

Received August 30, 2020, accepted September 3, 2020, date of publication September 8, 2020, date of current version September 23, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3022771

# Fault Detection for Aircraft Turbofan Engine Using a Modified Moving Window KPCA

HAO SUN<sup>1</sup>, YINGQING GUO<sup>1</sup>, AND WANLI ZHAO<sup>1</sup>

School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, China

Corresponding author: Hao Sun (sunhao2013@mail.nwpu.edu.cn)

**ABSTRACT** As a typical data-driven fault detection approach, the moving window kernel principal component analysis (MWKPCA) method has attracted attention for fault detection of turbofan engines considering the presence of component degradation, but the conventional MWKPCA method uses a fixed step to update the KPCA model periodically until anomaly data is detected, this will increase the amount of calculation. To address this weakness, a modified MWKPCA method is proposed based on an adaptive updating mechanism for the KPCA model in this study. To realize the capability of updating KPCA model adaptively, k-means clustering method is utilized to divide a certain amount of newly acquired sampling data and the same amount of oldest data in the current time window into two categories, and calculates the Mahalanobis distance of the two clustering centers. Then the distance is compared with a prescribed threshold to determine whether to update the KPCA model. The proposed method is applied to an illustrative case, and the fault detection results under normal condition and sensor faults condition show that compared with the conventional MWKPCA algorithm, the modified MWKPCA method does not weaken the ability of fault detection and has better performances in terms of computation efficiency than the conventional MWKPCA algorithm with a fixed moving step.

**INDEX TERMS** KPCA, MWKPCA, k-means clustering, aircraft turbofan engine, fault detection.

## I. INTRODUCTION

The aircraft turbofan engine, an extremely complex aerothermodynamics system operating under harsh environments, various faults will be inevitable happened in a long-time service process [1], [2]. These faults may lead to poor aircraft engine performance or even affect the flight safety. Online and real-time engine fault detection can help improve the safety and reliability by providing accurately and effectively anomaly monitoring.

In recent years, aircraft turbofan engine fault detection has been widely studied, and the main technologies can be categorized into two distinct groups [3], [4], including the model-based approaches [5]–[7] and data-driven approaches [8]–[11]. Model-based approaches provide most solutions to real-time monitoring and control problems [12]. But these approaches require a reasonably high-fidelity mathematical model of turbofan engine, which is unfortunately rarely available. This fact has propelled extensive research activities into data-driven-based fault detection. Many data-driven methods

are based on statistical and feature extraction methods [13], neural networks and machine learning [14]–[16], and fuzzy logic [17]–[19]. In these data-driven solutions, multivariate statistical process monitoring (MSPM) has been successfully applied to online process quality monitoring. And with the continuous improvement of aircraft engine data acquisition technology, the dimension and quantity of data acquisition are greatly increased. MSPM can handle high-dimensional, noisy, and highly correlated data and realize data dimensionality reduction and abnormal data detection. MSPM methods such as principal component analysis (PCA) [20]–[22], modified principal component analysis (MPCA) [23], partial least squares (PLS) [24], and independent component analysis (ICA) [25], [26] have been developed and applied for fault detection. Among MSPM methods, PCA is the most widely used technique, which relates to its conceptual simplicity. The commonly used fault detection indices in PCA method are the Hotelling  $T^2$  and the Squared Prediction Error (SPE) [27]. However, PCA is a method of linear projection and performs poorly in dealing with nonlinear processes, like aircraft turbofan engine fault detection. To overcome this limitation,

The associate editor coordinating the review of this manuscript and approving it for publication was Noor Zaman<sup>1</sup>.

the kernel principal component analysis (KPCA) [28] have been developed.

KPCA first projects the input data space onto a feature space by means of nonlinear mapping and then computes the principal components via eigen decomposing the kernel matrix in the feature space. The two main advantage of KPCA are that: (1) It can efficiently extract the principal features using integral operators and nonlinear kernel functions and without involving any complex nonlinear optimization. (2) It can handle a wide range of nonlinearities by incorporating different kernel functions. KPCA has been successfully applied for nonlinear system fault detection. Similar as PCA, the Hotelling  $T^2$  and SPE are also used in fault detection process based on KPCA method.

There are two main challenges of KPCA-based fault detection, firstly, as the training data size increases, the calculation of feature decomposition of the symmetric kernel matrix will be time consuming [29]. Secondly, the KPCA model remains time invariant once it is established based on the training data [28], [29], but most real industrial processes are time varying, and a fixed KPCA model cannot reflect the latest measuring information of system parameters, and false alarms may occur in fault detection process. To address the first challenge, Fazai et al. [29] and Jaffel et al. [30] proposed a reduced kernel principal component analysis (RKPCA) method, and k-means clustering was applied to reduce the number of training samples. To overcome the second limitation, several extensions for KPCA have been proposed as moving window kernel principal component analysis (MWKPCA) [30], [31], variable moving window kernel principal component analysis (VMWKPCA) [32] and adaptive kernel principal component analysis (AKPCA) [33]. The MWKPCA method adds the latest normal sampling data to the current time window and excludes the oldest observation from the current time window. For most industrial processes monitoring, like aircraft turbofan engine, in most cases, they are operated in the stable state. During this period, the measured output parameters contain a lot of redundant information, so it is necessary to study whether the KPCA model needs to be updated periodically. However, the conventional MWKPCA method lacks an effective mechanism for KPCA model updating and usually uses a fixed step to continuously update the KPCA model until anomaly data is detected. Therefore, fault detection based on the conventional MWKPCA method will increase the amount of calculation.

This article combines k-means clustering with conventional MWKPCA, and proposes a modified MWKPCA used for aircraft engine fault detection. K-means clustering method divides a certain amount of newly acquired sampling data and the same amount of oldest data in the time window into two categories, and calculates the Mahalanobis distance of the two clustering centers. Then the distance is compared with a prescribed threshold. If the Mahalanobis distance of the two clustering centers does not exceed the threshold, it means that the newly acquired data does not bring more new information, so it is not necessary to update the KPCA

model, otherwise the KPCA model need to be updated. The modified MWKPCA method, on the basis of not affecting the original MWKPCA method fault detection performance, will greatly reduce the calculation amount of the algorithm.

The remainder of this article is organized as follows: In Section. II, the fault detection method based on KPCA and conventional MWKPCA method are introduced. The modified MWKPCA method and the key technologies are introduced in Section. III. In Section. IV, aiming at the normal operating condition and sensor faults condition of an aircraft engine, the conventional MWKPCA and the modified MWKPCA algorithm are simulated and verified, and the fault detection performance and calculation amount of the two algorithms are compared. The conclusion is summarized in Section. V.

## II. PREVIOUS WORKS

Before introducing the modified MWKPCA algorithm, we first review the fault detection methods based on KPCA and MWKPCA, respectively.

### A. FAULT DETECTION BASED ON KPCA

KPCA maps sample data ( $X \in \mathbb{R}^{m \times n}$ , where  $m$  denotes the number of samples and  $n$  denotes the dimensions of the original space) from its original space into a higher dimensional feature space ( $\Gamma \in \mathbb{R}^{h \times h}$ ,  $h \gg n$ ) using a nonlinear mapping function and the principal components are obtained in  $\Gamma$ .

$X \in \mathbb{R}^{m \times n}$  is extended into  $\Gamma$  by using nonlinear function  $\phi(x_i)$  ( $i = 1, \dots, m$ ). The covariance matrix  $C \in \mathbb{R}^{h \times h}$  in  $\Gamma$  is defined as

$$C = \frac{1}{m-1} \sum_{i=1}^m \phi(x_i)\phi^T(x_i) \quad (1)$$

It is assumed that  $\phi(x_i) \in \mathbb{R}^h$  is the sample in  $\Gamma$  with zero-mean and unit-variance. The eigenvalue decomposition of  $C$  can be solved by

$$\begin{aligned} \lambda_j v_j &= C v_j \\ &= \frac{1}{m} \sum_{i=1}^m \langle \phi(x_i), v_j \rangle \phi(x_i) \end{aligned} \quad (2)$$

where  $v_j \in \mathbb{R}^{h \times 1}$  ( $j = 1, \dots, h$ ) and  $\lambda_j \in \mathbb{R}^{1 \times 1}$  are the  $j$ th eigenvector and eigenvalue, respectively. In space  $\Gamma$ ,  $v_j$  can be linearly expanded by coefficients  $\alpha_i^j \in \mathbb{R}^{1 \times 1}$  ( $i = 1, \dots, m$ ) and  $\phi(x_i)$ , such that

$$v_j = \sum_{i=1}^m \alpha_i^j \phi(x_i) \quad (3)$$

Combining equation (2) and (3), we can obtain

$$\begin{aligned} \lambda_j \sum_{k=1}^m \alpha_k^j \langle \phi(x_i), \phi(x_k) \rangle \\ = \frac{1}{m} \sum_{k=1}^m \alpha_k^j \left\langle \phi(x_i), \sum_{q=1}^m \phi(x_q) \langle \phi(x_q), \phi(x_k) \rangle \right\rangle \\ j = 1, \dots, h; \quad i = 1, \dots, m \end{aligned} \quad (4)$$

To avoid obtaining  $\phi(\mathbf{x})$  and doing eigenvalue decomposing of  $\mathbf{C}$  directly, the kernel matrix  $\mathbf{K} \in \mathbb{R}^{m \times m}$  [28] is given by

$$[\mathbf{K}]_{i,j} = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle = k(\mathbf{x}_i, \mathbf{x}_j) \quad (5)$$

Define a vector  $\boldsymbol{\alpha}^j \in \mathbb{R}^{m \times 1}$  as

$$\boldsymbol{\alpha}^j = [\alpha_1^j, \alpha_2^j, \dots, \alpha_m^j]^T \quad (6)$$

Combining equation (5) and (6) into (4), we can get

$$m\lambda_j \boldsymbol{\alpha}^j = \mathbf{K} \boldsymbol{\alpha}^j, \quad j = 1, \dots, m \quad (7)$$

Kernel function should satisfy Mercers theorem [33], and polynomial, sigmoid, and radial basis kernels are the widely used kernel functions. Before doing eigenvalue decomposition in equation (7), the kernel matrix  $\mathbf{K}$  should be conventionalized and centralized to  $\bar{\mathbf{K}}$ :

$$\bar{\mathbf{K}} = \mathbf{K} - \mathbf{1}_m \mathbf{K} - \mathbf{K} \mathbf{1}_m + \mathbf{1}_m \mathbf{K} \mathbf{1}_m \quad (8)$$

where  $\mathbf{1}_m$  is a square matrix of order  $m$ , the elements are all  $1/m$ . The equation (7) can be expressed as

$$m\lambda_j \boldsymbol{\alpha}^j = \bar{\mathbf{K}} \boldsymbol{\alpha}^j; \quad j = 1, \dots, m \quad (9)$$

Solve the equation (9), then the orthonormal eigenvectors ( $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \dots, \boldsymbol{\alpha}^m$ ) and the corresponding eigenvalues ( $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$ ) can be obtained. The number of principal components  $d$  can be chosen by Cumulative Percent Variance (CPV) [34] and it is given by

$$\frac{\sum_{j=1}^d \lambda_j}{\sum_{j=1}^m \lambda_j} \geq th_{CPV} \quad (10)$$

$th_{CPV}$  is a user-defined threshold. For all  $p = 1, \dots, d$ , the  $p$ th score values  $t_p(\mathbf{x})$  of the new measurement  $\mathbf{x} \in \mathbb{R}^{1 \times n}$  can be obtained by projecting  $\phi(\mathbf{x})$  onto the direction of  $\mathbf{v}_p$  in space  $\Gamma$ , as shown in the following formula

$$\begin{aligned} t_p(\mathbf{x}) &= \langle \mathbf{v}_p, \phi(\mathbf{x}) \rangle = \sum_{i=1}^m \alpha_p^i \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle \\ &= \sum_{i=1}^m \alpha_p^i k(\mathbf{x}, \mathbf{x}_i) \end{aligned} \quad (11)$$

$\alpha_p^i$  denotes the  $i$ th element of  $\alpha_p$ .

The two statistics Hotelling  $T^2$  and SPE are usually used to detect faults in nonlinear system, and they are defined as [35]

$$\begin{cases} T^2 = [t_1, t_2, \dots, t_d] \boldsymbol{\Lambda}^{-1} [t_1, t_2, \dots, t_d]^T \\ \text{SPE} = \sum_{i=1}^m t_i^2 - \sum_{i=1}^d t_i^2 \end{cases} \quad (12)$$

where  $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_d)$ ,  $T_\beta^2$  and  $\text{SPE}_\beta$  denote the confidence limit of statistics Hotelling  $T^2$  and SPE with the confidence level  $\beta$ , respectively.  $T_\beta^2$  and  $\text{SPE}_\beta$  can be calculated as [28], [36]

$$\begin{cases} T_\beta^2 = [d(m-1)/(m-d)] F_{d,m-d,\beta} \\ \text{SPE}_\beta = [b/(2a)] \chi_\beta^2(2a^2/b) \end{cases} \quad (13)$$

where  $F_{d,m-d,\beta}$  denotes the critical value of  $F$ -distribution with  $(d, m-d)$  degrees of freedom and the confidence level  $\beta$ . The  $a$  and  $b$  are the estimated mean and variance of the SPE determined from training data.  $\chi_\beta^2(2a^2/b)$  means the  $\chi^2$ -distribution with  $2a^2/b$  degrees of freedom and the confidence level  $\beta$ .

The fault detection process based on KPCA are divided into two phases: building a KPCA model under the normal operating condition by off-line training and online real-time monitoring. The details of the phases are described in Table. 1.

### B. FAULT DETECTION BASED ON CONVENTIONAL MWKPCA

The conventional MWKPCA algorithm is applied by moving a fixed length data window in real time to update the KPCA model when a certain number of new normal samples are obtained. And the new normal samples are added to the current data window and to keep the length of the data window constant, the same number of oldest samples are removed. We assumed  $L$  denotes the window length, and one move step is  $h$ .

The key to updating the KPCA model is to update the centralized kernel matrix  $\bar{\mathbf{K}}$ . The update process includes two stages: (1) Adding  $h$  new data samples  $\{\mathbf{x}_{L+1}, \mathbf{x}_{L+2}, \dots, \mathbf{x}_{L+h}\}$  to the current data window  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L\}$ . (2) Discarding  $h$  oldest data samples  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_h\}$  from the new data window  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{L+h}\}$ . The kernel matrix  $\bar{\mathbf{K}}_{L+h} \in \mathbb{R}^{(L+h) \times (L+h)}$  after adding  $h$  data samples can be expressed as shown in equation (14), where the upper left corner is the kernel matrix in the original model  $\bar{\mathbf{K}}_L \in \mathbb{R}^{L \times L}$ :

$$\bar{\mathbf{K}}_{L+h} = \begin{bmatrix} \bar{\mathbf{K}}_L & \mathbf{A} \\ \mathbf{A}^T & \mathbf{B} \end{bmatrix} \quad (14)$$

where

$$\begin{cases} \bar{\mathbf{K}}_L = \begin{bmatrix} \bar{k}_{11} & \dots & \bar{k}_{1L} \\ \vdots & \ddots & \vdots \\ \bar{k}_{L1} & \dots & \bar{k}_{LL} \end{bmatrix} \in \mathbb{R}^{L \times L} \\ \mathbf{A} = \begin{bmatrix} \bar{k}_{1(L+1)} & \dots & \bar{k}_{1(L+h)} \\ \vdots & \ddots & \vdots \\ \bar{k}_{L(L+1)} & \dots & \bar{k}_{L(L+h)} \end{bmatrix} \in \mathbb{R}^{L \times h} \\ \mathbf{B} = \begin{bmatrix} \bar{k}_{(L+1)(L+1)} & \dots & \bar{k}_{(L+1)(L+h)} \\ \vdots & \ddots & \vdots \\ \bar{k}_{(L+h)(L+1)} & \dots & \bar{k}_{(L+h)(L+h)} \end{bmatrix} \in \mathbb{R}^{h \times h} \end{cases} \quad (15)$$

where  $\bar{k}_{11} = \bar{k}(\mathbf{x}_1, \mathbf{x}_1)$ , the other elements in equation (15) can be deduced like that. After removing the first  $h$  rows and the first  $h$  columns in equation (14), and the updated kernel

matrix  $\bar{K}'_L$  can be obtained as equation (16):

$$\bar{K}'_L = \begin{bmatrix} \bar{k}_{(h+1)(h+1)} & \cdots & \bar{k}_{(h+1)(L+h)} \\ \vdots & \ddots & \vdots \\ \bar{k}_{(L+h)(h+1)} & \cdots & \bar{k}_{(L+h)(L+h)} \end{bmatrix} \quad (16)$$

TABLE 1. Fault detection algorithm based on KPCA.

<b>offline training:</b>
1. Obtain a normal condition data set with $m$ samples and $n$ variables ( $\mathbf{X} \in \mathbb{R}^{m \times n}$ ), and obtain $\bar{\mathbf{X}} \in \mathbb{R}^{m \times n}$ through data normalization, set kernel parameters.
2. Compute the kernel matrix ( $\mathbf{K} \in \mathbb{R}^{m \times m}$ ), and obtain $\bar{\mathbf{K}}$ through kernel matrix centered.
3. Calculate the eigenvalues and eigenvectors of $\bar{\mathbf{K}}$ , and determine the number of principals.
4. Determine the control limit $T_{lim}^2$ and $SPE_{lim}$ .
<b>Online fault detection:</b>
5. Obtain a new test data set $\mathbf{x}_k \in \mathbb{R}^{1 \times n}$ . Normalize $\mathbf{x}_k$ with the same estimated mean and conventional deviation from the normal operating condition modeling step.
6. Calculate the projection of $\mathbf{x}_k$ in principal component direction by using equation (11).
7. Calculate the two statistics $T_k^2$ and $SPE_k$ by using equation (12).
8. Decide whether the $T_k^2$ or the $SPE_k$ exceeds the corresponding control limit $T_{lim}^2$ and $SPE_{lim}$ .
9. Return to step 5.

After updating the kernel matrix, calculate the principal eigenvalues and eigenvectors of  $\bar{K}'_L$ . The control limits of  $T_{lim}^2$  and  $SPE_{lim}$  also need to recalculate according to the new data window  $\{\mathbf{x}_{h+1}, \mathbf{x}_{h+2}, \dots, \mathbf{x}_{L+h}\}$ . And the next new continuous data samples are monitored using the updated KPCA model. The implementation of this algorithm requires the establishment of an initial KPCA model by training normal data in offline. The KPCA offline training phase is shown in Table. 1. And the details of the online fault detection phase based on the conventional MWKPCA algorithm is described in Table. 2.

### III. THE PROPOSED MODIFIED MWKPCA METHOD

The conventional MWKPCA method would update the KPCA model when the newly acquired  $h$  samples are normal. It is not necessary to update the KPCA model, if the  $h$  oldest samples to be discarded are relatively stable compared to the newly acquired  $h$  samples. Due to the lack of an adaptive KPCA model updating mechanism, the conventional MWKPCA method updates KPCA models too frequently. Therefore, a new KPCA model updating mechanism for MWKPCA algorithm needs to be considered.

Suppose the data set  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_h, \dots, \mathbf{x}_L\}$  denotes the current data window. If the KPCA model is updated, the data set in the next time window is  $\{\mathbf{x}_{h+1}, \mathbf{x}_{h+2}, \dots, \mathbf{x}_L, \dots, \mathbf{x}_{L+h}\}$ . To decide whether to update the KPCA model established based on data set  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_h, \dots, \mathbf{x}_L\}$ , the similarity of the two data sets  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_h\}$  and  $\{\mathbf{x}_L, \mathbf{x}_{L+1}, \dots, \mathbf{x}_{L+h}\}$  needs to be compared. Given the number of clusters, k-means clustering method can maximize the distance between clusters and minimize the distance

TABLE 2. Online fault detection algorithm based on the conventional MWKPCA.

#### Online fault detection:

1. Set the length of data window  $L$ , moving forward step  $h$  and kernel parameters. Obtain a normal condition data set  $\mathbf{X} \in \mathbb{R}^{L \times n}$ , and get  $\bar{\mathbf{X}} \in \mathbb{R}^{L \times n}$  through data normalization, set initial cumulative number  $i = 0$ .
2. Compute the kernel matrix ( $\mathbf{K} \in \mathbb{R}^{L \times L}$ ), and obtain  $\bar{\mathbf{K}}$  through kernel matrix centered.
3. Calculate the eigenvalues and eigenvectors of  $\bar{\mathbf{K}}$ , and determine the number of principal components.
4. Determine the control limits  $T_{lim}^2$  and  $SPE_{lim}$ .
5. Obtain a new test data sample  $\mathbf{x}_k \in \mathbb{R}^{1 \times n}$  and scale it.
6. Calculate the projection of  $\mathbf{x}_k$  in principal component direction by using equation (11).
7. Calculate the two statistics  $T_k^2$  and  $SPE_k$  by using equation (12).
8. If the  $T_k^2$  and the  $SPE_k$  do not exceed the corresponding control limit  $T_{lim}^2$  and  $SPE_{lim}$ ,  $\mathbf{x}_k$  is considered a normal data and perform an accumulation operation ( $i = i + 1$ ). If the  $T_k^2$  or the  $SPE_k$  exceeds the control limit, the cumulative number  $i$  remains unchanged.
9. If the data collected for  $h$  consecutive times is normal sample ( $i = h$ ), update the current data window and return to step1. If the cumulative number  $i$  is less than  $h$ , then return to step 5.

between internal data points. Therefore, we choose k-means clustering method to measure the similarity between the two data sets. Divide the two data sets into two categories using k-means clustering method, and obtain the two cluster center points  $(\mathbf{x}_{c1}, \mathbf{x}_{c2})$ . Because the Mahalanobis distance [35] between two points is independent of the measurement unit of the original data, we calculate the Mahalanobis distance of the two cluster center points to measure the similarity between the two categories, as shown in equation (17).

$$d_{Ma}(\mathbf{x}_{c1}, \mathbf{x}_{c2}) = \sqrt{(\mathbf{x}_{c1} - \mathbf{x}_{c2})^T \boldsymbol{\omega}^{-1} (\mathbf{x}_{c1} - \mathbf{x}_{c2})} \quad (17)$$

In equation (17),  $\boldsymbol{\omega}^{-1}$  represents the inverse matrix of the diagonal matrix formed by the conventional deviation of the measurement parameters.

$d_{threshold}$  is a presetting threshold for Mahalanobis distance between the two cluster center points  $\mathbf{x}_{c1}$  and  $\mathbf{x}_{c2}$ . If  $d_{Ma}(\mathbf{x}_{c1}, \mathbf{x}_{c2})$  does not exceeds the threshold  $d_{threshold}$ , which means data set  $\{\mathbf{x}_L, \mathbf{x}_{L+1}, \dots, \mathbf{x}_{L+h}\}$  is similar to data set  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_h\}$ , therefore the current data window and the corresponding KPCA model do not need to be updated. There are two problems need to be solved before implementing the new KPCA model updating mechanism to MWKPCA algorithm: firstly, how to perform effective recursive calculation on  $(\mathbf{x}_{c1}, \mathbf{x}_{c2})$ , secondly, how to preset the threshold  $d_{threshold}$ . The following two parts will introduce the solutions to these two problems in detail.

#### A. K-MEANS CLUSTERING METHOD

K-means is a typical clustering algorithm, which is simple and possesses rapid convergence rate. K-means method partitions the input dataset into k clusters and find out k clustering centers to minimize the distance between each sample point and its nearest clustering center [37]. The general calculation steps of k-means clustering method are as follows:

(1) Determining  $K$  initial cluster centers randomly. (2) For each data sample to be classified, find the nearest cluster center, and divide it into the data set. (3) Recalculate the clustering center. (4) Calculate the sum of the minimum squared distance between each data point and its nearest cluster center  $D_c$ . (5) If the  $D_c$  converges, terminate the clustering process, otherwise return step (2).

Assume that the data set in the current data window is  $\{x_{i+1}, x_{i+2}, \dots, x_{i+h}\}$ , after obtaining  $h$  normal samples, k-means clustering is used to divide dataset  $\{x_{i+1}, x_{i+2}, \dots, x_{i+h}\}$  and dataset  $\{x_{i+L+1}, x_{i+L+2}, \dots, x_{i+L+h}\}$  into two categories, and the distance between two clustering centers is calculated. To reduce the amount of calculation in the calculation process of k-means clustering centers, we consider a recursive method as shown in Fig. 1.

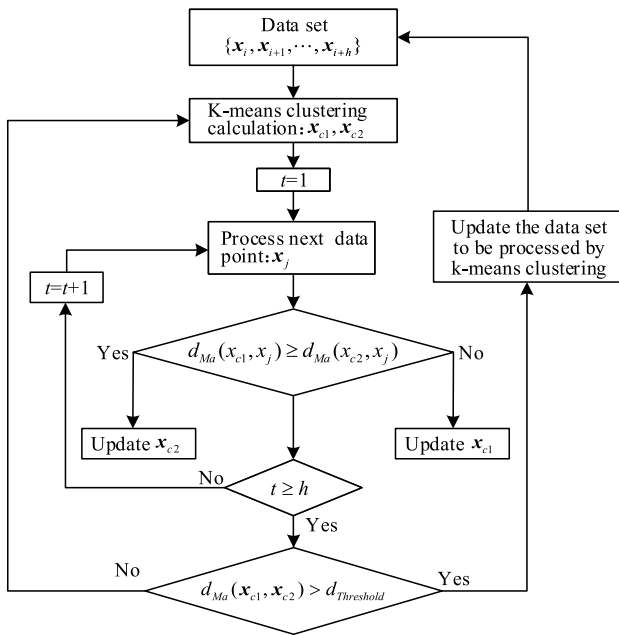


FIGURE 1. The flowchart of k-means clustering calculation.

Firstly, two clustering centers of data set  $\{x_{i+1}, x_{i+2}, \dots, x_{i+h}\}$  are calculated using the k-means clustering method as the initial values of the recursive calculation. Then the Mahalanobis distances between the next sample data  $x_j$  and each clustering center are calculated using the equation (17). Divide  $x_j$  into the data category with the smaller Mahalanobis distance, and the clustering center of the data category is updated using the equation (18):

$$x'_{ci} = \frac{nx_{ci} + x_j}{n + 1} \quad (18)$$

In equation (18),  $x'_{ci}(i = 1, 2)$  denotes the updated clustering center,  $n$  denotes the number of samples in this data category.

After completing the classification of the  $h$  data samples and the recursive calculation of the cluster points, then it is determined whether the Mahalanobis distance between

the two clustering centers calculated using equation (17) exceeds the presetting threshold  $d_{threshold}$ . If the Mahalanobis distance exceeds  $d_{threshold}$ , the KPCA model and clustering data set  $\{x_i, x_{i+1}, \dots, x_{i+h}\}$  need to be updated. And the two k-means cluster centers of the updated data set need to be calculated as the new initial values of the iterative calculation process. Otherwise, if the Mahalanobis distance does not exceed  $d_{threshold}$ , then continue to process new data samples based on the current cluster centers of the data set  $\{x_i, x_{i+1}, \dots, x_{i+h}\}$ .

### B. DETERMINE THE PRESETTING THRESHOLD

The threshold  $d_{threshold}$  is used to determine whether to update the time window data and the corresponding KPCA model. The setting of  $d_{threshold}$  should comprehensively consider the impact of engine component performance degradation on sensor measurement parameters and the fault detection performance of the improved MWKPCA algorithm, such as false alarm rate, fault detection rate, fault detection time and other performance indicators. It is assumed that the same baseline model can be used for fault detection if the relative change of the performance of engine components is less than 0.5%. The idea of obtaining the appropriate setting of  $d_{threshold}$  is: referring to the degradation law of the turbofan engine, taking 0.5% as performance degradation change step of engine components, calculate the Mahalanobis distance of the output parameters in each adjacent degradation situation and a data set of the Mahalanobis distance is formed, and the minimum value of the Mahalanobis distance is chosen as the  $d_{threshold}$ .

In the actual operation of aircraft engines, the performance of their mechanical rotating components will degrade slowly, in order to simulate the normal degradation process, the performance degradation of HPT component is considered. The HPT component operates in a high-temperature and high-pressure environment. Compared with other mechanical rotating components, the performance degradation rate of HPT components is faster. The HPT efficiency coefficient  $e_{HPT}$  and flow coefficient  $f_{HPT}$  are used to measure the health condition of HPT component. With reference to relevant literature [38], the relationship between the performance degradation level  $F$  of HPT and the changes of  $e_{HPT}$  and  $f_{HPT}$  is as follows:

$$\Delta e_{HPT} = \frac{F}{\sqrt{1 + (\Delta f_{HPT} / \Delta e_{HPT})^2}} \quad (19)$$

$$\Delta f_{HPT} = \Delta e_{HPT} * (\Delta f_{HPT} / \Delta e_{HPT}) \quad (20)$$

where  $\Delta f_{HPT} / \Delta e_{HPT} \in [-1, -0.5]$ , the relationship between the Mahalanobis distance of the output parameters in each adjacent degradation situation and the performance degradation level  $F$  of high-pressure turbine components is simulated as shown in Fig. 2.

Combining the simulation results and the  $d_{threshold}$  setting idea, the minimum value of the data set  $\{d'_{Ma}\}$  is set as the

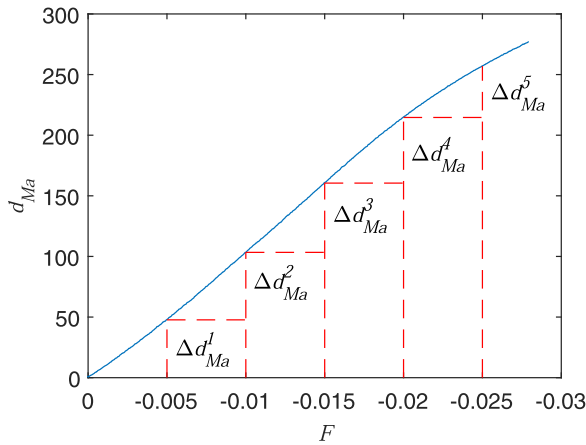


FIGURE 2. The simulation of the relationship between  $F$  and the Mahalanobis distance of the output parameters in each adjacent degradation situation.

$d_{threshold}$ , that is:

$$d_{threshold} = \min\{d_{Ma}^i\}, \quad i = 1, \dots, 5 \quad (21)$$

**C. OVERALL STRUCTURE AND ALGORITHM IMPLEMENTATION OF THE MODIFIED MWKPCA ALGORITHM**

The flowchart of the modified MWKPCA algorithm is shown in Fig. 3. Firstly, in the offline training phase, the initial KPCA model is trained using the normal samples and the corresponding control limit  $T_{lim}^2$  and  $SPE_{lim}$  are calculated. Like all other data-driven fault detection approaches, a normal data set is required for the initial training stage of the proposed method. In the real application of the algorithm, the initial training data can be extracted from the aircraft engine ground test data sets. Then in the online real-time fault detection phase, the real time test data is monitored by the trained KPCA model and the corresponding statistics  $T^2$  and  $SPE$  of the test data are calculated and compare with the control limits. If one of the statistics exceeds the corresponding control limit, the test data is considered as abnormal data, and the parameter for counting  $i$  needs to recount. If  $i$  accumulate to  $h$ , whether to update the KPCA model is determined by applying k-means clustering method, which help to calculate the Mahalanobis distance between the two cluster centers  $x_{c1}$  and  $x_{c2}$ . If the Mahalanobis distance does not exceed the threshold limit, continue to use the current KPCA model for online fault detection, otherwise, the current KPCA model and the corresponding control limits need to be updated using the latest obtained normal data.

The specific algorithm implementation steps of the modified MWKPCA fault detection method are shown in Table. 3.

**IV. COMPARATIVE STUDY**

In this section, the conventional MWKPCA algorithm and the modified MWKPCA algorithm are applied to the fault detection of a nonlinear model of a commercial aircraft turbofan engine. The fault detection performances (false alarm

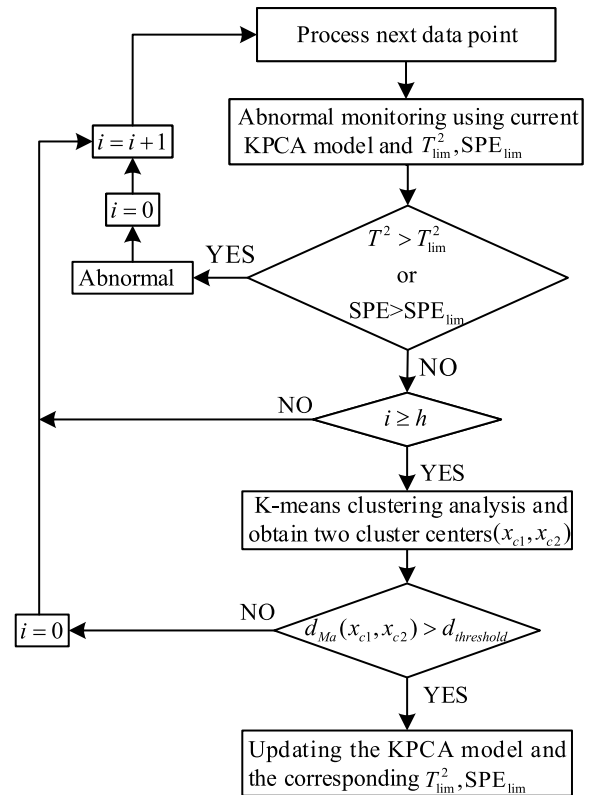


FIGURE 3. The flowchart of fault detection algorithm based on the modified MWKPCA.

rate, fault detection rate, fault detection time, etc.) and the calculation amount of the two algorithms are compared under the normal operating condition and the sensor faults condition, respectively. The comparison and simulation verification of the two algorithms are mainly divided into three parts: (1) Introduction of the aircraft turbofan engine model. (2) Simulation results and comparative analysis under normal condition. (3) Simulation results and comparative analysis under sensor faults condition. The three parts are described in detail as follows.

**A. THE DESCRIPTION OF THE AIRCRAFT TURBOFAN ENGINE NONLINEAR MODEL**

The architecture of a civil aircraft turbofan engine is shown in Fig. 4. The main components consist of the fan, low pressure compressor (LPC), high pressure turbine (HPC), combustor, high pressure turbine (HPT), low pressure turbine (LPT), and two rotating spools. The station definitions are shown along the bottom of the Fig. 4. The aircraft turbofan engine model is developed in [39]. The sampling period of the engine model is 0.02s. The control inputs to this engine model are the fuel flow ( $WFM$ ), variable bleed valve position (VBV) and variable stator vane angle (VSV). The controller of this engine model is designed in [40]. The control law of the engine under steady-state condition is to control the fuel flow to keep the fan speed constant by giving the desired speed

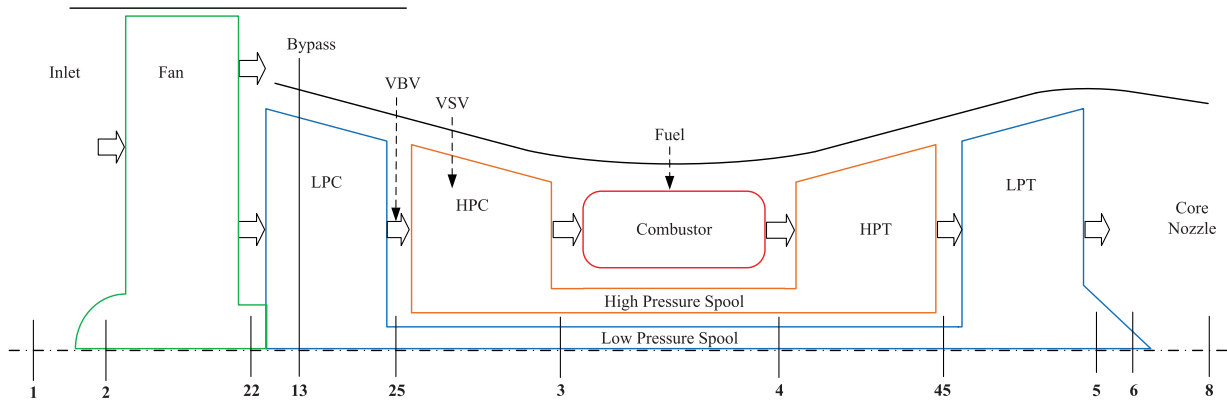


FIGURE 4. Diagram of the commercial turbofan engine components and station designations.

TABLE 3. Fault detection algorithm based on the modified MWKPCA.

**offline training:**

1. Obtain a normal condition data set  $\mathbf{X} \in \mathbb{R}^{L \times n}$ , and get  $\bar{\mathbf{X}} \in \mathbb{R}^{L \times n}$  through data normalization, set initial cumulative number  $i = 0$ .
2. Compute the kernel matrix ( $\mathbf{K} \in \mathbb{R}^{L \times L}$ ), and obtain  $\bar{\mathbf{K}}$  through kernel matrix centered.
3. Calculate the eigenvalues and eigenvectors of  $\bar{\mathbf{K}}$ , and determine the number of principals.
4. Determine the control limit  $T_{lim}^2$  and  $SPE_{lim}$ .

**Online fault detection:**

5. Set the length of data window  $L$ , moving forward step  $h$ , kernel parameters and the threshold of Mahalanobis distance of cluster centers  $d_{threshold}$  and set initial cumulative number  $i = 0$ .
6. Obtain a new test data sample  $\mathbf{x}_k \in \mathbb{R}^{1 \times n}$  and scale it.
7. Calculate the projection of  $\mathbf{x}_k$  in principal component direction by using equation (11).
8. Calculate the two statistics  $T_k^2$  and  $SPE_k$  by using equation (12).
9. If the  $T_k^2$  and the  $SPE_k$  do not exceed the corresponding control limit  $T_{lim}^2$  and  $SPE_{lim}$ ,  $\mathbf{x}_k$  is considered as a normal data and perform an accumulation operation ( $i = i + 1$ ). If the  $T_k^2$  or the  $SPE_k$  exceeds the control limit, the cumulative number  $i$  returns to the initial value of zero.
10. If the data collected for  $h$  consecutive times is normal sample ( $i = h$ ), turn to step 11. If the cumulative number  $i$  is less than  $h$ , return to step 6.
11. Using k-means clustering method to calculate the two clustering centers of the data set constructed by the latest  $h$  normal samples and the oldest  $h$  normal samples in the current data window, calculate the Mahalanobis distance of the two clustering centers  $d(\mathbf{x}_{c1}, \mathbf{x}_{c2})$ .
12. Determine whether the Mahalanobis distance  $d(\mathbf{x}_{c1}, \mathbf{x}_{c2})$  exceeds the threshold  $d_{threshold}$  or not. If it exceeds the threshold, update the KPCA model and turn to step 13, otherwise the KPCA model will not be updated and return to step 6.
13. Update the KPCA model and the control limit of the fault detection statistics.
14. Return to step 6.

of the fan. Two spool rotation and five gas path sensors are chosen as measurement parameters of the engine, which are used in most commercial turbofan engines [41]. The Gaussian white noise is added to the output of the simulation results of the seven parameters, which ensuring that the simulation approximates the actual engine conditions as closely as possible. The description of the seven parameters and their conventional deviations are listed in Table. 4, the installation position of the sensors can be seen from the indices at bottom of the Fig. 4.

**B. SIMULATION RESULTS UNDER NORMAL OPERATING CONDITION**

In this part, the cruise operating condition is chosen for this simulation. In this simulation process, the changes of the performance degradation level  $F$  of HPT, the fuel flow  $WFM$  and the seven sensor measurement parameters are shown in Fig. 5. Under the action of the engine controller, the fan speed remains basically unchanged in this simulation process. The simulation ran for 500s then 25000 samples were collected.

A radial basis kernel function is selected as the kernel function, as shown in equation (22).

$$k(x, y) = \exp(-\|x - y\|^2 / \gamma) \tag{22}$$

where  $\gamma = am\sigma^2$ , and  $m, \sigma^2$  denote the input space and the variance of the data, respectively.  $a$  is a constant, which can be determined by consideration of the process to be monitored [28]. The  $th_{CPV}$  in equation (10) is set to 0.95, and the confidence limit of the  $T^2$  and SPE index is set to 0.99. The length of the data window and the moving forward step  $h$  are set to 200 and 5, respectively. According to the above parameter settings, the conventional MWKPCA algorithm can be implemented. And the  $d_{threshold}$  can be determined according to equation (21), then the modified MWKPCA algorithm can be implemented in the same fault detection process as the conventional MWKPCA algorithm. It is worth mentioning that the parameter settings of the proposed method, such as the length of the time window, the fixed moving steps and the kernel function selection can be optimized based on the fault detection performance. Since the purpose of this article has been limited to reducing the computation cost of the conventional MWKPCA method, considering the parameters optimization is beyond the scope of the present study, the parameter settings are kept consistent before and after the algorithm modification.

The fault detection results of the conventional MWKPCA and the modified MWKPCA using  $T^2$  statistic and SPE statistic in the case of normal operating condition are presented respectively in Fig. 6 and Fig. 7. The performance of the two algorithms is presented in Table. 5. We notice

TABLE 4. Sensors and conventional deviations for the commercial turbofan engine model.

Symbol	Description	Unit	Stand Deviation
$N_l$	Fan speed	r/min	0.25%
$N_h$	Core speed	r/min	0.17%
$T_{25}$	Total temperature at LPC outlet	K	0.5%
$P_{25}$	Total pressure at LPC outlet	kPa	0.2%
$T_3$	Total temperature at HPC outlet	K	0.16%
$P_{s3}$	Static pressure at HPC outlet	kPa	0.16%
$T_{45}$	Total temperature at LPT outlet	K	0.5%

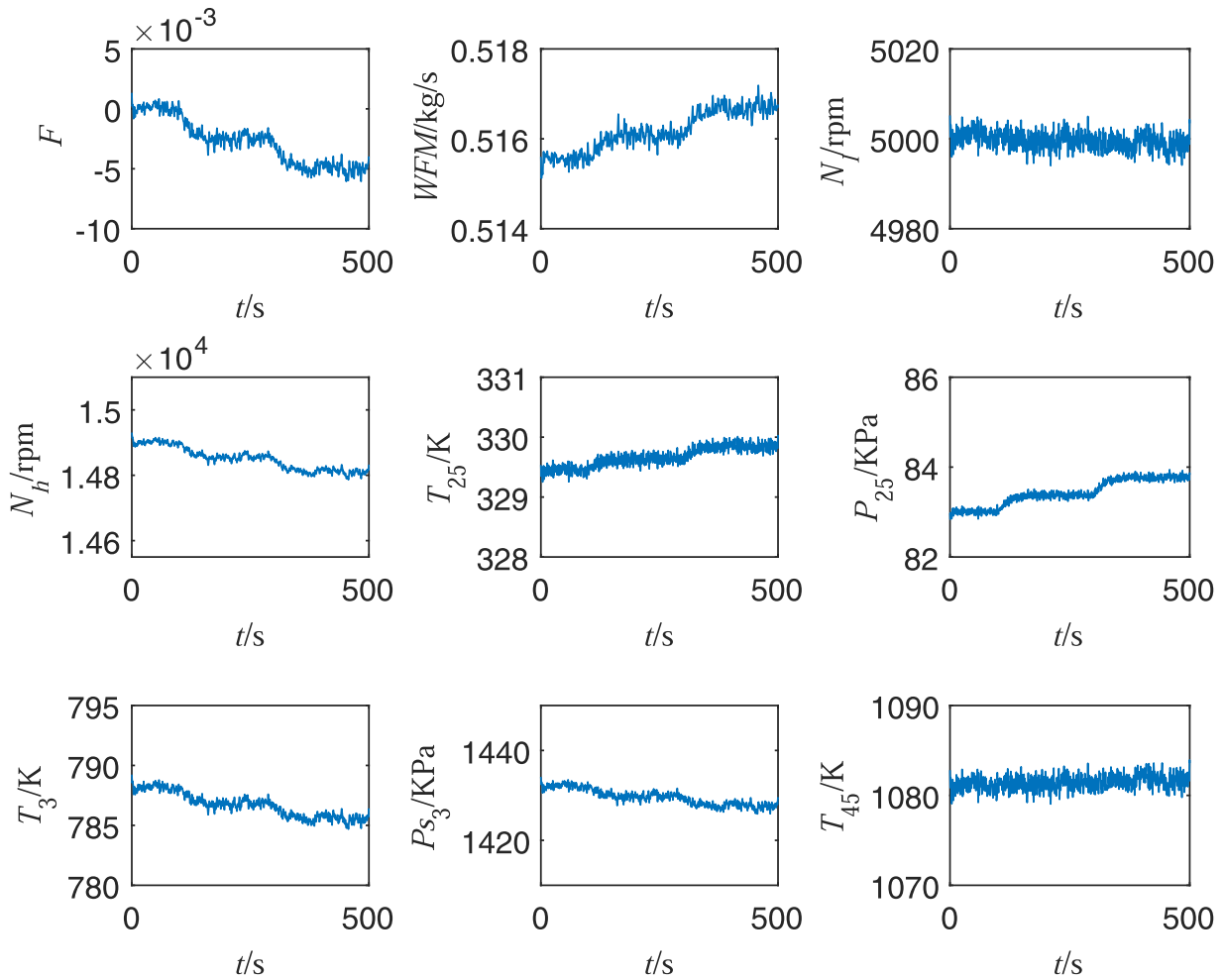


FIGURE 5. The setting of engine performance degradation and the change of the measurement parameters.

that both of the two algorithms meet the requirement of the false alarm rate limit ( $T_{FAR}$ ). However, the conventional MWKPCA algorithm updates the KPCA model in fixed steps and the number of the total update times is 4492, the modified MWKPCA algorithm updates the KPCA model adaptively in the fault detection process and the number of the total update times is 76. The eigenvalues and eigenvectors of a kernel matrix of size  $(L, L)$  are needed to be calculated when the KPCA model is updating. And the updating process consumes  $O(L^3)$ , which is far great than the calculation cost of the recursive calculation process of cluster centers.

Therefore, by comparing the update times of KPCA model in the fault detection process, the modified MWKPCA algorithm performs better than the conventional MWKPCA algorithm in terms of calculation amount.

C. SIMULATION RESULTS UNDER SENSOR FAULTS CONDITION

The fault detection performance of the two algorithms under sensor faults condition is compared in this part. The setting of the flight condition and the component degradation level of HPT are the same as the normal operating condition.



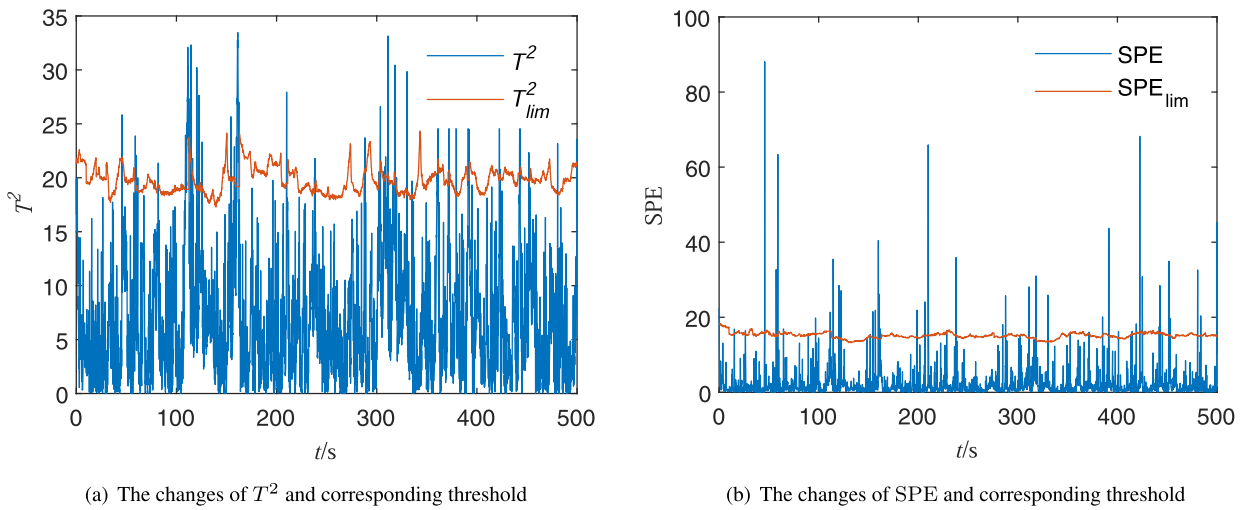


FIGURE 6. The changes of  $T^2$  and SPE based on the conventional MWKPCA.

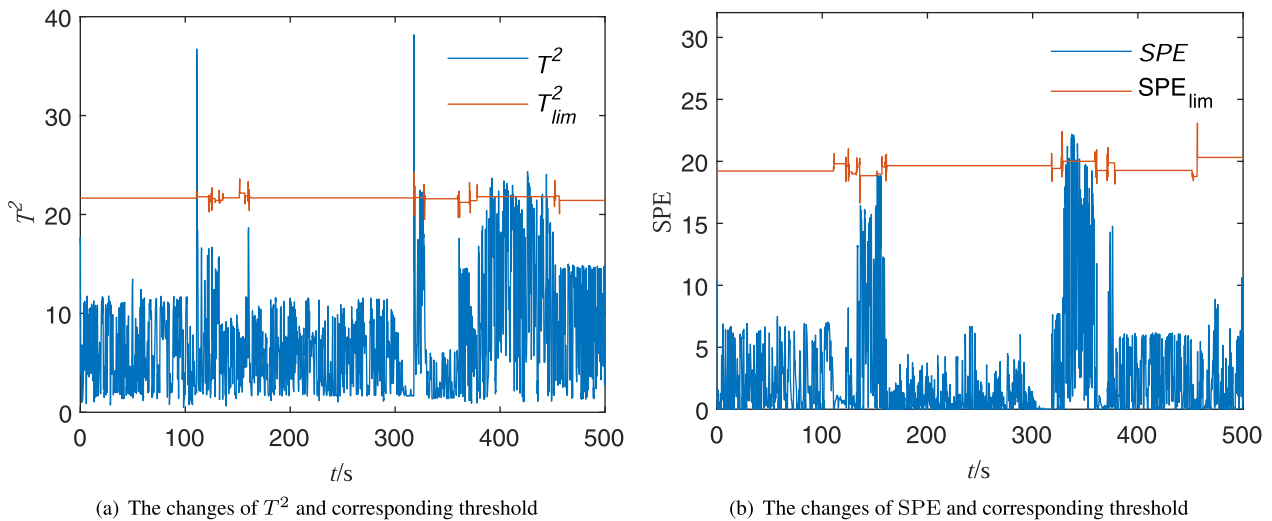


FIGURE 7. The changes of  $T^2$  and SPE based on the modified MWKPCA.

TABLE 5. Performance of the conventional MWKPCA and the modified MWKPCA for online fault detection under normal condition.

Approach	$T_{FAR}$	KPCA Model Update times
Conventional MWKPCA	2.78%	4492
Modified MWKPCA	2.82%	76

And it is assumed that three sensor faults have been occurred in three different time intervals for the measurements  $N_h$ ,  $P_{S3}$  and  $T_{45}$ . These faults are presented in Table. 6. The change of the performance degradation level  $F$  of HPT, the fuel flow  $WFM$  and the seven sensor measurement parameters in this simulation process are shown in Fig. 8.

The setting of the kernel function and the related parameters in this fault detection process is as same as the simulation under the normal operation, which is shown in the former part. The monitoring results of the conventional MWKPCA

TABLE 6. Sensor faults description.

case	Faults description		
	Affected variable	Fault description	Fault time (s)
Bias	$N_h$	$f_1 = 1.5\%$	30-70
Drift	$P_{S3}$	$f_2 = -0.05\% \times \Delta t$	230-270
Bias	$T_{45}$	$f_3 = -2\%$	430-470

and the modified MWKPCA using  $T^2$  and SPE are presented respectively in Fig. 9 and Fig. 10. From these figures, all the injected faults are detected at time.

The fault detection results of the two algorithms are shown in Table. 7. In Table. 7 the false alarm rate, fault detection time, and the update times of KPCA models are summarized. We notice that the modified MWKPCA algorithm provides a comparable performance in term of false alarm rate and the fault detection time than the conventional MWKPCA

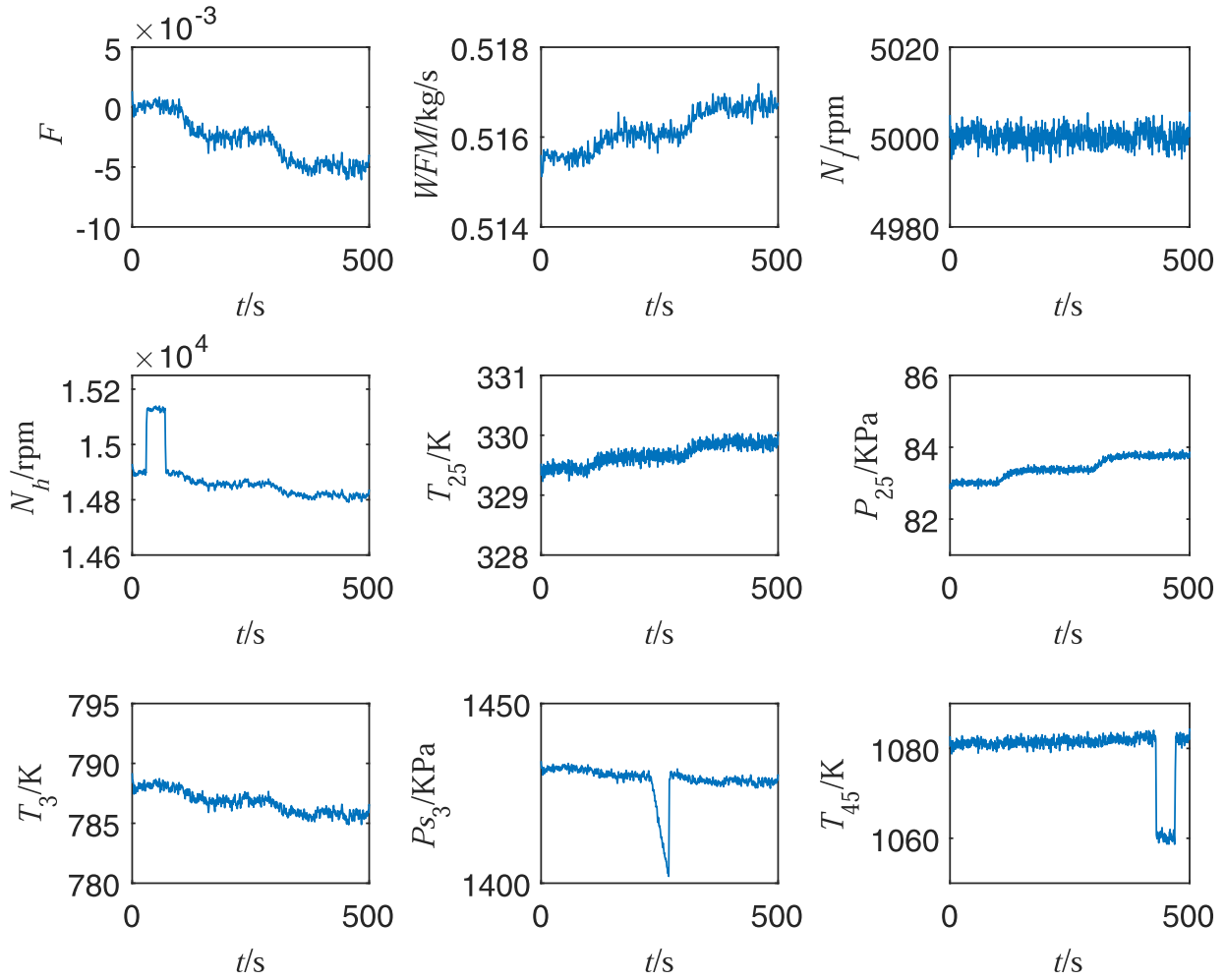


FIGURE 8. The setting of engine performance degradation and the change of the measurement parameters with sensor faults.

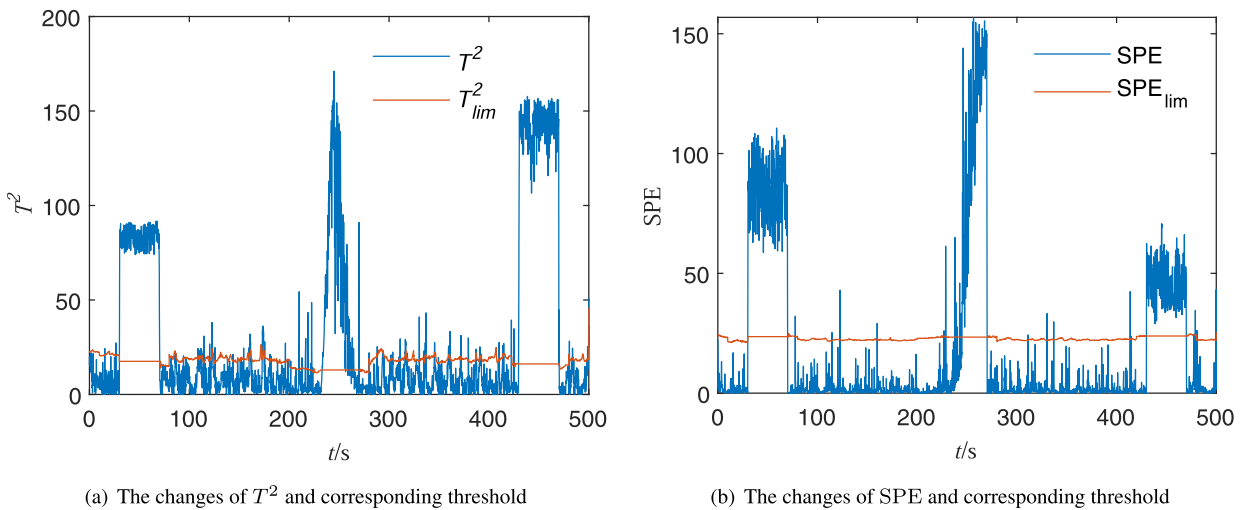


FIGURE 9. The changes of  $T^2$  and SPE based on the conventional MWKPCA.

algorithm. However, the two algorithms have significant differences in the update times of KPCA model in the process of fault detection, which means that in the case of sensor fault,

the modified MWKPCA can reduce the calculation amount of the algorithm and improve the execution efficiency of the algorithm.

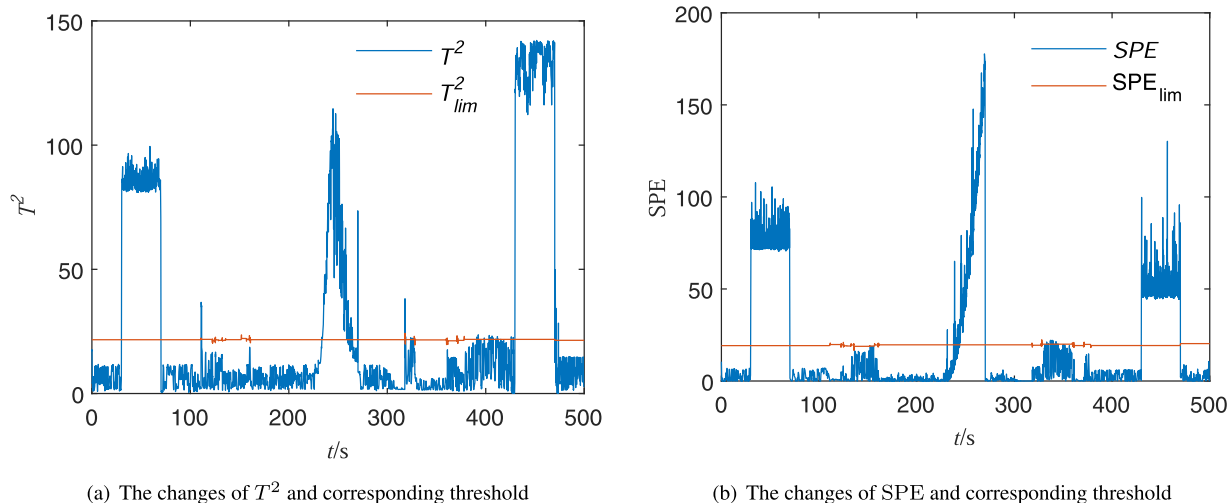


FIGURE 10. The changes of  $T^2$  and SPE based on the modified MWKPCA.

TABLE 7. Performances of the conventional MWKPCA and the modified MWKPCA for online fault detection under sensor faults condition.

Approach	$T_{FAR}$	Time to detection (s)			KPCA model update times
		$N_h$ bias fault	$P_{s3}$ drift fault	$T_{45}$ bias fault	
Conventional MWKPCA	2.43%	0.1	4.28	0.04	3419
Modified MWKPCA	2.79%	0.12	4.02	0.08	75

V. CONCLUSION

In this article, a novel fault detection based on the modified MWKPCA is proposed for aircraft turbofan engine. In this method, a new KPCA updating mechanism for the conventional MWKPCA is established based on the k-means clustering and Mahalanobis distance of the clustering centers. The proposed method is verified under normal operating condition and sensor faults condition, and the normal degradation of the components of the aircraft turbofan engine is considered in both of the two simulation cases. Compared with the conventional MWKPCA algorithm, the improved MWKPCA algorithm can realize updating KPCA model adaptively according to the changes of the system output parameters. And the modified MWKPCA algorithm has less KPCA update operations in the fault detection process, therefore, it has better performances in terms of computation cost. A future research direction is to design a real time fault isolation and fault reconstruction strategy based on the proposed modified MWKPCA algorithm.

REFERENCES

- [1] M. Tahan, E. Tsoutsanis, M. Muhammad, and Z. A. Abdul Karim, "Performance-based health monitoring, diagnostics and prognostics for condition-based maintenance of gas turbines: A review," *Appl. Energy*, vol. 198, pp. 122–144, Jul. 2017.
- [2] J. Lu, J. Huang, and F. Lu, "Kernel extreme learning machine with iterative picking scheme for failure diagnosis of a turbofan engine," *Aerosp. Sci. Technol.*, vol. 96, Jan. 2020, Art. no. 105539.
- [3] N. B. Zhao, X. Y. Wen, and S. Y. Li, "A review on gas turbine anomaly detection for implementing health management," in *Proc. ASME Turbo Expo Turbomachinery Tech. Conf. Expo.*, Jun. 2016, Art. no. V001T22A009.
- [4] A. J. Volponi, "Gas turbine engine health management: Past, present, and future trends," *J. Eng. Gas Turbines Power*, vol. 136, no. 5, May 2014, Art. no. 051201.
- [5] D. L. Simon and A. W. Rinehart, "Model-based anomaly detection approach for analyzing streaming aircraft engine measurement data," in *Proc. ASME Turbo Expo Turbine Tech. Conf. Expo.*, Jun. 2014, pp. 1–11.
- [6] E. Naderi, N. Meskin, and K. Khorasani, "Nonlinear fault diagnosis of jet engines by using a multiple model-based approach," *J. Eng. Gas Turbines Power*, vol. 134, no. 1, Jan. 2012, Art. no. 011602.
- [7] R. Rausch, D. Viassolo, A. Kumar, K. Goebel, N. Eklund, B. Brunell, and P. Bonanni, "Towards in-flight detection and accommodation of faults in aircraft engines," in *Proc. AIAA 1st Intell. Syst. Tech. Conf.*, Sep. 2004, p. 6463.
- [8] E. Naderi and K. Khorasani, "Data-driven fault detection, isolation and estimation of aircraft gas turbine engine actuator and sensors," *Mech. Syst. Signal Process.*, vol. 100, pp. 415–438, Feb. 2018.
- [9] M. Tahan, M. Muhammad, and Z. A. A. Karim, "A multi-nets ANN model for real-time performance-based automatic fault diagnosis of industrial gas turbine engines," *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 39, no. 7, pp. 2865–2876, Mar. 2017.
- [10] M. Amozegar and K. Khorasani, "An ensemble of dynamic neural network identifiers for fault detection and isolation of gas turbine engines," *Neural Netw.*, vol. 76, pp. 106–121, Apr. 2016.
- [11] Y. Jiang, S. Yin, and O. Kaynak, "Optimized design of parity relation based residual generator for fault detection: Data-driven approaches," *IEEE Trans. Ind. Informat.*, early access, Apr. 16, 2020, doi: 10.1109/TII.2020.2987840.
- [12] S. Yin, J. J. Rodríguez-Andina, and Y. Jiang, "Real-time monitoring and control of industrial cyberphysical systems: With integrated plant-wide monitoring and control framework," *IEEE Ind. Electron. Mag.*, vol. 13, no. 4, pp. 38–47, Dec. 2019.
- [13] S. Sarkar, X. Jin, and A. Ray, "Data-driven fault detection in aircraft engines with noisy sensor measurements," *J. Eng. Gas Turbines Power*, vol. 133, no. 8, pp. 783–789, Aug. 2011.
- [14] J. Yang, Y. Guo, and W. Zhao, "Long short-term memory neural network based fault detection and isolation for electro-mechanical actuators," *Neurocomputing*, vol. 360, pp. 85–96, Sep. 2019.

- [15] Z. N. Sadough Vanini, K. Khorasani, and N. Meskin, "Fault detection and isolation of a dual spool gas turbine engine using dynamic neural networks and multiple model approach," *Inf. Sci.*, vol. 259, pp. 234–251, Feb. 2014.
- [16] S. Sina Tayarani-Bathaie and K. Khorasani, "Fault detection and isolation of gas turbine engines using a bank of neural networks," *J. Process Control*, vol. 36, pp. 22–41, Dec. 2015.
- [17] S. Yazdani and M. Montazeri-Gh, "A novel gas turbine fault detection and identification strategy based on hybrid dimensionality reduction and uncertain rule-based fuzzy logic," *Comput. Ind.*, vol. 115, Feb. 2020, Art. no. 103131.
- [18] S. O. T. Ogaji, L. Marinai, S. Sampath, R. Singh, and S. D. Prober, "Gas-turbine fault diagnostics: A fuzzy-logic approach," *Appl. Energy*, vol. 82, no. 1, pp. 81–89, Sep. 2005.
- [19] Y. K. Lee, D. N. Mavris, V. V. Volovoi, M. Yuan, and T. Fisher, "A fault diagnosis method for industrial gas turbines using Bayesian data analysis," *J. Eng. Gas Turbines Power*, vol. 132, no. 4, Apr. 2010, Art. no. 041602.
- [20] F. Nadir, H. Elias, and B. Messaoud, "Diagnosis of defects by principal component analysis of a gas turbine," *Social Netw. Appl. Sci.*, vol. 2, no. 5, pp. 980–987, Apr. 2020.
- [21] A. Bakdi, A. Kouadri, and A. Bensmail, "Fault detection and diagnosis in a cement rotary kiln using PCA with EWMA-based adaptive threshold monitoring scheme," *Control Eng. Pract.*, vol. 66, pp. 64–75, Sep. 2017.
- [22] I. Portnoy, K. Melendez, H. Pinzon, and M. Sanjuan, "An improved weighted recursive PCA algorithm for adaptive fault detection," *Control Eng. Pract.*, vol. 50, pp. 69–83, May 2016.
- [23] S. Yin, C. Yang, J. Zhang, and Y. Jiang, "A data-driven learning approach for nonlinear process monitoring based on available sensing measurements," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 643–653, Jan. 2017.
- [24] S. J. Zhao, J. Zhang, and Y. M. Xu, "Performance monitoring of processes with multiple operating modes through multiple PLS models," *J. Process Control*, vol. 16, no. 7, pp. 763–772, Aug. 2006.
- [25] C. Zhao, F. Wang, Z. Mao, N. Lu, and M. Jia, "Adaptive monitoring based on independent component analysis for multiphase batch processes with limited modeling data," *Ind. Eng. Chem. Res.*, vol. 47, no. 9, pp. 3104–3113, May 2008.
- [26] K. Nishimura, S. Matsuura, and H. Suzuki, "Multivariate EWMA control chart based on a variable selection using AIC for multivariate statistical process monitoring," *Statist. Probab. Lett.*, vol. 104, pp. 7–13, Sep. 2015.
- [27] C. Zhao, F. Wang, N. Lu, and M. Jia, "Stage-based soft-transition multiple PCA modeling and on-line monitoring strategy for batch processes," *J. Process Control*, vol. 17, no. 9, pp. 728–741, Oct. 2007.
- [28] J.-M. Lee, C. Yoo, S. W. Choi, P. A. Vanrolleghem, and I.-B. Lee, "Nonlinear process monitoring using kernel principal component analysis," *Chem. Eng. Sci.*, vol. 59, no. 1, pp. 223–234, Jan. 2004.
- [29] R. Fazai, O. Taouali, M. F. Harkat, and N. Bouguila, "A new fault detection method for nonlinear process monitoring," *Int. J. Adv. Manuf. Technol.*, vol. 87, nos. 9–12, pp. 3425–3436, Apr. 2016.
- [30] I. Jaffel, O. Taouali, M. F. Harkat, and H. Messaoud, "Moving window KPCA with reduced complexity for nonlinear dynamic process monitoring," *ISA Trans.*, vol. 64, pp. 184–192, Sep. 2016.
- [31] Y. Su, L. Shu, and K.-L. Tsui, "Adaptive EWMA procedures for monitoring processes subject to linear drifts," *Comput. Statist. Data Anal.*, vol. 55, no. 10, pp. 2819–2829, Oct. 2011.
- [32] I. B. Khediri, M. Limam, and C. Weihs, "Variable window adaptive kernel principal component analysis for nonlinear nonstationary process monitoring," *Comput. Ind. Eng.*, vol. 61, no. 3, pp. 437–446, Oct. 2011.
- [33] M. Navi, N. Meskin, and M. Davoodi, "Sensor fault detection and isolation of an industrial gas turbine using partial adaptive KPCA," *J. Process Control*, vol. 64, pp. 37–48, Apr. 2018.
- [34] Y. Zhang, S. Li, and Y. Teng, "Dynamic processes monitoring using recursive kernel principal component analysis," *Chem. Eng. Sci.*, vol. 72, pp. 78–86, Apr. 2012.
- [35] I. Jaffel, O. Taouali, M. F. Harkat, and H. Messaoud, "A fault detection index using principal component analysis and mahalnobis distance," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 1397–1401, Oct. 2015.
- [36] P. Nomikos and J. F. MacGregor, "Multivariate SPC charts for monitoring batch processes," *Technometrics*, vol. 37, no. 1, pp. 41–59, Feb. 1995.
- [37] K. P. Sinaga and M.-S. Yang, "Unsupervised K-means clustering algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020.
- [38] D. L. Simon, "Propulsion diagnostic method evaluation strategy (ProDiMES) user's guide," Glenn Res. Center, Cleveland, OH, USA, Tech. Rep. NASA/TM—2010-215840, Jan. 2010.
- [39] S. G. Zhang, Y. Q. Guo, and J. LU, "Research on aircraft engine component-level models based on GasTurb/MATLAB," *J. Aerosp. Power*, vol. 27, no. 12, pp. 2850–2856, Dec. 2012.
- [40] J. F. Wu and Y. Q. Guo, "Design and simulation of aero-engine steady-state and transient-state control integration," *J. Aerosp. Power*, vol. 28, no. 6, pp. 1436–1440, Jun. 2013.
- [41] J. DeCastro, J. Litt, and D. Frederick, "A modular aero-propulsion system simulation of a large commercial aircraft engine," in *Proc. 44th AIAA/ASME/SAE/ASEE Joint Propuls. Conf. Exhib.*, Jul. 2008, p. 4579.



**HAO SUN** received the B.S. and M.S. degrees in power machinery and engineering from Northwestern Polytechnical University, China, in 2013 and 2016, respectively, where he is currently pursuing the Ph.D. degree. His research interests include fault diagnosis and fault tolerant control of aircraft turbofan engine.



**YINGQING GUO** received the Ph.D. degree in engineering from Northwestern Polytechnical University, China, in 2000. He has been a Professor with the School of Power and Energy, Northwestern Polytechnical University, since 2003. His main research interests include advanced engine control technology, system status monitoring, and health management technology. He is a Senior Member of the American Aerospace Association (AIAA) and a member of the Engine Control Professional Committee of the China Aviation Society.



**WANLI ZHAO** received the B.S. degree from Harbin Engineering University, in 2017. He is currently pursuing the Ph.D. degree in power and energy engineering with Northwestern Polytechnical University, China. His research interests include gas turbine fault diagnosis and hardware design in loop simulation.

...