Renewable Energy 34 (2009) 583–590

Contents lists available at ScienceDirect

Renewable Energy

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

Models for monitoring wind farm power

Andrew Kusiak*, Haiyang Zheng, Zhe Song

Department of Mechanical and Industrial Engineering, 3131 Seamans Center, The University of Iowa, Iowa City, IA 52242-1527, USA

ARTICLE INFO

Article history:
Received 2 December 2007
Accepted 20 May 2008
Available online 9 July 2008

Keywords:
Wind farm
Data mining
Power prediction
Monitoring
Evolutionary computation
Control chart

ABSTRACT

Different models for monitoring wind farm power output are considered. Data mining and evolutionary computation are integrated for building the models for prediction and monitoring. Different models using wind speed as input to predict the total power output of a wind farm are compared and analyzed. The *k*-nearest neighbor model, combined with the principal component analysis approach, outperforms other models studied in this research. However, this model performs poorly when the conditions of the wind farm are abnormal. The latter implies that the original data contains many noisy points that need to be filtered. An evolutionary computation algorithm is used to build a nonlinear parametric model to monitor the wind farm performance. This model filters the outliers according to the residual approach and control charts. The *k*-nearest neighbor model produces good performance for the wind farm operating in normal conditions.

1. Introduction

The generation of wind energy on an industrial scale is relatively new. It is then natural that the performance of wind power farms has not been adequately studied. One of the weakest points in wind power generation is the low predictive accuracy of the energy output. Like industrial corporations managed by enterprise-wide systems, a software solution for prediction of wind farm performance (including the amount of energy produced) is needed. The envisioned wind farm performance prediction models should be able to predict the amount of energy produced on different time scales, e.g., 10 min, 1 h, a day, etc. Such models could transform a wind farm into a wind power plant.

Researchers have applied different methodologies in studying wind farms. Cameron and Michael [3] combined the fuzzy set and neural network approaches in an adaptive-neurons-fuzzy inference system to forecast a wind time series. Landberg [6] builts a model to predict the power produced by a wind farm using the data from the weather prediction model (HIRLAM) and the local weather model (WASP). Li et al. [13] compared regression and neural network (NN) models in order to estimate a turbine's power curve. They reported that the NN model outperformed the regression model. Goh et al. [11] proposed a neural network architecture, the complex-valued pipelined recurrent neural network (CPRNN) using a complex value (combined wind speed and direction into one complex value) as input, for predicting the turbine output. Santoso and Le [14] focused on modeling fixed-speed wind turbines. They modeled the component blocks of a turbine (for aerodynamic, mechanical, and electrical components), and aggregated them into models of a single turbine and a wind farm. Lange and Focken [28] presented various models for short-term wind power prediction, including physics-based, fuzzy, and neuro-fuzzy models. Barbounis et al. [29] constructed a local recurrent neural network model for long-term wind speed and power forecasting based on the meteorological data. Hourly forecasts for up to 72 h ahead were produced for a wind park.

Wind energy has become one of the most important sources of energy. Building accurate models for predicting power output and health monitoring of wind farms is needed by this new industry. Developing such models is challenging, as a large number of parameters are involved. The high dimensional and stochastic nature of a wind farm environment calls for new modeling approaches. The developments in data mining (DM) and evolutionary computation (EC) offer promising approaches to model wind farms. Numerous applications of data mining in manufacturing, marketing, medical informatics and the energy industry have proven to be effective in support of decision making [2,5,7,8,24]. Successful applications of evolutionary computation have also been reported in many other domains [1,4,9,10,12,15].

In this paper, a variety of different approaches, including data mining, evolutionary computation, principal component analysis (PCA), residual approach, and control charts, have been used to build prediction models and characterize power curves of a wind farm by a nonlinear parametric model. The models are built using

* Corresponding author.
E-mail address: andrew-kusiak@uiowa.edu (A. Kusiak).

0960-1481/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.renene.2008.05.032

historical data collected by SCADA (Supervisory Control and Data Acquisition) systems at a wind farm.

2. Models for computing power output of a wind farm

2.1. Data description

The data used in this research was generated at a wind farm with about 100 turbines. The data was collected by a SCADA system installed at each wind turbine. Each SCADA system collects data on more than 120 parameters. Though the data is sampled at a high frequency, e.g., 2 s, the data is averaged and stored at 10-min intervals (referred to as 10-min data). The data used in this research was collected over a period of one month at all turbines of the wind farm. The data included one file for each turbine containing over 100 parameters of intervals of an average of 10 min from different sensors and monitoring channels. Examples of parameters included wind speed, wind direction, outside temperature, and turbine control parameters, and were all time stamped. In this research, the wind speed measured at each wind turbine and the corresponding power output were selected for analysis. This was largely dictated by the interests of the wind industry, and more importantly, a complex data-release process controlled by the industry.

2.2. Data pre-processing

The collected data from a wind farm is voluminous, and it usually contains errors caused by sensors and malfunctions of the data collection system. Such errors manifest themselves in missing values, out-of-range values, and so on. For example, the SCADA recorded wind speed should be in the range [0, 20 m/s], and the power should be in the range [0, 1600 kW]. After filtering the raw data, the final data set for 89 turbines was produced, and thus the wind farm considered in this paper included 89 turbines. The speed of 89 turbines and the total power output of 89 turbines as the output recorded at 10-min intervals resulted in 4347 instances (data set 1 in Table 1). Data set 1 was divided into two data sets, data set 2 and data set 3. Data set 2 contains 3476 data points, and it was used to develop a prediction model with data mining algorithms. Data set 3 is comprised of 871 data points, and it was used to test the prediction performance of the model learned from data set 2.

A wind turbine is expected to produce a certain amount of energy for a given wind speed. The relationship between the wind speed and its power output is expressed as a power curve, which has a logistic function shape. In the research reported in this paper, power curves of 89 wind turbines have been analyzed. A typical power curve for a wind turbine is shown in Fig. 1. For a variety of reasons discussed later in the paper, it is clearly seen that the power curve is not an ideal logistic function. In fact, all regions outside of the logistic curve represent power losses. This abnormality of power curves is one of the central factors that motivated the research reported in this paper.

To present a global view of the wind farm, the power curve for the entire wind farm (total power of 89 turbines), included in data set 1 of Table 1, is shown in Fig. 2.

Table 1
The data set description

Data set	Start time stamp	End time stamp	Description
1	1/1/07 12:00 AM	1/31/07 11:50 PM	Total data set; 4347 observations
2	1/1/07 12:00 AM	1/25/07 6:20 PM	Training data set; 3476 observations
3	1/25/07 6:30 PM	1/31/07 11:50 PM	Test data set; 871 observations

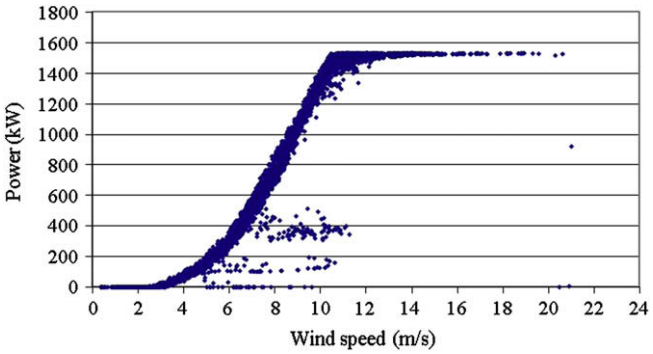

Fig. 1. A typical power curve of a single turbine.

The overall shape of the power curve in Fig. 2 is similar to that of the individual turbine in Fig. 1.

2.3. Extracting models from wind farm data

In this paper, five different data mining algorithms are used to build power prediction models for a wind farm based on data set 2. These algorithms include the multi-layer perceptron algorithm (MLP) [21,19], REP tree [19], M5P tree [20,19], bagging (bootstrapping aggregating) tree [23,22,19], and the *k*-nearest neighbor (*k*-NN) algorithm [19]. MLP is an algorithm with multi-layer perceptron structure. It is usually used in nonlinear regression and classification modeling. REP tree builds a classification or a regression tree using information gain or variance and prunes it using reduced-error pruning with back-fitting. M5P tree is an algorithm for generating trees and rules. Bagging involves aggregation of multiple classifiers or regression trees, and leads to the reduction of misclassification error. The *k*-NN algorithm predicts values based on training examples that are similar to the case considered. It can be used for classification and regression. To test the accuracy of these algorithms, models trained from data set 2 were tested on data set 3. Table 2 shows the prediction accuracy of the models generated by the five algorithms, where Std denotes standard deviation.

Fig. 3 shows the first 200 observed and *k*-NN predicted power values for data set 3. It can be seen from Table 2 and Fig. 3 that for $k = 100$ the *k*-NN algorithm outperforms the other four algorithms. The MLP algorithm produces the lowest accuracy prediction, and the bagging tree and the M5P tree algorithms perform quite well.

In this research, *k*-NN is used to predict wind farm power based on the wind speed. The basic steps of the *k*-NN algorithm are as follows [19]:

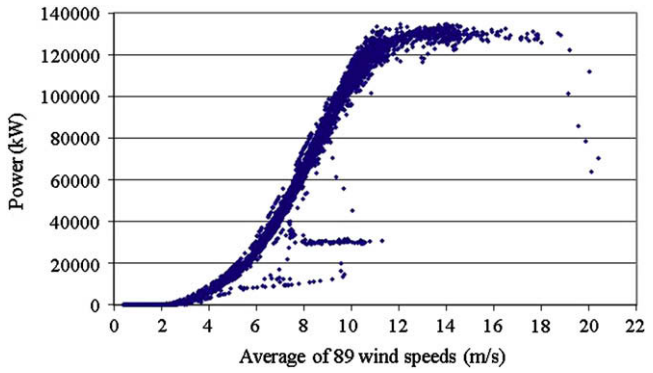

Fig. 2. Cumulative power curve of the wind farm (data set 1 of Table 1).

Table 2
Prediction accuracy of models generated by five different algorithms

Algorithm	Mean absolute error (kW)	Absolute error Std (kW)	Mean relative error (%)	Relative error Std (%)
MLP	4748.0384	6226.7351	49.0306	223.9045
M5P tree	3518.2017	4711.1737	18.7217	47.23016
REP tree	4888.7874	5575.3074	19.9904	32.89106
k-NN ($k=100$)	2872.2923	2949.7033	10.5013	30.74105
Bagging tree	3199.8823	3303.4856	16.1681	37.53675

1. Represent each instance in a multi-dimensional space.
2. Divide the entire data set into training and test data sets.
3. Given a test instance, a distance metric is computed between the test instance and all training instance, then the k-nearest neighbors are selected from the training data.
4. Compute the average distance of the k-nearest neighbors. This distance becomes the predicted value for the test instance.

Different distance metrics are used, including Euclidean, Manhattan, and so on. The parameter k is significant in k-NN algorithm and its best value depends on the data structure and conditions. In this research, the Euclidean distance metric is selected and k is set to 100 based on the model's prediction accuracy.

2.4. Principal component analysis

To obtain insights into the data, the correlation among all 89 inputs (wind speeds from 89 individual turbines) has been computed. The results show that the wind speeds are not highly correlated. To reduce the input dimensionality, the principal component analysis (PCA) [16] was chosen to transform the 10-min wind speed data measured at 89 turbines into a low dimension input for predictive modeling. The k-nearest neighbor (k-NN) algorithm provided good quality results in a similar project and therefore it has been selected for further investigation.

The PCA expresses the variance–covariance structure of a set of variables by a few linear combinations. The basic steps of the PCA are as follows [16]:

1. Compute a correlation matrix.
2. Compute the eigenvectors and eigenvalues of the correlation matrix.
3. Select the components to form an eigenvector.
4. Derive the new data comprised of the principal component of the original data.

Table 3 presents the first 10 eigenvalues of the correlation matrix and the related statistics. Based on the eigenvalue statistics, the first principal component explains 95.7% of the total variance, and

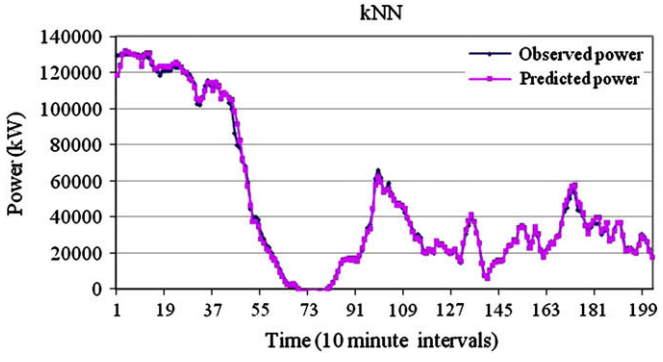

Fig. 3. Predicted and observed power for the first 200 data points of data set 3.

Table 3
Eigenvalues of the correlation matrix and the related statistics

Value number	Eigenvalue	% Total variance	Cumulative Eigenvalue	Cumulative %
1	85.1795	95.7071	85.1795	95.7073
2	0.5450	0.6124	85.7245	96.3197
3	0.3690	0.4146	86.0935	96.7344
4	0.2052	0.2306	86.2988	96.9650
5	0.1844	0.2072	86.4833	97.1723
6	0.1523	0.1711	86.6356	97.3434
7	0.1381	0.1552	86.7738	97.4987
8	0.1319	0.1482	86.9058	97.6469
9	0.1132	0.1272	87.0190	97.7742
10	0.0995	0.1118	87.1185	97.8860

therefore a subset (in particular, one) of eigenvalues is selected. Thus the dimensionality of the data stream (89 inputs) is reduced. The principal components, which are uncorrelated linear combinations of the 89 original wind speeds, should form the new coordinate and input for the k-NN model discussed in Section 2.3.

The wind speed of data set 2 and data set 3 in Table 1 are both transformed into lower dimensional data sets. The models built from the transformed data set 2 are tested using the transformed data set 3. The data in Table 4 illustrates the prediction accuracy of the k-NN model for different numbers of principal components, $P=1$, $P=2$, $P=5$, and $P=10$. The k-NN model for different numbers of components P is denoted as k-NN-P1 through k-NN-P10. The parameter k in each k-NN model has been optimized for prediction accuracy, and the values of k providing the best accuracy is shown in Table 4.

The data in Table 4 shows that the best performance of the k-NN model has been produced for $k=250$ and the number of principal components $P=1$ (this model is labeled as k-NN-P1), and the prediction accuracy worsens as the number of P increases. Compared with the k-NN ($k=100$) model in Section 2.3, the dimension of the input of k-NN-P1 ($k=250$) has been reduced from 89 to 1, and the prediction accuracy has been significantly enhanced:

- The mean relative error (%) was reduced from 10.5013 to 8.69855.
- The corresponding Std (%) improved from 30.74105 to 21.3092.
- The mean absolute error (kW) was enhanced from 2872.2923 to 2255.2954.
- The corresponding Std (kW) was reduced from 2949.7033 to 2174.7299.

The power predicted by the k-NN-P1 ($k=250$) model and the observed power of data set 3 in Table 1 are compared in Fig. 4. The predicted power curve appears to closely follow the measured power.

The relative error (%) produced by the k-NN-P1 ($k=250$) model is shown in Fig. 3.

The error chart of Fig. 5 indicates numerous points with low prediction accuracy. The absolute error of some predictions is larger than 350%, and for some other points the error is between 30 and 100%. However, among the 871 prediction points (the entire data

Table 4
Prediction accuracy of k-NN model with different numbers of principal components P

Algorithm	Mean absolute error (kW)	Absolute error Std (kW)	Mean relative error (%)	Relative error Std (%)
k-NN-P1 ($k=250$)	2255.2954	2174.7299	8.69855	21.3092
k-NN-P2 ($k=40$)	2814.9793	2854.9392	12.5694	61.2684
k-NN-P5 ($k=20$)	3545.3267	3612.7334	15.6346	62.6743
k-NN-P10 ($k=5$)	4612.3748	4790.2389	20.8753	75.1256

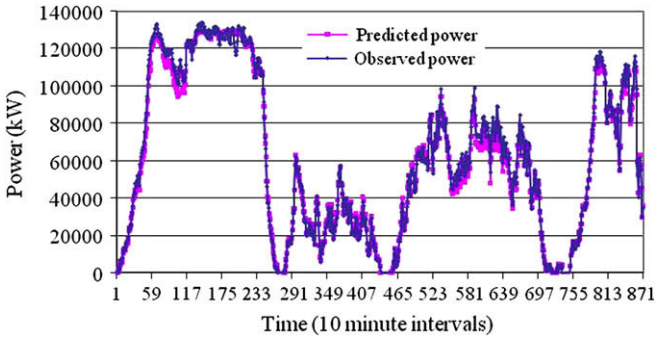

Fig. 4. Power predicted by the k-NN-$P1$ ($k = 250$) model and the observed power.

set 3 in Table 1), only 70 predictions produced an error larger than 15%. These points represents only 8% of the data set 3 in Table 1, and hence these points with a large prediction relative error are considered here as outliers. Analysis of the outlier data should lead to solutions improving the accuracy of the k-NN-$P1$ model.

3. Analysis of outlier data

In this section the 70 points with large relative error (larger than 15%) predicted by the k-NN-$P1$ ($k = 250$) model are analyzed.

According to the manual of the wind turbine on the farm, the cut-off wind speed of a single turbine is 3.5 m/s. The data in Table 5 indicates that the average wind speed for all outliers is around 2.4 m/s (below the turbine's cut-off wind speed), and the corresponding average power of 70 outlier data produced is approximately 1010 kW. Therefore, generally, if the average wind speed is below the cut-off point, the turbines are working in abnormal conditions, which then produce the outliers for the k-NN-$P1$ model. Low wind speed is not the only reason for outliers. In fact, some outliers in Table 5 correspond to the wind speed higher than 4 m/s as indicated by its standard deviation and the maximum.

Fig. 6 illustrates the observed power curve as the function of the principal component derived from the 89 wind speeds from data set 1 (data set 2 plus data set 3) of Table 1. It is obvious from the chart that the outliers do exist as Figs. 1 and 2, making the power curve irregular and affecting the accuracy of the k-NN-$P1$ model.

The operating experience of wind farms and the statistical results discussed point to three main sources of outliers:

1. Wind speed. The low output power is due to the wind speed around the cut-in point (the cut-in speed is set at 2.4 m/s) or the wind speed around the cut-out point (the cut-out speed is set at 20 m/s). A turbine with wind speed below the cut-in point operates abnormally because insufficient wind energy cannot power the turbine. On the other hand, wind speed above the cut-out point causes the turbine to vibrate. To avoid the negative impact of high wind speed on the turbine's life-cycle, the control system shuts down its operation.
2. Environmental issues other than the wind speed may produce power curve outliers. Blades affected by dirt, bugs, and ice may impact power curves of individual turbines and produce outliers reflected in the wind farm power curve.
3. Wind farm shut-down due to maintenance or energy curtailment. Scheduled or unscheduled maintenance operations as well as energy curtailment due to diminished transmission capacity may lead to disrupted operations of individual turbines or the entire wind farm.
4. Control system issues. The conditions of the wind could be in the normal range, yet the power produced could be below the values indicated by the power curve. A possible reason is that the control parameters may be not appropriate for the wind regime. Such power anomalies are clearly visible, for example, the points not following the logistic function shape of the power curves in Figs. 1 and 2. The specific reason may be attributed to the malfunction of the sensors, pitch control malfunctions, blade pitch angle errors, blade damage, control program problems, incorrect controller settings, constrained operations, and so on.

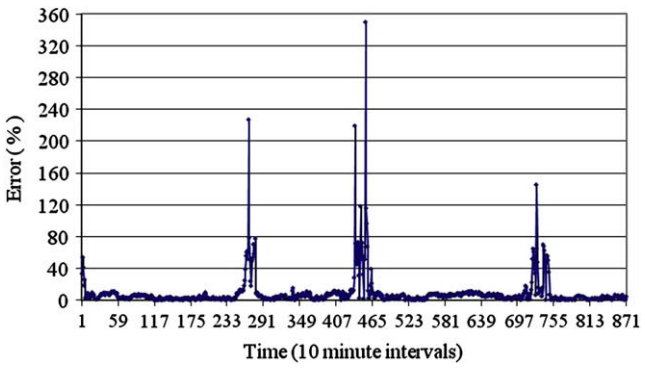

Fig. 5. Relative error (%) of the k-NN-$P1$ ($k = 250$) model.

Table 5
Statistics for all outliers produced by the k-NN-$P1$ ($k = 250$) model

All outliers (70)	Mean	Standard deviation	Minimum	Maximum
Average wind speed (m/s)	2.3868	0.8439	0.4266	4.5873
Predicted power (kW)	1097.4237	2340.9837	−316.0187	11085.0537
Observed power (kW)	830.6759	1969.0140	−604.6623	9399.7344
Relative error (%)	60.1507	51.8928	15.0991	350.1841

4. Nonlinear parametric modeling of wind farm power curves

The quality of the power generated by a wind farm is characterized by its power curve (see Fig. 2). Thus far the existing wind energy literature and practices assume that the power curve is static. This research shows that the power curve is not static, and it should be constructed as parametric. A parametric power curve adjusts to the current operational conditions by modifying its parameters, resulting in an enhanced performance of a wind turbine. One way of using the parametric power curve is to monitor the power of a wind turbine (or the entire wind farm) and detect outlier data. To construct a parametric power curve, the outliers from data set 2 and data set 3 of Table 1 will be detected and filtered out by the approach discussed in Section 5.1.

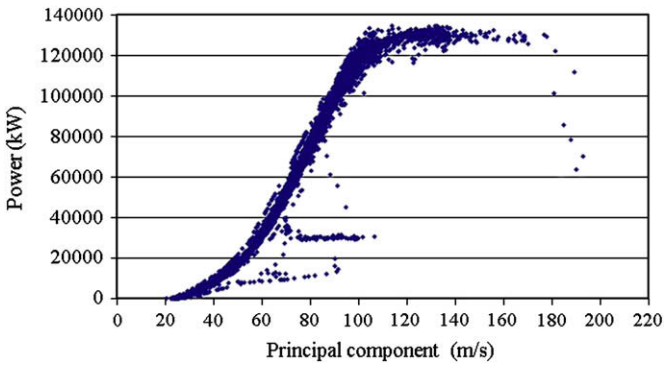

Fig. 6. Observed power curve as the function of the principal component derived from 89 wind speeds.

4.1. Learning parametric model from training data

Based on the analysis of the historical data (data set 1), the wind farm power curve is approximated by logistic function [25] (1)

$$y = f(x, \boldsymbol{\theta}) = a\frac{1 + me^{-x/\tau}}{1 + ne^{-x/\tau}}, \quad \boldsymbol{\theta} = (a, m, n, \tau) \tag{1}$$

where x is the principal component of 89 wind speeds (the predictor in k-NN-P1 model), y is the power of the wind farm, and $\boldsymbol{\theta} = (a, m, n, \tau)$ is a 4-dimension vector parameter of logistic function that determines the shape of the power curve.

Assume there exists a training data set composed of N pairs of data points $[y(i), x(i)]$, $i = 1,\ldots,N$ characterizing the power curve when the wind farm operates under normal conditions. LSM (least squares method) [26] is commonly used to model numeric observed data by adjusting the parameters of the model in order to best fit the data. The residual in regression analysis is defined as the difference between the predicted value of the model and the observed value. The best fit model is characterized by the sum of the squared residual over the training data set that has the least value.

To best estimate $\boldsymbol{\theta}$ of the parametric model (1) from the training data set, LSM is used in this research to best fit the power curve, and thus the sum of squared residuals of LSM over the training data set is used as the cost function (2) to be minimized:

$$S_{(x,y)} = \sum_{i=1}^{N}\left[a\frac{1 + me^{-x(i)/\tau}}{1 + ne^{-x(i)/\tau}} - y(i)\right]^2 \tag{2}$$

Therefore, the estimate of vector parameter $\widehat{\boldsymbol{\theta}}$ can be calculated from Eq. (3)

$$\widehat{\boldsymbol{\theta}} = \underset{a,m,n,\tau}{\operatorname{argmin}} S_{(x,y)}(x(1), y(1); \ldots; x(N), y(N)|a, m, n, \tau) \tag{3}$$

Solving Eq. (3) and building the nonlinear parametric model of the power curve for a wind farm (or a wind turbine) poses several challenges:

1. As the original data in Table 1 contains outliers data (see Fig. 6), the original observed data does not represent the normal power curve of the wind farm. Under such circumstances, using the least squares method (LSM) can lead to a biased estimation of the vector parameter $\boldsymbol{\theta}$. Finding suitable training data which characterizes the normal power curve is then necessary.
2. The function (1) is nonlinear and the vector parameter $\boldsymbol{\theta} = (a, m, n, \tau)$ contains four variables. An algorithm is needed to search for the best estimate of the vector parameter in a 4-dimensional space.

To obtain a training data representing a desirable and normal power curve, the k-NN-P1 model ($k = 250$) is applied to the data set 2 of Table 1. Fig. 7 shows the power observed and predicted by the k-NN-P1 ($k = 250$) model of data set 2 (Table 1). The predicted power curve in Fig. 7 is far less scattered than the observed power curve, and it contains no abnormal points. The points represented by the k-NN-P1 ($k = 250$) predicted power curve are suitable as a training data set for building a nonlinear parametric model.

The new training data for the parametric model includes the principal component derived from 89 wind speeds used as $x(i)$ in function (2) and the power predicted by the k-NN-P1 ($k = 250$) model used as $y(i)$ in function (2). The new training data includes 3476 pairs of data points $[y_{k\text{-NN}}(i), x(i)]$ (obtained from data set 2 of Table 1), and therefore the problem of finding a training data characterizing the normal power curve of a wind farm has been solved.

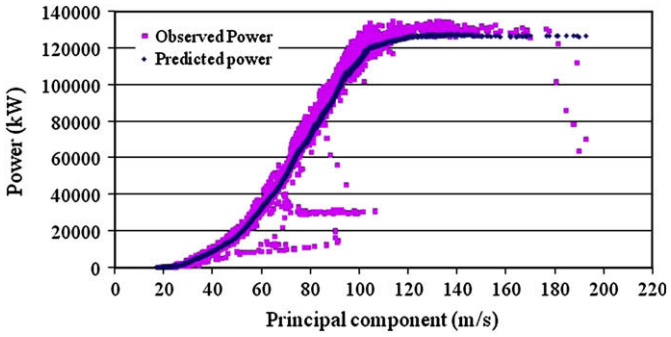

Fig. 7. Power curve of the observed power and the power predicted by the k-NN-P1 ($k = 250$) model.

The basic procedure for building a parametric model for wind farm power is as follows:

1. Transform the original observed wind speed of 89 turbines into the principal component wind speed (PCWS) using the PCA approach.
2. Use the k-NN model to predict the desirable (normal) power based on the PCWS values.
3. The training data set characterizing the normal performance of the wind farm includes PCWS values and power predicted by the k-NN model with these PCWS values. Note that the training data set contains a small number of outliers.
4. Learn the parametric model from the training data $[y_{k\text{-NN}}(i), x(i)]$ by the evolutionary strategy (ES) algorithm (see Section 4.2).

4.2. Learning parametric model by evolutionary strategy (ES) algorithm

To minimize the cost function (2) and estimate the value of parameter $\boldsymbol{\theta}$, an evolutionary strategy (ES) approach is used. There are two basic reasons for using the ES algorithm [1], scalability and computational efficiency. ES algorithms are suitable for solving large-scale problems and at the same time are efficient.

The basic steps of the evolutionary strategy algorithm are [1]:

1. Initialize μ individuals (candidate vector parameter) to form the initial parent population.
2. Repeat until the stopping criteria are satisfied.
   2.1. Select from the parent population and recombine two parents λ times to generate λ children.
   2.2. Mutate λ children.
   2.3. Select the best μ individuals from the children and parent pool based on the fitness function values.
   2.4. Use these selected μ individuals as parents for the next generation.
3. Apply one of the stopping criteria, e.g., the allowable number of generations.

In this research, the cost function (2) serves as the fitness function of the ES algorithm. The ES algorithm calls for initial values of the vector parameter $\boldsymbol{\theta}$. Several heuristic experiments have been performed to generate the initial parametric model of the power curve. The result of one of these experiments is shown in Fig. 8. It is clearly seen that the heuristically-generated parametric power curve and the one generated from the new training data $[y_{k\text{-NN}}(i), x(i)]$ differ; however, their shapes are similar. The ES algorithm determines parameters that make the parametric power curve fit

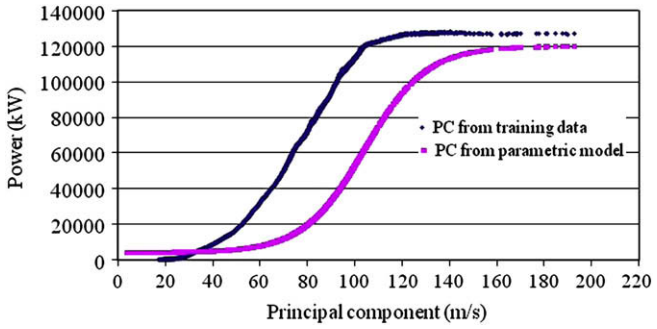

Fig. 8. Power curve (PC) obtained from the parametric model with heuristic parameters and the training data set.

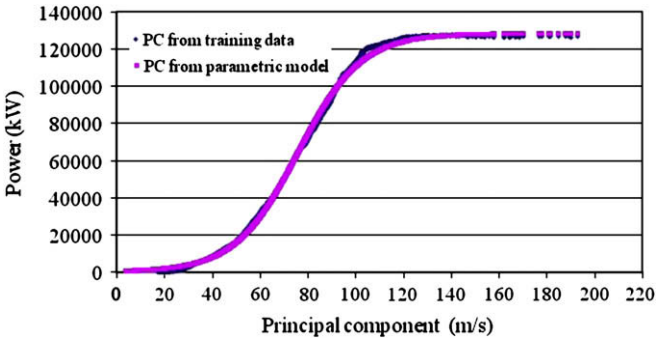

Fig. 10. Power curve (PC) generated from the parametric model and the training data set.

the training data. The cost function (2) based on the LSM is used as the fitness function of the ES algorithm. In summary, the nonlinear parametric model is built based on the LSM concept and solved by the ES algorithm.

Fig. 9 shows that the fitness function converges to the minimum value after 400 generations. To clearly illustrate the convergence, Fig. 9 begins with the fifth generation of the ES algorithm.

The $\hat{\boldsymbol{\theta}}$ computed by the ES algorithm is $\hat{\boldsymbol{\theta}} = (128545.6123, 0.7106, 320.8248, 13.1239)$. Fig. 10 shows that the parametric power curve and the power curve based on training data generated from the k-NN-$P1$ ($k = 250$) model are essentially identical.

The parametric model fits well the training data $[y_{k\text{-NN}}(i), x(i)]$. The power in the training data is predicted by the k-NN model, and it is different from the observed power of data set 2 in Table 1. Fig. 11 illustrates that the parametric power curve learned from the training data $[y_{k\text{-NN}}(i), x(i)]$ fits well the observed power curve of the wind farm. It contains no abnormal or outlier data shown in Figs. 2 and 6. The power curve of the wind farm in normal conditions can be characterized by the parametric model, and thus it can be used as a reference power curve to monitor the power generating process of a wind farm (for details see Section 5.1).

5. Filtering outliers by residual approach and control charts

A formal approach to detect outliers in data is needed. The high quality data (without noise and outliers) can be used to build models of high prediction accuracy when the wind farm operates under normal conditions.

5.1. Forming control charts to filter outliers

The power curve of the parametric model built in Section 4.2 characterizes the wind farm in normal conditions, and therefore it can be used as a true (dynamic) reference power curve. The residual (control theory) and control chart (quality control) approaches are used to analyze residuals of the parametric model and the observed power. The residual ε is expressed as [17,27]:

$$\varepsilon = \hat{y} - y, \tag{4}$$

where $\hat{y} = f(x, \boldsymbol{\theta}) = a(1 + me^{-x/\tau}/1 + ne^{-x/\tau})$,

$$\hat{\boldsymbol{\theta}} = (128545.6123, 0.7106, 320.8248, 13.1239)$$

where y is the observed power, $\hat{\boldsymbol{\theta}}$ is the estimate of the parameters of the model computed by the ES algorithm (Section 4.2), $f(x, \boldsymbol{\theta})$ is the parametric model built in Section 4.2 and $\hat{y}$ is the reference power of the parametric model, ε is the residual of the parametric and the observed power, x is the principal component of 89 wind speeds used by the k-NN-$P1$ ($k = 250$) model (Section 2.4).

The control chart approach [17,18,27] allows the residuals and their variations to be monitored, thus detecting the outlier and abnormal data indicating the abnormal conditions of the wind farm. A data set comprised of 2000 observations was considered. This training data set (2000 observations) without outliers was selected from the data set 2 in Table 1 to form a control chart. The training data set can be represented as, $y\_\text{TrainSet} = [y(i), \hat{y}(i)]$, $i = 1,\ldots,N$, where $N = 2000$.

Using the training data set ($N = 2000$), the residual ε for each point is computed, as well as the mean and standard deviation of ε. The mean residual μ_{Train} is $1/N\sum_{i=1}^{N}(\hat{y}(i) - y(i))$, and the standard deviation σ_{Train} is $\sqrt{1/(N-1)\sum_{i=1}^{N}((\hat{y}(i) - y(i)) - \mu_{\text{Train}})^2}$.

For the testing data set, expressed as $y\_\text{TestSet} = [y(i), \hat{y}(i)]$, $i = 1,\ldots,n$. The testing data set includes n consecutive data points drawn according to the time sequence from the entire data set (4347 instances in data set 1 from Table 1).

Similarly, the mean residual μ_{Test} and the standard deviation σ_{Test} of the test data set are expressed as:

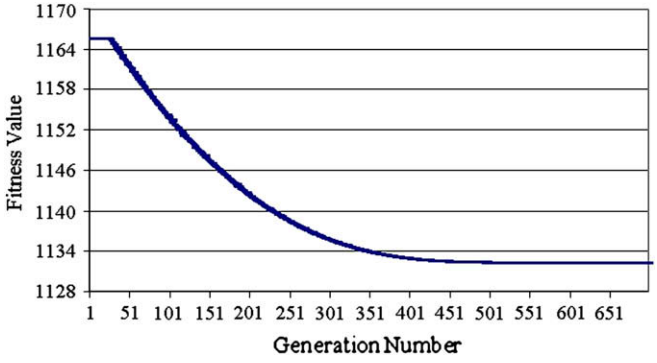

Fig. 9. Convergence process of the fitness function of Eq. (2).

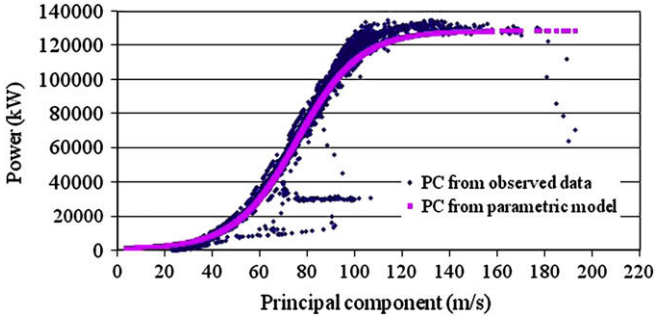

Fig. 11. Power curve generated from the observed and the parametric model.

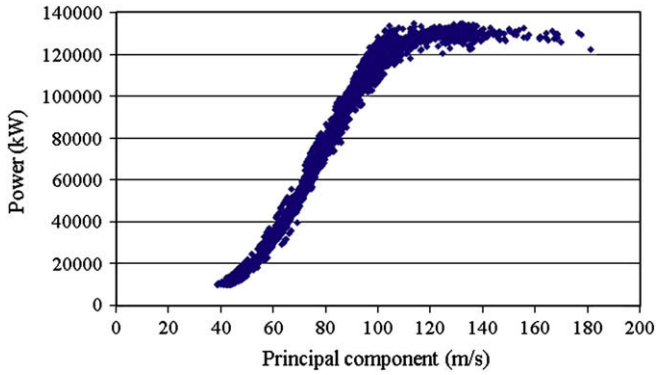

Fig. 12. Power curve after filtering out the outlier and abnormal data.

$$\mu_{\text{Test}} = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}(i) - y(i)\right), \quad \sigma_{\text{Test}} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(\left(\hat{y}(i) - y(i)\right) - \mu_{\text{Test}}\right)^2}.$$

Once μ_{Train} and σ_{Train} are known, the upper and lower control limits are computed and used to detect the abnormal or outlier data. Based on model (5) [17], control limits are derived as:

$$\begin{aligned} UCL_1 &= \mu_{\text{Train}} + 4\frac{\sigma_{\text{Train}}}{\sqrt{n}} \\ \text{Center Line}_1 &= \mu_{\text{Train}} \\ LCL_1 &= \mu_{\text{Train}} - 4\frac{\sigma_{\text{Train}}}{\sqrt{n}} \end{aligned} \tag{5}$$

where n is the number of points in y_TestSet, and the constant 4, other than the widely used constant 3, in model (5) is used to make the control chart less sensitive to the data variability. Such a chart reduces the risk of mistaking normal points for abnormal ones and filtering out many normal points in error. In this paper, n was chosen to be 2 to make the control chart less sensitive to the data variability. If μ_{Test} is above UCL_1 or below LCL_1, the data points in y_TestSet are considered as abnormal or outlier points.

Similarly, the control limits for σ^2_{Test} are calculated from Eq. (6) [17]:

$$\begin{aligned} UCL_2 &= \frac{\sigma^2_{\text{Train}}}{n-1} \times \chi^2_{\alpha/2,n-1} \\ \text{Center Line}_2 &= \sigma^2_{\text{Train}} \\ LCL_2 &= 0 \end{aligned} \tag{6}$$

where n is the number of points in y_TestSet, $\chi^2_{\alpha/2,n-1}$ denotes the right $\alpha/2$ percentage points of the chi-square distribution, $n-1$ is the degree of freedom of the chi-square distribution. If σ^2_{Test} is above UCL_2, the data in y_TestSet is considered as abnormal or outlier points. LCL_2 is set to 0, which indicates that the measurement of the sensor is correct, and the wind farm status is normal, and therefore there is little chance that the test points are abnormal or outliers.

The computed values of the upper and lower control limits are $UCL_1 = 8274.58$, $LCL_1 = -11554.38$, $UCL_2 = 81525027.82$, $LCL_2 = 0$. Using these values, the outliers and abnormal data in the entire set (data set 2 plus data set 3 in Table 1) could be detected by the control charts. Both the training and test data sets with abnormal or outlier data filtered out describe normal operations of the wind farm.

Table 6
The description of reduced data sets

Reduced data set	Start time stamp	End time stamp	Description
1	1/1/07 12:00 AM	1/31/07 11:50 PM	Total data set; 3460 observations
2	1/1/07 12:00 AM	1/25/07 6:20 PM	Training data set; 2589 observations
3	1/25/07 6:30 PM	1/31/07 11:50 PM	Test data set; 871 observations

5.2. *The* k-*NN*-P *model based on the filtered data*

Control charts can be used to remove outlier data. However, in practice, if the wind farm is producing less than a certain percentage of its capacity (here 10,000 kW), the corresponding SCADA data points are considered as outliers. Basically, there are two main types of the outlier data, the one detected by the control charts and the data associated with the low power production (here less than 10,000 kW per wind farm). The power curve for the filtered data with the approach discussed here is shown in Fig. 12.

A filtered data set was created from the data set 1 of Table 1, with the filtered out points including:

- 156 outlier points detected by the control chart
- 731 points with observed power less than 10,000 kW

The outlier data points detected by the control chart are usually due to the control, environmental, maintenance, or power curtailment issues. Other outliers are predominantly due to the low wind speed. The reduced data set is divided into test and training data sets. Table 6 shows the characteristics of the reduced data set. For comparison with the k-NN-P1 ($k = 250$) model in Section 2.4, the test data contains the same number of data points; however, the training data set has been reduced from 3476 to 2589 points.

Table 7 compares the relative error (%) of prediction of two different models, the k-NN-P1 ($k = 250$) model (the model built in Section 2.3), and k-NN-P1-F ($k = 70$) model, which is the model built on the filtered data (F stands for filtered). It is clearly seen in Table 7 that model k-NN-P1-F ($k = 70$) has reduced the mean standard deviation and the maximum relative error. The model k-NN-P1-R ($k = 70$) is thus more robust, accurate, and stable. It makes accurate predictions when the wind farm is working in normal conditions. In addition, for smaller values k the computational effort is reduced.

6. Conclusion

Models for computing power produced by a wind farm under normal operating conditions were developed. In particular, a nonlinear parametric model of a power curve was constructed. To develop these models algorithms from four different domains were used, namely: data mining, evolutionary computation, principal component analysis, and statistical process control. The focus of the paper was on studying a wind farm operating in normal conditions. The normal conditions exclude states where the wind speed is too low or high, turbines undergo maintenance or are affected by environmental issues, and low power output due to control issues. The data sets available for the study were not sufficient for in-depth investigation of abnormal states. In the studied period, the wind farm was operating predominantly in normal conditions. A nonlinear parametric model of the power curve of the wind farm was constructed with an evolutionary strategy algorithm. One application of the parametric model was to monitor online performance of a wind farm. This parametric power curve served as a reference to monitor the generated power. Another application was to improve the quality of the data by filtering abnormal data. The filtered

Table 7
The comparison of the *k*-NN-*P*1 model before and after data filtering

Relative error (%)	*k*-NN-*P*1 ($k=250$)	*k*-NN-*P*1-F ($k=70$)
Mean	8.6985	3.5763
Standard deviation	21.3092	2.5464
Maximum	350.1842	14.6894
Minimum	0.0001	0.0055

data enhanced accuracy, stability, and robustness of the prediction model. In addition, the computational time was reduced.

The ultimate goal of the research initiated in this paper was to derive accurate predictive models for a wind farm working in normal conditions. Once additional data, e.g., wind direction, air density, temperature and so on, becomes available such models are likely to be developed. Although the *k*-NN-*P*1 model was used in this paper, other data mining algorithms could enhance prediction accuracy. The parametric power curve derived in this paper should have a great impact as a performance monitoring tool. This parametric model can become a basis of improvements in the control, monitoring, and optimization of wind farm performance.

Acknowledgement

The research reported in the paper has been partially supported by funding from the Iowa Energy Center Grant No. 07-01.

References

[1] Eiben AE, Smith JE. Introduction to evolutionary computation. New York: Springer; 2003. p. 299.
[2] Kusiak A, Song Z. Combustion efficiency optimization and virtual testing: a data-mining approach. IEEE Transactions on Industrial Informatics 2006; 2(No. 3):176–84.
[3] Cameron WP, Michael N. Very short-term wind forecasting for Tasmanian power generation. IEEE Transactions on Power Systems 2006;21(No. 2):1–8.
[4] Benini E, Toffolo A. Optimal design of horizontal-axis wind turbines using blade-element theory and evolutionary computation. Journal of Solar Energy Engineering 2002;124(No. 4):357–63.
[5] Harding JA, Shahbaz M, Srinivas S, Kusiak A. Data mining in manufacturing: a review. ASME Transactions: Journal of Manufacturing Science and Engineering 2006;128(No. 4):969–76.
[6] Landberg L. Short-term prediction of the power production from wind farms. Journal of Wind Engineering and Industrial Aerodynamics 1998;80(No. 1–2): 207–20.
[7] Berry MJA, Linoff GS. Data mining techniques: for marketing, sales, and customer relationship management. 2nd ed. New York: Wiley; 2004.
[8] Tan PN, Steinbach M, Kumar V. Introduction to data mining. Boston, MA: Pearson Education/Addison Wesley; 2006.
[9] Grady SA, Hussaini MY, Abdullah MM. Placement of wind turbines using genetic algorithm. Renewable Energy 2004;30(No. 2):259–70.
[10] Thilagar SH, Rao GS. Parameter estimation of three-winding transformers using genetic algorithm. Engineering Applications of Artificial Intelligence 2002;15(No. 5):429–37.
[11] Goh SL, Popovic DH, Mandic DP. Complex-valued estimation of wind profile and wind power. In: Proceedings of 12th IEEE Mediterranean Electrotechnical Conference; 2004. p. 1037–40.
[12] Obayashi S, Tsukahara T, Nakamura T. Multiobjective genetic algorithm applied to aerodynamic design of cascade airfoils. IEEE Transactions on Industrial Electronics 2000;47(No. 1):211–6.
[13] Li S, Wunsch DC, O'Hair E, Giesselmann MG. Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation. Journal of Solar Energy Engineering 2001;123(No. 4):327–32.
[14] Santoso S, Le HT. Fundamental time–domain wind turbine models for wind power studies. Renewable Energy 2007;32(No. 14):2436–52.
[15] Cai Z, Wang Y. A multi-objective optimization-based evolutionary algorithm for constrained optimization. IEEE Transactions on Evolutionary Computation 2006;10(No. 6):658–75.
[16] Johnson RA, Wichern DW. Applied multivariate statistical analysis. 4th ed. New Jersey: Prentice Hall; 2005. p. 799.
[17] Mitra A. Fundamentals of quality control and improvement. 2nd ed. Upper Saddle River, New Jersey: Prentice Hall; 1998. p. 752.
[18] Montgomery DC. Introduction to statistical quality control. 5th ed. New York: John Wiley & Sons; 2005. p. 776.
[19] Witten IH, Frank E. Data mining: practical machine learning tools and techniques. 2nd ed. San Francisco, CA: Morgan Kaufmann; 2005. p. 525.
[20] Frank E, Wang Y, Ingis S, Holmes G, Witten IH. Using model trees for classification. Machine Learning 1998;32(No. 1):63–76.
[21] Seidel P, Seidel A, Herbarth O. Multilayer perceptron tumor diagnosis based on chromatography analysis of urinary nucleoside. Neural Networks 2007;20(No. 5):646–51.
[22] Dietterich TG. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Machine Learning 2000;40(No. 2):139–57.
[23] Hothorn T, Lausen B. Bundling classifiers by bagging trees. Computational Statistics and Data Analysis 2005;49(No. 4):1068–78.
[24] Shah S, Kusiak A. Cancer gene search with data-mining and genetic algorithms. Computers in Biology and Medicine 2007;37(No. 2):251–61.
[25] Available from: <http://en.wikipedia.org/wiki/Logistic_function>. Accessed November 20, 2007.
[26] Available from: <http://en.wikipedia.org/wiki/Least_squares_method>. Accessed November 20, 2007.
[27] Kang L, Albin SL. On-line monitoring when the process yields a linear profile. Journal of Quality Technology 2000;32(No. 4):418–26.
[28] Lange M, Focken U. Physical approach to short-term wind power prediction. Berlin: Springer-Verlag; 2006.
[29] Barbounis TG, Theocharis JB, Alexiadis MC, Dokopoulos PS. Long-term wind speed and power forecasting using local recurrent neural network models. IEEE Transactions on Energy Conversion 2006;21(No. 1):273–84.