This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Advanced Algorithms for Wind Turbine Power Curve Modeling

M. Lydia, A. Immanuel Selvakumar, *Member, IEEE*, S. Suresh Kumar, *Member, IEEE*, and G. Edwin Prem Kumar

***Abstract*—A wind turbine power curve essentially captures the performance of the wind turbine. The power curve depicts the relationship between the wind speed and output power of the turbine. Modeling of wind turbine power curve aids in performance monitoring of the turbine and also in forecasting of power. This paper presents the development of parametric and nonparametric models of wind turbine power curves. Parametric models of the wind turbine power curve have been developed using four and five parameter logistic expressions. The parameters of these expressions have been solved using advanced algorithms like genetic algorithm (GA), evolutionary programming (EP), particle swarm optimization (PSO), and differential evolution (DE). Nonparametric models have been evolved using algorithms like neural networks, fuzzy c-means clustering, and data mining. The modeling of wind turbine power curve is done using five sets of data; one is a statistically generated set and the others are real-time data sets. The results obtained have been compared using suitable performance metrics and the best method for modeling of the power curve has been obtained.**

***Index Terms*—Data mining, differential evolution (DE), particle swarm optimization (PSO), power curve modeling.**

I. Introduction

WIND energy holds a promising answer to the world which is threatened by an energy crisis and other related environmental disasters. To make wind energy a reliable source of energy, efficient and accurate models for monitoring and forecasting of wind power is the need of the hour.

Wind power generated from a wind farm chiefly depends on the wind speed and direction at 50–100 m from ground level [1]. The conversion of available wind power into actual power varies nonlinearly due to the transfer functions of the electrical generators. The relationship between the wind speed and generated power is depicted by the wind turbine power curve. The wind turbine power curve is essentially comprised of three regions, as shown in Fig. 1. The first region below a minimum speed threshold known as the cut-in speed (u_c) has zero output. The second region, which spans until the machine reaches a nominal output power, marks rapid growth until the rated speed (u_r), and the third region gives a constant output, until the cutout speed (u_s) is attained. The cut-in speed is the minimum speed required for power generation and the cut-out speed is the maximum speed beyond which the generated power remains constant.

The theoretical power obtained from wind is given by

$$P = 0.5\rho\pi R^2 C_p(\lambda, \beta) u^3 \tag{1}$$

where P is the power captured by the rotor of a wind turbine, ρ is the air density, R is the radius of the rotor determining its swept area, C_p is the power coefficient, β is the blade-pitch angle, λ is the tip-speed ratio, and u is the wind speed [2].

Wind turbine power curves primarily indicate the performance of turbines. Hence developing accurate models for power curves is a very important area of research. As the wind turbine power curves under normal conditions are made available by the turbine manufacturers, any anomalies can be effectively detected by monitoring them. An accurately modeled power curve can serve as a tool for prediction too. Since wind energy is stochastic in nature, prediction models for forecasting wind power are required. Power curves can be used as a reference for predicting the output power of a turbine if the forecasted wind speed is available. Power curves modeled for many turbines can help in simulation of planned expansions in a power system with wind power [3]. Accurately modeled power curves can also serve as a basis of comparison between the performances of available turbines, helping the customers to make a proper choice based on their requirements.

Wind turbine power curve models can be classified into parametric and nonparametric models. In [2], the parametric least-squares model and the nonparametric k-nearest neighbor model showed greater accuracy than the other models. A parametric power curve which could be used for prediction, control, monitoring, and optimization of wind farm performance has also been developed in [4]. The estimation of the wind turbine power curve based on cluster center fuzzy logic modeling has been presented in [5]. It was also suggested that more accurate modeling could be obtained by increasing the number of clusters. Five different methods for wind turbine power curve modeling based on statistical tools have been compared in [6]. The best results were obtained when the fuzzy logic tool was used. A probabilistic model to characterize the dynamics of the output power of wind turbine has been discussed in [7]. The output power has been assumed to follow a normal distribution with a varying mean and a constant standard deviation. A critical analysis of the various methods used for modeling of wind turbine power curves like weibull distribution, method of least squares, and cubic spline interpolation have been compared in [8].

Manuscript received March 21, 2012; revised October 18, 2012; accepted February 07, 2013.

M. Lydia and A. Immanuel Selvakumar are with the Electrical and Electronics Engineering Department, Karunya University, Coimbatore, Tamil Nadu, India (e-mail: lydiaedwin.05@gmail.com; iselvakumar@yahoo.co.in).

S. Suresh Kumar is with the Electronics and Communication Engineering Department, Dr. NGP Institute of Technology, Coimbatore, Tamil Nadu, India (e-mail: sskpsg@gmail.com).

G. E. P. Kumar is with the Department of Information Technology, Karunya University, Coimbatore, Tamil Nadu, India (e-mail: edwinpremkumar@gmail.com).

Digital Object Identifier 10.1109/TSTE.2013.2247641

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

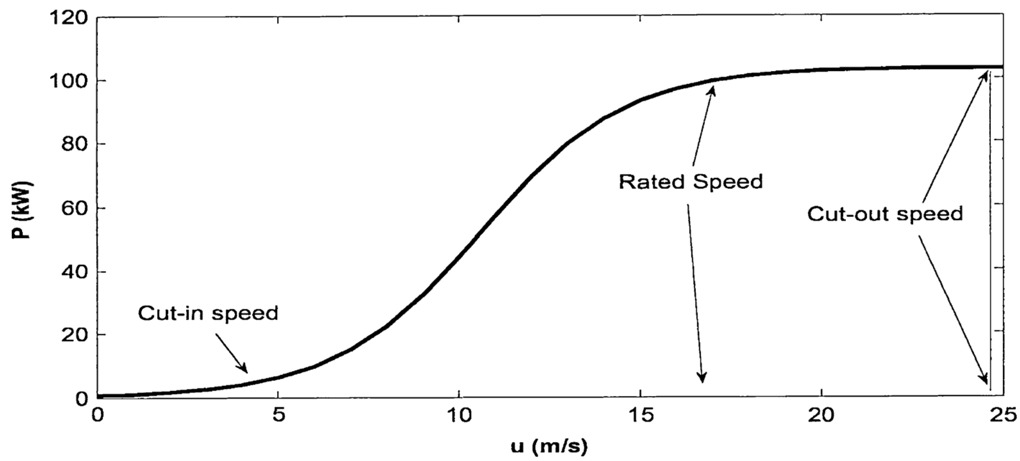


Fig. 1. Wind turbine power curve.

This paper has exploited the advantages offered by the modern nontraditional optimization techniques for modeling of wind turbine power curve. The previous works in this area except [2] and [4] have involved only traditional techniques. Solution of parametric expression using traditional techniques of optimization for huge datasets is a very complicated process. The application of nontraditional techniques enhances the accuracy and is very easy to implement.

This paper presents the implementation of advanced algorithms for modeling of wind turbine power curves. Parametric and nonparametric models of wind turbine power curves have been developed and their accuracy has been compared using the root mean squared error (RMSE) and mean absolute error (MAE) as performance metrics. Parametric models of the wind turbine power curve have been built using logistic expression and least squares algorithm. The optimum values of the parameters in the logistic expression have been solved using algorithms like genetic algorithm (GA), evolutionary programming (EP), particle swarm optimization (PSO), and differential evolution (DE). A linearized segmented model of the power curve has also been developed. Nonparametric models of wind turbine power curve have been developed using neural networks, fuzzy c-means algorithms, and data mining techniques. The models are developed using five sets of data. The first set is statistically generated based on a probabilistic model and the other sets are real-time data.

II. Power Curve Models

The wind turbine power curve can be modeled using parametric and nonparametric methods. In statistics, a parametric model or finite-dimensional model is a family of distributions that can be described using a finite number of parameters. These parameters are usually collected together to form a single k-dimensional parameter vector $\theta = (\theta_1, \theta_2, \theta_3, \ldots, \theta_k)$. In contrast, nonparametric models are characterized by parameters, which are in infinite-dimensional parameter spaces. The term nonparametric does not mean to imply that these models lack parameters, but that the number and nature of the parameters are flexible and not fixed in advance.

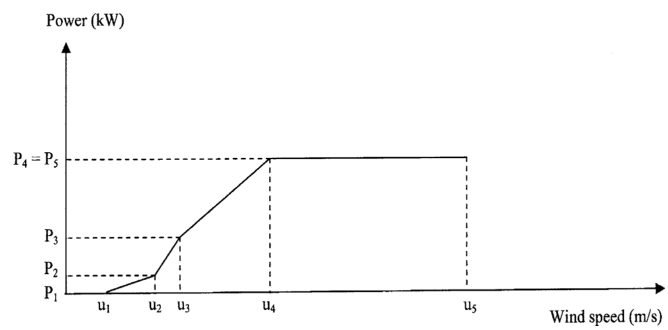


Fig. 2. Linearized segmented model.

A. Parametric Models

Parametric models of a wind turbine power curve are basically built on a set of mathematical expressions. The parameters of these expressions are found using advanced algorithms.

1) Linearized Segmented Model: A linearized segmented model of the power curve is obtained by approximating it to different linear sections as shown in Fig. 2 [9]. The piecewise approximation has been carried out using the equation of a straight line

$$P = m\,u + c \tag{2}$$

where P is the output variable, power of the wind turbine, u is the input variable, the wind speed, and the parametric vectors are defined by $\theta = (m, c)$, m being the slope of the segment and c a constant.

2) Four-Parameter Logistic Expression: The shape of the power curve can be approximated using a logistic expression with four parameters [2]. Fig. 3 shows the logistic curve, whose expression is given by

$$P = f(u, \theta) = a(1 + me^{-u/\tau}/1 + ne^{-u/\tau}). \tag{3}$$

Here $\theta = (a, m, n, \tau)$ is a vector parameter of the logistic expression that determines its shape.

In order to find out the value of the vector parameter, the modeling of the power curve using a four-parameter logistic expression can be defined as an optimization problem as follows:

$$\text{Objective function} : \min \sum_{i=1}^{N} [P_e(u_i) - P_a(i)]^2 \tag{4}$$

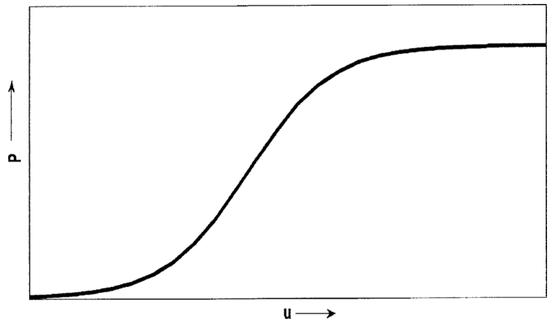


Fig. 3. Logistic curve.

where $P_e(u_i) = a(1 + me^{-u_i/\tau}/1 + ne^{-u_i/\tau})$ is the output power estimated through the four parameter logistic expression, $P_a(i)$ is the actual power, N is the number of data, and i is the index of the data. The following are the constraints:

$$\left.\begin{array}{l} a_{\min} \le a \le a_{\max}, m_{\min} \le m \le m_{\max} \\ n_{\min} \le n \le n_{\max}, \tau_{\min} \le \tau \le \tau_{\max} \end{array}\right\}. \tag{5}$$

3) Five-Parameter Logistic Expression: The logistic curve can also be approximated using five parameters. The application of the five-parameter logistic expression has been reported in [10] for biological applications. In this paper, the five-parameter logistic expression has been applied for the wind turbine power curve and the performance has been compared with the four-parameter logistic expression. The five-parameter logistic function is given by

$$P = f(u, \theta) = d + (a - d) \Big/ \left(1 + \left(\frac{u}{c}\right)^b\right)^g \tag{6}$$

where $\theta = (a, b, c, d, g)$ is the parameter vector of the five-parameter logistic and the range of the parameters are restricted so that $c > 0$ and $g > 0$. The fifth parameter controls the degree of asymmetry and models the given data much better than the four-parameter model. The lack-of-fit errors of the four-parameter logistic model are eliminated by the five-parameter model.

The values of the vector parameter are obtained by defining the modeling of power curve using five-parameter logistic expression as an optimization problem with the objective function as expressed in (4). The estimated power is calculated by $P_e(u_i) = d + ((a - d)/(1 + ((u_i)/(c))^b)^g)$. The following are the constraints:

$$\left.\begin{array}{l} a_{\min} \le a \le a_{\max}, b_{\min} \le b \le b_{\max}, c_{\min} \le c \le c_{\max} \\ d_{\min} \le d \le d_{\max}, g_{\min} \le g \le g_{\max} \end{array}\right\}. \tag{7}$$

The four and five parameter logistic expressions are solved using GA, EP, PSO, and DE.

B. Nonparametric Models

A nonparametric model is the one in which no assumption is made about the functional form of any distribution. A nonparametric model of the wind turbine power curve is built on the assumption $P = f(u)$ [2]. In this paper, nonparametric models are developed using neural networks, fuzzy c-means clustering algorithm, and data mining algorithms.

III. Power Curve Modeling Techniques

The different modeling techniques used in this paper for the development of parametric and nonparametric models of wind turbine power curve have been discussed in detail in this section.

A. Techniques for Parametric Models

Parametric fitting involves the determination of parameters or coefficients for one or more models on to which the data is fitted.

1) Modeling of Power Curve Using Least Squares Method: A linearized segmented model of the power curve has been developed by dividing it into five linear segments. The available data is fitted on to each linear segment using the method of least squares. The least squares method minimizes the summed square of residuals, in order to obtain the coefficient estimates. The residual of the ith data point r_i is defined as the difference between the actual power output $P_a(i)$ and the fitted response value $P_e(i)$, and is identified as the error associated with the data. The summed square of residuals (S) is given by

$$S = \sum_{i=1}^{N} r_i^2 = \sum_{i=1}^{N} (P_a(i) - P_e(i))^2. \tag{8}$$

2) Modeling of Power Curve Using GA: GA is a search technique based on Darwin's theory of evolution of biological systems [11]. The procedure for logistic function parameter optimization using a real-coded GA algorithm is given in the following steps:

Step 1. Representation of the Population: For a logistic function with N_p parameters, the individual (chromosome) is represented as a vector of length N_p. The complete population with M individuals is represented as a matrix as follows:

$$\mathbf{Pop} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_j, \ldots, \mathbf{X}_M] \tag{9}$$

where $\mathbf{X}_j = [X_{1j}, X_{2j}, \ldots, X_{N_p j}]^T$ represents the jth individual and it is one of the solutions to parameter estimation with X_{ij} as the ith parameter for the jth solution. The population is initialized randomly within the minimum and maximum limits of the parameters of the logistic function.

Step 2. Evaluating the Individuals: Each individual $\mathbf{X}_j$ is evaluated using the objective function as follows:

$$F(\mathbf{X}_j) = F_j = \sum_{i=1}^{N} (P_e(\mathbf{X}_j, u_i) - P_a(u_i))^2. \tag{10}$$

Step 3. Selection: M individuals are selected based on stochastic uniform selection to form the competing pool.

Step 4. Creation of Offspring (Crossover and Mutation): From the competing pool, offspring are created by applying crossover with the crossover probability p_c and mutation with mutation probability p_m as follows.

Crossover: Two parents are selected randomly. Let parent chromosomes X_1 and X_2 are selected to be crossed and parameter p_r be a random number chosen from [0, 1]. If $p_r \le p_c$, then the following crossover on X_1 and X_2 are performed:

$$\left.\begin{array}{l} \text{if } F(\mathbf{X}_1) < F(\mathbf{X}_2) \text{ then } \mathbf{X}_1 \leftarrow \mathbf{X}_1 + r(\mathbf{X}_1 - \mathbf{X}_2) \\ \text{and } \mathbf{X}_2 \leftarrow \mathbf{X}_2 + r(\mathbf{X}_1 - \mathbf{X}_2) \\ \text{else } \mathbf{X}_1 \leftarrow \mathbf{X}_1 + r(\mathbf{X}_2 - \mathbf{X}_1) \text{ and } \mathbf{X}_2 \leftarrow \mathbf{X}_2 + r(\mathbf{X}_2 - \mathbf{X}_1) \end{array}\right\}. \tag{11}$$

Mutation: A parent is selected randomly. Let parent chromosome X_1 be selected to be mutated and parameter p_r be a random number chosen from [0, 1]. If $p_r \le p_m$, then the following mutation on X_1 is performed:

$$X_1' = X_1 + K\Phi \quad (12)$$

where K is a constant and Φ is a real vector to produce a small perturbation.

Step 5. Termination: A maximum number of iterations is chosen as the stopping criterion. If the stopping criterion is not satisfied, the above procedure is repeated from step 2. Otherwise, the best chromosome of the current population is the optimum parameter set.

3) Modeling of Power Curve Using EP: EP is an optimization algorithm based on the mechanics of natural selection—mutation, competition, and evolution [12]. The solution procedure for optimizing the parameters of logistic functions using EP is described below.

Step 1. Representation of the Population: The population is represented as in the case of GA.

Step 2. Evaluating the Individuals: Each individual $\mathbf{X}_j$ is evaluated using the objective function given in (10).

Step 3. Creation of Offspring (Mutation): From each individual (called parent) an offspring is created as follows:

$$X_{ij}' = X_{ij} + N(0, \sigma_{ij}) \quad (13)$$

where $N(0, \sigma_{ij})$ is the normally distributed random number with mean 0 and standard deviation σ_{ij}

$$\sigma_{ij} = \beta_s \times \frac{F_j}{\max\_F} \times (\max\_X_i - \min\_X_i) \quad (14)$$

where β_s is the scaling factor.

Step 4. Selection: The parents and the offspring constitute the competing pool of size $2M$. From the competing pool, M individuals are selected as parents of the next generation based on tournament selection.

Step 5. Termination: If the maximum number of iterations is not reached, the above procedure is repeated from step 2. Otherwise, the best individual of the current population is the optimum parameter set.

4) Modeling of Power Curve Using PSO: PSO is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point (particle) in a multidimensional space. Particles are accelerated towards those particles within their communication grouping, which have better fitness values [13]. The parameter estimation of four and five parameter logistic functions can be optimized by PSO as follows.

Step 1. Representation of the Swarm: For a logistic function with N_p parameters, the particle position is represented as a vector of length N_p. The complete swarm with M particles is represented as a matrix as follows:

$$\mathbf{Swarm} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_j, \ldots, \mathbf{X}_M] \quad (15)$$

where $\mathbf{X}_j = [X_{1j}, X_{2j}, \ldots, X_{N_p j}]^T$ is the position vector of the jth particle and it is one of the solutions to parameter estimation.

Step 2. Initialization of the Swarm: The swarm is initialized randomly within the minimum and maximum limits of the parameters of the logistic function. The velocities of the particles are initialized to zero

$$V_{ij} = 0, i = 1, 2, \ldots, N_p, j = 1, 2, \ldots, M. \quad (16)$$

Step 3. Evaluating the Particle: Each particle $\mathbf{X}_j$ of the swarm is evaluated using the objective function given in (10).

Step 4. Initialization of the Best Positions: In the strategy of PSO, the particle's best position (**Pbest**) and global best position (**Gbest**) are the key factors. The position with minimum F is the particle's best position. Initially the starting positions of particles are taken as their **Pbest**. The best position out of all **Pbest** is **Gbest**.

Step 5. Movement of the Particles: The particles in the swarm are accelerated to new positions by adding new velocities to their present positions. The new velocities are calculated as follows:

$$\begin{aligned} V_{ij} \leftarrow w \times V_{ij} + C_1 \times r_1 \times (\mathrm{Pbest}_{ij} - X_{ij}) \\ + C_2 \times r_2 \times (\mathrm{Gbest}_j - X_{ij}) \end{aligned} \quad (17)$$

where w is the inertia constant; C_1 and C_2 are acceleration coefficients; and r_1 and r_2 are two separately generated random numbers in the range [0, 1]. The positions of the particles are updated using

$$X_{ij} \leftarrow X_{ij} + V_{ij}. \quad (18)$$

Step 6. Updating the Best Positions: The particles are evaluated in the new positions by the objective function. Then **Pbest** of particles are updated as follows:

$$\left.\begin{array}{ll} \mathbf{Pbest}_j = \mathbf{X}_j + \mathbf{V}_j, & \text{if } F_j' < F_j \\ \mathbf{Pbest}_j = \mathbf{Pbest}_j, & \text{if } F_j' \ge F_j \end{array}\right\} \quad (19)$$

where F_j' is the objective function value of jth particle evaluated at the new position. The best position out of all the new **Pbests** is taken as **Gbest**.

Step 7. Termination: If the maximum number of iterations is not reached, the above procedure is repeated from step 5. Otherwise, **Gbest** is the optimum parameter set.

5) Modeling of Power Curve Using DE: DE, proposed by Storn and Price [14], is a simple yet powerful algorithm for real parameter optimization. DE has many variants with the naming convention is DE/x/y/z, where DE stands for "differential evolution," x represents a string denoting the base vector to be mutated, y is the number of difference vectors considered for mutation of x, and z stands for the type of crossover being used (exp: exponential; bin: binomial). To optimize the parameters of the logistic function, a DE/best/1/bin scheme is used in this paper.

Step 1. Representation of the Population: The population is represented as in the case of GA.

Step 2. Evaluating the Individuals: Each individual $\mathbf{X}_j$ of the population is evaluated using the objective function given in (10)

Step 3. Mutation: In DE, a parent vector from the current generation is called target vector, a mutant vector obtained through the differential mutation operation is known as donor vector, and finally an offspring formed by recombining the donor with the target vector is called trial vector. In DE/best/1/bin, to create the donor vector for each jth target vector from the current population, two other distinct parameter vectors, say X_{r1} and X_{r2}, are sampled randomly. The indices $r1$ and $r2$ are mutually exclusive integers randomly chosen from the range $[1, M]$. Now the difference of these two vectors is scaled by a scalar number F (that typically lies in the interval [0.4, 1]) and the scaled difference is added to the best vector $\mathbf{X}_{\text{best}}$ to obtain the donor vector V_j

$$\mathbf{V}_j = \mathbf{X}_{\text{best}} + F(\mathbf{X}_{r1} - \mathbf{X}_{r2}). \tag{20}$$

Step 4. Crossover: Crossover is applied to each pair of the target vector and its donor vector to generate a trial vector. The binomial crossover used in this paper is described as follows:

$$U_{ij} = \begin{cases} V_{ij}, \text{ if } r \leq C_R \text{ or } j = j_{\text{rand}} \\ X_{ij}, \text{ otherwise} \end{cases} \tag{21}$$

where r is a random number in the range [0, 1]; C_R is the crossover rate; and j_{rand} is a random integer in the range $[1, M]$.

Step 5. Selection: The population members for next generation are created as follows.

$$\mathbf{X}_j = \begin{cases} \mathbf{U}_j, \text{ if } F(\mathbf{U}_j) \leq F(\mathbf{X}_j) \\ \mathbf{X}_j, \text{ otherwise.} \end{cases} \tag{22}$$

Step 6. Termination: If the maximum number of iterations is not reached, the above procedure is repeated from step 2. Otherwise, best individual of the current population is the optimum parameter set.

B. Techniques for Nonparametric Models

The techniques used to develop nonparametric models of wind turbine power curve have been discussed below. These include neural networks (NNs), fuzzy c-means clustering (FCM), and data mining algorithms.

1) Modeling of Power Curve Using NN: The power curve has been modeled using a multilayer feed forward back propagation network. According to the back propagation rule, the error to be minimized is

$$E = 0.5 \sum [P_e(i) - P_a(i)]^2. \tag{23}$$

2) Modeling of Power Curve Using FCM: Data clustering is a process of grouping similar data. According to Hammouda [15], a clustering algorithm partitions a data set into several groups such that the similarity within a group is larger than among groups. FCM uses fuzzy partitioning and determines the cluster centers c_i and the membership matrix Q using the following algorithm:

Step 1: Initialize the membership matrix $\mathbf{Q}$ with random values between 0 and 1 with the following constraint:

$$\sum_{i=1}^{c} q_{ij} = 1, \quad \forall j = 1, \ldots, N. \tag{24}$$

Step 2: The fuzzy cluster centers $c_i, i = 1, 2, \ldots, N_c$ are calculated using the following formula:

$$c_i = \sum\nolimits_{j=1}^{N} q_{ij}^m x_j \Big/ \sum\nolimits_{j=1}^{N} q_{ij}^m \tag{25}$$

where

$$q_{ij} = 1 \Big/ \sum\nolimits_{k=1}^{N_c} \left(\frac{d_{ij}}{d_{kj}} \right)^{2/m-1} \tag{26}$$

where q_{ij} is between 0 and 1; c_i is the cluster center of fuzzy group $i, d_{ij} = ||c_i - x_j||$ is the Euclidean distance between the ith cluster center and the jth data point, and $m \in [1, \infty)$ is a weighting component.

Step 3: The cost function of FCM is computed as follows:

$$J(Q, c_1, \ldots, c_c) = \sum_{i=1}^{N_c} J_i = \sum_{i=1}^{N_c} \sum_{j=1}^{N} q_{ij}^m d_{ij}^2. \tag{27}$$

Stop if the cost function is below a certain tolerance value or its improvement over the previous iteration is below a certain threshold.

Step 4: Compute a new Q using (26). Go to step 2.

The variable region of the wind turbine power curve was modeled using five clusters. The cluster coordinates were determined using the FCM algorithm. For a set of c cluster centers, u^* contains the coordinates of the cluster centers of the inputs and p^* contains the coordinates of the cluster centers of the outputs [6]. The output wind power vector $\boldsymbol{P}$ for a particular input wind speed vector $\boldsymbol{U}$ of the test set can be calculated using the following expression:

$$P = \sum\nolimits_{i=1}^{c} \mu_i p_i^* \Big/ \sum\nolimits_{i=1}^{c} \mu_i \tag{28}$$

where

$$\mu_i = e^{\frac{-4\|u - u_i^*\|^2}{r_a^2}} \tag{29}$$

where r_a is a positive constant that represents the radius defining a neighborhood.

3) Modeling of Power Curve Using Data Mining: Data mining refers to extracting or mining knowledge from large amounts of data. Nonparametric modeling of power curve has been done using four data mining algorithms namely bagging, the M5P algorithm, the REPTree algorithm, and M5Rules.

The bagging (or bootstrap aggregation) algorithm works in the following manner [16]:

Step 1: Given a set S of s samples, for iteration $t (t = 1, 2, \ldots, T)$, a training set S_t is sampled with replacement from the original set of samples S.

Step 2: A classifier C_t is learned for each training set S_t.

Step 3: To classify an unknown sample X, each classifier C_t returns its class prediction, which is counted as one vote.

Step 4: The bagged classifier C^* counts the votes and assigns the class with the most votes to X.

Bagging can be applied to the prediction of continuous values by taking the average value of each classifier prediction.

Witten and Frank [17] state that model trees that are used for numeric prediction are just like ordinary decision trees except that at each leaf they store a linear regression model that predicts the class value of the instances that reach the leaf. The initial implementation of the concept of model trees was defined and improved in a system called M5′. M5P is the model tree learner whose algorithm is explained in [18] as follows:

TABLE I
PARAMETERS OF LINEARIZED SEGMENTED MODEL

Seg	Range of Wind speed (m/s)					*m*					*c*				
	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
Seg1	0–2	0 – 4	0– 4	0–4	0 – 4	0	1.565	7.899	3.825	1.006	0	0	0	0	0
Seg2	2.1–4	4.1 – 7	4.1– 7.5	4.1– 8	4.1 – 8	0.996	33.18	82.63	25.61	12.46	-1.763	-129	-316.8	-88.73	-50.13
Seg3	4.1–6	7.1 – 11	7.6 -10	8.1-12.5	8.1-12	3.027	54.53	146.4	39.17	20.77	-10.14	-265	-786.2	-184.1	-116.7
Seg4	6.1– 8	11.1–14	10.1-13	12.6-15.5	12.1-15.5	5.877	17.91	93.04	25	10.9	-27.43	140.8	-254.2	-3.687	-3.131
Seg5	8.1–18	14.1–20	13.1-17	15.6 – 21	15.6-19	0	-2.461	3.661	2.01	1.187	20	423.8	893.5	346.7	142.1

Step 1: M5′ constructs a tree by splitting the data to minimize intrasubset variation in the class values of instances down each branch.

Step 2: A linear model is computed for each node and then the tree is pruned back from the leaves, as long as the expected estimated error decreases.

Step 3: A smoothing procedure is used to compensate for sharp discontinuities that occur between linear models at the leaves of the pruned tree.

REPTree is a fast decision tree learner and it builds a decision or regression tree using information gain/variance reduction and prunes it using reduced-error pruning [16]. M5Rules obtains regression rules from model trees built using M5′. The M5Rules algorithm works as follows [18]:

Step 1: A model tree is applied to the full training dataset and a pruned tree is learned.

Step 2: The best leaf is made into a rule and the tree is discarded.

Step 3: All instances covered by the rule are removed from the dataset. The process is applied recursively to the remaining instances and terminates when all instances are covered by one or more rules.

These data mining algorithms bagging, M5PTree, RepTree, and M5Rules are used to model the wind turbine power curve and their performances are compared using suitable metrics.

IV. RESULTS AND ANALYSIS

A. Experimental Data

The modeling of the wind turbine power curve has been carried out using five sets of data. The first set of data was generated based on normal distribution with a varying mean and constant standard deviation (σ_ε) [7]. The actual wind turbine generator power output $(P_a(u))$, a random variable, can be expressed as

$$P_a(u) = P(u) + N\left(0, \sigma_\varepsilon^2\right). \tag{30}$$

The actual wind turbine generator output of the probabilistic model can be obtained using the cubic power model

$$P_a(u, \sigma_\varepsilon) = \begin{cases} 0, & u < u_c, u > u_s \\ cu^3 + N\left(0, \sigma_\varepsilon^2\right), & u_c \le u \le u_r \\ P_r, & u_r \le u \le u_s \end{cases} \tag{31}$$

where u is the wind speed, c is the power curve coefficient, and $c = 0.5\eta_{\max}\rho\pi R^2, \eta_{\max}$ is a factor that accounts for the conversion rate of total wind power to electrical power, which is usually around 0.5926. The wind turbine power curve is generated using the following data: $P_r = 20$ kW, $u_c = 2$ m/s, $u_r = 8$ m/s, $u_s = 18$ m/s, $c = 0.03906$. The total number of data generated is 1008 with the standard deviation of 0.5 kW.

TABLE II
CONTROL PARAMETERS OF VARIOUS ALGORITHMS

Algorithm	Control Parameters
GA	$p_c = 0.8, p_m = 0.01, K = 0.2$
EP	$\beta_s = 0.5$
PSO	$w = 0.5, C_1 = 2, C_2 = 2$
DE	$F = 1, C_R = 0.5$

The second set of data is a real-time data obtained from Sotavento Galicia Plc., which is an experimental wind farm supported by the "Xunta de Galicia," the regional autonomous government, with an objective to create a new concept in wind farms.[1] A total of 4388 data, the 10-min averaged data of December 2011, has been used. The other three real-time datasets were obtained from the National Renewable Energy Laboratory, which is operated by the Alliance for Sustainable Energy for the U.S. Department of Energy. The hourly data of October–December 2006 of three different wind farms have been used. A total of 2208 data have been used in the datasets 3, 4, and 5.

Nonparametric models were trained using 50% of the datasets and the remaining 50% were used for testing. For implementation of optimization algorithms, the total number of data is considered for all the datasets.

B. Results of Different Algorithms

Parametric and nonparametric models of wind turbine power curve were developed for both the sets of data, using the modeling techniques discussed above and the results were compared using the performance metrics MAE and RMSE.

1) Results of Parametric Models: The parameters of the linearized segmented model of the power curve $\theta = (m, c)$ were obtained as given in Table I for all five datasets. The different segments (Seg) of the power curve were modeled using linear equations in the MATLAB curve fitting tool box.

For applying GA, EP, PSO, and DE algorithms, the population size and the maximum number of iterations are selected as 50 and 100, respectively. The other control parameters of different algorithms used in optimizing the logistic expressions are listed in Table II.

The parameters of the four parameter logistic function $\theta = (a, m, n, \tau)$, obtained using GA, EP, PSO, and DE for all five datasets have been tabulated in Table III. The parameters of the five parameter logistic functions $\theta = (a, b, c, d, g)$ obtained using GA, EP, PSO, and DE for all five datasets have been tabulated in Table IV.

[1]Available: http://www.sotaventogalicia.com/

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE III
PARAMETERS OF 4-PARAMETER LOGISTIC FUNCTION

	Technique	a	m	n	τ
Dataset 1	GA	20	65.805	1386.77	0.8810
	EP	20	54.9554	1269.6142	0.8829
	PSO	20	46.4749	1194.5077	0.8896
	DE	20	43.8621	1163.9926	0.8926
Dataset 2	GA	396.1002	-1.7731	257.4049	1.5835
	EP	391.0775	5.9340	366.6552	1.4412
	PSO	383.4822	-0.3050	271.1179	1.5341
	DE	384.09	-3.4636	216.5773	1.5634
Dataset 3	GA	1177.2019	-0.7386	145.6586	1.8785
	EP	1177.125	-0.5819	145.8057	1.88
	PSO	1130.04	-0.5586	162.6148	1.8048
	DE	973.5215	-1.9582	128.8459	1.7667
Dataset 4	GA	491.1323	-1.2056	58.5472	2.6867
	EP	491.125	-1.2046	58.4675	2.687
	PSO	471.48	-1.1564	62.8034	2.5795
	DE	406.4674	-1.7667	52.2499	2.4473
Dataset 5	GA	206.7508	-0.6230	115.6154	2.2791
	EP	206.75	-0.6206	108.3286	2.28
	PSO	198.48	-0.5958	125.2240	2.1888
	DE	165.4	-1.7910	118.5	1.98

TABLE IV
PARAMETERS OF 5-PARAMETER LOGISTIC FUNCTION

	Technique	a	b	c	d	g
Dataset 1	GA	0.9995	5.2147	17.9621	20.081	155.1248
	EP	0.7300	4.6891	27.9471	20.0351	700.5979
	PSO	0.8662	5.0549	24.6756	20.0153	668.4471
	DE	0.8969	5.0754	25.6579	20	832.3047
Dataset 2	GA	364.1621	-6.2991	8.6091	3.8034	0.7804
	EP	399.4739	-6.5669	10.0970	-1.7044	0.5117
	PSO	395.9986	-6.31994	9.7276	-2.7150	0.5682
	DE	393.9342	-6.4761	9.7280	-3.0050	0.5521
Dataset 3	GA	1125.253	-8.1867	11.0326	10.0449	0.3993
	EP	1123	-8.1138	11.07	14.7125	0.4104
	PSO	1078.08	-7.7892	10.6272	14.124	0.4514
	DE	933.0140	-9.7576	10.7806	-2.9902	0.2906
Dataset 4	GA	477.751	-5.6324	13.6797	-6.1363	0.4321
	EP	477.75	-5.6288	13.68	4.5054	0.4560
	PSO	458.64	-5.4036	13.1328	3.5174	0.4829
	DE	408.6903	-6.4037	12.7074	-0.9182	0.3859
Dataset 5	GA	201.1514	-6.6926	12.7207	4.2393	0.4811
	EP	201.125	-6.6913	12.73	4.2513	0.4817
	PSO	193.08	-6.4236	12.2208	4.0812	0.523
	DE	167.5672	-7.5487	12.1602	-0.1446	0.3654

2) Results of Nonparametric Models: The neural network architecture used for modeling of the wind turbine power curve for all five datasets is shown in Table V.

The FCM algorithm was implemented in MATLAB for all the datasets. The number of clusters used was 8 and $m = 25$. The value of the constant $r_a = 0.2$ for dataset 1, $r_a = 3$ for datasets 2, 3, and 4, and $r_a = 1$ for dataset 5. The error values were calculated after performing 25 iterations. The four different data mining algorithms bagging, M5P, REP Tree, and M5Rules were developed in WEKA. The models were trained using the training data and their performances were tested using the test data of the respective datasets.

TABLE V
NN ARCHITECTURE USED FOR POWER CURVE MODELING

Architecture	Feed-Forward Backpropagation
Training Set	50% data of wind speed and power
Testing Set	50% data of wind speed and power
Hidden neurons	5
Data preprocessing	'mapminmax'
Transfer Function	'tansig' for hidden layers and 'purelin' for output layer
Network Training Function	Levenberg-Marquardt backpropagation
Weight/bias Learning Function	Gradient descent with momentum weight and bias learning function

C. Analysis of Results

The performance of these parametric and nonparametric models has been evaluated using the performance metrics MAE and RMSE which are defined as follows:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |P_e(u_i) - P_a(i)| \tag{32}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (\, P_e(u_i) - P_a(i))^2}. \tag{33}$$

The RMSE is the square root of the mean of the squared difference between the actual and estimated values of power. It is a measure of the average magnitude of error and is a quadratic scoring rule. The MAE is the mean of the absolute values of the differences between the actual and estimated values of power. Unlike RMSE, MAE is a linear score since the individual differences are weighted equally in the average.[2] The value of RMSE is always greater than or equal to MAE. The difference between the RMSE and MAE values highlights the variance in the individual errors in the sample.

The MAE and RMSE values of the discussed algorithms have been tabulated in Table VI for all the datasets. The algorithms have been ranked based on their error measures in Table VII, for each dataset. In order to determine the best modeling technique, the average rank is calculated and the overall performance is ascertained based on both the error measures. It can be observed that the wind turbine power curve model was built using the five-parameter logistic function, whose parameters are solved by the DE algorithm ranks number 1, for both the error measures. When the RMSE measure alone is considered, NN is ranked 2, and the power curve model built using the five-parameter logistic function with the parameters solved using PSO is ranked 3. If the MAE measure is considered, the linearized segmented model is ranked 2 and NN is ranked 3.

The overall performance and the ranking of all the algorithms excluding dataset 1, is tabulated in Table VIII. It is very interesting to note that, when the statistically generated data, dataset 1 is ignored, and the overall performance is calculated only for

[2]Available: http://www.eumetcal.org/resources/ukmeteocal/verification/www/english/msg/ver_cont_var/uos3/uos3_ko1.htm

TABLE VI
COMPARISON OF POWER CURVE MODELING TECHNIQUES

	Model	Algorithm	Dataset 1		Dataset 2		Dataset 3		Dataset 4		Dataset 5	
			RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Parametric Models	Linearized segmented Curve	Least Squares	0.4691	0.3476	28.3212	18.7661	103.0337	69.8420	57.2217	41.0345	19.6978	13.8003
	4-parameter logistic function	GA	0.7398	0.5532	29.9651	20.774	102.0199	70.2737	60.0751	43.9406	21.2214	15.8045
		EP	0.7283	0.5519	28.6444	21.2753	101.9812	70.416	60.0728	43.9422	21.3141	15.9936
		PSO	0.7271	0.5495	28.8801	20.4939	100.9199	69.3364	59.2724	43.0904	20.7257	15.3152
		DE	0.7272	0.5483	27.9386	19.5688	97.5718	66.9535	57.2270	42.7302	19.1279	13.4238
	5-parameter logistic function	GA	0.6406	0.4998	28.9433	20.3428	102.4655	69.7091	58.6015	42.1676	20.7457	15.4607
		EP	0.7059	0.4984	27.8933	19.5262	101.8671	69.8481	58.4756	42.5721	20.7286	15.4384
		PSO	0.6339	0.4886	27.7349	19.3082	100.9022	68.6344	57.7735	41.7241	20.0994	14.7064
		DE	0.6408	0.4874	27.7232	19.3138	97.4912	66.7710	57.2153	40.9138	19.0894	13.4317
Nonparametric Models	Neural networks	BP	0.4874	0.3787	28.1673	19.6073	102.8068	69.5992	57.0671	41.1966	19.6208	13.6773
	Data clustering	FCM	1.1641	0.9153	32.0332	23.4100	106.9193	74.6151	58.2425	40.8216	21.3577	15.2028
	Data mining algorithms	Bagging	0.4755	0.3499	29.2362	19.9826	105.1013	71.4188	58.4727	42.3972	20.3453	14.0917
		M5P	0.632	0.48	28.3008	19.5136	103.6517	71.1526	57.7767	42.4693	19.8768	13.823
		REPTree	0.4787	0.3513	29.2851	20.1967	108.85	74.7756	60.6107	43.8595	21.0684	14.5008
		M5Rules	0.5429	0.4143	28.2686	19.5360	103.7123	71.1429	57.7906	42.5811	19.8689	13.8316

TABLE VII
ANALYSIS OF RESULTS

	Model	Algorithm	Dataset 1 Rank		Dataset 2 Rank		Dataset 3 Rank		Dataset 4 Rank		Dataset 5 Rank		Average Rank		Overall Performance	
			RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Parametric Models	Linearized segmented Curve	Least Squares	1	1	8	1	10	7	3	3	4	4	5.2	3.2	4	2
	4-parameter logistic function	GA	14	14	14	13	7	9	14	14	13	14	12.4	12.8	13	13
		EP	13	13	9	14	6	10	13	15	14	15	11	13.4	11	14
		PSO	11	12	10	12	4	4	12	12	9	11	9.2	10.2	9	11
		DE	12	11	4	7	2	2	4	11	2	1	4.8	6.4	3	5
	5-parameter logistic function	GA	8	10	11	11	8	6	11	6	11	13	9.8	9.2	10	9
		EP	10	9	3	5	5	8	10	9	10	12	7.6	8.6	7	8
		PSO	7	8	2	2	3	3	5	5	7	9	4.8	5.4	3	4
		DE	9	7	1	3	1	1	2	2	1	2	2.8	3	1	1
Nonparametric Models	Neural networks	BP	4	4	5	8	9	5	1	4	3	3	4.4	4.8	2	3
	Data clustering	FCM	15	15	15	15	14	14	8	1	15	10	13.4	11	14	12
	Data mining algorithms	Bagging	2	2	12	9	13	13	9	7	8	7	8.8	7.6	8	7
		M5P	6	6	7	4	11	12	6	8	6	5	7.2	7	6	6
		REPTree	3	3	13	10	15	15	15	13	12	8	11.6	9.8	12	10
		M5Rules	5	5	6	6	12	11	7	10	5	6	7	7.6	5	7

the real-time datasets, it is observed that the parametric models perform better than the nonparametric models. The parametric models obtain the first three ranks, and the NN algorithm is ranked 4 for both error measures. When the RMSE measure is considered, the model built by the five-parametric logistic expression solved using DE is ranked 1, the model developed by the four-parametric logistic expression solved by DE is ranked 2, and the model based on the five-parametric logistic expression solved using PSO is ranked 3.

When the MAE measure is considered, the model built by the five-parametric logistic expression solved using DE is ranked 1, the model developed by the linearized segmented model is ranked 2, and the model based on the five-parametric logistic expression solved using PSO is ranked 3. In general, when the dataset is huge, an accurate model of the wind turbine power curve can be developed using the five-parameter logistic function, whose parameters are determined by the DE algorithm.

V. CONCLUSION

The wind turbine power curve has been modeled using advanced algorithms in this paper. Application of the DE algorithm to a five-parameter logistic function gives the best parametric model of a wind turbine power curve. The neural network algorithm gives the best nonparametric model. These models can be used to monitor the online performance of the wind turbine and also for developing wind power forecasting models. Since the wind turbine power curve is one of the important indicators of its performance, accurately modeled power curves will

TABLE VIII
ANALYSIS OF RESULTS (EXCLUDING DATASET 1)

	Model	Algorithm	Average Rank		Overall Performance	
			RMSE	MAE	RMSE	MAE
Parametric Models	Linearized segmented Curve	Least Squares	6.25	3.75	5	2
	4-parameter logistic function	GA	12	12.5	11	13
		EP	10.5	13.5	10	14
		PSO	8.75	9.75	8	10
		DE	3	5.25	2	5
	5-parameter logistic function	GA	10.25	9	9	9
		EP	7	8.5	6	8
		PSO	4.25	4.75	3	3
		DE	1.25	2	1	1
Nonparametric Models	Neural networks	BP	4.5	5	4	4
	Data clustering	FCM	13	10	12	11
	Data mining algorithms	Bagging	10.5	9	10	9
		M5P	7.5	7.25	7	6
		REPTree	13.75	11.5	13	12
		M5Rules	7.5	8.25	7	7

definitely improve the turbine performance, making the power generated more reliable and contribute tremendously in transforming a wind farm into a wind power plant.

ACKNOWLEDGMENT

The authors acknowledge with gratitude the wind power data (Data) provided by the National Renewable Energy Laboratory (NREL), which is operated by the Alliance for Sustainable Energy (Alliance) for the U.S. Department of Energy (DOE).

REFERENCES

[1] C. Monteiro, R. Bessa, V. Miranda, A. Botterud, J. Wang, and G. Conzelmann, Wind Power Forecasting: State-of-the-Art 2009, Decision and Information Services Division, Argonne National Laboratory, ANL/DIS-10-1, pp. 1–216.

[2] A. Kusiak, H. Zheng, and Z. Song, "On-line monitoring of power curves," *J. Renew. Energy*, vol. 34, no. 6, pp. 1487–1493, 2009.

[3] P. Norgaard and H. Holttinen, "A multi-turbine power curve approach," in *Proc.2004 Nordic Wind Power Conf.*, pp. 1–5.

[4] A. Kusiak, H. Zheng, and Z. Song, "Models for monitoring wind farm power," *J. Renew. Energy*, vol. 34, no. 3, pp. 583–590, 2009.

[5] T. Ustuntas and A. D. Sahin, "Wind turbine power curve estimation based on cluster center fuzzy logic modeling," *J. Wind Eng. Ind. Aero.*, vol. 96, no. 5, pp. 611–620, 2008.

[6] D. Cabezon, I. Marti, M. J. S. Isidro, and I. Perez, "Comparison of methods for power curve modeling," in *Proc. 2004 Global WindPower*, pp. 1–10.

[7] T. Jin and Z. Tian, "Uncertainty analysis for wind energy production with dynamic power curves," in *Proc. 2010 Int. Conf. Probabilistic Methods Applied to Power Systems*, pp. 745–750.

[8] V. Thapar, G. Agnihotri, and V. K. Sethi, "Critical analysis of methods for mathematical modeling of wind turbines," *J. Renew. Energy*, vol. 36, no. 11, pp. 3166–3177, 2011.

[9] M. G. Khalfallah and A. M. Koliub, "Suggestions for improving wind turbine power curves," *J. Desalination*, vol. 209, no. 1–3, pp. 221–229, 2007.

[10] P. G. Gottschalk and J. R. Dunn, "The five parameter logistic: A characterization and comparison with the four-parameter logistic," *J. Analytical Biochem.*, vol. 343, no. 1, pp. 54–65, 2005.

[11] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA, USA: MIT Press, 1992.

[12] D. B. Fogel, "Evolutionary optimization," in *Proc. 1992 Conf. Record of the XXVI ASILOMAR Conf. Signals, Systems and Computer (IEEE Comput. Soc. Press)*, pp. 409–414.

[13] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *1995 Proc. IEEE Int. Conf. Neural Networks IV*, Piscataway, NJ, USA, pp. 1942–1948.

[14] R. Storn and K. V. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimizat.*, vol. 11, no. 4, pp. 341–359, 1997.

[15] K. Hammouda, A comparative study of data clustering techniques Design 625, pp. 1–21 [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download, Citeseer. Retrieved from DOI: 10.1.1.126.3224

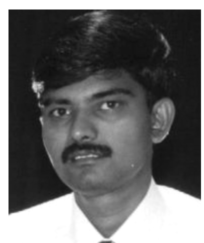
[16] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann, 2006.

[17] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann, 2005.

[18] G. Holmes, M. Hall, and E. Frank, "Generating rule sets from model trees," in *1999 Proc. 12th Australian Joint Conf. Artificial Intelligence*, pp. 1–12.

M. Lydia received the undergraduate degree in electrical and electronics engineering from Bharathiar University, Coimbatore, India, in 2001, and the postgraduate degree specializing in power electronics and drives from Anna University, Chennai, India, in 2004. She is pursuing her research in the area of wind power forecasting at Karunya University, Coimbatore, India.

She is currently Assistant Professor (SG) in the Department of Electrical and Electronics Engineering at Karunya University, Coimbatore, India. Her research interests include renewable energy technologies and power electronics and drives.



A. Immanuel Selvakumar (M'03) received the M.E. degree in power systems engineering from Thiagarajar College of Engineering, Madurai, India, in 2001, and the Ph.D. degree from Anna University, Chennai, India, in 2009.

He is currently working as Associate Professor in the Department of Electrical and Electronics Engineering, Karunya University, Coimbatore, India. His research topics include power system operation and control, renewable energy, soft computing techniques, and metaheuristic optimization techniques.

S. Suresh Kumar (M'07) received the B.E., M.E., and Ph.D. degrees in electrical engineering from Bharathiar University, Coimbatore, India, in 1992, 1997, and 2007, respectively.

He is currently working as Director-Research and Professor and Head in the Department of Electronics and Communication Engineering at Dr. N.G.P. Institute of Technology, Coimbatore, India. He has more than 20 years of teaching and research experience in electrical engineering, particularly in signal and image processing, power quality, renewable energy technology, and power electronics and drives. He is a member of many professional societies including Institution of Engineering and Technology, Association for Computing Machinery, etc.

G. Edwin Prem Kumar received the undergraduate engineering degree in information technology, from the Madurai Kamaraj University and the postgraduate degree specializing in computer science and engineering from Karunya University, Coimbatore, India. He is pursuing his research in the area of wireless sensor networks at Karunya University, Coimbatore, India.

He is currently Assistant Professor in the Department of Information Technology at Karunya University, Coimbatore, India. His research interests include wireless sensor networks and data mining. He is a life time member of Computer Society of India (CSI).