Electrical Power and Energy Systems 67 (2015) 230–237

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Short term load forecast using fuzzy logic and wavelet transform integrated generalized neural network

D.K. Chaturvedi [a,*], A.P. Sinha [a], O.P. Malik [b]

[a] Dept. of Electrical Engineering, D.E.I. Dayalbagh, Agra, UP 282110, India
[b] Dept. of Electrical Engineering, University of Calgary, Calgary, Canada

ARTICLE INFO

Article history:
Received 20 September 2013
Received in revised form 16 October 2014
Accepted 24 November 2014
Available online 11 December 2014

Keywords:
Load forecasting
ANN
Generalized neural network
Wavelet
Adaptive genetic algorithms
Fuzzy systems

ABSTRACT

Application of Artificial Neural Networks (ANNs) for electrical load forecasting has been proposed in the literature. ANNs have some inherent drawbacks and limitations, such as difficulty in deciding the structure of ANN, selection of type of neuron, large training time, sticking to local minima, etc. To overcome the drawbacks of ANN, a Generalized Neural Network (GNN) has been proposed in the past. An algorithm that integrates wavelet transform, adaptive genetic algorithm and fuzzy system with GNN is described and applied to the short term week day electrical load forecasting problem. Performance of the proposed algorithm is compared with other GNN variants on the basis of prediction error.

Introduction

Short-term load forecasting (STLF) approaches available in the literature can be divided into two main categories: statistical methods and artificial intelligence based methods. The statistical category includes multiple linear regression [1], stochastic time series [2], ARIMAX and general exponential smoothing [3–5], state space model [6], and support vector regression (SVR) [7,8], whereas expert system [9], artificial neural network [10–14] and fuzzy inference [15–16] belong to the artificial intelligence category.

The use of Artificial Neural Networks (ANNs or simply NNs) for load forecasting has been proposed since the 1990s. Normally, ANN is trained using back propagation or its variants, but back-propagation learning has many limitations. A Generalized Neural Network (GNN) has been developed to overcome the drawbacks of ANN and used for modeling [17], forecasting [18–20] and control applications [21–23]. The performance of ANN and GNN is compared in paper [23].

Genetic algorithms (GAs) are more robust than the directed search (gradient back propagation) methods and also possess other useful characteristics. For example, hill climbing methods provide local optimum values and these values depend on the selection of a starting point. Also there is no information available on the relative error with respect to global optimum. To increase the success rate in the hill climbing method, it is executed for large number of randomly selected different starting points. On the other hand, GA optimization is a random search [24] and does not need the derivative of error. Hence, any continuous or discontinuous function may also be used as a threshold function of NNs. Employing a random search GA guarantees global optimum.

To improve its performance, fuzzy rules can be used to guide it.

A GNN model with four wavelet components as inputs (called GNN-W) and trained using adaptive GA with fuzzy concepts (GAF) is developed. The proposed GNN-W-GAF model is used for STLF and its performance compared with that of regular GNN and GNN-W with back-propagation training.

STLF using GNN with back-propagation training model (GNN-BKP)

The GNN consists of a single higher order neuron as shown in Fig. 1 [17–19]. In the GNN model A_1, A_2 are summation and product aggregation functions and f_1, f_2 are sigmoid and Gaussian activation functions, respectively.

The GNN model was initially trained using error back-propagation (BKP) gradient search learning algorithm and applied to the STLF problem on datasets obtained from a 15 MVA, 33/11 kV sub-

* Corresponding author.
E-mail addresses: dkc.foe@gmail.com (D.K. Chaturvedi), anandsinha.dei@gmail.com (A.P. Sinha), maliko@ucalgary.ca (O.P. Malik).

http://dx.doi.org/10.1016/j.ijepes.2014.11.027
0142-0615/

station at Dayalbagh Educational Institute (D.E.I.), Agra, India. Although the results obtained during the training and testing of the GNN model with single generalized neuron were quite promising (Ref. Fig. 2), they showed some room for improvement. A comparison of ANN-BKP (structure 4-4-1) and GNN-BKP is shown in Fig. 2. This provided motivation to seek further improvement and the GNN was trained using adaptive GA – Fuzzy system.

STLF using GNN trained with adaptive genetic algorithm and fuzzy system (GNN-GAF)

Training a feed-forward GNN for the STLF problem using the back propagation learning mechanism has some drawbacks as below:

i. It is a slow learning process, especially when large training sets or large networks have to be used.
ii. Network may get stuck in local minima.
iii. The threshold function should be differentiable and non-decreasing.
iv. The training time in backprop depends upon
  a. Training parameters and initial weights.
  b. The error function used.
  c. The normalization range of training data and input output mapping.

The central theme of research on genetic algorithms has been robustness, the balance between efficiency and efficacy necessary for survival in many different environments. The following are the advantages of GA:

i. It is a sophisticated search procedure based on the mechanics of natural genetics. The search is absolutely blind, but guided by pre-designated precise operators.
ii. It has a good potential as a problem solving tool, especially in finding near optimal solutions.
iii. GA based methods search from a population of potential solutions unlike other methods, such as hill climbing method, that process a single point of the search space.
iv. It uses pay off information (objective function), not derivatives or auxiliary knowledge.
v. It uses probabilistic transition rules, not deterministic rules.
vi. GAs work with coding of the parameter themselves.

Operators of GA

The chromosomes of GA consist of weights of GNN and they are modified using GA operators to get new population. The crossover and mutation are the most important operators of the genetic algorithm. Depending on the number of variables GA optimization can be slow. To improve the convergence of GA, adaptive GA (GAF) is developed, in which the GA parameters {crossover probability (P_c), mutation probability (P_m) and population size} are modified using fuzzy rules to improve its performance. The initial parameters of GAF are given below.

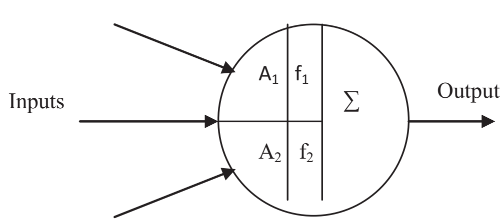

Fig. 1. GNN model.

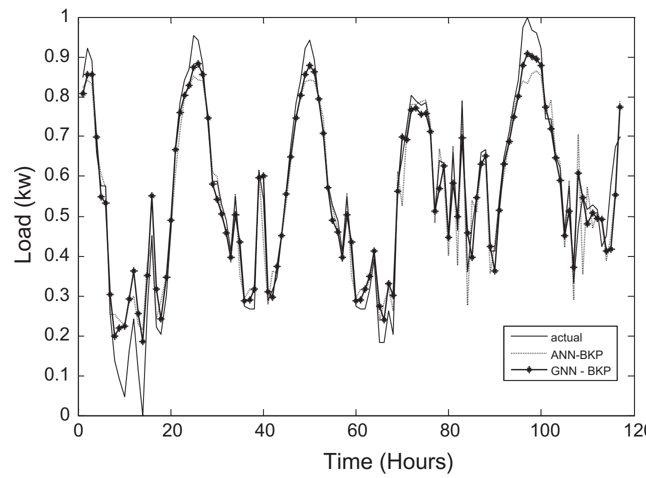

Fig. 2. Test performance of GNN-BKP and ANN-BKP model.

Population size: 50.
Crossover probability, initial value: 0.9.
Mutation probability, initial value: 0.1.
Selection operator: tournament selection.
Number of generations: 100.

The application of GNN-GAF model is applied for load forecasting as shown in Fig. 3. The GNN is trained for past three electrical load values as input and next hour load as output. The error function is calculated from predicted load of GNN and actual load and it is minimized using adaptive GAF. Flow chart for GNN-GAF is given in Fig. 4.

Development of adaptive genetic algorithm using fuzzy system (GAF)

Details of parameter variations and their influence on the optimization process have been studied by many researchers [25–30]. In all these studies the objective function is optimized using GA for different sets of parameters that are initialized at the time of starting. Normally, these GA parameters are kept constant during optimization. In the adaptive GA the parameters such as:

i. crossover Probability (P_c), and
ii. mutation probability (P_m)

are varied dynamically during the execution of the program. For this variation the fuzzy knowledge base that has been developed from experience to maximize the efficiency of GA, is used.

Basis of variation of P_c and P_m
Philosophy behind the variation of these parameters is that the GA optimization depends on the crossover and mutation operation.

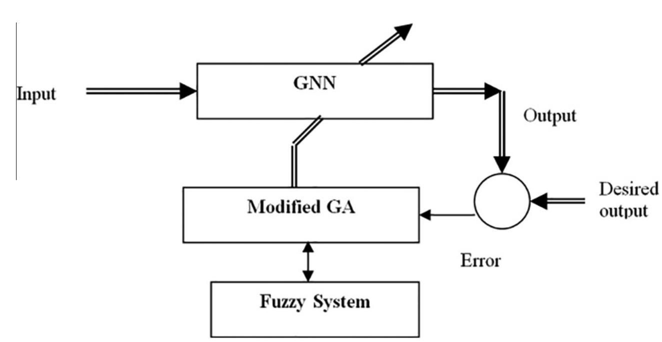

Fig. 3. GA as learning tool for GNN.