

A NOVEL APPROACH BASED ON GENETIC FUZZY CLUSTERING AND ADAPTIVE NEURAL NETWORKS FOR SALES FORECASTING

¹Attarius Hichama, ¹Bouhorma Mohameda and ²El Fallahi Abdellahb

¹Laboratory LIST, Group Research in Computing and Telecommunications (ERIT) FST, Abdelmalek Essaadi University, BP: 416, Old Airport Road, Tangier, Morocco

²Laboratory LITT, Group Research in Computing and Logistic (ERIT) ENSA, Abdelmalek Essaadi University, Mhannech II, BP: 2121, Tetouan, Morocco

Received 2012-12-24, Revised 2013-03-03; Accepted 2013-07-02

ABSTRACT

This article proposes a new hybrid sales forecasting system based on genetic fuzzy clustering and Back-Propagation (BP) Neural Networks with adaptive learning rate (GFCBPN). The proposed architecture consists of three stages: (1) utilizing Winter's Exponential Smoothing method and Fuzzy C-Means clustering, all normalized data records will be categorized into k clusters; (2) using an adapted Genetic Fuzzy System (MCGFS), the fuzzy rules of membership levels to each cluster will be extracted; (3) each cluster will be fed into parallel BP networks with a learning rate adapted as the level of cluster membership of training data records. Compared to previous researches which use Hard clustering, this research uses the fuzzy clustering which capable to increase the number of elements of each cluster and consequently improve the accuracy of the proposed forecasting system. Printed Circuit Board (PCB) will be utilized as a case study to evaluate the precision of our proposed system. Experimental results show that the proposed model outperforms the previous and traditional approaches. Therefore, it is a very promising method for financial forecasting.

Keywords: Sales Forecasting, Fuzzy Clustering, Genetic Fuzzy System, Printed Circuit Boards, Back Propagation Network, Hybrid Intelligence Approach

1. INTRODUCTION

Introduction and related research Reliable prediction of sales becomes a vital task of business decision making. Companies that use accurate sales forecasting system earn important benefits. Sales forecasting is both necessary and difficult. It is necessary because it is the starting point of many tools for managing the business: production schedules, finance, marketing plans and budgeting and promotion and advertising plan. It is difficult because it is out of reach regardless of the quality of the methods adopted to predict the future with certainty. The parameters which intervenient are numerous, complex and often unquantifiable.

Recently, the combined intelligence technique using Artificial Neural Networks (ANNs), fuzzy logic, Particle Swarm Optimization (PSO) and Genetic Algorithms (GAs) has been demonstrated to be an innovative forecasting approach. Since most sales data are non-linear in relation and complex, many studies tend to apply Hybrid models to time-series forecasting. A combination of neural networks and fuzzy systems (Kuo and Chen, 2004) is used to effectively deal with the marketing problem.

Many researchers conclude that the application of BPN is an effective method as a forecasting system and can also be used to find the key factors for enterprisers to improve their logistics management level. Zhang and Chang (2009) utilized Back Propagation Neural

Corresponding Author: Attarius Hichama, Laboratory LIST, Group Research in Computing and Telecommunications (ERIT) FST, Abdelmalek Essaadi University, BP: 416, Old Airport Road, Tangier, Morocco Tel: +212 672 23 72 13; +212 539 99 46 24

networks (BPN) in order to forecast safety stock. Zhang and Huang (2010) used BPN for Sales Forecasting Based on ERP System. They found out that BPN can be used as an accurate sales forecasting system.

The rate of convergence of the traditional back propagation networks is very slow because it's dependent upon the choice of value of the learning rate parameter. However, the experimental results (Iranmanesh and Mahdavi, 2009) showed that the use of an adaptive learning rate parameter during the training process can lead to much better results than the traditional neural network model (BPN).

Many papers indicate that the system which uses the hybridization of fuzzy logic and neural networks can more accurately perform than the conventional statistical method and single ANN. Kuo and Xue (1999) proposed a Fuzzy Neural Network (FNN) as a model for sales forecasting. They utilized fuzzy logic to extract the expert's fuzzy knowledge. Chen (2003) used a model for wafer fab prediction based on a fuzzy back Preprint submitted to Elsevier 24 décembre 2012 Propagation Network (FBPN). The proposed system is constructed to incorporate production control expert judgments in enhancing the performance of an existing crisp back propagation network. The results showed the performance of the FBPN was better than that of the BPN. Efendigil *et al.* (2009) utilized a forecasting system based on artificial neural networks ANNs and Adaptive Network-Based Fuzzy Inference Systems (ANFIS) to predict the fuzzy demand with incomplete information.

A Hybrid Intelligent Clustering Forecasting System was proposed (Oh and Han, 2001). It was based on change point detection and artificial neural networks. The basic concept of proposed model is to obtain significant intervals by change point detection. They found out that the proposed model is more accurate and convergent than the traditional neural network model (BPN).

Recently, some researchers have shown that the use of the hybridization between fuzzy logic and GAs leading to Genetic Fuzzy Systems (GFSs) (Cordon *et al.*, 2001) has more accurate and efficient results than the traditional intelligent systems. Orriols-Puig *et al.* (2009) and Martinez-Lopez and Casillas (2009), utilized GFS in various case Management. They have all obtained good results.

This article proposes a new hybrid system based on genetic fuzzy clustering using a Genetic Fuzzy System (GFS) (Cordon and Herrera, 1997) and Back propagation Neural Networks with adaptive learning rate (GFCBPN) for sales forecasting in Printed Circuit Board (PCB) industry.

The data test used in this study was collected from sales forecasting case study, called Printed Circuit Board (PCB) industry in Taiwan, which has been frequently used by the other authors as a case study. The total number of training samples was collected from January 1999 to December 2002 while the total number of testing samples was from January 2003 to December 2003.

1.1. PCB Sales Forecasting

Due to the important role of Printed Circuit Board (PCB) industry in Taiwan's economy, there are several studies in the literature which have considered PCB sales forecasting as the case study (**Table 1**).

Chang and Lai (2005) used Back Propagation Neural networks (BPN) trained by a genetic algorithm (ENN) to estimate demand production of Printed Circuit Board (PCB). The experimental results show that the performance of ENN is greater than BPN.

Chang *et al.* (2006) used a Fuzzy Back Propagation Network (FBPN) for sales forecasting. The opinions of sales managers about the importance of each input, were converted into pre-specified fuzzy numbers to be integrated into a proposed system. They concluded that FBPN approach outperforms other traditional methods such as Grey Forecasting, Multiple Regression Analysis and back propagation networks.

Chang and Wang (2006) proposed a fusion of SOM, ANNs, GAs and FRBS for PCB sales forecasting. They found that performance of the model was superior to previous methods that proposed for PCB sales forecasting.

Chang *et al.* (2007) developed a Weighted Evolving Fuzzy Neural Network (WEFuNN) model for PCB sales forecasting. The proposed model was based on combination of sales key factors selected using GRA. The experimental results that this hybrid system is better than previous hybrid models.

Chang and Liu (2008) developed a hybrid model based on fusion of Cased-Based Reasoning (CBR) and fuzzy multicriteria decision making. The experimental results showed that FCBR model is superior to traditional statistical models and BPN.

Chang *et al.* (2009) developed a K-means clustering and Fuzzy Neural Network (FNN) to estimate the future sales of PCB. They used K-means for clustering data in different clusters to be fed into independent FNN models. The experimental results show that the proposed approach outperforms other traditional forecasting models, such as, BPN, ANFIS and FNN.

Table 1. Summarizes various methods developed for PBC sales forecasting

Authors	Year	Methods	Accuracy	
			Maape	RMSE
Chang and Lai	2005	GA + ANN	3,13	NA
Chang and Wang	2006	Fuzzy logic + ANN	3,09	NA
Chang <i>et al.</i>	2006	SOM+ANN+GA+FRBS	2,16	21,346
Chang <i>et al.</i>	2007	WEFuNN	2,11	24,909
Chang and Liu	2008	FCBR	4,82	43,385
Chang <i>et al.</i>	2009	K-means clustering+FNN	2,19	20,287
Hadavandi <i>et al.</i>	2011	K-means clustering+KGFS	1,46	19,354

Esmaeil *et al.* (2011) proposed a novel sales forecasting approach by the integration of Genetic Fuzzy Systems (GFS) and data clustering to construct a sales forecasting expert system. They use GFS to extract the whole knowledge base of the fuzzy system for sales forecasting problems. Experimental results show that the proposed approach outperforms the other previous approaches.

2. MATERIALS AND METHODS

2.1. Development of the GFCBPN Model

The proposed architecture consists of three stages as shown in **Fig. 1**: (1) all normalized records of data are categorized into K clusters by using the fuzzy c -means model; (2) the fuzzy distances from all records data (x_i) to different cluster centers (c_j) founded by fuzzy C -means will be introduced into independent Membership Cluster Genetic Fuzzy Systems (MCGFS); (3) For each cluster, we train a parallel BP networks with a learning rate adapted according to the level of cluster membership of each record of training data set.

2.2. Data Preprocessing Stage

Historical data of an electronic company in Taiwan is used to choose the key variables (K_1, K_2, K_3) (**Table 2**) that are to be considered in the forecasted model. Monthly sales amount of Printed Circuit Board (PCB) is considered as a case of the forecasting model which has been used as the case in different studies.

2.3. Winter's Exponential Smoothing

In order to take the effects of seasonality and trend into consideration, Winter's Exponential Smoothing is used to preliminarily forecast the quantity of PCB production. For time serial data, Winter's Exponential Smoothing is used to preprocess all the historical data and use them to predict the production demand (**Fig. 2**), which

will be entered into the proposed hybrid model as input variable (K_4) (**Table 2**). Similar to the previous researches, we assume $\alpha = 0.1$, $\beta = 0.1$ and $\gamma = 0.9$.

2.4. Data Normalization

The input values (K_1, K_2, K_3, K_4) will be ranged in the interval $[0.1, 0.9]$ to meet property of neural networks. The normalized Equation 1 is as follows:

$$N_i = 0.1 + 0.8 * (K_i - \min(K_i)) / (\max(K_i) - \min(K_i)) \quad (1)$$

where, K_i presents a key variable, N_i presents normalized input (**Table 2**), $\max(K_i)$ and $\min(K_i)$ represent maximum and minimum of the key variables, respectively.

2.5. Fuzzy C-Means Clustering

In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. In Fuzzy C-Means (FCM) (developed by (Dunn, 1973) and improved by (Bezdek, 1981), data elements can belong to more than one cluster and associated with each element is a set of membership levels. It is based on minimization of the following objective function:

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2, 1 \leq m < \infty$$

where, u_{ij} is the degree of membership of x_i in the cluster j , x_i is the i^{th} of measured data and c_j is the center of the j^{th} cluster. The algorithm is composed of the following steps:

- Step 1: Initialize randomly the degrees of membership matrix $U = [u_{ij}]$, $U^{(0)}$
- Step 2: Calculate the centroid for each cluster $C^{(k)} = [c_j]$ with $U^{(k)}$:

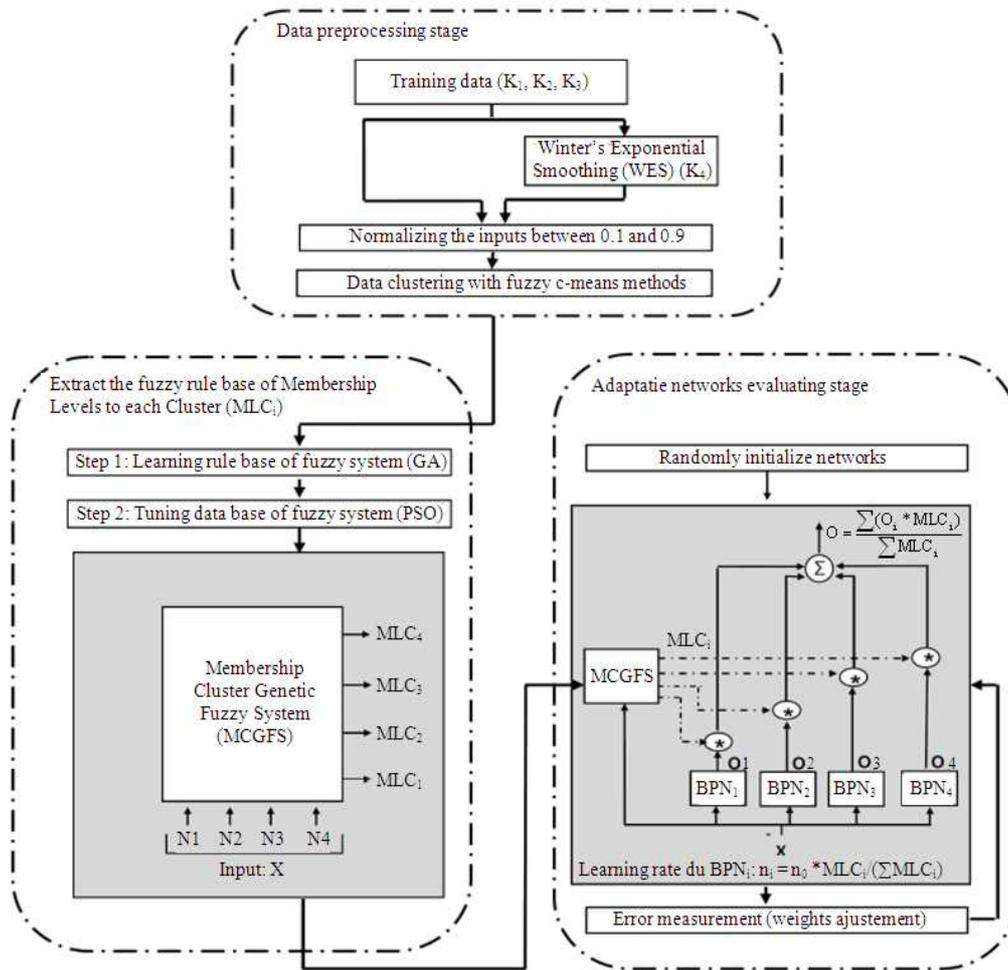


Fig. 1. The architecture of GFCBPN model

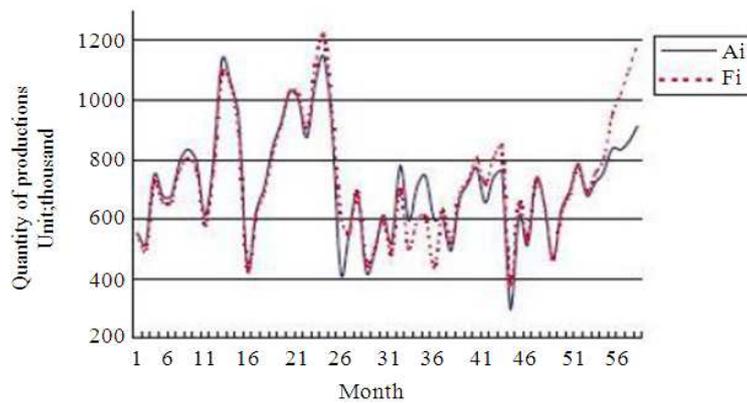


Fig. 2. Comparison of forecasting results of winter's exponential smoothing to the real number

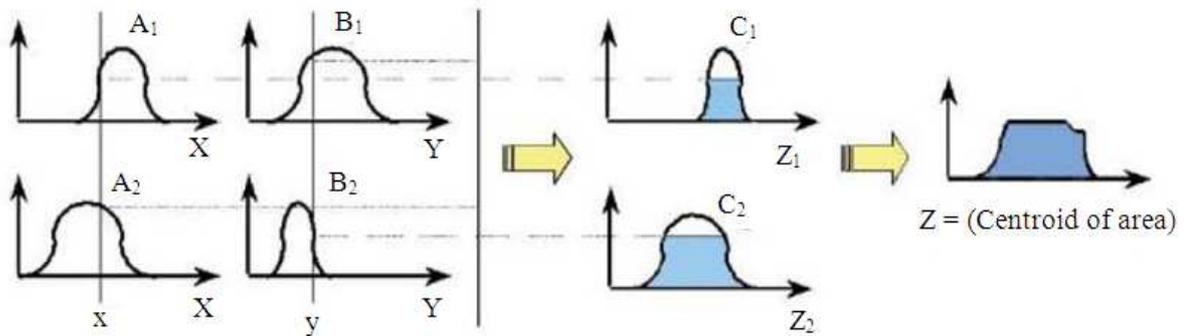


Fig. 3. Fuzzy inference system

Table 2. Description of input forecasting model

Input	Description
K ₁	Consumer price index
N ₁	Normalized Consumer price index
K ₂	Liquid crystal element demand
N ₂	Normalized Liquid crystal element demand
K ₃	PCB total production value
N ₃	Normalized PCB total production value
K ₄	Preprocessed historical data (WES)
N ₄	Normalized preprocessed historical data (WES)
Y'	Actual historical monthly PCB sales Data
Y	Normalized Actual historical monthly PCB sales data

Table 3. Comparison of different clustering algorithms in total distance

Clustering groups	Fuzzy c-means total distance
Clustering 2	Groups 23.7904
Clustering 3	Groups 20.2777
Clustering 4	Groups 18.1477

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m}$$

Step 3: For each point, update its coefficients of being in the clusters (U^(k), U^(k+1)):

$$u_{ij} = \frac{1}{\sum_{i=1}^c \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{2/(m-1)}}$$

Step 4: If $\|U^{(k+1)} - U^{(k)}\| < \epsilon$, $0 < \epsilon < 1$. Then STOP; otherwise return to step 2.

This procedure converges to a local minimum or a saddle point of J_m. According to (Bezdek, 1981), the

appreciated parameter combination of two factors (m and ε) is m = 2 and ε = 0.5.

Using fuzzy c-means, Table 3 shows that the use of four cluster is the best among all different clustering numbers.

2.6. Extract the Fuzzy IF-THEN Rules of Membership Levels to Each Cluster

The degree of membership to a cluster is related inversely to the distance from data record to the cluster center. Thus, the degree of membership to a cluster has a strong dependency to the position of the cluster center which has to move each time a new data record is added. To get around this dependency, we use a kind of Genetic Fuzzy Systems (MCGFS) to extract the fuzzy rules which define the distance between the data records and cluster. Therefore, in adding a new record data, using the fuzzy rules generated we can estimate the distances between data record and cluster centers and to know its degree of membership to each clusters.

In recent years, fuzzy system, used for several complex problems, has become a popular research topic. The fundamental methodologies of fuzzy systems are to store the available knowledge in the form of fuzzy linguistic IF-THEN rules (Fig. 3). It is composed of The Rule Base (RB), constituted by the collection of rules in their symbolic forms and the Data Base (DB), which contains the linguistic term sets and the membership functions defining their meanings (Casillas et al., 2004).

Obviously, it is difficult for human experts to express their knowledge in the form of fuzzy IF-THEN rules.

To cope with this problem, several approaches for automatically extracting fuzzy rules from historical data have been proposed. In this sense, Intelligent

Systems, like Genetic Algorithms (GA) (Casillas *et al.*, 2004; Cordon and Herrera, 1997), or Particle Swarm Optimization (PSO) (Esmine, 2007) have been demonstrated to be a powerful tool to perform tasks such as generation of fuzzy IF-THEN rules. These approaches can be given the general name of Genetic Fuzzy Systems (GFS) (Cordon and Herrera, 1997).

This stage uses a new kind of Genetic Fuzzy System (GFS), called Membership Cluster Genetic Fuzzy System (MCGFS), to extract fuzzy IF-THEN rules of the degree of belonging (Membership levels) to clusters. For each cluster, MCGFS returns the two best fuzzy linguistic rules which present the distance between data records and cluster centers. The proposed MCGFS consists of two general steps **Fig. 4**.

The derived RB is the best individual of the last population:

Step 1: A Genetic Algorithm (GA) will be used to Genetic learn the Fuzzy Rule Based Systems (FRBS). Extract for each cluster, using Genetic Algorithm (GA), the two best fuzzy Rule Bases (RB) of distance between training records and cluster center.

Step 2: Particle Swarm Optimization (PSO) will be used to tune data base of fuzzy system. This proposed process, using Particle Swarm Optimization method (PSO), modifies the shapes of the membership functions of preliminary two best fuzzy rule bases (RB) to improve the precision of results.

2.7. Genetic Rule Base Learning Process for FRBS (GA)

The main aim of this sub stage is to extract, for each cluster K , the two best fuzzy Rule Bases (RB) of the distances between training records x_i and cluster center (c_k). The Data Base (DB) definition constituted by uniform fuzzy partitions, with triangular membership functions crossing a height 0.5 for input and output variables linguistic terms, is considered. Each variable is defined by a fuzzy linguistic term (Binary numbers corresponding), such as, non (00), small (01), medium (10) and large (11). For each cluster, all distance from each cluster center c_k to each record data x_i will be introduced into independent (GFS) models by cluster, with the ability of fuzzy rule bases extraction. For each cluster, the derived Rule

Base (RB) is the two of the best chromosomes of the last population. This stage consists of the following steps.

Step 1: Encoding of chromosomes.

Each two genes present the triangular membership functions for input or output variables linguistic terms. A chromosome is constructed from a series of genes. A sample coded combination of fuzzy rule base as chromosomes with four inputs as well as an output variable which present the fuzzy distance from training data x_i to the k^{th} cluster is shown in **Fig. 5**.

Step 2: Generating the initial values.

Initial binary values of linguistic variables (chromosomes) are randomly generated. The initial chromosomes generated form the first population (N_{pop}).

Step 3: Calculating the fitness values.

In this step, the Mean Squared Error (MSE) is used as the objective function to evaluate the deviation of the training data, which is computed as:

$$\text{MSE}(C_j^k) = \frac{1}{N} \sum_{i=1}^N (\text{Dist}_i^k - \text{Out}_i^k)$$

where, Dist_i^k is the actual distance between the i^{th} training element x_i and k^{th} cluster center and Out_i^k , obtained from the FRBS using the RB coded in j^{th} chromosome (C_j^k), is the output distance between the i^{th} training element x_i and k^{th} cluster center and N is the number of training data.

Step 4: Reproduction and selection.

The roulette wheel selection (Goldberg, 1989) is applied in this stage. The two best individuals of each generation were copied without changes in the next generation. We use a binary tournament selection scheme for the selection procedure to generate ($N_{\text{pop}} - 2$). In binary tournament selection, two individuals are randomly selected; the better of the two is selected as a parent.

Step 5: Crossover.

After the parameter design, two-point crossover method (Goldberg, 1989) is applied in this step.

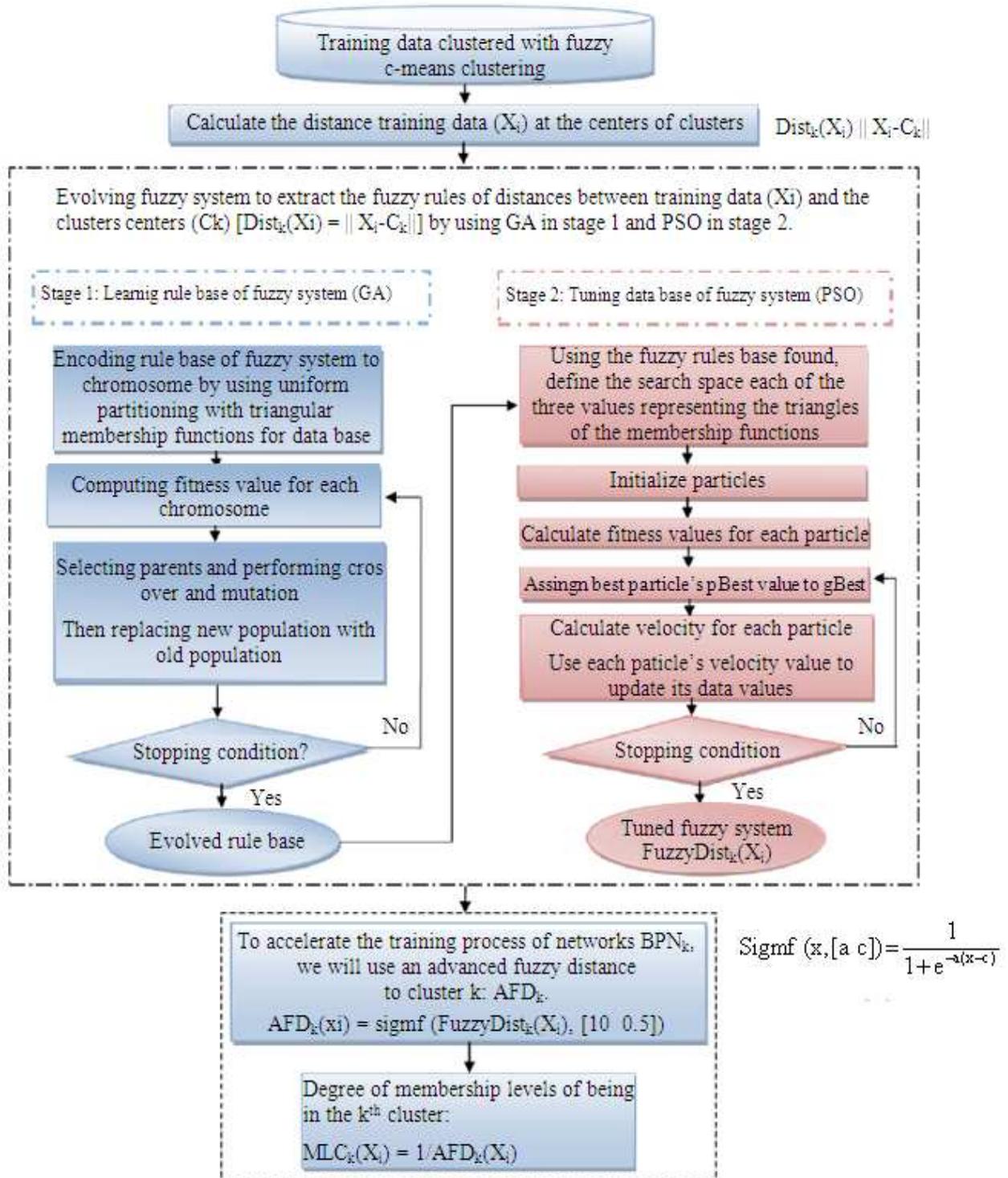


Fig. 4. Architecture of MCGFS

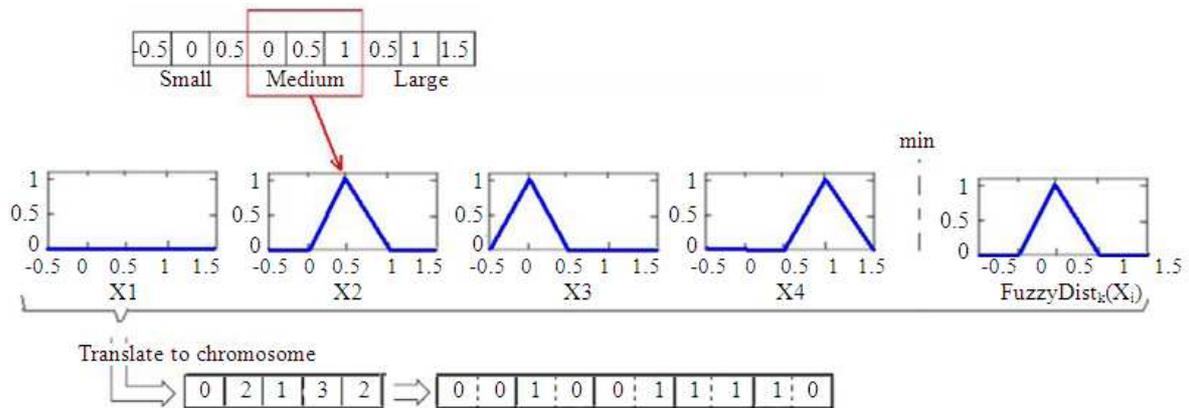


Fig. 5. Coding combination of fuzzy rule base as chromosomes

Step 6: Mutation.

After the parameter design, one-point mutation method (Goldberg, 1989) is applied in this step.

Step 7: Replacement.

The new population generated by the previous steps updates the old population.

Step 8: Stopping criteria.

If the number of generations is equal to the maximum generation number then stop, otherwise go to step 3.

2.8. Tuning Process of Fuzzy Rule Bases (PSO)

In order to improve the precision of the two best fuzzy rule bases of each cluster returned by the above generation method, This substage applies a tuning process of fuzzy rule-based similar to the genetic tuning process that was proposed by (Cordon and Herrera, 1997). The proposed tuning process uses the Particle Swarm Optimization method (PSO) to modify the shapes of the membership functions of the preliminary two best Rule Bases (RB) of each Cluster.

The Particle Swarm Optimization Algorithm (PSO) is a population based optimization method that finds the optimal solution using a population of particles (Eberhart and Kennedy, 1995). Every swarm of PSO is a solution in the solution space. PSO is basically developed through simulation of bird flocking. PSO can yield faster convergence when compared to Genetic Algorithm (GA), because of the balance between exploration and exploitation in the search space (Sivanandam and Visalakshi, 2009).

For each cluster fuzzy Rule Base (RB), we apply Particle Swarm Optimization Algorithm (PSO) to adjust the parameters (shapes) of membership functions to

improve the accuracy of estimates of the distances between training records and cluster center. The proposed tuning process is presented as follows:

Step 1: Defining of the search space.

The PSO algorithm works by having a search space (called a swarm) of candidate solutions (called particles). In our case, the search space is the set of all possible three values representing the triangles of the membership functions. The dimensionality of the search space is 15. Each particle is represented by:

$$P_i = (a_i^1, b_i^1, c_i^1, a_i^2, b_i^2, c_i^2, a_i^3, b_i^3, c_i^3, a_i^4, b_i^4, c_i^4, o_{1,i}^j, o_{2,i}^j, o_{3,i}^j)$$

where, $[a_i^k, b_i^k, c_i^k]$ are the three parameters which define the input triangle fuzzy membership function of the K^{th} variable (X_k) and $[o_{1,i}^j, b_{2,i}^j, c_{3,i}^j]$ are the three parameters which define the output triangle fuzzy membership function of fuzzy distance between cluster center c_j and normalized record data $X(X_1, X_2, X_3, X_4)$.

Step 2: Generating the initial population.

Initialization the positions of the particles: $p_i = (p_i^1 \dots p_i^{15})$ Where p_i^k is initialized with a uniformly distributed random: $p_i^k \sim U(p_i^{k,l}, p_i^{k,r})$ Where $p_i^{k,l}$ and $p_i^{k,r}$ are the lower and upper boundaries of the k^{th} dimension of the search-space. If $t \bmod 3 = 1$, then p_i^1 is the left value of the support of a triangular fuzzy number. The triangular fuzzy number is defined by the three parameters $(p_i^1, p_i^{t+1}, p_i^{t+2})$ and the intervals of performance are as follows:

$$\begin{aligned}
 p_i^t &\in [p_i^{t,l}, p_i^{t,r}] = [p_i^t - \frac{1}{2}, p_i^t + \frac{1}{2}] \\
 p_i^{t+1} &\in [p_i^{t+1,l}, p_i^{t+1,r}] = [p_i^{t+1} - \frac{1}{2}, p_i^{t+1} + \frac{1}{2}] \\
 p_i^{t+2} &\in [p_i^{t+2,l}, p_i^{t+2,r}] = [p_i^{t+2} - \frac{1}{2}, p_i^{t+2} + \frac{1}{2}]
 \end{aligned}$$

Step 3: Calculate fitness values for each particle.

The fitness value for each particle is calculated utilizing MSE over a training data set, which is computed as:

$$\text{MSE}(P_i) = \frac{1}{N} \sum_{i=1}^N (\text{Dist}_i^k - \text{Out}_i^k)$$

where, Dist_i^k is the actual distance between the i^{th} training element (x_i) and k^{th} cluster center (c_k) and Out_i^k , obtained from fuzzy rule coded in the particle P_i , is the output distance between the i^{th} training element x_i and k^{th} cluster center and N is the number of training data.

Step 4: Assign best particle's P_i^{best} value to g^{best} .

Compare each particle's P_i fitness evaluation with its P_i^{best} . If the current value is better than P_i^{best} , set the P_i^{best} value to the current value P_i . Compare the population's fitness evaluation with the population's overall previous best (g^{best}). If the current value is better than g^{best} , reset the g^{best} location to the current particle's location.

Step 5: Calculate velocity for each particle.

The velocity of each of the particles (P_i) for the next generation $t+1$ are updated as:

$$V_i^{t+1} = V_i^t + C_1 \cdot r \text{ and } () \cdot \|P_i^{\text{best}} - P_i\| + C_2 \cdot r \text{ and } () \cdot \|g^{\text{best}} - P_i\|$$

where, constants c_1 and c_2 represent the weights of the stochastic acceleration terms that draw each particle i towards P_i^{best} and g^{best} positions. In this research, we assume $C_1 = 2$ and $C_2 = 2$.

Step 6: Update particles position.

The particles are moved to their new positions according to:

$$P_i^{t+1} = P_i^t + V_i^{t+1}$$

Until a termination criterion is met (number of iterations performed, or adequate fitness reached), go to step 3.

2.9. The Degree of Membership Levels (MLC_k)

Using the two previous stages, we get six fuzzy rules as Results (Fig. 6). Each pair of rules presents the distance between records data (X_i) and a cluster center (c_j).

In this stage, we will use the sigmoid function (Fig. 7) to improve the precision of results and to accelerate the training process of neural networks. Then, the advanced fuzzy distance to cluster k (AF D_k) will be presented as follow:

$$\text{AF } D_k (X_i) = \text{sigmf} (\text{FuzzyDist}_k (X_i), [50, 0.5])$$

$$\text{sigmf} (x, [a, c]) = \frac{1}{1 + e^{-a(x-c)}}$$

The degree of membership levels of belonging of a record X_i to k^{th} cluster ($\text{MLC}_k (X_i)$) is related inversely to the distance from records data X_i to the cluster center c_k (AF $D_k (X_i)$):

$$\text{MLC}_k (X_i) = \frac{1}{\text{AF } D_k (X_i)}$$

Thus, we can construct a new training sample (X_i , $\text{MLC}_1(X_i)$, $\text{MLC}_2(X_i)$, $\text{MLC}_3(X_i)$, $\text{MLC}_4(X_i)$) for the adaptive neural networks evaluating (Fig. 1).

2.10. Adaptive Neural Networks Evaluating Stage

The Artificial Neural Networks (ANNs) concept is originated from the biological science (neurons in an organism). Its components are connected according some pattern of connectivity, associated with different weights. The weight of a neural connection is updated by learning. The ANNs possess the ability to identify nonlinear patterns by learning from the data set. The Back Propagation (BP) training algorithms are probably the most popular ones. The structure of BP neural networks consists of an input layer, a hidden layer, as well as an output layer. Each layer contains I ; J and L nodes denoted. The w_{ij} is denoted as numerical weights between input and hidden layers and so is w_{jl} between hidden and output layers as shown in Fig. 8.

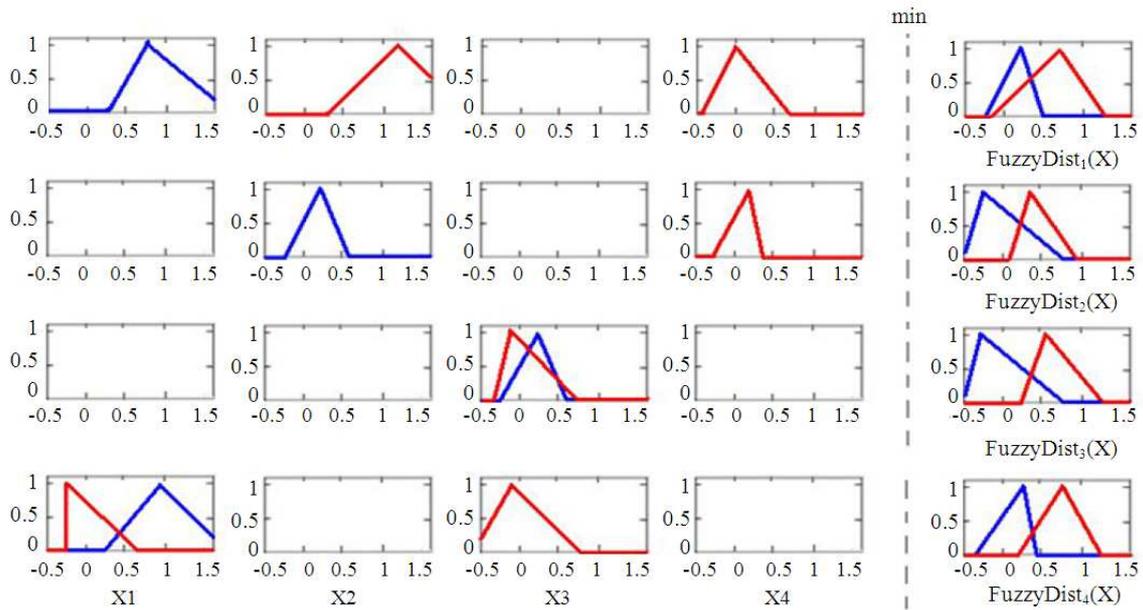


Fig. 6. The tuned membership functions of input and output variables for clusters GFS

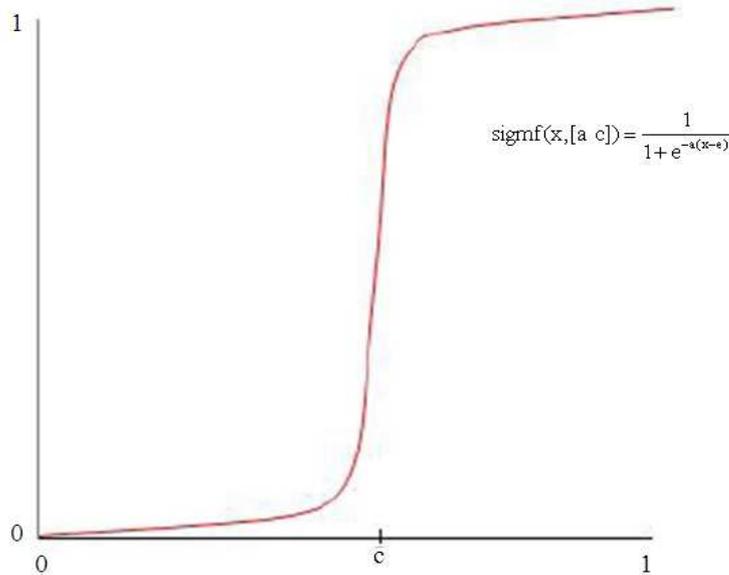


Fig. 7. Sigmoid function, a = 50 and c = 0,5

In this stage, we propose an adaptive neural networks evaluating system which consists of four neural networks. Each cluster K is associated with the K^{th} BP network. For each cluster, the training sample will be fed into a parallel Back Propagation Networks (BPN) with a learning rate adapted according to the level of clusters

membership (MLC_k) of each records of training data set. The structure of the proposed system is shown in **Fig. 1**.

The Adaptive neural networks learning algorithm is composed of two procedures: (a) a feed forward step and (b) a back-propagation weight training step. These two separate procedures will be explained in details as follows:

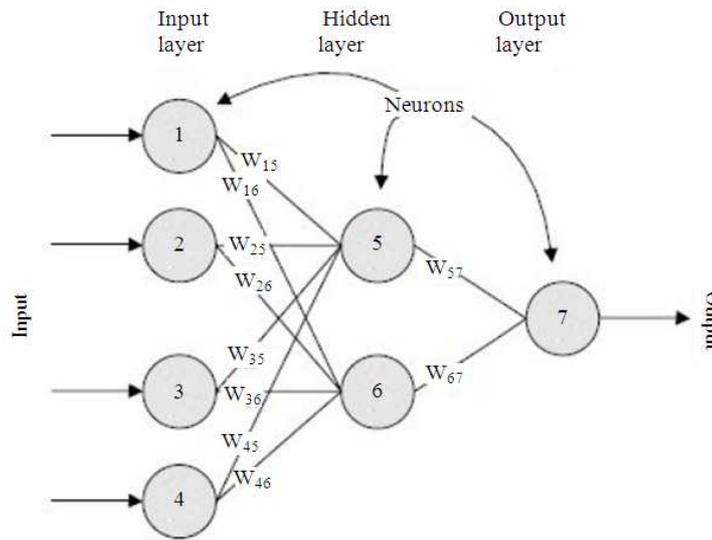


Fig. 8. The structure of back-propagation neural network of this research

Step 1: All BP networks are initialized with the same random weights.

Step 2: Feed forward.

For each BPN_k (associate to the K^{th} cluster), we assume that each input factor in the input layer is denoted by x_i . y_j^k and o_i^k represent the output in the hidden layer and the output layer, respectively and y_j^k and o_i^k can be expressed as follows:

$$Y_j^k = f(X_j) = f(w_{oj}^k + \sum_{i=0}^I w_{ij}^k x_i)$$

And:

$$o_i^k = f(X_i^k) = f(w_{oi}^k + \sum_{j=1}^J w_{ji}^k y_j^k)$$

where, the w_{oj}^k and w_{oi}^k are the bias weights for setting threshold values, f is the activation function used in both hidden and output layers and x_j^k and y_i^k are the temporarily computing results before applying activation function f . In this study, a sigmoid function (or logistic function) is selected as the activation function. Therefore, the actual outputs y_j^k and o_i^k in hidden and output layers, respectively, can also be written as:

$$y_j^k = f(X_j^k) = \frac{1}{1 + e^{-X_j^k}}$$

And:

$$o_i^k = f(Y_i^k) = \frac{1}{1 + e^{-Y_i^k}}$$

The activation function f introduces the non-linear effect to the network and maps the result of computation to a domain (0, 1). In our case, the sigmoid function is used as the activation function:

$$f(t) = \frac{1}{1 + e^{-t}}, f' = f(1 - f)$$

The global output of the adaptive neural networks is calculated as:

$$o_1 = \frac{\sum_{k=1}^4 (MLC_k(X_j) \times o_i^k)}{\sum_{k=1}^4 MLC_k(X_j)}$$

As shown above, the effect of the output o_i^k on the global output o_1 is both strongly and positively related to the membership level (MLC_k) of data record X_i to k^{th} cluster.

Step 3: Back-propagation weight training. The error function is defined as:

$$E = \frac{1}{2} \sum_{i=1}^L e_i^2 = \frac{1}{2} \sum_{i=1}^L (t_i - o_i)^2$$

where, t_k is a predefined network output (or desired output or target value) and e_k is the error in each output node.

The goal is to minimize E so that the weight in each link is accordingly adjusted and the final output can match the desired output. The learning speed can be improved by introducing the momentum term α . Usually, η falls in the range [0, 1]. For the iteration n and for BPN_k (associated to k^{th} cluster), the adaptive learning rate in BPN_k and the variation of weights Δw_k can be expressed as:

$$\eta_k = \frac{MLC_k(X_j)}{\sum_{k=1}^4 MLC_k(X_j)} \times \eta$$

$$\Delta w_k(n+1) = \eta_k \times \Delta w_k(n) + \alpha \times \frac{\delta E}{\delta w_k(n)}$$

As shown above, we can conclude that the variation of the BPN_k network weights (w_{oj}^k and w_{oi}^k) are more important as long as the the Membership Level (MLC_k) of data record X_j to k^{th} cluster is high. If the value of Membership Level (MLC_k) of data record X_j to k^{th} cluster is close to zero then the changes in BPN_k network weights are very minimal.

The configuration of the proposed BPN is established as follows:

- Number of neurons in the input layer: I = 4
- Number of neurons in the output layer: L = 1
- Single hidden layer
- Number of neurons in the hidden layer: J = 2
- Network-learning rule: delta rule
- Transformation function: sigmoid function
- Learning rate: $\eta = 0.1$
- Momentum constant: = 0.02
- Learning times: 15000

3. RESULTS

3.1. Constructing GFCBPN Sales Forecasting ES

Our proposed system (GFCBPN) has three stages: first, all normalized records of data are categorized into k clusters by using the fuzzy c-means model. Secondly, using a Genetic Fuzzy System (GFS), extract the fuzzy rules of membership levels to each cluster. Finally, for each cluster, we train a parallel BP networks with a learning rate adapted according to the level of cluster membership of each record of training data set.

Table 4. The forecasted results by GFCBPN method

Month	Forecasted values	Actual values
2003/1	638,749	649,066
2003/2	443,585	466,750
2003/3	633,837	633,615
2003/4	675,897	693,946
2003/5	747,220	785,838
2003/6	686,641	679,312
2003/7	724,807	723,914
2003/8	754,198	757,490
2003/9	826,618	836,846
2003/10	849,560	833,012
2003/11	874,510	860,892
2003/12	895,338	912,182

Table 5. The forecasted results by KGFS methods

Month	Actual values	KGFS forecasts
2003/1	649,066	645649.2
2003/2	466,750	462041.4
2003/3	633,615	636362.1
2003/4	693,946	701704.2
2003/5	785,838	799244.6
2003/6	679,312	678026.7
2003/7	723,914	730172.5
2003/8	757,490	755321.4
2003/9	836,846	848193.6
2003/10	833,012	852101.9
2003/11	860,892	849898.4
2003/12	912,182	852563.3

Table 6. The forecasted results by FNN methods

Month	Forecasted values (FNN)	Actual values
2003/1	584,901.9	649,066
2003/2	483,872.3	466,750
2003/3	713,874.6	633,615
2003/4	711,356.1	693,946
2003/5	769,881.6	785,838
2003/6	684,634.5	679,312
2003/7	721,192.4	723,914
2003/8	770,609	757,490
2003/9	817,423.4	836,846
2003/10	851,827	833,012
2003/11	884,484.1	860,892
2003/12	912,129.1	912,182

The proposed GFCBPN system was applied as case to forecast the sales data of the PCB. The results are presented in **Table 4**.

Using Membership Cluster Genetic Fuzzy System MCGFS (**Fig. 4**) of cluster 1, cluster 2, cluster 3 and cluster 4, the fuzzy rules of membership levels to each cluster will be extracted and the results are summarized in **Fig. 6**.

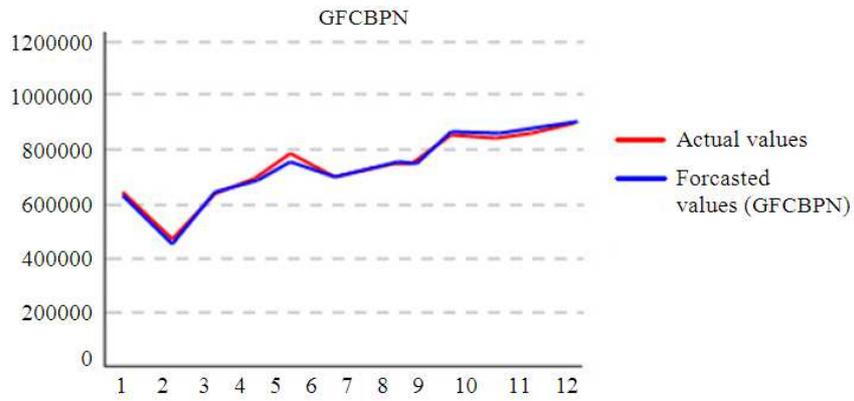


Fig. 9. The MAPE of GFCBPN

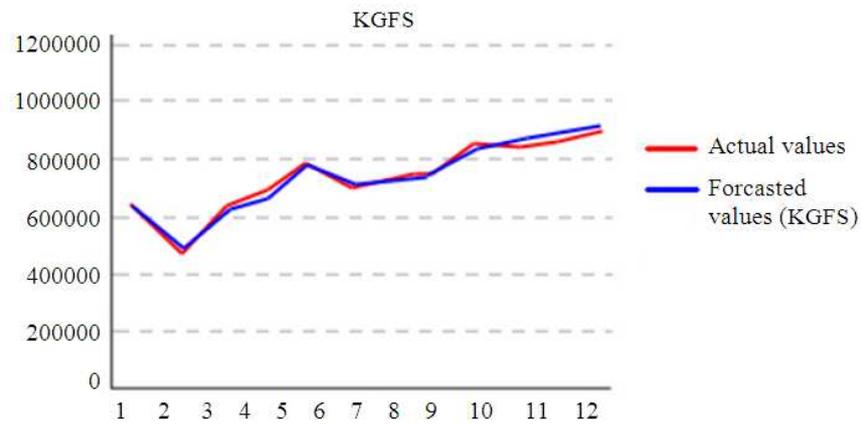


Fig. 10. The MAPE of KGFS

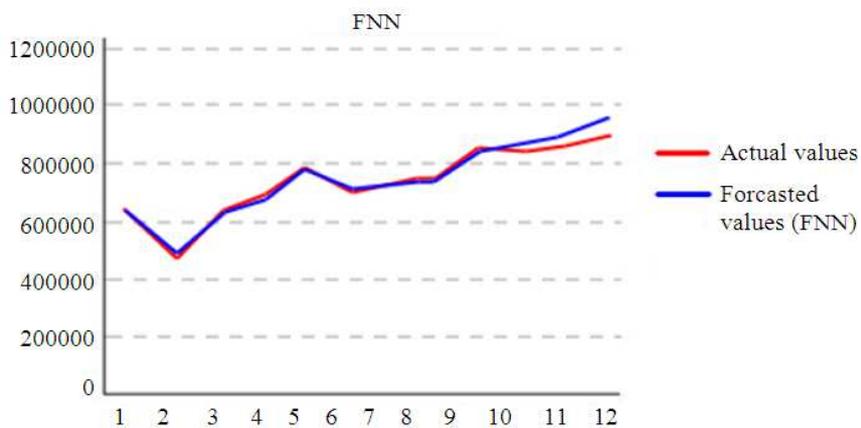


Fig. 11. The MAPE of FNN

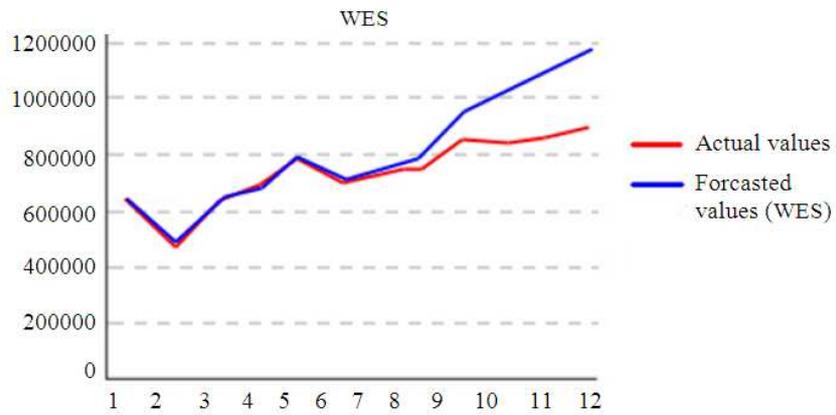


Fig. 12. The MAPE of WES

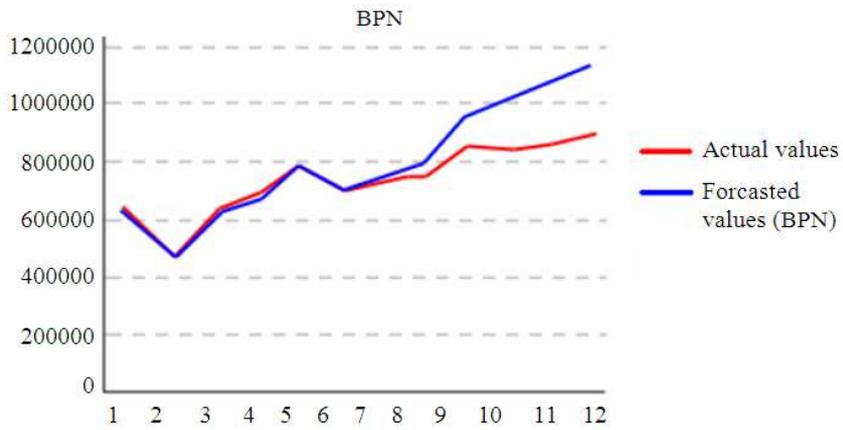


Fig. 13. The MAPE of BPN

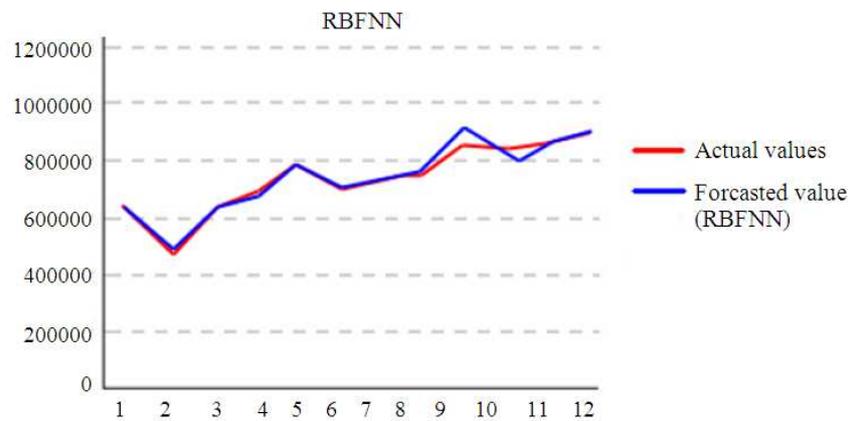


Fig. 14. The MAPE of RBFNN

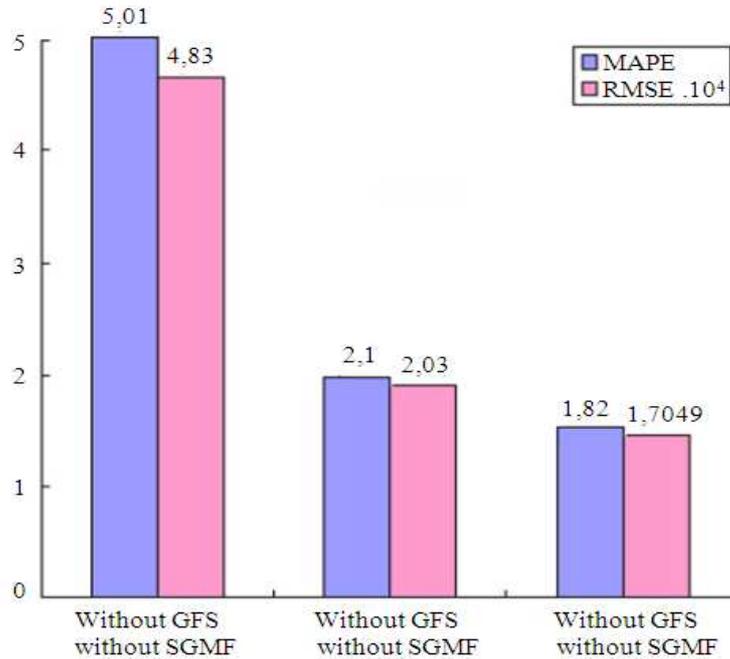


Fig. 15. The performance improvement of GFCBPN after using GSF and sigmoid function

Table 7. The forecasted results by Winter’s method

Month	Forecasted values	Actual values
2003/1	649,700.0	649,066
2003/2	465,219.0	466,750
2003/3	623,542.0	633,615
2003/4	681,530.0	693,946
2003/5	783,733.0	785,838
2003/6	693,935.0	679,312
2003/7	753,675.0	723,914
2003/8	800,210.0	757,490
2003/9	949,143.0	836,846
2003/10	1,019,900.0	833,012
2003/11	1,100,546.0	860,892
2003/12	1,189,945.0	912,182

Table 9. The forecasted results by RBFNN method

Month	Forecasted values	Actual values
2003/1	574,952	649,066
2003/2	465,762	466,750
2003/3	634,449	633,615
2003/4	693,809	693,946
2003/5	785,878	785,838
2003/6	678,838	679,312
2003/7	723,553	723,914
2003/8	759,976	757,490
2003/9	875,283	836,846
2003/10	800,874	833,012
2003/11	860,366	860,892
2003/12	905,347	912,182

Table 8. The forecasted results by BPN’s method

Month	Forecasted values	Actual values
2003/1	622,402.3	649,066
2003/2	456,226	466,750
2003/3	618,346	633,615
2003/4	669,445.5	693,946
2003/5	795,971.6	785,838
2003/6	682,646.4	679,312
2003/7	741,996.5	723,914
2003/8	789,756.8	757,490
2003/9	945,738.1	836,846
2003/10	1,006,899	833,012
2003/11	1,077,823	860,892
2003/12	1,141,621	912,182

3.2. Comparisons of GFCBPN Model with Other Previous Models

Experimental comparison of outputs of GFCBPN with other methods show that the proposed model outperforms the previous approaches (Table 4-9 and Fig. 9-14). We apply two different performance measures called Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE), to compare the GFCBPN model with the previous methods, i.e., KGFS, KFNN, FNN, WES, BPN and RBFNN:

Table 10. History of PCB sales forecasting

Authors	Year	Methods	Accuracy	
			Mape	RMSE
Chang and Lai	2005	GA + ANN	3,13	NA
Chang and Wang	2006	Fuzzy logic + ANN	3,09	NA
Chang <i>et al.</i>	2006	SOM+ANN+GA+FRBS	2,16	21,346
Chang <i>et al.</i>	2007	WEFuNN	2,11	24,909
Chang and Liu	2008	FCBR	4,82	43,385
Chang <i>et al.</i>	2009	K-means clustering+FNN	2,19	20,287
Hadavandi <i>et al.</i>	2011	K-means clustering+KGFS	1,46	19,354
GFCBPN	2012	Fuzzy clustering+GFCBPN	1,85	17,049

$$MAPE = 100 \times \frac{1}{N} \sum_{t=1}^N \frac{|Y_t - P_t|}{Y_t}$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (Y_t - P_t)^2}$$

where, P_t is the expected value for period t , Y_t is the actual value for period t and N is the number of periods.

As shown in **Fig. 15**, the use of MCGFS enhanced by the sigmoid function in the proposed acupuncture (GFCBPN) has better precision results than the use of the fuzzy c-means clustering in the test stage. GFCBPN has made 1,7 as MAPE evaluation and 1820 as RMSE evaluation. Therefore, the forecasting accuracy of GFCBPN out performs the previous approaches regarding MAPE and RMSE evaluations which is summarized in **Table 10**.

4. CONCLUSION

This article proposes a new hybrid system based on genetic fuzzy clustering and Back-propagation Neural Networks with adaptive learning rate (GFCBPN) for sales forecasting.

The experimental results of the proposed approach show that the effectiveness of the GFCBPN outperforms the previous and traditional approaches: WES, BPN, RBFNN, KFNN, FNN, WES, BPN and KGFS. Furthermore, it also demonstrates that our modeling approach (GFCBPN) has properties, such as, fast convergence, high precision, robust and accurate forecasting techniques.

Compared to previous researches which tend to use the classical hard clustering methods (K-means clustering) to divide data set into subgroups in order to reduce the noise and form more homogeneous clusters (Chang *et al.*, 2009), the advantage of our proposed system (GFCBPN) is that it uses a fuzzy clustering (fuzzy c-means

clustering) which permits each data record to belong to each cluster to a certain degree, which allows the clusters to be larger which consequently increases the accuracy of forecasting system results.

Another advantage of our system is that it uses a kind of Genetic Fuzzy System (GFS), called MCGFS, which allows estimation of the degree of membership of each data record to each cluster with no dependencies to the position of the cluster centers.

We applied GFCBPN for sales forecasting in Printed Circuit Board (PCB) as a case study. The results demonstrated the effectiveness and superiority of the GFCBPN compared to the previous approaches regarding MAPE and RMSE evaluations. Other academic researchers and industrial practitioners may find these contributions interesting.

5. REFERENCES

- Bezdek, J.C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. 1st Edn., Plenum Press, New York, pp: 256.
- Casillas, J., O. Cordon, F. Herrera and P. Villar, 2004. A hybrid learning process for the knowledge base of a fuzzy rule-based system. Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, (BS' 04), Perugia, Italy, pp: 2189-2196.
- Chang, P.C. and C.H. Liu, 2008. A TSK type fuzzy rule based system for stock price prediction. Expert Syst. Appli., 34: 135-144. DOI: 10.1016/j.eswa.2006.08.020
- Chang, P.C. and C.Y. Lai, 2005. A hybrid system combining self-organizing maps with case-based reasoning in wholesaler's new-release book forecasting. Expert Syst. Appli., 29: 183-192. DOI: 10.1016/j.eswa.2005.01.018

- Chang, P.C. and Y.W. Wang, 2006. Fuzzy delphi and back-propagation model for sales forecasting in PCB industry. *Expert Syst. Appli.*, 30: 715-726. DOI: 10.1016/j.eswa.2005.07.031
- Chang, P.C., C.H. Liu and C.Y. Fan, 2009. Data clustering and fuzzy neural network for sales forecasting: A case study in printed circuit board industry. *Knowl. Based Syst.*, 22: 344-355. DOI: 10.1016/j.knosys.2009.02.005
- Chang, P.C., C.H. Liu and Y.W. Wang, 2006. A hybrid model by clustering and evolving fuzzy rules for sales decision supports in printed circuit board industry. *Decision Support Syst.*, 42: 1254-1269. DOI: 10.1016/j.dss.2005.10.013
- Chang, P.C., Y.W. Wang and C.H. Liu, 2007. The development of a weighted evolving fuzzy neural network for PCB sales forecasting. *Expert Syst. Appli.*, 32: 86-96. DOI: 10.1016/j.eswa.2005.11.021
- Chen, T., 2003. A fuzzy back propagation network for output time prediction in a wafer fab. *Applied Soft Comput.*, 2: 211-222. DOI: 10.1016/S1568-4946(02)00066-2
- Cordon, O. and F. Herrera, 1997. A three-stage evolutionary process for learning descriptive and approximate fuzzy-logic-controller knowledge bases from examples. *Int. J. Approximate Reason.*, 17: 369-407. DOI: 10.1016/S0888-613X(96)00133-8
- Cordon, O., F. Herrera, F. Hoffmann and L. Magdalena, 2001. *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. 1st Edn., World Scientific, Singapore, ISBN-10: 9810240171, pp: 462.
- Dunn, J.C., 1973. A fuzzy relative of the ISODATA Process and its use in detecting compact well-separated clusters. *J. Cybernetics*, 3: 32-57. DOI: 10.1080/01969727308546046
- Eberhart, R.C. and J. Kennedy, 1995. A new optimizer using particle swarm theory. *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, Oct. 4-6, IEEE Xplore Press, Nagoya, pp: 39-43. DOI: 10.1109/MHS.1995.494215
- Efendigil, T., S. Onu and C. Kahraman, 2009. A decision support system for demand forecasting with artificial neural networks and neuro-fuzzy models: A comparative analysis. *Expert Syst. Appli.*, 36: 6697-6707. DOI: 10.1016/j.eswa.2008.08.058
- Esmail, H., S. Hassan and G. Arash, 2011. An improved sales forecasting approach by the integration of genetic fuzzy systems and data clustering: Case study of printed circuit board. *Expert Syst. Appli.*, 38: 9392-9399. DOI: 10.1016/j.eswa.2011.01.132
- Esmine, A., 2007. Generating fuzzy rules from examples using the particle swarm optimization algorithm. *Proceedings of the 7th International Conference IEEE Hybrid Intelligent Systems*, Sept. 17-19, IEEE Xplore Press, Kaiserlautern, pp: 340-343. DOI: 10.1109/HIS.2007.52
- Goldberg, D.A., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. 1st Edn., Addison-Wesley, Reading, ISBN-10: 0201157675, pp: 412.
- Hadavandi, E., H. Shavandi and G. Arash, 2011. An improved sales forecasting approach by the integration of genetic fuzzy systems and data clustering: Case study of printed circuit board. *Expert Syst. Appli.*, 38: 9392-9399. DOI: 10.1016/j.eswa.2011.01.132
- Iranmanesh, S. and M.A. Mahdavi, 2009. A differential adaptive learning rate method for back-propagation neural networks. *Proceedings of the 10th WSEAS International Conference on Neural Networks*, (NN'09), ACM Press, USA., pp: 30-34.
- Kuo, R.J. and J.A. Chen, 2004. A decision support system for order selection in electronic commerce based on fuzzy neural network supported by real-coded genetic algorithm. *Expert Syst. Appli.*, 26: 141-154. DOI: 10.1016/S0957-4174(03)00115-5
- Kuo, R.J. and K.C. Xue, 1999. Fuzzy neural networks with application to sales forecasting. *Fuzzy Sets Syst.*, 108: 123-143. DOI: 10.1016/S0165-0114(97)00326-6
- Martinez-Lopez, F. and J. Casillas, 2009. Marketing intelligent systems for consumer behaviour modelling by a descriptive induction approach based on genetic fuzzy system. *Indus. Market. Manage.*, 38: 714-731. DOI: 10.1016/j.indmarman.2008.02.003
- Oh, K.J. and I. Han, 2001. An intelligent clustering forecasting system based on change-point detection and artificial neural networks: Application to financial economics. *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Jan. 3-6, IEEE Xplore Press. DOI: 10.1109/HICSS.2001.926304

- Orriols-Puig, A., J. Casillas and F. Martinez-Lopez, 2009. Unsupervised learning of fuzzy association rules for consumer behavior modeling. *Mathware Soft Comput.*, 16: 29-43.
- Sivanandam, S.N. and P. Visalakshi, 2009. Dynamic task scheduling with load balancing using parallel orthogonal particle swarm optimisation. *Int. J. Bio-Inspired Comput.*, 1: 276-286. DOI: 10.1504/IJBIC.2009.024726
- Zhang, M. and Y. Huang, 2010. Sales forecasting based on ERP system through BP neural networks. *Proceedings of the 5th International Conference on Advances in Computation and Intelligence, (ISICA' 10)*, Springer-Verlag Berlin, Heidelberg, pp: 289-296. DOI: 10.1007/978-3-642-16493-4_30
- Zhang, W. and Chang, 2009. A model on forecasting safety stock of erpbased on bpneural network. *Proceedings of the 2008 IEEE ICMIT*.