Performance Predictions Of Wind Turbine Blades Using Artificial Neural Network Method

Ziaul Huque[1,3], Isaac K. Ofuafor[1,3], Donald Harby[1,3], Raghava R. Kommalapati[2, 3]

[1]Department of Mechanical Engineering
[2]Department of Civil and Environmental Engineering
[3]Center for Energy and Environmental Sustainability
Prairie View A&M University
TX 77446, USA

Abstract

Nowadays the Blade Element Momentum (BEM) theory is used to determine a given element's airfoil section performance coefficient. The motivation behind this work is to improve the current method of determining the airfoil section performance coefficients and to determine these coefficients where little or no experimental data exists such as angles of attack of stall, post stall and at very low Reynolds numbers (Re). Artificial Neural Network (ANN) method is used that can supply high quality airfoil sectional lift and drag coefficients over a wide range of angle of attacks and Reynolds numbers. Two wind turbine blades (E387 and FX63137) are analyzed using BEM theory and Artificial Neural Network method. It is found that the predicted performance from the BEM theory and the Artificial Neural Network is very sensitive to the angle of attack and Reynolds number of the elemental airfoil sections.

Keywords— Wind turbine, Artificial Neural Network, Stall, Angle of attack, airfoils.

1. Introduction

Amidst rising oil prices, it is very important to reduce the US crude oil import by one-third by 2025 through expanded exploration of crude oil and natural gas and investment in alternative energy sources including nuclear energy, biofuels and wind energy. The volatility in crude oil prices has led to renewed drive for renewable energy sources. Wind power is a renewable energy source that is clean, environmentally friendly and helps the US meet its energy needs and also provide economic benefits.

Currently, wind power accounts for 3% of electricity generated in the US with individual turbine capacity of around 4MW. In spite of these developments, it is still expensive to analyze the aerodynamic characteristics and performance parameters of wind turbine rotors through wind tunnel experiments or practical site experiments to improve the design and efficiency of wind turbines rotors. It is imperative that less costly means be devised to analyze the aerodynamic and performance parameters of the wind turbine blades.

Wind tunnel experiments have limitations on getting data on minimum Reynolds numbers (Re) and angle of attacks (α). The objective of this work is to use Artificial Neural Network (ANN) models of blades to analyze performance characteristics at untried angle of attacks and Reynolds number for wind turbine blade designs. The current work uses global data fit type. The Artificial Neural Network model is used to approximate the lift and drag coefficients of the wind turbine blades (E387 and FX63137) from wind tunnel experimental data.

The Blade Element Momentum (BEM) theory is a mathematical model and also one of the most commonly used models in engineering for the fluid dynamics design of rotor blades and evaluation of wind turbine performances. In designing the wind turbine blade this model enables the choosing of geometric characteristics of the turbine such as the aerodynamic airfoils, rotor radius, chord length, pitch and twist angles. This also includes the evaluation of the forces acting on the blade, the torque and power at the rotor shaft. The mathematical model also enables turbine performance evaluation at wide range of wind speeds.
The experimental data used by this paper has a small range of angle of attack from -11^o to 20^o because of wind tunnel limitations. The flat plate theory enables the calculation of the peak post-stall data using the Viterna and Corrigan method [1]. The combined experimental and flat plate theory data when modeled using the Artificial Neural Network modeling technique could be used to simulate the blade design parameters and thus, the power coefficient, torque and normal and tangential forces acting on the blade. The peak rotor power and post-peak power is important in the predictive design of constant-speed and variable speed stall-regulated rotors.

Coefficient of power (C_p) is calculated using the equation:

$$C_P = \frac{P}{\frac{1}{2}\rho_\alpha A U_\alpha^2} \qquad (1)$$

Here, P = Power, ρ_α = Free stream flow density, A = Area of disk across the propeller plane, U_α = Free stream velocity.

Experimental studies lack a method for determining the blade's angle of attack (α) distribution so that normal and tangential force coefficients (C_n and C_t) acquired from chord-wise pressure measurements can be converted into lift and drag coefficients (C_l and C_d) for engineering calculations.

By definition, 'stall' means a particular angle of attack when the blade does not produce any lift force, therefore we do not get any power from wind turbine. The region just after stall condition is known as 'post-stall'.

References [2-4] have concentrated on the determination of the angle of attack distribution from experimental data. Under the sponsorship of the National Renewable Energy Laboratory (NREL) the Lifting-Surface Wind Turbine (LSWT) performance-prediction methodology [5], provided a unique capability for deriving angle of attack. From Unsteady Aerodynamic Experiment (UAE) data of C_n and C_t could then be converted into values of C_l and C_d using angle of attack distributions derived from LSWT. Through an iterative process, agreement was achieved between UAE measured and LSWT-predicted C_n and C_t radial distributions [1].

This agreement yielded angle of attack distributions compatible with the measured post-stall 3D (three dimensional) aerodynamic characteristics. Unlike blade-element momentum (BEM), the LSWT methodology accounts for the induced effects of the blade configuration and those from the span-wise distribution of trailing vorticity in calculating the angle of attack distribution [2].

After an angle of attack of 20^o the lift/drag ratio for the five radial locations at which pressure measurements were acquired essentially followed simple flat plate theory [6]. The difficulty of relating BEM-predicted angle of attack distributions to the post-stall 3-D aerodynamics and measured power are demonstrated in [2-4]. This finding pointed to the need for a global post-stall approach, as for instance, previously developed by Viterna [7] and [8] for generating post-stall C_l and C_d based on both airfoil and blade specific stall characteristics.

These generated post-stall data are combined with the experimental data to create Artificial Neural Network model that are used to predict the lift and drag coefficients at untried location of the Reynolds number and angle of attack (untried sites) to predict peak power and other rotor performance characteristics.

The Artificial Neural Network is a good tool and provides good predictive values. The program used is the "Matlab Neural Network Toolbox" that was used to construct the model for the Artificial Neural Network [9].

2. Generating Global Data

Since the experimental data of lift coefficient (C_l) and drag coefficient (C_d) are available for mainly pre-stall angle of attacks, flat plate theory was used to calculate the post-stall C_l and C_d from the given experimental data according to Viterna and Corrigan method [1,7]. Following are the set of equations used [1, 7] to get global data:

Drag coefficient:

$$C_D = B_1 Sin^2 \propto + B_2 Cos \propto, \quad \propto = 15^o to 90^o \qquad (2)$$

$$C_{d\ max} = 1.11 + 0.018\, AR, \quad \propto = 90^o \qquad (3)$$

where: $B_1 = C_{d\ max}$ (4)

$$B_2 = \frac{C_{d\ stall} - C_{d\ max} sin^2 \propto_{stall}}{\cos \propto_{stall}} \qquad (5)$$

Lift coefficient:

$$C_l = A_1 sin2 \propto + A_2 \frac{cos^2 \propto}{\sin \propto}, \propto = 15^o to 90^o \qquad (6)$$

where: $A_1 = \frac{B_1}{2}$ (7)

$$A_2 = (C_{l\ stall} - C_{d\ max} sin \propto_{stall} \ cos \propto_{stall}) \frac{sin \propto_{stall}}{cos^2 \propto_{stall}} \qquad (8)$$

An initial angle of attack (α_{stall}) with its associated lift ($C_{l,stall}$) and drag ($C_{d,stall}$) coefficients with the blade aspect ratio (AR) is required. The C_l/C_d will not agree with the flat plate theory if the C_l/C_d at the initial angle of attack is not satisfied.

3. Artificial Neural Network Method

The brains of biological creatures have long been an area of intense study to understand its working mechanism. At beginning of the 20th century it was found that neurons are the structural constituents of the brain. The neurons interact with each other through synapses, and are connected by axons (transmitting lines) and dendrites (receiving branches). It is estimated that there are on the order of 10 billion neurons in the human cortex, and about 60 trillion synapses. Although neurons are 5~6 orders of magnitude slower than silicon logic gates, they are organized such that the brain is capable of performing certain tasks (for example, pattern recognition, and motor control etc.) much faster than the fastest digital computer nowadays. Also, the energetic efficiency of the brain is about 10 orders of magnitude lower than the best computer today. So it can be said, in the sense that a computer is an information-processing system, the brain is a highly complex, nonlinear, and efficient parallel computer [10].

Neural Networks (NN) are computational systems inspired by the biological brain in their structure, data processing and restoring method, and learning ability. More specifically, a neural network is defined as a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for future use by resembling the brain in two aspects: (a) Knowledge is acquired by the network through a learning process; (b) Inter-neuron connection strengths known as synaptic weights (or simply weights) are used to store the knowledge. They have found numerous applications in science and engineering, from biological and medical sciences, to information technologies such as artificial intelligence, pattern recognition, signal processing and control, and to engineering areas as civil and structural engineering.

Figure 1 shows a typical neuron where the sum of the weighted input and bias are transferred through the function f. The weight and the bias are the design variables that are adjusted so the neuron could be trained.

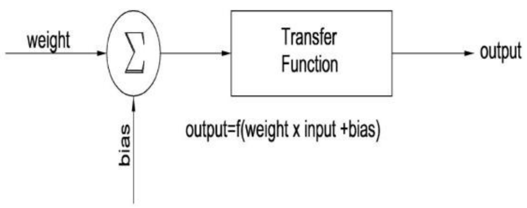

Figure 1. A typical neuron [12]

The three most widely used transfer functions are the sigmoid (LOGSIG), hyperbolic tangent (TANSIG) and the linear functions (PURELIN). These functions are shown in Figure 2 below. The sigmoid transfer function takes the input, which can have any value between plus and minus infinity, and squashes the output into the range 0 to 1, while the hyperbolic transfer function squashes the same input to -1 to 1. Neurons of the linear transfer function type are used as linear approximations in linear filters.

Figure 2 below is the graphical depiction of the transfer functions of the neural network shown mathematically;

Linear function (purelin): $f(n) = n$

Sigmoid function (logsig): $f(n) = \dfrac{1}{1+e^{-n}}$

Hyperbolic tangent function (tansig): $f(n) = \dfrac{2}{1+e^{-2n}}$

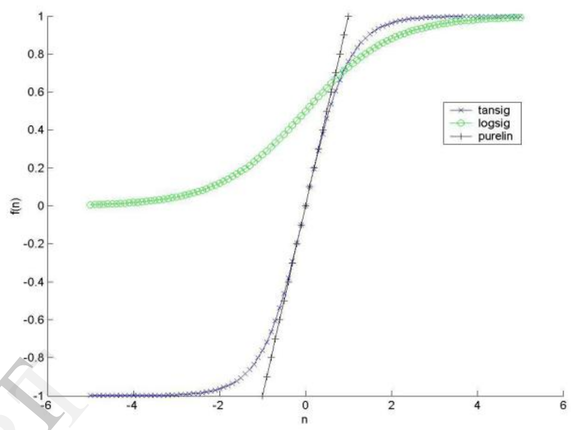

Figure 2. Transfer functions.

Note how the structure is inherently parallel in nature. Even though the neurons and structure are very simple, if the numbers of neurons in the internal layers are large enough, the network can be trained to represent complex non-linear systems. These large NN's are good at approximating practically any non-linear function [9].

The number of inputs and outputs a NN can have is only limited by the computer memory available. Therefore, a standard feed-forward NN of sufficient size should be able to approximate most models. Normally the NN is trained to approximate a function so that a set of inputs leads to a set of target outputs.

Neural networks are good at fitting functions and recognizing patterns. A fairly simple neural network can fit any practical function. The Neural Network Toolbox software provides a flexible network object type that allows many kinds of networks to be created and then used with functions such as *init*, *sim*, and *train*.

The networks have an object-oriented representation which allows you to define various architectures and assign various algorithms to these architectures.

3.1. Types of Neural Networks:

As simplified models of the biological brain, ANNs have lots of variations due to specific requirements of their tasks by adopting different degree of network complexity, type of interconnection, choice of transfer function, and even differences in training method.

According to the types of network, there are Single Neuron network (1-input, 1-output, and no hidden layer), Single-Layer NN or Perceptron (no hidden layer), and Multi-Layer NN (1 or more hidden layers). According to the types of inter-connection, there are Feed-Forward network (values can only be sent from neurons of a layer to the next layer), Feed-Backward network (values can only be sent in the opposite direction, i.e. from the present layer to the previous layer), and Recurrent network (values can be sent in both directions).

Feed-Forward Multi-Layer Neural Network

An example of feed-forward multi-layer neural network is shown in Figure 3, where the numbers of input and output are 3 and 2 respectively, and there are two hidden layers with 5 neurons in the first hidden layer, and 3 neurons in the second hidden layer.

As shown in Figure 3, in the j^{th} layer, the i^{th} neuron has inputs from the $(j\text{-}1)^{th}$ layer of value $x_k^{j-1} (k = 1,..., n_{j-1})$, and has the following output

$$x_i^j = f(r_i^j)$$

where $r_i^j = \sum_{k=1}^{n_{j-1}} w_{ki}^{j-1} x_k^{j-1} - b_i^j$

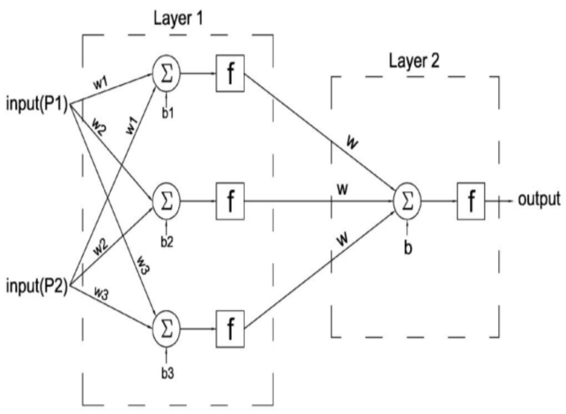

Figure 3. A typical 2 layer feed forward neural network

in which w_{ki}^{j-1} is the weight between node k of the $(j\text{-}1)^{th}$ layer and node I of the j^{th} layer, and b_i^j is the bias (called the threshold). The above relation can also be written as:

$$r_i^j = \sum_{k=0}^{n_{j-1}} w_{ki}^{j-1} x_k^{j-1}$$

where $x_0^{j-1} = b_i^j$ and $w_{0i}^{j-1} = -1$, or $x_0^{j-1} = -1$ and $w_{0i}^{j-1} = b_i^j$ [11]

Though the neurons and structure are very simple, if the numbers of neurons in the internal layer are large enough, the network can be trained to represent complex non-linear systems. The number of inputs and outputs a neural network can have is only limited by the computers' memory available. The neural network is trained to approximate a function so that a set of inputs lead to a set of target outputs.

3.2. Neural Network Simulation

In the present work a two layer feed forward network (newff) with 40 sigmoid hidden neurons were used. The default training parameter used is the Levenberg-Marquardt back-propagation algorithm. The data validation and testing was done by random division of the sample: 60% for training, 20% for validation and 20% for testing. Training and learning functions are mathematical procedures used to automatically adjust the network's weights and biases. The training function dictates a global algorithm that affects all the weights and biases of a given network. The learning function can be applied to individual weights and biases within a network [4]. The gradient descent weight/bias learning function (learngd) was used for the learning process while the training functions used for the neural network models were the Levenberg-Marquardt (trainlm) and the BFGS quasi-Newton backpropagation (Trainbfg) parameters. The data used for the neural network simulations were those for the E387 and FX63137 airfoils.

Figure 4 is the best linear fit regression plots of the training, validation and tests of the output and target data for E387 airfoil. All three regression plots are approximately equal for the training, validation and tests respectively, which show a close relationship between the output and target data.

Figure 5 is the performance plot of the Mean Squared Error during iteration of the process. This figure shows the best validation performance with MSE of 0.00015327 occurred at iteration 240. The iteration continues to an epoch of 246 until there is no improvement in the MSE. The training function outputs the trained network and a history of the training performance as shown in Figure 5 and the Mean Squared Errors are plotted against the iterations.

Figures 6 and 7 below are the artificial neural network model of the lift and drag coefficients of the E387 and FX63137 airfoils respectively. Figures 6 and 7 were made using the learning (learngd) and the training (trainlm) functions of the neural network toolbox.

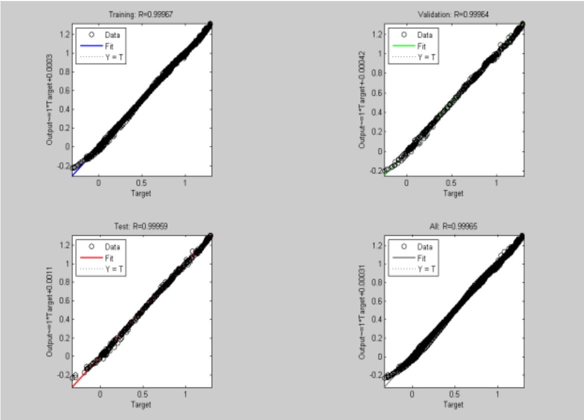

Figure 4. The regression plots

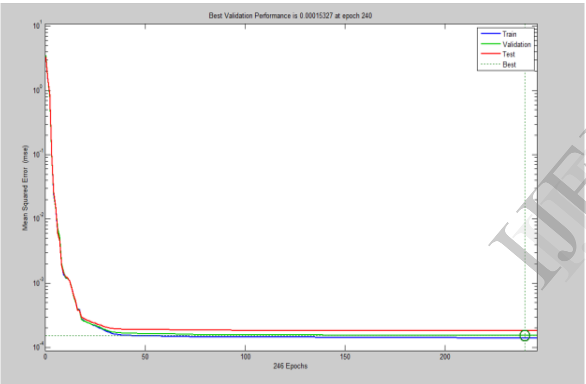

Figure 5. The performance plot of MSE and epochs

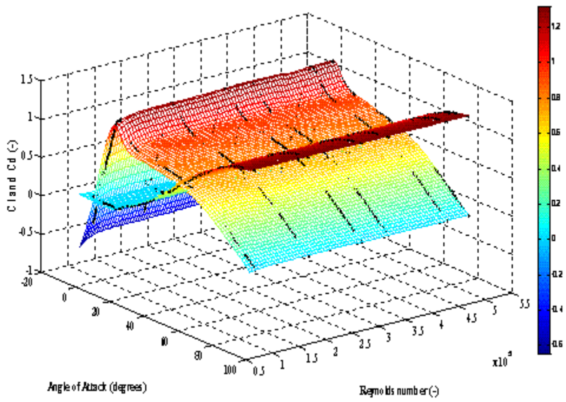

Figure 6. Neural network model of C_l and C_d of the E387 airfoil

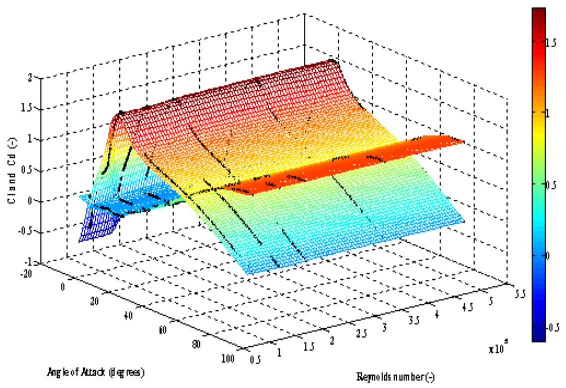

Figure 7. Neural network model of C_l and C_d of the FX63137 airfoil

Each of the figures shows two surfaces, one for the lift coefficient and the other for the drag coefficient. The data points of the lift and drag coefficients are represented by the black straight lines and the colored grids are the prediction results. This is a reasonably good representation of the shape of the data points including the stall crest of the lift coefficients at various Reynolds numbers. This model now can be used for prediction at untried sites. The Figures also show that where the crest of the models at the stall lift coefficient is, the predictions are level across the range of Reynolds numbers. As a result the Neural Network model might be accurate at spatial separation locations.

Comparisons of predicted outputs of Cl and Cd for both the airfoils with the different transfer functions were performed. The results from the trainlm function are found to more closely mirror the experimental and calculated post-stall data and as such will be used in subsequent inputs of the lift and drag coefficients.

4. Application of the Artificial Neural Network Model to BEM Theory:

In this work the Blade Element Momentum theory is used to predict the performance of wind turbines using Artificial Neural Network model for untried sites of the wind tunnel experimental data. The performance of wind turbines is predicted using the Blade Element Momentum theory using the WT_Perf code [12] which was originally developed by Oregon State University as the PROP code, but has since been rewritten and modernized with added new functionalities and algorithms by the staff at National Wind Technology Center (NWTC). A hypothetical wind turbine is described below:

Blade radius is 5m, hub radius is 0.2m with no twist and no taper. The blade radius is divided into 20 dimensionless equal elements. With aerodynamic data assumed for standard temperature and pressure at sea level the air density is taken as 1.23 kg/m3 and kinematic viscosity is taken as $1.464\text{x}10^{-5}$ m^2/s. The simulations produced the power coefficient, torque coefficient, thrust coefficient, power, flap bending moment and the thrust of the rotor at incremental wind speed. Table 4 below shows the C_l and C_d from untried sites of the E387 neural network model.

Figure 8 shows the sensitivity of the lift coefficients to Reynolds number at the pre-stall angle of attack of 10^o and post-stall angle of attack of 15^o from the Neural Network model of the E387 airfoil. At $5\text{x}10^4$ Reynolds number the lift coefficient is larger than the experimental lift coefficient at 10^5 Reynolds number which showed that the model optimizes the observed data at untried locations.

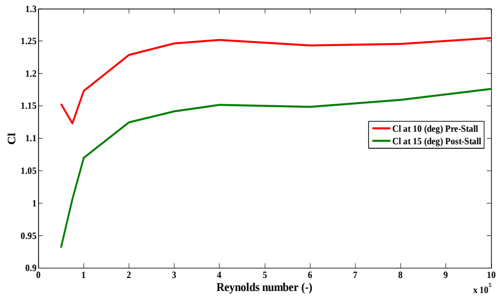

Figure 8. Plots of C_l at pre-stall and post-stall angles of attack of the E387 airfoil

The power coefficient of the E387 airfoil using the design parameters above is 0.355 and conditions leading to this are 7 m/s wind speed and 100 rpm rotor speed as shown in Figure 9. The same design parameters were applied to the FX63137 airfoil. A power coefficient of 0.413 at 8m/s and 100 rpm rotor speed was observed. Figure 10 shows the power generated when the E387 airfoil was simulated with the wind turbine performance code across various wind speeds. The power is fairly constant at 100 rpm at the different wind speeds.

From the blade element data of the performance output at maximum power coefficient condition, Figure 11 shows the distribution of the thrust coefficient (dF), torque coefficient (dT) and power coefficient (C_p) along the blade length while Figure 12 depicts the thrust and torque acting on the rotor blade. This is the distribution of the power, forces and torque on the rotor along the length of the blade. These parameters increase gradually along the rotor radius from the hub to the tip where there is dramatic drop as a result of tip loss.

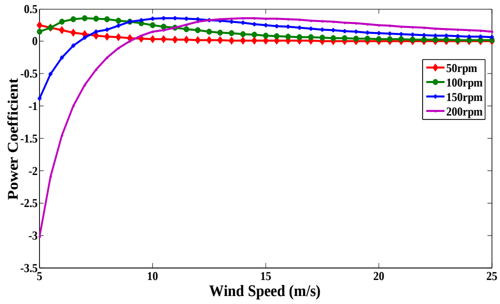

Figure 9. Power coefficient versus wind speed of the E387 airfoil

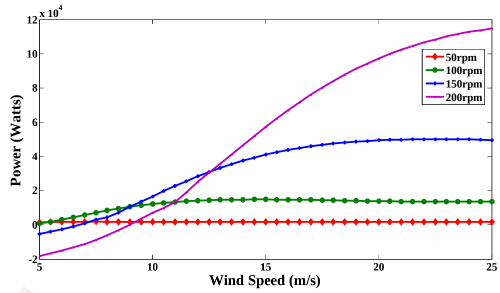

Figure 10. Power versus Wind speed

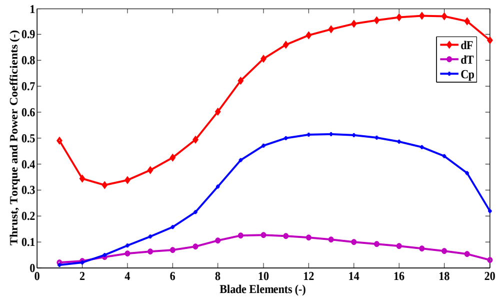

Figure 11. Plot of Thrust (dF), Torque (dT) and Power coefficients (C_p)

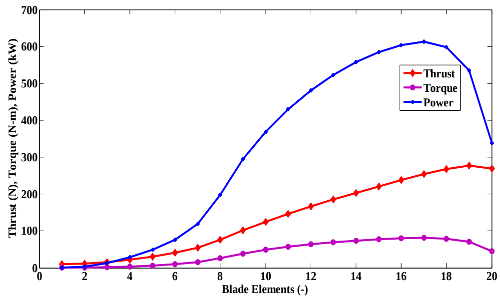

Figure 12. Plots of Thrust, Torque and Power

5. Conclusions

Several innovative techniques have been developed for predicting the lift and drag coefficients of wind turbine blades over a wide range angle of attacks and Reynolds numbers. These unique Artificial Neural Network modeling methods optimize the target data and accurately predict or simulate the lift and drag at untried sites. The feed forward neural network proved effective in simulating the experimental data and it is a tool with immense potential in solving or simulating a wide range of problems. With the right combination of learning and training parameters more could be done with designing wind turbine blades and simulating the performance parameters using Artificial Neural Network modeling. Depending on the sensitivity of the experimental data all Artificial Neural Network models in this research produced close approximation models. The Artificial Neural Network models increased the accuracy of the BEM theory through accurate prediction of the lift and drag coefficients.

Acknowledgments

This work is supported by the National Science Foundation (NSF) through the Center for Energy and Environmental Sustainability (CEES), a CREST Center, award no. 1036593.

References

[1] Tangler, J.L., and Kocurek J. D., "Wind Turbine Post-Stall Airfoil Performance Characteristics", Guidelines for Blade-Element Momentum Methods, *NREL/CP-500-36900*, October 2000.

[2] Tangler, J.L., "Insight into Wind Turbine Stall and Post-Stall Aerodynamics", *AWEA 2003*, also *Wind Energy*, vol. 7, Issue 3, pp: 247-260, www3.interscience.wiley.com, July/September 2004.

[3] Lindenburg, C., "Stall Coefficients: Aerodynamic Airfoil Coefficients at large Angles of Attack", *ECN RX 01-004*

[4] Gerber, B., Tangler, J.L., Duque, E., and Kocurek, J.D., "Peak and Post-Peak Power Aerodynamics from NASA Ames Phase VI Wind Turbine", Special Topic Conference The Science of making Torque from Wind, TU Delft, The Netherlands, April 2004.

[5] Kocurek, D., "Lifting Surface Performance Analysis for Horizontal Axis Wind Turbines", *SERI/STR-217-3163*, 1987

[6] Timoshenko, S., and Woinowsky-Krieger, S., "The theory of plate and shells", McGraw-Hill, New York, 1959

[7] Viterna, L.A., and Corrigan, R.D., "Fixed Pitch Rotor Performance of Large Horizontal Axis Wind Turbines", DOE/NASA Workshop on Large HORIZONTAL Axis Wind Turbines, Cleveland, Ohio, July 1981.

[8] Viterna, L.A., and Janetzke, D.C., "Design of a Tapered and Twisted Blade for the NREL Combined Experiment Rotor", *NASA TM-82944*, September 1982.

[9] Beale, M.H., Hagan, M.T., Demuth, H.B., "Neural Network Toolbox 7 User's Guide", The MathWorks, Inc, 3 Apple Hill Drive, Natick, MA 0176-2098, 2010.

[10] Liu, Y., "Efficient Methods for Structural Analysis of Built-Up Wings", PhD Dissertation in Aerospace Engineering, Virginia Polytechnic Institute and State University, 2000.

[11] Harby, D., "Parametric and Optimal Design of Modular Machine Tools", PhD Thesis, Graduate School, University of Missouri-Columbia, 2007.

[12] BEM theory, 2011 from: wind.nrel.gov/designcodes/simulators/wtperf