

# Fuzzy wavelet neural network for prediction of electricity consumption

RAHIB H. ABIYEV

Department of Computer Engineering, Near East University, Mersin, Turkey

(RECEIVED October 13, 2006; ACCEPTED November 21, 2007)

## Abstract

The development of a fuzzy wavelet neural network (FWNN) for the prediction of electricity consumption is presented. The fuzzy rules that contain wavelets are constructed. Based on these rules, the structure of FWNN-based system is described. The FWNN system is applied for modeling and prediction of complex time series. The gradient algorithm and genetic algorithm are used for learning of FWNN parameters. The developed FWNN is applied for prediction of electricity consumption. This process has high-order nonlinearity. The statistical data for the last 10 years are used for the development of FWNN prediction model. The effectiveness of the proposed system is evaluated with the results obtained from the simulation of FWNN-based prediction system and with the comparative simulation results of previous related models.

**Keywords:** Fuzzy Wavelet Neural Network; Neurofuzzy Modeling; Prediction of Electricity Consumption; Time Series Prediction; Wavelet Network

## 1. INTRODUCTION

The future behavior of a dynamic system can be predicted on the basis of historical states of that system or relationship of this behavior to other relevant variables. Electricity consumption prediction mainly deals with the amount of electricity that should be generated over a specific period of time. Such prediction plays an important role in the planning of electricity systems and greatly influences the national economy and people's daily life. The accuracy of electricity prediction is important for electricity utilities and for consumers.

A number of studies have been published about modeling of electricity consumption using regression analysis and econometric models and time series analysis. Recently, electricity consumption models using weather and population size (Yan, 1998), using economical and climatic variables (Rajan & Jain, 1999) and using economical variables (Egelioglu et al., 2001; Ceylan & Ozturk, 2004) have been studied. These models need measuring the number of climatic and economical variables. Sometimes obtaining the values of these variables is very difficult over the prediction period, and this is not enough for accurate model development.

Time series models have been extensively used in electricity consumption prediction. These are the growth curve

model (Ang & Ng, 1992) and the autoregressive moving average (ARIMA) model (Abdel-Aal & Al-Garni, 1997a; Saab et al., 2001). These models need a large number of historical data to obtain satisfactory prediction accuracy, and this accuracy depends on the order of nonlinearity of the considered problem. Like ARIMA models, the gray forecasting models were developed for electricity demand and load forecasting (Yao et al., 2003; Yao & Chi, 2004). These prediction models are based on technical analysis of time series, such as looking for trends, stationarity, seasonality, random noise variation, and moving average (Box et al., 1994). These time series models are linear models, and they do not provide enough satisfactory prediction accuracy for nonlinear processes.

Many studies have been devoted to the development and improvement of time series forecasting models. Chaotic time series were modeled and predicted using soft computing methodologies such as neural networks (NNs), fuzzy logics, and genetic algorithms (GAs; Lapades & Farber, 1987; Hill et al., 1994; Jang, 1997; Nunnari et al., 1998; Smaoui, 2000; Zhang & Chan, 2000). These models are nonlinear models, and have shown clear advantages over the traditional statistical ones (Maddala, 1996). NNs have recently been widely used for prediction of time series. Because of nonlinearity existing in electricity load series, NN models have gained importance in electricity demand and load forecasting (Abdel-Aal & Al-Garni, 1997b; Al-Shehri, 1999; Hsu & Chen, 2003). Their prediction capability exceeds those of the conventional methods. In this

Reprint requests to: Rahib H. Abiyev, Department of Computer Engineering, Near East University, P.O. Box 670, Lefkosh, TRNC, Mersin-10, Turkey. E-mail: rahib@neu.edu.tr

paper, to increase prediction accuracy and reduce search space and time for achieving the optimal solution, the combination of soft-computing technologies, such as wavelet NNs (WNNs) with a fuzzy knowledge base is used for time series prediction, in particular, for the prediction of electricity consumption in North Cyprus.

Fuzzy technology is an effective tool for dealing with complex, nonlinear processes characterized with ill-defined and uncertain factors. Traditionally, to develop a fuzzy system, human experts often carry out the generation of IF–THEN rules by expressing their knowledge. In the case of complicated processes, it is difficult for human experts to test all the input–output data and find necessary rules for the fuzzy system. To solve this problem and simplify the generating of IF–THEN rules, several approaches have been applied (Yager & Zadeh, 1994; Jang, 1997). Today, the use of NNs takes more importance for this purpose. NN models basically use the sigmoid activation function in neurons. However, the sigmoid function is not orthogonal, and the energy of the sigmoid function is limitless, and this leads to slow convergence. Wavelet function is a waveform that has limited duration and an average value of zero. The integration of the localization properties of wavelets and the learning abilities of NN shows advantages of WNNs over NN in complex nonlinear system modeling. A WNN that uses wavelet functions have been proposed by researchers for solving approximation and classification problems (Kugarajah & Zhang, 1995; Zhang & Benviste, 1995; Zhang et al., 1995). WNNs are used for the prediction of chaotic time series (Cao et al., 1995), for short-term and long-term prediction of electricity load (Chang et al., 1998; Khao et al., 2004). The wavelet analysis approximates the decomposed time series at different levels of resolution. Fuzzy WNN (FWNN) combines wavelet theory, fuzzy logic, and NNs. The synthesis of fuzzy wavelet neural inference system includes the finding of the optimal definitions of the premise and consequent part of fuzzy IF–THEN rules through the training capability of WNNs, evaluating the error response of the system. A combination of fuzzy technology and WNN has been considered for solving signal processing and control problems (Lin & Wang, 1996; Thuillard, 2000). Fuzzy systems with linear combination of the basis function (Lin & Wang, 1996), wavelet network model of fuzzy inference system (Thuillard, 2001; Guo et al., 2005) are proposed. Thuillard (2001) proposed to choose the membership functions from the family of scaling functions, and to construct the fuzzy system using wavelet techniques. Fuzzy wavelet network that includes combinations of three subnets: pattern recognition subnet, fuzzy reasoning subnet, and control synthesis subnet is introduced (Lin et al., 2005). The use of such multilayer structures complicates the architecture of the system. The FWNN structure that is constructed on the base of a set of fuzzy rules is proposed and used for approximating nonlinear functions (Daniel et al., 2001). The FWNN-based controller is developed for the control of dynamic plants (Abiyev, 2005) and time series prediction (Abiyev, 2006). The combination of wavelet network and fuzzy logic allows us to develop a system that has fast training speed, and to describe nonlinear objects

that are characterized with uncertainty. Wavelet transform has the ability to analyze nonstationary signals to discover their local details. Fuzzy logic allows us to reduce the complexity of the data and to deal with uncertainty. An NN has a self-learning characteristic that increases the accuracy of the prediction. These methodologies are used here to construct fuzzy wavelet neural inference system to solve electricity consumption prediction problem.

The paper is organized as follows: Section 2 presents the structure and learning algorithms of the FWNN system. Brief descriptions of the gradient descent algorithm and GA used for learning of FWNN are given. Section 3 contains simulation results of the FWNN used for prediction of chaotic time series. The application of developed structure is used for electricity consumption prediction. Comparative results of different models for time series prediction are given. Finally, a brief conclusion is presented in Section 4.

## 2. FWNN

Wavelets are defined in the following form:

$$\Psi_j(x) = \frac{1}{\sqrt{|a_j|}} \psi\left(\frac{x - b_j}{a_j}\right), \quad a_j \neq 0, j = 1, \dots, n. \quad (1)$$

Here,  $\Psi_j(x)$  represents the family of wavelet obtained from the single  $\psi(x)$  function by dilations and translations, where  $a_j = \{a_{1j}, a_{2j}, \dots, a_{mj}\}$  and  $b_j = \{b_{1j}, b_{2j}, \dots, b_{mj}\}$  are the dilation and translation parameters, respectively;  $x = \{x_1, x_2, \dots, x_m\}$  are input signals; and  $\psi(x)$  is localized in both time space and frequency space and is called a mother wavelet.

Wavelet networks include wavelet functions in the neurons of hidden layer of network. The output of WNN is calculated as

$$y = \sum_{j=1}^k w_j \Psi_j(x) = \sum_{j=1}^k w_j |a_j|^{-1/2} \psi(a_j^{-1}x - d_j). \quad (2)$$

Here,  $d_j = a_j^{-1} \times b_j$ ,  $\Psi_j(x)$  is the wavelet function of the  $j$ th unit of the hidden layer,  $w_j$  is the weight coefficients between the input and hidden layers, and  $a_i$  and  $b_j$  are parameters of the wavelet function. WNN has good generalization ability, can approximate complex functions to some precision very compactly, and can be easily trained than other networks, such as multilayer perceptrons and radial-based networks (Szu et al., 1996; Khao, 2004). A good initialization of the parameters of WNNs allows to obtain fast convergence. A number of methods is implemented for initializing wavelets, such as orthogonal least square procedure, clustering method (Kugarajah & Zhang, 1995; Zhang & Benviste, 1995). The optimal dilation and translation of the wavelet increases training speed and obtains fast convergence. The approximation and convergence properties of WNN are presented in Zhang et al. (1995).

This paper presents FWNN that integrate wavelet functions with the Takagi–Sugeno–Kang (TSK) fuzzy model. The kernel of the fuzzy system is the fuzzy knowledge base that consists of input–output data points of the system interpreted into

linguistic interpretable fuzzy rules. The consequent parts of TSK type fuzzy IF–THEN rules are represented by either a constant or a function. As a function, most of the fuzzy and neuro-fuzzy models use linear functions. Neurofuzzy systems can describe the considered problem by means of combination of linear functions. Sometimes these systems need more rules for modeling complex nonlinear processes to obtain the desired accuracy. Increasing the number of the rules leads to increasing number of neurons in the hidden layer of the network. To improve the computational power of the neurofuzzy system, we use wavelets in the consequent part of each rule. In this paper, the fuzzy rules that are constructed by using wavelets are used. They have the following form:

$$\begin{aligned}
 &\text{If } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{12} \text{ and } \dots \text{ and } x_m \text{ is } A_{1m}, \\
 &\text{then } y_1 \text{ is } \sum_{i=1}^m w_{i1}(1 - z_{i1}^2)e^{-(z_{i1}^2/2)} \\
 &\text{If } x_1 \text{ is } A_{21} \text{ and } x_2 \text{ is } A_{22} \text{ and } \dots \text{ and } x_m \text{ is } A_{2m}, \\
 &\text{then } y_2 \text{ is } \sum_{i=1}^m w_{i2}(1 - z_{i2}^2)e^{-(z_{i2}^2/2)} \\
 &\dots \\
 &\text{If } x_1 \text{ is } A_{n1} \text{ and } x_2 \text{ is } A_{n2} \text{ and } \dots \text{ and } x_m \text{ is } A_{nm}, \\
 &\text{then } y_n \text{ is } \sum_{i=1}^m w_{in}(1 - z_{in}^2)e^{-(z_{in}^2/2)}. \tag{3}
 \end{aligned}$$

Here,  $x_1, x_2, \dots, x_m$  are input variables,  $y_1, y_2, \dots, y_n$  are output variables that are Mexican Hat wavelet functions,  $A_{ij}$  is a membership function for the  $i$ th rule of the  $j$ th input defined

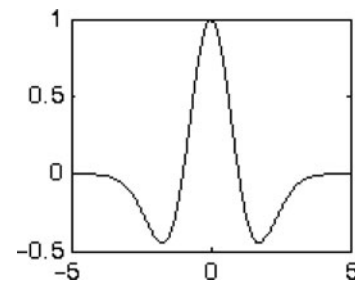


Fig. 1. Mexican hat wavelet.

as Gaussian membership function.  $n$  is the number of fuzzy rules. Conclusion parts of rules contain WNNs. The WNNs include Mexican Hat wavelet function (Fig. 1). Here fuzzy rules provide the influence of each WNN to the output of FWNN. The use of WNN with different dilation and translation values allows to capture different behaviors and essential features of the nonlinear model under these fuzzy rules. The proper fuzzy model that is described by IF–THEN rules can be obtained by learning dilation and translation parameters of conclusion parts and the parameters of membership function of premise parts. Here, because of the use of wavelets, the computational strength and generalization ability of FWNN is improved, and FWNN can describe the nonlinear processes with desired accuracy.

The structure of fuzzy wavelet system is given in Figure 2. The FWNN includes six layers. In the first layer, the number of nodes is equal to the number of input signals. These nodes are used for distributing input signals. In the second layer, each node corresponds to one linguistic term. For each input

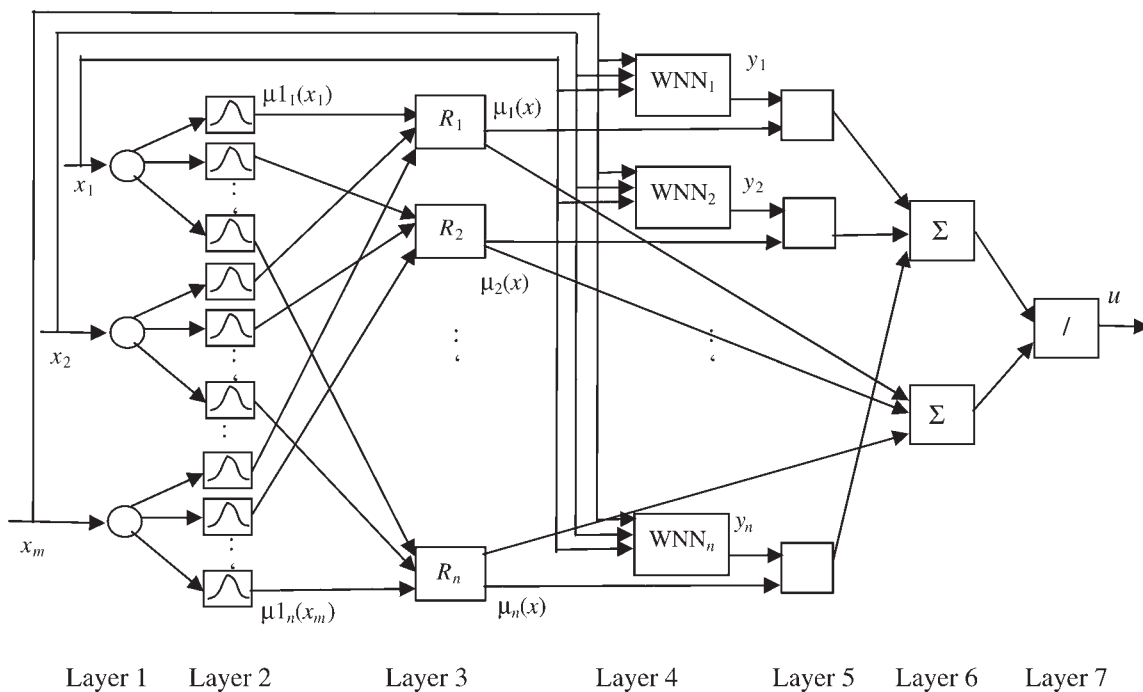


Fig. 2. Structure of fuzzy wavelet neural network.

signal entering to the system, the membership degree to which input value belongs to a fuzzy set is calculated. To describe linguistic terms, the Gaussian membership function is used.

$$\mu_{1j}(x_i) = e^{-[(x_i - c_{ij})^2 / \sigma_{ij}^2]}, \quad i = 1, \dots, m, j = 1, \dots, n. \quad (4)$$

Here,  $m$  is number of input signals;  $n$  is number of fuzzy rules (hidden neurons in third layer);  $c_{ij}$  and  $\sigma_{ij}$  are center and width of the Gaussian membership functions, respectively; and  $\mu_{1j}(x_i)$  is membership function of  $i$ th input variable for  $j$ th term.

In the third layer, the number of nodes corresponds to the number of rules  $R_1, R_2, \dots, R_n$ . Each node represents one fuzzy rule. The AND (min) operation is used here to calculate the values of output signals of the layer.

$$\mu_j(x) = \prod_i \mu_{1j}(x_i), \quad i = 1, \dots, m, j = 1, \dots, n, \quad (5)$$

where  $\prod$  is the min operation.

These  $\mu_j(x)$  signals are input signals for the next layer. This layer is a consequent layer. It includes  $n$  WNNs that are denoted by  $WNN_1, WNN_2, \dots, WNN_n$ . In the fifth layer, the output signals of third layer are multiplied by the output signals of wavelet networks. The output of  $j$ th wavelet network is calculated as

$$y_j = w_j \Psi_j(z); \quad \Psi_j(z) = \sum_{i=1}^m \frac{1}{\sqrt{|a_{ij}|}} (1 - z_{ij}^2) e^{-z_{ij}^2/2}. \quad (6)$$

Here,  $z_{ij} = (x_i - b_{ij})/a_{ij}$ . Here,  $a_{ij}$  and  $b_{ij}$  are parameters of the wavelet function between  $i$ th ( $i = 1, \dots, n$ ) input and  $j$ th ( $j = 1, \dots, m$ ) WNN. In sixth and seventh layers defuzzification is made to calculate the output of whole network. In this layer the contribution of each WNN to the output of the FWNN is determined.

$$u = \sum_{j=1}^n \mu_j(x) y_j / \sum_{j=1}^n \mu_j(x). \quad (7)$$

Here,  $y_j$  are the output signals of WNNs. After calculating the output signal of the FWNN, the training of the network starts.

### 2.1. Learning using gradient descent

At the beginning, the parameters of FWNN are generated randomly. The parameters are the membership function of linguistic values in the second layer of the network and the parameters of the WNN. To generate a proper FWNN model, the training of the parameters has been carried out. Training includes the adjusting of the parameter values of the membership functions  $c_{ij}(t)$  and  $\sigma_{ij}(t)$  ( $i = 1, \dots, m, j = 1, \dots, n$ ) in the premise part and parameter values of the WNNs  $w_j(t), a_{ij}(t), b_{ij}(t)$  ( $i = 1, \dots, m, j = 1, \dots, n$ ) in the consequent part. In this paper we applied gradient learning with an adaptive learning rate. The adaptive learning rate guarantees the converg-

ence and speeds up the learning of the network. In addition, the momentum is used to speed up the learning processes.

At first, on the output of network, the value of cost function is calculated.

$$E = \frac{1}{2} \sum_{i=1}^O (u_i^d - u_i)^2. \quad (8)$$

Here,  $O$  is the number of output signals of the network (in a given case  $O = 1$ ),  $u_i^d$  and  $u_i$  are desired and current output values of the network, respectively. The parameters  $w_j, a_{ij}, b_{ij}$  ( $i = 1, \dots, m, j = 1, \dots, n$ ) of WNN and parameters of membership function  $c_{ij}$  and  $\sigma_{ij}$  ( $i = 1, \dots, m, j = 1, \dots, n$ ) of neurofuzzy structure are adjusted using the following formulas.

$$\begin{aligned} w_j(t+1) &= w_j(t) + \gamma \frac{\partial E}{\partial w_j} + \lambda (w_j(t) - w_j(t-1)), \\ a_{ij}(t+1) &= a_{ij}(t) + \gamma \frac{\partial E}{\partial a_{ij}} + \lambda (a_{ij}(t) - a_{ij}(t-1)), \\ b_{ij}(t+1) &= b_{ij}(t) + \gamma \frac{\partial E}{\partial b_{ij}} + \lambda (b_{ij}(t) - b_{ij}(t-1)), \\ i &= 1, \dots, m; j = 1, \dots, n; \end{aligned} \quad (9)$$

$$c_{ij}(t+1) = c_{ij}(t) + \gamma \frac{\partial E}{\partial c_{ij}}, \quad \sigma_{ij}(t+1) = \sigma_{ij}(t) + \gamma \frac{\partial E}{\partial \sigma_{ij}}, \quad i = 1, \dots, m, j = 1, \dots, n \quad (10)$$

Here,  $\gamma$  is the learning rate,  $\lambda$  is the momentum,  $m$  is the number of input signals of the network (input neurons), and  $n$  is the number of fuzzy rules (hidden neurons).

The values of derivatives in Eq. (9) are computed using the following formulas.

$$\begin{aligned} \frac{\partial E}{\partial w_j} &= \frac{\partial E}{\partial u} \frac{\partial u}{\partial y_j} \frac{\partial y_j}{\partial w_j} = (u(t) - u^d(t)) \cdot \mu_j \cdot \psi_j(z) / \sum_{j=1}^n \mu_j, \\ \frac{\partial E}{\partial a_{ij}} &= \frac{\partial E}{\partial u} \frac{\partial u}{\partial y_j} \frac{\partial y_j}{\partial \psi_j} \frac{\partial \psi_j}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial a_{ij}} = \delta_j (3.5 z_{ij}^2 - z_{ij}^4 - 0.5) e^{-z_{ij}^2/2} / (\sqrt{a_{ij}^3}), \\ \frac{\partial E}{\partial b_{ij}} &= \frac{\partial E}{\partial u} \frac{\partial u}{\partial y_j} \frac{\partial y_j}{\partial \psi_j} \frac{\partial \psi_j}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial b_{ij}} = \delta_j (3 z_{ij} - z_{ij}^3) e^{-z_{ij}^2/2} / (\sqrt{a_{ij}^3}), \end{aligned} \quad (11)$$

here

$$\delta_j = \frac{\partial E}{\partial u} \frac{\partial u}{\partial y_j} \frac{\partial y_j}{\partial \psi_j} = (u(t) - u^d(t)) \cdot \mu_j \cdot w_j / \sum_{j=1}^n \mu_j, \quad i = 1, \dots, m, j = 1, \dots, n.$$

The derivatives in Eq. (10) are determined by the following formulas.

$$\frac{\partial E}{\partial c_{ij}} = \sum_j \frac{\partial E}{\partial u} \frac{\partial u}{\partial \mu_j} \frac{\partial \mu_j}{\partial c_{ij}}, \quad (12)$$

$$\frac{\partial E}{\partial \sigma_{ij}} = \sum_j \frac{\partial E}{\partial u} \frac{\partial u}{\partial \mu_j} \frac{\partial \mu_j}{\partial \sigma_{ij}} \tag{13}$$

Here

$$\frac{\partial E}{\partial u} = u(t) - u^d(t), \quad \frac{\partial u}{\partial \mu_j} = \frac{y_j - u}{\sum_{j=1}^n \mu_j},$$

$$i = 1, \dots, m, j = 1, \dots, n \tag{14}$$

$$\frac{\partial \mu_j(x_i)}{\partial c_{ij}} = \begin{cases} \mu_j(x_i) \frac{2(x_i - c_{ij})}{\sigma_{ij}^2} & \text{if } j \text{ node} \\ \text{is connected to rule node } j \\ 0, & \text{otherwise} \end{cases}$$

$$\frac{\partial \mu_j(x_i)}{\partial \sigma_{ij}} = \begin{cases} \mu_j(x_i) \frac{2(x_i - c_{ij})^2}{\sigma_{ij}^3} & \text{if } j \text{ node} \\ \text{is connected to rule node } j \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

Using Eqs. (11)–(15), the derivatives in Eqs. (9) and (10) are calculated and the correction of the parameters of FWNN is carried out.

### 2.2. GA learning

Sometimes the network learning using the gradient method for nonlinear processes cannot guarantee an optimal solution. In practical applications, the gradient method might find the set of suboptimal weight from which it cannot escape. That is, sometimes the gradient method has “local minima” problem and could not find a global optimal solution. GAs are effective optimization techniques that can be used to improve training of the FWNN and avoid the local minima problem. GA is a directed random search method that exploits historical information to direct the search into the region of better performance within the search space (Goldberg, 1998). In this paper, the real coded GA with gradient descent algorithm is applied for searching optimal values of the parameters of the FWNN.

During optimization, the number of chromosomes, which is defined as population size, is generated randomly. Chromosomes consist of genes and represent the network parameters. These parameters characterize the parameters of antecedent and consequent parts of the FWNN. The long chromosomes are used to represent all parameter values.

GA learning is applied to train the parameter values in chromosomes. GA learning is carried out by GA operators. The main operations in GA are selection, crossover, and mutation. The aim of the selection is to give more reproductive chance to population members (or solutions) that have higher fitness. The tournament selection is applied for selection of new generation. In this method, two members of the population are selected and their fitness values are compared. The member with high fitness is selected for the next generation.

Crossover and mutation are two main components in the reproduction process in which selected pairs mate to produce the next generation. The purpose of crossover and mutation is to give the next generation of solutions chance to differ from their parental solutions. Both components intend to give children chance to differ from their parents, and hope that some of the children can be closer to the optimal destination than their parents.

The real coded multipoint crossover operation is used for the correction of individuals. According to crossover rate, the individuals are selected for the crossover operation to generate a new solution. The high value of the crossover rate led to a quick generation of a new solution. The typical value of the crossover rate is selected in the interval [0.5, 1]. In crossover operation two parent members  $X = (x_1, x_2, \dots, x_n)$  and  $Y = (y_1, y_2, \dots, y_n)$  are selected. After crossover operation, the new members will have the form  $X' = (x'_1, x'_2, \dots, x'_n)$  and  $Y' = (y'_1, y'_2, \dots, y'_n)$ . The crossover operation has been performed using the following formula.

$$\begin{aligned} x'_i &= x_i + \delta(y_i - x_i), \\ y'_i &= x_i + \delta(x_i - y_i), \end{aligned} \tag{16}$$

when  $F(X) > F(Y)$ . Here,  $x_i$  and  $y_i$  are the  $i$ th genes of the parents  $X$  and  $Y$ ,  $x'_i$  and  $y'_i$  are the  $i$ th genes of the parents  $X'$  and  $Y'$ . The value  $\delta$  is changed between 0 and 1.

The simple mutation operation is applied. In this operation, for each gene, a random number is generated. If this random number is less than the mutation rate, then the corresponding gene is selected for mutation. During mutation a small random number, taken from the interval [0, 1], is added to the selected gene to determine its new value. A large value of the mutation rate leads to a purely random search. The typical value of mutation rate is selected from the interval [0, 0.1].

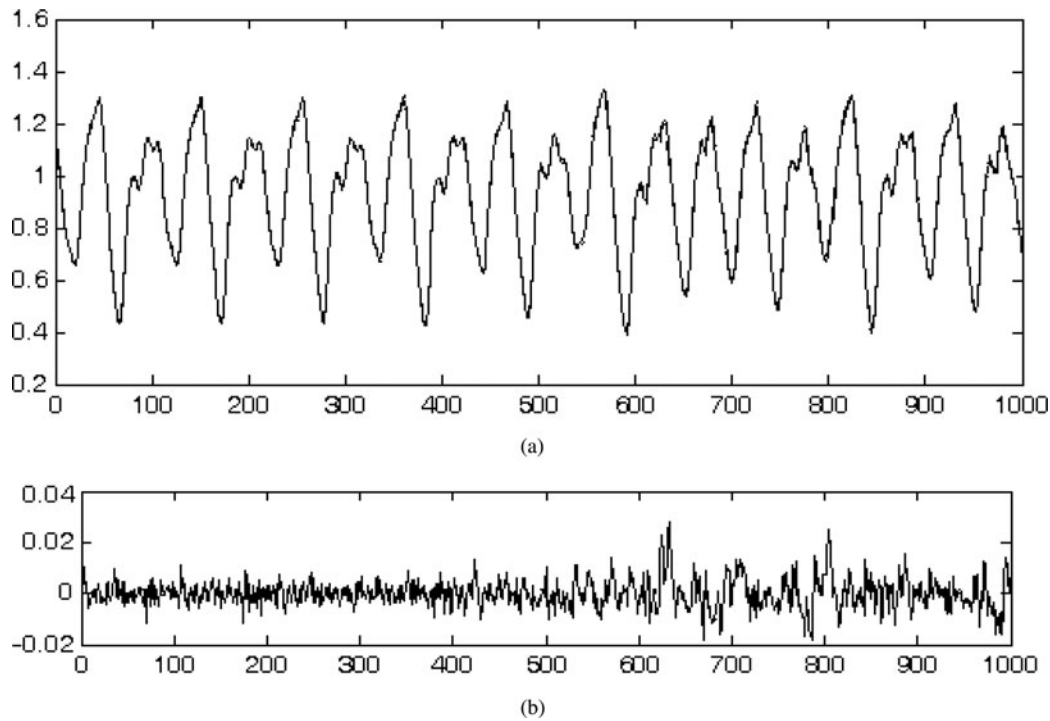
## 3. SIMULATION OF TIME SERIES PREDICTION

### 3.1. Time series prediction

The FWNN structure and its learning algorithms are applied for modeling and predicting the future values of chaotic time series. As an example, the Mackey–Glass time series data set was taken. This is a benchmark problem in the areas of NN and fuzzy systems. This time series data set was created with the use of the following Mackey–Glass time-delay differential equation.

$$\frac{dx(t)}{dt} = \frac{0.2x(t - \tau)}{1 + x^{10}(t - \tau)} - 0.1x(t). \tag{17}$$

This time series is chaotic, and the trajectory is highly sensitive to initial conditions. To obtain the data set, the fourth-order Runge–Kutta method is applied to find a numerical solution to the above Mackey–Glass equation. Here we assume that  $x(0) = 1.2$ ,  $\tau = 17$ , and  $x(t) = 0$  for  $t < 0$ . The task is to predict the values  $x(t + pr)$  from input vectors  $[x(t - 18) \ x(t - 12)$



**Fig. 3.** (a) The six-step ahead prediction for Mackey–Glass time series from  $t = 117–1118$  and (b) prediction error. The first 500 data points are used for training and the second 500 for testing.

$x(t - 6) x(t)]$  for any value of the time  $t$ , where  $pr$  is the predicting step. The value of  $pr$  is assumed to be 6. Using statistical data obtained from Eq. (17), the learning of FWNN has been carried out. The learning is accomplished by using GA and the gradient descent algorithm. The gradient method has good learning speed, but it has a local minima problem. GA can find a global optimal solution, but it has low learning speed. During learning, at first step, the GA operators are applied for learning the network parameters. This process is continued up to given number of iterations. The learned values of parameters are saved in the file. Then using the gradient descent algorithm, the learning of the same parameter values is continued. The usage of such approach allows us to speed up learning process and finds a global optimal solution.

For comparative analysis, the obtained results are compared with existing online models applied to the same task. The 16 rules are used in the neurofuzzy part of the FWNN. As a performance criterion, the nondimensional error index (NDEI), which is defined as the root mean square error (RMSE) divided by the standard deviation of target series is used. NDEI is determined as

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i^d - x_i)^2},$$

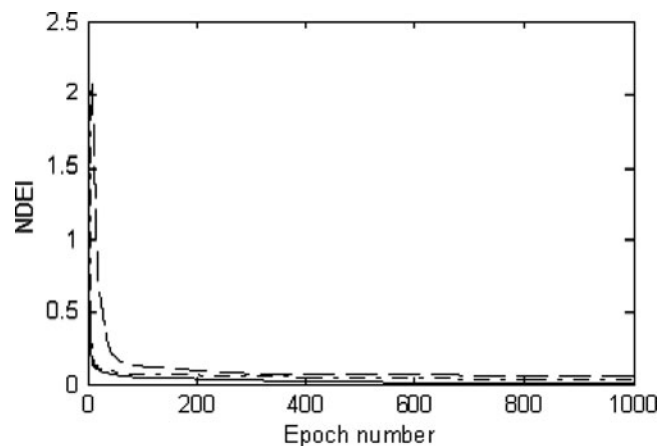
$$NDEI = \frac{RMSE}{\sigma} = \sqrt{\frac{\sum_{i=1}^N (x_i^d - x_i)^2}{\sum_{i=1}^N (x_i^d - \bar{x})^2}} \quad (18)$$

Here,

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i^d - \bar{x})^2}, \quad \bar{x} = \frac{\sum_{i=1}^N x_i^d}{N},$$

where  $x_i^d$  and  $x_i$  denote the desired output and model output, respectively, and  $\bar{x}$  is the mean of target series.

In the first experiment 1000 data points ( $t = 117–1118$ ) are extracted from the time series and used as a learning data. The first 500 data points were used for learning, and the second



**Fig. 4.** Convergence graphics. (---) NDEI of NN, (- · -) NDEI of WNN, and (—) NDEI of FWNN.

**Table 1.** Six-step ahead prediction results for Mackey–Glass time series

Method	Epochs	NDEI	
		Training	Testing
Feedforward NN	5000	0.0391	0.041
WNN	2000	0.0302	0.031
FWNN	1000	0.0154	0.016

500 data points were used for testing. During learning, the values of RMSE and NDEI were 0.00345 and 0.015401, respectively. After learning, in the generalization step, RMSE = 0.0036 and NDEI = 0.016. In Figure 3a, the trajectories of desired and predicted values for both training and checking data for  $pr = 6$  are shown. Here, the solid line indicates the trajectory of statistical data and the dashed line indicates the predicted value of time series. The difference between them is very small. These differences might only be seen in large scale. In Figure 3b, the prediction error is shown. For comparative analysis, the feedforward NN and WNN-based prediction models are developed. The result of the feedforward NN-based model is obtained when the number of hidden neurons was 60. The convergence graphic, describing the learning process of NN-, WNN-, and FWNN-based prediction models are given in Figure 4. The result of the WNN-based prediction model is obtained using 16 hidden neurons. Table 1 demonstrates the training and offline prediction results of feedforward NN-, WNN-, and FWNN-based models for Mackey–Glass time series. Comparisons of test results of different prediction models for Mackey–Glass time series are given in Table 2. In addition, using 58 fuzzy rules, the learning of FWNN has been performed. Figure 5 depicts the prediction error obtained for the test data from the FWNN model.

In the second experiment, using 58 fuzzy rules, the learning of FWNN for  $pr = 84$  has been performed. The value of NDEI was 0.046. The increasing number of rules affects to the decreasing of the NDEI value. Table 3 demonstrates the off-line prediction results of different models used for Mackay–Glass time series. In the table the results from second to fifth lines are from Crowder (1990). As shown, the prediction error of the FWNN model is lower than those obtained from other models.

**Table 2.** Comparisons of test results of different prediction models for Mackey–Glass time series

Method	Prediction Error, NDEI
Autoregressive model	0.19
Backpropagation NN	0.02
Sixth-order polynomial	0.04
Linear predictive method	0.55
ANFIS (Jang, 1993)	0.007
Neural tree (Chen et al., 2005)	0.0069
WNN	0.0071
FWNN	0.0036

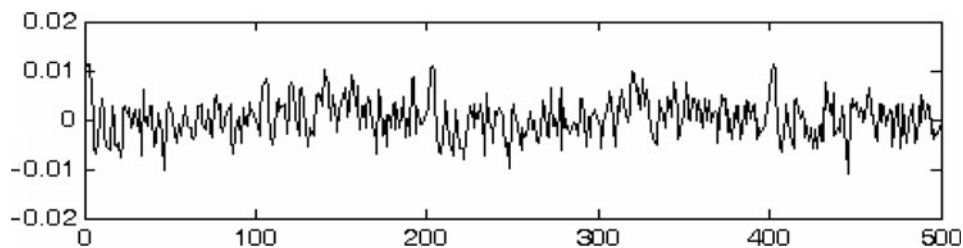
**Table 3.** Prediction results comparisons

Method	Epochs	NDEI
FWNN	500	0.046
Cascaded-correction NN	500	0.32
Sixth-order polynomial	500	0.85
Backpropagation NN	500	0.05
Linear predictive method	2000	0.60

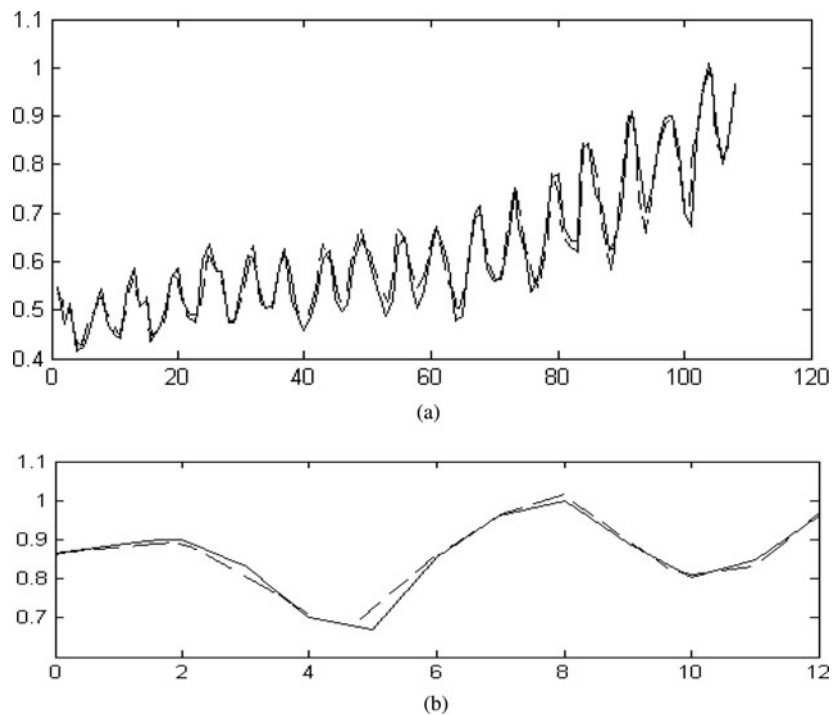
### 3.2. Modeling of electricity consumption

The FWNN system is applied for constructing a prediction model of electricity consumption in North Cyprus. Cyprus does not have petroleum and gas reserves and imports them from abroad. Energy is supplied by the KIB-TEK Company. Here the main goal is to meet customer demand. For planning utilities, it is needed to develop an electricity consumption model. The statistical data for the last 10 years were obtained from KIB-TEK. It was important to know in what volume of electricity would be used in the near future (after a few month), even an approximate value would be sufficient.

The FWNN structure and its learning algorithm is used to construct the prediction model. In the prediction problem, it is needed to predict the value of electricity consumption in the near future  $x(t + pr)$  on the base of sample data points  $\{x(t - (D - 1)\Delta), \dots, x(t - \Delta), \dots, x(t)\}$ . Here,  $pr$  is the prediction step. Five input data points  $[x(t - 12) x(t - 6) x(t - 5) x(t - 2) x(t)]$  are used as input to the prediction model. The output training data corresponds to  $x(t + 12)$ . In other words, because the electricity consumption is considered monthly, the value that is to be predicted will be after  $pr = 12$  months.



**Fig. 5.** Prediction error.



**Fig. 6.** The twelve-step ahead prediction. Plot of output signals: (dotted line) generated by FWNN and (solid line) predicted signal. (a) Curves describing learning and testing data together and (b) curves for testing data.

The training input/output data for the prediction system will be a structure whose first component is the five dimension input vector, and the second component is the predicted output.

To start the training, the FWNN structure is generated. It includes five input and one output neurons. Sixteen hidden neurons (rules) are used in the hidden layer of the neurofuzzy part of FWNN. The second layer of the system includes a Gaussian membership function for each input signal. Eleven neurons are used in the conclusion part of the hidden layer of the WNN network. The initial values of the membership functions are generated equally spaced, and these values cover the whole input space. The training of the parameters was performed by using learning algorithms described in Section 2.

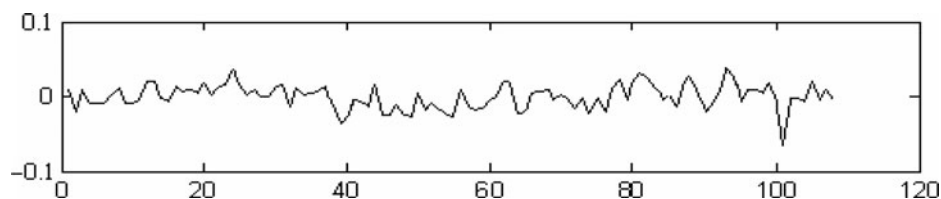
For training of the system, the statistical data describing monthly electricity consumption from January 1995 to December 2005 are considered. The data from January 2006 to December 2006 are used for diagnostic testing. All input and output data are scaled in the interval  $[0, 1]$ . The training is carried out for 1000 epochs. The values of the parameters of the FWNN system were determined at the conclusion of training.

Once the FWNN has been successfully trained, it is then used for the prediction of the 2006 monthly electricity consumption. The training and test values of NDEI were 0.2288 and 0.2441 correspondingly.

In Figure 6a, the output of the FWNN system for 12-step ahead prediction of electricity consumption for learning and generalization step is shown. Here the solid line is desired output, and the dashed line is the FWNN output. Figure 6b demonstrates the 12-step ahead prediction of FWNN.

The plot of prediction error is shown in Figure 7. As shown in the figure, in the generalization step (end part of the error curve), the value of error increases. The result of the simulation of the FWNN prediction model is compared with the result of simulation of the feedforward NN-based prediction model. To estimate the performance of the neural and FWNN prediction systems, the NDEI values of errors between the predicted and current output signal are compared.

Table 4 provides the comparative results of simulations. As shown in the table, the performance of FWNN prediction is better than the performance of the NN model.



**Fig. 7.** A plot of the prediction error.



**Table 4.** Comparative results of simulation

Method	No. of Rules	Epochs	NDEI
Feedforward NN	16	10,000	0.3287
FWNN	16	1,000	0.2441
FWNN	58	1,000	0.1462

The simulation results satisfy the efficiency of the application of FWNN technology in constructing a prediction model of electricity consumption.

#### 4. CONCLUSION

The time series prediction model is developed by integrating fuzzy logic, neural networks, and wavelet technology. The wavelet networks are used to construct the fuzzy rules, and the functionality of the fuzzy system is realized by the neural network structure. The gradient and GAs are applied for optimization of the parameters in the premise and consequent parts of fuzzy rules in the FWNN structure. The structure and learning algorithms of the FWNN system is applied for modeling and prediction of complex time series. Simulation results demonstrated that the applied FWNN structure has better performance than other models. The developed FWNN structure is applied to develop a model for predicting future values of electricity consumption. This process is high order nonlinear. Using statistical data, the prediction model is constructed. The test results of the developed system are compared with these obtained from the feedforward NN-based system, and the first one has demonstrated better performance.

#### ACKNOWLEDGMENT

The author thanks the anonymous referees for their helpful suggestions in improving this manuscript.

#### REFERENCES

- Abdel-Aal, R.E., & Al-Garni, A.Z. (1997a). Forecasting monthly electric energy consumption in eastern Saudi Arabia using univariate time series analysis. *Energy* 22, 1059–1069.
- Abdel-Aal, R.E., & Al-Garni, A.Z. (1997b). Forecasting monthly electric energy consumption in eastern Saudi Arabia using abductives networks. *Energy* 22, 911–921.
- Abiyev, R.H. (2005). Controller based of fuzzy wavelet neural network for control of technological processes. *CIMSA 2005, IEEE Int. Conf. Computational Intelligence for Measurement Systems and Applications*, pp. 215–219, Giardini Naxos, Italy.
- Abiyev, R.H. (2006). Time series prediction using fuzzy wavelet neural network model. *ICANN 2006, Part II. Lecture Notes in Computer Sciences*, Vol. 4132. Berlin: Springer-Verlag, pp. 191–200.
- Al-Shehri, A. (1999). Artificial neural network for forecasting residential electrical energy. *International Journal of Energy Research* 23, 649–661.
- Ang, B.W., & Ng, T.T. (1992). The use of growth curves in energy studies. *Energy* 17, 25–36.
- Box, G.E.P., Jenkins, G.M., & Reinsel, G.C. (1994). *Time Series Analysis, Forecasting and Control*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall.

- Cao, L., Hong, Y., Fang, H., & He, G. (1995). Predicting chaotic time series with wavelet networks. *Physica D* 85, 225–238.
- Ceylan, H., & Ozturk, H. (2004). Estimating of energy demand of Turkey based on economic indicators using genetic algorithm approach. *Energy Conversion and Management* 45, 2525–2537.
- Chang, P.R., Weihui, Fu, & Minjun, Yi. (1998). Short term load forecasting using wavelet networks. *Engineering Intelligent Systems for Electrical Engineering and Communications* 6, 217–230.
- Chen, Y., Yang, B., Dong, J., & Abraham, A. (2005). Time series forecasting using flexible neural tree model. *Information Sciences* 174(3–4), 219–235.
- Crowder, III R.S. (1990). Predicting the Mackey–Glass time series with cascade correlation learning. *Proc. 1990 Connectionist Models Summer School* (Touretzky, D., Hinton, G., & Sejnowski, T., Eds.), pp. 117–123. Pittsburgh, PA: Carnegie Mellon University.
- Daniel, W.C.H., Ping-An, Z., & Jinhua, X. (2001). Fuzzy wavelet networks for function learning. *IEEE Transactions on Fuzzy Systems* 9(1), 200–211.
- Egelioglu, F., Mohamad, A.A., & Guven, H. (2001). Economic variables and electricity consumption in Northern Cyprus. *Energy* 26, 355–362.
- Goldberg, D.E. (1998). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley.
- Hill, T., O'Connor, M., & Remus, W. (1994). Artificial neural network models for forecasting and decision making. *International Journal of Forecasting* 10, 5–15.
- Hsu, C.C., & Chen, C.Y. (2003). Regional load forecasting in Taiwan—applications of artificial neural networks. *Energy Conversion and Management* 44, 1941–1949.
- Jang, J.-S.R. (1993). ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics* 23(3), 665–683.
- Jang, J.-S.R. (1997). *Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence*. Englewood Cliffs, NJ: Prentice-Hall.
- Khao, T.Q.D., Phuong, L.M., Binh, P.T.T., & Lien, N.T.H. (2004). Application of wavelet and neural network to long-term load forecasting. *Int. Conf. Power System Technology, POWERCON 2004*, pp. 840–844, Singapore.
- Kugarajah, T., & Zhang, Q. (1995). Multidimensional wavelet frames. *IEEE Transactions on Neural Networks* 6, 1552–1556.
- Lapades, A., & Farber, R. (1987). *Nonlinear Signal Processing Using Neural Networks: Prediction and Signal Modeling*, Vol. 87545. Technical Report. Los Alamos, NM: Los Alamo National Lab.
- Lin, C.K., & Wang, S.D. (1996). Fuzzy modelling using wavelet transform. *Electronics Letters* 32, 2255–2256.
- Lin, Y., & Wang, F.Y. (2005). Predicting chaotic time series using adaptive wavelet-fuzzy inference system. *Proc. IEEE Intelligent Vehicles Symposium*, pp. 888–893.
- Maddala, G.S. (1996). *Introduction to Econometrics*. Englewood Cliffs, NJ: Prentice-Hall.
- Nunnari, G., Nucifora, A., & Randieri, C. (1998). The application of neural techniques to the modeling of time series of atmospheric pollution data. *Ecological Modelling* 111, 187–205.
- Guo, Q.J., Yu, H.B., & Xu, A.D. (2005). Wavelet fuzzy network for fault diagnosis. *Proc. Int. Conf. Communications, Circuits and Systems*, pp. 993–998.
- Rajan, M., & Jain, V.K. (1999). Modelling of electrical energy consumption in Delhi. *Energy* 24, 351–361.
- Saab, S., Badr, E., & Nasr, G. (2001). Univariate modelling and forecasting of energy consumption: the case of electricity in Lebanon. *Energy* 26, 1–14.
- Smaoui, N. (2000). An artificial neural network noise reduction method for chaotic attractors. *International Journal of Computer Mathematics* 73, 417–431.
- Szu, H., Telfer, B., & Garcia, J. (1996). Wavelet transforms and neural networks for compression and recognition. *Neural Networks* 9, 695–708.
- Thuillard, M. (2000). Fuzzy logic in the wavelet framework. *Proc. Toolmet 2000*, April 13–14, Oulu.
- Thuillard, M. (2001). *Wavelets in Softcomputing*. Singapore: World Scientific Press.
- Yager, R.R., & Zadeh, L.A., Eds. (1994). *Fuzzy Sets, Neural Networks and Soft Computing*. New York: Van Nostrand Reinhold.
- Yan, Y.Y. (1998). Climate and residential electricity consumption in Hong Kong. *Energy* 23(1), 17–20.

- Yao, A.W.L., & Chi, S.C. (2004). Analysis and design of a Taguchi–Grey based electricity demand predictor for energy management systems. *Energy Conversion and Management* 45, 1205–1217.
- Yao, A.W.L., Chi, S.C., & Chen, J.H. (2003). An improved grey-based approach for electricity demand forecasting. *Electric Power Systems Research* 67, 217–214.
- Zhang, J., Walter, G.G., & Wayne Lee, W.N. (1995). Wavelet neural networks for function learning. *IEEE Transaction on Signal Processing* 43(6), 1485–1497.
- Zhang, Q., & Benviste, A. (1995). Wavelet networks. *IEEE Transactions on Neural Networks* 3, 889–898.
- Zhang, Y.Q., & Chan, L.W. (2000). Fourier recurrent networks for time series prediction. *Proc. Int. Conf. Neural Information Processing, ICONIP 2000*, pp. 576–582, Tacjon, Korea.

of the Computer Engineering Department and Director of the Applied Artificial Intelligence Laboratory at Near East University. He received his PhD degree in electrical and electronic engineering from Azerbaijan State Oil Academy (USSR) in 1997. He worked as a Research Assistant at the Industrial Intellectual Control Systems Laboratory. Dr. Abiyev's research interests are NNs, fuzzy systems, digital signal processing, control systems, optimization, GAs, and pattern recognition.

---

**Rahib Hidayat Abiyev** has been an Associate Professor in the Department of Computer Engineering at Near East University, TRNC, Turkey, since 1999. He is Vice Chairman