Research Article

Performance Assessment for a Fleet of Machines Using a Combined Method of Ant-Based Clustering and CMAC

Lei Zhang, Qixin Cao, and Jay Lee

1 Research Institute of Robotics, Shanghai Jiao Tong University, Shanghai 200240, China
2 NSF Center for Intelligent Maintenance Systems, University of Cincinnati, Cincinnati, OH 45221, USA

Correspondence should be addressed to Lei Zhang; zhanglei@sjtu.edu.cn

Received 2 April 2013; Revised 15 July 2013; Accepted 17 July 2013

Academic Editor: Liang Gong

Copyright © 2013 Lei Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a combined method of ant-based clustering and cerebellar model articulation controller for performance assessment for a fleet of machines. A novel ant-based clustering algorithm with kernel method is used to cluster machines in a fleet. The algorithm has two features. First, a projection based on kernel principal component analysis replaces random projection to improve the efficiency. Second, the clustering is performed on the feature space after kernel mapping to improve the clustering accuracy. The algorithm can cluster machines in a self-organizing way to achieve the horizontal assessment. The vertical assessment for the single machine based on CMAC is presented. Then, how to combine the vertical and horizontal assessment results is discussed. The outlier mining method to detect abnormal machines based on the clustering results is also proposed. Cluster-based global outlying factor is suggested to measure the outlying degree of abnormal machine. Finally, the case study on axial fans shows that the combined method can give a more comprehensive assessment for fans’ performance monitoring.

1. Introduction

Intelligent Maintenance (IM) of machine is emerging as a replacement of traditional reactive maintenance style. Different from traditional “fail and fix (FAF)” maintenance practice, IM focuses on “predict and prevent (PAP)” methodology to achieve near-zero downtime performance. Usually, the machine and components go through a series of degradation states before failure. The predictive tools are needed to monitor degradation states rather than detect the faults [1]. Once the degradation has been detected, the remaining useful life of machine could be predicted. Therefore, the key challenges of implementing IM are machine degradation assessment and prediction.

Many efforts have been made to develop methods and tools for these challenges. Yan and Lee [2] proposed a logic regression method to assess performance degradation of an elevator door system. They [3] also presented the further study combining logic regression and fuzzy logic for nozzle life prediction in gas turbine. Logistic regression can easily represent the daily maintenance records as a dichotomous problem. However, this method is applicable only when both normal and failure behavior are available. Yu et al. [4] presented a machine performance assessment approach based on Gaussian Mixture Model (GMM). Experimental results of real industrial run-to-failure bearing had shown that the model was efficient. Gaussian Mixture Model integrated with locality preserving projection was also suggested to assess the bearing performance degradation [5]. Mixture of Gaussian can be utilized to approximate an arbitrary distribution, while the number of the mixtures is not easy to be determined and the optimization parameters can be affected by different initialization methods. Lee [6] first proposed a pattern discrimination model based on the Cerebellar Model Articulation Controller (CMAC). Experiments on the stepping motor and the robot had proven the feasibility of the model. Lin and Wang [7] also used enhanced CMAC in performance analysis of rotating machinery. Zhang et al. [8] suggested a modified CMAC algorithm for performance degradation assessment of self-maintenance machine. Xu et al. [9] proposed a fuzzy based extension of CMAC to analyze two types of machine degradation severities. In one case, the network was trained by signals from different levels of machine degradation states. In the other case, the
network was only trained by signals of normal state. The generalization ability of CMAC makes it very suitable for performance degradation assessment. It can be trained only using the data in the normal state. Wu et al. [10] proposed an online adaptive condition-based maintenance method with pattern discovery and fault learning capabilities. The method was mainly based on Self-Organizing Map (SOM). An experiment on the machine tool test bed validated the proposed approach. SOM provides a way of representing multidimensional feature space in a one- or two-dimensional space while preserving the topological properties of the input space. It is an unsupervised learning and no prior outputs are needed. Qiu et al. [11] proposed an adaptive wavelet filter method for hazard rate prediction of bearing. Further, they developed a robust performance degradation assessment method for rolling bearing [12]. This method was based on a combination of wavelet filter and SOM for fault identification and assessing performance degradation, respectively. Pan et al. [13] suggested a Fuzzy C-means (FCM) method for bearing performance degradation assessment. Lifting wavelet packet decomposition was used to create feature vectors.

Due to the complexity of the machine, the combination of multiple methods is usually used for identifying working condition. Lapira et al. [14] proposed a systematic framework that utilized multiregime modeling approach to trend and assess wind turbine degradation. Three methods composed by SOM, GMM, and neural network were combined and chosen to handle condition data with multiple operating regimes. Caesarendra et al. [15] proposed the combination between Relevance Vector Machine (RVM) and Logistic Regression (LR) for bearing performance assessment. Liao and Lee [16] presented a novel machine performance degradation scheme based on Fixed Cycle Features Test (FCFT). FCFT introduced a new testing method which obtained data during the transient periods of different working loads. Wavelet packet analysis, PCA, and mixture model were combined. A case study for chiller system was used as an example. A comparative study of maintenance data classification based on neural networks, logistic regression, and support vector machines was also provided [17].

Remaining Useful Life (RUL) prediction methods are important to extrapolate the machine’s process behavior over time and predict its life in the future. Autoregressive Moving Average (ARMA) model is usually used for modeling and predicting future values in time series. Yu et al. [18] presented Elman Recurrent Neural Network (ERNN) for predicting the behavior of a boring process during its full life-cycle. This prediction was achieved by the fusion of the predictions of three principal components extracted from the spindle load. Liu et al. [19] proposed a match matrix method for manufacturing process performance prediction and diagnosis. By constructing a match matrix, the similarity between two feature series can be represented by the best match index. Siegel et al. [20] suggested a robust regression curve fitting approach for Remaining Useful Life prediction for the helicopter oil-cooler bearings. Tran et al. [21] presented a three-stage method for both performance degradation assessment and RUL prediction. In the first stage, ARMA model was generated by using only normal operating data to identify the behavior of the complex system. In the second stage, the Cox proportional hazard model was established to estimate the system survival function. In the last stage, SVM for regression [22], with multistep ahead prediction ability, was utilized to forecast the RUL. Huang et al. [23] suggested a new scheme for the prediction of a ball bearing’s RUL based on self-organizing map (SOM) and back propagation neural network. Gebrael et al. [24] also suggested a neural network approach for RUL predictions of bearing from vibration-based degradation signals. Two classes of neural network models, single-bearing model, and clustered-bearing model were developed.

A toolbox named as Watchdog Agent [25] has been developed, which integrated many algorithms mentioned above. The toolbox has been embedded in the Labview software for public usage.

Reviewing all the assessment and predicting methods above, we can see that most methods use time-domain comparison, which means machine’s current condition is compared with its historical data to assess various degradation states. All these methods are used for single machine or system. This kind of methods can be concluded as vertical assessment and prediction. But in IM, there is also a need for peer to peer comparison for a fleet of machines. So-called “a fleet of machines” means that many machines, with the same or similar structures and working condition, compose a fleet. For example, many pumps in a factory or several elevators in a building can be taken as a fleet of machines. The running information of a fleet of machines can be exchanged or collected easily for analysis. Machines in the fleet can be compared to each other to assess their performance degradation. We call this comparison as horizontal assessment and prediction. Table 1 shows the features of vertical and horizontal assessment methods. The horizontal comparison of a fleet of machines presents the chance and innovation brought by Internet to the maintenance style. But now the relevant research and report on this field are few.

For this purpose, this paper proposed a horizontal method based on ant-based clustering. This kind of ant-based clustering algorithm imitates the ant’s behavior of clustering their corpses [26]. The object with \( n \) attributes can

<table>
<thead>
<tr>
<th>Machine type</th>
<th>Vertical assessment and prediction</th>
<th>Horizontal assessment and prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compared data</td>
<td>Compared with historical data of itself</td>
<td>Compared with the data of the same (similar) machines</td>
</tr>
<tr>
<td>Analyzing domain</td>
<td>Time domain</td>
<td>Space domain (peer to peer)</td>
</tr>
<tr>
<td>Methods</td>
<td>CMAC, SOM, GMM, etc.</td>
<td>Cluster-based method</td>
</tr>
</tbody>
</table>

Table 1: The features of vertical and horizontal assessment and prediction.
be looked as a point in $n$-dimensional space. The objects are projected into a low dimension space (often a two-dimensional plane). Then the ants compute the similarity of the objects and decide to pick up or drop them. Compared with traditional clustering methods, such as partition-based methods, hierarchy-based methods, and density-based methods, ant-based clustering algorithm does not need any prior knowledge. This is an important merit which makes it suitable in our application. Because we usually do not know how many clusters there are in a fleet of machines, it should be created by a self-organizing way. Second, ant-based clustering process is visible on the projecting plane. This is very helpful in the monitoring of machine performance. The visible clustering results can be directly integrated into the interface of the monitoring software.

In our application, on one hand, the efficiency of the algorithms should be improved because some maintenance decisions or actions are dependent on the results of clustering analysis. On the other hand, the algorithm should have the ability to cluster the datasets with different structures. We have proposed a novel ant-based clustering algorithm using the kernel method to overcome these problems [27]. In this paper, we will show how this algorithm is applied in performance assessment for a fleet of machines.

The remainder of this paper is organized as follows. Section 2 described the system framework of proposed methods. Section 3 proposed the methods for performance assessment for a fleet of machines, including horizontal assessment, vertical assessment, and the fusion method. Section 4 gave the case study and analysis. Finally, Section 5 gave the conclusion and further research directions.

2. The System Framework of Proposed Methods

In this paper, we will propose a combined method of ant-based clustering and CMAC for performance assessment for a fleet of machines. As shown in Figure 1, data acquirers can get the data of fans in monitoring nodes. The data can be collected and transferred by router to the database server by the Ethernet. The following analysis can be performed.

(i) Machines in a fleet are clustered using ant-based clustering method.

(ii) Performance assessment for the single machine is conducted based on CMAC.

(iii) The horizontal and vertical assessments are combined to get the final conclusion.

(iv) The outliers are detected based on outlying factor in a very small cluster.

3. The Assessment Method for a Fleet of Machines

3.1. Ant-Based Clustering Algorithm with the Kernel Method for Horizontal Assessment

3.1.1. The Basic Ant-Based Clustering Algorithm. The algorithm introduced by Lumer and Faieta [28] represents the basic ant-based clustering method. Some important concepts are firstly introduced through Figure 2.

The projection plane: the objects and ants are initially projected onto a two-dimensional plane. Each object or ant is projected randomly. The size of the plane can be determined based on the number of objects.

The local neighbourhood of object $o_i$: it is a neighbouring region of the object $o_i$ and written as $\text{Neigh}(o_i)$. It is often a square with size $s \times s$ ($s = 2r + 1$), where $r$ is the radius of $\text{Neigh}(o_i)$. The center of $\text{Neigh}(o_i)$ is the position of $o_i$.

The local similarity: it is the similarity of the object $o_i$ with other objects in $\text{Neigh}(o_i)$. It is often measured by the distance
between objects. In Figure 2, the object $o_i$ is assumed to locate at the coordinate $(x_i, y_j)$. The local similarity of $o_i$ is given by

$$f(o_i) = \begin{cases} \frac{1}{\alpha} \sum_{o_j \in \text{Neigh}(o_i)} [1 - \frac{d(o_i, o_j)}{\alpha}], & \text{when } f > 0 \\ 0, & \text{otherwise} \end{cases}$$

(1)

where $d(o_i, o_j)$ is the distance between two objects. Typically Euclidean distance is used. $\alpha$ is a factor that defines the scale for dissimilarity.

The probability conversion function: it is a function that converts the local similarity of $o_i$ into the probability of being picked up (or dropped) by ants. The probability that an ant will pick up or drop the object is

$$P_p(o_i) = \left( \frac{k_1}{k_1 + f(o_i)} \right)^2,$$

(2)

$$P_d(o_i) = \begin{cases} 2f(o_i) & \text{when } f(o_i) < k_2, \\ 1 & \text{when } f(o_i) \geq k_2, \end{cases}$$

(3)

where $k_1$, $k_2$ are two constants. $k_1$ and $k_2$ adjust the probabilities of picking up and dropping objects. $P_p(o_i)$ and $P_d(o_i)$ are compared with a random real number $p$ ($p \in [0, 1]$), and the results determine whether the object $o_i$ should be picked up or dropped.

The process of LF algorithm can be generalized as the following steps.

1. Projection: all objects and ants are randomly projected onto the grid.
2. Calculating the similarity: each ant calculates the object’s similarity to others in the object’s local neighborhood.
3. The similarity is transformed to the probability to pick up or drop object.
4. Ants pick up or drop objects.
5. Ants move.
6. Repeat (2)–(5).

A number of modifications have been introduced to the basic LF algorithm to improve clustering quality and convergent speed. A latest review of ant-based clustering can be referred to [29].

### 3.1.2. Ant-Based Clustering Algorithm with the Kernel Method.

To improve the algorithm's efficiency and clustering quality, we incorporated the kernel method into ant-based clustering and created the novel ant-based clustering with the kernel method (ACK) [27]. The applications of kernels in ACK are shown in two ways. First, Kernel Principal Component Analysis (KPCA) is used to modify the random projection of all objects. Second, the Euclidean distance in the feature space is applied as a measure of the similarity between the objects.

**Projection of the Objects Based on KPCA.** In ant-based clustering algorithms, the objects are randomly projected onto the plane. One pattern corresponds randomly with a pair of coordinates. This random projection leads to few similarities between the objects in the local neighbourhood at the early stage. It takes a long time for an object to be similar to nearby objects from the inception of the algorithm. We have suggested a modified projection based on Principal Component Analysis (PCA) [30]. Then we applied KPCA to replace PCA. Compared with linear PCA, KPCA can extract features that are more useful for classification.

**The Ant Movement Model Operating in the Feature Space.** In ACK algorithm, we applied the ant movement model suggested by Xu et al. [31], where each object is taken as an ant. Each ant has two states: movement and sleep. If the ant finds that a location is suitable for it to rest, it will stop moving and enter a sleeping state; otherwise, it will continue to move to another place. The fitness of the local neighbourhood is computed as (4), which is similar to (1) in Section 3.1.1. Consider

$$f(o_i) = \begin{cases} \frac{1}{s^2} \sum_{o_j \in \text{Neigh}(o_i)} \left[ 1 - \frac{d(o_i, o_j)}{\alpha_i} \right], & \text{when } f > 0 \\ 0, & \text{otherwise} \end{cases}$$

(4)

where $\alpha_i = (1/(N - 1)) \sum_{j=1}^{N} d(o_i, o_j)$ is the average distance between $o_i$ and other objects.

If we apply a kernel function to map the objects into the feature space, then the clustering can be performed according to the similarities of the objects in the feature space. The fitness becomes

$$f(o_i) = \begin{cases} \frac{1}{s^2} \sum_{o_j \in \text{Neigh}(o_i)} \left[ 1 - \frac{d_F(o_i, o_j)}{\alpha_i} \right], & \text{when } f > 0 \\ 0, & \text{otherwise} \end{cases}$$

(5)

where $\alpha_i = (1/(N - 1)) \sum_{j=1}^{N} d_F(o_i, o_j)$ and $d_F(o_i, o_j)$ is the distance between $o_i$ and $o_j$ in the feature space. $d_F(o_i, o_j)$ can be obtained by the kernel function according to

$$d_F(o_i, o_j) = \sqrt{K(o_i, o_j) - 2K(o_i, o_j) + K(o_j, o_j)}.$$

(6)

A Gaussian kernel is applied in this study as follows:

$$K(x, y) = \exp \left( -\frac{||x - y||^2}{2\sigma^2} \right).$$

(7)
0 /* initialization */
1 All objects are placed on the grid based on KPCA
2 Initialize all parameters: $r_1, t_{max}, \alpha, \beta$
3 /* main loop */
4 for $t = 1$ to $t_{max}$ do
5 for $i = 1$ to $N$ do ( $N$ is the number of the objects)
6 compute $f(o_i)$ and $P_a(o_i)$
7 draw a random real number $p \in (0,1)$
8 if ($p \leq P_a(o_i)$) then
9 activate ant and move to next place
10 else
11 stay at current site and sleep
12 endif
13 endfor
14 adjust $\alpha, \beta$
15 endfor
16 Output locations of all objects;

Pseudocode 1: The pseudocode of the ACK algorithm.

For an ant, the probability of being activated by the local neighbourhood is

$$
P_a(o_i) = \left(\frac{\beta}{\beta + f(o_i)}\right)^2,
$$

(8)

where $\beta \in R^+$ is the threshold of the ant’s active fitness. Equation (8) is also similar to (2) in Section 3.1.1. When $f < \beta$, $P_a(o_i)$ is close to 1. Thus, if the fitness of $o_i$ is much smaller than the threshold, $o_i$ has a high probability of being activated. The active $o_i$ moves in the plane and searches for a more comfortable place to sleep. When $f > \beta$, $P_a(o_i)$ is close to 0. Therefore, $o_i$ does not wake up and continues to sleep. The principle of the ant movement model is the same as that in the LF algorithm [28]. The big difference is that the passive movement of objects is transformed into active movement.

The pseudocode of the main body of the ACK algorithm is given in Pseudocode 1.

3.2. CMAC for Vertical Assessment and Comparison. CMAC neural network has been employed to evaluate machine performance degradation states [7, 9]. The main advantages of CMAC against other neural networks are its local generalization, extremely fast learning speed, and easy implementation in software and hardware.

As shown in Figure 3, CMAC can be considered as an associative memory network, which performs two mappings:

$$
S \rightarrow A \rightarrow P,
$$

(9)

where $S$ is the $m$-dimensional input space. The input variable $S_i = [s_{i1}, s_{i2}, \ldots, s_{in}]$ ($i = 1, 2, \ldots, N$). $A$ is an $n$-dimensional memory cell vector. Every input variable $S_i$ in $S$ only activates $g$ elements in $A$. $g$ is called as generalization parameter ($g = 3$ in Figure 3). $P$ is one-dimensional output space. The weights in the activated memory cells are added to create the output.

Before the mapping, each dimension $s^k_i$ ($k = 1, 2, \ldots, m$) in input variable $S_i$ should be quantized as

$$
\hat{s}^k_i = \frac{s^k_i - s^\text{min}_k}{r_k},
$$

(10)

where $s^\text{min}_k$ is the minimum of the $k$th dimension of $S_i$; $r_k$ is the quantization parameter of the $k$th dimension of $S_i$. The mapping address of $s^k_i$ ($k = 1, 2, \ldots, m$) will be determined according to its quantization value. Then, the first mapping combines all $m$-dimensional mapping address and projects the point $S_i$ in the input space into a binary associative vector $A_i$. The elements in $A_i$ are defined as

$$
a_{ij} = \begin{cases} 
1 & \text{if the } j\text{th element is activated by the } k\text{th sample,} \\
0 & \text{otherwise,}
\end{cases}
$$

(11)

where $1 \leq j \leq n$. 

![Figure 3: The structure of CMAC.](image-url)
Based on Clustering Results. Cluster analysis can create some outliers as side products. As for the performance analysis of a fleet of machines, the outliers may represent the machines in fault states. Therefore, outlier detection is very important to find abnormal machines. He et al. [32] presented a definition of cluster-based local outlier factor (CLOF) for identifying the physical significance of an outlier. The definition can be briefly described as follows.

Let the data set \( D \) be a set of points. After \( D \) is analyzed by a clustering algorithm, the clustering result is described as \( C = \{C_1, C_2, \ldots, C_k\} \), where

\[
C_i \cap C_j = \varnothing, \quad C_i \cup C_j \cup \cdots \cup C_k = D,
\]

\( 1 \leq i, j \leq k, \ i \neq j \).

\( k \) is the number of the clusters. They then defined Large Cluster (LC) and Small Cluster (SC) based on two parameters [32]. For the limit of the space, we will not describe in detail. Suppose that \( C = \{C_1, C_2, \ldots, C_k\} \) is the set of clusters with the sequence \(|C_1| \geq |C_2| \geq \cdots \geq |C_k|\), where \(|C_i| (i = 1, 2, \ldots, k)\) is the number of the points in cluster \( C_i \). As for each data point \( t \), the CLOF of \( t \) is defined as

\[
\text{CLOF} = \min \left( \frac{\text{dis}(t, C_j)}{|C_j|}, \right.
\]

where \( \text{dis}(t, C_i) \) is the distance between \( t \) and \( C_i \), which can be calculated by any distance formula in the clustering algorithm.

CLOF gives the way to measure the degree of the data point being an outlier. CLOF can indicate the degree of the abnormal machine departing from its nearest large cluster. But this nearest large cluster may represent machines with degradation performance. In fact, people are more interested in the degree of the outlier departing from machines with normal performance. A factor is needed to measure not only local but also global departing degree of outliers. So we suggest a definition of cluster-based global outlier factor (CGOF). Consider

\[
\text{CGOF} = \min \left( \frac{\text{dis}(t, C_j)}{|C_j|} \right) * \overline{\text{CV}}(C_j),
\]

where \( \overline{\text{CV}}(C_j) \) is the mean confidence value of all objects in the cluster \( C_j \), which can be obtained by any vertical
We then used the ACK algorithm in a dataset which came from the monitoring data of the axial fans in one water cooling plant. There are 32 fans which are distributed in different places of the plant. The data of the fans are collected by the field bus to the server. These fans are of the same structure and work in the similar condition. They can compare each other to assess their performance. Three attributes to describe the fan’s performance are gotten. They are the bearing temperature, the motor current, and the vibration of the spindle.

4. Case Study and Analysis

4.1. The Horizontal Assessment Based on ACK. We design two algorithms to compare with ACK. One applies random projection, and the other applies PCA projection. The clustering parts of two algorithms are the same as those in ACK. Figure 5 shows the comparison of three algorithms on the initial projections and the results after 600 cycles. Random projection needs a long time to create rough clusters. Some small rough clusters can be found after about 600 cycles. Compared with random projection, PCA and KPCA projection can create rough clusters at the initial stage. KPCA projection can create more separate and compressed clusters compared with PCA projection. These rough clusters provide bases to create larger clusters around them, so the clustering time will be saved significantly.

Table 2 shows the comparison of clustering time. All the algorithms are run 10 times to get the average clustering time. The saved time is computed based on the time of random projection. Both PCA and KPCA projections can save time because of the creations of rough clusters at initial stage. KPCA can save more time compared with PCA. Compared with random projection, KPCA can save almost a half part of time.

The parameters of ACK in this case are listed as follows $t_{max} = 2000$, $r = 2$, $\beta = 0.1$, $\sigma = 0.75$. Figure 6 shows the final clustering results. The fans are clustered to three different groups. The features of each group are listed in Table 3.

To compare and validate the clustering results, we use other three methods, $K$-means, $KK$-means (Kernel $K$-means), and SOM, to cluster the data. As for $K$-means and $KK$-means, we set the number of cluster centers as three. SOM does not need this prior knowledge. Actually, SOM gets the number of the clusters as four. But there are only two fans in the fourth cluster. For certain points, these four methods get disputable results, which are listed in Table 4. We can see that ACK gets the similar results with $KK$-means. The reason is that these two algorithms cluster the data in the feature space after kernel mapping. Except for fans number 3 and number 10, the clustering results of ACK are consistent with the final possible conclusions.

Through above analysis, we can see that the performance assessment of the axial fans can be achieved by ant-based clustering algorithm in a self-organizing way. The number of the clusters is not needed as a prior knowledge here. In the present condition monitoring system, a lot of data created by a fleet of machines, like fans and motors, are not used sufficiently. The common style to process these data is showing them in the display screen, or giving the diagrams of certain important parameters. The comparison among machines in a fleet has not been performed.

Figure 7(a) shows the radar chart of the temperature feature of axial fans. Around the circumference is the number of fans, and the corresponding radial value is its temperature. The advantage of radar chart is that it can compare one single feature for machines clearly. Some special points, such as the minimum and maximum points, can be easily detected for each feature. But it cannot show the comparison of the general performance including all the features. Figure 7(b) shows the combination of ant-based clustering assessing results with the radar chart of the temperature. Different signs and colors are used to identify the different clusters. So we can easily recognize which group the fan belongs to and pay more attention to its difference with its similar ones. But this chart still cannot show the comprehensive performance of each machine. We will show the final assessing results of radar chart in Section 4.3 where the clustering results are fused with the vertical assessing results of the single machine.

4.2. The Vertical Assessment Based on CMAC. If the historical data of one fan for a long time can be gotten, CMAC can be used to assess the fan’s performance degradation during its running. In this case, we can get the full monitoring data of fan number 7. Figure 8 shows the averaged features every day. The initial running states of 46 days are taken as the normal states. The Confidence Values (CVs) at these days are 1. Then the features and CVs are inputted to CMAC to train it. The training parameters in this case are $r = [1, 10, 1]$, $\beta = 1.0$, and $Max_{cycle} = 1000$. After CMAC is trained, the features at the operational condition are inputted to CMAC to get the CVs outputs. Figure 8 shows the CV outputs from CMAC. The CVs at different time can show the degree of performance degradation based on the normal state.

As for the single machine’s performance assessment, the best way is that the machine’s historical data can be collected. But the historical data may be not rich enough for every machine in a fleet because of all kinds of reasons. In our case, if only the data of fan number 7 is full and collected for a long time, can we assess the other fans’ performance based...
Figure 5: The comparison of three projection ways and clustering results after 600 cycles.
Table 3: The features of three clusters (L: low; M: medium; H: high).

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Number of fans</th>
<th>Members</th>
<th>Current (A)</th>
<th>Temperature (°C)</th>
<th>Vibration (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>8</td>
<td>1, 4, 5, 11, 16, 17, 18, 19</td>
<td>200.10 (L)</td>
<td>44.80 (L)</td>
<td>6.24 (L)</td>
</tr>
<tr>
<td>C2</td>
<td>10</td>
<td>2, 3, 6, 7, 8, 20, 21, 27, 28, 29</td>
<td>222.59 (H)</td>
<td>50.76 (H)</td>
<td>7.45 (M)</td>
</tr>
<tr>
<td>C3</td>
<td>14</td>
<td>9, 10, 12, 13, 14, 15, 22, 23, 24, 25, 26, 30, 31, 32</td>
<td>208.95 (M)</td>
<td>48.25 (M)</td>
<td>9.09 (H)</td>
</tr>
</tbody>
</table>

Figure 6: The final clustering results of ACK.

Table 4: The points with disputable clustering results in four methods.

<table>
<thead>
<tr>
<th>Fan no.</th>
<th>ACK</th>
<th>K-means</th>
<th>KK-means</th>
<th>SOM</th>
<th>Final conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2 or 3</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3 or 1</td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>27</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>31</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>32</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
</tbody>
</table>

on these data? In the next section, we will discuss how to fuse horizontal and vertical assessments to answer this question.

4.3. The Combination of Horizontal and Vertical Assessment Results. In Section 4.1, horizontal comparison of all fans is conducted using ant-based clustering algorithm. The used data are the average features at the 240th day (which is shown as the horizontal assessing time in Figure 8). In this section, we will discuss how to combine horizontal and vertical assessment results in two possible conditions. The first is only that fan number 7 has a full historical data. The second is that all fans' historical data are collected sufficiently.

4.3.1. The Fans Have no Enough Historical Data except Fan number 7. The conclusion of vertical assessment is that fan number 7 has a confidence value 0.804. The conclusion of horizontal assessment is that fan number 2 and number 7 are clustered in the same group. Then the simplest combined conclusion is the confidence value of fan number 2 is around 0.804. If we can have another fan's full historical data, for example, fan number 6, then the confidence value of fans number 2 can be gotten by averaging the confidence values of fan number 7, and fan number 6 (fan number 2, fan number 7 and fan number 6 are clustered in the same group). By this combination, although fan number 2 has no historical data, we can also assess its performance based on its same machine in a fleet.

Moreover, we can set up another CMAC model. Use the normal features of fan number 7 to train it (the output CV is 1); then the features of other fans at the horizontal assessing time are inputted to CMAC to get the corresponding CV outputs. Figure 9 shows a radar chart of all fans' CVs in the horizontal assessment time in Figure 8. Different clusters are fitted in different circles. The figure can give a clear display of the comprehensive performance comparison of fans. The confidence values smaller than 0.5 should be paid more attention because these machines are more likely to degrade to a failure state. Although this assessment is only based on one machine's normal data, it is still a good way to assess a fleet of machines especially in the case that the historical data for most machines are not sufficient.

4.3.2. All Fans’ Historical Data Are Collected Sufficiently. If all fans’ data can be collected sufficiently, the results of horizontal and vertical assessments can be used to test whether the assessing conclusion is consistent. For example, the conclusions of vertical assessment are that fan number 7 has a confidence value 0.804 and fan number 2 has a confidence value 0.798. The conclusion of horizontal assessment is that fans number 2 and number 7 are clustered in the same group. Then the results of horizontal and vertical assessments are consistent. If fan number 2 has a confidence value 0.398 (degraded seriously) while fans number 2 and number 7 are still clustered in the same group in horizontal assessments, the results of horizontal and vertical assessment are contradictory. Then further analysis should be conducted to make sure which conclusion is correct. For example, other clustering methods or vertical assessing methods should be used to analyze data again.

4.4. Discussion on Outlier Mining Based on Assessing Results. We use an example to show how to detect the outliers by ant-based clustering. During the operation of the fans, an artificial interference is added to three fans to make their vibration higher than normal operation. The clustering results of the fans are shown in Figure 10.

Three outliers are obviously clustered to a single cluster C4. C4 is a small cluster. Based on the analysis of Section
3.1.3, we get that the local neighborhood cluster of C4 is C3. Actually, based on the assessment results, the cluster C3 is a group with degradation state. The average CV of C3 is 0.677. Then the outlier factors can be computed using (15) and (16) in Section 3.4. The outlier factors are listed in Table 5. We can see that the factors can quantitatively represent the outlying degree of each outlier. The outliers should be paid more attention because they may mean that the abnormal behavior has happened.

5. Conclusions

A combined method of ant-based clustering and CMAC is suggested to assess the performance of machines in a fleet. Machines in a fleet can be compared to each other because they have the same structures and work in the same condition. Ant-based clustering algorithm does not need any prior knowledge. It is proper to cluster machines in a self-organizing way. The application of kernel in ant-based clustering can improve the efficiency and clustering accuracy, which is achieved by modifying projection based on KPCA and clustering in feature space. The paper also described vertical assessment method based on CMAC for the single machine. The confidence value of the machine can indicate its degradation degree compared with historical normal data.
The suggested method was finally used for assessing the performance of axial fans. The results show that the method performs well. The method gives performance assessment for the machines in time domain and space domain. Even if the historical data of one machine is not available, the performance analysis can be conducted based on its similar ones in the fleet.

The following items are needed to be further studied:

(i) the adaptive adjusting method of the parameters in ACK algorithm,
(ii) the applications of the method in more datasets,
(iii) the deep fusion of vertical and horizontal assessing results.

Acknowledgments

This paper is supported by the National Natural Science Foundation of China (no. 50705054) and the Key National Project of ITER China (no. 2011GB113005).

References


