Advanced Engineering Informatics 46 (2020) 101139

Contents lists available at ScienceDirect

Advanced Engineering Informatics

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

Anomaly detection and critical attributes identification for products with multiple operating conditions based on isolation forest

Hansi Chen[a], Hongzhan Ma[a], Xuening Chu[a,*], Deyi Xue[b]

[a] *School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China*
[b] *Departments of Mechanical and Manufacturing Engineering, University of Calgary, Calgary, Alberta, Canada*

ARTICLE INFO

Keywords:
Condition monitoring data
Operating condition
Critical attributes identification
Isolation forest

ABSTRACT

Performance analysis of the existing mechanical products is critical to identifying design defects and improving product reliability. With the advances of information technologies, product operating data collected through continuous condition monitoring (CM) serve as main sources for analysis of performance and detection of anomaly. Most of the existing anomaly detection methods, however, are not effective when CM data are very high dimensional, leading to poor quality of assessment results. Besides, the effects of multiple operating conditions on anomaly detection are seldom considered in these existing methods. To solve these problems, an integrated approach for anomaly detection and critical behavioral attributes identification based on CM data is developed in this research. Gaussian mixed model GMM) is employed to develop a method for clustering of operating conditions. Isolation forest (*i*Forest) method is used to detect anomaly instances, and further to identify the critical attributes related to product performance degradation. The effectiveness of the developed approach is demonstrated by an application with collected operating data of a wind turbine.

1. Introduction

Modern mechanical products and systems such as wind turbines and aircraft engines are increasingly becoming complex, and these products and systems are often operated under irregular load patterns, intermittent durations and harsh weather conditions in their utilization stages [1]. As a result, it is a great challenge to improve the reliability of complex mechanical products and systems to maintain their performances. During the operation stage, the performance of a product may deviate from the expected one, leading to performance anomaly at some time instances or intervals. With the advances of information technologies such as product embedded information devices (PEIDs), microsensors, and wireless telecommunication, it is possible to continuously monitor product status including working environment and operational performance. The condition monitoring (CM) is often conducted through a collection of multi-dimensional operation data.

Many methods have been developed to analyze product performances with CM data. These methods are primarily classified into model-based approach and data-driven approach. In the model-based approach, mathematical models are employed to describe the physical products and their behaviors [2]. In this research area, Li and Lee [3] developed a model-based method to predict the remaining useful life of a gear considering fatigue crack. In this work, a gear dynamic model was constructed to simulate gear meshing dynamics [3]. Li et al. [4] developed a framework to predict the wear of wheels of railway vehicles, in which various models were developed for analysis of coupling dynamics between vehicle and track. These models were the coupling dynamics model, the three-dimensional contact geometry analysis model, and Archard wear model. In the model-based approach, sophisticated physical and engineering knowledge is required to describe the underlying physical processes that lead to system performance degradation or failure. The knowledge, however, is not always available for complex products and systems [5]. In addition, the high costs and special knowledge for achieving these models prevent them from being easily applied to other types of products and systems [6].

In the data-driven approach, on the other hand, statistical pattern recognition and machine learning methods are employed to detect changes in performance monitoring data [7]. In this approach, it is assumed that the statistical characteristics of the performance data remain relatively constant when the product or system is in healthy conditions. Many methods and tools, such as multivariate statistical methods [8,9], state space models [10,11], and regressive models [12,13], were developed to determine the health states of products and systems through detecting trends, patterns and/or anomalies in the operating data. In recent years, artificial intelligence methods such as artificial neural network (ANN) [14,15], recurrent neural network

[*] Corresponding author.
E-mail addresses: scirocco@sjtu.edu.cn (H. Chen), mahongzhan@sjtu.edu.cn (H. Ma), xnchu@sjtu.edu.cn (X. Chu), dxue@ucalgary.ca (D. Xue).

https://doi.org/10.1016/j.aei.2020.101139
Received 14 March 2019; Received in revised form 11 July 2020; Accepted 11 July 2020

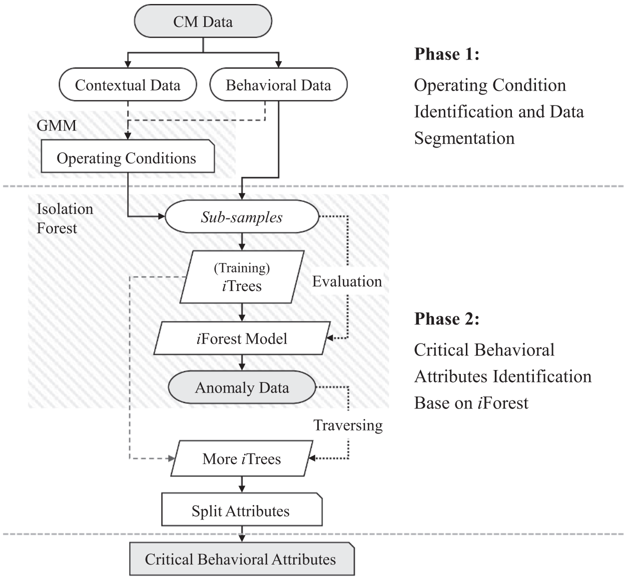


Fig. 1. The framework of the proposed approach.

(RNN) [16,17], dynamic wavelet neural network (DWNN) [18], self-organizing feature map (SOM) [19,20], support vector machine (SVM) [21], support vector regression (SVR) [22], relevance vector machine (RVM) [23], and extreme learning machine (ELM) [24] were also widely used for performance degradation assessment. These data-driven methods can be used to handle a variety of data types and exploit the nuances in the data that are hard to detect [25]. One of the advantages of the data-driven approach is that the behavior of the product/system is learned automatically from the collected monitoring data without specific knowledge on the product/system. Furthermore, data-driven approaches can be applied to complex products and systems such as wind turbines in wind farms, since the complex relationships between monitoring parameters and health statuses can be modeled by the data [5]. In addition, it is possible to use a data-driven approach to isolate the parameters that contribute significantly to the anomalous behaviors for identification of the causes.

A limitation of the data-driven approach lies on the assumption that the collected data for learning have to be in stable conditions (e.g., constant speeds and loads) which are not always true in the industry [1]. Mechanical products and systems are often operated under irregular load patterns and different environmental conditions during their operation stages. For example, a data instance might be considered normal in a given context, but another data instance with the same monitoring parameter values might be considered abnormal because it belongs to a different context [26]. When data from different operating conditions are mixed together for anomaly detection, some anomalies may be ignored due to the data points with similar values from other operating conditions. To avoid wrong decision-making in anomaly detection, clustering of multiple operating conditions should be conducted first.

Although many methods have been developed for anomaly detection and critical attributes identification, these methods are not effective for problems with high-dimensional CM data [27]. In some scenarios, real data sets may contain hundreds of dimensions. With increase of dimensionality, many conventional anomaly detection methods deteriorate either in efficiency, or in quality, or both in efficiency and quality. The *Isolation Forest* algorithm is used in this research to treat the CM data with high dimensions. Isolation forest is an unsupervised and nonparametric method specially designed for multivariate data anomaly detection. Isolation forest is one of the fastest anomaly detection methods that can be easily scaled up for problems with data in high dimensions.

To explore the applications of the product CM data for detection of anomalies and identification of critical attributes, three major issues need to be addressed.

(1) Clustering of multiple operating conditions from the collected CM data
(2) Detection of abnormal product CM data for each operating condition
(3) Isolation of critical attributes that cause anomalies

For addressing issue (1), the attributes related to operating conditions are first selected from both contextual and behavioral attributes. Then, a clustering method based on Gaussian mixed model (GMM) is employed to identify multiple operating conditions from the collected CM data. This process resolves the problem that some anomalies are covered when data from different operating conditions are blended for anomaly detection. To address the other two issues, a method for anomaly detection and critical behavioral attributes identification is introduced in this research based on the Isolation Forest method. Isolation Forest delivers high detection quality with high efficiency when dealing with high-dimensional data. The information contained in the anomaly detection process can be used to isolate critical attributes that cause anomalies.

The rest of this paper is organized as follows. The framework of this research is shown in Section 2. The newly developed approach is explained in detail in Section 3 and Section 4. A case study for evaluating the performance and discovering the causes of anomalies of a wind turbine is given in Section 5 to demonstrate the effectiveness of the developed approach. Conclusions and future work are provided in Section 6.

2. Framework

A systematic approach is developed in this research to identify critical behavioral attributes leading to anomaly as shown in Fig. 1. This approach is composed of two phases: (1) operating condition identification and data segmentation, and (2) critical behavioral attributes identification base on *i*Forest. In Phase I, appropriate attributes related to operating conditions are identified from both the contextual and behavioral attributes, and these attributes are then used for data clustering based on the GMM algorithm. In Phase II, anomaly detection is conducted for each of the multiple operating conditions, and critical behavioral attributes leading to those anomalies are identified based on the isolation forest method. Details of these two phases are provided in Sections 3 and 4, respectively.

3. Phase I: operating condition identification and data segmentation

Since ignoring the multiple operating conditions can have significant negative impacts on anomaly detection, the classification of CM data based on multiple operating conditions should be conducted first. In this study, GMM is used for operating conditions clustering. Attributes related to operating conditions are selected first as the input data. Then, the CM data are divided into different segments according to the clustering results of operating conditions. These clustered data sets are used in Phase II for anomaly detection.

3.1. Selection of attributes related to operating conditions

Generally, the CM data are composed of *contextual attributes* (also referred as environmental attributes) and *behavioral attributes* (also referred as indicator attributes) based upon whether the attributes can be used to describe an anomaly [28]. The values of contextual attributes are not directly used for describing an anomaly. However, these contextual attributes cannot be ignored because they have influences on the behavioral attributes whose values are directly related to anomalous events.

The contextual attributes are used to model the context for an instance. Typical contextual attributes include time, geographical location, temperature, humidity and other environmental factors. For example, in a data set collected through monitoring the air quality in major cities across the country, the longitude and latitude of a location are the contextual attributes. The behavioral attributes are used to define the non-contextual characteristics of an instance. For example, in a data set for describing the air quality in major cities across the country, the air quality index at a particular location is a behavioral attribute.

The first step for operating conditions clustering is to determine attributes related to operating conditions. Both contextual attributes and behavioral attributes can be selected in this step. The selection of appropriate attributes related to operating conditions serves as the basis for effective clustering of operating conditions. For different products and systems with different forms and functions, different attributes are selected for the clustering of operating conditions. The following rules can be used as a guideline for the selection of attributes related to operating conditions.

(1) The selected attribute should have a direct or indirect impact on the operating state of the product/system. For example, *transmission position* is a typical attribute related to the operating conditions of a vehicle.
(2) The attributes related to the default operating conditions described in the technical specifications should be selected.
(3) The contextual attributes that cause corresponding changes in the operating states should be selected.
(4) The behavioral attributes for modeling performance behaviors should be selected. For example, *output power* and *fuel consumption* are two behavioral attributes of a vehicle.

3.2. Clustering of CM data base on GMM

The attributes selected in Section 3.1 are then used to cluster the operating conditions based on Gaussian mixed model (GMM). Among various data clustering methods such as k-means method, k-nearest-neighbors (KNN) method and hierarchical clustering method, GMM method has been selected in this study due to the following two considerations. First, those selected attributes of CM data are generally complex data with nonparametric distributions, and GMM is a powerful tool for data clustering when the input data have such characteristics [29]. Second, the results of data-based operating condition identification sometimes differ slightly from the actual situation, e.g., data belonging to the same operating condition is segmented into two groups due to uneven data distribution or insufficient data volume. Therefore, the results of data clustering should be checked and adjusted, if necessary, based on the theoretical knowledge of the specific product. Since the probability density of each point corresponding to each component (i.e., a cluster) is calculated, GMM-based clustering provides more information for each data rather than simply classifying it into a cluster. Such information can be used to adjust the results of operating condition identification and data segmentation.

In GMM, data $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ in $\mathbf{R}^D$ are assumed to arise from a random vector with density

$$\Phi(\mathbf{x}) = \sum_{k=1}^{K} p_k \phi(\mathbf{x}|\mu_k, \Sigma_k) = \sum_{k=1}^{K} p_k \frac{1}{(2\pi)^{D/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T(\Sigma_k)^{-1}(\mathbf{x}-\mu_k)} \tag{1}$$

where $\Phi(\mathbf{x})$ is the probability density function in GMM distribution model, K is the number of single Gaussian components, p_k is the mixing proportion ($0 < p_k < 1$ for all $k = 1, \cdots, K$ and $\sum_k p_k = 1$)), and $\phi(\mathbf{x}|\mu_k, \Sigma_k)$ is the density of a D-dimensional multivariate Gaussian random variable with mean μ_k and covariance matrix Σ_k. Generally, the mixture parameters $\theta = (p_1, \cdots, p_K, \mu_1, \cdots, \mu_k, \Sigma_1, \cdots, \Sigma_K)$ are estimated by maximizing the loglikelihood:

$$L(\theta|\mathbf{x}_1, \cdots, \mathbf{x}_n) = \sum_{i=1}^{n} \ln\left[\sum_{k=1}^{K} p_k \phi(\mathbf{x}_i|\mu_k, \Sigma_k)\right] \tag{2}$$

The standard tool for finding the maximum likelihood solution is the expectation–maximization (EM) algorithm [30]. However, the EM algorithm has some known limitations [31], including (1) the number K of mixing components is usually unknown, (2) there is no widely accepted ‘good’ method for initializing the parameters, and (3) the algorithm can get trapped in one of the many local maxima of the likelihood function. Therefore, a greedy EM algorithm [31] for learning a Gaussian mixture is used in this work to overcome these limitations. The process of GMM clustering using a greedy EM algorithm is composed of the following steps:

Step 1: Initialize the parameters considering one component with $\mu = E[\mathbf{x}]$ and $\Sigma = \mathrm{Cov}[\mathbf{x}]$.
Step 2: Perform EM steps until convergence $\left|\frac{L_k^t}{L_k^{t-1}} - 1\right| < 1e^{-6}$ is achieved. Terminate if appropriate stopping conditions are satisfied.
Step 3: Search over all $\mathbf{x}_k$ for candidate locations of the new component. Set μ to the $\mathbf{x}_k$ that maximizes

$$\hat{L}_{k+1} = \sum_{i=1}^{n} \log\frac{f_k(\mathbf{x}_i) + \phi(\mathbf{x}_i|\mu, \Sigma)}{2} + \frac{1}{2}\frac{\left[\sum_{i=1}^{n} \frac{f_k(\mathbf{x}_i) - \phi(\mathbf{x}_i|\mu,\Sigma)}{f_k(\mathbf{x}_i) + \phi(\mathbf{x}_i|\mu,\Sigma)}\right]^2}{\sum_{i=1}^{n}\left[\frac{f_k(\mathbf{x}_i) - \phi(\mathbf{x}_i|\mu,\Sigma)}{f_k(\mathbf{x}_i) + \phi(\mathbf{x}_i|\mu,\Sigma)}\right]^2} \tag{3}$$

where $f_k(\mathbf{x})$ is the mixture density for a random vector $\mathbf{x}$ assuming with k components, $f_k(\mathbf{x}) = \sum_{l=1}^{k} p_l \phi(\mathbf{x}|\mu_l, \Sigma_l)$.

Step 4: Initialize the partial EM with the estimated value of $\mu$and Σ. Apply the partial EM until convergence condition defined in Step 2 is reached.
Step 5: If $L_{k+1} \le L_k$, then terminate. Otherwise allocate the new component and go back to Step 2.
Step 6: Adjust the optimal $\hat{k}$ ($\hat{k} \le k$) based on some existing selection criteria. In this study, the number of default operating conditions is used as the reference value of $\hat{k}$.
Step 7: After the estimated values of μ, Σ and p_k are obtained, the probability density of each point corresponding to each component is calculated, and then each point is assigned to the component with the highest probability density.

By applying the above steps, CM data is segmented into several clusters. The optimal number of components obtained in GMM is used as the number of operating conditions. In practice, the number of operating conditions obtained by data clustering may need to be adjusted based on specific product.

4. Phase II: critical behavioral attributes identification based on *i*Forest

For each data set generated in Phase I, *anomaly detection* is conducted to identify any performance deviations of the product/system from the required ones and simultaneously to provide information about the severity of the anomalies. Only the behavioral attributes are considered in this phase. Anomaly detection is primarily carried out in two steps:

(1) The baseline (training) data are first analyzed for determining what values are normal or typical for the behavioral attributes.
(2) When a test data instance is observed, it is labeled as normal or anomalous depending on the differences between its values of the behavioral attributes and the normal values of these behavioral attributes obtained in Step (1) [28].

The isolation-based anomaly detection method, isolation forest (*i*Forest) [32], is employed in this phase to identify anomaly behavioral data.

4.1. Anomaly detection with isolation forest

The Isolation Forest algorithm was selected for anomaly detection in this study for the following reasons. In practice, many CM data sets are very high in dimensionality (i.e., large number of attributes) and contain many attributes that are irrelevant to the problem. In high-dimensional space, the useful information is diluted, and the true patterns are blurred by the noises of less-important attributes, when the data are analyzed with full dimensionality. As one of the methods with subspaces, Isolation Forest delivers high detection quality with high efficiency when dealing with high-dimensional data. In addition, CM data are usually unlabeled and have only a small fraction of outliers. Therefore, the anomaly detection method should also be able to detect anomalies effectively when the training set contains normal instances only. In Isolation Forest, the presence of anomalies is irrelevant to its detection performance. In summary, Isolation Forest is an unsupervised, and nonparametric approach that works well when anomalies are not available in training sample [32].

In an isolation-based method, isolation susceptibilities of individual instances are measured, and anomalies are those with high susceptibilities [31]. In this method, *isolation* is the process to separate an instance from the rest of the instances. For example, in the two-dimensional dataset shown in Fig. 2(a), a binary search tree is randomly constructed to isolate the anomalous point (13,11) (shown in red circle) with just one separation operation, whereas the medoid point (7,7) (shown in yellow circle) is isolated with four separation operations. The anomaly point has depth of 1 to the root node in the tree, while the medoid point has depth of 4 to the root node in the tree (see Fig. 2(b)). The number of splitting operations required to isolate a point is the path length (i.e., depth) from the root node to the termination node in the tree. This path length can be used to measure abnormality since the path lengths for the anomaly points with the random partitioning process are noticeable short. When a particular observation has short path lengths in a forest of randomly created binary trees, this observation is highly to be anomaly. The anomaly score of the observation is calculated from the mean path length across all the trees in the forest.

Due to the large number of the collected data in an operating condition, anomaly detection using *i*Forest is carried out in a two-stage process to improve computation efficiency. In the first stage, isolation trees are constructed using sub-samples of the training dataset. In the second stage, every test instance is passed to each of all the isolation trees in the forest to obtain its path length. In the training stage, *i*Trees are constructed by recursively partitioning instances in the sub-samples until all instances are isolated. A split operation is conducted by randomly selecting an attribute, and then randomly selecting a split value between the maximum and minimum values of the selected attribute. Each split operation is defined by a split attribute and its value, and all these split operations are used to construct an *i*Tree as shown in green boxes in Fig. 2(b). Each *i*Tree is constructed using a sub-sample X' which is randomly selected from input data X. Two parameters should be selected for the *i*Forest algorithm in the training stage, the size of sub-samples ψ and the number of trees *t*. Since the number of sub-samples ψ is much smaller than the number of the collected data *N* in this operating condition, the *t* *i*Trees in the training forest can be created efficiently. In the evaluation stage, a forest of *i*Trees is built (see Fig. 3) for a given dataset. Anomalies are those instances, which have short average path lengths in the *i*Trees. A path length $h(x)$ is obtained by counting the number of edges from the root node to a termination node where the instance x is located through an *i*Tree.

Anomaly score is required for an anomaly detection method to evaluate the degree of anomaly quantitatively. For *i*Forest, the average path length cannot be used directly as the anomaly score due to the selection of different sizes of sub-samples. While the maximum possible height of *i*Tree is determined by the size of sub-samples ψ, the average height is related to the order of logψ. To avoid comparing the path lengths directly using the *i*Trees built with different sub-sample sizes, a normalized anomaly score is introduced in this research. In research on the isolation forest, Liu et al. [30] used an analysis method with a binary search tree (BST) to estimate the average path length of *i*Tree. For a given sample set with ψ instances, Preiss [33] calculated the average path length of unsuccessful searches in BST by:

$$c(\psi) = \begin{cases} 2H(\psi-1) - 2(\psi-1)/\psi & \text{for } \psi > 2, \\ 1 & \text{for } \psi = 2, \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $H(i)$ is the harmonic number estimated by $\ln i + 0.5772156649$ (Euler's constant). $c(\psi)$ is the average path length for a sample set with ψ instances. The anomaly score of an instance x is defined as:

$$s(x, \psi) = 2^{-\frac{E(h(x))}{c(\psi)}} \tag{5}$$

where $E(h(x))$ is the average of $h(x)$ from a collection of *i*Trees. According to Eq. (5), the following conclusions can be derived:

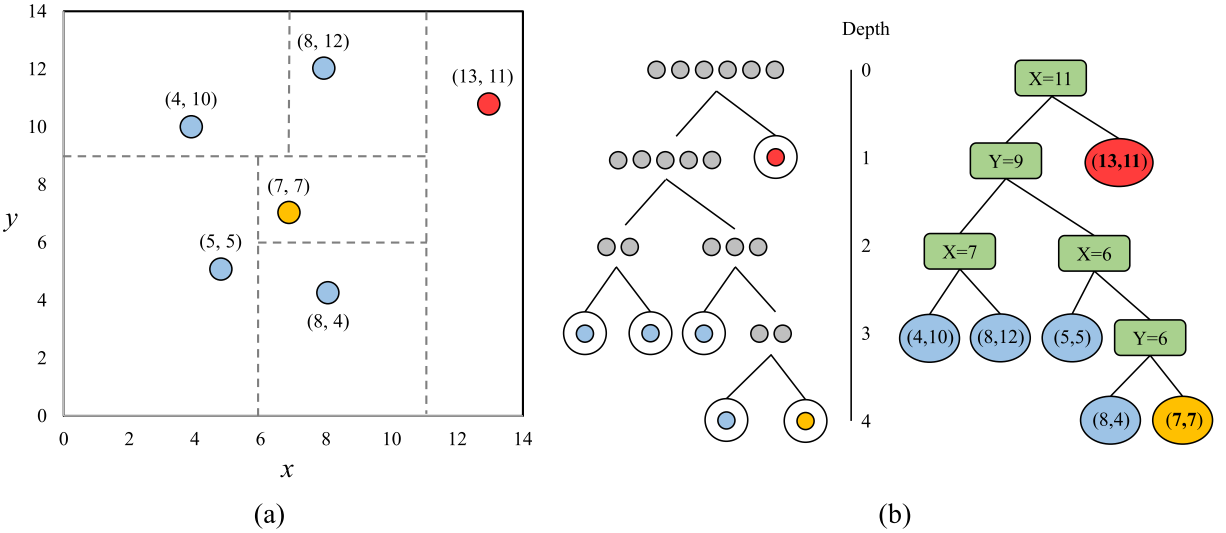


Fig. 2. A randomly constructed binary search tree to isolate a two-dimensional data set: (a) Data points and isolation operations; (b) Binary tree and isolation process.

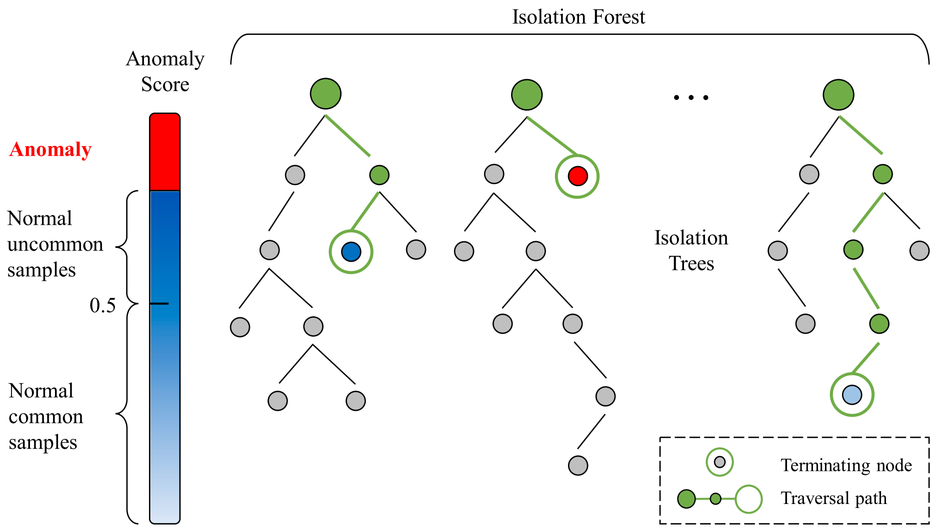


Fig. 3. Anomaly detection with *i*Forest.

(1) When $E(h(x)) \rightarrow 0$, $\rightarrow 1$. It indicates that when the average path length of this instance is close to zero, it is extremely easy to isolate this instance. This instance is considered an anomaly.
(2) When $E(h(x)) \rightarrow \psi - 1$, $s \rightarrow 0$. It indicates that when the average path length of this instance is close to the maximum height of the *i*Trees, it is extremely hard to isolate this instance. This instance is considered a normal common instance in the data set.
(3) When $E(h(x)) \rightarrow c(\psi)$, $s \rightarrow 0.5$. It indicates that it is not apparent whether the instance is anomaly.

After the anomaly scores of all instances are calculated, these instances are sorted based on descending order to find out the top anomalies. The detected anomalies indicate that the statuses of the product or system are different from the healthy baseline. They are suspected to be related to performance degradation. These anomaly field data are used in the next step to identify the attributes that contribute to the abnormal statuses of the product or system.

4.2. Critical behavioral attributes identification

After the detection of anomaly data, the attributes that contribute significantly to the anomaly data should be identified. The anomaly may be caused by an individual attribute or multiple attributes. Since these attributes lead to the changes in the performance of a product or system, it is critically important to identify these attributes for improvement of design. Although the abnormal instances for each operating condition are identified using isolation forest, these anomalies are still modeled by high-dimensional data with values of all behavioral attributes. Therefore, identification of the critical attributes from the abnormal data needs to be investigated. Based on the process to identify the abnormal data, a method is developed in this research to identify the critical behavioral attributes for abnormal data, and then to provide information for the identification of causes.

As described in Section 4.1, random decision trees of a forest are used in isolation forest method to detect data anomalies. In this method, observations are isolated by randomly selecting an attribute, and then randomly selecting a split value between the maximum and the minimum values of the selected attribute. The average number of splits over a forest with many random trees is used to measure the abnormality. When an instance is isolated with short path lengths collectively in a forest of random trees, this instance is highly to be anomaly [30]. In the process to detect anomalies using the *i*Forest method, only the average path length is used as an evaluation indicator. The information contained in the split operations on the paths to traverse the *i*Trees have not been utilized.

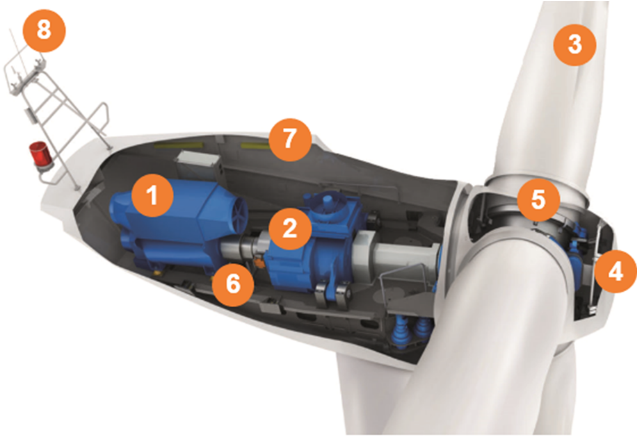
In practice, multi-dimensional CM data are typically composed of dozens of monitoring attributes. If an observation is easy to be isolated, it is often caused by one or several attributes deviating significantly from the normal points. As stated by Liu et al. [30], instances with distinguishable attribute-values were more likely to be separated in the early partitioning stage. Using the information contained in split operations, those critical dimensions can be identified. As illustrated in Fig. 2(a), the red point (13,11) is isolated early by split operation (X = 11). We can observe that the projection of this red point in the *y* direction (the value of dimension *y*) is very close to the projections of other points, but the projection of this red point in the *x* direction (the value of dimension *x*) is relatively far from the projections of other points. For this reason, when the selected attribute of the split operation is dimension *x* instead of dimension *y*, it is easier to 'isolate' point (13,11) from the others and the path length is short. Therefore, dimension *x* can be considered as the critical attribute of this point, and it 'causes' the average path of this point in a forest to be short, indicating the point is recognized as an anomaly.

Therefore, by recording the attributes to which split operations are conducted, the critical attributes can be obtained. For a *D*-dimensional data set, first *M* anomalies are identified through *i*Forest. Then, *T* trees are constructed to isolate anomalies. For each anomaly in each tree, the path length and the attributes for the split operations are recorded. For each anomaly, the shorter the path length is to traverse a tree, the higher the probability that the split operations contain critical attributes is. Hence, based on the anomaly score calculated by Equation (5), the weight w_t^m of the path of the *m*-th anomaly in the *t*-th tree can be defined as:

$$w_t^m = 2^{-\frac{h_t^m}{c(\psi)}} \tag{6}$$

where h_t^m is the path length of the *m*-th anomaly in the *t*-th tree. To ensure that all attributes appear at each depth with the same probability, sufficient trees should be constructed in the same way as in *i*Forest. Empirically, we have found that when T is selected as $128D$, sufficient trees are generated for isolation of critical attributes. The weights of split operations in different depths are the same, since the selection of attribute for each split operation is carried out randomly and independently. Therefore, the criticalness of attribute (dimension) *d* under certain operating condition can be defined as:

$$C_d = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} w_t^m f_d^{mt}}{N} \tag{7}$$



1. Generator/frequency converter 2. Gear system 3. Rotor blade

4. Rotor hub 5. Pitch system 6. Brake 7. Nacelle 8. Wind vane

Fig. 4. Main components/subsystems of Senvion MM82 wind turbine.

where f_d^{mt} is the frequency that the attribute d appears in the path of the m-th anomaly in the t-th tree, and N is the number of instances in this operating condition. In order to compare the criticalness measures of the same attribute among different operating conditions, N is used to normalize C_d.

5. Case study

The developed approach has been applied for critical attributes identification of a wind turbine. The monitoring data from ENGIE's 1st open data wind farm, namely La Haute Borne, in France were used in this application [34]. The 4 wind turbines in this farm have been providing electricity to the equivalent of 7300 people since 2009. In this case study, data of the wind turbine numbered R80711 throughout the year of 2017 (i.e., from January 1, 2017 to December 31, 2017) were selected. A total of 52,662 data points were recorded. The equipment model of R80711 was Senvion MM82, a popular wind turbine in the world for winds with high speeds. Fig. 4 shows the main modules of this wind turbine.

The original data set contained a total of 34 operating variables of wind turbines. A large raw dataset was created by the CM system from these variables. Normally, the data collected by a CM system are not in a format to be directly utilized. Data pre-processing is often conducted to reduce both data storage space and calculation time. The data for wind turbines were recorded for each variable at a certain time interval, and the average value of these data was calculated for a time period. The data provided by La Haute Borne were recorded every 10 min. The mean, maximum, minimum and standard deviation of each attribute were also calculated. For simplicity, only the mean values were selected as the input data in this study, and the number of dimensions of the data set was reduced to 25 (as listed in Table 1) based on physical relationships between the variables and the availability of the data. In addition, the redundant attributes, whose values were calculated from other attributes, were removed. In this work, *apparent power* (S) and *power factor* (*Cosphi*) were identified as redundant attributes, because *apparent power* (S) was calculated from *active power* (P) and *reactive power* (Q) by $S = \sqrt{P^2 + Q^2}$, and *power factor* (*Cosphi*) was calculated from *active power* and *apparent power* by $Cosphi = P/S$.

5.1. Operating conditions identification

5.1.1. Step 1: selection of attributes related to operating conditions

First the attributes related to operating conditions are selected. According to the discussions on contextual and behavioral attributes given in Section 3.1, the variables were classified into two categories as listed in Table 1. In this case study, since the contextual attributes were used to record changes in the operating environment, they were not directly related to anomaly. According to the rules explained in Section 3.1, five attributes were selected considering their relations with operating conditions. These five attributes included four contextual attributes (i.e., *absolute wind direction (Wa)*, *vane position (Va)*, *wind speed (Ws)* and *nacelle angle (Ya)*) and one behavioral attribute (i.e., *active power (P)*). Data pre-processing was conducted based on the physical relationships among the variables. After the examination of actual data, we observed that the *absolute wind direction* (*Wa*) was actually the sum of *vane position* (*Va*) and *nacelle angle* (*Ya*), i.e., $Wa = Va + Ya$ (as shown in Fig. 5). Therefore, *absolute wind direction* was not used for the clustering of operating conditions.

Table 1
List of variables recorded by CM system.

	Symbol	Name	Unit	Sensor location
Contextual attributes	*Wa*	Absolute wind direction	°	7
	Ot	Outdoor temperature	°C	7
	Va	Vane position	°	8
	Ws	Wind speed	m/s	7
	Ya	Nacelle angle	°	7
Behavioral attributes	*P*	Active power	kW	1
	Cm	Converter torque	Nm	1
	Gb1t	Gearbox bearing 1 temperature	°C	2
	Gb2t	Gearbox bearing 2 temperature	°C	2
	Git	Gearbox inlet temperature	°C	2
	Gost	Gearbox oil sump temperature	°C	2
	Db1t	Generator bearing 1 temperature	°C	2
	Db2t	Generator bearing 2 temperature	°C	2
	DCs	Generator converter speed	rpm	2
	Ds	Generator speed	rpm	2
	Dst	Generator stator temperature	°C	2
	Nf	Grid frequency	Hz	1
	Nu	Grid voltage	V	1
	Rt	Hub temperature	°C	4
	Yt	Nacelle temperature	°C	7
	Q	Reactive power	kW	1
	Ba	Pitch angle	°	8
	Rs	Rotor speed	rpm	4
	Rbt	Rotor bearing temperature	°C	4
	Rm	Torque	Nm	2
Redundant attributes	*S*	Apparent power	kW	1
	Cosphi	Power factor		

5.1.2. Step 2: clustering of operating conditions

According to the selected four attributes (*Va*, *Ya*, *Ws*, and *P*), the collected data were then clustered using GMM clustering algorithm provided in Section 3.2. The results with the greedy EM algorithm are provided in Table 2. The number of components was selected as $k = 3$. However, according to the technical specification sheet of Sevion MM82, a total of 4 operating conditions were considered:

(1) Start-up phase. At very low wind speeds, there is insufficient torque exerted by the wind on the turbine blades to make them rotate. However, as the speed increases, the wind turbine will begin to rotate and start to generate electrical power. The speed at which the turbine first starts to rotate is called the *cut-in speed*.
(2) Maximum wind energy tracking phase. As the wind speed rises

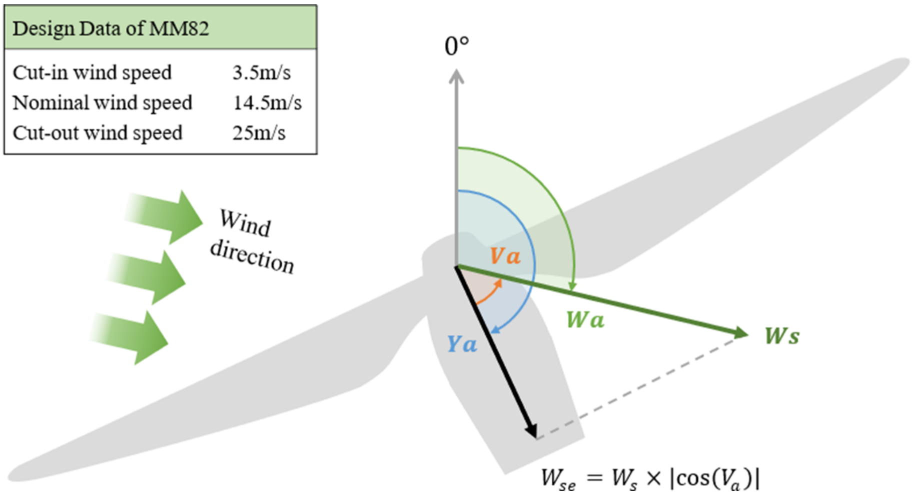


Fig. 5. Diagram for the calculation of effective wind speed.

above the cut-in speed, the power extracted by the turbine increases as the wind speed increases.

(3) Nominal power output phase. If the wind speed continues to rise, the power output will reach the limit that the electrical generator is capable of. This limit to the generator output is called the *nominal power output* and the wind speed at which it is reached is called the *nominal wind speed*. If the wind speed continues to rise, the power output will also increase, so the control system is required to keep the power constant at the design limit.

(4) Cut-out phase. As the speed increases above the *cut-out wind speed*, the forces on the turbine structure continue to rise and, at some point, there is a risk of damage to the rotor. As a result, a braking system is employed to bring the rotor to a standstill.

To validate the clustering results, the operating conditions of wind turbines were analyzed. In the technical specifications, the operating conditions were defined based on the effective wind speeds, which were the wind speeds perpendicular to the planes of blades. According to the relationships among variables shown in Fig. 5, the effective wind speed W_{se} can be calculated by:

$$W_{se} = W_s \times |\cos(V_a)| \tag{8}$$

Through the calculation with the entire data set, we found that only 30 recorded points from 51,000 effective data points had the maximum wind speeds over 25 m/s (i.e., the cut-out wind speed), meaning this operating condition could be ignored due to limited available data. Therefore, three operating conditions were identified from the data set with the optimal number of components (i.e., number of clusters in GMM) $\hat{k} = 3$. Then, each data point in the data set was assigned to the corresponding operating condition when it had the highest probability density to that component. The number of data points in operating conditions 1, 2, and 3 were 8617, 40464, and 2118, respectively.

5.2. Field data anomaly detection

To simplify the explanation, only the data in the second operating condition (i.e., cluster 2) were used to show the anomaly detection process. Anomaly detection using iForest was conducted in a two-stage process. In the training stage, a series of *i*Trees were built using sub-samples of the training set. In the evaluation stage, instances were passed through the formerly constructed isolation forest to obtain anomaly scores of these instances.

5.2.1. Step 1: construction of the iForest

Two parameters were selected for the iForest algorithm in the training stage. They were the size for sub-sampling ψ and the number of trees t. According to Liu et al. [30], sub-sampling with size of ψ selected as 2^8 or 256 was generally sufficient for anomaly detection across a wide range of data, and the path length was usually converged well before $t = 100$. Therefore, $\psi = 256$ and $t = 100$ were selected in this case study. At the end of the training process, a forest of 100 *i*Trees was obtained for the evaluation stage.

5.2.2. Step 2: identification of anomaly field data

In the evaluation stage, an anomaly score s was calculated from the average path length for each instance by Equation (5). In the iForest method, the default value of the evaluation height limit was assigned with the maximum value, i.e., $\psi - 1 = 255$, so that the anomaly score had the highest granularity. The anomaly scores of over 41,000 instances were calculated. Potential anomalies were identified as data points with large anomaly scores (i.e., $s > 0.7$). The number of anomaly data points under the second operating condition was 301.

5.3. Critical behavioral attributes identification

In Section 5.2, anomaly instances were identified based on their path depth measures. According to the method given in Section 4.2, the number of trees T was selected as $T = 128D = 2,560$ and the criticalness of each behavioral attribute was calculated by Equation (7). The results of the calculation are shown in Table 3. From this table, the behavioral attributes *Active power* (*P*), *Generator stator temperature* (*Dst*), *Generator speed* (*Ds*), and *Generator bearing 2 temperature* (*Db2t*) had obvious high criticalness values, indicating these attributes were highly correlated with product performance degradation due to the defects in

Table 2
Clustering results.

	Operating condition	μ
Cluster 1	Start-up phase	(5.53, 183.57, 2.02, 0.35)
Cluster 2	Maximum wind energy tracking phase	(0.03, 195.29, 6.53, 488.96)
Cluster 3	Nominal power output phase	(-0.74, 231.18, 16.26, 1796.99)

Table 3
Criticalness measures of behavioral attributes.

Behavioral attribute	Frequency	Criticalness
Active power	1263	**9.3951**
Generator stator temperature	1020	**7.5875**
Generator speed	999	**7.4313**
Generator bearing 2 temperature	874	**6.5014**
Generator converter speed	550	4.0913
Generator bearing 1 temperature	533	3.9648
Converter torque	475	3.5334
Torque	460	3.4218
Grid voltage	432	3.2135
Reactive power	348	2.5887
Gearbox oil sump temperature	325	2.4176
Gearbox bearing 1 temperature	283	2.1052
Gearbox bearing 2 temperature	231	1.7183
Grid frequency	208	1.0985
Rotor bearing temperature	186	0.9824
Gearbox inlet temperature	152	0.8028
Rotor speed	103	0.5440
Pitch angle	95	0.5017
Hub temperature	86	0.4478
Nacelle temperature	63	0.3280

the current design. These critical behavioral attributes were required to be investigated further by manufacturers to identify the causes of these anomalies for design improvement in the future.

5.4. Discussions

To test the validity and superiority of the proposed approach, several discussion and comparison have been conducted in this section.

5.4.1. Necessity of operating condition identification and data segmentation

When sufficient data, appropriate algorithms, and both the contextual attributes and the behavioral attributes are used for anomaly detection, the operating conditions (i.e., data patterns) can be achieved effectively. In these cases, the pre-process to select appropriate attributes related to operating conditions seems unnecessary. When the cases are not ideal, two problems were found in our validation with real data sets. (1) Some operating conditions contain relatively small numbers of data points. As a result, these operating conditions may not be recognized and the data from these operating conditions are considered as anomalies. (2) When both the contextual attributes and the behavioral attributes are used for clustering, normal instances could be wrongly identified as anomalies due to the differences in the values for the contextual attributes. To address the first problem, pre-process to select appropriate attributes related to operating conditions has to be conducted. In addition, both the contextual attributes and the behavioral attributes should be considered for selection of appropriate attributes. To address the second problem, only the behavioral attributes should be considered in the clustering of patterns. It was noted that when data from different operating conditions were mixed for anomaly detection, some anomalies in one operating condition were ignored due to their similar values as normal data for other operating conditions. In other words, a data instance might be considered anomaly for one operating condition, but normal in a different operating condition [26]. To demonstrate the influence of operating conditions clustering on anomaly detection, two small sets of CM data randomly selected from operating condition 2 and operating condition 3 respectively were tested. For ease of explanation, only two critical attributes isolated in Section 5.3, the *active power* and *generator stator temperature*, were selected for this discussion. In Fig. 6(b), the data point (1758.71,61.55) (shown in red) from operating condition 3 was detected as an anomaly. When data from the two operating conditions were not clustered as shown in Fig. 6(a), this data point was considered normal due to its similarity to the normal instances with similar values collected under operating condition 2. Therefore, to avoid errors in anomaly detection, clustering of multiple operating conditions should be conducted first.

5.4.2. Verification of the proposed approach

To demonstrate the effectiveness of the proposed method, the CM data collected for three other wind turbines of the same model (Senvion MM82) numbered R80721, R80736, and R80790 in the same wind farm are used. CM data of these wind turbines throughout the same year (i.e., from January 1, 2017 to December 31, 2017) were selected. Then, the anomaly instances were identified, and the criticalness of each behavioral attribute was calculated for each wind turbine. The results of the calculation are listed in Table 4. From Table 4, it is found that the identified behavioral attributes with higher criticalness values of the four wind turbines are almost the same, while the ranking orders of those attributes are slightly different. Our research aims to identify the attributes that contribute significantly to the abnormal statuses of the product/system, so that the existing designs can be modified to improve their reliabilities. Therefore, the most critical information is which attributes need to be concerned, and the subtle differences in the order between them are not that important. The main reason for some subtle differences may be due to the difference in the locations of the four wind turbines. Thus, it can be concluded that the proposed method can provide a reliable result of the critical behavioral attributes identification issue based on CM data.

5.4.3. Performance comparison of anomaly detection methods

Many methods have been developed in the past for anomaly detection, including the Local Outlier Factor (LOF) method, one-class SVM method, clustering-based methods, etc. When data with high dimensionality are considered, many of these conventional methods are not effective for anomaly detection. As summarized by Aggarwal [27] that the selection of a subspace seemed to be the only meaningful method for anomaly detection with high dimensionality. Two main challenges need to be addressed in anomaly detection with subspaces: (1) design of efficient algorithms for the exploration of subspaces, and (2) aggregation of the results achieved from the data in different subspaces [27]. As one of the methods with subspaces through a simple statistical test, such as Kurtosis, for selection of the subsamples, Isolation Forest provided good results for anomaly detection with high-dimensional data [32]. Isolation Forest was efficient in subspace exploration with reduced processing time [1], and an anomaly score was constructed based on the results from different subspaces. To demonstrate the performance of *i*Forest, an anomaly detection performance comparison of *i*Forest, ORCA [35], SVM and LOF was conducted with one-year data of the second operating condition and one-month data of the second operating condition, respectively. The results were shown in Table 5. It was observed that the anomalies obtained by *i*Forest, ORCA, and SVM were relatively similar to each other, while the LOF got poorer results when the amount of data is large. The results also demonstrated that iForest has a significantly advantage in processing time, especially when dealing with large datasets.

5.4.4. Influence of the number of the trees on quality for identification of critical attributes

The influence of the selected number of training trees on the stability of the results for critical attribute identification was studied in this research. In this investigation, the numbers of trees were selected as t = D, 2D, 4D, 8D, $\cdots$,1024D. Each time, an anomaly point from the results given in Section 5.2 was selected and passed to a tree in the forest to obtain the path length. The criticalness of an attribute (i.e., a particular dimension) d is calculated by:

$$c_d = \frac{\sum_{i=1}^{t} w_i f_d^i}{t} \tag{9}$$

where w_i is the weight of the path in the i-th tree, and f_d^i is the

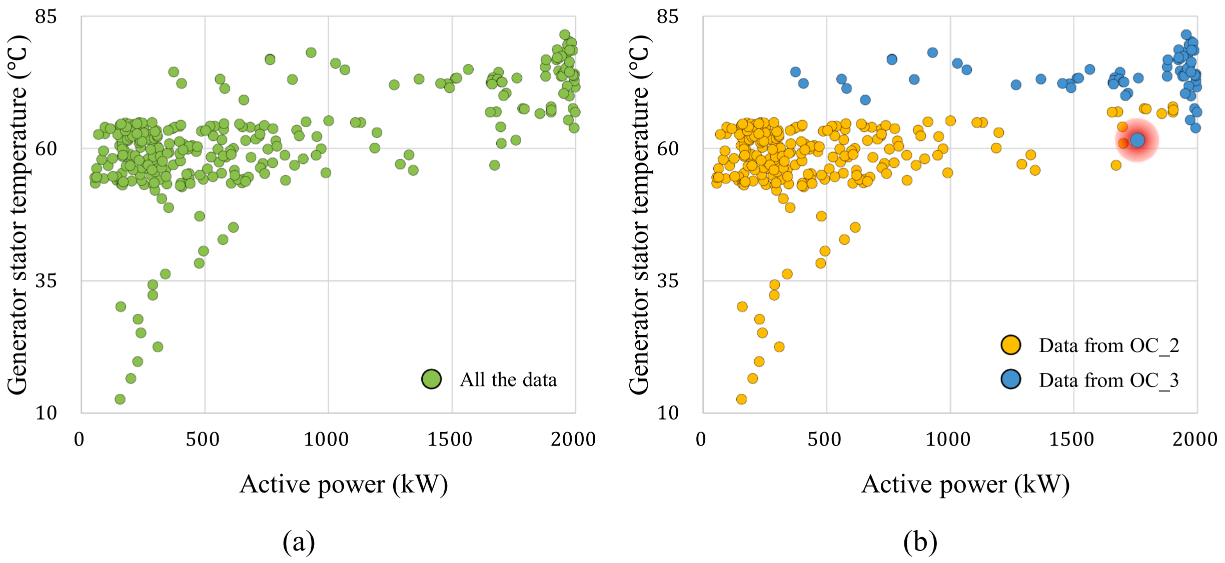


Fig. 6. Influence of clustering of operating conditions on anomaly detection: (a) Data without clustering of operating conditions; (b) Data with clustering of operating conditions.

Table 4
Criticalness calculation results for the wind turbines.

Rank order	R80711	R80721	R80736	R80790
1	*P*	*P*	*P*	*P*
2	*Dst*	*Ds*	*Dst*	*Dst*
3	*Ds*	*Dst*	*Ds*	*Db2t*
4	*Db2t*	*Db2t*	*Db2t*	*Ds*
5	*DCs*	*DCs*	*Db1t*	*DCs*
6	*Db1t*	*Cm*	*Cm*	*Db1t*
7	*Cm*	*Db1t*	*DCs*	*Rm*
…	…	…	…	…

Table 5
Performance comparison of *i*Forest, ORCA, SVM and LOF.

	Number of CM data instance	Number of anomalies	Overlap ratio with *i*Forest result	Processing time (seconds)
*i*Forest	40,464	301	/	7.34
ORCA		287	0.88	864.27
SVM		327	0.95	4197.40
LOF		138	0.35	130518.36
*i*Forest	3185	36	/	0.87
ORCA		33	0.94	4.13
SVM		45	0.89	5.74
LOF		26	0.39	101.57

frequency that the attribute *d* appears in the path of the *i*-th tree. For each attribute *d,* construction of tree forest was conducted *R* times, and the criticalness for the *r*-th test was denoted as c_{dr}. The performance stability of the critical attribute isolation method is defined as:

$$S = \max_{r=1,2,\cdots,R} c_{dr} - \min_{r=1,2,\cdots,R} c_{dr} \tag{10}$$

The smaller the value of *S* is, the better the stability of the method provides. The repeat time *R* was selected as 10. Using the dataset given in Section 5.3, the results were achieved as shown in Fig. 7. It was observed that high stability was obtained when the number of trees was at around $t = 128$D. Therefore, the default value of tree number should be selected by $t = 128$D.

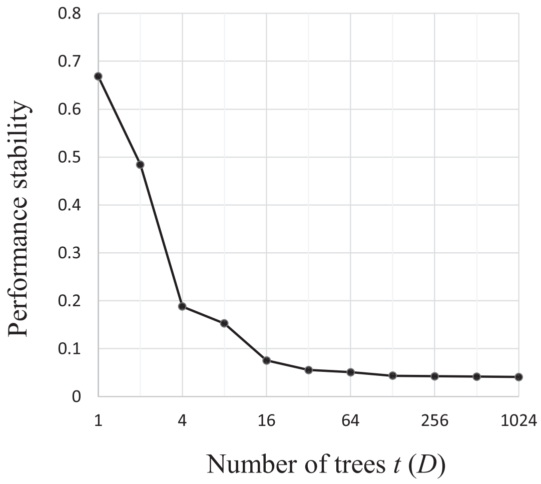


Fig. 7. The stability of the attribute isolation results under different tree numbers.

6. Conclusions

To improve the quality and efficiency of performance analysis and critical CM data attributes identification for products under multiple operating conditions, an integrated approach based on Isolation Forest is developed using collected high-dimensional condition monitoring data. The major contributions of this research are summarized as follows:

(1) To solve the problem that some anomalies might not be detected when the CM data are collected under various and different operating conditions, clustering of operating conditions is conducted such that anomalies are identified using the CM data for each of these operating conditions. First, the proper attributes related to operating conditions are identified from both the contextual attributes and the behavioral attributes. Rules are developed as the guidelines for attribute selection. Then, a GMM-based clustering method is used to identify different operating conditions and CM data in them.

(2) Most of the conventional anomaly detection methods are not very effective when the data sets are very high dimensional and have

only a small fraction of outliers. To improve the performance of anomaly detection of CM data with such characteristics, the Isolation Forest algorithm is employed after the data segmentation process. The comparison results showed that Isolation Forest can significantly reduce the processing time of anomaly detection while ensuring the accuracy, especially for problems with large datasets.

(3) A method is developed based on Isolation Forest to identify the critical behavioral attributes that lead to the anomalies. The attribute information contained in the split operations on the paths of anomaly field data traversing the isolation trees are utilized for the identification process. The influence of the number of trees on the stability of critical attribute identification results has also been investigated.

While the insights and results are encouraging, there remains potential for further studies and advances. First, integration of the model-based approach and the data-driven approach will be studied such that the advantages of these two approaches are employed. In addition, the relations between critical behavioral attributes and design parameters should be explored, such that the existing designs are modified to improve their reliabilities. Finally, the developed approach can be further improved by considering various data sources such as function failure data and product testing data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (Grant Nos. 51875345, 51475290).

References

[1] S.T. Kandukuri, A. Klausen, H.R. Karimi, K.G. Robbersmyr, A review of diagnostics and prognostics of low-speed machinery towards wind turbine farm-level health management, Renewable Sustainable Energy Rev. 53 (2016) 697–708.

[2] J. Liu, D. Djurdjanovic, K.A. Marko, J. Ni, A divide and conquer approach to anomaly detection, localization and diagnosis, Mech. Syst. Sig. Process. 23 (8) (2009) 2488–2499.

[3] C.J. Li, H. Lee, Gear fatigue crack prognosis using embedded model, gear dynamic model and fracture mechanics, Mech. Syst. Sig. Process. 19 (4) (2005) 836–846.

[4] X. Li, X. Jin, Z. Wen, D. Cui, W. Zhang, A new integrated model to predict wheel profile evolution due to wear, Wear 271 (1–2) (2011) 227–237.

[5] M. Pecht, R. Jaai, A prognostics and health management roadmap for information and electronics-rich systems, Microelectron. Reliab. 50 (3) (2010) 317–323.

[6] T. Brotherton, G. Jahns, J. Jacobs, D. Wroblewski, Prognosis of faults in gas turbine engines, in: 2000 IEEE Aerospace Conference, Big Sky, MT, USA, Mar. 25–25, 2000, pp. 163–171.

[7] M. Pecht, Prognostics and Health Management of Electronics, Encyclopedia of Structural Health Monitoring, John Wiley & Sons, New York, 2009.

[8] J. Yu, Bearing performance degradation assessment using locality preserving projections and Gaussian mixture models, Mech. Syst. Sig. Process. 25 (7) (2011) 2573–2588.

[9] H. Ma, X. Chu, G. Lyu, D. Xue, An integrated approach for design improvement based on analysis of time-dependent product usage data, ASME J. Mech. Des 139 (11) (2017) 111401.

[10] T. Liu, J. Chen, X. Zhou, W. Xiao, Bearing performance degradation assessment using linear discriminant analysis and coupled HMM, J. Phys.: Conf. Ser. 364 (2012) 012028.

[11] S. Sankararaman, S. Mahadevan, Bayesian methodology for diagnosis uncertainty quantification and health monitoring, Struct. Control Hlth. 20 (1) (2013) 88–106.

[12] J. Yan, M. Koc, J. Lee, A prognostic algorithm for machine performance assessment and its application, Prod. Plan. Control 15 (8) (2004) 796–801.

[13] W. Caesarendra, A. Widodo, B.-S. Yang, Application of relevance vector machine and logistic regression for machine degradation assessment, Mech. Syst. Sig. Process. 24 (4) (2010) 1161–1171.

[14] L. Mi, W. Tan, R. Chen, Multi-steps degradation process prediction for bearing based on improved back propagation neural network, J. Mech. Eng. Sci. 227 (7) (2013) 1544–1553.

[15] C.S. Byington, M. Watson, D. Edwards, Data-driven neural network methodology to remaining life predictions for aircraft actuator components, in: 2004 IEEE Aerospace Conference, Big Sky, MT, USA, Mar. 6–13, 2004, pp. 3581–3589.

[16] P. Baraldi, F. Di Maio, D. Genini, E. Zio, Comparison of data-driven reconstruction methods for fault detection, IEEE Trans. Reliab. 64 (3) (2015) 852–860.

[17] P. Tse, D. Atherton, Prediction of machine deterioration using vibration based fault trends and recurrent neural networks, J. Vib. Acoust. 121 (3) (1999) 355–362.

[18] P. Wang, G. Vachtsevanos, Fault prognostics using dynamic wavelet neural networks, AI EDAM 15 (4) (2001) 349–365.

[19] R. Huang, L. Xi, X. Li, C.R. Liu, H. Qiu, J. Lee, Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods, Mech. Syst. Sig. Process. 21 (1) (2007) 193–207.

[20] A. Bellas, C. Bouveyron, M. Cottrell, J. Lacaille, Anomaly Detection Based on Confidence Intervals Using SOM with An Application to Health Monitoring, Advances in Self-Organizing Maps and Learning Vector Quantization, Springer International Publishing, Switzerland, 2014, pp. 145–155.

[21] H.T. Pham, B.-S. Yang, T.T. Nguyen, Machine performance degradation assessment and remaining useful life prediction using proportional hazard model and support vector machine, Mech. Syst. Sig. Process. 32 (2012) 320–330.

[22] R. Kromanis, P. Kripakaran, Support vector regression for anomaly detection from measurement histories, Adv. Eng. Inf. 27 (4) (2013) 486–495.

[23] B. Wang, Y. Lei, N. Li, J. Lin, An improved fusion prognostics method for remaining useful life prediction of bearings, in: 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), Jun 19–21, Dallas, TX, USA, 2017, pp. 18–24.

[24] K. Javed, R. Gouriveau, N. Zerhouni, A new multivariate approach for prognostics based on extreme learning machine and fuzzy clustering, IEEE Trans. Cybern. 45 (12) (2015) 2626–2639.

[25] Y. Peng, M. Dong, M.J. Zuo, Current status of machine prognostics in condition-based maintenance: a review, Int. J. Adv. Manuf. Technol. 50 (1–4) (2010) 297–313.

[26] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey, ACM Comput. Surv. 41 (3) (2009) p. 15:1–58.

[27] C.C. Aggarwal, Outlier Analysis, Springer Publishing Company, New York, 2013.

[28] X. Song, M. Wu, C. Jermaine, S. Ranka, Conditional anomaly detection, IEEE Trans. Knowl. Data Eng. 19 (5) (2007) 631–645.

[29] G. McLachlan, D. Peel, Finite Mixture Models, Willey Series in Probability Aad Statistics, John Wiley & Sons, New York, 2000.

[30] C. Biernacki, G. Celeux, G. Govaert, Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models, Comput. Stat. Data Anal. 41 (3–4) (2003) 561–575.

[31] N. Vlassis, A. Likas, A greedy EM algorithm for Gaussian mixture learning, Neural Process. Lett. 15 (1) (2002) 77–87.

[32] F.T. Liu, K.M. Ting, Z.-H. Zhou, Isolation-based anomaly detection, ACM Trans. Knowl. Discovery Data 6 (1) (2012) 3.

[33] B.R. Preiss, Data Structures and Algorithms, John Wiley & Sons, New York, 1999.

[34] ENGIE Group, La Haute Borne Data (2017–2020), Retrieved from: https://opendata-renewables.engie.com/explore/dataset/la-haute-borne-data-2017-2020/table/ (accessed in Oct 2018), 2017.

[35] S.D. Bay, M. Schwabacher, Mining distance-based outliers in near linear time with randomization and a simple pruning rule, in: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Aug 24–27, Washington, D.C., USA, 2003, pp. 29–38.