Proceedings of the Seventh IASTED International Conference
POWER AND ENERGY SYSTEMS
August 29–31, 2007, Palma de Mallorca, Spain
ISBN Hardcopy: 978-0-88986-689-8 / CD: 978-0-88986-690-4

GENERATOR ECONOMIC DISPATCH BEHAVIOR MODELING USING THE MLP NEURAL NETWORK TRAINED WITH THE LEVENBERG-MARQUARDT ALGORITHM

Victor H.C. Watt, Audley B. Darmand, and Dwight Reid

School of Engineering
University of Technology
237 Old Hope Road Kingston 6
Jamaica, W.I.

ABSTRACT
This paper presents a method to model the economic dispatch outputs of the generators in the Jamaica power system. This is done by using an MLP neural network to model the function that relates the generators' outputs and total system demand at a time t to the output of a particular generator at time (t+T), where T is the dispatch period. This work is done as a first step in economic dispatch prediction where not only the system demand is predicted but also the economic outputs of the system generators with inherent knowledge of systems constraints and operating policies.

KEY WORDS
Economic Dispatch, Artificial Neural Networks, Power System Modeling, Levenberg-Marquardt.

1. Introduction

Economic dispatch (ED) is done by utilities on a day to day basis in order to satisfy the total system load at minimum operating costs [1]. The economic dispatch procedure requires information concerning the generating units available for dispatching. The procedure to determine what units to make available is known as unit commitment (UC). The UC procedure aims to determine the best set of units to make available to supply the forecasted system load in a future time period [2]. The UC therefore requires a forecast of the system load for its computations, the more accurate this forecast the more economical the outcome of the UC calculations [3].

The load forecast can be classified as short term, with forecasting times of a few minutes to several days, up to long term with forecasting times measured in years [4]. For the ELD operation it is the short-term or sometimes very short-term load forecasting that is of interest [5]. The methods to solve the short-term load forecasting have been roughly classified into two types in [6], they are: statistical methods and Artificial Intelligence (AI) methods. The statistical methods include stochastic time series, Multiple Linear Regression and Auto Regressive Moving Average (ARMA) methods, an overview of these methods is given in [7]. The AI techniques include Expert Systems [8] and Artificial Neural Networks (ANN) [9]. The ANN has also been used to model the load as a Fourier expansion [10].

The load forecasting techniques are used to obtain an estimate of the entire system load using historical data, weather parameters, etc. to get the load on a single generator it is necessary know the outputs of the other system generator. This paper presents a method to model the individual generator outputs as determined by the Economic Dispatch calculations. The modeling is done using a fully connected, feed-forward Multilayered Perceptron (MLP) ANN that is trained on historical economic dispatch data and is done as a first step toward Economic Load Forecasting (ELF). ELF attempts to forecasts the load to put on a generator that will allow the generator to make its contribution to serving the system load in the most economic way.

Section II presents an overview of the problem then Section III gives a brief overview of artificial neural networks and the Levenberg-Marquardt training algorithm. The architecture of the implemented ANN is given in Section IV and the simulation design and results obtained using historical ELD data are presented in Section V. Section VI presents a brief discussion of the work presented and Section VII the Conclusions.

2. Problem Development

The economic dispatch problem can be described mathematically as shown in Equation (1).

$$\min\left(\boldsymbol{F}\left(\boldsymbol{P}_{\boldsymbol{G}}\right)\right)=\sum_{\boldsymbol{i}=1}^{\boldsymbol{N}}\boldsymbol{P}_{\boldsymbol{Gi}} \tag{1}$$

subjected to the following inequalities:

$$\boldsymbol{P}_{\boldsymbol{Gi}\min}\leq\ \boldsymbol{P}_{\boldsymbol{Gi}}\leq\ \boldsymbol{P}_{\boldsymbol{Gi}\max} \tag{2}$$

and the power balance constraint:

$$\sum_{i=1}^{N} P_{Gi} = P_D \quad (3)$$

where $F(P_G)$ is the total fuel cost of the system, P_{Gi} is the power generated by unit i where $i = 1,2,3,\ldots N$ (the total number of generating units), $P_{Gi,min}$ and $P_{Gi,max}$ are the lower and upper power generating limits of unit i, P_D is the total power demand [11].

The result of the ELD calculations is the output of each generator on the system that will meet the system load the most economically. This calculation is normally done periodically throughout the day during normal system operation. For proper operation the load has to be forecasted with at least the same time resolution as the ELD updates, in addition the ELD data is usually archived and so there is a large amount of it making it suitable for use in ANN training.

For the ELD calculations a load forecast is needed and this can be obtained by modelling the load. The power system load can be modelled as a combination of four components as shown in Equation (4) below.

$$L = L_n + L_w + L_s + L_r \quad (4)$$

where L_n is the normal part of the load as determined from historical load behavior, L_w is the weather sensitive part of the load, L_s is the part of the load sensitive to special days and L_r is the random part of the load determined by random behavior of utility customers [12]. Since all generators contribute to the system load, the output of any one generator must be dependent on the outputs of the other generator's on the system at any given time. An extension of the previous statement is that the output of a generator in time period t is dependent on the outputs of the other system generators and its output in time period $(t-1)$. This statement is illustrated mathematically in Equation (5) below where, $P_{Gi}(t)$ is the output of generator i at time t.

$$P_G(t) = f(P_{G1}(t-1), P_{G2}(t-1), \ldots, P_{GN}(t-1)) \quad (5)$$

In this work Equation (5) is modelled with an MLP ANN that is trained on historical data of the ELD power outputs of the system generators at different time periods.

3. The Levenberg-Marquardt Algorithm

The MLP ANN is composed of a network of artificial neurons arranged in various layers that are connected together through connection weights in a fully connected feed-forward architecture [13]. The neurons used for implementing the MLP ANN in this work are continuous output neurons. The neurons are made continuous by using a sigmoid activation function for the neurons, this allows a single neuron produce a continuous output between zero and one. Figure 1 shows the neuron model used to build the network.

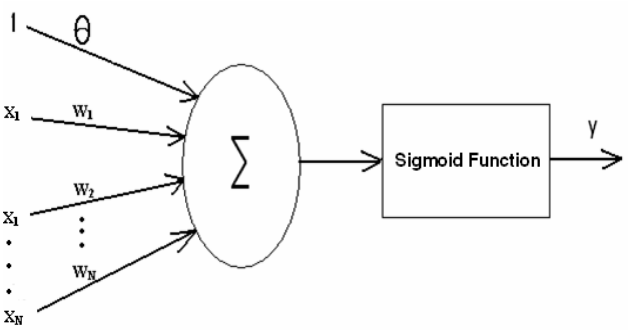

Figure 1: Artificial neuron

The neurons are McCulloch and Pitts [13] neurons and work by finding the weighted sum of its inputs and then pass this sum to the activation function. This is illustrated in Equations (6).

$$y = f\left(\sum_{i=1}^{N} w_i x_i + \theta_i\right) \quad (6)$$

In Equation (6) w_i is the weight of connection i, x_i is input i, θ_i is the bias and f represents the activation function. The activation function used in this work is the sigmoid function shown in Equation (7) below.

$$f(y) = \frac{1}{1+e^{-y}} \quad (7)$$

The training of the MLP ANN is carried out by adjusting the weights connecting the neurons in the network [14], this can be done according to Equation (8) below.

$$w_{k+1} = w_k + \Delta w_k \quad (8)$$

In Equation (8) w_k represents the connection weight at time $t=k$. The network is trained by finding a suitable value for the Δw term. The most popular algorithm for training the MLP ANN is the Back-Propagation Algorithm (BPA) that adjusts the weights on the various layers of the MLP ANN by propagating the error at the output layer backwards through the network to obtain corresponding errors on the inner layers [15]. The BPA generally requires a lot of time for training making it unsuitable for real time or close to real time online operations. This low training speed of the BPA has led researchers to develop faster training algorithms over the years since its introduction. One of the most popular developments of the BPA is the Levenberg-Marquardt (LM) algorithm which gives faster training at the expense of more memory use. The LM algorithm is an optimization algorithm used to solve non-linear least square optimization problems. It is derived from a combination of the Gradient Descent (GD) and Gauss-

Newton (GN) optimization methods and is illustrated in Equation (9) below [16].

$$\Delta \boldsymbol{w} = \left[\boldsymbol{J}^{\boldsymbol{T}} \boldsymbol{J} + \boldsymbol{\mu}\, \boldsymbol{I} \right]^{-1} \boldsymbol{J}^{\boldsymbol{T}} \boldsymbol{e} \qquad (9)$$

In (9) J is the Jacobian, I the identity matrix, e the error given by Equation (10) below and μ the LM learning rate that is adjusted depending on whether the error increases in a step. If the error increases μ is increased otherwise μ is progressively decreased at each step.

$$\boldsymbol{e}_{\boldsymbol{i}} = \boldsymbol{d}_{\boldsymbol{i}} - \boldsymbol{y}_{\boldsymbol{i}} \qquad (10)$$

In Equation (10) e_i is the error, d_i represents the desired output of the network and y_i the actual output.

4. ANN for ELD Modelling

The system under study has 26 generators with the outputs of all used as inputs to the ANN. Also there is a portion of the load that is un-serviced at times, this is also used as an input to the ANN. We therefore have 27 inputs and a single output representing the output of a single generator in the future time period. In implementation the ANN has 27 input neurons, one for each input, 15 hidden layer neurons and a single output neuron, all neurons use a log-sigmoid activation function. The architecture of the ANN used is shown in Figure 2.

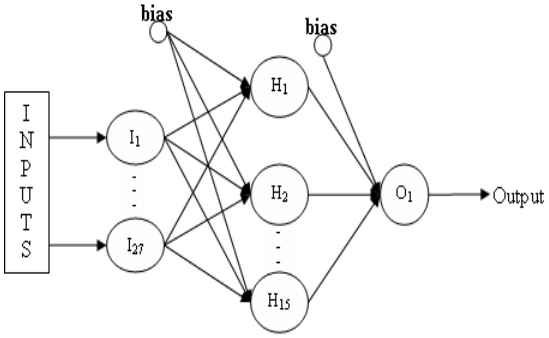

Figure 2: ANN Structure

The network is trained incrementally using the LM algorithm. The weights of the ANN are adjusted each time an input is presented to it and an output obtained, when all training samples are presented once this is referred to in this work as one training cycle or epoch. The training is terminated after 1000 epochs.

The inputs to the ANN are generator output values in MW and since the neurons use a log- sigmoid activation function, whose output is bounded between 0 and 1, they have to be normalized before they can be presented to the ANN. All inputs are therefore divided by the maximum input value which is 123.9MW to normalize them before they are presented to the ANN. The data set used for this work is a whole month of ELD data from the Jamaica power system sampled at half hour intervals giving a total of 1488 data points.

The ANN was trained on 200 data points at a time and after each training and evaluation the ANN weights were then randomized and the ANN trained on the next set of 200 data points. This was done to obtain a basic view of how the ANN performs when trained on a limited amount of data points, this is important for online operation since it allows for faster training and hence greater close-to-real time operation which is necessary to ensure that the prediction results are obtained in a timely manner.

5. Simulation and Results

The ANN was implemented in the Matlab environment on a 1.6 GHz personal computer. The training algorithm use is the LM algorithm with incremental weight update.

The initial values for the weights are randomly generated and are between 0 and 1. The inputs and the desired outputs are normalized to the maximum input value to ensure that they have values between 0 and 1.
The training set consisted of 200 data vectors each with 27 values that produce a corresponding 200 output values that represent the output of one system generator. The present output of the generator whose output is being modelled is also an input and at time *(t+T)* the actual value of that generator is also used as an input and not the forecasted value.

The training was carried out for 1000 epochs and results showing a graphical comparison between actual and ANN outputs are shown below in Figure 2 for the first 200 data points and Figure 3 the second.

The performance of the ANN was evaluated by the mean square error (MSE) equation shown below in Equation (11).

$$\boldsymbol{MSE} = \frac{1}{\boldsymbol{N}} \sum_{\boldsymbol{i}=\boldsymbol{i}}^{\boldsymbol{N}} \left(\boldsymbol{d}_{\boldsymbol{i}} - \boldsymbol{y}_{\boldsymbol{i}} \right)^2 \qquad (11)$$

In Equation (11) d_i represents the desired output, y_i the actual output at interval i and N the total number of samples. The MSE error obtained for the first 200 data points is 6.563MW and for the second 200 points is 14.013.

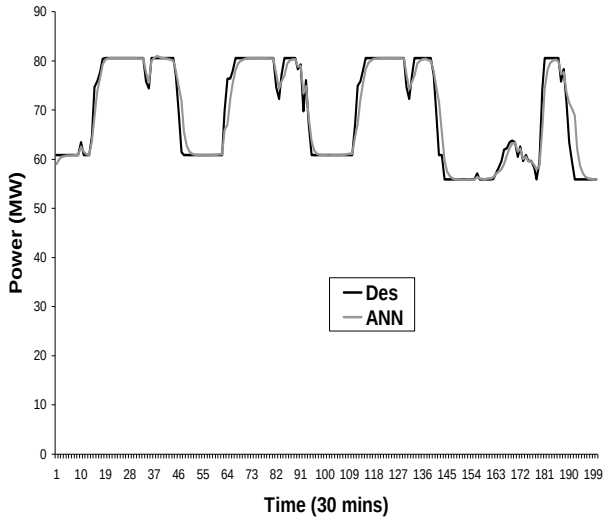

Figure 3: Comparison for First 200 Points

.

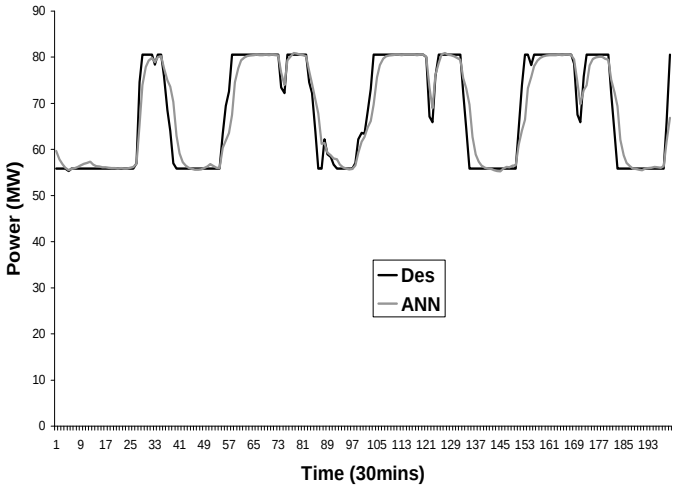

Figure 4: Comparison for Second 200 Points

6. Discussion

It should be observed that in addition to the constraints presented above in the definition of the ELD problem, utilities generally have operating polices that are determined by system security, stability, supply contracts etc. that dictate the operation of some of the generating units. The effects of these constraints are inherently contained to the ELD output data and this knowledge is captured by the ANN during training.

7. Conclusion

A method to model the ELD behaviour of generators in the Jamaica power system by using an Artificial Neural Network is presented in this paper. The proposed approach uses a 3 layer MLNN to produce the ELD output of a generator in the system at a time t given the outputs of all the system generators and the un-serviced load at time *(t-T)* where *T* is the ELD period. The results obtained show the comparison of the ANN output and the actual ELD outputs when the ANN was trained for 1000 epochs on only 200 data points. The results show that the ANN is able to model the ELD outputs with an MSE of 6.563 and 14.013. The modelling of the ELD outputs of system generators is helpful in system analysis and possibly load forecasting on individual generators and system control.

Further work entails simulations of data acquisition and training of ANN in real time system.

References

[1] Chao-Ming Huang and Yann-Chang Huang, "A Novel Approach to Real-Time Economic Emission Power Dispatch". IEEE Transactions on Power Systems, VOL. 18, No. 1, February 2003.

[2] John J. Granger and William D. Stevenson Jr, "Power System Analysis". McGraw Hill Inc. New York.

[3] Chuan-Ping Cheng, Chih-Wen Liu, and Chun-Chang Liu, "Unit Commitment by Lagrangian Relaxation and Genetic Algorithms". IEEE Transactions on Power Systems, Vol. 15 No.2, May 2000.

[4] Henrique Steinherz Hippert, Carlos Eduardo Pedreira and Reindaldo Castro Souza, "Neural Networks for Short-Term Load Forecasting: A Review and Evaluation". IEEE Transactions on Power Systems, Vol. 16 No.1, February 2001.

[5] Wiktor Charytoniuk and Mo-Shing Chen, "Very Short-Term Load Forecasting Using Artificial Neural Networks". IEEE Transactions on Power Systems, Vol. 15 No. 1 February 2000.

[6] Hong-Tzer Yang and Chao-Ming Huang, "A New Short-Term Load Forecasting Approach Using Self-Organizing Fuzzy ARMAX Models". IEEE Transactions on Power Systems Vol. 13, No. 1 February 1998.

[7] Ibrahim Moghram, Saifur Rahman, "Analysis and Evaluation of Five Short-Term Load Forecasting Techniques". IEEE Transactions on Power Systems, vol. 4, No.4, October 1989.

[8] M. S. Kandil, S. M. El-Debeiky and N. E. Hasanien, "Long-Term Load Forecasting for Fast Developing Utility Using a Knowledge-Based Expert System". IEEE Transactions on Power Systems, vol. 17, No.2, May 2002.

[9] Henrique Steinherz Hippert, Carlos Eduardo Pedreira, and Reinaldo Castro Souza, "Neural Networks for Short-Term Load Forecasting: A Review and Evaluation". IEEE Transactions on Power Systems, vol. 16, No.1, February 2001.

[10] Audley Darmand, Masters Thesis, "Intelligent Power System Load Modeling using Group Method of Data Handling Artificial neural Networks". North Carolina A & T State University, North Carolina 2001.

[11] V. C. Watt, A. B. Darmand, D. D. O. Reid, "Economically Dispatching Generators in the Jamaica Power System using the Classical Technique in Conjunction With Artificial Neural Networks".

Proceedings of the 8th IASTED International Conference on Power and Energy Systems, October 2005
[12] P.K. Dash etal., “A Real-Time Short-Term Load Forecasting System Using Functional Link Network”. ”. IEEE Transactions on Power Systems, Vol. 1 No.2, May 1997.
[13] Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach”. Pearson Education Inc, New Jersey 2003.
[14] B. S. Lathika and Shameer A Koya, “Active Power Line Filter using Artificial Neural Network and Fuzzy Control”. Proceedings of the 8th IASTED International Conference on Power and Energy Systems, October 2005.
[15] Ali Zilouchian, Mo Jamshidi, “Intelligent Control Systems Using Soft Computing Methodologies”. CRC Press LLC, 2001.
[16] Amir Abolfazl Suratgar, Mohammad Bagher Tavakoli, and Abbas Hoseinabadi, “Modified Levenberg-Marquardt Method for Neural Networks Training”. Transactions on Engineering, Computing and Technology, World Enformatika Society June 2005.