# A Wind Turbine Fault Detection Approach Based on Cluster Analysis and Frequent Pattern Mining

**Frank Elijorde[1], Sungho Kim[2], and Jaewan Lee[3]**
[1]*Institute of Information and Communication Technology, West Visayas State University*
Iloilo City, Philippines
[2]*Department of Control and Robotics Engineering, Kunsan National University*
[3]*Department of Information and Communication Engineering, Kunsan National University*
Gunsan, South Korea
[e-mail: frank, shkim, jwlee @kunsan.ac.kr ]
*Corresponding author: Jaewan Lee

*Received December19, 2013; accepted January 21, 2014; published February 28, 2014*

## *Abstract*

Wind energy has proven its viability by the emergence of countless wind turbines around the world which greatly contribute to the increased electrical generating capacity of wind farm operators. These infrastructures are usually deployed in not easily accessible areas; therefore, maintenance routines should be based on a well-guided decision so as to minimize cost. To aid operators prior to the maintenance process, a condition monitoring system should be able to accurately reflect the actual state of the wind turbine and its major components in order to execute specific preventive measures using as little resources as possible. In this paper, we propose a fault detection approach which combines cluster analysis and frequent pattern mining to accurately reflect the deteriorating condition of a wind turbine and to indicate the components that need attention. Using SCADA data, we extracted operational status patterns and developed a rule repository for monitoring wind turbine systems. Results show that the proposed scheme is able to detect the deteriorating condition of a wind turbine as well as to explicitly identify faulty components.

***Keywords:*** Condition Monitoring System, Wind Turbine, SCADA, cluster analysis, frequent pattern mining

A preliminary version of this paper was presented at APIC-IST 2013 and was selected as an outstanding paper. This research was financially supported by the Ministry of Education Science Technology (MEST) and National Research Foundation of Korea (NRF) through the Human Resource Training Project for Regional Innovation

**http://dx.doi.org/10.3837/tiis.2014.02.0020**

## 1. Introduction

The large-scale wind energy industry is relatively new and is rapidly expanding. Today, wind energy is considered the fastest growing alternative source of electricity around the world. As a result, wind farms contribute a significant volume of electrical generating capacity as they consist hundreds of units built worldwide. Most Wind Turbines (WT) are three-blade units composed of a number of major components. Driven by the wind, the blades and rotor is able to pass energy to the generator, such that the generator speed is as close as possible to optimal generation of electricity. Therefore, the ability of a wind turbine to extract power from the wind is a function of three main factors: the measured wind speed, the power curve of the turbine, and the ability of the turbine to handle wind fluctuations [1]. Most subsystems in wind turbines may fail during operation, including rotors and blades, pitch control systems, gearboxes and bearings, yaw systems, generators, power electronics, electric controls and brakes among others. As wind turbines are located at remote locations that may be difficult to access, their maintenance becomes an issue.

An efficient condition monitoring system for wind turbines is required to ensure operational reliability, high availability of energy production and at the same time reduce operating and maintenance costs [2]. As wind turbines improved their capacity, preventive maintenance has become more favorable. Preventive maintenance can be easily carried out by scheduled maintenance which involves the repair or replacement at regular time intervals as recommended by the supplier and regardless of condition. However, reducing failures in this manner comes at the cost of performing maintenance tasks more frequently than necessary and fully utilizing the functional lifetime of the various components. For that matter, an excellent alternative is to prevent major component failure and system breakdown with condition based maintenance (CBM) in which continuous monitoring and inspection techniques are employed to detect incipient faults early, and to determine any necessary maintenance tasks ahead of failure [3]. CBM has been shown to minimize the costs of maintenance, improve operational safety, and reduce the quantity and severity of in-service system failures. Currently, modern turbines come with some form of integrated system that can monitor the main componentsand keep track of various. Data regarding these parameters is collected and stored via a supervisory control and data acquisition (SCADA) system that usually archives the information for all of the turbines in the wind farm. The accumulated data could be beneficial if analyzed and interpreted automatically to assist operators in detecting and identifying WT faults.

The fault effect is commonly observed in WT power curve in which the curve slightly deviates from its normal position prior to a maintenance period. However, the symptom is quite weak, generic and not always considered as a convincing proof of the fault. Because of this, it is imperative that even though the power curve is popularly used in WT SCADA systems, it still has its limitation in detecting component-level faults. In this paper, we aim to tackle the problem by uncovering more relevant information hidden in SCADA data. By combining cluster analysis and data mining, we assert that the faulty WT status patterns in SCADA data would be easily detected once the operational condition has been captured. Thus, any deviation of the actual value from its expected value would indicate a fault. Since the value of SCADA data is dependent on WT operational conditions, the strategy to properly classify status patterns as well as to evaluate the deviations under varying conditions is our main concern.

## 2. Related Work

### 2.1 Condition Monitoring Systems

To date, vibration analysis remains the most popular condition monitoring technology employed in WT especially for rotating equipment [4]. It is well-suited for monitoring the gearbox, bearings, and other selected WT elements. The measurement and interpretation of acoustic emission parameters for fault detection in ball bearings has been demonstrated at different speed ranges in [5]. Furthermore, its application for the detection of bearing failures has been presented in [6]. In a way, acoustic monitoring is similar with vibration monitoring only that vibration sensors are installed on the component involved to detect movement [7]. From a case study of a WT gearbox in [8], vibration may possibly not be evident while faults are developing, but analysis of the oil can provide early warnings. For lifetime forecasting and protection against high stress levels especially in the blades, stress measurement is another viable option. In [9], an assessment of strain gauge signal interpretation from strain gauge sensors installed on the blade has been performed in order to adjust calibration practices and sensor selection. Thermography is often used for monitoring electronic and electric components and identifying failure. The technique is only applied off-line, and often involves visual interpretation of hot spots that arise due to bad contact or a system fault. The work in [10] used infrared cameras to visualize variations in blade surface temperature and can effectively indicate cracks as well as places threatened by damage.

### 2.2 Cluster Analysis in Wind Turbine Operation

Clustering has been long proven to be useful in several exploratory pattern analysis, grouping, decision making, and machine learning applications. For WTs, an accurate monitoring of a turbine's performance is instrumental for detecting a potentially deteriorating state. In [11], a performance monitoring system for wind turbines based on a hybrid neuro-fuzzy clustering is presented. By taking advantage of the combined strengths of neural networks and fuzzy inference systems, an accurate modeling of wind turbine performance is established. The work in [12] proposes a method to cluster wind turbines, which can be applied for modeling a large-scale wind farm in complex terrain or irregular layout. According to the real-time operating data, the wind turbines are divided into different groups by the method which is based on spectral clustering algorithm to capture the similarity of output characteristics of wind turbines. In [13] they presented an approach for fault detection using available SCADA-data from wind turbines. Systematic analysis of data indicated clear distinctions between fault and no-fault conditions in relationships among several parameters. These distinctions in relationships were exploited in the development of automated fault diagnostics algorithms. Principal component analysis and self-organizing feature maps were used in the algorithms. Fuzzy clustering method and similarity theory is used in [14] to classify different periods into different time category and then choose a fixed output value to represent the output of wind power in category respectively. The wind speed distribution function is used to describe the characteristics of classified wind speed data, and expected output of wind power is obtained via a typical wind turbine power curve.

### 2.3 Data Mining in Wind Turbine CMS

Data mining has been successfully utilized in several applications in manufacturing, marketing, medical informatics, and the energy industry. The previous efforts using data

mining in wind energy has primarily focused on estimating and optimizing the power output. Using data mining techniques for modeling WT performance, we can take advantage of the data provided by SCADA systems. In [15], they presented a review of works focused on forecasting wind speed and generated power using both physical models and data mining methods. Moreover, models for long-term and short-term predictions of power with data mining techniques are discussed in [16]. In [17], four data-mining approaches for wind turbine power curve monitoring are compared. Power curve monitoring is applied to evaluate the turbine power output and detect deviations which cause financial loss. While in [18], they presented a generic approach to use appropriate techniques for forecasting based on data availability and its characteristics which can be exclusively used for very short term period with a 30 minutes time interval. In [19], a short-term prediction model with a maximum 12-hour forecast length and a long-term prediction model with a maximum 84-hour forecast length were built using weather forecasting data as predictors. The boosting tree algorithm and PCA transformation were used to reduce the predictor data dimension and enhance prediction accuracy.

## 3. Fault Detection Based on Cluster Analysis and Frequent Pattern Mining

In this section, we discuss the design of our proposed WT fault detection strategy. As shown in **Fig. 1**, the fault detection scheme is composed of various procedures which include the acquisition and pre-processing of SCADA data, clustering and classification, itemset generation, frequent pattern mining, rule generation, and fault detection.

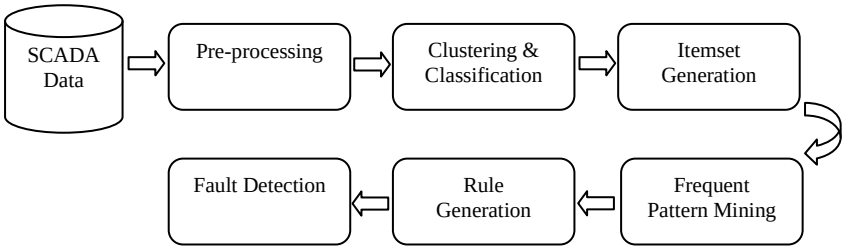


**Fig. 1**. Overall system design.

### 3.1 Data Acquisition and Pre-processing

The fault detection system presented in this work is applicable to wind turbines equipped with SCADA system. During the operation of a wind turbine, a normal behavior can be characterized by its power curve. An apparent advantage of using normal behavior models to monitor wind turbine signals is that no prior knowledge about the signal behavior is necessary. Thus normal behavior SCADA data were used in the initial stage of clustering and classification.

However, it is known that even at a normal status, the power output of a wind turbine could sometimes be inconsistent with the reported wind speed. This is for the reason that the blade pitch angle also needs to be restrained so as to protect the turbine against extreme winds. This would obviously generate data which are actually deviant from the expected normal behavior. Hence, it is crucial that we first identify outliers and eliminate them from the data set so as to assure its integrity. Since data mining algorithms construct models using large datasets, it

requires data preprocessing that is time-consuming. Thus, a significant portion of the analysis time may be spent on data sampling, parameter selection, and other data analysis tasks.

### 3.2 Cluster Analysis Using K-means

Clustering has been long proven to be useful in several exploratory pattern analysis, grouping, decision making, and machine learning applications. In this work, K-means Clustering [20] is used to perform classification of SCADA data. The K-means clustering, being simple and generally fast makes it easy to handle large amounts of data, therefore widely used in signal processing. Briefly, the following procedures show the major steps of applying the K-means clustering algorithm to the clustering and classification of SCADA data:

1) Specify the number of *K* clusters.
2) Initialize the centroid for each K cluster. This is done by randomly dividing all objects into K clusters, deriving their centroids, and verifying that all centroids are different from each other. As an alternative, the centroids can be initialized to K randomly chosen objects.
3) Iterate over all objects while computing their distances to the centroids of all clusters. Each object is then assigned to the cluster with the nearest centroid.
4) Recalculate the centroids of both modified clusters.
5) Repeat step 3 until the centroids do not change any more.

Each SCADA log composed of various turbine operation parameters is assigned to the closest centroid. To do this, a distance function that quantifies the perception of closeness for the specific data under consideration is needed. The function used is the Euclidean Distance defined as:

$$d(x,y) = \sqrt{\sum_{i=1}^{m}(x^i - y^i)^2} \tag{1}$$

where $x = (x_i, \ldots x_m)$ and $y = (y_i, \ldots y_m)$ are two input vectors with $m$ quantitative features. In the Euclidean distance function, all features contribute equally to the function value. The optimal clustering number K is then identified based on Davies & Bouldin rule [21]. The Davies & Bouldin factor is defined as:

$$DB = \frac{1}{k}\sum_{i=1}^{k} max\left\{\frac{d_i + d_j}{D_{ij}}\right\} \tag{2}$$

In which, $d_i$ is the average distance in class $i$, $d_j$ is the average distance in class $j$ , $D_{ij}$ is the distance between class $i$ and class $j$ . Clustering result is the best when $DB$ reaches minimum. So, $k$ is identified when $DB$ reaches minimum in the range from 2 to 10.

The raw SCADA data is complex and highly dimensional. Prior to subjecting the data set to cluster analysis, we need to select a number of suitable WT performance parameters. In this work, we used power, wind speed, pitch angle, and rotor speed. Because of their correlations, they would provide valuable hints for detecting WT faults. As a result of the clustering process, the entire data set is divided into K clusters. Each cluster covers a subset of the SCADA data composed of the readings for the four parameters. After all the WT operation logs have been assigned to the clusters, we can now generate the itemsets that classify parameter values based on their ranges. For example, if power output range is 626.2 to 648.65 power = "P1", if wind speed range is 10.1 to 11.4 wspeed = "W1", if pitch angle range is 1.92 to 3.57 pitch = "PA3",

if rotor speed range is 1914 to 1927.5 rpm = "RPM1". As shown in **Fig. 2**, applying this to all the clusters we can derive the itemsets which are then used to represent parameter values which are treated as frequent patterns.

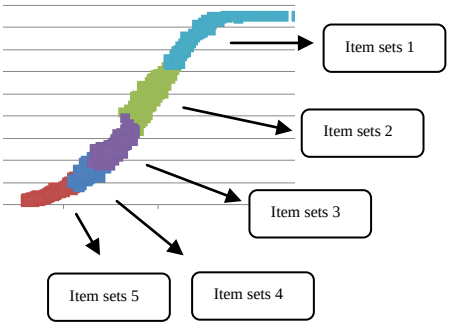


**Fig. 2**. Mapping of itemsets to clusters.

### 3.3 Frequent Pattern Mining with FP-Growth Algorithm

Frequent itemsets serve an important purpose in many data mining tasks aimed to find interesting patterns from databases. Originally, the idea of frequent itemset was intended for mining transaction databases. For example, there exists a set of all items $I = \{i_1, i_2, \ldots, i_n\}$. A $k$-itemset $\alpha$, composed of $k$ items from set $I$, is considered frequent if $\alpha$ occurs in a transaction database $D$ no lower than $\theta|D|$ times, where $\theta$ is a user-specified *minimum support threshold*, and $|D|$ is the total number of transactions in $D$. The task of discovering all frequent itemsets is quite challenging. This is due the fact that the search space is exponential in the number of items occurring in the database. The support threshold is intended to reduce the output to a confidently reasonable subspace. Thus, in the case of SCADA systems, such databases are expectedly massive, containing records of wind turbine operation which would make determining *support* a difficult problem.

The FP-Growth [22] algorithm mines the complete set of frequent itemsets without candidate generation by taking advantage of the *divide-and-conquer* strategy. The initial scan of the database retrieves a list of frequent items in which items are ordered according to their number of occurrence in a descending order. Based from the list generated, the database is compressed into a frequent-pattern tree, or *FP-tree*, in which the itemset association information is kept. Starting from each frequent length-1 pattern, the FP-tree is mined by constructing its *conditional pattern base* which is a "subdatabase", then constructing its conditional FP-tree, in which mining is performed recursively. Finally, the pattern growth is achieved by the concatenation of the suffix pattern with the frequent patterns generated from a conditional FP-tree.

Applying the method, the first scan of the data set in **Table 1** derives the set of frequent items and their support counts. With a minimum support threshold of 2, the set of frequent items is sorted according to support count in descending order. This resulting set or *list* is denoted as $L$ = {{P1: 6}, {WS2: 4}, {PA3: 4}, {RPM3: 3}, {RPM4: 3}, {P2:2}, {WS3:2}, {WS4:2}, {PA4:2}, {PA5:2}, {RPM:2}}. Once the list of frequent items is ready, the FP-tree is then constructed. First, the root of the tree is created and labeled with "null". The database $D$ is then scanned for the second time. At this point, the items in each record are sorted according to descending support count and a branch is created for each record, whereas the items represent the nodes. For instance, the scan of the first record, "P1, WS2, PA5, RPM4" containing four

items will be represented as “P1, WS2, RPM4, PA5”. This is followed by the construction of the first branch of the tree with four nodes, <P1: 1>, <WS2:1>, <RPM4:1>, and <PA5: 1>, where <P1> is linked as a child of the root, <WS2> is linked to <P1>, <RPM4> is linked to <WS2> and <PA5> is linked to <RPM4>. Meanwhile, the second record “102: P2, WS4, PA3, RPM3” will be re-ordered as “PA3, RPM3, P2, WS4”. The item set would result in a new branch directly connected to the root node, since there is currently no node labeled as PA3. Going to the third record, the item set will be ordered as “PA3, RPM4, P2, WS3”. However, the resulting branch would share a common prefix PA3, with the existing branch generated from record 101. Thus, this branch will simply be linked to node PA3 and the count of node PA3 is incremented by 1.A new node <RPM4: 1> is then created, which is linked as a child of <PA3: 2>. Basically, whenever a branch is to be added for each record, the count of each node along a common prefix is incremented by 1, and nodes for the items following the prefix are created and linked accordingly.

| LogID | Itemset |
|---|---|
| 101 | P1, WS2, PA5, RPM4 |
| 102 | P2, WS4, PA3, RPM3 |
| 103 | P2, WS3, PA3, RPM4 |
| 104 | P1, WS2, PA4, RPM2 |
| 105 | P1, WS3, PA4, RPM2 |
| 106 | P1, WS4, PA5, RPM3 |
| 107 | P1, WS2, PA3, RPM3 |
| 108 | P1, WS2, PA3, RPM4 |

**Table 1.** Data set representing SCADA data.

To enable tree traversal, an item header table is built so that each item points to its occurrences in the tree via a chain of node-links. As shown in **Fig. 3**, the FP-Tree with the associated node-links is created after scanning all the records. With this approach, the problem of mining frequent patterns in SCADA databases is transformed to that of mining the FP-tree. After the FP-tree has been created, frequent itemsets can now be mined using the FP-Growth method. The extraction of itemsets is done in a bottom-up fashion, from the leaves towards the root. This is where the the header table is used to find the paths ending with X. For instance, it is shown in **Fig. 4** that paths ending with PA5 are found by following the node links.

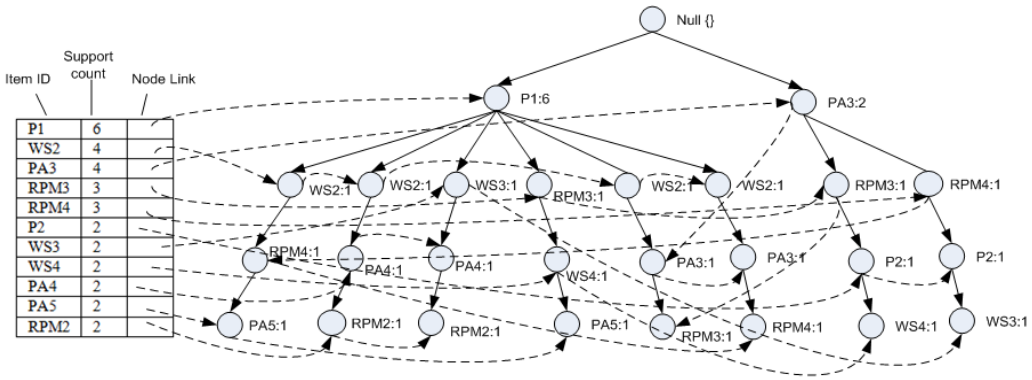


**Fig. 3**. The FP-Tree generated from the sample data set.

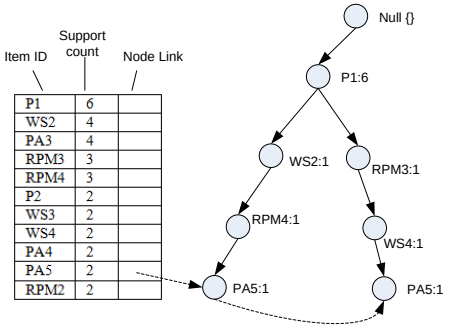


**Fig 4.** Conditional FP-Tree for node PA5.

These paths, often called prefix paths are derived by retaining only paths ending with PA5. Starting with leaf node PA5 which satisfies a minimum support threshold of 2, it is considered a frequent itemset. Doing the same process from the leaves going upward, the conditional FP-Tree for suffix PA5 is then used to solve {PA5, RPM4, WS2, P1} and {PA5, WS4, RPM3, P1}. Same process is used with leaf nodes RPM2, RPM3, RPM4, WS4, and WS3.

### 3.4 Generation of Rules from Frequent Patterns

After all the itemsets which represent frequent patterns from turbine operation data have been found, we generate strong association rules from them. A rule is considered strong if it satisfies both minimum support and minimum confidence. Rule support and confidence are two measures of rule interestingness. They respectively reflect the usefulness and certainty of discovered rules. A rule's confidence measure can be derived as:

$$confidence(A \Rightarrow B) = \frac{support\_count(A \cup B)}{support\_count(A)} \tag{3}$$

The conditional probability is expressed in terms of itemset support count, where support_count(A ∪ B) is the number of records containing the itemsets A ∪ B, and support_count(A) is the number of records containing the itemset A. Because the rules are generated from frequent itemsets, each one automatically satisfies minimum support. For example given the itemset {P1, WS2, PA3, RPM4}, the following are some rules that can be generated: P1 ⇒WS2 ∧ PA3 ∧ RPM4, RPM4 ∧ P1 ⇒WS2 ∧ PA3, WS2 ⇒ RPM4 ∧ P1 ∧ PA3, WS2 ⇒ PA3 ∧ RPM4 ∧ P1. In **Table 2**, sample rules are shown.

| Rule | Description |
|---|---|
| P1 ⇒WS2 ∧ PA3 ∧ RPM4 | A power output of P1 is associated to a wind speed, pitch angle, and rotor speed classified as WS2, PA3, and RPM4 respectively. |
| WS2 ⇒ PA3 ∧ RPM4 ∧ P1 | A wind speed of WS2 is associated to a pitch angle, rotor speed, and power output classified as PA3, RPM4, and P1 respectively. |
| P1 ∧ WS2 ⇒ PA3 ∧ RPM4 | A power output of P1 and wind speed of WS2 is associated to a pitch angle and rotor speed classified as PA3, RPM4 respectively. |

**Table 2.** Sample rules generated from an itemset.

| Timestamp | Power Output | Wind Speed | Pitch Angle | Rotor Speed |
|---|---|---|---|---|
| 2011-07-07 PM 2:30:00 | P4 | WS1 | X | RPM9 |
| 2011-07-07 PM 2:40:00 | X | WS4 | X | X |
| 2011-07-07 PM 2:50:00 | X | WS3 | PA1 | RPM9 |
| 2011-07-07 PM 3:00:00 | P6 | WS3 | X | RPM4 |
| 2011-07-07 PM 3:10:00 | P5 | WS1 | X | RPM2 |
| 2011-07-07 PM 3:20:00 | P1 | WS2 | PA1 | RPM2 |
| 2011-07-07 PM 3:30:00 | P6 | WS4 | PA2 | X |
| 2011-07-07 PM 3:40:00 | P4 | WS2 | X | X |

**Table 3.** Sample scenario of a faulty turbine.

Once the rules have been established, the idea of detecting wind turbine fault is straightforward. Each time a turbine status pattern is generated it is compared against the existing rules, thus a deviating pattern is an indicator of the wind turbine's deteriorating condition. Moreover, we can further pinpoint specific faults by looking at the parameter which has the most deviations. A sample scenario shown in **Table 3** indicates that the turbine is suffering from blade pitch angle error as it shows that most of the deviating patterns are concentrated on the pitch angle parameter.

## 4. Implementation and Evaluation

### 4.1 Implementation

As discussed in the previous section, it is important that data should first undergo a pre-processing phase in order to remove the outliers which could affect the integrity of the derived model. Using K-means, 10 months of SCADA is processed and after a number of trials it was decided that 5 is the best value for the K number of clusters. For each cluster, a centroid value is calculated which represents the center of the cluster.

After all status data had been assigned to their respective clusters, the itemsets that will represent the respective values of the selected WT parameters were generated, and from which rules will be made. However, it should be noted that not all rules that can be possibly mined from the main data set are interesting; many of them are very rare or almost does not occur which makes them not good candidate rules. Nevertheless, many interesting rules can be found using low support thresholds. Thus to generate a reasonable number of rules that could cover a vast range of WT parameter values we utilized a support threshold of 5% and a confidence of 90%. Through this, we were able to discard a huge number of uninteresting and non-occurring patterns. In **Table 4** it shows that each cluster has a corresponding number of rule items for each parameter, the number of possible rules, and the number of interesting rules

that were extracted from them.

| Cluster | Number of Rule Items | | | | Total no. of Rules | Rules Extracted |
|---|---|---|---|---|---|---|
| | Power Output | Wind Speed | Pitch Angle | Rotor Speed | | |
| 1 | 10 | 2 | 2 | 10 | 400 | 128 |
| 2 | 10 | 2 | 3 | 10 | 600 | 95 |
| 3 | 10 | 3 | 3 | 10 | 900 | 121 |
| 4 | 10 | 2 | 2 | 8 | 320 | 87 |
| 5 | 10 | 6 | 10 | 2 | 1200 | 60 |

**Table 4.** Actual number of rules extracted from the total number of rules.

Finally, using the extracted rules, an FP-Tree was created as shown in **Fig 5**. The plausible rules are then stored to the rule repository which will be used for the analysis of SCADA data.

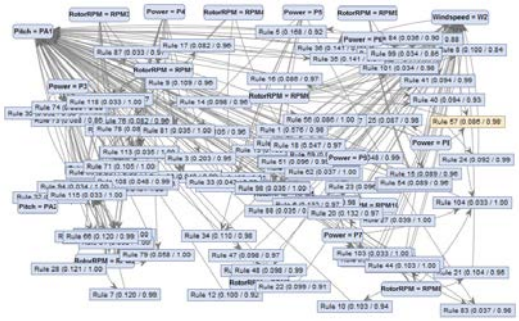

**Fig. 5.** The FP-Tree for the proposed scheme.

## 4.2 Evaluation

To evaluate the proposed scheme, another batch of SCADA data is used. Whenever a set of wind turbine parameters is captured, a corresponding status pattern is generated. It is then compared against the rules in the repository whereas the deviating patterns indicate the deteriorating condition of the wind turbine. Basically, the accumulating deviation of status patterns from the observed normal behavior is just a general indicator of a turbine's condition. Looking further into individual parameters, they can be used to identify faults at the component level through the number of deviations they exhibit. In this case, the miscorrelating patterns can be attributed to faults originating from the generator power, the pitch angle of blades, and the rotor speed.

As shown in **Fig. 6**, the status patterns of a normal SCADA were logged using the proposed approach. As can be seen, a number of fault patterns were detected; at less than 5% of the entire data set, it is still far from the alarming level. Next we fed another set of SCADA data into the system, this time faulty ones. In **Fig. 7a** to **Fig. 7c**, it can be seen that the proposed approach is able to indicate the respective fault levels of the three wind turbine components. The faulty status pattern logs are shown in its first week of manifestation where the figure indicates a looming fault within the turbine components. Within one month, it can be noticed that the level of fault as indicated by the pattern logs, have doubled. After two months of logging the status patterns, the fault levels have greatly increased to an alarming level. Through time, as the faults worsen, the number of fault patterns increases which indicates the deteriorating condition of the wind turbine as it comes close to a maintenance period. The

figure shows that most of the deviating patterns were caused by the power output and rotor speed parameters, while that of the pitch angle is not as serious. Using this information, the maintenance and repair of the turbines can be focused on the exact faulty components thus saving significant cost, time, and effort.

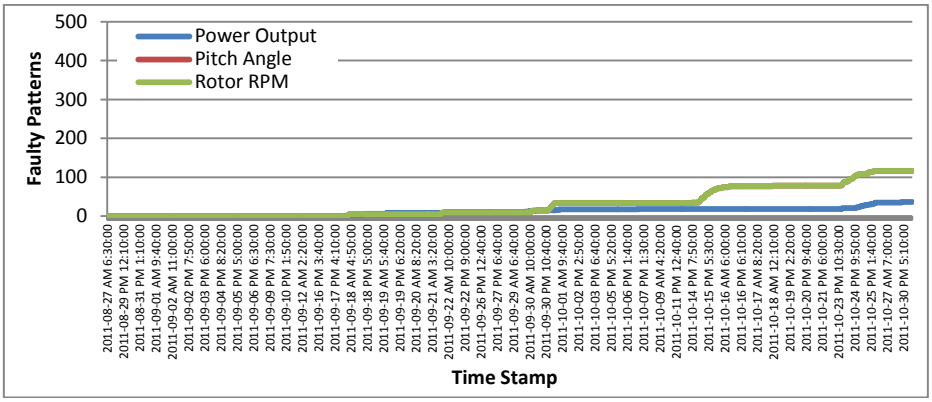


**Fig. 6**. Status pattern log of normal SCADA data.

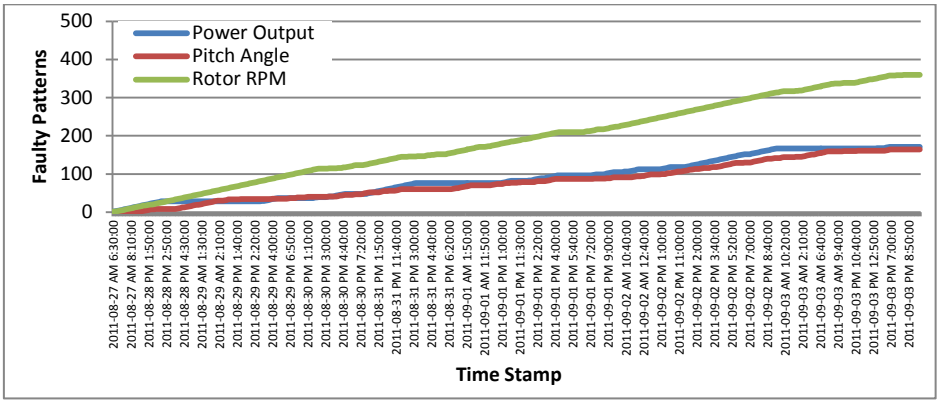


**Fig. 7a**. Status pattern log of faulty SCADA data at 1 week.

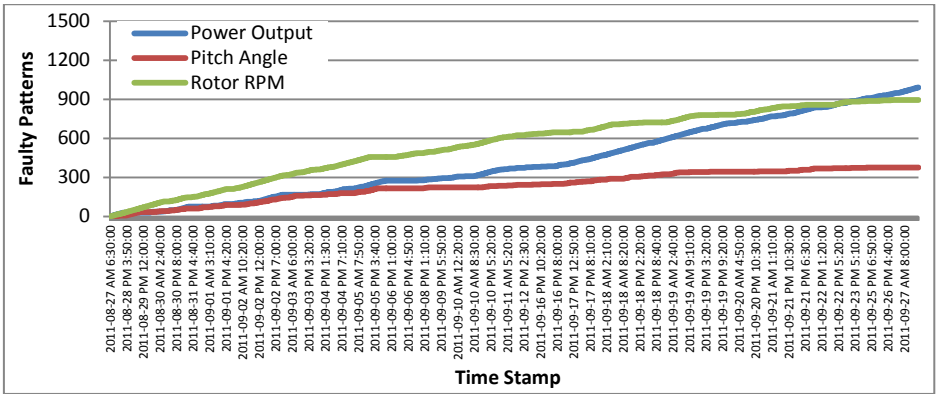

**Fig. 7b**. Status pattern log of faulty SCADA data at 1 month.

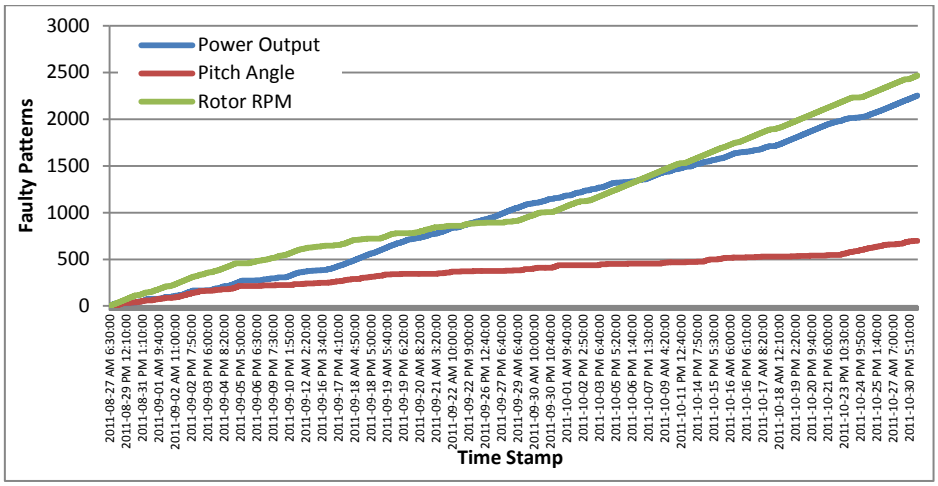


**Fig. 7c**. Status pattern log of faulty SCADA data at 2 months.

## 5. Conclusion

In this paper, an approach for wind turbine fault detection is presented. Aside from monitoring the condition of the wind turbine in general, the proposed scheme is also able to identify specific faults at the component level. By synergistically incorporating cluster analysis and data mining, we put forward a simple yet efficient method to keep track of a wind turbine's operating condition and its primary components. Initially, the system is modeled using considerable amount of operational SCADA data from which various ranges of parameter values were classified and represented as itemsets. This stage was carried out by the use of K-means Clustering which is a fast and simple technique well-suited for large data sets. After all data samples were assigned to a cluster, the next stage involved the use of the FP-Tree approach to extract frequent itemsets that represent regularly-occurring status patterns of a wind turbine. Finally, using the FP-growth technique, interesting rules were derived from the itemsets which were stored in the rule repository to be used as benchmark for incoming status readings from SCADA data. Evaluation results show that the proposed approach is able to keep track of a wind turbine's overall condition as well as to pinpoint specific faulty components. By testing the proposed scheme using normal and faulty SCADA data, our work is able to accurately determine whether a wind turbine is in pristine or deteriorating condition. Moreover, we can easily determine which component has the most serious fault level. By enabling wind turbine operators to explicitly identify faults, significant savings in terms of cost, time, and effort are gained by performing timely and well-advised troubleshooting and maintenance routines.

## References


[1] B. Boukhezzar, H. Siguerdidjane, and M. M. Hand, “Nonlinear control of variable-speed wind turbines for generator torque limiting and power optimization,” *ASME Trans.: J. Solar Energy Eng.*, vol. 128, no. 4, pp. 516–531, 2006. Article (CrossRef Link).

[2] M. Kruger, S.X. Ding, A. Haghani, P. Engel, T. Jeinsch, "A data-driven approach for sensor fault diagnosis in gearbox of wind energy conversion system," In *Proc of 10th IEEE International Conference on Control and Automation*, pp.227-232, 2013. Article (CrossRef Link).

[3] F.P. Garcia Marquez, D.J. Pedregal, and C. Roberts, “Time series methods applied to failure prediction and detection,” *Reliability Engineering & System Safety*, pp. 698-703, 2010. Article (CrossRef Link).

[4] Z. Hameed, Y.S. Hong, Y.M. Choa, S.H. Ahn, and C.K. Song, “Condition monitoring and fault detection of wind turbines and related algorithms: a review,” *Renewable and Sustainable Energy Reviews*, pp. 1-39, 2009. Article (CrossRef Link).

[5] N. Tandon, B.C. Nakra, “Defect detection in rolling element bearings by acoustic emission method,” *Journal of Acoustic Emission* pp. Vol 9, No. 1, pp. 25-28, 1990.

[6] C.C. Tan, “Application of acoustic emission to the detection of bearing failures,” In *Proc of Tribology Conference*, pp. 110-114, 1990.

[7] T.W. Verbruggen TW, “Wind turbine operation and maintenance based on condition monitoring,” *WT-O. Final report*, 2003.

[8] S. Leske and D. Kitaljevich, “Managing gearbox failure,” *Dewek. Dewi Magazine*, No. 29, 2006.

[9] E. Morfiadakis, K. Papadopoulos, and T.P. Philippidis, “Assessment of the strain gauge technique for measurement of wind turbine blade loads,” *Wind Energy*, Vol. 3 No. 1, pp. 35-65, 2000. Article (CrossRef Link).

[10] M.A. Rumsey and W. Musial, “Application of infrared thermography nondestructive testing during wind turbine blade Tests,” *Journal of Solar Energy Engineering*, 2001. Article (CrossRef Link).

[11] F.I. Elijorde, D. Moon, S. Ahn, S. Kim, and J. Lee, “Wind Turbine Performance Monitoring Based on Hybrid Clustering Method,” *Future Information Communication Technology and Applications, Lecture Notes in Electrical Engineering*, Vol. 235, pp. 317-325, 2013. Article (CrossRef Link).

[12] L. Lin, Y. Chen, N. Wang, "Clustering wind turbines for a large wind farm using spectral clustering approach based on diffusion mapping theory," In *Proc of IEEE International Conference on Power System Technology* pp.1-6, 2012. Article (CrossRef Link).

[13] K. Kim, G. Parthasarathy, O. Uluyol, W. Foslien, S. Sheng, and P. Fleming, “Use of SCADA Data for Failure Detection in Wind Turbines,” In *Proc of Energy Sustainability Conference and Fuel Cell Conference*, 2011.

[14] L. Suhua, L. Zhiheng, and W. Yaowu, "Clustering analysis of the wind power output based on similarity theory," *Electric Utility Deregulation and Restructuring and Power Technologies*, In *Proc of DPRT International Conference*, 2008. Article (CrossRef Link).

[15] L. Ma, S. Luan, C. Jiang, H. Liu, and Y. Zhang, “A review on the forecasting of wind speed and generated power,” *Renew. Sustain. Energy Rev.*, vol. 13, no. 4, pp. 915–920, 2009. Article (CrossRef Link).

[16] A. Kusiak, H. Zheng, and Z. Song, “Short-term prediction of wind farm power: A data-mining approach,” *IEEE Trans. Energy Convers.*, vol. 24, no. 1, pp. 125–136, 2009. Article (CrossRef Link).

[17] M. Schlechtingen, I.F. Santos, S. Achiche, "Using Data-Mining Approaches for Wind Turbine Power Curve Monitoring: A Comparative Study," *IEEE Transactions on Sustainable Energy*, vol.4, no.3, pp.671-679, 2013. Article (CrossRef Link).

[18] M. Negnevitksy and P. Jhonson, “Very short term wind power prediction: A Data Mining Approach,” *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008. Article (CrossRef Link).

[19] A. Kusiak , H. Zheng and Z. Song "Wind farm power prediction: A data-mining approach," *Wind Energy*, vol. 12, no. 3, pp.275 -293, 2009. Article (CrossRef Link).
[20] J. MacQueen, “Some methods for classification and analysis of multivariate observations,” In *Proc of the 5th Berkeley symposium on mathematical statistics and probability*, 1967.
[21] D.L. Davies and D. W. Bouldin, "A Cluster Separation Measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1979. Article (CrossRef Link).
[22] J. Han , J. Pei , and Y. Yin, “Mining frequent patterns without candidate generation,” In *Proc of ACM SIGMOD international conference on Management of data*, 2000. Article (CrossRef Link).

**Frank Elijorde** received his B.S. degree in Information Technology and M.S. degree in Computer Science from Western Visayas College of Science and Technology, Philippines, in 2003 and 2007 respectively. Currently, he is a Doctor of Engineering candidate in Information and Telecommunications and working as a research assistant at the Distributed Systems Laboratory. His research interests include distributed systems, cloud systems, data mining, ubiquitous sensor networks, and RFID.

**Sung-Ho Kim** received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Korea University in 1984, 1986, and 1991, respectively. Currently, he is a professor at the Department of Control & Robot Engineering in Kunsan National University, Gunsan City, South Korea. His research interests include wind turbine system, fault detection and diagnosis and process control system.

**Jaewan Lee** received his B.S., M.S., and Ph.D. degrees in Computer Engineering from Chung-Ang University in 1984, 1987, and 1992, respectively. Currently, he is a professor at the Department of Information and Communication Engineering in Kunsan National University, Gunsan City, South Korea. His research interests include distributed systems, database systems, data mining and cloud systems.