Neural Network Modelling With Discrete Wavelet Transform Pre-Processing for Financial Time Series Data[1]

Budi Warsito[#2], Suparti[#3], Subanar[*4]

[1]*Supported by Hibah PEKERTI Dikti 2010*

[#]*Study Program of Statistics, Department of Mathematics, DiponegoroUniversity*

Kampus Undip Tembalang, Semarang, Indonesia

[2]budi_wrst@yahoo.com

[3]suparti.undip@ac.id

[*] *Study Program of Statistics, Department of Mathematics, Gadjah Mada University*

FMIPA, Sekip Utara Yogyakarta, Indonesia

[4]subanar@yahoo.com

***Abstract*— This paper discuss about Feed Forward Neural Network (FFNN) modelling by using the discrete wavelet transform (DWT) as pre-processing to the input and target. Before the training to the FFNN be done, the wavelet decomposition is performed from the input and target with the DWT at a level decomposition and result the approximation coefficient. After training, the reconstruction process as a post-processing will returns the output that be resulted from the FFNN to the term at first. We call the process as an inverse discrete wavelet transform (IDWT). The next step is do the predict in-sample and also predict out of sample from FFNN with Haar discrete wavelet transform at certain level decomposition. The FFNN training method that be used is Levenberg-Marquardt with the logistic sigmoid as activation function and network architectur is determined before. Then the model is applied to the financial time series data.**

Index Term— **FFNN, pre-processing, DWT, financial**

I. Introduction

Neural Network modelling or especially Feed Forward Neural Network (FFNN) model has made a rapid grow. FFNN Neural networks are developed to emulate the human brain that is powerful, flexible and efficient. In the subsequent development the FFNN model also have been applied at the various field, for instance in classification, clustering and predict the time series data. Many research have be done to apply the FFNN model to predict the time series nonlinear (e.g. [1], [2]). However, conventional neural networks process signals only on their finest resolutions [3]. The introduction of wavelet decomposition as in [4], provides a new tool for approximation. It produces a good local representation of the signal in both the time and the frequency domains.

Wavelet is a basis function that be used in representing data or other functions. The wavelet algorithm process a data at the certain scale and the different resolution. The construction of orthonormal system proposed at first by Haar and it was a foundation of modern wavelet theory. Wavelet functions have a different value from zero in a short time interval relatively. At this condition wavelet is different with neither the normal function nor wave function like sinusoida, where all of them are established in a time domain $(-\infty,\infty)$. There are some basic wavelet functions that are Mexican hat, Gauss Wavelet, Morlet Wavelet and also Daubechies Wavelet family. The result of wavelet transformation will give a set of wavelet coefficients that be obtained from point observation (location) at different level (scale) and range width [5]. There are some way that can be done, one of them is Discrete Wavelet Transform (DWT) as be suggested as in [4] that proposed a fast algorithm to determine wavelet coefficients. The superiority of the using the DWT is its multi resolution representation. DWT can give a good local representation in both time and frequency domain.

The reprecentation of a function with wavelet will more efficient because it localized in time domain (the mean is that the wavelet function will give a null value when the domain is relatively huge). It is caused the number of coefficients of wavelet that have non zero value in reconstruction of function with wavelet is relativelly few. This matter is strengthened with the rate of convergention, that is integrated mean square error (IMSE) from wavelet estimator that form one of goodness measure from an estimator. The optimal of IMSE of wavelet estimator from smooth function will fast to zero [6]. Beside that, wavelet also can to representative the functions that have unsmooth and function with high volatility because the basis in wavelet is established by position and scale (translation dan dilatation). At a part of unsmooth function, wavelet representation will use narrow support and at a part of smooth function will use more wide support. At the other hand, wavelet functions have locally adaptive support.

Inspired by both the FFNN and wavelet decomposition, many research then proposed wavelet network. This has led to rapid development of neural network models integrated with wavelets. Most researchers used wavelets as basis functions that allow for hierarchical, multi resolution learning of input-output maps from data. The new model that be constructed is an FFNN with the wavelet packet as a feature extraction method to obtain time-frequency information interrelated with input [7]. Some research applied this model in various fields, i.e. to estimate a function [8], to predict the exchange rate integrated with genetic algorithm [9], to analyze the nonlinear time series [10,11] and to synthesis the Wavelet Filters [12].

This paper discuss FFNN modelling with discrete wavelet transform as pre-processing, especially by using Haar wavelet at various level and be applied in time series financial data.

II. Neural Network

Neural Network model that be observed specifically in this resesarch is Feed Forward Neural Network (FFNN) model. FFNN architecture that be used to predict time series data with configuration input unit X_1 until X_p and one constant unit (bias), a hidden layer that contain n neurons and 1 output unit is shown in fig. 1.

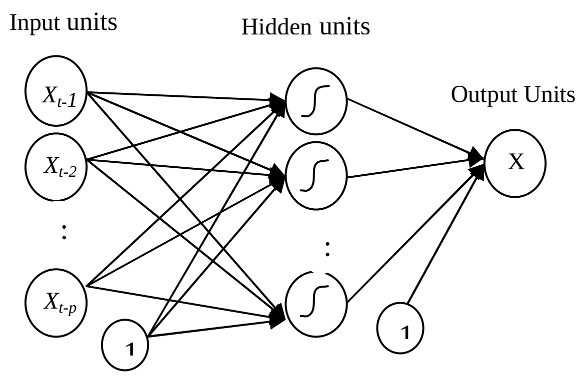

Fig. 1 FFNN architecture to predict time series data with one hidden layer that contain n neuron and input variables are the values at p lag time $X_{t-1}, X_{t-2}, \ldots, X_{t-p}$

FFNN model with one hidden layer and input $X_{t-1}, X_{t-2}, \ldots, X_{t-p}$ can be explained as :

$$\hat{X}_t = \psi_0 \left\{ w_{co} + \sum_n w_{no} \psi_n \left(w_{cn} + \sum_i w_{in} X_{t-j_i} \right) \right\} \quad (1)$$

where $\{w_{cn}\}$ is weight between constant unit and neuron, and w_{co} is weight between constant unit and output. The weights $\{w_{in}\}$ and $\{w_{no}\}$ are the connection weights between input with neuron and between neuron with output, respectively. The functions ψ_n and ψ_o are the activation functions that be used at neuron and output, respectively. Notation that be used to FFNN model is NN($j_1, \ldots, j_k$, n) that explain FFNN with input variables at lag $j_1, \ldots, j_k$ and n neuron in one hidden layer.

Training algorithm that be used in FFNN is backpropagation that involves three stages: the feedforward of the input training pattern, the backpropagation of the associated error, and the adjusment of the weights. During feedforward each input unit (x_i) receives an input signal and broadcasts this signal to the each of the hidden units $z_1, \ldots, z_p$. Each hidden unit then computes its activation and weighted summation of the inputs in the form :

$$z\_in_j = \sum_i w_{ji} x_i + w_{bj} \quad (2)$$

where x_i is activation from input unit i that sends its signal to hidden unit j and w_j is weights of the sent signal and j = 1,2, …, q is the number of hidden unit. w_{bj} is the weight from bias to hidden unit j. The results of the summation then be transformed with nonlinear activation function f(·) to get activation z_j of unit j in the form :

$$z_j = f\left(z\_in_j\right) \quad (3)$$

After all hidden unit computes its activation then send its signal (z_j) to output unit. Output unit then computes its activation to give respon of the network to the input pattern that be sent in the form :

$$g(w, z) = \sum_j w_{jo} z_j + w_{bo} \quad (4)$$

The function at (4) is the output of the network :

$$y = \sum_j w_{jo} f(a_j) + w_{bo} \quad (5)$$

where w_{bo} is the weight from bias to the output unit. During training the network, output unit compares its computed activation y with its target t to determine the associated error for that pattern with that unit [13].

III. Discrete Wavelet Transform

Wavelet function is a mathematics function that have certain characteristics, among them oscillation in surrounding zero (like sinus and cosinus function) and be localized in time domain that the mean is at the value of the domain is heavy relatively, the value of wavelet function is zero. Wavelet function is divided into two type, father wavelet (ϕ) and mother wavelet (ψ) that have characteristics :

$$\int_{-\infty}^{\infty} \phi(x)dx = 1 \text{ and } \int_{-\infty}^{\infty} \psi(x)dx = 0 .$$

With the diadic dilatation and the integer translation, father wavelet and mother wavelet bear the family of wavelet, i.e. $\phi_{j,k}(x) = (p2^j)^{1/2} \phi(p2^j x - k)$ and $\psi_{j,k}(x) = (p2^j)^{1/2} \psi(p2^j x - k)$ for a scalar p>0, and without reduce the generalization, it can be taken p=1, so that $\phi_{j,k}(x) = 2^{j/2} \phi(2^j x - k)$ and $\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k)$. The function $\phi_{j,k}(x)$ and $\psi_{j,k}(x)$ have characteristics :

$$\int_{-\infty}^{\infty} \phi_{j,k}(x) \phi_{j,k'}(x) dx = \delta_{k,k'}$$

$$\int_{-\infty}^{\infty} \psi_{j,k}(x) \phi_{j,k'}(x) dx = 0$$

$$\int_{-\infty}^{\infty} \psi_{j,k}(x) \psi_{j',k'}(x) dx = \delta_{j,j'} \delta_{k,k'}$$

where $\delta_{i,j} = \begin{cases} 1 & if \ i = j \\ 0 & if \ i \neq j. \end{cases}$

The most simple examples of Haar wavelet are :

$$\psi(x) = \begin{cases} 1, & 0 \le x < 1/2 \\ -1, & 1/2 \le x < 1 \\ 0, & others \end{cases} \quad (6)$$

and

$$\phi(x) = \begin{cases} 1, & 0 \le x < 1 \\ 0, & others \end{cases} \quad (7)$$

Beside Haar, another examples of wavelet are Daubechies (Daublet), symmetris (Symmlet), and Coifman (Coiflet) wavelet. Visualisation of those are shown in fig. 2 :

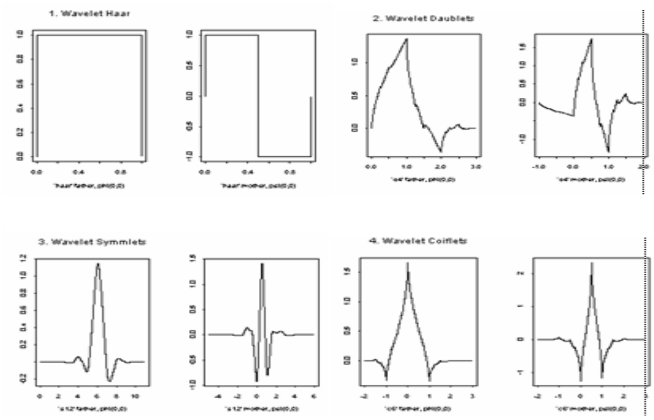

Fig. 2 Visualisation of some Wavelets

Multi resolution analysis $L^2(R)$ is closed state space $\{V_j, j \in Z\}$ that follow :

i) $\ldots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \ldots$

ii) $\bigcap_{j \in Z} V_j = \{0\}$, $\overline{\bigcup_{j \in Z} V_j} = L^2(R)$

iii) $f \in V_j \Leftrightarrow f(2.) \in V_{j+1}$

iv) $f \in V_0 \Rightarrow f(.-k) \in V_0, \forall k \in Z$

v) There is a function $\phi \in V_0$ so that $\phi_{0,k} = \phi(.-k), k \in Z$ form orthonormal basis to V_0 for all j,k ∈ Z, $\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k)$.

If $\{V_j, j \in Z\}$ is multi resolution analysis from $L^2(R)$, then there is orthonormal basis $\{\psi_{j,k}; j,k \in Z\}$ for $L^2(R)$: $\psi_{j,k} = 2^{j/2}\psi(2^j x - k)$, so that for any arbitrary function f ∈ $L^2(R)$,

$$P^j f = P^{j-1} f + \sum_{k \in Z} < f, \psi_{j-1,k} > \psi_{j-1,k}.$$

where $\psi(x)$ is generated from

$$\psi(x) = \sum_{k \in Z} (-1)^k c_{(-k+1)} \phi_{1,k}(x).$$

Consequence. If ϕ is a scale function that form multiresolution analysis and

$$\psi(x) = \sum_{k \in Z} (-1)^k c_{(-k+1)} \phi_{1,k}(x)$$

then decomposition into orthonormal wavelet for arbitrary function $f \in L^2(R)$ can be explained as :

$$f(x) = \sum_{k \in Z} c_{jo,k}\phi_{jo,k}(x) + \sum_{j \ge jo}\sum_{k \in Z} d_{j,k}\psi_{j,k}(x) \quad (8)$$

where $c_{jo,k} =< f, \phi_{jo,k} >$ and $d_{j,k} =< f, \psi_{j,k} >$.

Linier Wavelet Estimator

There is a collect of independent data $\{(Xi, Yi)\}_{i=1}^n$ and $n = 2^m$, where m is a positif number. If X_i is design of reguler point at interval [0,1] where $X_i = i/n$, then projection f at space V_J can be explained as :

$$(P^J f)(x) = \sum_{k \in Z} c_{J,k}\phi_{J,k}(x) \text{ or}$$

$$f_J(x) = \sum_{k \in Z} c_{J,k}\phi_{J,k}(x)$$

where $c_{J,k} = \langle f, \phi_{J,k} \rangle = \int_0^1 f(x)\phi_{J,k}(x)dx$. Based on decomposition of function into orthonormal wavelet (4) to any arbitrary function $f \in L^2(R)$ we get

$$f_J(x) = \sum_{k \in Z} c_{jo,k}\phi_{jo,k}(x) + \sum_{j \ge jo}^{J-1}\sum_{k \in Z} d_{j,k}\psi_{j,k}(x)$$

where

$$c_{jo,k} =< f, \phi_{jo,k} > = \int_0^1 f(x)\phi_{jo,k}(x)dx \text{ and}$$

$$d_{j,k} =< f, \psi_{j,k} > = \int_0^1 f(x)\psi_{j,k}(x)dx.$$

Because the function of regression f is unknown then estimator f at space V_J can be explained as :

$$\hat{f}_J(x) = \sum_{k \in Z} \hat{c}_{J,k}\phi_{J,k}(x)$$

where $\hat{c}_{J,k} = \frac{1}{n}\sum_{i=1}^n Y_i\phi_{J,k}(X_i)$, or

$$\hat{f}_J(x) = \sum_{k \in Z} \hat{c}_{jo,k}\phi_{jo,k}(x) + \sum_{j \ge jo}^{J-1}\sum_{k \in Z} \hat{d}_{j,k}\psi_{j,k}(x) \quad (9)$$

where $\hat{c}_{jo,k} = \frac{1}{n}\sum_{i=1}^n Y_i\phi_{jo,k}(X_i)$

$$\hat{d}_{j,k} = \frac{1}{n}\sum_{i=1}^n Y_i\psi_{j,k}(X_i),$$

that is an unbiased estimator from $c_{jo,k}$ and $d_{j,k}$. Wavelet estimator at (9) is called linear wavelet estimator.

IV. Pre-Processing Neural Network with DWT

Construction of model that be used at this research is shown in fig. 3. At the new architecture, the three stages that have to be done are pre-processing to the input and target, training the network and then post-processing. The wavelet that be used is Haar with a certain level. These procedure can be explained below :

1) Decomposition (Pre-Processing):
   Choose a wavelet and a level of decomposition N, and then compute the wavelet decompositions of the signals at level N
2) Training Network :
   Do the training process to the network with training algorithm that be chosen before
3) Reconstruction (Post-Processing):
   Compute wavelet reconstructions using the original approximation coefficients of level N

The new model is expected can give a better prediction result.

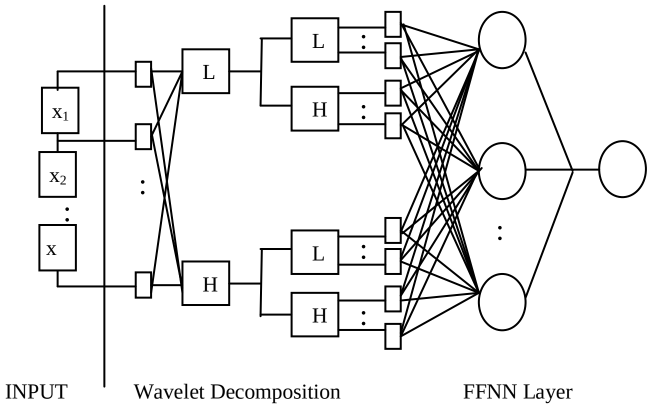

Fig. 3 Construction of FFNN model with wavelet as pre-processing

V. Simulation Result

We have chosen a financial time series often found in the literature as a benchmark, that is IHSG at Bursa Efek Jakarta that contain 833 series, from January 2^{nd} at 2007 until May 12^{th} at 2010. The data set is divided into two sections for training and testing. The FFNN architecture that be used is contain one lag time and the bias as input, the number of hidden unit is five with logistic sigmoid as activation function and Levenberg-Marquardt as the algorithm to update the weights. The maximum epoch that be determined is 1000. The wavelet discrete that be chosen is Haar at some level. When the maximum epoch or minimum error is reach, training stops. In order to have a fair comparison, simulation is carried out for each network over some trials (weight initialization can be different from trial to trial, depend on the random initial starting point). We divide the data become two part, the first contain 803 as training data and the last 30 data as testing data. We use Matlab routine as execute program that use `newff` as function to generate FFNN. The `dwt` and `idwt` syntax are used to decomposition and reconstruction of wavelet at single level, respectively, whereas `wavedec` and `waverec` syntax are used to decomposition and reconstruction of wavelet at multi level. The summary of the simulation result is shown in the table below.

TABLE 1
SIMULATION RESULT FROM IHSG BY USING FFNN MODEL WITH HAAR DWT AS PRE-PROCESSING AT SOME LEVEL

No	Level of Haar Wavelet	RMSE Training	RMSE Testing
1	1	19.77947	65.96432
2	2	33.38706	70.64850
3	3	36.79194	147.7053
4	4	34.28148	73.48404

From the table 1, we can make a justification that the best model to predict the IHSG data is FFNN with Haar at level one, in sample and also out-of sample. We have also found that `dwt` and `idwt` function give more stable resut to predict the data, whereas the `wavedec` and `waverec` function give unstable result, and we must chosen the few best results from some trials.

The plot of the best model at a trial, coverage of predict in sample and also predict out of sample are shown in fig. 4(a) and 4(b), respectively. From the plot in the figure we can look that the model give good result in predict the IHSG data.

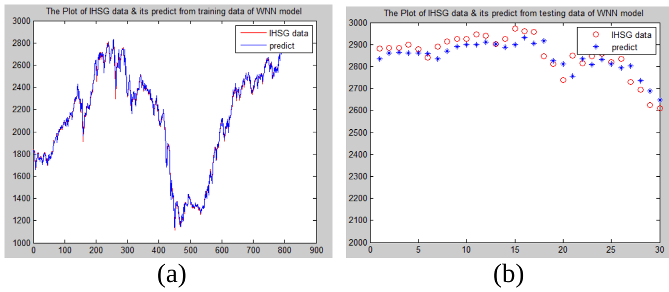

Fig. 4 Predict in-sample with 803 training data and predict out of sample with 30 testing data from IHSG data by using FFNN architecture with Haar wavelet as pre-processing

VI. Closing

There are still many others possibility to construct model FFNN with discrete wavelet transform as pre-processing to improve the performance of the network. We can try to use other wavelet like Daubechies, Symlets or Coifflets to get more powerful architecture, and compare them at various level. We can also use various methods to get optimal FFNN architecture and associate it with wavelet to get optimal of the new architecture.

References

[1] Kao, J.J & Huang, S.S, *Forecasts Using Neural Network versus Box-Jenkins Methodology for Ambient Air Quality Monitoring Data*, Journal of the Air and Waste Management Association,50:219-226, 2000

[2] Barai, S.V., Dikshit, A.A., Sharma, S., *Neural Network Models for Air Quality Prediction : A Comparative Study*, working paper, 2006

[3] Teo, K.K., Wang, L. and Lin, Z., *Wavelet Packet Multi-layer Perceptron for Chaotic Time Series Prediction: Effects of Weight Initialization*, V.N. Alexandrov et al. (Eds.): ICCS, LNCS 2074, pp. 310–317. Springer-Verlag Berlin Heidelberg, 2001

[4] Mallat, S., *A Wavelet Tour of Signal Processing*. New York: Academic Press, 1998

[5] Kozłowski, B., *Time Series Denoising with Wavelet Transform*, Journal of Telecommunications and Information Technology, Warsawa, Polandia, 2005

[6] Suparti and Subanar, *Estimasi Regresi dengan Metode Wavelet Shrinkage*, Jurnal Sains & Matematika, 8/3:105-113, 2000

[7] Banakar, A. and Azeem, M.F., *Generalized Wavelet Neural Network Model and its Application in Time Series Prediction,* International Joint Conference on Neural Networks, Vancouver, BC, Canada, July 16-21, 2006

[8] Sahoo, D. and Dulikravich, G.S., *Evolutionary Wavelet Neural Network for Large Scale Function Estimation in Optimization*, 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 6 - 8 September 2006, Portsmouth, Virginia, 2006

[9] Tao, H., *A Wavelet Neural Network Model for Forecasting Exchange Rate Integrated with Genetic Algorithm*, IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.8A, August, 2006

[10] Zhou, B., Shi, A., Cai, F. and Zhang, Y., *Wavelet Neural Networks for Nonlinear Time Series Analysis*, ISNN LNCS 3174, pp. 430–435, Springer-Verlag Berlin Heidelberg, 2004

[11] Suhartono and Subanar, *Development of Building Procedures in Wavelet Neural Networks for Forecasting Non-Stationary Time Series,* European Journal of Scientific Research, ISSN 1450-216X, Vol.34 No.3, pp. 416-427, @EuroJournals Publishing, Inc. 2009

[12] Bellil, W., Amar, C.B. and Alimi, A.M., *Synthesis of Wavelet Filters using Wavelet Neural Networks*, Proceedings Of World Academy Of Science, Engineering And Technology Volume 13 May Issn 1307-6884, 2005

[13] Fausset, L., *Fundamentals of Neural Network : Architectures, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, NJ07632, 1994