

l_{21} -Norm-Based Robust Feature Extraction Method for Fault Detection

Xin Sha* and Naizhe Diao*

Cite This: *ACS Omega* 2022, 7, 43440–43449

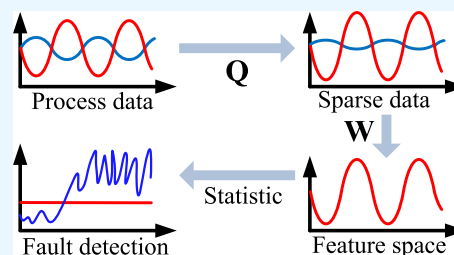
Read Online

ACCESS |

Metrics & More

Article Recommendations

ABSTRACT: Industrial data are in general corrupted by noises and outliers, which do not meet the application assumptions in feature extraction. Many existing feature extraction algorithms are not robust, overly consider the less important features of the data, and cannot capture the key features of the data. To this end, the two-level feature extraction method (TFEM) based on l_{21} -norm is proposed in this study. Compared with single-projection feature extraction algorithms, TFEM consists of two projections: the nonreduced and reduced dimensionality projections. The nonreduced dimensionality projection can remove the parts of less important features that are unrelated to the key features of the data. The reduced dimensionality projection can reduce the dimensionality of the data and further extract the features of the data. In addition, l_{21} -norm is used to make the algorithm more robust. Finally, the convergence of the proposed algorithm is analyzed. Extensive experiments have been conducted on the Tennessee Eastman and Penicillin Fermentation processes to demonstrate that the proposed method is more effective than other state-of-the-art fault detection methods.



1. INTRODUCTION

Process monitoring, which is an important technique to improve process safety and ensure product quality, has attracted extensive attention from both academia and the industry over the past 20 years.^{1–3} Data-based process monitoring methods, especially multivariate statistical process monitoring, have attracted considerable attention and developed rapidly because they do not require rigorous system models or prior knowledge about the process.

Data-based fault diagnosis methods often establish statistics through feature extraction algorithms and then judge whether a fault occurs according to the statistics. Many traditional feature extraction algorithms have been improved to adapt to different industrial scenarios. Jiang et al.⁴ proposed a parallel PCA-KPCA (P-PCA-KPCA) modeling and monitoring scheme combining randomized algorithm and genetic algorithm for process monitoring of linearly correlated and nonlinearly correlated variables. Tian et al.⁵ proposed a weighted copula-correlation multi-block principal component analysis method, which avoids the participation of noise variables and retains important information. Zhang et al.⁶ proposed a process monitoring method based on feature extraction, which is a common subspace feature extraction method based on tensor decomposition, considering both common scores and weights. Ye et al.⁷ proposed an improved multilinear feature extraction method and feature selection strategy. The method extracts features and monitors multi-channel data in conjunction with multivariate control charts. Zhang et al.⁸ proposed an improved locality preserving projection based on the heat-

kernel cosine weight matrix, namely, heat-kernel and cosine weights locality preserving projections (HC-LPP). Compared with other related methods, the HC-LPP method has higher diagnosis accuracy. However, the above situation may not hold in the following two cases: (1) industrial data are generally corrupted by noises and outliers; (2) there are many non-obvious features in the data, that is, less important features, which can affect the detection or classification performance.

Many studies have demonstrated that the l_0 -norm and l_1 -norm can improve the robustness of the algorithm,^{9–11} and many fault diagnosis methods based on l_0 -norm and l_1 -norm have been proposed. Rahoma et al.¹² proposed a fault detection and diagnosis method based on sparse principal component analysis (SPCA). The method selects the number of non-zero loads (NNZLs) of SPCA according to the false alarm rate and fault detection rate (FDR) and has certain robustness. Xiao et al.¹³ proposed a dynamic process monitoring method based on sparse representation. The method reduces dimensionality in clean data space and is robust to noise and outliers. Xu and Ding¹⁴ proposed an efficient and robust process monitoring method based on similar sparse cooperative embedding. The algorithm learns a

Received: May 26, 2022

Accepted: October 24, 2022

Published: November 25, 2022



sparse coefficient matrix as a sparse constraint for reconstruction errors through l_1 -norm regularization and is robust to data contaminated by outliers. However, l_0 -norm optimization involves a combinatorial optimization problem, which requires searching and optimizing all possible solutions, and its solution is an NP-hard problem.^{15,16} Moreover, the l_1 -norm does not have rotation invariance. Therefore, the l_{21} -norm-based PCA algorithm was developed by Ding et al.¹⁷ and Nie et al.¹⁸ In addition, studies have shown that the l_{21} -norm can effectively find outliers in data set compared with the l_1 -norm.¹⁹ In summary, l_{21} -norm is more suitable for enhancing the robustness of algorithms than other types of norms.

Another method to eliminate the influence of outliers is the low-rank representation (LRR), which aims to capture the lowest rank representation of the data. Pan et al.²⁰ introduced the traditional LRR into the principal component pursuit (PCP) method, constructing a low-rank coefficient matrix to represent explicit relationships between variables. Subsequently, Pan et al.²¹ proposed a new fault detection method, robust PCP. Applying the proposed robust PCP method, low-rank matrices and explicit variable relationships containing important process information are obtained, as well as block sparse matrices containing small faults. Fu et al.²² proposed a low-rank joint embedding method. This method captures the global structure of the raw data through low-rank joint embeddings, alleviating the negative effects of outliers. At the same time, the monitoring capability is enhanced by introducing manifold regularization to preserve the local geometry of the data. However, the rank function is a discrete function, minimizing the rank is an NP-hard problem.

In this study, the two-level feature extraction method (TFEM) based on l_{21} -norm is proposed. The method mainly includes the nonreduced and reduced dimensionality projections, which are used to capture the key features of the data. To construct the statistics, the T^2 statistics are used for both the feature space and the residual space, and the T^2 statistic considers the correlation among variables. The main contributions of this study are summarized as follows:

1. Two projections, namely, the nonreduced and reduced dimensionality projections, are used simultaneously. The nonreduced dimensionality projection removes less important features that cannot reflect the features of data. The reduced dimensionality projection is used to reduce the dimension of data.
2. l_{21} -norm is used to increase the robustness of the algorithm. The biconvex optimization is used to optimize the projection matrix that makes the optimization solution simpler, and its convergence is analyzed.
3. A large number of tests are performed on the TE and Penicillin fermentation processes data set. Random noise is added to several training data. The experiments demonstrated that our algorithm can achieve higher FDR.

The remainder of the paper is organized as follows: Section 2 explains the algorithm proposed in this study, and Section 3 applies the algorithm in the Tennessee Eastman (TE) process, followed by comparing it with the other methods. Section 4 applies the algorithm in the Penicillin fermentation process. Section 5 presents the conclusions and prospects of the algorithm proposed in this study.

2. ALGORITHM

2.1. Brief Introduction to Standard Notations and Terminology. Let us assume a matrix $\mathbf{M} = (m_{ij})$, whose i -th row and j -th column are represented as m_{ij} . The l_{21} -norm of the matrix is expressed as¹⁹

$$\|\mathbf{M}\|_{l_{21}} = \sum_{i=1}^n \sqrt{\sum_{j=1}^m m_{ij}^2} = \sum_{i=1}^n \|\mathbf{m}^i\|_2 \quad (1)$$

where \mathbf{m}^i is the i -th row of matrix \mathbf{M} .

2.2. Two-Level Feature Extraction Method. This section mainly introduces the TFEM algorithm. The non-reduced dimensionality projection is employed to maintain the dimensionality of data, while simultaneously making the projected data sparse. When the dimensionality of data is too high, its visibility deteriorates; thus, a reduced dimensionality is added to the data. During the reduced dimensionality of data, we need to highlight the features of data while minimizing the loss function. l_{21} -norm is used to improve the robustness of the algorithm. Therefore, we described the final model as follows

$$\begin{aligned} \arg \min_{\mathbf{Q}, \mathbf{W}} \quad & \frac{1}{2} \|(\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T\|_{l_{21}} + \frac{1}{2} \lambda \|(\mathbf{Q}^T \mathbf{X})^T\|_{l_{21}} \\ \text{s. t.} \quad & \mathbf{W}^T \mathbf{W} = \mathbf{I}, \lambda > 0 \end{aligned} \quad (2)$$

where $\mathbf{X} \in \mathbf{R}^{m \times n}$ is a data set, $\mathbf{Q} \in \mathbf{R}^{m \times m}$ is the nonreduced dimensionality projection matrix, m is the number of process variables, and n is the total number of measurements for each process variable. $\mathbf{W} \in \mathbf{R}^{m \times l}$ is the reduced dimensionality projection matrix. λ is the trade-off parameter used to balance the weight relationship between the first and second items. l is the dimensionality of the feature space obtained through many experiments. The first item is the process of projecting the high-dimensional data onto low-dimensional data while ensuring that the loss function is minimized. The second item can reduce affected by less important that cannot reflect the features of data.

Figure 1a shows 500 fault 0 data after the standardization of the TE process data set. Figure 1b shows 800 fault 1 data after

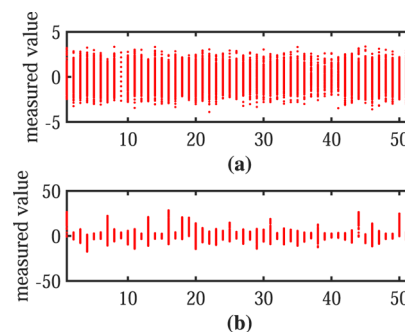


Figure 1. Data after standardization of the TE process data set. (a) Fault 0. (b) Fault 1.

the standardization of the TE process data set. The horizontal axis and vertical axis represent the process variables and measured values for each process variable, respectively. As shown in Figure 1a, the fault 0 data after being standardized fluctuate around 0 for most process variables. That is, if most process variables are kept near 0, it can reflect the feature of

fault 0 data efficiently. Generally, fluctuations near 0 are caused by less important which cannot reflect the feature of the fault 0 data. Therefore, a nonreduced dimensionality projection matrix \mathbf{Q} is used to make most process variables of data as sparse as possible without reducing the dimensionality of data. As shown in Figure 1b, the fault 1 data after being standardized usually fluctuates around 0 for some process variables. When the nonreduced dimensionality projection matrix \mathbf{Q} is used, the difference between the fault 1 data and fault 0 data is more obvious.

As shown in Figure 2, data set (a) is distributed along the X -axis, and the features of data have a significant relationship with

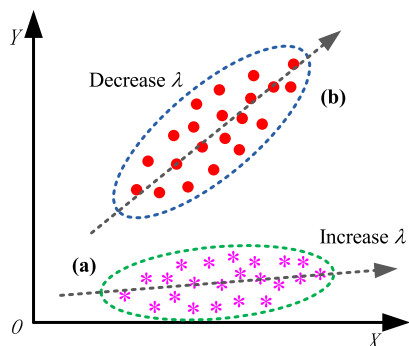


Figure 2. Two data sets that require different λ . (a) Data need to increase λ . (b) Data need to decrease λ .

the X direction of data; thus, the value of λ needs to be increased. As shown in Figure 2, data set (b) is not distributed along the X -axis or Y -axis; thus, each direction of data can reflect the features of data, and the features of data have a great relationship with any direction of data. In order to maintain the features of data, the weight of the loss function needs to be increased, and the value of λ needs to be reduced. After a lot of experiments, we found that the experiment effect is best when $0 \leq \lambda \leq 1$.

2.3. Algorithm Solving. Because both the terms are nonsmooth, it is difficult to directly solve the equation for the optimization problem. We simplify eq 2 as¹⁹

$$\arg \min_{\mathbf{Q}, \mathbf{W}} \text{tr}((\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T) + \lambda \text{tr}(\mathbf{Q}^T \mathbf{X} \mathbf{D}_2 (\mathbf{Q}^T \mathbf{X})^T) \quad (3)$$

Among them \mathbf{D}_1 and \mathbf{D}_2 are diagonal matrices. Define $((\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T)_i$ ($i = 1, 2, \dots, n$) as the i -th row of the matrix $(\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T$. Define $((\mathbf{Q}^T \mathbf{X})^T)_i$ as the i -th row of the matrix $(\mathbf{Q}^T \mathbf{X})^T$. Moreover, the i -th diagonal element of \mathbf{D}_1 is $\frac{1}{2 \|((\mathbf{W}^T \mathbf{X} - \mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T)_i\|_2}$, the i -th diagonal element of \mathbf{D}_2 is $\frac{1}{2 \|((\mathbf{Q}^T \mathbf{X})^T)_i\|_2}$.

Fix \mathbf{W} , by simplifying eq 3, we get

$$\arg \min_{\mathbf{Q}} \text{tr}(\mathbf{W} \mathbf{W}^T (\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T) + \lambda \text{tr}(\mathbf{Q}^T \mathbf{X} \mathbf{D}_2 \mathbf{X}^T \mathbf{Q}) \quad (4)$$

The partial derivative of eq 4 with respect to \mathbf{Q} is

$$\mathbf{X} \mathbf{D}_1 \mathbf{X}^T (\mathbf{Q} - \mathbf{I}) \mathbf{W} \mathbf{W}^T + \lambda \mathbf{X} \mathbf{D}_2 \mathbf{X}^T \mathbf{Q} = 0 \quad (5)$$

Due to the constraint $\mathbf{W}^T \mathbf{W} = \mathbf{I}$, we multiply both sides of eq 5 by \mathbf{W} , we have

$$\mathbf{X} \mathbf{D}_1 \mathbf{X}^T (\mathbf{Q} - \mathbf{I}) \mathbf{W} + \lambda \mathbf{X} \mathbf{D}_2 \mathbf{X}^T \mathbf{Q} \mathbf{W} = 0 \quad (6)$$

Then, we get

$$\mathbf{Q} = (\mathbf{X} \mathbf{D}_1 \mathbf{X}^T + \lambda \mathbf{X} \mathbf{D}_2 \mathbf{X}^T)^{-1} \mathbf{X} \mathbf{D}_1 \mathbf{X}^T \quad (7)$$

From eq 7, we can see that \mathbf{Q} does not have a trivial solution; thus, there exists no constraint on \mathbf{Q} in the objective function.

Next, we fix \mathbf{Q} and take derivatives of eq 3 with respect to \mathbf{W} . First, we construct the Lagrangian multiplier

$$\min_{\mathbf{Q}, \mathbf{W}} \text{tr}(\mathbf{W}^T (\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{W}) + \lambda \text{tr}(\mathbf{Q}^T \mathbf{X} \mathbf{D}_2 (\mathbf{Q}^T \mathbf{X})^T) - \sum_{i=1}^l \Lambda_i (\mathbf{W}_i^T \mathbf{W}_i - 1) \quad (8)$$

Remove the second item, we have

$$\arg \min_{\mathbf{Q}, \mathbf{W}} \sum_{i=1}^l (\mathbf{W}_i^T (\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T \mathbf{W}_i) - \Lambda_i (\mathbf{W}_i^T \mathbf{W}_i - 1) \quad (9)$$

where \mathbf{W}_i is the i -th column of matrix \mathbf{W} .

Take the partial derivative of eq 8 with respect to \mathbf{W}_i , we have

$$((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T + ((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T)^T) \mathbf{W}_i - 2 \Lambda_i \mathbf{W}_i = 0 \quad (10)$$

$$((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T + ((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T)^T) \mathbf{W}_i = 2 \Lambda_i \mathbf{W}_i \quad (11)$$

Let $\mathbf{K} = (1/2)((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T + ((\mathbf{X} - \mathbf{Q}^T \mathbf{X}) \mathbf{D}_1 (\mathbf{X} - \mathbf{Q}^T \mathbf{X})^T)^T)$. By eq 11, we

get the solution of \mathbf{W} . The column vectors of the optimal projection matrix \mathbf{W} are composed of the eigenvectors of \mathbf{K} corresponding to the first l minimum eigenvalues. The column vectors of residual projection matrix \mathbf{W}_{res} are composed of the eigenvectors of \mathbf{K} corresponding to the later $(m-l)$ minimum eigenvalues.

3. FAULT DETECTION

3.1. Construction of Statistics. The data do not necessarily obey the Gaussian distribution. Given that a linear transformation does not change the distribution of random variables. In addition, the feature space does not necessarily exhibit a Gaussian distribution, and its covariance matrix is²³

$$\Sigma = (\mathbf{W}^T \mathbf{Q}^T \mathbf{X}) (\mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T \quad (12)$$

Similarly, the residual space does not necessarily obey the Gaussian distribution, and its covariance matrix is

$$\Sigma_{\text{res}} = (\mathbf{W}_{\text{res}}^T \mathbf{Q}^T \mathbf{X}) (\mathbf{W}_{\text{res}}^T \mathbf{Q}^T \mathbf{X})^T \quad (13)$$

The following statistics can be constructed for fault detection

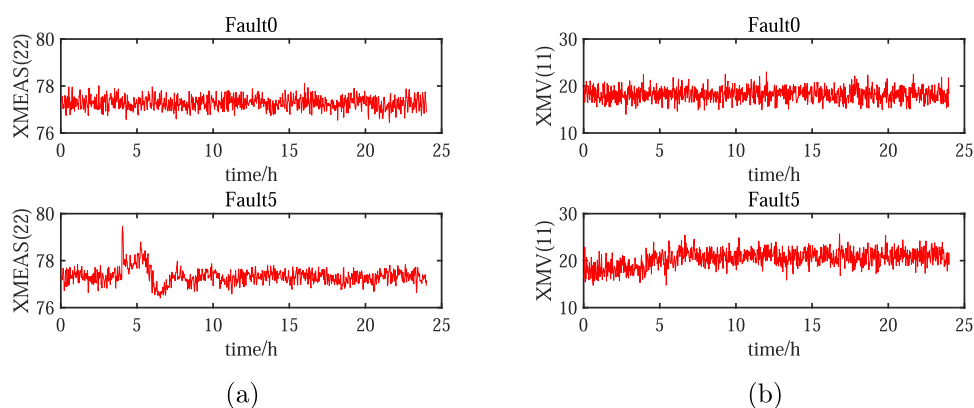


Figure 3. Fault 5. (a) Variable XMEAS 22: fault 0 and fault 5. (b) Variable XMV 11: fault 0 and fault 5.

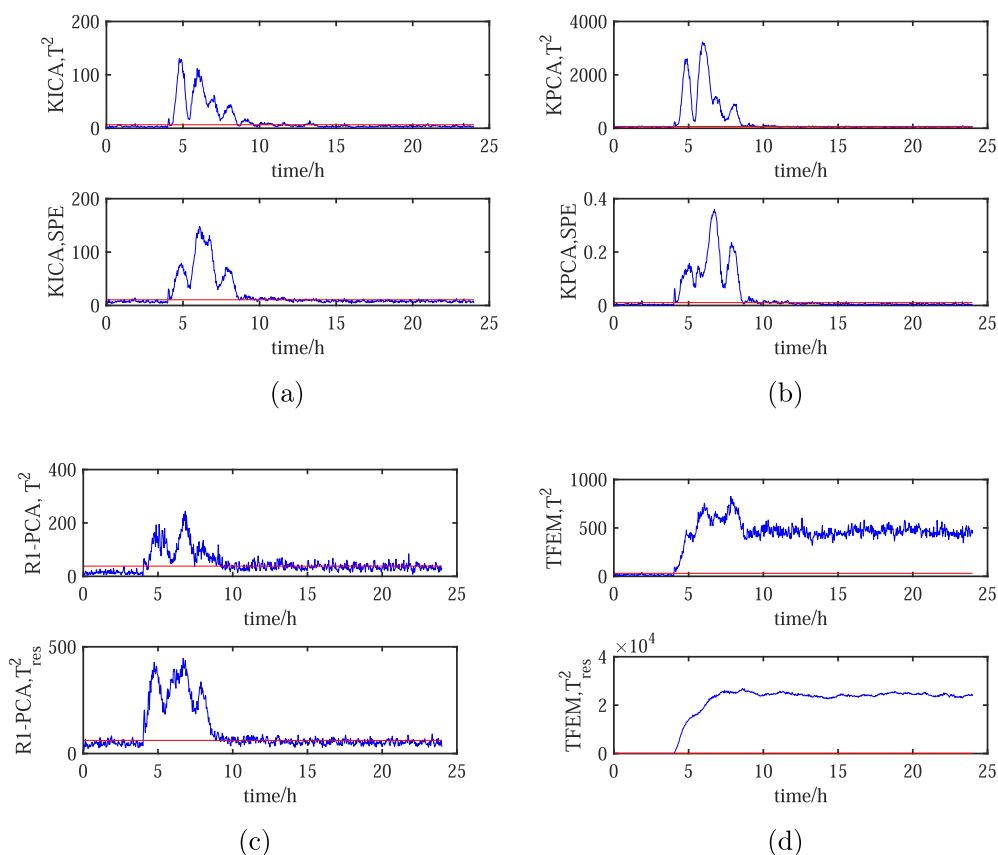


Figure 4. Detection results of fault 5. (a) Results for KICA. (b) Results for KPCA. (c) Results for R1-PCA. (d) Results for TFEM.

$$T = n \times (\mathbf{W}^T \mathbf{Q}^T \mathbf{X})^T \Sigma^{-1} (\mathbf{W}^T \mathbf{Q}^T \mathbf{X}) \quad (14)$$

$$T_{\text{res}} = n \times (\mathbf{W}_{\text{res}}^T \mathbf{Q}^T \mathbf{X})^T \Sigma_{\text{res}}^{-1} (\mathbf{W}_{\text{res}}^T \mathbf{Q}^T \mathbf{X}) \quad (15)$$

where \mathbf{x} is a column of matrix \mathbf{X} , representing a sample in data set \mathbf{X} .

We use the kernel density estimation (KDE) method in ref 24 to calculate the control limits J_{th} , $J_{\text{th,res}}$ of the T , and T_{res} , respectively.

3.2. Process of the TFEM Fault Detection Method.

Requirements:

- Training data set \mathbf{X} ;
- Testing data set \mathbf{X}_{new} ;
- Dimensionality of the feature space l ;
- Trade-off parameter λ .

Offline training

- Step 1 Initialize $\mathbf{W} \in \mathbf{R}^{m \times l}$ and $\mathbf{Q} \in \mathbf{R}^{m \times m}$;
- Step 2 from eq 7, we find \mathbf{Q} . From eq 11, we find \mathbf{W} ;
- Step 3 repeat Step 2 until converges, and we get \mathbf{W} and \mathbf{W}_{res} ;
- Step 4 Calculate T , T_{res} of training data set \mathbf{X} according to eqs 14 and 15;

- Step 5 Set control limit J_{th} , $J_{\text{th,res}}$ through the KDE method;

Online testing

- Step 1 Calculate T , T_{res} of testing data set \mathbf{X}_{new} according to eqs 14 and 15;
- Step 2 Perform fault detection based on the following detection logic:
 - $T > J_{\text{th}}$ or $T_{\text{res}} > J_{\text{th,res}}$, fault;
 - $T \leq J_{\text{th}}$ and $T_{\text{res}} \leq J_{\text{th,res}}$, normal;

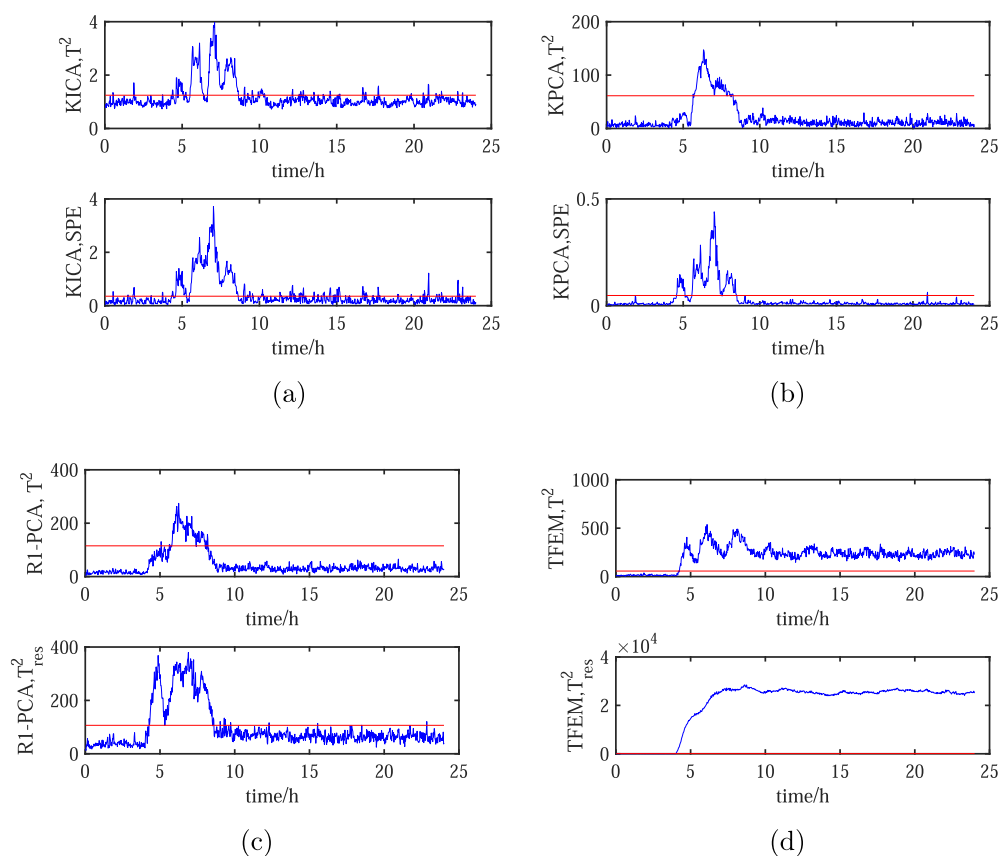


Figure 5. Detection results of fault 5 with Gaussian noise. (a) Results for KICA. (b) Results for KPCA. (c) Results for R1-PCA. (d) Results for TFEM.

3.3. TE Process Fault Detection. To evaluate the performance of the proposed fault detection method, we used it with the TE process data set and compared its performance with the R1-PCA,¹⁷ KICA,²⁵ and KPCA²⁶ methods.

3.4. Data Preparation. **3.4.1. TE Process Data.** TE process proposed by Downs et al. is a chemical simulative model based on the actual chemical production process, and it has been widely used to test the performance of fault detection and diagnosis. More details can be found in other works.²⁷ The TE process data set is composed of a training data set and a testing data set. The data in the TE process data set correspond to 22 different simulation operations. Each sample in the TE process data set has 52 process variables. Samples d00.dat to d21.dat constitute the training data set, while samples d00_te.dat to d21_te.dat make up the testing data set. Samples d00.dat and d00_te.dat are samples corresponding to normal working conditions. The d00.dat training sample is obtained from a 25 h simulation, and the total number of observations is 500. The d00_te.dat testing sample is obtained from a 48 h simulation, and the total number of observations is 960. d01.dat to d21.dat are training data set samples with faults, while d01_te.dat to d21_te.dat are testing data set samples with faults.

The testing data set samples with faults are obtained from a 48 h operations simulation, and the faults are introduced after 8 h. A total of 960 observed values are collected, of which the first 160 observed values are normal.

4. RESULTS

4.1. Case Study 1: Case Study of Fault 5. Fault 5 includes a step change in the inlet temperature of the cooling water in the condenser [(XMES(22))]. A significant effect of this fault is that it causes a step change in the flow of the cooling water in the condenser (see Figure 3a). When this fault occurs, the flow rate from the condenser outlet to the vapor/liquid separator [XMV(11)] also increases, resulting in an increase in the temperature of the vapor/liquid separator as well as that of the cooling water outlet of the separator (see Figure 3b). The control loop can compensate for this change and return the temperature in the separator to the set point. It takes approximately 10 h to reach the stable state. As for the remaining 52 monitored variables, 32 variables have a similar transition process, reaching stability after approximately 10 h.

We evaluate the proposed algorithm on the fault detection and compare it with KICA, KPCA, and R1-PCA. Moreover, we set λ , l as 1, 16. Figure 4 shows the FDR for the TFEM, KPCA, and KICA methods. The equation of FDR is shown as follows

$$\text{FDR} = \frac{\text{num}_n + \text{num}_f}{\text{num}_t} \quad (16)$$

where num_n is the number of successful detections in normal data, num_f is the number of successful detections in the fault data, and num_t is the number of all test data.

As can be seen from the Figure 4, the proposed TFEM method is superior to R1-PCA, KICA, and KPCA in fault 5 detection. In general, the total FDR of TFEM is 98.85%, while the KICA is 53.85%, the KPCA is 51.98%, and the R1-PCA is 69.27%. Moreover, the FDR of TFEM is 45% higher than that

Table 1. FDR (%) for the TE Process

| noise | without noise | | | | with Gaussian noise | | | | |
|---------|---------------|----------|----------|------------|---------------------|----------|----------|------------|----------|
| | fault | KICA (%) | KPCA (%) | R1-PCA (%) | TFEM (%) | KICA (%) | KPCA (%) | R1-PCA (%) | TFEM (%) |
| 1 | | 98.44 | 98.33 | 97.29 | 99.27 | 97.6 | 98.85 | 99.17 | 99.79 |
| 2 | | 96.88 | 97.92 | 97.60 | 98.44 | 97.29 | 97.29 | 98.33 | 98.96 |
| 3 | | 25.52 | 38.13 | 33.33 | 27.50 | 21.56 | 18.23 | 16.67 | 16.88 |
| 4 | | 97.81 | 97.92 | 97.81 | 98.85 | 24.27 | 17.81 | 31.77 | 100 |
| 5 | | 53.85 | 51.98 | 69.27 | 98.85 | 43.33 | 32.4 | 35.42 | 100 |
| 6 | | 99.17 | 99.48 | 99.38 | 99.79 | 99.48 | 99.69 | 100 | 100 |
| 7 | | 95.1 | 99.48 | 99.48 | 99.69 | 95.52 | 45.31 | 100 | 100 |
| 8 | | 94.48 | 97.92 | 96.98 | 97.5 | 93.54 | 86.25 | 97.29 | 98.13 |
| 9 | | 45.94 | 34.06 | 31.67 | 23.54 | 29.79 | 18.33 | 16.67 | 17.08 |
| 10 | | 81.15 | 81.56 | 81.04 | 92.08 | 61.67 | 26.77 | 28.02 | 88.54 |
| 11 | | 86.46 | 85.83 | 86.46 | 81.56 | 39.06 | 19.17 | 38.85 | 76.15 |
| 12 | | 95.1 | 96.35 | 97.50 | 98.85 | 86.56 | 78.54 | 95.1 | 99.79 |
| 13 | | 96.15 | 95.94 | 95.94 | 96.04 | 94.38 | 93.75 | 94.69 | 96.04 |
| 14 | | 94.48 | 98.54 | 98.02 | 99.06 | 29.69 | 17.81 | 77.81 | 100.00 |
| 15 | | 50.21 | 42.92 | 37.92 | 34.79 | 34.58 | 17.92 | 16.67 | 21.15 |
| 16 | | 76.77 | 76.67 | 83.85 | 92.4 | 35.21 | 20.1 | 58.54 | 90.42 |
| 17 | | 92.6 | 96.04 | 95.94 | 96.56 | 24.58 | 18.85 | 48.02 | 96.25 |
| 18 | | 92.50 | 92.29 | 91.98 | 92.19 | 89.79 | 88.02 | 90.21 | 91.88 |
| 19 | | 64.79 | 65.10 | 72.19 | 92.71 | 24.58 | 16.88 | 16.67 | 89.27 |
| 20 | | 82.60 | 82.29 | 76.67 | 92.40 | 30.31 | 18.02 | 42.08 | 91.77 |
| 21 | | 60.42 | 65.42 | 65.21 | 69.48 | 40.63 | 18.96 | 41.67 | 61.88 |
| average | | 80.02 | 80.67 | 81.32 | 84.84 | 56.83 | 45.19 | 59.22 | 82.57 |

of KICA, 46.87% higher than that of KICA, and 29.58% higher than that of R1-PCA. Thus, the fault detection performance of TFEM is almost twice as high as KICA and KPCA. Because most of the process variables in fault 5 have not changed, and a few of the process variables have changed very slightly, KICA and KPCA usually dilute the small changes arising during the dimensionality reduction projection process, while TFEM adopts the nonreduced dimensionality projection to highlight the data features. When the data change slightly, the nonreduced dimensionality projection will keep these small changes in some particular sensors. As a result, the TFEM method is able to detect small changes in fault 5. In addition, for both the feature space and the residual space, the TFEM method uses the T^2 statistic, which reflects both the relationship between process variables and the Euclidean distance between data. Hence, the detection performance is better.

It can be seen from Figure 4 that, after the fault leveled off, the R1-PCA, KICA, and KPCA methods are unable to detect it. This is because of the reason mentioned above. After the fault curve becomes smooth, the fault data exhibit only small changes. In this case, the R1-PCA, KICA, and KPCA methods usually cannot track the small changes that occur during the dimensionality reduction projection process. However, the TFEM method can make the process variables unrelated to the fault sparse in the process of nonreduced dimensionality projection. Therefore, these small changes in some process variables are more obvious. In the case of the R1-PCA, KICA, and KPCA methods, the data pass through a single projection, wherein the small changes are removed together. In contrast, the TFEM method is more sensitive and is able to detect even small changes. This is the reason the detection performance of this method is better.

We selected 12 samples in the original TE data to add Gaussian noise to verify the robustness of our algorithm. Moreover, we set λ , l as 0.5, 35. The detection results are

shown in Figure 5, and the proposed algorithm has the highest detection rate. The detection result of the R1-PCA algorithm is slightly better than that of KICA and KPCA because R1-PCA uses the l_{21} -norm as a metric, which has a certain robustness. The detection rate of R1-PCA is still not as high as that of TFEM because TFEM uses two projections and can remove interfering features in the raw data.

Case study 1 confirmed the effectiveness of the proposed fault detection method based on TFEM and showed that the method is more suitable for fault detection than the R1-PCA, KICA, and KPCA methods investigated.

4.2. Case Study 2: Case Study of All Faults. In this case study, all 21 TE process faults are used to further evaluate the performances of the R1-PCA, KICA, KPCA, and TFEM methods (see Table 1). We divided the experiments into two groups, those with and without noise. We selected 12 samples in the original TE data for the training data set to add Gaussian noise to verify the robustness of our algorithm. As shown in Table 1, a comparison of the FDR of the 21 faults for the R1-PCA, KICA, KPCA, and TFEM methods confirmed that the TFEM method has the higher FDR for most of the faults. For example, in the case of fault 19, the FDR of the TFEM method is 92.71%, while the KICA is 64.79%, KPCA is 65.10%, and R1-PCA is 72.19%. Thus, the proposed TFEM method is highly suited for fault detection.

It can also be seen from Table 1 that the FDR of the proposed method for most faults are higher than 90%. Thus, the stability of the TFEM method is higher owing to the nonreduced dimensionality projection process, which retains the small changes in data. The T^2 statistics is used for the residual space, which can simultaneously measure the Euclidean distance between data and reflect the relationship between process variables. Thus, the fact that the residual space can be detected more accurately in the case of the 21 faults further highlights the superiority of the TFEM method in fault detection. We find an interesting phenomenon that the

detection results of the TFME method with noise are better than those without noise. We have done multiple experiments and the results are still the same.

4.3. Penicillin Fermentation Process Fault Detection.

4.3.1. Data Preparation. The production process of penicillin is a typical nonlinear, multi-modal production process. The fermentation process can be divided into three stages: the stage of rapid growth of the cells, the stage of cell synthesis of penicillin, and the stage of cell autolysis. Based on the Pensim simulation platform, this section verifies the effectiveness of the fault detection method based on TFEM.

Pensim simulation platform has four controlled variables that can control the changes of fermentation process parameters, four manipulated variables, two inputs and generated heat variables, and six outputs (state variables). Twelve variables are selected. The selection of variables is shown in Table 2. The data are generated using the Pensim

Table 2. Selection of Variables in the Penicillin Fermentation Process

| serial number | variable | variable range | variable type |
|---------------|--------------------------------|----------------|---------------------------|
| 1 | acid flow rate (mL/h) | 0–0.01 | manipulated variable |
| 2 | cold water flow rate (L/h) | 0–150 | manipulated variable |
| 3 | base flow rate(ml/h) | 0–0.2 | manipulated variable |
| 4 | aeration rate (L/h) | 8.58–8.68 | controlled variable |
| 5 | substrate feed rate (L/h) | 0–0.05 | controlled variable |
| 6 | agitator power (W) | 0–30.5 | controlled variable |
| 7 | pH | 4.9–5.4 | inputs and generated heat |
| 8 | temperature (K) | 297.5–298.5 | inputs and generated heat |
| 9 | substrate conc (g/L) | 0–15 | outputs (state variables) |
| 10 | culture vol (L) | 95–105 | outputs (state variables) |
| 11 | Penicillin conc (g/L) | 0–1.5 | outputs (state variables) |
| 12 | CO ₂ conc (mmole/L) | 0–3 | outputs (state variables) |

simulation platform, the simulation time is set to 400 h, and the sampling time was set to 1 h. The training data set is generated by adjusting the initial variable values under normal conditions, and the data set includes 400 samples.

Pensim simulation platform can introduce a disturbance to the first three variables (aeration rate, stirring power, and substrate flow rate). There are two types of disturbances: step and ramp. Moreover, the amplitude, introduction time, and termination time of the two disturbances can be further set. In order to test the effectiveness of the method, this chapter produces three fault batches in the experiment, and the fault types and amplitudes of the three fault batches are shown in Table 2. We evaluate the proposed algorithm on the fault detection and compare it with KICA, KPCA, and R1-PCA. Also, we set λ , l as 1, 5 (Table 3).

5. RESULTS

From Figure 6, it can be seen that the proposed TFEM method outperforms R1-PCA, KICA, and KPCA in the detection of fault 2. Overall, the total FDR is 90.00% for TFEM, 87.75% for KICA, 89.75% for KPCA, and 88.00% for

Table 3. Three Batch Faults Set During Penicillin Fermentation

| fault batch | variable | fault type | amplitude | introduction time | termination time |
|-------------|----------|------------|-----------|-------------------|------------------|
| f1 | 1 | step | 0.3 | 101 | 400 |
| f2 | 2 | ramp | 3 | 101 | 400 |
| f3 | 3 | ramp | 2 | 101 | 400 |

R1-PCA. The TFEM algorithm finds faults after about 150 samples, while KICA and R1-PCA both find faults later than the TFEM method. Although the detection rate of KPCA is relatively high, the detection rate of T^2 in KPCA is very low. The proposed algorithm uses two projections, which can ensure that tiny changes in data can detect faults in time. In addition, for both feature space and residual space, the TFEM method uses the T^2 statistic, which reflects both the relationship between the process variables and the Euclidean distance between the data.

It can also be seen from Table 4 that the proposed method has the highest FDR for most faults. Therefore, the stability of the TFEM method is higher. As mentioned above, this is due to the nonreduced dimensionality projection process, which preserves small changes in the data. The residual space uses the T^2 statistic, which can detect the residual space more accurately.

To verify the robustness of the proposed algorithm, the Gaussian noise is added to the train data set. The Gaussian noise is added into the data with 1.1 density. Twelve samples are randomly selected to add Gaussian noise to the training data set. The compared algorithms are KICA, KPCA, and R1-PCA. The experimental results are shown in Table 5.

As can be seen from Table 5, the comprehensive results of the proposed algorithm are the best. The results of R1-PCA are better than KPCA and KICA, which is because R1-PCA is a robust algorithm. R1-PCA only uses a single projection and cannot detect small changes in the data. Therefore, the detection rate of the R1-PCA algorithm is lower than that of TFEM. The detection result of KICA is the worst. We adjusted the kernel parameters many times, but the result is still not ideal. The results of KPCA are slightly better than KICA. The kernel parameters of KPCA have also been adjusted many times, and the result is still lower than the detection rate of the robust method.

The above experiments show that the proposed algorithm has a certain robustness, and can still have an ideal detection effect in the absence of outliers.

6. CONCLUSIONS

In this paper, we propose and evaluate a TFEM, which uses nonreduced dimensionality projection to make the process variables unrelated to the fault sparse and the reduced dimensionality projection to highlight the features of data. As a result, l_{21} -norm-based nonreduced dimensionality projection ensures the small changes in some process variables are more obvious, and l_{21} -norm-based reduced dimensionality projection enhances the robustness of the algorithm. The suitability of the proposed method is verified by TE and Penicillin fermentation processes data set. In order to further improve the fault detection performance of data, the TFEM method can be combined with kernel techniques. Moreover, the proposed TFEM method can be extended to nonlinear processes. This is something we plan to explore in future studies.

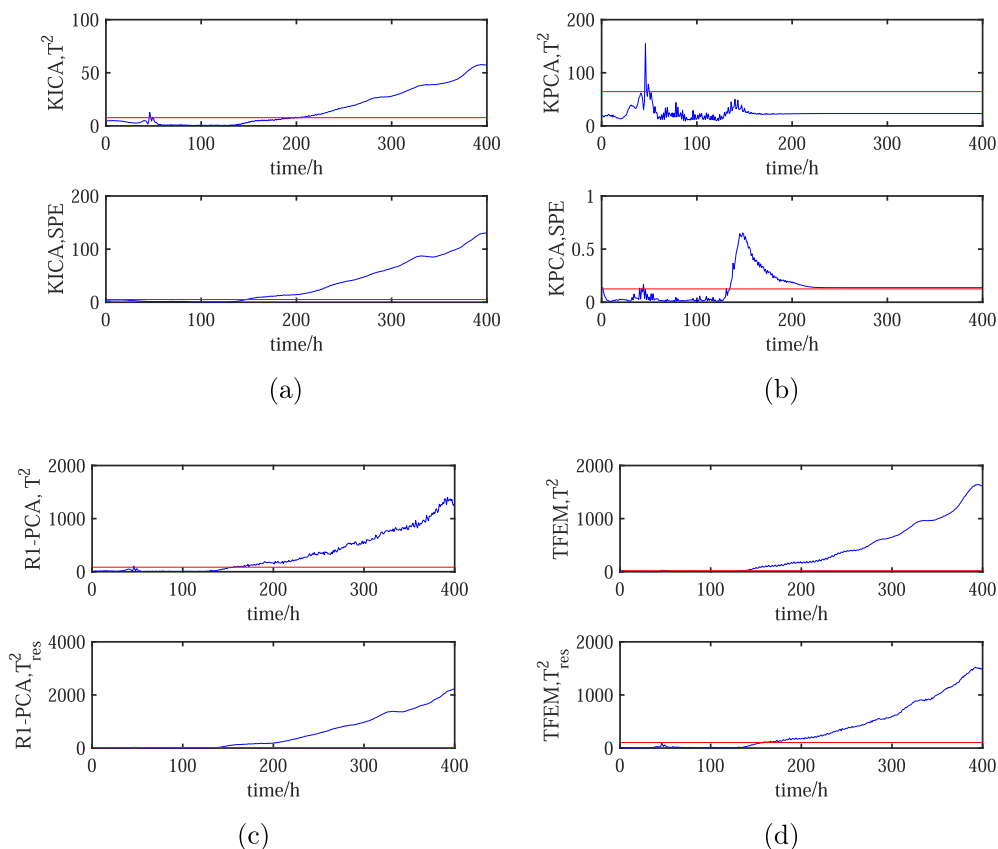


Figure 6. Detection results of fault 2. (a) Results for KICA. (b) Results for KPCA. (c) Results for R1-PCA. (d) Results for TFEM.

Table 4. Simulation Results of the Penicillin Fermentation Process

| fault | KICA (%) | KPCA (%) | R1-PCA (%) | TFEM (%) |
|-------|----------|--------------|------------|--------------|
| f1 | 62.75 | 98.00 | 97.25 | 99.00 |
| f2 | 87.75 | 89.75 | 88.00 | 90.00 |
| f3 | 92.75 | 96.50 | 94.75 | 96.50 |

Table 5. Simulation Results of Penicillin Fermentation Process with Gaussian Noise

| fault | KICA (%) | KPCA (%) | R1-PCA (%) | TFEM (%) |
|-------|----------|----------|--------------|--------------|
| f1 | 25.00 | 21.25 | 95.25 | 97.50 |
| f2 | 25.00 | 82.00 | 86.75 | 87.75 |
| f3 | 90.50 | 87.00 | 96.75 | 95.50 |

APPENDIX

6.1. Convergence Analysis

During the optimization of \mathbf{W} and \mathbf{Q} in each iteration, the value of the objective function decreases monotonically until it converges to the optimal solution. Therefore, we have

$$\begin{aligned} J(\mathbf{Q}^{(t)}, \mathbf{W}^{(t)}, \mathbf{D}_1^{(t-1)}, \mathbf{D}_2^{(t-1)}) \\ \leq J(\mathbf{Q}^{(t-1)}, \mathbf{W}^{(t-1)}, \mathbf{D}_1^{(t-1)}, \mathbf{D}_2^{(t-1)}) \end{aligned} \quad (17)$$

Then

$$\begin{aligned} & \text{tr}((\mathbf{W}^{(t)\top} \mathbf{X} - \mathbf{W}^{(t)\top} \mathbf{Q}^{(t)\top} \mathbf{X}) \mathbf{D}_1^{(t-1)} (\mathbf{W}^{(t)\top} \mathbf{X} \\ & \quad - \mathbf{W}^{(t)\top} \mathbf{Q}^{(t)\top} \mathbf{X})^\top) + \lambda \text{tr}(\mathbf{Q}^{(t)\top} \mathbf{X} \mathbf{D}_2^{(t-1)} (\mathbf{Q}^{(t)\top} \mathbf{X})^\top) \\ & \leq \text{tr}((\mathbf{W}^{(t-1)\top} \mathbf{X} - \mathbf{W}^{(t-1)\top} \mathbf{Q}^{(t-1)\top} \mathbf{X}) \mathbf{D}_1^{(t-1)} \\ & \quad (\mathbf{W}^{(t-1)\top} \mathbf{X} - \mathbf{W}^{(t-1)\top} \mathbf{Q}^{(t-1)\top} \mathbf{X})^\top) \\ & \quad + \lambda \text{tr}(\mathbf{Q}^{(t-1)\top} \mathbf{X} \mathbf{D}_2^{(t-1)} (\mathbf{Q}^{(t-1)\top} \mathbf{X})^\top) \end{aligned} \quad (18)$$

donate $\mathbf{W}^\top \mathbf{X} - \mathbf{W}^\top \mathbf{Q}^\top \mathbf{X} = \mathbf{U}$, $\mathbf{Q}^\top \mathbf{X} = \mathbf{V}$

$$\begin{aligned} & \sum_{i=1}^n \frac{\|\mathbf{U}_i^{(t+1)}\|_2^2}{\|\mathbf{U}_i^{(t)}\|_2^2} + \lambda \sum_{i=1}^m \frac{\|\mathbf{V}_i^{(t+1)}\|_2^2}{\|\mathbf{V}_i^{(t)}\|_2^2} \\ & \leq \sum_{i=1}^n \frac{\|\mathbf{U}_i^{(t)}\|_2^2}{\|\mathbf{U}_i^{(t)}\|_2^2} + \lambda \sum_{i=1}^m \frac{\|\mathbf{V}_i^{(t)}\|_2^2}{\|\mathbf{V}_i^{(t)}\|_2^2} \end{aligned} \quad (19)$$

where \mathbf{U}_i is the i -th row of \mathbf{U} and \mathbf{V}_i is the i -th row of \mathbf{V}

Multiply both sides by p , $p > 0$

$$\begin{aligned} & p \sum_{i=1}^n \frac{\|\mathbf{U}_i^{(t+1)}\|_2^2}{\|\mathbf{U}_i^{(t)}\|_2^2} + p\lambda \sum_{i=1}^m \frac{\|\mathbf{V}_i^{(t+1)}\|_2^2}{\|\mathbf{V}_i^{(t)}\|_2^2} \\ & \leq p \sum_{i=1}^n \frac{\|\mathbf{U}_i^{(t)}\|_2^2}{\|\mathbf{U}_i^{(t)}\|_2^2} + p\lambda \sum_{i=1}^m \frac{\|\mathbf{V}_i^{(t)}\|_2^2}{\|\mathbf{V}_i^{(t)}\|_2^2} \end{aligned} \quad (20)$$

our goal is to prove that

$$\begin{aligned} & \sum_{i=1}^n \|\mathbf{u}_i^{(t+1)}\|_2 + \sum_{i=1}^m \|\mathbf{v}_i^{(t+1)}\|_2 \\ & \leq \sum_{i=1}^n \|\mathbf{u}_i^{(t)}\|_2 + \sum_{i=1}^m \|\mathbf{v}_i^{(t)}\|_2 \end{aligned} \quad (21)$$

In order to get the above formula, we have to change the goal to

$$\begin{aligned} & \sum_{i=1}^n \|\mathbf{u}_i^{(t+1)}\|_2 + \sum_{i=1}^m \|\mathbf{v}_i^{(t+1)}\|_2 + p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} + p \\ & \quad \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} \\ & \leq \sum_{i=1}^n \|\mathbf{u}_i^{(t)}\|_2 + \sum_{i=1}^m \|\mathbf{v}_i^{(t)}\|_2 + p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t+1)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} \\ & \quad + p \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t+1)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} \end{aligned} \quad (22)$$

That is, we change aim to prove the following formula

$$\begin{aligned} & \sum_{i=1}^n \|\mathbf{u}_i^{(t+1)}\|_2 - p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t+1)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} \\ & \quad - \sum_{i=1}^n \|\mathbf{u}_i^{(t)}\|_2 + p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} + \lambda \sum_{i=1}^m \|\mathbf{v}_i^{(t+1)}\|_2 \\ & \quad + p \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} - \lambda \sum_{i=1}^m \|\mathbf{v}_i^{(t)}\|_2 - p \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t+1)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} \\ & \leq 0 \end{aligned} \quad (23)$$

When $p = 1/2$, for any $\|\mathbf{u}_i^{(t+1)}\|_2$, $\|\mathbf{u}_i^{(t)}\|_2$, we always have¹⁹

$$\begin{aligned} & \sum_{i=1}^n \|\mathbf{u}_i^{(t+1)}\|_2 - p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t+1)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} - \sum_{i=1}^n \|\mathbf{u}_i^{(t)}\|_2 \\ & \quad + p \sum_{i=1}^n \frac{\|\mathbf{u}_i^{(t)}\|_2^2}{\|\mathbf{u}_i^{(t)}\|_2} \\ & \leq 0 \end{aligned} \quad (24)$$

Similarly, we have

$$\begin{aligned} & \lambda \sum_{i=1}^m \|\mathbf{v}_i^{(t+1)}\|_2 + p \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t+1)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} - \lambda \sum_{i=1}^m \|\mathbf{v}_i^{(t)}\|_2 \\ & \quad - p \lambda \sum_{i=1}^m \frac{\|\mathbf{v}_i^{(t+1)}\|_2^2}{\|\mathbf{v}_i^{(t)}\|_2} \\ & \leq 0 \end{aligned} \quad (25)$$

By combining eqs 24 and 25, we can get formula eq 23, and convergence is proved.

AUTHOR INFORMATION

Corresponding Authors

Xin Sha – The College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; orcid.org/0000-0003-2167-3127; Email: xinsha123@163.com

Naizhe Diao – The College of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; Email: diaonaizhe@163.com

Complete contact information is available at:

<https://pubs.acs.org/10.1021/acsomega.2c03295>

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

Thanks to the reviewers and editors for their support of this paper.

REFERENCES

- (1) Kopbayev, A.; Khan, F.; Yang, M.; Halim, S. Z. Gas leakage detection using spatial and temporal neural network model. *Process Saf. Environ. Prot.* **2022**, *160*, 968–975.
- (2) Xue, C.; Zhang, T.; Xiao, D. Output-related and -unrelated fault monitoring with an improvement prototype knockoff filter and feature selection based on laplacian eigen maps and sparse regression. *ACS Omega* **2021**, *6*, 10828–10839.
- (3) Arunthavanathan, R.; Khan, F.; Ahmed, S.; Imtiaz, S. A deep learning model for process fault prognosis. *Process Saf. Environ. Prot.* **2021**, *154*, 467–479.
- (4) Jiang, Q.; Yan, X. Parallel PCA-KPCA for nonlinear process monitoring. *Control Eng. Pract.* **2018**, *80*, 17–25.
- (5) Tian, Y.; Yao, H.; Li, Z. Plant-wide process monitoring by using weighted copula-Correlation based multiblock principal component analysis approach and online-horizon Bayesian method. *ISA Trans.* **2020**, *96*, 24–36.
- (6) Zhang, K.; Peng, K.; Zhao, S.; Wang, F. A Novel Feature Extraction-Based Process Monitoring Method for Multimode Processes with Common Features and Its Applications to a Rolling Process. *IEEE Trans. Ind. Inf.* **2020**, *99*, 1.
- (7) Ye, F.; Guo, Y.; Xia, Z.; Zhang, Z.; Zhou, Y. Feature extraction and process monitoring of multi-channel data in a forging process via sensor fusion. *Int. J. Comput. Integrated Manuf.* **2021**, *34*, 95–109.
- (8) Zhang, N.; Xu, Y.; Zhu, Q. X.; He, Y. L. Improved Locality Preserving Projections Based on Heat-Kernel and Cosine Weights for Fault Classification in Complex Industrial Processes. *IEEE Trans. Reliab.* **2021**, 1–10.
- (9) Torre, F. D. L.; Black, M. J. A Framework for Robust Subspace Learning. *Int. J. Comput. Vis.* **2003**, *54*, 117–142.
- (10) Gao, Q.; Gao, F.; Zhang, H.; Hao, X.; Wang, X. Two-Dimensional Maximum Local Variation Based on Image Euclidean Distance for Face Recognition. *IEEE Trans. Image Process.* **2013**, *22*, 3807–3817.
- (11) He, R.; Hu, B. G.; Zheng, W. S.; Kong, X. W. Robust Principal Component Analysis Based on Maximum Correntropy Criterion. *IEEE Trans. Image Process.* **2011**, *20*, 1485.
- (12) Rahoma, A.; Imtiaz, S.; Ahmed, S. A new criterion for selection of non-zero loadings for sparse principal component analysis (SPCA). *Can. J. Chem. Eng.* **2021**, *99*, S356.
- (13) Xiao, Z.; Wang, H.; Zhou, J. Robust dynamic process monitoring based on sparse representation preserving embedding. *J. Process Control* **2016**, *40*, 119–133.
- (14) Xu, X.; Ding, J. Similarity and sparsity collaborative embedding and its application to robust process monitoring. *Control Eng. Pract.* **2022**, *122*, 105113.

- (15) David, L. For most large underdetermined systems of linear equations the minimal L1-norm solution is also the sparsest solution. *IEEE Trans. Image Process.* **2006**.
- (16) Natarajan, B. K. Sparse Approximate Solutions to Linear Systems. *SIAM J. Comput.* **1995**, *24*, 227–234.
- (17) Ding, C.; Zhou, D.; He, X.; Zha, H. R1-PCA: rotational invariant L1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd international conference on Machine learning (ICML '06)*; ACM, 2006.
- (18) Wang, R.; Nie, F. P.; Yang, X. J.; Gao, F. F.; Yao, M. Robust 2DPCA With Non-greedy L1-Norm Maximization for Image Analysis. *IEEE Trans. Cybern.* **2015**, *45*, 1108–1112.
- (19) Nie, F. P.; Huang, H.; Cai, X.; Ding, C. Efficient and Robust Feature Selection via Joint L2, 1-Norms Minimization. *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems*; Neural Information Processing Systems, 2010.
- (20) Pan, Y.; Yang, C.; An, R.; Sun, Y. Fault detection with improved principal component pursuit method. *Chemometric Intell. Lab. Syst.* **2016**, *157*, 111–119.
- (21) Pan, Y.; Yang, C.; An, R.; Sun, Y. Robust principal component pursuit for fault detection in a blast furnace process. *Ind. Eng. Chem. Res.* **2018**, *57*, 283–291.
- (22) Fu, Y.; Luo, C.; Bi, Z. Low-Rank Joint Embedding and Its Application for Robust Process Monitoring. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13.
- (23) Chen, Z. W. Data-Driven Fault Detection for Industrial Processes. *Journal of Process Control*; Springer, 2017.
- (24) Odiwei, P. E. P.; Cao, Y. Nonlinear Dynamic Process Monitoring Using Canonical Variate Analysis and Kernel Density Estimations. *IEEE Trans. Ind. Inf.* **2010**, *6*, 36–45.
- (25) Zhang, Y. W. Fault Detection and Diagnosis of Nonlinear Processes Using Improved Kernel Independent Component Analysis (KICA) and Support Vector Machine (SVM). *Ind. Eng. Chem. Res.* **2008**, *47*, 6961–6971.
- (26) Wang, Y.; Yu, H.; Li, X. Efficient iterative dynamic kernel principal component analysis monitoring method for the batch process with super-large-scale data sets. *ACS Omega* **2021**, *6*, 9989–9997.
- (27) Ricker, N. L. Optimal steady-state operation of the Tennessee Eastman challenge process. *Comput. Chem. Eng.* **1995**, *19*, 949–959.