained when the band is limited in the other intervals, so we limit the frequency band of wavelet network to $[-\pi/T, \pi/T]$ in our new algorithm.

According to (18), the input weights can be calculated using following formula:

$$2^j = \pi/(b_w \times T) \tag{20}$$

This formula is the consequence of a distinguishing property of wavelet function that its energy is well localized in frequency domain. For constructing the structure of wavelet network, the property of energy concentration of wavelet in time domain should be employed. In wavelet network, the k^{th} node has the following input-output function:

$$S_{out} = \phi(2^j S_{in} - k) \tag{21}$$

Where S_{in} is the input, 2^j's are the input weights, k is the k^{th} threshold and $\phi(\circ)$ is the scaling function. If the support of scaling function is limited to $[0, N_\phi]$, then the k^{th} node of network has the following support:

$$[2^{-j}k, 2^{-j}(N_\phi + k)] \tag{22}$$

Assume the domain of interest for estimation of function is the interval [a,b], then the translations are found as follows:

$$2^j a - N_\phi \le k \le 2^j b \tag{23}$$

Many methods have been proposed for training output weights of wavelet neural network based on minimizing error function:

$$J(f_s, f_{ne}) = \sum_{i=1}^{N} |f_s(x_i) - f_{ne}(x_i)|^2 \tag{24}$$

Where$\{x_i, f_s(x_i)\}$'s are samples and$f_{ne}(x_i)$'s are output of approximator. Without any additive term, this cost function is widely used in the training of networks because of convenient implementation.

Three commonly used methods are direct solution method, iterative method and inner product method. In this paper, the iterative method is employed for training the output weights. In this method the output weights can be calculated as follows:

$$E^{(k+1)} = F_s - \phi_{m\times n} C^{(k+1)} \tag{25}$$

$$C^{(k+1)} = C^{(k)} + A_f E^{(k)} \tag{26}$$

The column vector $E^{(k)}$ denotes the error of interpolation by the wavelet network at k^{th} iteration, the column vector $C^{(k)}$ represents the output weights at k^{th} iteration and the matrix A_f is the feedback matrix. The values of elements in the feedback matrix indicate that how much the errors in each data point would affect on output weights. The $\phi_{m\times n}$ matrix is

$$\phi_{m\times n} = \begin{bmatrix} \phi_{J,I_0}(x_1) & \cdots & \phi_{J,k}(x_1) & \cdots & \phi_{J,I_1}(x_m) \\ \vdots & \cdots & & \cdots & \vdots \\ \phi_{J,I_0}(x_m) & \cdots & \phi_{J,k}(x_m) & \cdots & \phi_{J,I_1}(x_m) \end{bmatrix} \tag{27}$$

Where the subscript $I_0 = [2^J a - N_\phi]$ and the subscript $I_1 = [2^J b]$, which denote respectively, the minimum and maximum of translation k obtained from (23).

The structure of wavelet network with two sub wavelet network is depicted in Fig.2:

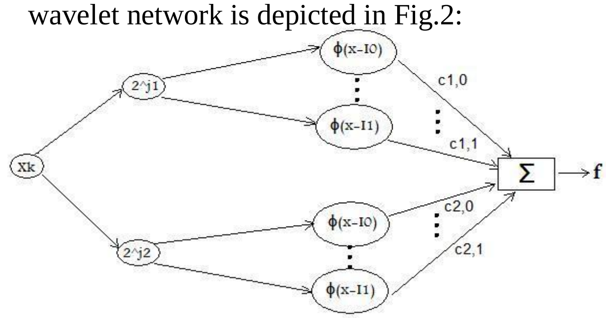

Fig.2 structure of wavelet neural network with two sub network

Zhang [1] has shown that we always can find a feedback matrix that causes the iterative course to converge to fix point. However there is not a unique method for constructing the feedback matrix. In the next section, an intuitive method for finding this matrix is proposed.

3 Modifications in the training of wavelet network

We propose three modifications for improvement in training of wavelet network that will be discussed in the following sections.

3.1 Determination of appropriate feedback matrix

In this part, an intuitive approach for finding the appropriate feedback matrix is proposed. In this method the feedback matrix is constructed based on the ϕ matrix. This method uses the receptive field of each node or scaling function in wavelet network. It means that each output weight is trained only using the training data that lie in the receptive field of its node. However, all of these training data should not have the same effect in training of output weight. It means that the training data that the scaling function has a larger value in its location have a stronger effect in training of output weight.

On the other hand, the elements of feedback matrix represent the effect of each training data in training of output weight in each node. Since the values of ϕ matrix represents the amplitude of scaling function in each data point, the feedback matrix should be constructed based on ϕ matrix. We use the forth order cardinal scaling function as activation function that shown in Fig.3 as follows:

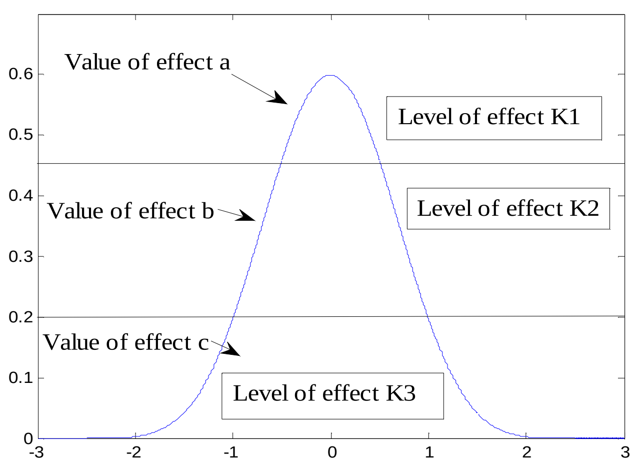

Fig.3 Fourth order cardinal scaling function at scale=1

According to Fig.3, the domain that the scaling function has larger amplitude is in the vicinity of zero. In other words, the training data that lie in this vicinity have intensive effect on training of the correspondent output weight. Therefore we can define different *levels of effect* for training data in each node. Therefore, the procedure for finding the appropriate feedback matrix is stated as follow:

1. Generate raw feedback matrix: $A_f = \phi^T$
2. Define *levels of effect* by partitioning the amplitude of scaling function by levels $K1$ & $K2$ & ... as depicted in Fig.1 (about 2 or 3 levels is appropriate.)
3. Assign the values of a, b and c that represent the *values of effect* for each level. These values can be calculated by trial and error.

Performance of the algorithm is indicated by training a sinusoidal function. In this example, different groups of *levels of effect* and *values of effects* are applied in training course. The convergence of four training courses is compared together. The results are shown below:

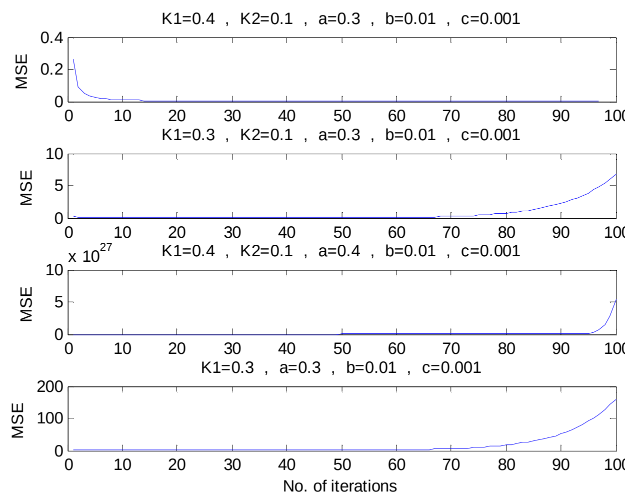

Fig.4 Comparison of MSE for different feedback matrix

In Fig.4, mean square error (MSE) versus Number of iteration is depicted. In comparison between 4 figures, it's concluded that by appropriate assignment of *levels of effect* and *values of effect*, the MSE converges to zero by increasing of iteration.

3.2 Improvement in algorithm for training non uniform data

For uniform sampled data, training wavelet neural network based on sampling theory shows quite acceptable results. However, for non uniform data, the algorithm encounters severe problems such as high overfitting error and deviation of estimated function from the actual target function. Here we use some available techniques to overcome this problem. These techniques are described in detail in the following sections.

3.2.1 Optimum number for sub wavelet networks

For training of non uniform data, the domain of interest for estimation is divided in some clusters that the sampling rate is approximately uniform in each cluster. The formula that is proposed by Zhang in [1] is described as follows:

$$D = \sup_{k \in \mathbb{Z}} |x_k - kT| < 0.25 \times T \qquad (28)$$

In this formula, x_k denotes the training data at k^{th} point, T denotes the approximate sampling rate and D denotes the maximum distance between training point in cluster with data in uniform sampling rate. In implementation of this method in simulation, it will be proved that because of generating multitude sub wavelet network, noise contents will be trained in estimation of wavelet network. This event causes overfitting error in estimation. In simulation, it is proved that the best criterion for generating the appropriate clusters for domain of interest is as follow:

$$D = \sup_{k \in \mathbb{Z}} |x_k - kT| < 0.99 \times T \qquad (29)$$

Using formula (29), the number of sub wavelet network is optimized and results in less noise content in estimation of wavelet neural network.

The performance of the new proposed formula is shown in training of exponential function in Fig.5.

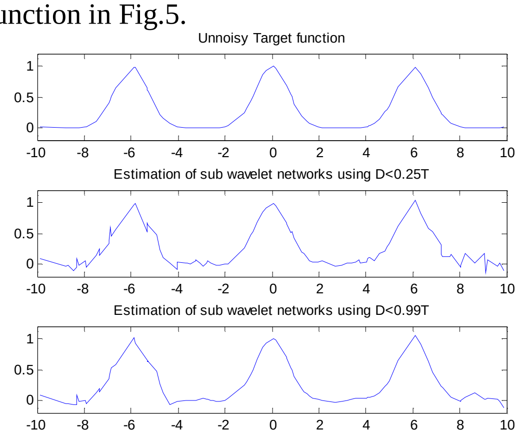

Fig.5 Comparison of estimations for 2 methods

3.2.2 Application of wavelet thresholding

The wavelet thresholding is an effective way for removing noise content from training data [6,7,8].

Hard and soft thresholding could be employed for this purpose. Hard thresholding can be described as the usual process of setting to zero the output weights whose absolute values are lower than the threshold. Soft thresholding is an extension of hard thresholding, first setting to zero the output weights whose absolute values are lower than the threshold, and then shrinking the nonzero weights towards 0. Fig.6 depicts ramp signal that is thresholded in the amplitude of 0.4.

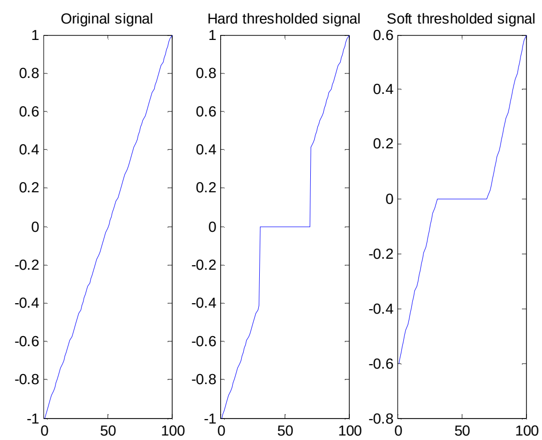

Fig.6 Comparison between hard and soft thresholding

There are some approaches for determining the value of threshold. The stein's unbiased risk estimate (sure), fixed form threshold, the mixture of two methods and minimax estimation principle, are four methods that is presented in the literature [6,7,8,9,10,11].

In addition, there are different types of noise such as white noise, unscaled white noise and non white noise that can be treated by different types of wavelet thresholding. These methods of wavelet thresholding could be added to the training procedure of wavelet network for reducing the effect of overfitting. In simulation (section 4), it is proved that by defining a threshold, the estimation performance is improved.

3.2.3 Apply early stopping

Early stopping technique is widely used in neural network training for reducing the effect of noise and overfitting. This technique causes the training course to stops when the test error begins to increase. In simulation (section 4), it is shown that this technique intensively affect on reducing the overfitting error.

4 Simulation

In this part, we indicate the performance of the proposed modifications described previously. This section includes two parts. In the first part, the one dimensional function is employed and in the second part the two dimensional function is used.

4.1 Learning one dimensional function

4.1.1 The target function

For comparison purposes, we use the function that is used in [1]. The target function is:

$$f_s(t) = \sum_{k=-3}^{3} e^{-(t-6k)^2} \qquad (30)$$

And the interval [-10, 10] is taken as domain of interest. The shape of target function is shown in Fig.7 as below:

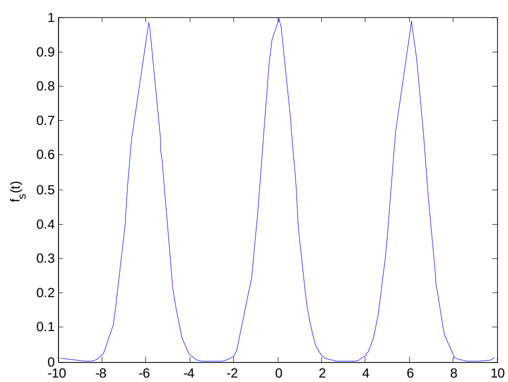

Fig.7 Target function

For producing non uniform data the random data with uniform distribution is employed. The noise function $n(t)$ of Gaussian distribution is added to training data. The noise has zero mean and variable variance with time as follows:

$$\sigma^2(t) = (0.005 + 0.009|t|)^2 \qquad (31)$$

The noise function is depicted in Fig.8 as shown below:

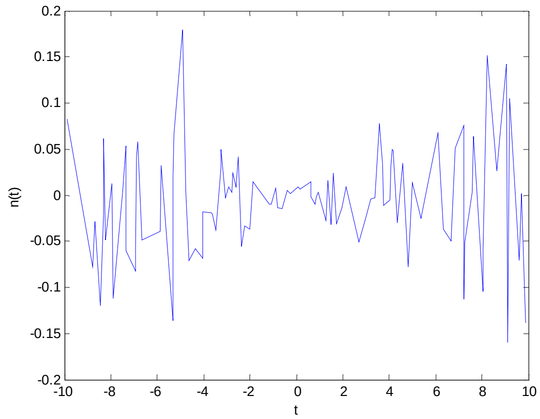

Fig.8 The noise function

Now the training data are generated. For generating the test data, we use the same method.

In each iteration, the wavelet network is tested with test data and if the mean square error begins to increase, then the training course stops. The objective of this simulation is to compare the performance of method in [1] with the improved one that described in previous sections. First we should find the appropriate feedback matrix for training.

4.1.2 Results of training

After some iteration, the values that are achieved for *levels of effect* and *values of effect* values are as follows:

$$k_1 = 0.4 \, , k_2 = 0.1$$
$$a = 0.3 \, , b = 0.01 \, , c = 0 \qquad (32)$$

In simulation, the method that is described in [1] is called method I and the improved one that is described in this paper is called method II. This method contains decreasing of sub wavelet networks, applying wavelet thresholding and addition of early stopping to training procedure. The structures of wavelet network that are generated of two methods are shown in Table 1.

Table 1
Comparison of two wavelet networks

	No. of sub wavelet networks	No. of nodes
Method I	81	413
Method II	7	72

Table 2
Statistic errors of two methods

	Maximum absolute error	Mean absolute error	Root mean square error
Method I	0.20019	0.043885	0.058885
Method II	0.11919	0.026758	0.036794

According to Table 1, the wavelet network of method II is so smaller in number of sub wavelet network and also nodes. The training course of method I is stopped after 200 iterations, but in method II the training course stopped after 8 iterations. The value for wavelet thresholding in method II is chosen0.009. The estimation results are depicted in the Fig.9 and Fig.10. This result indicates that in addition of reducing the sub wavelet networks, the iterations of training course and also the overfitting error are significantly reduced. By comparison of two figures, it's clear that in addition to noise reduction, the estimated function in method II is smoother than the one in method I.

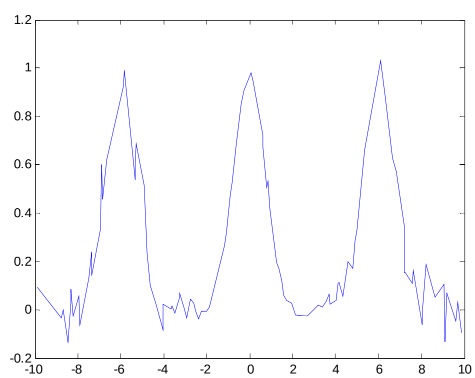

Fig.9 Estimation of wavelet network with method I

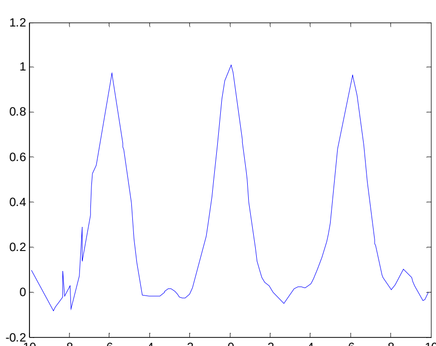

Fig.10 Estimation error of wavelet network with method II

4.2 Learning two dimensional function

4.2.1 The target function

The target function that is used for this course of training is in domain $[-10,10] \times [-10,10]$ and is as follows:

$$f_s = e^{-0.25(x^2+y^2)} + e^{-0.25(\sqrt{x^2+y^2}-10)^2} + e^{-0.25(\sqrt{x^2+y^2}-20)^2} \quad (33)$$

The noise function is of Gaussian distribution with zero mean and variable variance as follows:

$$\sigma^2 = 10^{-4} + 10^{-3}\rho^2 \quad (34)$$

The shapes of target function and noise function are shown in Fig.11 and Fig.12 as below:

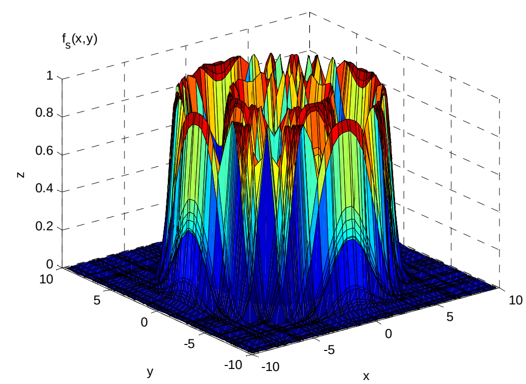

Fig.11 The target function

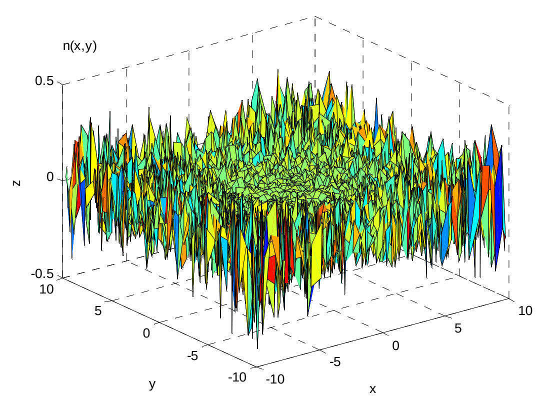

Fig.12 The noise function

Table 4
Statistic errors of two methods

	Maximum absolute error	Mean absolute error	Root mean square error
Method I	0.45125	0.060125	0.080736
Method II	0.33816	0.045253	0.062437

4.2.2 Results of training

After trial and error, the *levels of effect* and *values of effect* of feedback matrix are calculated as follows:

$k_1 = 0.35$

$a = 0.22\,, b = 0$ (35)

Table 3 compares the structure of wavelet networks that are generated by these two methods.

Table 3
Comparison of two wavelet networks

	No. of sub wavelet networks	No. of nodes
Method I	81	170569
Method II	10	7114

According to Table 3, it's clear that the wavelet network that is generated by method II is much smaller than the first one. This leads to less mathematic computation and compensating in time of training course.

The Fig.13 to Fig.16 depicts the results of two methods. As it is clear in these figures, the results of method II contain less noise and so its performance in rejecting noise is better than the method I in [1].

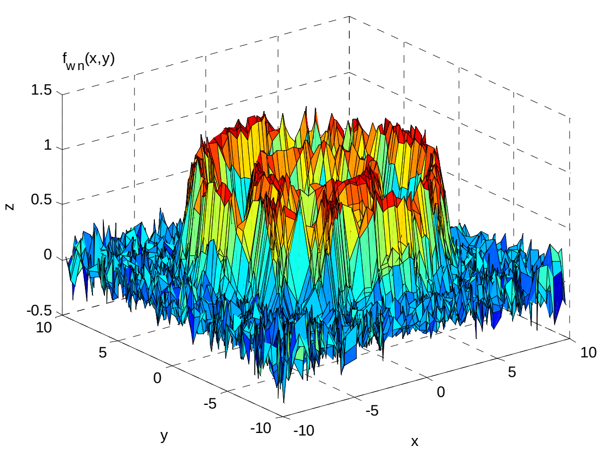

Fig.13 Estimation of wavelet network with method I

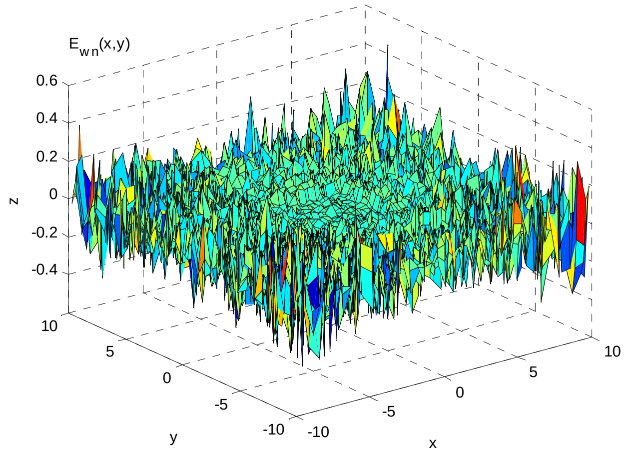

Fig.14 Error surface of method I

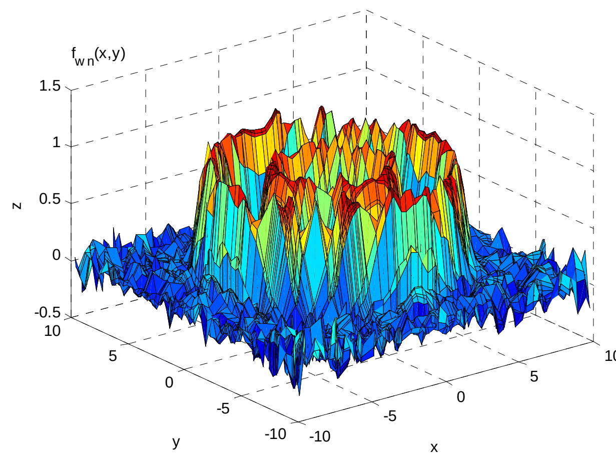

Fig.15 Estimation of wavelet network with method II

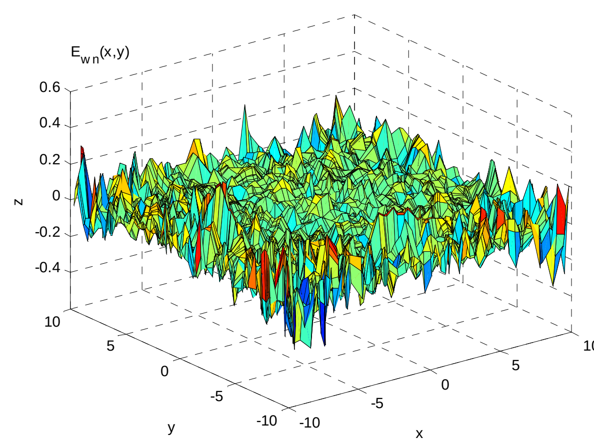

Fig.16 Error surface of method II

The results of method I are achieved after 100 iterations but in method II, the results are gained after 15 iterations. The wavelet threshold value that is chosen in this course of training is 0.009. Table 4 also shows that the maximum absolute error, mean absolute error and root mean square error are decreased by reducing the number of sub wavelet networks and also the number of nodes and also applying early stopping technique. Therefore method II avoid wavelet neural network from over training the noise contents of target function in training course.

5. Conclusion

Training wavelet neural network based on sampling theory has been shown to have good performance for uniform data. For non uniform noisy data, however, it encounters severe problems such as overfitting and large number of sub wavelet networks and long training course. This paper has proposed modifications to this approach to improve its performance. We proposed a new method for determining the feedback matrix, reduction the number of sub wavelet networks and employing some useful techniques for complexity control like wavelet thresholding and early stopping to overcome overfitting. The simulation results proved that applying this modification results in smaller wavelet network with faster training and less overfitting. Actually the main problem in wavelet networks based on sampling theory is that in practice, no information about noise is employed in determination of input weights. Therefore we recommend new researches on including the information of target function and/or noise in determining the input and output weights of wavelet network.

References:

[1] Zhigou zhang, Learning algorithm of wavelet network based on sampling theory, *Neuro computing*, January 2007.

[2] Daubechies I., Orthonormal Bases of Compactly Supported Wavelets, *Com. Pure Appl. Math.*, Vol.XLI, 1988, pp.909-996.

[3] S.G. Mallat, A theory for multi-resolution signal decomposition: the wavelet representation, *IEEE Trans. Pattern Analysis Mach. Int.*, Vol.11, No.7, 1989, pp.674-693.

[4] B.R. Bakshi, G. Stephanopoulos, Wave-net: a multiresolution hierarchical neural network with localized learning, *AIChE Journal*,Vol.39, No.1, 1993, pp.57-81.

[5] A.A. Safavi, *Wavelet neural networks and multiresolution analysis with applications to process systems engineering*, PhD thesis, Department of Chemical Engineering, The University of Sydney, Australia, January 1996.

[6] D.L.Donoho, De-noising via soft-thresholding, *IEEE Trans. Inform.*, Theory 41, 1995, pp.613-627.

[7] D.L.Donoho, I.M.Johnstone, Ideal spatial adaptation by wavelet shrinkage, *Biometrika 81* , 1994, pp.425-455.

[8] D.L. Donoho, I.M.Johnstone, Adapting to unknown smoothness via wavelet shrinking, *J. Amer. Statist. Assoc.*, Vol.90, 1995, pp.1200-1224.

[9] D.L. Donoho, I.M.Johnstone, Minimax estimation via wavelet shrinkage, *Ann. Statist.*, Vol.26, 1998, pp.879-921.

[10] A.Kovac, B.W.Silverman, Extending the scope of wavelet regression methods by coefficient-dependent thresholding, *J. Amer. Statist. Assoc.*, Vol.95, 2000, pp.172-183.

[11] M. Jansen, G.Nason, B. Silverman, Scattered data smoothing by empirical Bayesian shrinkage of second generation wavelet coefficients, *IX Proceedings of the SPIE*, vol.4478, 2001, pp.87-97.

[12] I.M.Johnstone, B.W. Silverman, Needles and straw in haystacks: empirical Bayes estimates of possibly sparse sequences, *Ann. Statist.*, Vol.32, No.4, 2004, pp.1594-1649.

[13] I.M.Johnstone, B.W. Silverman, Empirical Bayes selection of wavelet thresholds, *Ann. Statist.*, Vol.33, 2005.

[14] H.Q.Thuan, S. Rudy, Effective neural network pruning using crossvalidation, *in: Proceedings of the International Joint Conference on Neural Networks*, 2005, pp. 972–977.

[15] Van Der, N.T.Merwe, A.J. Hoffman, Developing an efficient cross validation strategy to determine classifier performance (CVCP), *Proc. Int. Jt. Conf. Neural Networks*, Vol.3 2001, pp.1663–1668.

[16] C.E.Vasios, G.K.Matsopoulos, E.M. Ventouras, K.S. Nikita, N.Uzunoglu, Cross-validation and neural network architecture selection for the classification of intracranial current sources, *Seventh Seminar on Neural Network Applications in Electrical Engineering, Proceedings NEUREL*, 2004, pp. 151–158.

[17] M. Farokh, *Nonuniform Sampling Theory and Practice*, Kluwer Academic/Plenum Publishers, Dordrecht/New York, 2001.

[18] Z. Jun, G.G. Walter, Y. Miao, Wavelet neural networks for function learning, *IEEE Trans. Signal Process*., Vol.43, No.6, 1995, pp.1485–1496.

[19] Z. Zainuddin, O. Pauline, Function approximation using artificial neural networks, WSEAS Trans. On Mathematics, 2008.

[20] P. Radonja, S. Stankovic, D. Drazic, B. Matovic, Development and Construction of Generalized Process Models by using Neural Networks and Multiple Linear Regression, WSEAS Trans. On systems, Vol.6, No.287, 2007.

[21] V. Niola, R. Oliviero, G. Quaremba, A neural network application for signal denoising, WSEAS Trans. on signal processing, Vol.2, No.88, 2006.