Chapter 2
Soft Computing Techniques and Their Applications

D.K. Chaturvedi

Abstract The modern science is still striving to develop consciousness-based machine. The forecasting is an intuition-based or consciousness-based problem. It is an important problem for planning, decision-making and designing of an appropriate controller for the systems. The paper deals with the synergism of soft computing techniques mainly artificial neural network, fuzzy logic systems, and genetic algorithms and their applications in forecasting.

Keywords Artificial neural network · Fuzzy systems · Genetic algorithms · Synergism of soft computing techniques · Forecasting

2.1 Introduction

In the last century, enormous industrial and technological developments had taken place. Technology had developed laterally well up to the biggest giant-sized complexes and also to the smallest molecular nano-mechanisms. Thus, having explored to the maxima of the two extreme fields, technology is exploring now vertically to reach the dizzy heights of soft computing, subtle soft computing, and the millennium wonder of reaching the almost unchartered height of evolving consciousness in computers (machines). This presentation makes its small and humble contribution to this new astounding scenario and possibly the greatest of all mechanical wonders, to transfer consciousness of man to machine [1]. Prior to World War II, numerical calculations were done with mechanical calculators. Simulated by military requirements during World War II, the first version modern digital computers began to make their appearance in late 1940s and early 1950s. During that pioneering period, a number of different approaches to digital computer organization and digital computing techniques were investigated. Primarily, as a result of the constraints imposed

D.K. Chaturvedi (✉)
Department of Electrical Engineering, D.E.I. (Deemed University), Dayalbagh, Agra
e-mail: dkc.foe@gmail.com
URL: http://www.works.bepress.com/dk_chaturvedi

A.H. Siddiqi et al. (eds.), *Mathematical Models, Methods and Applications*,
Industrial and Applied Mathematics, DOI 10.1007/978-981-287-973-8_2

by the available electronics technology, the designers of digital computers soon focused their attention on the concept of computer system architecture, which was championed by Dr. John Von Neumann, who first implemented it in the computer constructed for the Institute of Advanced Studies at Princeton. Because of the pervasiveness of the Von Neumann architecture in digital computers, during the 1950s and 1960s, most numerical analysts and other computer users concentrated their efforts on developing algorithms and software packages suitable to these types of computers. In 1960s and 1970s, there were numerous modifications and improvements to computers of the earlier generation. The "bottle neck" of Neumann computers was the memory buffer sizes and speeds on it. In the 1990s, there was a quantum leap in the size of computer memory and speeds. As a result of this, supercomputers have been developed, which could do lakhs of calculations within a fraction of a second. Supercomputers can also do all routine tasks, and it could handle it better with multi-coordination than a human being, and thus reducing a series of simple logical operations. It could store vast information and process the same in a flash. It does not also suffer from the human moods and many vagaries of mind.

But, the supercomputers cannot infer or acquire any knowledge from its information contents. It cannot think sensibly and talk intelligently. It could not recognize a person or could not relate his family background. On the other hand, as human beings, we continuously evolve our value judgment about the information we receive and instinctively process them. Our judgment is based on our feelings, tastes, knowledge, and experience. But computers are incapable of such judgments. A computer can be programmed (instructed), i.e., to generate poetry or music, but it cannot appraise or judge its quality.

Hence, there is a genuine and compulsory need for some other logic, which can handle such real-life scenario. In 1965, Prof. Lofti A. Zadeh at the University of California introduced an identification tool by which this degree of truth can be handled by fuzzy set theoretic approach. With the invention of fuzzy chips in 1980s fuzzy logic received a great boost in the industry.

Now in this twenty-first century fuzzy logic, artificial neural network (ANN), and evolutionary algorithms (EA) are receiving intensive attention in both academics and industry [1–15]. All these techniques are kept under one umbrella called "soft computing." Enormous research had already been done on soft computing techniques to identify a model and control of its different systems.

This paper deals with the synergism of soft computing techniques which are fuzzy logic, ANN, and EA for electrical load forecasting problem. The wavelet transform is used to decompose the past load pattern and used for training and testing of proposed method.

2.2 Wavelet Analysis

The underlying mathematical structure for wavelet bases of a function space is a multiscale decomposition of a signal, known as multi resolution or multiscale analysis. It is called the heart of wavelet analysis.

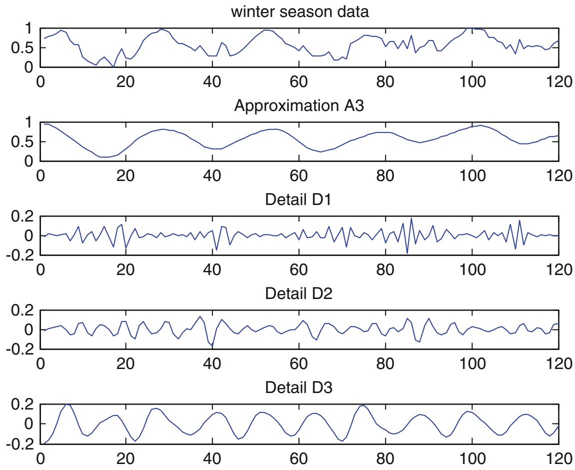

Fig. 2.1 Wavelet decomposition of hour load data into wavelet components

The first step of discrete wavelet transform corresponds to the mapping f to its wavelet coefficients and from these coefficients two components are received, namely a smooth version, named approximation and a second component that corresponds to the deviations or the so-called details of the signal. A decomposition of f into a low-frequency part a, and a high-frequency part d, is represented by $f = a_1 + d_1$. The same procedure is performed on a_1 in order to obtain decomposition in finer scales: $a_1 = a_2 + d_2$. A recursive decomposition for the low-frequency parts follows the directions that are illustrated in the following diagram.

$$\begin{array}{ccccccccc} \boldsymbol{f} & \cdots\cdots & \boldsymbol{a_1} & \cdots\cdots & \boldsymbol{a_2} & \cdots\cdots & \boldsymbol{a_3} & \cdots\cdots & \boldsymbol{a_n} \\ & \searrow & & \searrow & & \searrow & & \searrow & \\ & & \boldsymbol{d_1} & & \boldsymbol{d_2} & & \boldsymbol{d_3} & & \boldsymbol{d_4}\cdots\cdots\boldsymbol{d_n} \end{array}$$

The resulting low-frequency parts a_1, a_2, … a_N are approximations of f, and the high-frequency parts d_1, d_2, … d_N contain the details of f. Figure 2.1 illustrates a wavelet decomposition into four levels and corresponds to a_3, d_1, d_2, and d_3.

$$f = d_1 + d_2 + d_3 + \cdots + d_{N-1} + d_N + a_N.$$

2.3 Generalized Neural Network

In a simple neuron model the aggregation function is summation, which has been modified to obtain a generalized neuron network (GNN) model using fuzzy compensatory operators as aggregation operators to overcome the problems such as large number of neurons and layers required for complex function approximation, which affect not only the training time but also the fault tolerant capabilities of the artificial neural network (ANN) [2].

The common ANN is consisting of summation as aggregation function. As mentioned by Minsky and Parpet [16] in their book that linear perceptron could not be trained for non-separable problems. The multilayer ANN introduced to overcome the problems of perceptron and it was found that three-layer ANN could map any function. The three-layer ANN with simple back-propagation learning algorithm requires large training time. Then large number of back-propagation variants came up with time to improve its training performance. Basically, the training time of ANN depends on the number of unknown weights to be determined. This large number of unknown weights in huge ANN is required to map with complex functions. To obtain large number of weights, large number of training data is required. It is very difficult or sometimes impossible to collect accurate and sufficient training data for real-life problems. The noisy training data affect the testing performance of ANN.

The general structure of the common neuron is an aggregation function and its transformation through a filter. It is shown in the literature [4] that the ANNs can be universal function approximators for given input–output data. The common neuron structure has summation or product as the aggregation function with linear or nonlinear (sigmoid, radial basis, tangent hyperbolic, etc.) as the threshold function.

Different structures at neuron level have been tried to overcome above-mentioned drawbacks of ANN [1]. In this regard ANN consisting of Σ neurons (Σ-ANN), ANN consisting of Π neurons (Π-ANN), and combination of the above two have been tried and the results obtained are quite encouraging [1].

The proposed generalized neuron model shown in Fig. 2.2 has summation and product as aggregation and sigmoid and Gaussian as activation functions. The final

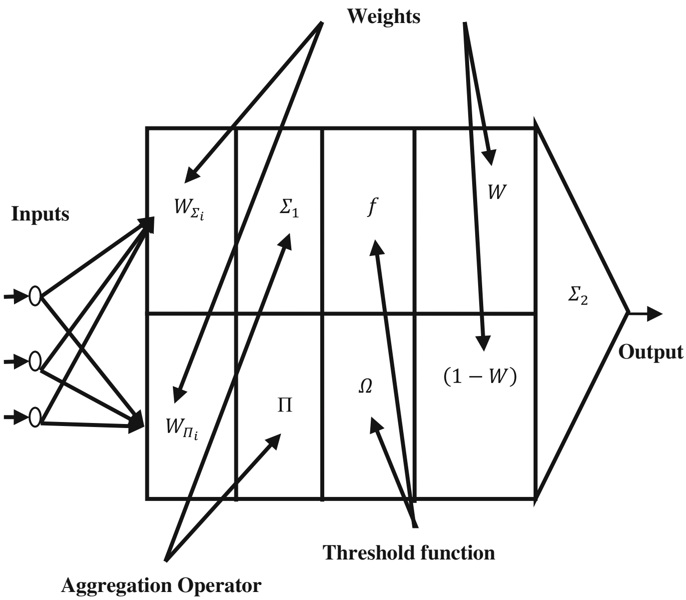

Fig. 2.2 Generalized neural network (GNN)

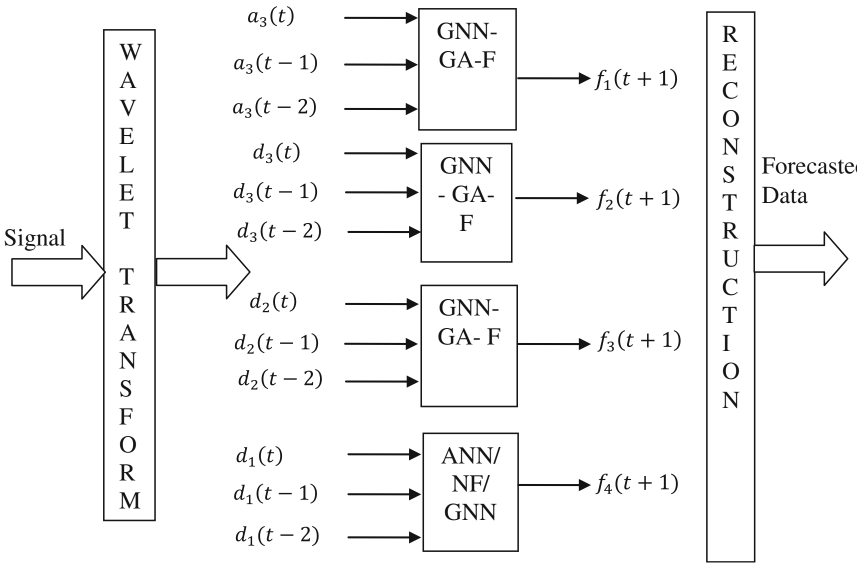

Fig. 2.3 Mechanism for short-term load forecasting

output of the neuron is a function of output of all activation functions. The learning of GNN is explained in [1].

The are many advantages of GNN such as less number of unknown weights, less training time, less number of training patterns, less complexity, and more flexibility.

The basic idea is to use the wavelet transforms and predict the data by synergism of soft computing techniques GNN-W-GA-F for individual coefficients of wavelet transform represented by a_3, d_1, d_2, and d_3. The input to the architecture to predict the wavelet coefficients is explained in Fig. 2.3.

2.4 Adaptive GA with Fuzzy System (GA-F)

Genetic algorithm (GA) simulates the strategy of evolution and survival of fittest. It is a powerful domain-free approach integrated with GNN as a learning tool. The GNN–GA integrated approach is applied to different problems to test this approach. It is well known that the GA optimization is slow and depends on the number of variables. To improve the convergence of GA, adaptive GA is developed, in which, the GA parameters are modified using fuzzy rules [5]. The initial parameters of GAF are given below:

GAF Parameters

- Population size: 50
- Initial crossover probability: 0.9

- Initial mutation probability: 0.1
- Selection operator: tournament selection
- Number of generations: 100

2.5 Short-Term Load Forecasting Using Generalized Neural Network-Wavelet-Genetic Algorithm-Fuzzy System (GNN-W-GA-F)

The neural network (NN) is widely used for short-term load forecasting applications in the past few decades. To improve the performance of ANN, GNN is developed. The GNN is then used to predict each wavelet component separately and combine the (predicted components) to get forecasted load.

The following steps are used in forecasting using GNN-W-GA-F.

Step-I Data collection
The electrical load data was collected from 33/11 kV substation of Dayalbagh Educational Institute (D.E.I.) Dayalbagh Agra. India has been recorded at every 1 h interval for each day for 1 year. The week containing no national holidays, Saturday, and Sunday, or religious holidays are not considered as desired data in the forecasting model. Furthermore, special holidays cannot be used as inputs since they have lower loads than a regular Monday to Friday and mislead the training.

Step-II Preprocessing of Data
The data collected in earlier step is preprocessed.

(a) Filtering of data
In this preprocessing of data, the data is de-noised, i.e., remove bad data. The data of Saturday and Sunday is removed, because the load patterns of these days are quite different and also they are not used in forecasting. Also the error data due to sensor problem or any other fault is removed.

(b) Normalization of data
The filtered and de-noised data is then used for electrical load forecasting after normalizing them.
The normalization range used in normalization process is from 0.1 to 0.9 and not in the range 0–1. This is because in extrapolation there is a tolerance of 0.1 on both sides.

Step-III Wavelet decomposition of Electrical load pattern
The wavelet transform is used to decompose the normalized electrical load pattern into a number of wavelet **components** as shown in Fig. 2.1. The original normalized signal of load demand is decomposed to high-

and low-frequency component by using **db8,** mother wavelet (**db8**) for calculating the coefficient of the details (***d***) and approximate (***a***) components.

Step-IV Selection of training pattern

The first step for training is obtaining an accurate and sufficient historical data after preprocessing. The data should be chosen that is relevant to the model. How well the data is chosen is the defining factor in how well the model output will match the event being modeled. There should be some correlation between the training data and the testing data. In the load data, for example, all the Monday's data look alike and this holds good for all the days of the week with some variations.

The wavelet-decomposed components are used for training.

The training patterns are consisting of decomposed wavelet components of given load pattern at time t, $t - 1$, $t - 2$ (past three points) as input and the forecasted wavelet component at $t + 1$ as output. Hence, training patterns expressed as pair of set of input and output.

Training Pattern = [Input vector] →
[Output Vector]

Roughly 85 % of total load data is used for training and rest 15 % load data is used for testing of models. The pseudo code of GNN-W-GA-F is given below.

```
Begin GNN-W-GAF
  Collect a set of data.
  Decompose the data into wavelet components
  Initialize parameters of GAF and GNN-W
      For I =1:Gmax
        while pop < popmax
          Evaluate the fitness using GNN-W
        end
        generation= generation + 1
        Modify crossover, mutation rate using FS
        select parental chromosomes
        Perform GA operators
        Get new population
      End
```

Step-V Forecasting using GNN-W

The forecasting models using GNN-W-GA-F for wavelet components have been used after proper training.

Step-VI Reconstruction of forecasted load

The forecasted load pattern is reconstructed after combining the wavelet components. In the comparisons of model performance, the load forecast accuracy is determined by RMSE.

2.6 Results and Discussions

The training of $\boldsymbol{a}_3$ component using GNN-W-GAF is shown in Fig. 2.4 as maximum fitness of GA-F. Actual load and forecasted load using **GNN-W-DA-F during testing** is given graphically in Fig. 2.5.

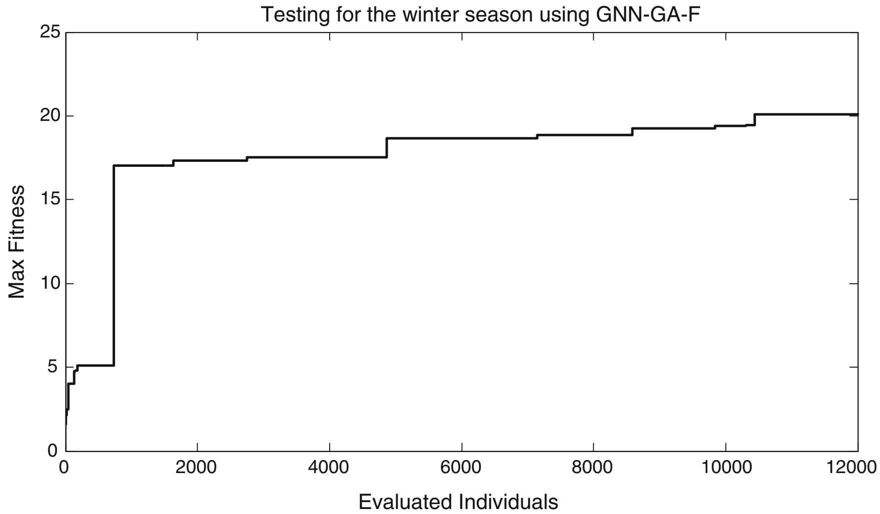

Fig. 2.4 Maximum Fitness of GA fuzzy during training of a_3 wavelet component of using GNN-W-GA-F

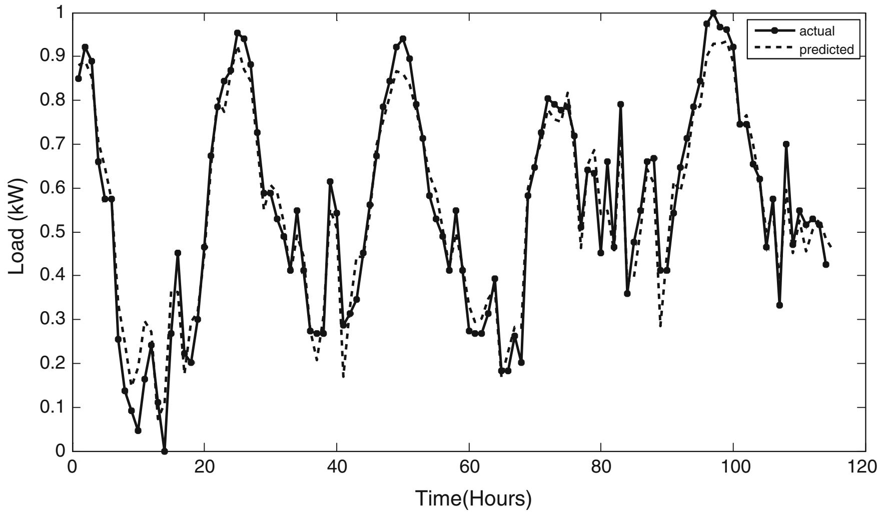

Fig. 2.5 Testing performance of GNN-W-GAF model

2.7 Conclusion

The paper deals with short-term electrical load forecasting problem using integrated approach of soft computing techniques and wavelet transform. The techniques and forecasting models were applied to datasets available from 33/11 kV substation of Dayalbagh Educational Institute, Dayalbagh, Agra, U.P. India. The soft computing technique, GNN-W-GA-F, has been applied to develop models for STLF. The integrated model, i.e., GNN-W-GAF gives the least RMSE in comparison to all the other ANN-based models.

References

1. Chaturvedi DK (2008) Soft computing techniques and applications to electrical engineering. Springer
2. Chaturvedi DK, Satsangi PS, Kalra PK (1999) New neuron model for simulating rotating electrical machines and load forecasting problems. Int J Elect Power Syst Res Elsevier Science, Ireland **52**:123–131
3. Chaturvedi DK, Satsangi PS, Kalra PK (2001) Fuzzified neural network approach for load forecasting problems. Int J Eng Intell Syst CRL Publishing, UK 9(1):3–9
4. Chaturvedi DK, Kumar R, Mohan M, Kalra PK (2001) Artificial neural network learning using improved genetic algorithm. J IE(I), EL, 82
5. Chaturvedi DK, Mohan M, Singh RK, Kalra PK (2004) Improved generalized neuron model for short term load forecasting. Int J Soft Computing—A Fusion of Foundations, Methodologies and Applications, Springer-Verlag Heidelberg 8(1):10–18
6. Chaturvedi DK, Satsangi PS, Kalra PK (1997) Short term load forecasting using generalized neural network (GNN) approach. J Inst Eng (India) 78:83–87
7. Chaturvedi DK (2009) Modeling and simulation of systems using MATLAB® and Simulink®. CRC Press (2009)
8. Chaturvedi DK, Premdayal SA, Chandiok A (2010) Load forecasting using soft computing. Int J Commun Netw Syst Sci 3:273–279
9. Chaturvedi DK, Singh AP (2011) Evapotranspiration model for irrigation scheduling. Int J Math Sci Appl 2(1):393–399
10. Chaturvedi DK, Premdayal SA (2012) Neural-wavelet based hybrid model for shortterm load forecasting. In: Proceedings of national conference on emerging trends in electrical, instrumentation and communication engineering (ETEIC—2012), Agra, 6th and 7th April 2012, pp 359–362
11. Chaturvedi DK, Chandiok A, Kamaruddin M (2009) Fuzzy system model of internet system exploitation and usage in society. IEEE Int Conf Innov Technol (IEEE- IMS/EMBS2009), New Delhi, 18–19th June 2009
12. Chaturvedi DK, Chandiok A, Siddiqui AH (2009) Prediction for the time series of annual rainfall using wavelets and neuro-fuzzy approach. Int workshop on Wavelets, Istanbul, Turkey, 4–7th June 2009
13. Chaturvedi DK, Siddiqi AH, Chandiok A, Agarwal S (2009) Annual rainfall forecasting using soft computing techniques. In: Proceedings of joint international conference on applied systems research and national systems conference, (NSC) Nov. 27–29, 2009 organized by D. E.I. (Deemed Univ.) Dayalbagh, Agra, India, TMH-2009:103–106

14. Banakar MFAA (2008) Artificial wavelet neural network and its application in Neurofuzzy models. Elsevier Appl Soft Comput
15. Sinha N, Lai LL, Ghosh PK, Yingnan M (2007) Wavelet-GA-ANN based hybrid model for accurate prediction of short-term load forecast. In: Proceedings international conference on intelligent systems applications to power systems (ISAP), 5–8 Nov, pp 1–8
16. Minsky M, Papert S (1968) Perceptions. The MIT Press, Cambridge, MA

Springer
http://www.springer.com/978-981-287-971-4

Mathematical Models, Methods and Applications
Siddiqi, A.H.; Manchanda, P.; Bhardwaj, R. (Eds.)
2015, XIX, 298 p. 103 illus., 60 illus. in color., Hardcover
ISBN: 978-981-287-971-4