**Proceedings of the Canadian Society for Mechanical Engineering International Congress 2021**
**CSME Congress 2021**
**June 27-30, 2021, Charlottetown, PE, Canada**

# ARTIFICIAL NEURAL NETWORK BASED PREDICTION OF WIND TURBINE POWER CURVE USING VARIOUS TRAINING ALGORITHMS

Muhammad Usman Saram[1*], Jianming Yang[1], Zaheer Ahmad[2], Sadaf Zahoor[3]
[1]Faculty of Engineering and Applied Science, Memorial University of Newfoundland, St. John's, NL, Canada
[2]Department of Industrial Engineering, University of Engineering & Technology, Taxila, Pakistan
[3]University of Windsor, Windsor, ON, Canada
*e-mail: musaram@mun.ca

***Abstract***—The size and power outputs of wind power plants have been improved in the last decade due to the technical developments in this field. Now wind energy experts are more interested in studies that aim at improving the output of wind forms. There is a non-linear relationship between the power output of a wind turbine and its primary and derived parameters, for which, artificial intelligence techniques can be usefully adopted. Artificial Neural Network (ANN) has been successfully used as an interdisciplinary tool in many applications. The present paper aimed to apply the ANN methodology for the assessment of wind energy output of a group of turbines. The ANN modelling was executed using MATLAB deep learning toolbox. ANN was trained using three different training algorithms namely Levenberg-Marquardt (LM), Quasi-Newton (QN) and Resilient Back Propagation (RBP). Then the effect of these training algorithms on learning performance of neural networks on the prediction of wind turbines power output was investigated. Conclusively, Levenberg-Marquardt (LM) training algorithm predicted the best results for output power of a group of turbines.

***Keywords – wind power; wind speed; neural network; deep learning, training algorithms***

## I. INTRODUCTION

Wind energy generation on industrial scale is quite new. The performance of wind power farms has not been adequately studied. Low accuracy of the energy output prediction is one of the weakest point in wind power generation. This factor motivates the researchers for the analysis and prediction of wind power generation. Researchers have applied different methods in estimating wind farms power output. In technical literature, however, artificial intelligence techniques such as neural network and fuzzy logic are found to be more accurate as compared to other traditional predicting methods.

Pelletier et al. [1] has successfully forecasted the wind turbine power curves with six input parameters using ANN technique. The six parameters namely: wind speed, wind shear, turbulence intensity, air density, wind direction and yaw error, has been selected from more than fifty parameters. Results were also compared with other predicting methods and found to be in good agreement. Mabel and Fernandez [2] used ANN method to predict the power output of a wind farm from data that has been collected over a period of 3 years. They used three parameters as input: wind speed, relative humidity and generation hours. Li et al. [3] performed a comparative analysis of regression and ANN method for wind turbine power curve estimation and found that ANN performs better than regression analysis.

In this study, ANN is used to predict the wind turbine power generation for the data collected from technical literature. The aim of this work is to investigate the predicting performance of ANN using various algorithms for the power generation of a wind turbine. The MATLAB deep learning toolbox is used to perform the ANN. The ANN method has been used for some researchers to predict the wind turbine power curves. However, the distinguished characteristic of this paper from the other papers is that this paper discusses the prediction results through the performance of three different ANN training algorithms, while previous works have not mentioned the use of these training algorithms.

## II. WIND POWER [2]

The power available at the wind turbine in watts is given by

$$P = (0.5)\rho A V^3 \qquad (1)$$

where $V$ is the wind speed in m/s, $A$ is the swept area of wind turbine in $m^2$ and $\rho$ is the density of air in $kg/m^3$. The wind speed $V$ has a major influence on wind turbine power output, as the power output varies with the cubic value of speed. The variation in air density during different period of a year and at different location is less compared to the variation in wind speed. The correlation study shows a strong dependence of the

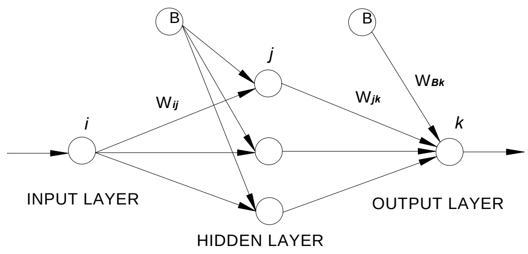


Figure 1. A typical neural network structure with one input and one output layers, and a hidden layer with three neurons.

wind speed on power generation. Substantial variations in seasonal or monthly average wind speed are common in the most parts of the worlds.

The power output is also directed by the design features of wind turbine such as cut-in speed, rated wind speed and cut-out speed. The cut-in speed is the minimum wind speed required for the turbine to start delivering power and generally it is in the range of 2.5 – 3.5 m/s. The rated wind speed is the wind speed above which the machine delivers the rated power output and is approximately 14 m/s. Cut-out speed is the maximum wind speed the turbine is allowed to deliver power, this speed is in the range of 20 – 25 m/s [4].

## III. Neural Network Structure

ANN method is from the artificial intelligence family and has been developed to model nonlinear processes in numerous disciplines. The main idea of ANN methodology looks like the functioning of human brain. It is self-adaptive to environment so that it can sensibly respond to different inputs [5]. Some benefits of ANN are adoption, learning, ease in implementation and generality [6]. The three main features of ANN are: the structure, the type of activation function and learning algorithm. The structure refers to the number of nodes in each layer and the type of connection between these nodes. The activation function indicates the transfer function of each neuron, and the cost function of the network outputs. The last feature learning deals with the use of a learning algorithm, and the parameters in that algorithm.

ANN can be examined in two categories namely: feedback neural networks and feed forward neural networks. The back propagation neural network is the most widely used network model. It has the strong learning ability in training and mapping the relations between inputs and outputs [6].

Back propagation neural network is feed forward, multilayer network with number of hidden layers and trained using learning algorithm as shown in Fig. 1. Learning is carried out in multi-layer preceptor (MLP) model with the use of iterative gradient algorithm in order to minimize the mean square error between the actual outputs of the network and the desired output in response to given inputs [7]. Training is accomplished by forward and backward operations. The actual output is produced for a certain input using the weights of current connection. The backward operation is performed afterwards to modify the weights to decrease the error. The two parameters namely, learning rate ($\eta$), and momentum coefficient ($\alpha$) affect the adjustment of weights. Learning rate describes the range of changes in connection weights. Momentum coefficient improve the learning process by adding a term to the weight alteration that is proportional to the previous weight change [8].

The neurons can be categorized into three types based on their inputs and outputs: input, output and hidden neurons. Input neurons are the neurons that receive input from the environment, such as wind speed in this study. Output neurons are the ones that send the signals out of the system. The neurons which have inputs and outputs within the system are known as hidden neurons.

The input is finally passed through the neural network again with updated weights and errors, if any, are calculated again. This process is repeated until the error is acceptable. The process is continued for all the training data. Lastly, the test data are used to verify the non-linear relationship between the input and output data sets [9].

## IV. ANN Training Algorithms

To design an ANN requires a training algorithm. During the training the process, weights and biases are adjusted to minimize the error to attain a high-performance solution. During the training and at the end of training, mean squared error is calculated between desired outputs and target outputs. There are numerous training algorithms used in ANN. Predicting which of the training algorithms would be the fastest for any problem is challenging [10]. Largely, it depends on some factors; number of hidden layers, weights and biases, aimed error at learning and application area. Generally, training algorithms are based on Taylor's series, but the presence of multiple points including local minima makes the analyses difficult and complicated. The fundamental principles of the used training algorithms are given below.

### *A. Levenberg-Marquardt (LM)*

LM like QN training algorithm is based on approaching second-order training speeds without having the computation of Hessian matrix [11]. The benefit of LM training method is its convergence about minimum and giving more precise results. The LM training algorithm was found to be the fastest algorithm, while it requires more memory as compared to the other training algorithms [12].

### *B. Quasi Newton (QN)*

The convergence of Newton's algorithm is faster than conjugate gradient descent methods. However, the computation of second derivatives is complex in Newton's method. In order to overcome this computational complexity, the QN training method is developed based on Newton's method, because the QN dos not require the computation of second derivatives. The principle of QN algorithm is based on updating the Hessian matrix at each iteration. The update is calculated as a function of the gradient [13].

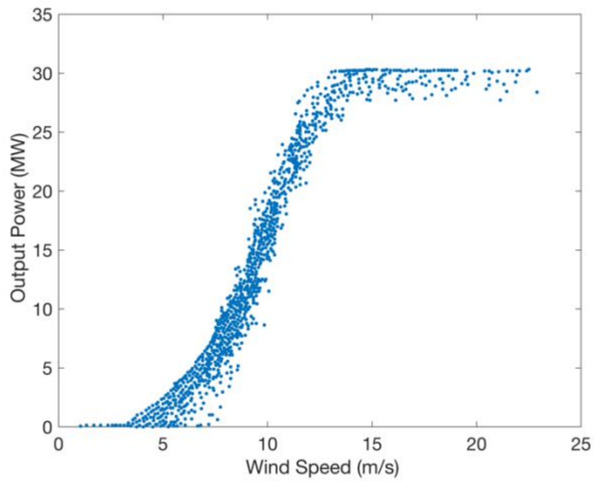


Figure 2. Power curve of actual data for a group of wind turbines at a wind farm adopted from technical literature and used in this study.

### C. Resilient Back Propagation (RBP)

In common gradient descent methods, due to the limiting influence of the slope of the sigmoid activation function, the size of the derivative decreases as exponentially with the distance between the weight and the output layer. As a result, the weights which are far away from the output layer are less modified and their learning is much slower as well. However, using RBP training method, the size of the weight-step only depends on the sequence of signs and not on the magnitude of the derivative. Hence, the entire network learns equally; the weights near the output layer have the equal chance to grow and learn as the weights near the output layer. RBP is very efficient training method with respect to both time and storage [14].

## V. Neural Network Framework For This Study

ANN is used to predict the wind power curve of a group of turbines in this study. MATLAB deep learning toolbox is used for the simulation. In the design of ANN, training and test sets have been prepared using some data obtained from technical literature [15] as shown in Fig.2. Three different training algorithms namely: LM, QN and RBP have been used in this study. The sigmoid activation function is used in each training algorithm. All the data in training and test sets have been normalized between 0 and 1. Feed forward neural networks have been used in all cases. The following steps are used for each training algorithm: (a) making the database for training and test data for each training algorithm (a) normalization of the data (c) training of the ANN using each training algorithm (d) testing of the trained data (v) using the trained ANN for prediction. Three-layer feed-forward structure ANN is implemented with an input layer, a hidden layer and an output layer. The designed ANN has one input (wind speed (m/s)) and one output (output power (MW)) neurons. Number of neurons in the hidden layer exhibiting the least Mean Square Error (MSE) and best fit for each training algorithm were selected during the simulation. There is no known method to find the number of neurons in the hidden layer. Therefore, selecting the number of neurons in the hidden layer can take considerable amount of time.

## VI. Results And Discussion

MSE values for training data are given for each training algorithm in Fig. 3. It also shows the accuracy of each training algorithm depending on the number of neurons in the hidden layer. MSE is a good measure to get information about learning performance. The iterations were continually performed until minimum MSE is achieved. It is evident from Fig. 3 that the minimum MSE was obtained with LM training algorithm having 10 neurons in the hidden layer, followed by RBP with 8 neurons in the hidden layer and then the QN with 8 neurons in the hidden layer. RBP and QN showed almost the same value of MSE, however, MSE of QN is slightly more than RBP. There is no known method to find the number of neurons in the hidden layer. Therefore, selecting the number of neurons in the hidden layer can take considerable amount of time.

The curves of actual data and the ANN predicted results of output power verses wind speed of a group of turbines for three training methods are shown in Fig. 4. In Fig. 4 (a), the prediction results of LM training method are given. It is evident that the actual and predicted results are in good agreement for LM training method. Fig. 4 (b) shows the prediction results of QN training method versus the actual data. The actual and predicted results were found closer for QN method; however, some small errors are present between the curves. Lastly, Fig. 4 (c) displays the prediction results by the RBP training method. Likewise, QN results, the RBP prediction results verses the actual data were found closer with small errors between the curves.

It has been evidently seen that the LM training method has provided the best results for the output power verses the wind speed of the data of a group of wind turbines in terms of MSE and curve fitting.

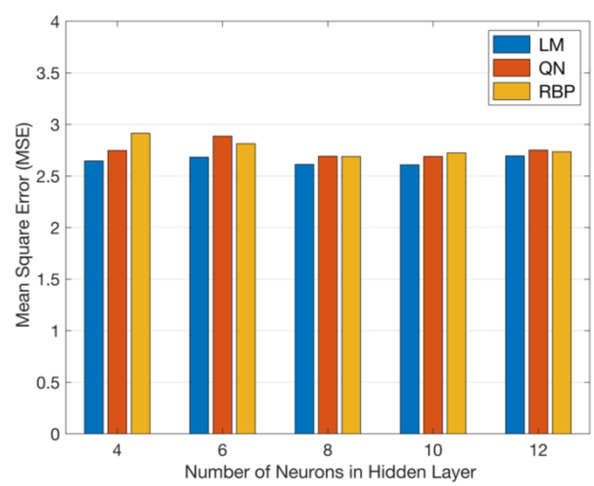


Figure 3. MSE values of each training algorithm for different number of neurons in the hidden layer.

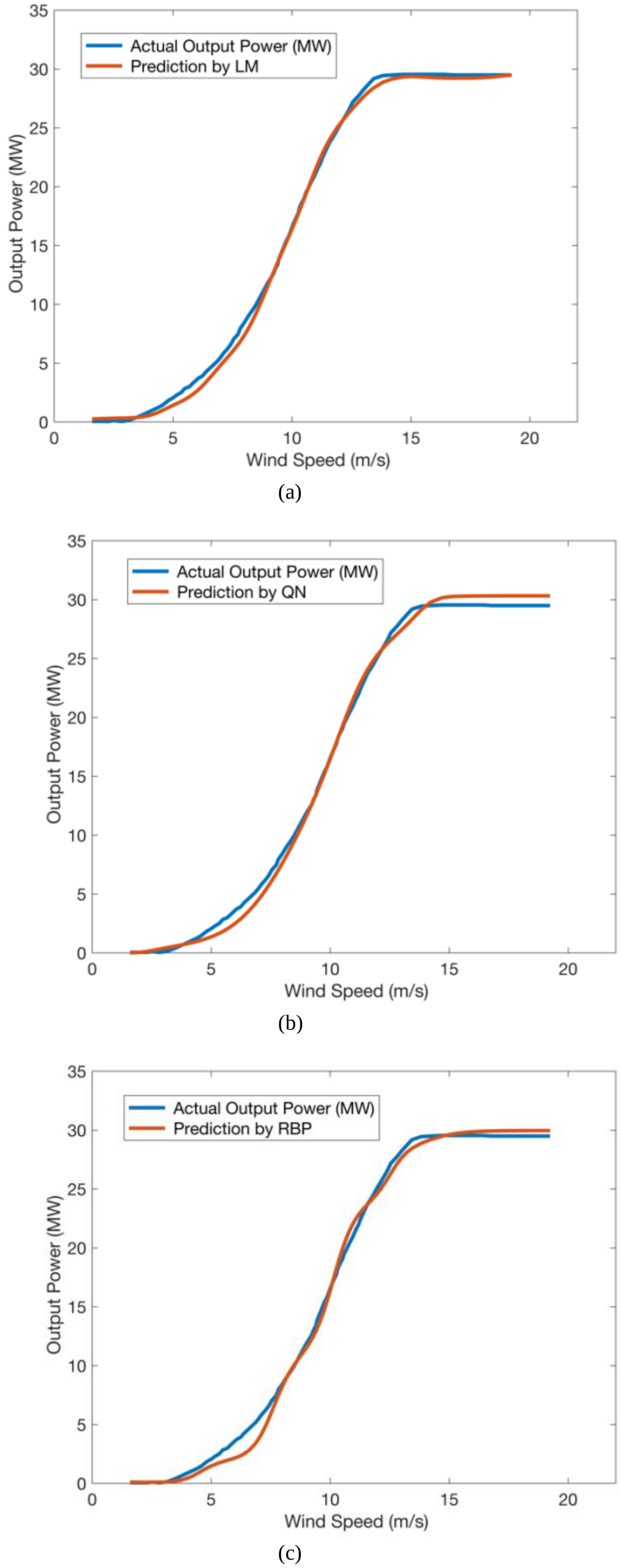


Figure 4. Actual and ANN predicted results of output power verses wind speed for (a) LM training algorithm (b) QN training algorithm and (c) RBP training algorithm.

However, QN and RBP training algorithms have also predicted the results well with an acceptable amount of error. Indeed, it is very challenging to generalize which training algorithm will be the fastest and accurate one for any given problem. For the present work, LM training method has produced the best results than others.

## VII. CONCLUSION

The present study has used the ANN model to predict the power curve of wind turbines. The experimental data was selected from technical literature and MATLAB deep learning toolbox was used to perform the ANN. The effect of three training algorithms on learning performance of the neural networks on the prediction of output power curve was studied. Furthermore, the effect of number of neurons in the hidden layer of each training algorithms was also examined. The results predicted by ANN showed a good agreement with the actual data for each training algorithm. However, the LM training algorithm has shown the best results with lesser MSE and better fit than the other algorithms. Consequently, a considerable savings of cost and time were gained by using ANN methodology.

## REFERENCES

[1] F. Pelletier, C. Masson and A. Tahan, “Wind turbine power curve modelling using artificial neural network,” *Renewable Energy*, vol. 89, pp. 207-214, 2015. doi:10.1016/j.renene.2015.11.065.

[2] M. Carolin Mabel and E. Fernandez, “Analysis of wind power generation and prediction using ANN: a case study,” *Renewable Energy*, vol. 33, pp. 986-992, 2007. doi:10.1016/j.renene.2007.06.013.

[3] S. Li, D. C. Wunsch, E. O’Hair and M. G. Giesselmann, “Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation”, *J. Solar Energy Eng*., vol. 123, pp. 327-332, 2001. doi:10.1115/1.1413216.

[4] J.F. Manwell, J.G. McGowan and A.L. Rogers, Wind energy explained: theory, design and application, Wiley, New York, 2002.

[5] S. Haykin, Neural networks, Macmillan College Publishing Company, New York, pp. 138, 1994.

[6] R. K. Jain, V. K. Jain and P. K. Karla, “Modelling of abrasive flow machining process: a neural network approach”, *Wear*, vol. 231, pp. 242-248, 1999. doi:10.1016/S0043-1648(99)00129-5.

[7 T. Cakar and I. Cil, “Artificial neural networks for design of manufacturing systems and selection of priority rules”, *Int. J. Comput. Integrated Manuf.*, vol. 17(3), pp. 195-211, 2004. doi:10.1080/09511920310001607078.

[8] N. Atlinkok and R. Koker, “Neural network approach to prediction of bending strength and hardening behaviour of particulate reinforced (Al-Si-Mg)-aluminium matrix composites”, *Mater. Design*, vol. 25, pp. 595-602, 2004. doi:10.1016/j.matdes.2004.02.014.

[9] P. D. Wasserman, Neural computing, Van Nostrand Reinhold, New York, 1989.

[10] E. Charpentier and J. J. Laurin, “Sectorial direction finding antenna array with a MLP beam former”, in *Proceedings of the IEEE Int. Symposium on antennas and propagation*, Montreal, Canada, vol. 4, pp. 2270-2273, 1997. doi:10.1109/APS.1997.625422.

[11] M. T. Hagan, H. B. Demuth and M. H. Beale, Neural network design, PWS Publishing, Boston (MA), 1996.

[12] M. T. Hagan and M. B. Menhaj, “Training feed forward networks with the marquardt algorithm”, in *IEEE Transactions on Neural Networks*, vol. 5, pp. 989-993, 1994. doi:10.1109/72.329697.

[13] R. Battiti, "First- and second-order methods for learning: between steepest descent and newton's method," *Neural Computation*, vol. 4, pp. 141-166, 1992. doi:10.1162/neco.1992.4.2.141.

[14] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," in *IEEE Int. Conf.*

*on Neural Networks*, San Francisco, USA, vol. 1, pp. 586-591, 1993. doi:10.1109/ICNN.1993.298623.

[15] V. Sohoni, S. C. Gupta and R. K. Nema, “A critical review on wind turbine power curve modelling techniques and their applications in wind based energy systems”, *J. Energy,* vol. 2016, pp. 1-18, 2016. doi:10.1155/2016/8519785.