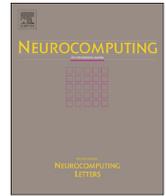




ELSEVIER

Contents lists available at ScienceDirect

## Neurocomputing

journal homepage: [www.elsevier.com/locate/neucom](http://www.elsevier.com/locate/neucom)

# Noise-resistant joint diagonalization independent component analysis based process fault detection <sup>☆</sup>



Xuemin Tian <sup>a</sup>, Lianfang Cai <sup>a</sup>, Sheng Chen <sup>b,c,\*</sup>

<sup>a</sup> College of Information and Control Engineering, China University of Petroleum, Qingdao 266580, China

<sup>b</sup> Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

<sup>c</sup> Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia

## ARTICLE INFO

### Article history:

Received 11 February 2014

Received in revised form

27 May 2014

Accepted 4 August 2014

Communicated by J. Zhang

Available online 15 August 2014

### Keywords:

Fault detection

Independent component analysis

Mixing matrix

Measurement noise

Time-delayed covariance matrices

Kurtosis

## ABSTRACT

Fast independent component analysis (FastICA) is an efficient feature extraction tool widely used for process fault detection. However, the conventional FastICA-based fault detection method does not consider the ubiquitous measurement noise and may exhibit unsatisfactory performance under the adverse effects of the measurement noise. To solve this problem, we propose a new process fault detection method based on noise-resistant joint diagonalization independent component analysis (NRJDICA), which explicitly takes the measurement noise into consideration. Specifically, the NRJDICA algorithm is developed to estimate the mixing matrix and the independent components (ICs) by whitening the measured variables and performing the joint diagonalization of the whitened variables' time-delayed covariance matrices. The relationships between the kurtosis statistics of the ICs and the fourth-order cross cumulant statistics of the measured variables are then derived based on the estimated mixing matrix to help sorting the estimated ICs and selecting the dominant ICs. The serial correlation information of each dominant IC is next estimated by using a moving window technique, based on which a monitoring statistic is constructed to conduct fault detection. The simulation studies using a three-variable system and a continuous stirred tank reactor show that the proposed method has superior fault detection performance over the traditional FastICA-based fault detection.

© 2014 Elsevier B.V. All rights reserved.

## 1. Introduction

With the advancement in measurement technology and distributed control systems, modern industrial processes have become increasingly more complex. To ensure process safety and stability as well as to maintain high quality of final products, reliable and timely fault detection has emerged as an essential task. Due to the convenient availability of substantial measured data in industrial plants, multivariate statistical analysis methods, which can extract meaningful feature information from large amounts of the measured data for detecting various faults or abnormal situations of industrial processes, have attracted much attention from both process engineers and academic researchers [1–18]. Principal component analysis (PCA), as one of the classical

multivariate statistical analysis approaches, has found wide-ranging applications in the fault detection field [11–16,19–22]. PCA projects the high-dimensional and correlated measured variables onto a smaller set of the uncorrelated latent variables called the principal components (PCs) that retain most of the original variance. However, PCA can only utilize the second-order zero-delayed covariance information and it cannot take the meaningful time-delayed covariance statistical information or the higher-order statistical information of the measured data into consideration [23–25], which may lead to insufficient feature extraction and unsatisfactory fault detection performance. Moreover, fault detection based on PCA assumes that the measured data follow a multivariate Gaussian distribution, in order to derive the control limits of Hotelling's T-squared ( $T^2$ ) and the companion squared prediction error (SPE) monitoring statistics. In practice, industrial process data usually obey non-Gaussian distribution due to process nonlinearity, operating condition shifts or other reasons [26,27], and the control limits derived based on the Gaussian distribution assumption may be ill-suited for fault detection purpose as the resulting fault indication may be biased.

More recently, fault detection based on independent component analysis (ICA) has become a hot topic [2–6,8,10,23–40]. ICA was originally derived for solving the blind source separation

<sup>☆</sup>This work is supported by the National Natural Science Foundation (NNSF) of China (61273160), the NNSF of Shandong Province of China (ZR2011FM014), the Doctoral Fund of Shandong Province (BS2012ZZ011), the Fundamental Research Funds for the Central Universities (14CX02174A), and the Postgraduate Innovation Funds of China University of Petroleum (CX2013060).

\* Corresponding author at: Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK.

E-mail addresses: [tianxm@upc.edu.cn](mailto:tianxm@upc.edu.cn) (X. Tian), [cailianfang@163.com](mailto:cailianfang@163.com) (L. Cai), [sqc@ecs.soton.ac.uk](mailto:sqc@ecs.soton.ac.uk) (S. Chen).

problem [41–48] and was introduced to fault detection by Kano et al. [29]. As a multivariate statistical analysis method, ICA can exploit the higher-order statistical information [41–45] or the second-order time-delayed covariance statistical information [32,47] to extract mutually independent latent variables called independent components (ICs) from the measured non-Gaussian variables, and it can be regarded as a useful extension of PCA. ICA is especially suitable for process fault detection as real-world processes are usually non-Gaussian [23–27]. Different criteria have been considered to develop various ICA algorithms [41], including the maximization of the non-Gaussian measures, such as negentropy, the minimization of the mutual information and the maximum likelihood estimation. Among the existing ICA algorithms, the fast ICA algorithm (FastICA) [41] based on the maximum negentropy criterion is widely used in the ICA-based fault detection methods because of its fast convergence rate and good non-Gaussian feature extraction ability [34,35]. Lee et al. [25] developed a FastICA-based fault detection method by utilizing the FastICA to extract the non-Gaussian ICs from the measured data and constructing three elliptical-type monitoring statistics, known as  $I^2$ ,  $I_c^2$  and SPE. Hsu et al. [36] argued that  $I^2$ ,  $I_c^2$  and SPE may not always appropriately capture the characteristics of the extracted ICs by the FastICA because of the ICs' skewed distributions, and developed a rectangular-type monitoring statistic named the adjusted outlyingness to monitor non-Gaussian processes. After concluding that both the elliptical-type and rectangular-type monitoring statistics may not always accurately estimate the nonlinear feature space boundary of normal operating condition (NOC), Lee et al. [37] constructed a monitoring statistic by using a local outlier factor method on the ICs extracted by the FastICA, which can effectively determine the nonlinear decision boundary of NOC. Zhao et al. [6] combined the ideas of partial least squares (PLS) and FastICA under the same mathematical umbrella by constructing a dual-objective optimization criterion that can consider higher-order statistical independence and quality-related requirements simultaneously. The modified independent component regression method proposed in [6] can extract the latent variables, which have close correlation with the quality properties and are more comprehensible for regression modeling.

Furthermore, by considering different process characteristics, such as nonlinearity, dynamic or multi-modality, researchers have proposed various improved fault detection methods based on the FastICA. In particular, Stefatos and Hamza [38] developed a dynamic ICA method for dynamic process fault detection by augmenting the original measured data with the previous observations and applying the FastICA to extract the ICs from the augmented data. Odiwei and Cao [39] integrated the canonical variate analysis with the FastICA and developed a state-space ICA based fault detection method for dynamic processes. Tian et al. [34] proposed a multiway kernel FastICA method based on feature samples for monitoring nonlinear batch processes. Cai et al. [33] presented a nonlinear process fault detection method by integrating the kernel FastICA with a newly emerging manifold learning method known as locality preserving projection. Basically, kernel FastICA integrates kernel PCA (KPCA) with FastICA [2,3,24,34], and thus it combines the advantages of both KPCA [22] and FastICA [41]. In other words, kernel FastICA possesses the unique capability of data processing which KPCA alone does not have. More specifically, kernel FastICA can indirectly excavate the second-order time structure information of time-series data by utilizing the higher-order information to extract mutually independent kernel ICs. This is important because the second-order time structure information, such as the time-delayed covariance information, can be utilized as a viable alternative for the higher-order information [45]. By contrast, KPCA is blind to the second-order time structure information, and can only utilize the second-order

zero-delayed covariance information to extract kernel PCs which are only uncorrelated but not independent. Therefore, kernel FastICA is more appropriate for time-series data than KPCA. Zhang [2] also pointed out that ICs can reveal more dynamic information from the measured data than PCs. To account for the process multimodal characteristics, Rashid and Yu [28] proposed a hidden Markov model based adaptive FastICA approach for monitoring non-Gaussian processes with multimodality. Zhang et al. [3] developed a multimodal process fault detection method based on the Kronecker product and modified kernel FastICA. In the above studies, the FastICA has been employed to extract non-Gaussian ICs for fault detection. However, this widely used FastICA is basically a “noise-free” algorithm, which adopts the noise-free ICA model and does not explicitly take the influence of the measurement noise into account. In reality, the measurement noise corruption always exists in industrial processes, as pointed out by Ge and Song [12] and Wang [49]. Under the adverse effects of measurement noise, the FastICA may not conduct effective and reliable feature extraction from the measured data. Moreover, the traditionally used monitoring statistics are also vulnerable to the measurement noise. These limitations may degrade the performance of the FastICA-based fault detection methods drastically.

Recently, researchers have begun to focus on the challenging problem of how to extract more accurately the features from the measured noisy data for fault detection. Kim and Lee [16] extended the conventional PCA to the probabilistic PCA (PPCA) for process monitoring, which considers the noise information of the measured data. Zhu et al. [14] further proposed the robust mixture PPCA for process monitoring. Compared to the PPCA, the robust mixture PPCA can reduce the negative effect of outliers and deal with the missing data problem more effectively. However, PPCA based techniques are only suitable for linear processes. By adopting a kernel technique, Ge and Song [12] proposed the kernel generalization of PPCA to extract nonlinear features for monitoring nonlinear processes. To meet the critical demand of identifying fault variables that contribute to process faults, Chen and Sun [20] proposed a probabilistic contribution analysis method in conjunction with the PPCA model based on the concept of missing variable. Noting that the PPCA requires the same noise level in all the measured variables, Kim et al. [17] utilized the factor analysis (FA), which can be regarded as an extension of PPCA and is capable of dealing with practical situations where the noise levels are different in different measured variables. Considering that fault information may not have a definite mapping relationship to a single factor and the useful information captured by some factors may be submerged in the insignificant information of the other factors, Jiang and Yan [18] further proposed the weighted FA (WFA) for process monitoring, which is capable of highlighting the useful information that is submerged in the insignificant information by suppressing the latter. These existing works have led to some successes in practical applications, but there still exist many critical and important issues which require further analysis and research.

Specifically, most existing works assume that both latent variables and noise variables obey Gaussian distributions. This assumption is usually violated in real-life industrial processes where the related variables often have non-Gaussian characteristics, owing to various reasons, such as shifting operating conditions, feedstock changes, production strategy changes, non-Gaussian disturbances, and process nonlinearity [18,27]. The assumption that all the noise variables have the identical power level, as required by the PPCA-related methods [12,14,16,20], is also too rigorous and unrealistic. Thus, an appropriate multivariate statistical analysis approach is urgently demanded, which can overcome these above-mentioned limitations and is capable of meeting the practical requirements of fault detection in real-life

industrial environments. This is the main motivation of our work. Because ICA has good ability of extracting non-Gaussian latent variables, fault detection in high-noise industrial environments based on ICA is a promising topic worthy researching. Currently, there exist some “noise-resistant” ICA algorithms [43,47,44,57] in the blind source separation field, which can explicitly consider the measurement noise. Nevertheless, these noise-resistant ICA algorithms were not introduced for the fault detection purpose. The main reason can be attributed to the fact that these noise-resistant ICA algorithms generally demand some strict assumptions, such as that the covariance matrix of the measurement noise is a diagonal matrix with the same diagonal elements or has been obtained by prior knowledge, which cannot be met in the application to real industrial processes. Cai et al. [30] proposed a noisy ICA method for fault detection, which removes the restricted assumption imposed on the covariance matrix of the measurement noise by these existing noise-resistant ICA algorithms, but the method of [30] still requires that the measurement noise is Gaussian distributed. Consequently, developing an appropriate noise-resistant ICA algorithm, which can effectively eliminate or alleviate the effects of the measurement noise without requiring strict and unrealistic assumptions, is extremely important for meeting the need of the fault detection in actual process environments. Moreover, constructing a monitoring statistic which can reduce the effects of the measurement noise is also significant for further improving the fault detection performance.

Against this background, in this work, our main contribution is to propose a new process fault detection method based on a novel noise-resistant joint diagonalization ICA (NRJDICA) algorithm. Specifically, the measurement noise is considered explicitly in the noisy ICA model, and the proposed NRJDICA algorithm, which does not require either the covariance matrix of the measurement noise or the Gaussian distribution assumption for the measurement noise, is derived to estimate the mixing matrix by whitening the measured variables and performing the joint diagonalization of the whitened variables' time-delayed covariance matrices with the least-squares based non-orthogonal joint diagonalization algorithm of [50]. Furthermore, the estimated mixing matrix is used for setting up the relationships between the kurtosis statistics of the ICs and the fourth-order cross cumulant statistics of the measured variables, to help sorting the estimated ICs and selecting the dominant ICs. Our second contribution is to construct a new monitoring statistic to detect process faults based on the serial correlation information of each dominant IC calculated using a moving window technique. This newly derived monitoring statistic is capable of further reducing the influence of the measurement noise. Therefore, our proposed NRJDICA approach can effectively suppress the effects of the measurement noise and it offers a convenient and more effective means for the fault detection in real industrial processes.

The remainder of this paper is organized as follows. After briefly reviewing the conventional FastICA-based fault detection method in Section 2, our proposed NRJDICA-based fault detection method is formulated in detail in Section 3. To investigate the effectiveness of the proposed approach, its performance is evaluated and compared with that of the conventional method using a simple three-variable system and a continuous stirred tank reactor system in Section 4. Our conclusions are drawn in Section 5.

The following notational conventions are adopted throughout this contribution. Boldface capital and lower-case letters stand for matrices and column vectors, respectively, while  $\mathbb{R}$  denotes the field of real numbers. The transpose and inverse operators are denoted by  $(\bullet)^T$  and  $(\bullet)^{-1}$ , respectively, while  $\mathbf{I}_m$  denotes the  $m \times m$  identity matrix and  $\text{diag}\{a_1, a_2, \dots, a_m\}$  represents the  $m \times m$  diagonal matrix with  $a_1, a_2, \dots, a_m$  as its diagonal elements. Furthermore,  $E\{\bullet\}$  denotes the expectation operator, and  $\|\bullet\|_2^2$

stands for the squared Frobenius norm. Additionally, for  $\mathbf{A} \in \mathbb{R}^{m \times m}$  and  $p \leq m$ ,  $\mathbf{A}_{1:p, \cdot} \in \mathbb{R}^{p \times m}$  consists of the first  $p$  rows of  $\mathbf{A}$ , while  $\mathbf{A}_{\cdot, 1:p} \in \mathbb{R}^{m \times p}$  contains the first  $p$  columns of  $\mathbf{A}$ .

## 2. The conventional fastICA-based fault detection method

The conventional FastICA-based fault detection generally consists of two steps: (1) use the FastICA algorithm to estimate the mixing matrix and the ICs, and (2) construct the monitoring statistics to detect process faults. Usually, the FastICA algorithm adopts the following noise-free ICA model:

$$\mathbf{x} = \mathbf{A}\mathbf{s}, \quad (1)$$

where  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_m]^T \in \mathbb{R}^m$  denote the  $m$  zero-mean measured variables and  $\mathbf{s} = [s_1 \ s_2 \ \dots \ s_m]^T \in \mathbb{R}^m$  denote the  $m$  zero-mean ICs, while  $\mathbf{A} \in \mathbb{R}^{m \times m}$  is the unknown mixing matrix. The task of the FastICA is to estimate both  $\mathbf{A}$  and  $\mathbf{s}$  given only the measured  $\mathbf{x}$ . Alternatively, the objective of the FastICA is to estimate the ICs by finding a de-mixing matrix  $\mathbf{W} \in \mathbb{R}^{m \times m}$  from the measured  $\mathbf{x}$ , such that the estimate of the ICs can be expressed by

$$\hat{\mathbf{s}} = \mathbf{W}\mathbf{x}, \quad (2)$$

in which the elements of  $\hat{\mathbf{s}} = [\hat{s}_1 \ \hat{s}_2 \ \dots \ \hat{s}_m]^T$  are as independent of each other as possible.

The eigen-decomposition of the measured variables' covariance matrix  $\mathbf{C}_x = E\{\mathbf{x}(t)\mathbf{x}^T(t)\}$ , in which the vector  $\mathbf{x}(t)$  contains the sample values of the measured variables at sample time  $t$ , is given by

$$\mathbf{C}_x = \mathbf{V} \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_m\} \mathbf{V}^T. \quad (3)$$

Here  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$  are the eigenvalues of  $\mathbf{C}_x$ , while  $\mathbf{V} \in \mathbb{R}^{m \times m}$  is the matrix whose column vectors are the eigenvectors of  $\mathbf{C}_x$  and therefore  $\mathbf{V}\mathbf{V}^T = \mathbf{V}^T\mathbf{V} = \mathbf{I}_m$ . The measured data need to be whitened first, and the whitening transformation can be implemented as

$$\mathbf{z} = \mathbf{Q}\mathbf{x}, \quad (4)$$

where  $\mathbf{Q} = \mathbf{\Lambda}^{-1/2} \mathbf{V}^T$  with  $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_m\}$  is the whitening matrix, and  $\mathbf{z} = [z_1 \ z_2 \ \dots \ z_m]^T \in \mathbb{R}^m$  are the  $m$  whitened variables. The whitened variables' covariance matrix  $\mathbf{C}_z = E\{\mathbf{z}(t)\mathbf{z}^T(t)\}$  satisfies the condition  $\mathbf{C}_z = \mathbf{I}_m$ .

Without loss of generality, the covariance matrix of the ICs  $\mathbf{s}$  can be assumed to be the  $m \times m$  identity matrix, that is,  $E\{\mathbf{s}(t)\mathbf{s}^T(t)\} = \mathbf{I}_m$ . Then, Eq. (2) can be rewritten as

$$\hat{\mathbf{s}} = \mathbf{W}\mathbf{x} = \mathbf{U}\mathbf{Q}\mathbf{x} = \mathbf{U}\mathbf{z}, \quad (5)$$

by choosing the de-mixing matrix  $\mathbf{W} = \mathbf{U}\mathbf{Q}$ , where  $\mathbf{U}$  is an orthonormal matrix satisfying  $E\{\hat{\mathbf{s}}(t)\hat{\mathbf{s}}^T(t)\} = E\{\mathbf{U}\mathbf{z}(t)\mathbf{z}^T(t)\mathbf{U}^T\} = \mathbf{U}E\{\mathbf{z}(t)\mathbf{z}^T(t)\}\mathbf{U}^T = \mathbf{U}\mathbf{U}^T = \mathbf{I}_m$ . Thus, the problem of finding the original de-mixing matrix  $\mathbf{W}$  is converted to a simpler problem of computing the orthonormal matrix  $\mathbf{U}$ . To calculate  $\mathbf{U}$ , the FastICA algorithm solves the following optimization which takes the objective function  $J$  as the approximation of negentropy [41]:

$$\begin{aligned} \max_{\mathbf{u}_i^T \in \mathbb{R}^{1 \times m}} J(\mathbf{u}_i^T) &= \max_{\mathbf{u}_i^T \in \mathbb{R}^{1 \times m}} (E\{G(\mathbf{u}_i^T \mathbf{z})\} - E\{G(v)\}), \\ \text{s.t. } \mathbf{u}_i^T \mathbf{u}_i &= 1, \quad \mathbf{u}_i^T \mathbf{u}_{i-1} = \mathbf{u}_i^T \mathbf{u}_{i-2} = \dots = \mathbf{u}_i^T \mathbf{u}_1 = 0, \end{aligned} \quad (6)$$

for  $1 \leq i \leq m$ , where  $v$  is a Gaussian variable with zero mean and unit variance, while  $G(\bullet)$  is a non-quadratic function which can be chosen as  $G(v) = -\exp(-v^2/2)$ . The resulting  $\mathbf{u}_i^T$  obtained by solving the optimization (6) forms the  $i$ th row of  $\mathbf{U}$ . The more detailed description of the FastICA algorithm can be found in [41].

After obtaining  $\mathbf{U}$  by the FastICA, the ICs can be estimated using Eq. (5). The estimated ICs can be arranged in the descending order according to their non-Gaussian degrees measured by the negentropy statistic [34], and the rows of the matrix  $\mathbf{U}$  are also sorted accordingly. The estimate of the mixing matrix  $\mathbf{A}$  is then calculated

according to

$$\hat{\mathbf{A}} = (\mathbf{U}\mathbf{Q})^{-1}. \quad (7)$$

The two widely used monitoring statistics for fault detection are given by [23,25,28,38,39]

$$I^2(t) = \left( (\hat{\mathbf{A}}^{-1})_{1:p, \cdot} \mathbf{x}(t) \right)^T (\hat{\mathbf{A}}^{-1})_{1:p, \cdot} \mathbf{x}(t) = \hat{\mathbf{s}}_p^T(t) \hat{\mathbf{s}}_p(t), \quad (8)$$

$$\text{SPE}(t) = (\mathbf{x}(t) - \hat{\mathbf{A}}_{:,1:p} \hat{\mathbf{s}}_p(t))^T (\mathbf{x}(t) - \hat{\mathbf{A}}_{:,1:p} \hat{\mathbf{s}}_p(t)), \quad (9)$$

where  $p$  is the number of the dominant ICs chosen, and  $\hat{\mathbf{s}}_p = (\hat{\mathbf{A}}^{-1})_{1:p, \cdot} \mathbf{x} \in \mathbb{R}^p$  denotes the vector that contains the first  $p$  extracted dominant ICs.  $I^2(t)$  is used to monitor the systematic part of the process variation, while SPE( $t$ ) is used to monitor the non-systematic part of the process variation.

### 3. The proposed NRJDICA-based fault detection method

In the above FastICA-based fault detection, the noise-free FastICA used for estimating the mixing matrix and the ICs does not consider the measurement noise, which may lead to an inadequate feature extraction in highly noisy environments. A possible solution is to replace this noise-free FastICA with the existing noise-resistant ICA algorithms of [43,44,47,57]. But these noise-resistant ICA algorithms require some prior knowledge of the measurement noise that are unavailable in most practical industrial processes and/or impose some unrealistic assumptions that may not be met in practice. In addition, the monitoring statistics  $I^2$  and SPE calculated using Eqs. (8) and (9) are also subject to the negative effects of the measurement noise. All of these may contribute to a degraded fault detection performance in noisy industrial environments. This motivates us to derive a new NRJDICA method for estimating the mixing matrix and the ICs, which can effectively reduce the effects of the measurement noise while imposing no unrealistic assumptions on the measurement noise, as well as to propose a new monitoring statistic which can further alleviate the adverse influence of the measurement noise to improve the fault detection performance.

#### 3.1. The NRJDICA algorithm

Instead of using the noise-free ICA model (1), we explicitly consider the measurement noise and use the following noisy ICA model:

$$\tilde{\mathbf{x}} = \mathbf{A}\mathbf{s} + \boldsymbol{\varepsilon}, \quad (10)$$

where  $\tilde{\mathbf{x}} \in \mathbb{R}^m$  are the measured variables which are contaminated with the measurement noise variables  $\boldsymbol{\varepsilon} = [\varepsilon_1 \ \varepsilon_2 \ \dots \ \varepsilon_m]^T \in \mathbb{R}^m$ . The ICs  $\mathbf{s}$  and the noise variables  $\boldsymbol{\varepsilon}$  satisfy the conditions: (1) the elements of  $\mathbf{s}$  are zero-mean and mutually independent; (2) the elements of  $\boldsymbol{\varepsilon}$  are zero-mean white variables and they are mutually independent; and (3) the elements of  $\mathbf{s}$  and the elements of  $\boldsymbol{\varepsilon}$  are mutually independent. It is worth emphasizing that unlike many existing approaches, the distribution of the measurement noise is not restricted to be Gaussian in our derivation.

As the objective of the NRJDICA is also to estimate the mixing matrix  $\mathbf{A}$  and the ICs  $\mathbf{s}$  from the measured variables  $\tilde{\mathbf{x}}$ , the whitening of  $\tilde{\mathbf{x}}$  can be carried out according to

$$\tilde{\mathbf{z}} = \tilde{\mathbf{Q}}\tilde{\mathbf{x}}, \quad (11)$$

where  $\tilde{\mathbf{z}} = [\tilde{z}_1 \ \tilde{z}_2 \ \dots \ \tilde{z}_m]^T \in \mathbb{R}^m$  are the obtained whitened variables, and the whitening matrix is given by  $\tilde{\mathbf{Q}} = \hat{\mathbf{\Lambda}}^{-1/2} \hat{\mathbf{V}}^T \in \mathbb{R}^{m \times m}$  in which  $\hat{\mathbf{\Lambda}} = \text{diag}\{\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_m\}$  has the eigenvalues of the measured variables' covariance matrix  $\mathbf{C}_{\tilde{\mathbf{x}}} = E\{\tilde{\mathbf{x}}(t)\tilde{\mathbf{x}}^T(t)\}$  as its diagonal

elements, while the columns of  $\hat{\mathbf{V}}$  are the eigenvectors of  $\mathbf{C}_{\tilde{\mathbf{x}}}$ . Clearly,  $\mathbf{C}_{\tilde{\mathbf{z}}} = E\{\tilde{\mathbf{z}}(t)\tilde{\mathbf{z}}^T(t)\} = \mathbf{I}_m$ .

By expressing the time-delayed covariance matrix of the lagged  $\tilde{\mathbf{z}}(t-i)$  and  $\tilde{\mathbf{z}}(t-j)$  as

$$\mathbf{C}_{\tilde{\mathbf{z}}}(i, j) = E\{\tilde{\mathbf{z}}(t-i)\tilde{\mathbf{z}}^T(t-j)\}, \quad (12)$$

we can define the time structure matrix of the whitened variables  $\tilde{\mathbf{z}}$  as the weighted sum of the whitened variables' different time-delayed covariance matrices, which can be written as

$$\begin{aligned} \mathbf{M}_{\tau_l, \tau_u} &= \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} \gamma_{ij} (E\{\tilde{\mathbf{z}}(t-i)\tilde{\mathbf{z}}^T(t-j)\} \\ &\quad + E\{\tilde{\mathbf{z}}(t-j)\tilde{\mathbf{z}}^T(t-i)\}) \\ &= \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} \gamma_{ij} (\mathbf{C}_{\tilde{\mathbf{z}}}(i, j) + \mathbf{C}_{\tilde{\mathbf{z}}}(j, i)), \end{aligned} \quad (13)$$

where  $\gamma_{ij}$  for  $\tau_l \leq i \leq \tau_u - 1$  and  $\tau_l + 1 \leq j \leq \tau_u$  are the weighting coefficients, while  $\tau_l$  and  $\tau_u$  define the predetermined minimum and maximum time lags, respectively. Typically, equal weighting is applied to form the time structure matrix (13). Substituting Eqs. (10) and (11) into Eq. (13) as well as noticing the conditions that  $\mathbf{s}$  and  $\boldsymbol{\varepsilon}$  satisfy as outlined after Eq. (10),  $\mathbf{M}_{\tau_l, \tau_u}$  can be expressed as

$$\begin{aligned} \mathbf{M}_{\tau_l, \tau_u} &= \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} \gamma_{ij} \tilde{\mathbf{Q}} (\mathbf{C}_{\tilde{\mathbf{x}}}(i, j) + \mathbf{C}_{\tilde{\mathbf{x}}}(j, i)) \tilde{\mathbf{Q}}^T \\ &= \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} \gamma_{ij} \tilde{\mathbf{Q}} \mathbf{A} (\mathbf{C}_s(i, j) + \mathbf{C}_s(j, i)) \mathbf{A}^T \tilde{\mathbf{Q}}^T \\ &= \mathbf{H}^T \text{diag} \left\{ \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} 2\gamma_{ij} E\{s_1(t-i)s_1(t-j)\}, \right. \\ &\quad \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} 2\gamma_{ij} E\{s_2(t-i)s_2(t-j)\}, \dots, \\ &\quad \left. \sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} 2\gamma_{ij} E\{s_m(t-i)s_m(t-j)\} \right\} \mathbf{H} \\ &= \mathbf{H}^T \mathbf{\Gamma}_{\tau_l, \tau_u} \mathbf{H}, \end{aligned} \quad (14)$$

where  $\mathbf{H}^T = \tilde{\mathbf{Q}}\mathbf{A}$ , and the  $g$ th diagonal element of the diagonal matrix  $\mathbf{\Gamma}_{\tau_l, \tau_u}$ , namely,

$$\sum_{i=\tau_l}^{\tau_u-1} \sum_{j=i+1}^{\tau_u} 2\gamma_{ij} E\{s_g(t-i)s_g(t-j)\},$$

is referred to as the time structure of the  $g$ th IC for  $1 \leq g \leq m$ .

From Eq. (14), it can easily be seen that the time structure matrix  $\mathbf{M}_{\tau_l, \tau_u}$  can be diagonalized by the matrix  $\mathbf{H}$ . By giving two different values for  $\tau$ , i.e.  $\tau_{l_1}$  and  $\tau_{l_2}$ , we have the two different time structure matrices,  $\mathbf{M}_{\tau_{l_1}, \tau_u}$  and  $\mathbf{M}_{\tau_{l_2}, \tau_u}$ . Then, the joint diagonalization degree of the matrices  $\mathbf{M}_{\tau_{l_1}, \tau_u}$  and  $\mathbf{M}_{\tau_{l_2}, \tau_u}$  can be used as the optimization objective to estimate the matrix  $\mathbf{H}$ . Specifically, we define the following weighted least-squares criterion to measure the attainable joint diagonalization degree by the matrix  $\hat{\mathbf{H}}$

$$\begin{aligned} J_{\text{WLS}}(\hat{\mathbf{H}}, \hat{\mathbf{\Gamma}}_{\tau_{l_1}, \tau_u}, \hat{\mathbf{\Gamma}}_{\tau_{l_2}, \tau_u}) \\ = \sum_{i=1}^2 w_i \|\mathbf{M}_{\tau_{l_i}, \tau_u} - \hat{\mathbf{H}}^T \hat{\mathbf{\Gamma}}_{\tau_{l_i}, \tau_u} \hat{\mathbf{H}}\|_F^2, \end{aligned} \quad (15)$$

where  $w_i$  for  $i=1,2$  are the positive weights, and  $\hat{\mathbf{H}}$  denotes the estimate of  $\mathbf{H}$ , while  $\hat{\mathbf{\Gamma}}_{\tau_{l_i}, \tau_u}$  for  $i=1,2$  are the estimates of  $\mathbf{\Gamma}_{\tau_{l_i}, \tau_u}$ . To estimate the matrix  $\mathbf{H}$  by minimizing the cost function (15), the so-called AC–DC algorithm [50] can be utilized, which alternates between the following two minimization procedures.

- (a) The alternating-columns (AC) phase minimizes  $J_{\text{WLS}}(\hat{\mathbf{H}}, \hat{\mathbf{\Gamma}}_{\tau_{l_1}, \tau_u}, \hat{\mathbf{\Gamma}}_{\tau_{l_2}, \tau_u})$  with respect to the  $i$ th column of  $\hat{\mathbf{H}}$ , while keeping its other columns as well as  $\hat{\mathbf{\Gamma}}_{\tau_{l_1}, \tau_u}$  and  $\hat{\mathbf{\Gamma}}_{\tau_{l_2}, \tau_u}$  fixed, for  $1 \leq i \leq m$ . This sweeping procedure is repeated  $q$  times, and

each sweep starts with the initial  $\hat{\mathbf{H}}$  obtained from the previous sweep. Typically,  $q=10$  is sufficient.

- (b) The diagonalizing-centers (DC) phase minimizes  $J_{\text{WLS}}(\hat{\mathbf{H}}, \hat{\mathbf{\Gamma}}_{\tau_{i_1}}, \tau_u, \hat{\mathbf{\Gamma}}_{\tau_{i_2}, \tau_u})$  with respect to  $\hat{\mathbf{\Gamma}}_{\tau_{i_1}, \tau_u}$  and  $\hat{\mathbf{\Gamma}}_{\tau_{i_2}, \tau_u}$ , while keeping  $\hat{\mathbf{H}}$  fixed.

The algorithm alternates between the two phases for a number of times, until the reduction in the cost function values of (15) between two consecutive iterations is below a threshold value, e.g.  $10^{-5}$ . The details of the AC–DC algorithm can be found in [50].

Once the estimated matrix  $\hat{\mathbf{H}}$  is obtained, the mixing matrix  $\mathbf{A}$  can be estimated using

$$\hat{\mathbf{A}} = \hat{\mathbf{Q}}^{-1} \hat{\mathbf{H}}^T, \quad (16)$$

and the ICs can then be estimated using

$$\hat{\mathbf{s}} = \hat{\mathbf{A}}^{-1} \hat{\mathbf{x}} = (\hat{\mathbf{H}}^T)^{-1} \hat{\mathbf{Q}} \hat{\mathbf{x}}, \quad (17)$$

where for notational simplification we still use the notation  $\hat{\mathbf{s}} = [\hat{s}_1 \hat{s}_2 \dots \hat{s}_m]^T$  to denote the estimated ICs by the NRJDICA algorithm.

Referring to Fig. 1, we now summarize the proposed NRJDICA algorithm.

- (i) Calculate the whitening matrix  $\hat{\mathbf{Q}}$  and whiten the measured variables  $\hat{\mathbf{x}}$  using Eq. (11) to obtain the whitened variables  $\hat{\mathbf{z}}$ .
- (ii) Construct the two time structure matrices of the whitened variables  $\hat{\mathbf{z}}$ , denoted as  $\mathbf{M}_{\tau_{i_1}, \tau_u}$  and  $\mathbf{M}_{\tau_{i_2}, \tau_u}$ , using Eq. (13).
- (iii) Use the constructed  $\mathbf{M}_{\tau_{i_1}, \tau_u}$  and  $\mathbf{M}_{\tau_{i_2}, \tau_u}$  to form the measure (15) of the joint diagonalization degree.
- (iv) Minimize the cost function (15) by the AC–DC algorithm to estimate the matrix  $\mathbf{H}$ .
- (v) With the estimated  $\hat{\mathbf{H}}$ , estimate the mixing matrix  $\mathbf{A}$  and the ICs  $\mathbf{s}$  using Eqs. (16) and (17), respectively.

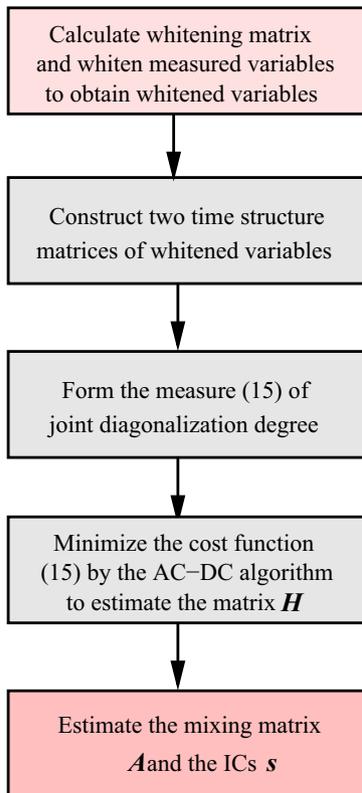


Fig. 1. The schematic of the proposed NRJDICA algorithm.

The computational complexities of the AC and DC phases in the AC–DC algorithm are respectively  $\mathcal{O}(qm^5)$  and  $\mathcal{O}(m^5)$  [50]. Therefore, the overall computational complexity of our NRJDICA algorithm is on the order of  $\mathcal{O}(qm^5)$ . By contrast, the computational complexity of the FastICA algorithm is on the order of  $\mathcal{O}(mN_1)$  [48], where  $N_1$  is the sample size of the training data. The sample size  $N_1$  and the number of the measured variables  $m$  are determined by the specific application. Usually, the number of the training data  $N_1$  is large. Thus, for the cases of modest  $m$ , the computational complexity of the NRJDICA may be comparable to that of the FastICA. It is worth noting that the operations of estimating the mixing matrix  $\mathbf{A}$  are executed offline and therefore the complexity of this off-line modeling stage is not too critical. The significant advantage of our NRJDICA algorithm over the FastICA algorithm is that it effectively takes the measurement noise into consideration in the modeling stage.

### 3.2. The construction of a noise-restraining monitoring statistic

After obtaining the estimates of the mixing matrix and the ICs by the NRJDICA algorithm, we may simply use the estimated ICs  $\hat{\mathbf{s}}$  in Eqs. (8) and (9) to calculate the monitoring statistics  $I^2$  and SPE for fault detection, as a usual practice. However, the estimated ICs  $\hat{\mathbf{s}}$  by the NRJDICA algorithm are still corrupted by the white noise variables  $\tilde{\mathbf{e}}$ , which can be clearly seen from

$$\hat{\mathbf{s}} = \hat{\mathbf{A}}^{-1} (\mathbf{A}\mathbf{s} + \mathbf{e}) \approx \mathbf{s} + \hat{\mathbf{A}}^{-1} \mathbf{e} = \mathbf{s} + \tilde{\mathbf{e}}. \quad (18)$$

Thus, the calculated  $I^2$  and SPE also suffer from the adverse effects of the measurement noise. In order to conduct reliable and efficient fault detection, it is highly desired to construct a noise-restraining monitoring statistic.

Before building a monitoring statistic, the estimated ICs should be arranged in the descending order according to their non-Gaussian degrees [34,42] and the dominant ICs are selected. Here, we apply the well-known non-Gaussian measure called kurtosis [51,52] to help achieving this purpose. More specifically, the relationships between the kurtosis statistics of the ICs  $\mathbf{s}$  and the fourth-order cross cumulant statistics of the measured variables can be firstly established based on the estimated mixing matrix  $\hat{\mathbf{A}}$ , through which the kurtosis statistics of the ICs  $\mathbf{s}$  can be estimated without the interference of the measurement noise. Then, the obtained kurtosis estimates of the ICs  $\mathbf{s}$  can be used to sort the estimated ICs  $\hat{\mathbf{s}}$  and to select the dominant ICs.

In particular, we consider the following fourth-order cross cumulant of the measured variables  $\tilde{x}_i$  and  $\tilde{x}_{i+1}$  for  $i = 1, 2, \dots, m-1$ :

$$cm(\tilde{x}_i, \tilde{x}_i, \tilde{x}_{i+1}, \tilde{x}_{i+1}) = E\{\tilde{x}_i^2 \tilde{x}_{i+1}^2\} - E\{\tilde{x}_i^2\}E\{\tilde{x}_{i+1}^2\} - 2(E\{\tilde{x}_i \tilde{x}_{i+1}\})^2. \quad (19)$$

According to the conditions that the ICs  $\mathbf{s}$  and the measurement noise variables  $\mathbf{e}$  satisfy, as stated after Eq. (10), as well as the multi-linearity property of cumulant [52], Eq. (19) can be expressed as

$$\begin{aligned} cm(\tilde{x}_i, \tilde{x}_i, \tilde{x}_{i+1}, \tilde{x}_{i+1}) &= cm\left(\sum_{g=1}^m a_{i,g} s_g + \varepsilon_i, \sum_{g=1}^m a_{i,g} s_g + \varepsilon_i, \sum_{g=1}^m a_{i+1,g} s_g + \varepsilon_{i+1}, \sum_{g=1}^m a_{i+1,g} s_g + \varepsilon_{i+1}\right) \\ &= \sum_{g=1}^m a_{i,g} a_{i,g} a_{i+1,g} a_{i+1,g} cm(s_g, s_g, s_g, s_g) \\ &= \sum_{g=1}^m a_{i,g}^2 a_{i+1,g}^2 k_4(s_g), \end{aligned} \quad (20)$$

where  $a_{i,g}$  is the  $i$ th-row and  $g$ th-column element of the mixing matrix  $\mathbf{A}$  and  $k_4(s_g) = cm(s_g, s_g, s_g, s_g)$  is known as the kurtosis

statistic of the  $g$ th IC  $s_g$ . From Eq. (20), the relationship between the kurtosis statistics of the ICs  $\mathbf{s}$  and the fourth-order cross cumulant statistics of the measured variables  $\tilde{\mathbf{x}}$  can be written in the matrix form

$$\begin{bmatrix} cm(\tilde{x}_1, \tilde{x}_1, \tilde{x}_2, \tilde{x}_2) \\ cm(\tilde{x}_2, \tilde{x}_2, \tilde{x}_3, \tilde{x}_3) \\ \vdots \\ cm(\tilde{x}_{m-1}, \tilde{x}_{m-1}, \tilde{x}_m, \tilde{x}_m) \\ cm(\tilde{x}_m, \tilde{x}_m, \tilde{x}_1, \tilde{x}_1) \end{bmatrix} = \begin{bmatrix} a_{1,1}^2 a_{2,1}^2 & a_{1,2}^2 a_{2,2}^2 & \cdots & a_{1,m}^2 a_{2,m}^2 \\ a_{2,1}^2 a_{3,1}^2 & a_{2,2}^2 a_{3,2}^2 & \cdots & a_{2,m}^2 a_{3,m}^2 \\ \vdots & \vdots & \cdots & \vdots \\ a_{m-1,1}^2 a_{m,1}^2 & a_{m-1,2}^2 a_{m,2}^2 & \cdots & a_{m-1,m}^2 a_{m,m}^2 \\ a_{m,1}^2 a_{1,1}^2 & a_{m,2}^2 a_{1,2}^2 & \cdots & a_{m,m}^2 a_{1,m}^2 \end{bmatrix} \begin{bmatrix} k_4(s_1) \\ k_4(s_2) \\ \vdots \\ k_4(s_{m-1}) \\ k_4(s_m) \end{bmatrix}. \quad (21)$$

It can be observed from Eq. (21) that the calculation of the fourth-order cross cumulant is not influenced by the measurement noise variables  $\epsilon$ . Let us define

$$\hat{\Phi} = \begin{bmatrix} \hat{a}_{1,1}^2 \hat{a}_{2,1}^2 & \hat{a}_{1,2}^2 \hat{a}_{2,2}^2 & \cdots & \hat{a}_{1,m}^2 \hat{a}_{2,m}^2 \\ \hat{a}_{2,1}^2 \hat{a}_{3,1}^2 & \hat{a}_{2,2}^2 \hat{a}_{3,2}^2 & \cdots & \hat{a}_{2,m}^2 \hat{a}_{3,m}^2 \\ \vdots & \vdots & \cdots & \vdots \\ \hat{a}_{m-1,1}^2 \hat{a}_{m,1}^2 & \hat{a}_{m-1,2}^2 \hat{a}_{m,2}^2 & \cdots & \hat{a}_{m-1,m}^2 \hat{a}_{m,m}^2 \\ \hat{a}_{m,1}^2 \hat{a}_{1,1}^2 & \hat{a}_{m,2}^2 \hat{a}_{1,2}^2 & \cdots & \hat{a}_{m,m}^2 \hat{a}_{1,m}^2 \end{bmatrix}, \quad (22)$$

where  $\hat{a}_{ig}$  is the  $i$ th-row and  $g$ th-column element of the estimated mixing matrix  $\hat{\mathbf{A}}$ , and

$$\mathbf{cm}(\tilde{\mathbf{x}}) = [cm(\tilde{x}_1, \tilde{x}_1, \tilde{x}_2, \tilde{x}_2)cm(\tilde{x}_2, \tilde{x}_2, \tilde{x}_3, \tilde{x}_3)\cdots cm(\tilde{x}_{m-1}, \tilde{x}_{m-1}, \tilde{x}_m, \tilde{x}_m)cm(\tilde{x}_m, \tilde{x}_m, \tilde{x}_1, \tilde{x}_1)]^T. \quad (23)$$

Further denote the kurtosis estimate of the  $i$ th IC as  $\hat{k}_4(s_i)$ . Then the estimated kurtosis statistics of the ICs  $\mathbf{s}$ , denoted as  $\hat{\mathbf{k}}_4 = [\hat{k}_4(s_1)\hat{k}_4(s_2)\cdots\hat{k}_4(s_m)]^T$ , can be calculated according to

$$\hat{\mathbf{k}}_4 = (\hat{\Phi}^T \hat{\Phi} + \mu \mathbf{I}_m)^{-1} \hat{\Phi}^T \mathbf{cm}(\tilde{\mathbf{x}}), \quad (24)$$

where  $\mu \geq 0$  is a small regularization parameter, e.g.  $10^{-6}$ .

Based on the estimated kurtosis statistics given in Eq. (24), the estimated ICs  $\hat{\mathbf{s}}$  can be arranged in the descending order according to the absolute values of  $\hat{k}_4(s_i)$  for  $1 \leq i \leq m$  and the rows of the matrix  $(\hat{\mathbf{H}}^T)^{-1}$  in Eq. (17) are also sorted accordingly. After this sorting, the first  $p$  estimated ICs are selected as the dominant ICs.

To reduce the effects of the measurement noise, we adopt the moving window technique to calculate the serial correlation information of each dominant IC as follows:

$$r_i(t) = \frac{1}{h-1} \sum_{j=0}^{h-1} \hat{s}_i(t-j)\hat{s}_i(t-1-j), \quad 1 \leq i \leq p, \quad (25)$$

where  $h$  is the window width. By calculating the serial correlation information  $r_i(t)$  of the  $i$ th dominant IC for  $1 \leq i \leq p$ , the effects of the measurement noise can be effectively alleviated. This is because from Eq. (18), we have

$$\begin{aligned} r_i(t) &= \frac{1}{h-1} \sum_{j=0}^{h-1} (s_i(t-j)s_i(t-1-j) \\ &\quad + s_i(t-j)\tilde{\epsilon}_i(t-1-j) + \tilde{\epsilon}_i(t-j)s_i(t-1-j) \\ &\quad + \tilde{\epsilon}_i(t-j)\tilde{\epsilon}_i(t-1-j)) \\ &\approx \frac{1}{h-1} \sum_{j=0}^{h-1} s_i(t-j)s_i(t-1-j), \end{aligned}$$

in which the approximation follows from the fact that the noise variables  $\tilde{\epsilon} = [\tilde{\epsilon}_1 \ \tilde{\epsilon}_2 \ \cdots \ \tilde{\epsilon}_m]^T$  in Eq. (18) are white, and  $\mathbf{s}$  and  $\tilde{\epsilon}$  are mutually independent. Thus, based on the serial correlation

information  $\mathbf{r}_p(t) = [r_1(t) \ r_2(t) \ \cdots \ r_p(t)]^T$  of the  $p$  estimated dominant ICs  $\hat{\mathbf{s}}_p(t) = [\hat{s}_1(t) \ \hat{s}_2(t) \ \cdots \ \hat{s}_p(t)]^T$ , the following noise-restraining monitoring statistic:

$$L^2(t) = \mathbf{r}_p^T(t) \mathbf{\Psi}^{-1} \mathbf{r}_p(t), \quad (26)$$

can be constructed to carry out fault detection, where  $\mathbf{\Psi} = E\{\mathbf{r}_p(t)\mathbf{r}_p^T(t)\}$  is the covariance matrix of  $\mathbf{r}_p(t)$  which can be estimated using the training data. The idea behind this new  $L^2(t)$  monitoring statistic is explained more explicitly in Appendix A.

After the construction of the noise-restraining monitoring statistic  $L^2(t)$ , the corresponding confidence limit needs to be determined for judging whether the monitored process is in control or not. The  $\delta$  confidence limit for the built monitoring statistic can be determined as follows. Split the measured normal operating data into the two parts: the training data with  $N_1$  samples and the validating data with  $N_2$  samples. Based on the training data, both the serial correlation information  $\{\mathbf{r}_p(t)\}_{t=h}^{N_1}$  of the dominant ICs and the covariance matrix  $\mathbf{\Psi}$  of  $\mathbf{r}_p(t)$  are estimated. Based on the validating data, the monitoring statistic values are calculated which are denoted as  $\{L^2(t)\}_{t=h}^{N_2}$ . Then round  $(N_2 - h + 1)(1 - \delta)$  towards the nearest integer denoted as  $c$  and adopt the  $c$ th highest value of  $\{L^2(t)\}_{t=h}^{N_2}$  as the confidence limit for the  $L^2(t)$  monitoring statistic. With this strategy, the false alarm rates of different fault detection methods can be adjusted to the same level approximately [53], and this makes it possible for fair and convenient comparison of different monitoring methods.

### 3.3. The NRJDICA-based fault detection with the new monitoring statistic

As illustrated in Fig. 2, the proposed NRJDICA based fault detection approach using the new noise-restraining monitoring

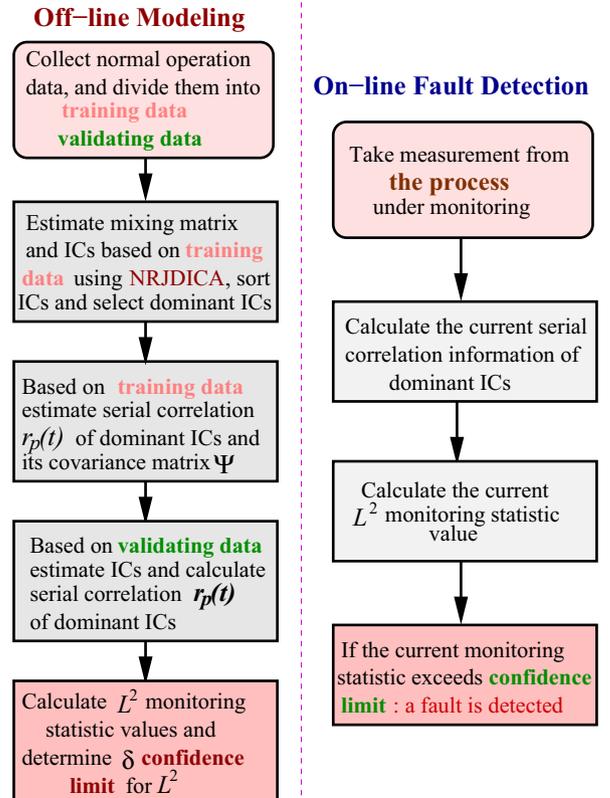


Fig. 2. The fault detection procedure using the NRJDICA algorithm with the new monitoring statistic.

statistic  $L^2(t)$  consists of the off-line modeling stage and the on-line monitoring stage, which are now detailed as follows.

*The off-line modeling stage:*

- (1) Collect the normal operating data from the process, and divide the data into the training data part, containing  $N_1$  samples, and the validating data part, containing  $N_2$  samples.
- (2) Based on the training data, estimate the mixing matrix and ICs by the NRJDICA algorithm, sort the estimated ICs and select the dominant ICs according to the estimated kurtosis statistics of the ICs defined in Eq. (24).
- (3) Based on the training data, calculate the serial correlation information  $\{\mathbf{r}_p(t)\}_{t=h}^{N_1}$  of the dominant ICs using Eq. (25) and estimate the covariance matrix of  $\{\mathbf{r}_p(t)\}_{t=h}^{N_1}$ .
- (4) Based on the validating data, estimate the ICs using Eq. (17) and calculate the serial correlation information  $\{\mathbf{r}_p(t)\}_{t=h}^{N_2}$  of the dominant ICs using Eq. (25).
- (5) Use the obtained serial correlation information  $\{\mathbf{r}_p(t)\}_{t=h}^{N_2}$  in Eq. (26) to calculate the monitoring statistic's values  $\{L^2(t)\}_{t=h}^{N_2}$ , and determine the  $\delta$  confidence limit for  $L^2(t)$ .

*The on-line monitoring stage:*

- (1) Take the current measurement from the process under monitoring, and estimate the current ICs according to Eq. (17).
- (2) Calculate the current serial correlation information of the dominant ICs.
- (3) Use the current serial correlation information of the dominant ICs obtained in Eq. (26) to calculate the current monitoring statistic value  $L^2(t)$ .
- (4) Determine whether the current monitoring statistic  $L^2(t)$  exceeds the confidence limit, and give an alarm if a fault is detected.

In the above procedure, the training data, the validating data and the current measured data for monitoring are all normalized with the means and variances of the measured variables in the training data. Additionally, the serial correlation information of the dominant ICs in the training data, the validating data and the current measured data are all normalized with the means of the serial correlation information of the dominant ICs calculated from the training data set. How to choose an appropriate window width  $h$  for calculating the serial correlation information  $\mathbf{r}_p(t)$  of the dominant ICs is discussed in Appendix B, while how to determine an appropriate number  $p$  of the dominant ICs is summarized in Appendix C.

## 4. Simulation studies

The proposed NRJDICA-based fault detection method is evaluated using the two case studies, a simple three-variable system and a continuous stirred tank reactor (CSTR) system. In both case studies, the fault detection performance of our NRJDICA-based fault detection method is compared with those of the FastICA-based fault detection method and the KPCA-based fault detection method [22].

### 4.1. A three-variable system

The three-variable system which is a modified version of the system studied by Kano et al. [29] is given by  $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{s} + \boldsymbol{\varepsilon}$  of Eq. (10),

with the mixing matrix defined by

$$\mathbf{A} = \begin{bmatrix} -0.4326 & 0.2877 & 1.1892 \\ -1.6656 & -1.1465 & -0.0376 \\ 0.1253 & 1.1909 & 0.3273 \end{bmatrix}. \quad (27)$$

The three ICs  $\mathbf{s} = [s_1 \ s_2 \ s_3]^T$  are given by

$$\begin{aligned} s_1(t+1) &= \sin(2\pi \cdot 18t/f_s), \\ s_2(t+1) &= ((\text{rem}(t, 23) - 11)/9)^5, \\ s_3(t+1) &= \sin(2\pi \cdot 0.9t/f_s) \cdot \sin(2\pi \cdot 30t/f_s), \end{aligned} \quad (28)$$

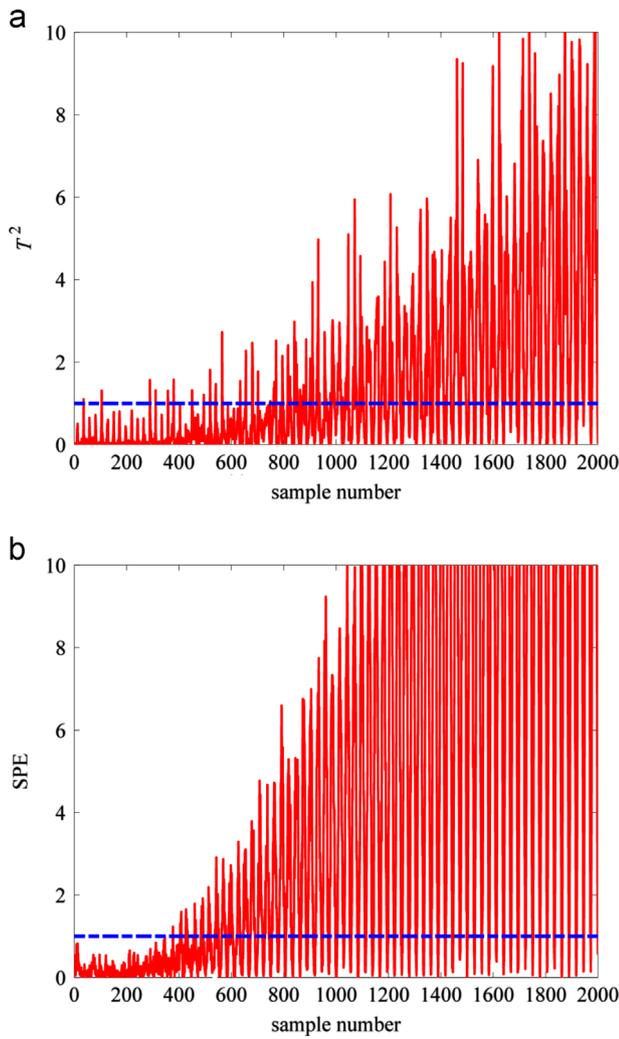
where  $f_s$  is the sampling frequency and the sample number  $t$  ranges from 0 to 3999, while  $\text{rem}(t, 23) = t - \lfloor t/23 \rfloor \cdot 23$  with  $\lfloor \bullet \rfloor$  denoting the integer floor operator. The three measured variables  $\tilde{\mathbf{x}} = [\tilde{x}_1 \ \tilde{x}_2 \ \tilde{x}_3]^T$  are contaminated with the three zero-mean measurement white noise variables  $\boldsymbol{\varepsilon} = [\varepsilon_1 \ \varepsilon_2 \ \varepsilon_3]^T$ , and  $\varepsilon_i$  follows the uniform distribution within  $[-\sqrt{3}\sigma_i, \sqrt{3}\sigma_i]$  with the variance  $\sigma_i^2$ . We define the noise intensity of  $\varepsilon_i$  as the ratio percentage of  $\varepsilon_i$ 's variance over the output  $\tilde{x}_i$ 's variance, for  $1 \leq i \leq 3$ . The noise intensities in the three output signals are set to 20%, 25% and 30%. A fault case, which is the ramp change of the second-row and the first-column element  $a_{2,1}$  in the mixing matrix  $\mathbf{A}$ , is simulated. The ramp change rate is set to 0.02. The normal operation data with 4000 samples and the fault data with 2000 samples are respectively generated by simulating the three-variable system. The normal operation data are divided into the training set of  $N_1 = 2000$  samples and the validating set of  $N_2 = 2000$  samples. For the fault data, the fault is introduced at the 101th sample.

Define  $\mathbf{G} = \hat{\mathbf{A}}^{-1} \mathbf{A} \in \mathbb{R}^{m \times m}$ . The following permutation error (PE) [46]

$$\text{PE} = \frac{1}{m} \sum_{i=1}^m \left( \sum_{j=1}^m \left( \frac{|g_{ij}|}{\max_l |g_{il}|} + \frac{|g_{ji}|}{\max_l |g_{li}|} \right) - 2 \right), \quad (29)$$

where  $g_{ij}$  denotes the  $i$ th-row and  $j$ th-column element of  $\mathbf{G}$ , is adopted to conduct quantitative performance comparison of the FastICA and NRJDICA algorithms. For the FastICA,  $\mathbf{G} = \mathbf{U}\mathbf{Q}\mathbf{A}$ , while for the NRJDICA,  $\mathbf{G} = (\hat{\mathbf{H}}^T)^{-1} \hat{\mathbf{Q}}\mathbf{A}$ . A smaller PE value indicates that the corresponding algorithm has the higher accuracy for estimating the mixing matrix and the ICs. The PE value for the FastICA is 1.801, while the PE value for the NRJDICA is only 0.5820. This clearly confirms that the NRJDICA algorithm can effectively attenuate the adverse effects of the measurement noise and, therefore, it achieves much more accurate estimates of the mixing matrix and the ICs than the FastICA algorithm. Consequently, the NRJDICA-based fault detection method is expected to outperform the FastICA-based fault detection method.

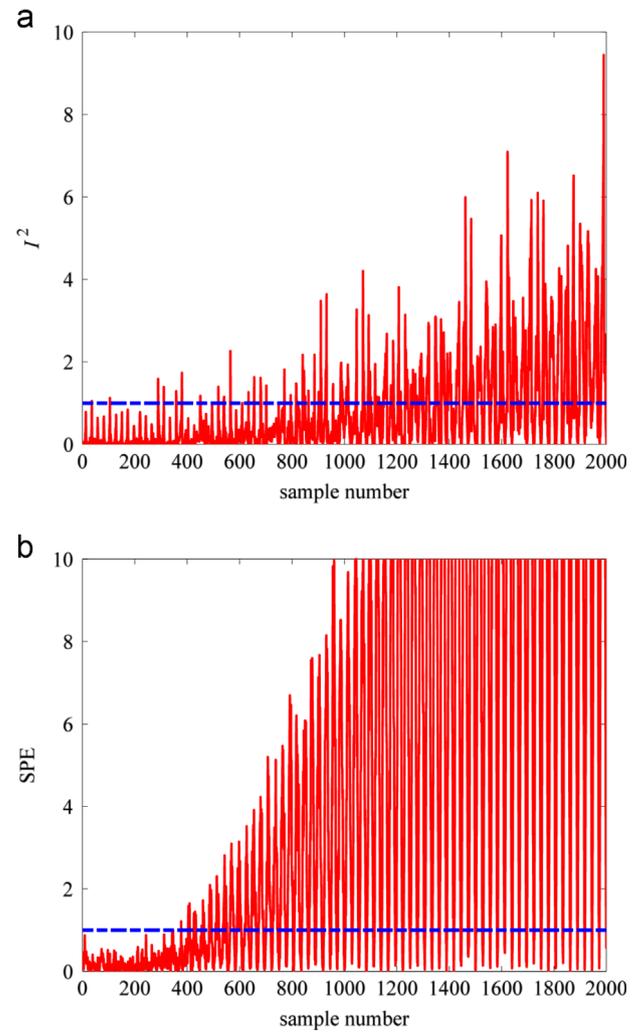
The number of the dominant ICs is set to  $p=1$  in both the FastICA-based and NRJDICA-based fault detection methods. The window width in Eq. (25) is empirically chosen to be  $h=54$  for the NRJDICA-based fault detection method. As this is a linear process, the linear kernel is chosen for the KPCA-based fault detection method. For a fair comparison, the number of the kernel PCs is also set to one. The fault detection performance is evaluated in terms of the fault detection rate, which is defined as the percentage of the fault samples whose monitoring statistic values exceed the corresponding confidence limit in all the fault samples, and the fault detection time. In order to reduce the false alarm, a fault is indicated only when six consecutive monitoring statistic values exceed the confidence limit and the fault detection time is then defined as the first sample at which the confidence limit is exceeded [54]. The  $\delta=99\%$  confidence limit is utilized as the alarming threshold. The monitoring charts of the three fault detection methods are illustrated in Figs. 3–5. For a convenience comparison, in each monitoring chart, the monitoring statistic values are normalized by the corresponding confidence limit so



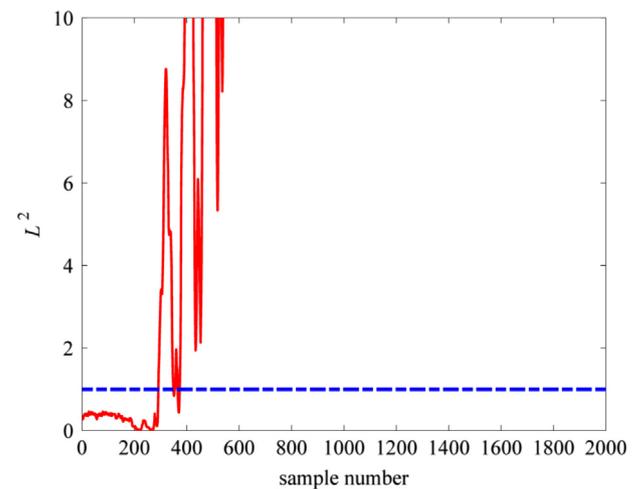
**Fig. 3.** The monitoring charts of the KPCA-based fault detection method for the simple three-variable system. The fault occurs at the 101th sample. (a) The  $T^2$  monitoring chart. (b) The SPE monitoring chart.

that the confidence limit is equal to 1. The monitoring statistic values are plotted as the solid line, while the related confidence limit is plotted as the dashed line.

It can be seen from the KPCA-based monitoring charts of Fig. 3 that after the occurrence of the ramp-change fault at the 101th sample, the  $T^2$  monitoring statistic values fluctuate around the confidence limit and cannot exceed the threshold continuously until about the 900th sample, while the SPE monitoring statistic values of many fault samples are still below the confidence limit, leading to a very late detection of the fault. Similarly, in the FastICA-based monitoring charts of Fig. 4, the  $I^2$  monitoring statistic values also fluctuate around the corresponding confidence limit until about the 1200th sample and the SPE monitoring statistic values of many fault samples are below the related confidence limit. By contrast, from the NRJDICA-based monitoring chart of Fig. 5, it can be seen that the  $L^2$  monitoring statistic values of the fault samples exceed the confidence limit quickly and evidently, resulting in a much earlier detection of the fault. Table 1 compares the fault detection times (the sample numbers) and the fault detection rates of the three fault detection methods. The results obtained for this three-variable system clearly show that the NRJDICA-based fault detection method has a faster fault detection time and achieves a higher fault detection rate than the FastICA-based and KPCA-based fault detection methods.



**Fig. 4.** The monitoring charts of the FastICA-based fault detection method for the simple three-variable system. The fault occurs at the 101th sample. (a) The  $I^2$  monitoring chart. (b) The SPE monitoring chart.



**Fig. 5.** The monitoring chart of the NRJDICA-based fault detection method for the simple three-variable system. The fault occurs at the 101th sample.

#### 4.2. A continuous stirred tank reactor system

The CSTR system is commonly used for testing the fault detection methods [29,55]. The schematic diagram of the CSTR

with the cascade control system is depicted in Fig. 6. In this CSTR, the reactant A flows into the reactor, and the first-order irreversible reaction  $A \rightarrow B$  happens. The single component B is produced as an outlet stream. Heat from the exothermic reaction is taken away by the cooling flow of the jacket. The temperature of the reactor is controlled by manipulating the coolant flow, while the level is controlled by manipulating the outlet flow. Based on the mass, energy and component balances, the dynamic model of this CSTR system can be established as follows:

$$\frac{dC_A}{dt} = -k_0 e^{-E_0/RT} C_A + \frac{Q_F(C_{AF} - C_A)}{Ah}, \quad (30)$$

$$\frac{dT}{dt} = \frac{k_0 e^{-E_0/RT} C_A (-\Delta H)}{\rho C_p} + \frac{Q_F(T_F - T)}{Ah} + \frac{UA_C(T_C - T)}{\rho C_p Ah}, \quad (31)$$

$$\frac{dT_C}{dt} = \frac{Q_C(T_{CF} - T_C)}{V_C} + \frac{UA_C(T - T_C)}{\rho_C C_{pC} V_C}, \quad (32)$$

$$\frac{dh}{dt} = \frac{Q_F - Q}{A}. \quad (33)$$

The system parameters in Eqs. (30)–(33) are described in Table 2. The details of these parameters and the nominal operating conditions of the CSTR can be found in [55].

The ten process variables are measured and they are described in Table 3. The non-Gaussian measurement noise variables  $\varepsilon_i$  for  $1 \leq i \leq 10$  which obey the uniform distributions within  $[-\sqrt{3}\sigma_i, \sqrt{3}\sigma_i]$  are added to all the 10 measured variables in the simulation procedure, and the noise intensity in each measured variable is also given in Table 3. The simulation data are sampled every 2 seconds, and 4000 samples are generated by simulating the CSTR system under the normal operating condition which are divided into the training data with  $N_1 = 2000$  samples and the validating data with  $N_2 = 2000$  samples. Eight fault patterns listed in Table 4 are simulated, which can be divided into two types according to fault characteristics: the step-change fault type and

the ramp-change fault type. The faults 1–3 are step-change faults, while the faults 4–8 are ramp-change faults. The data for each fault pattern are recorded with 1000 samples and the fault is introduced at the 190th sample. For both the FastICA-based and NRJDICA-based fault detection methods, the number of the dominant ICs is set to  $p=5$ . The window width in Eq. (25) is chosen to be  $h=76$  for our NRJDICA-based fault detection method. Similarly, for the KPCA-based fault detection method, the number of the kernel PCs is also set to 5. Furthermore, the Gaussian kernel is used for the KPCA and the kernel width is chosen to be 30m according to the empirical criterion given in [22], where  $m$  is the number of the measured variables. Again, the  $\delta=99\%$  confidence limit is also employed as the alarming threshold.

The monitoring charts obtained by the three fault detection methods under the fault pattern 5 are shown in Figs. 7–9. From the monitoring chart of our NRJDICA-based fault detection method shown in Fig. 9, it can be seen that when the sample number is

**Table 2**  
The parameters of the CSTR system.

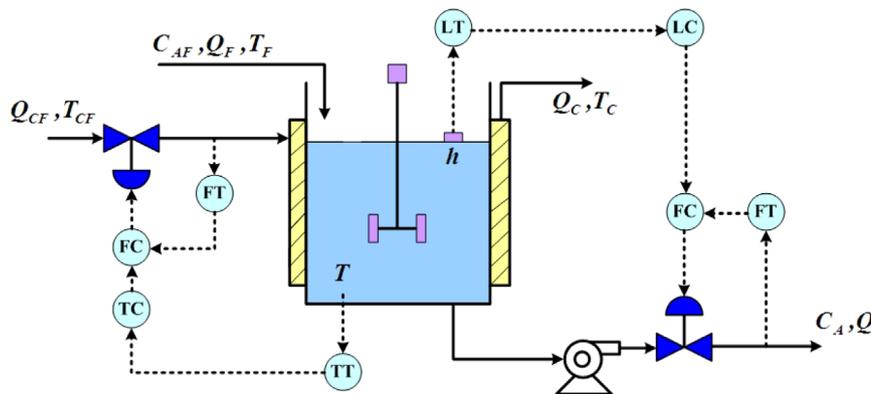
Parameter	Description
$k_0$	Preexponential factor
$E_0/R$	Activation energy
$A$	Reactor cross-sectional area
$\Delta H$	Reaction heat
$\rho$	Density of reactor contents
$C_p$	Heat capacity of reactor contents
$UA_C$	Heat-transfer coefficient
$V_C$	Capacity of cooling jacket
$C_{pC}$	Heat capacity of coolant
$\rho_C$	Density of coolant

**Table 3**  
The measured variables and the noise intensity in each measured variable for the CSTR system.

Measured variable	Variable description	Noise intensity (%)
$Q_F$	Reactor feed flow rate	17.16
$T_F$	Temperature of reactor feed stream	16.95
$C_{AF}$	Concentration of species A in reactor feed stream	6.61
$Q$	Reactor outlet flow rate	61.45
$T$	Reactor temperature	36.37
$C_A$	Concentration of species A in reactor	39.44
$T_{CF}$	Temperature of coolant feed	6.08
$T_C$	Temperature of coolant in cooling jacket	69.44
$Q_C$	Coolant flow rate	5.11
$h$	Reactor liquid level	7.31

**Table 1**  
The performance comparison of the three fault detection methods for the simple three-variable system.

Method	Monitoring statistic	Fault detection time	Fault detection rate (%)
KPCA	$T^2$	841	40.47
	SPE	538	62.74
FastICA	$I^2$	841	27.79
	SPE	537	64.42
NRJDICA	$L^2$	292	89.21

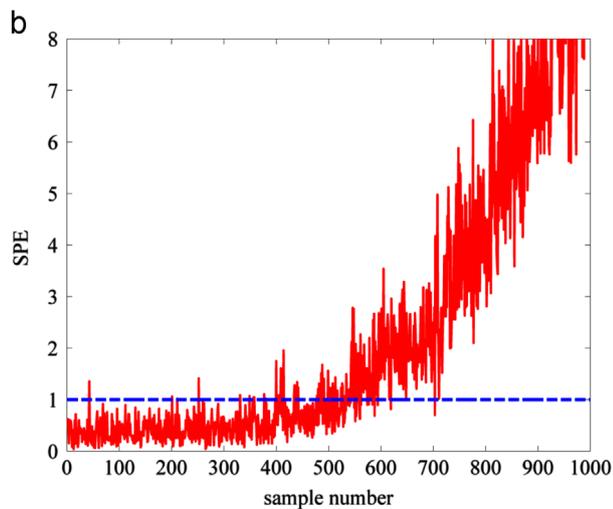
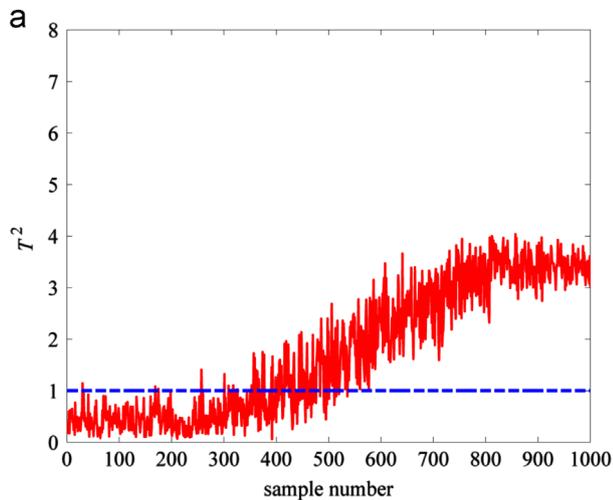


**Fig. 6.** The continuous stirred tank reactor with cascade control.

**Table 4**

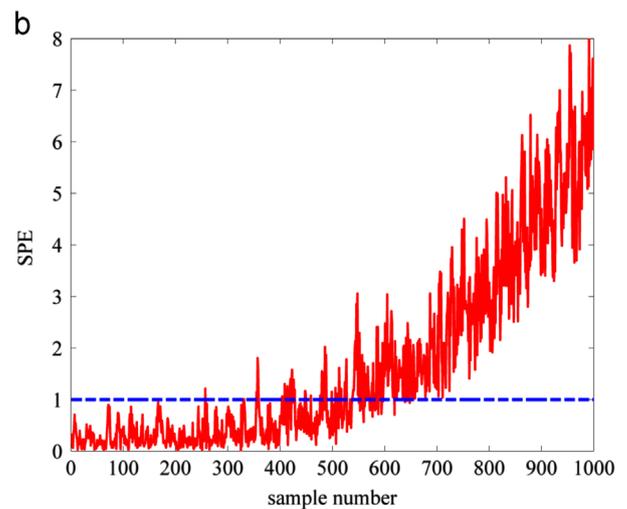
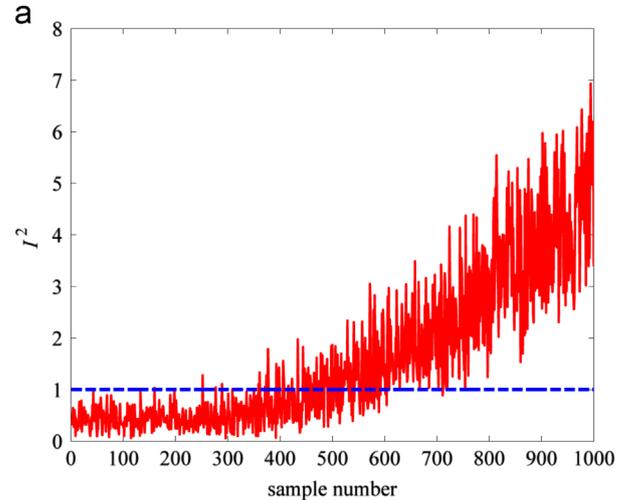
The simulated fault patterns for the CSTR system.

Fault	Description	Value
1	Step change in the reactor feed flow rate $Q_F$	-7 L/min
2	Set point change for the reactor temperature $T$	10 K
3	Bias in the measurement of the reactor temperature $T$	4 K
4	The reactor feed stream temperature $T_F$ ramps up with the ramp rate	0.3 K/min
5	The feed concentration $C_{AF}$ ramps up with the ramp rate	$7 \times 10^{-4}$ (mol/L)/min
6	The heat-transfer coefficient $UA_C$ ramps down with the ramp rate	-125 (J/(min K))/min
7	The catalyst activation energy $E_0/R$ ramps up with the ramp rate	6 K/min
8	The coolant feed temperature $T_{CF}$ ramps down with the ramp rate	-0.2 K/min



**Fig. 7.** The monitoring charts of the KPCA-based fault detection method for the CSTR system under the fault pattern 5. The fault occurs at the 190th sample. (a) The  $T^2$  monitoring chart. (b) The SPE monitoring chart.

greater than 290, all the  $L^2$  monitoring statistic values exceed the confidence limit and stay well above the confidence limit. By contrast, from the monitoring charts of the FastICA-based fault detection method depicted in Fig. 8, we observe that only when the sample numbers are greater than 500 and 530, respectively, can the  $I^2$  and SPE monitoring statistic values exceed their corresponding confidence limits clearly. Similarly, from the monitoring charts of the KPCA-based fault detection method depicted in Fig. 7, it can be seen that only when the sample numbers are greater than 410 and 560, respectively, can the  $T^2$  and SPE monitoring statistic values exceed their related confidence limits obviously. This illustrates that the NRJDICA-based fault detection



**Fig. 8.** The monitoring charts of the FastICA-based fault detection method for the CSTR system under the fault pattern 5. The fault occurs at the 190th sample. (a) The  $I^2$  monitoring chart. (b) The SPE monitoring chart.

method can detect the fault pattern 5 of the CSTR process much faster and much more effectively than the FastICA-based and KPCA-based fault detection methods.

The monitoring charts obtained by the three fault detection methods for the fault pattern 7 are also examined in Figs. 10–12. Compared with the monitoring charts of the FastICA-based fault detection method shown in Fig. 11, the monitoring chart of our NRJDICA-based fault detection method plotted in Fig. 12 reacts much more quickly and sharply to the occurring fault. Specifically, in the monitoring chart of the NRJDICA-based fault detection method, all the  $L^2$  monitoring statistic values exceed the confidence limit after the 260th sample. However, in the monitoring

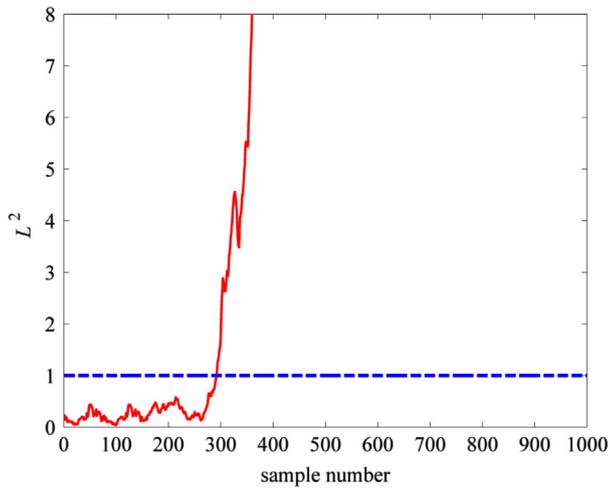


Fig. 9. The monitoring chart of the NRJDICA-based fault detection method for the CSTR system under the fault pattern 5. The fault occurs at the 190th sample.

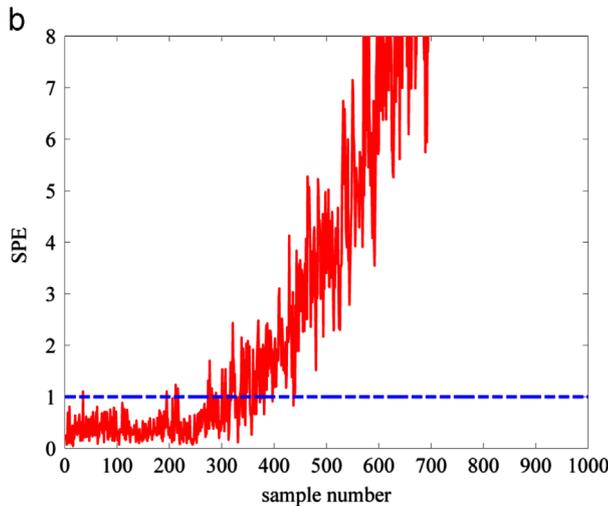
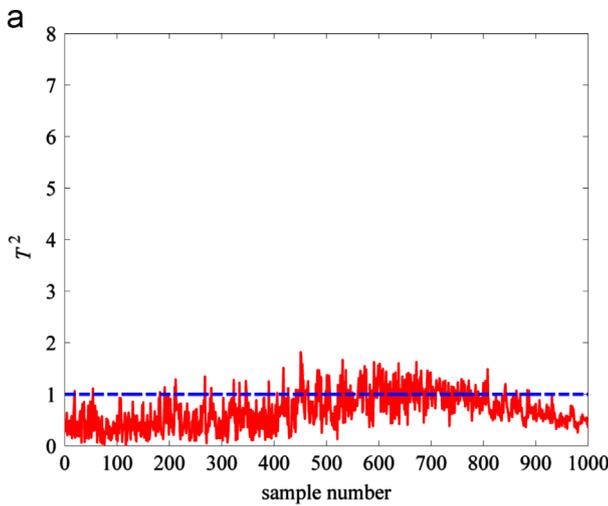


Fig. 10. The monitoring charts of the KPCA-based fault detection method for the CSTR system under the fault pattern 7. The fault occurs at the 190th sample. (a) The  $T^2$  monitoring chart. (b) The SPE monitoring chart.

charts of the FastICA-based fault detection method, most monitoring statistic values from the 260th sample to the 400th sample are still below the corresponding confidence limits, and thus cannot give a definite fault indication. In the monitoring charts of the

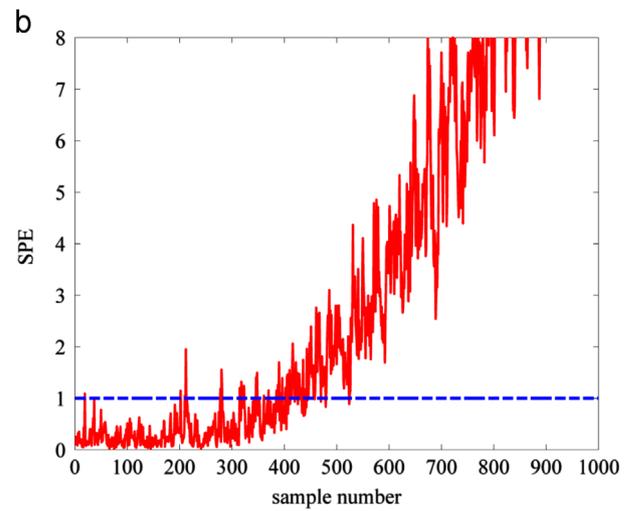
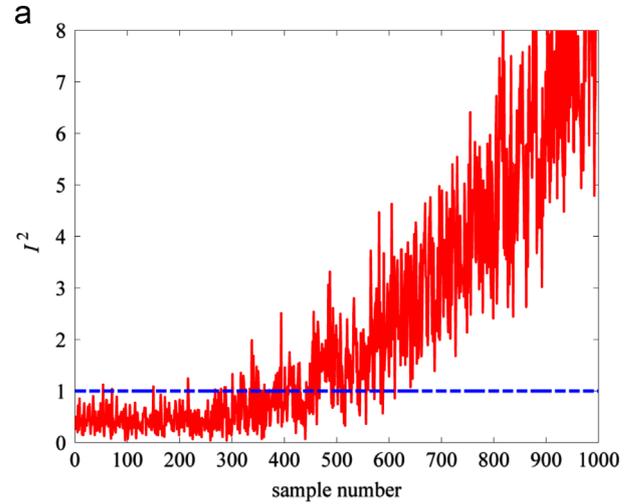


Fig. 11. The monitoring charts of the FastICA-based fault detection method for the CSTR system under the fault pattern 7. The fault occurs at the 190th sample. (a) The  $I^2$  monitoring chart. (b) The SPE monitoring chart.

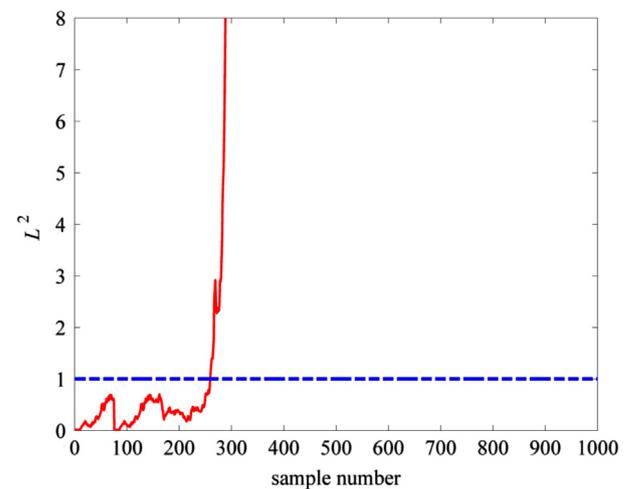


Fig. 12. The monitoring chart of the NRJDICA-based fault detection method for the CSTR system under the fault pattern 7. The fault occurs at the 190th sample.

KPCA-based fault detection method illustrated in Fig. 10, most of the  $T^2$  monitoring statistic values are below the related confidence limit and thus The KPCA-based  $T^2$  statistic fails to detect the fault pattern 7, while many of the SPE monitoring statistic values from

**Table 5**  
Comparison of the fault detection times (sample numbers) for the CSTR system.

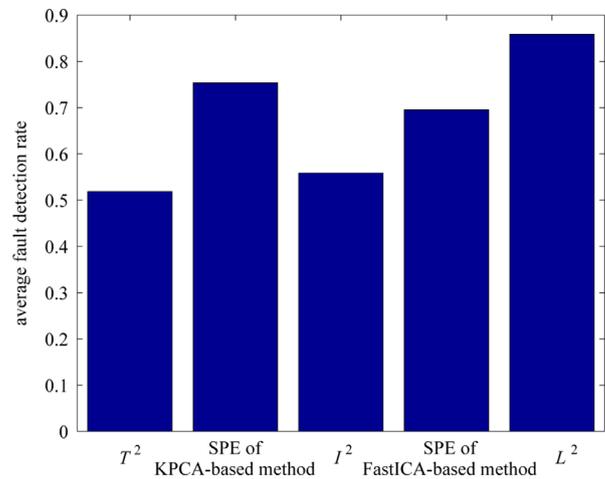
Fault no.	KPCA-based method		FastICA-based method		NRJDICA-based method $L^2$
	$T^2$	SPE	$I^2$	SPE	
1	Failed	190	190	191	193
2	Failed	190	219	190	191
3	190	190	251	190	193
4	383	366	511	379	294
5	416	563	485	539	292
6	448	396	Failed	382	379
7	571	319	409	412	260
8	560	386	528	515	305

**Table 6**  
Comparison of the fault detection rates for the CSTR system.

Fault no.	KPCA-based method (%)		FastICA-based method (%)		NRJDICA-based method (%) $L^2$
	$T^2$	SPE	$I^2$	SPE	
1	0.86	100.0	100.0	100.0	99.75
2	0.25	100.0	97.16	100.0	100.0
3	5.80	100.0	94.32	100.0	99.75
4	57.90	79.38	66.54	76.67	87.53
5	70.37	60.25	63.09	58.27	87.53
6	54.81	79.01	13.70	75.56	76.79
7	23.70	82.72	70.86	74.32	91.48
8	52.47	75.56	65.06	62.72	85.93

the 260th sample to the 330th sample are below the corresponding confidence limit. The results obtained again confirm that the NRJDICA-based fault detection method has better fault detection performance than the FastICA-based and KPCA-based fault detection methods.

We next investigate the achievable fault detection performance of the three fault detection methods on all the eight fault scenarios. The fault detection times and the fault detection rates obtained by the three methods are tabulated in Tables 5 and 6, respectively. From Table 5, it can be seen that all the three methods can detect the occurring fault immediately for the step-change fault patterns 1–3, although the  $T^2$  monitoring statistic of the KPCA-based fault detection method actually fails to detect the occurrence of the fault patterns 1 and 2. In contrast to the step-change faults, the ramp-change faults are more difficult to detect, as confirmed by the fault detection times for the ramp-change fault patterns 4–8. For the challenging problem of detecting the ramp-change faults, our NRJDICA-based method achieves much faster fault detection times than both the FastICA-based and KPCA-based fault detection methods, as can be seen in Table 5. Taking the fault 4 as an example, the  $I^2$  and SPE monitoring statistics of the FastICA-based fault detection method can only detect the fault at the 511th and 379th samples, respectively, while the  $T^2$  and SPE monitoring statistics of the KPCA-based fault detection method can only detect the fault at the 383th and 366th samples, respectively. But the  $L^2$  monitoring statistic of the NRJDICA-based method can detect the fault at the 294th sample. The results of Table 5 show the superior ability of the NRJDICA-based method in shortening the fault detection delay for the challenging ramp-change fault patterns 4–8, in contrast to the FastICA-based and KPCA-based methods. Similarly, from Table 6, it can be observed that all the three methods can achieve high fault detection rates equal or close to 1 for the step-change faults 1–3, which suggests that almost all the fault samples are successfully detected by the three fault detection methods, but again the  $T^2$  monitoring statistic



**Fig. 13.** Comparison of the average fault detection rates for the ramp-change fault patterns 4–8 of the CSTR process.

of the KPCA-based fault detection method has extremely low fault detection rates for the step-change faults 1–3 and thus fails to effectively and reliably detect these faults. For the challenging ramp-change faults 4–8, the NRJDICA-based method again attains higher fault detection rates than both the FastICA-based and KPCA-based methods, which demonstrates the superior fault detection ability of the NRJDICA-based fault detection method over the other two fault detection methods. The average fault detection rate of each monitoring statistic over the ramp-change faults 4–8 is illustrated in Fig. 13, which further confirms that the NRJDICA-based fault detection method outperforms both the KPCA-based and FastICA-based fault detection methods.

**5. Conclusions**

A novel NRJDICA-based fault detection method has been proposed in this paper. Our contribution is two-fold. Firstly, we have developed a NRJDICA algorithm to estimate the mixing matrix and the ICs, which explicitly takes the measurement noise into consideration while imposing no unrealistic assumptions on the measurement noise. Consequently, unlike the widely used FastICA algorithm which suffers from the adverse effects of the measurement noise, the proposed NRJDICA algorithm is capable of accurately estimating the mixing matrix and the ICs under highly noisy industrial environments. Secondly, we have constructed a new noise-restraining monitoring statistic, which can further effectively reduce the measurement noise's influence, for process fault detection. Simulation results obtained on the three-variable system and the CSTR system have demonstrated the superior fault detection performance of the proposed NRJDICA-based fault detection method, in terms of fault detection time and fault detection rate, over the FastICA-based and KPCA-based fault detection methods.

As a concluding remark, we note the three important issues which require further investigation. Firstly, the noise-restraining monitoring statistic  $L^2(t)$  is constructed based on the serial correlation information of the dominant ICs. How to make an effective use of the information containing in the noise-corrupted residuals for process monitoring purpose is a challenging problem which requires future research. Secondly, as industrial processes often exhibit nonlinear characteristics, our future study will consider nonlinear process behaviors in our proposed fault detection method to develop a kernel technique based generalization of the NRJDICA for nonlinear process fault detection. Thirdly, we implicitly assume that the monitored process is operating at one

mode in the current study. Our future work will investigate how to enhance our NRJDICA-based fault detection method to extend its application to multimode process fault detection problems.

### Appendix A. Derivation of the new $L^2$ monitoring statistic

Considering the extracted  $p$  zero-mean feature variables  $\hat{\mathbf{f}}_p(t) = [\hat{f}_1(t) \hat{f}_2(t) \cdots \hat{f}_p(t)]^T$ , the well-known Hotelling's  $T^2(t)$  monitoring statistic is expressed by [56]

$$T^2(t) = \hat{\mathbf{f}}_p^T(t) \Theta^{-1} \hat{\mathbf{f}}_p(t), \quad (34)$$

where the covariance matrix  $\Theta = E\{\hat{\mathbf{f}}_p(t) \hat{\mathbf{f}}_p^T(t)\}$  is estimated based on the training data collected from the process under the normal operation condition.

In particular, if the feature variables are extracted by an ICA, they are called the ICs, and we have  $\hat{\mathbf{f}}_p(t) = \hat{\mathbf{s}}_p(t)$  and the covariance matrix  $\Theta = \mathbf{I}_p$ . In this case Hotelling's  $T^2(t)$  monitoring statistic becomes

$$L^2(t) = \hat{\mathbf{s}}_p^T(t) \hat{\mathbf{s}}_p(t), \quad (35)$$

as given in Eq. (8). However, the estimated ICs  $\hat{\mathbf{s}}_p(t)$  are inevitably corrupted by the measurement noise, as can be seen clearly from Eq. (18).

The newly proposed monitoring statistic  $L^2(t)$  is also based on Hotelling's  $T^2(t)$  monitoring statistic. Specifically, instead of using the original extracted dominant IC vector  $\hat{\mathbf{s}}_p$  which may suffer from the adverse effects of the measurement noise, we use the serial correlation information  $\mathbf{r}_p(t)$  of  $\hat{\mathbf{s}}_p(t)$  which can effectively restrain the influence of the measurement noise. Using  $\mathbf{r}_p(t)$  and its covariance matrix  $\Psi$  in Eq. (34), we arrive at

$$L^2(t) = \mathbf{r}_p^T(t) \Psi^{-1} \mathbf{r}_p(t), \quad (36)$$

as introduced in Eq. (26).

### Appendix B. Empirical selection of the window width $h$

As currently there exists no method for determining the optimal window width  $h$ , this parameter is usually chosen by trial and error. Here, we present an empirical method to choose an appropriate value for  $h$ . We now explain the basic idea of this heuristic method. In the off-line modeling stage, the normal operating data are collected to form the training data set and the validating data set. Because the samples whose monitoring statistic values exceed the  $\delta$  confidence limit are considered as fault samples, we choose a value for the window width  $h$  to make the monitoring statistic values of the validating samples below the confidence limit as much as possible. By adopting this strategy, the region of normal operating conditions may be better preserved.

For the constructed noise-restraining monitoring statistic  $L^2$ , we define the index

$$\eta = \frac{R_1}{R_2}, \quad (37)$$

to measure the differences between the  $\delta$  confidence limit of the monitoring statistic  $L^2$  and the monitoring statistic values  $\{L^2(t)\}_{t=h}^{N_2}$  for the validating data. In Eq. (37),  $R_1$  denotes the number of the monitoring statistic values between  $L_{\text{lim},\delta}^2$  and  $L_{\text{lim},\delta}^2 - D$ , while  $R_2$  denotes the number of the monitoring statistic values lower than  $L_{\text{lim},\delta}^2 - D$ , where  $L_{\text{lim},\delta}^2$  is the  $\delta$  confidence limit of the  $L^2$  monitoring statistic and  $D$  is a predefined constant satisfying the condition  $D < L_{\text{lim},\delta}^2$ .

A smaller value of  $\eta$  suggests that more monitoring statistic values are far below the confidence limit and thus the region of normal

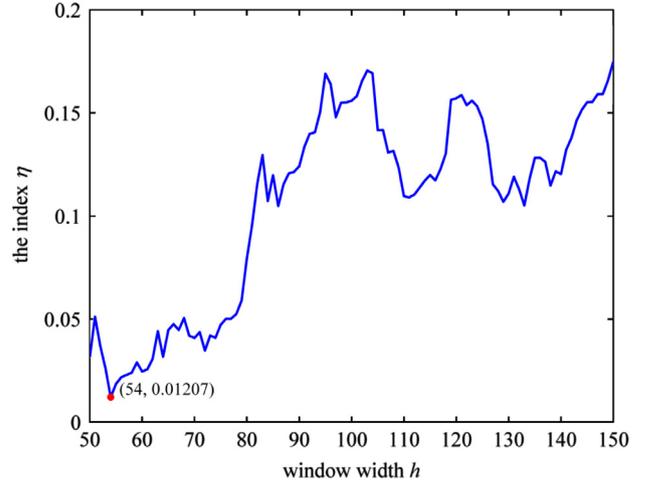


Fig. 14. The relationship between the index  $\eta$  and the window width  $h$  for the three-variable system.

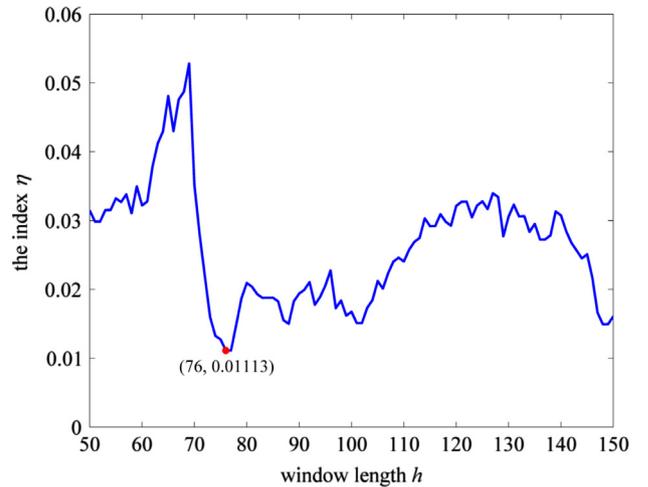


Fig. 15. The relationship between the index  $\eta$  and the window width  $h$  for the CSTR system.

operating conditions is better preserved. As the index  $\eta$  is directly influenced by the window width  $h$ ,  $\eta$  can be minimized to determine an appropriate value for  $h$  using the following search procedure.

- (1) Set the search range for  $h$  from  $h_{\min}$  to  $h_{\max}$  that covers all the possible choices of the window width. Choose the required  $\delta$  value and set another value  $\delta_0 < \delta$ .
- (2) Start with  $h = h_{\min}$ .
- (3) Calculate the monitoring statistic values  $\{L^2(t)\}_{t=h}^{N_2}$  for the validating data.
- (4) Determine the  $\delta$  confidence limit  $L_{\text{lim},\delta}^2$  for the  $L^2$  monitoring statistic. If  $h = h_{\min}$ : also determine the  $\delta_0$  confidence limit  $L_{\text{lim},\delta_0}^2$  for the  $L^2$  monitoring statistic, and set  $D = L_{\text{lim},\delta}^2 - L_{\text{lim},\delta_0}^2$ .
- (5) Calculate the index  $\eta$  of Eq. (37).
- (6) Set  $h = h + 1$ . If  $h \leq h_{\max}$  go back to step (3); otherwise, stop the procedure.

The value  $h$  in the search range  $h_{\min}$  to  $h_{\max}$  that approximately minimizes  $\eta$  may be chosen as the appropriate window width.

*The three-variable system:* A too small value of  $h$  may not effectively reduce the influence of the measurement noise in estimating the serial correlation information of the dominant ICs, while a too large value of  $h$  may significantly increase the

computational burden of the on-line fault detection. We set the search range for the window width from  $h_{\min} = 50$  to  $h_{\max} = 150$  as a good compromise. The values of  $\delta$  and  $\delta_0$  are set to 0.99 and 0.96, respectively. The index  $\eta$  as the function of the window width  $h$  is depicted in Fig. 14, where it can be found that  $h=54$  is an appropriate choice for the window width.

*The CSTR system:* The search range for  $h$  is also set from  $h_{\min} = 50$  to  $h_{\max} = 150$ , while the values of  $\delta$  and  $\delta_0$  are also set to 0.99 and 0.96, respectively. The index  $\eta$  as the function of the window width  $h$  is plotted in Fig. 15, where it can be seen that an appropriate choice for the window width is  $h=76$ .

### Appendix C. Determining the number of the dominant ICs

We use a scheme similar to the one discussed in [34] to determine an appropriate number  $p$  of the dominant ICs. The cumulative percent variance (CPV) criterion is commonly applied in the ICA-based fault detection methods [4,40]. We construct a CPV criterion using the absolute values of the ICs' kurtosis estimates given in Eq. (24) to help finding an appropriate number  $p$  of the dominant ICs. Assume that the estimated ICs have already been arranged in the descending order according to their non-Gaussian degrees measured by the absolute values of the ICs' kurtosis estimates. Then the CPV criterion is defined by

$$\text{CPV}(p) = \frac{\sum_{i=1}^p |\hat{k}_4(s_i)|}{\sum_{i=1}^m |\hat{k}_4(s_i)|} \times 100\%. \quad (38)$$

CPV( $p$ ) can be set to 90%, 95% or 99% to determine the corresponding value of  $p$ , as suggested in [4]. For our two case studies, CPV( $p$ ) = 90% is chosen to determine the number of the dominant ICs. For the three-variable system, this leads to  $p=1$ , while for the CSTR system, an appropriate number of the dominant ICs is found to be  $p=5$ . In order to conduct a fair comparison, we use the same  $p$  value for the FastICA-based fault detection method.

### References

- [1] Y. Zhang, T. Chai, Z. Li, C. Yang, Modeling and monitoring of dynamic processes, *IEEE Trans. Neural Netw. Learn. Syst.* 23 (February (2)) (2012) 277–284.
- [2] Y. Zhang, Fault detection and diagnosis of nonlinear processes using improved kernel independent component analysis (KICA) and support vector machine (SVM), *Ind. Eng. Chem. Res.* 47 (August (18)) (2008) 6961–6971.
- [3] Y. Zhang, J. An, H. Zhang, Monitoring of time-varying processes using kernel independent component analysis, *Chem. Eng. Sci.* 88 (January) (2013) 23–32.
- [4] Y.W. Zhang, Y. Zhang, Fault detection of non-Gaussian processes based on modified independent component analysis, *Chem. Eng. Sci.* 65 (August (16)) (2010) 4630–4639.
- [5] C. Zhao, F. Gao, F. Wang, Nonlinear batch process monitoring using phase-based kernel-independent component analysis-principal component analysis (KICA-PCA), *Ind. Eng. Chem. Res.* 48 (August (20)) (2009) 9163–9174.
- [6] C. Zhao, F. Gao, F. Wang, An improved independent component regression modeling and quantitative calibration procedure, *AIChE J.* 56 (June (6)) (2010) 1519–1535.
- [7] F. Wang, S. Tan, J. Peng, Y. Chang, Process monitoring based on mode identification for multi-mode process with transitions, *Chemom. Intell. Lab. Syst.* 110 (January (1)) (2012) 144–155.
- [8] L. Wang, H. Shi, Multivariate statistical process monitoring using an improved independent component analysis, *Chem. Eng. Res. Des.* 88 (April (4)) (2010) 403–414.
- [9] H. Ma, Y. Hu, H. Shi, A novel local neighborhood standardization strategy and its application in fault detection of multimode processes, *Chemom. Intell. Lab. Syst.* 118 (August) (2012) 287–300.
- [10] Z. Ge, Z. Song, Performance-driven ensemble learning ICA model for improved non-Gaussian process monitoring, *Chemom. Intell. Lab. Syst.* 123 (April) (2013) 1–8.
- [11] Z. Ge, Z. Song, Mixture Bayesian regularization method of PPCA for multimode process monitoring, *AIChE J.* 56 (November (11)) (2010) 2838–2849.
- [12] Z. Ge, Z. Song, Kernel generalization of PPCA for nonlinear probabilistic monitoring, *Ind. Eng. Chem. Res.* 49 (September (22)) (2010) 11832–11836.
- [13] M. Zhang, Z. Ge, Z. Song, R. Fu, Global-local structure analysis model and its application for fault detection and identification, *Ind. Eng. Chem. Res.* 50 (April (11)) (2011) 6837–6848.
- [14] J. Zhu, Z. Ge, Z. Song, Robust modeling of mixture probabilistic principal component analysis and process monitoring application, *AIChE J.* 60 (June (6)) (2014) 2143–2157.
- [15] S.J. Qin, Statistical process monitoring: basics and beyond, *J. Chemom.* 17 (8–9) (2003) 480–502.
- [16] D. Kim, I.-B. Lee, Process monitoring based on probabilistic PCA, *Chemom. Intell. Lab. Syst.* 67 (August (2)) (2003) 109–123.
- [17] D.S. Kim, C.K. Yoo, Y. Kim, J.H. Jung, I.-B. Lee, Calibration, prediction and process monitoring model based on factor analysis for incomplete process data, *J. Chem. Eng. Jpn.* 38 (12) (2005) 1025–1034.
- [18] Q. Jiang, X. Yan, Probabilistic monitoring of chemical processes using adaptively weighted factor analysis and its application, *Chem. Eng. Res. Des.* 92 (January (1)) (2014) 127–138.
- [19] Q. Jiang, X. Yan, Chemical processes monitoring based on weighted principal component analysis and its application, *Chemom. Intell. Lab. Syst.* 119 (October) (2012) 11–20.
- [20] T. Chen, Y. Sun, Probabilistic contribution analysis for statistical process monitoring: a missing variable approach, *Control Eng. Pract.* 17 (April (4)) (2009) 469–477.
- [21] M.A. Bin Shams, H.M. Budman, T.A. Duever, Fault detection identification and diagnosis using CUSUM based PCA, *Chem. Eng. Sci.* 66 (October (20)) (2011) 4488–4498.
- [22] J.-M. Lee, C.K. Yoo, S.W. Choi, P.A. Vanrolleghem, I.-B. Lee, Nonlinear process monitoring using kernel principal component analysis, *Chem. Eng. Sci.* 59 (January (1)) (2004) 223–234.
- [23] J.-M. Lee, S.J. Qin, I.-B. Lee, Fault detection and diagnosis based on modified independent component analysis, *AIChE J.* 52 (August (10)) (2006) 3501–3514.
- [24] J.-M. Lee, S.J. Qin, I.-B. Lee, Fault detection of non-linear process using kernel independent component analysis, *Can. J. Chem. Eng.* 85 (August (4)) (2007) 526–536.
- [25] J.-M. Lee, C. Yoo, I.-B. Lee, Statistical process monitoring with independent component analysis, *J. Process Control* 14 (August (5)) (2004) 467–485.
- [26] M.M. Rashid, J. Yu, A new dissimilarity method integrating multidimensional mutual information and independent component analysis for non-Gaussian dynamic process monitoring, *Chemom. Intell. Lab. Syst.* 115 (June) (2012) 44–58.
- [27] J. Yu, A nonlinear kernel Gaussian mixture model based inferential monitoring approach for fault detection and diagnosis of chemical processes, *Chem. Eng. Sci.* 68 (January (1)) (2012) 506–519.
- [28] M.M. Rashid, J. Yu, Hidden Markov model based adaptive independent component analysis approach for complex chemical process monitoring and fault detection, *Ind. Eng. Chem. Res.* 51 (March (15)) (2012) 5506–5514.
- [29] M. Kano, S. Tanaka, S. Hasebe, I. Hashimoto, H. Ohno, Monitoring independent components for fault detection, *AIChE J.* 49 (April (4)) (2003) 969–976.
- [30] L. Cai, X. Tian, S. Chen, A process monitoring method based on noisy independent component analysis, *Neurocomputing* 127 (March) (2014) 231–246.
- [31] L. Cai, X. Tian, A new fault detection method for non-Gaussian process based on robust independent component analysis, *Process Saf. Environ. Prot.* (2014), <http://dx.doi.org/10.1016/j.psep.2013.11.003>, in press.
- [32] L. Cai, X. Tian, N. Zhang, Process fault detection method using time-structure KICA and OCSVM, *J. Tsinghua Univ. (Sci. Technol.)* 52 (September (9)) (2012) 1205–1209 (in Chinese).
- [33] L. Cai, X. Tian, N. Zhang, Non-Gaussian process fault detection method based on modified KICA, *CIESC J.* 63 (9) (2012) 2864–2868 (in Chinese).
- [34] X. Tian, X. Zhang, X. Deng, S. Chen, Multiway kernel independent component analysis based on feature samples for batch process monitoring, *Neurocomputing* 72 (March (7–9)) (2009) 1584–1596.
- [35] X. Tian, L. Cai, Process fault detection method based on KICA-GMM, *CIESC J.* 63 (9) (2012) 2859–2863 (in Chinese).
- [36] C.-C. Hsu, L. Chen, C.-H. Liu, A process monitoring scheme based on independent component analysis and adjusted outliers, *Int. J. Prod. Res.* 48 (6) (2010) 1727–1743.
- [37] J. Lee, B. Kang, S.-H. Kang, Integrating independent component analysis and local outlier factor for plant-wide process monitoring, *J. Process Control* 21 (August (7)) (2011) 1011–1021.
- [38] G. Stefanos, A.B. Hamza, Dynamic independent component analysis approach for fault detection and diagnosis, *Expert Syst. Appl.* 37 (December (12)) (2010) 8606–8617.
- [39] P.P. Odiwoei, Y. Cao, State-space independent component analysis for nonlinear dynamic process monitoring, *Chemom. Intell. Lab. Syst.* 103 (August (1)) (2010) 59–65.
- [40] J.C. Wang, Y.B. Zhang, H. Cao, W.Z. Zhu, Dimension reduction method of independent component analysis for process monitoring based on minimum mean square error, *J. Process Control* 22 (February (2)) (2012) 477–487.
- [41] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, *Neural Netw.* 13 (4–5) (2003) 411–430.
- [42] A. Hyvärinen, Survey on independent component analysis, *Neural Comput. Surv.* 2 (4) (1999) 94–128.
- [43] A. Hyvärinen, Gaussian moments for noisy independent component analysis, *IEEE Signal Process. Lett.* 6 (June (6)) (1999) 145–147.
- [44] Y. Yang, C. Guo, Gaussian moments for noisy unifying model, *Neurocomputing* 71 (October (16–18)) (2008) 3656–3659.
- [45] L. Tong, R.W. Liu, V.C. Soon, Y.-F. Huang, Indeterminacy and identifiability of blind identification, *IEEE Trans. Circuits Syst.* 38 (May (5)) (1991) 499–509.

- [46] W. Lu, J.C. Rajapakse, Approach and applications of constrained ICA, *IEEE Trans. Neural Netw.* 16 (January (1)) (2005) 203–212.
- [47] W. Liu, D.P. Mandic, A. Cichocki, Blind second-order source extraction of instantaneous noisy mixtures, *IEEE Trans. Circuits Syst. II: Express Briefs* 53 (September (9)) (2006) 931–935.
- [48] R. Boscolo, H. Pan, V.P. Roychowdhury, Independent component analysis based on nonparametric density estimation, *IEEE Trans. Neural Netw.* 15 (January (1)) (2004) 55–65.
- [49] D. Wang, Robust data-driven modeling approach for real-time final product quality prediction in batch process operation, *IEEE Trans. Ind. Inform.* 7 (May (2)) (2011) 371–377.
- [50] A. Yeredor, Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation, *IEEE Trans. Signal Process.* 50 (July (7)) (2002) 1545–1553.
- [51] K.P. Balanda, H.L. MacGillivray, Kurtosis: a critical review, *Am. Stat.* 42 (May (2)) (1988) 111–119.
- [52] X.D. Zhang, *Time Series Analysis High Order Statistics Method*, Tsinghua University Press, Beijing, 1996, pp. 20–24 (in Chinese).
- [53] E.L. Russell, L.H. Chiang, R.D. Braatz, Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis, *Chemom. Intell. Lab. Syst.* 50 (May (1)) (2000) 81–93.
- [54] S. Mahadevan, S.L. Shah, Fault detection and diagnosis in process data using one-class support vector machines, *J. Process Control* 19 (December (10)) (2009) 1627–1639.
- [55] M.C. Johannesmeyer, A. Singhal, D.E. Seborg, Pattern matching in historical data, *AIChE J.* 48 (September (9)) (2002) 2022–2038.
- [56] T. Kourti, J.F. MacGregor, Process analysis monitoring and diagnosis using multivariate projection methods, *Chemom. Intell. Lab. Syst.* 28 (April (1)) (1995) 3–21.
- [57] M.L. Zhang, S.Z. Wang, X.Y. Jia, RLS adaptive noise cancellation via QR decomposition for noisy ICA, *Adv. Mater. Res.* 403–408 (2012) 1291–1296.



**Lianfang Cai** was born in Shandong, China, on September 30, 1986. He received his Bachelor degree in 2009 from China University of Petroleum.

He is currently a Ph.D. candidate in College of Information and Control Engineering, China University of Petroleum, Qingdao, China. His current research interests focus on fault detection and diagnosis in industrial processes.



**Sheng Chen** received his B.Eng. degree from the East China Petroleum Institute, China, in January 1982, and his Ph.D. degree from the City University, London, in September 1986, both in control engineering. In 2005, he was awarded the higher doctorate degree, Doctor of Sciences (D.Sc.), from the University of Southampton, Southampton, UK.

From 1986 to 1999, he held research and academic appointments at the Universities of Sheffield, Edinburgh and Portsmouth, all in UK. Since 1999, he has been with Electronics and Computer Science, the University of Southampton, UK, where he currently holds the post of Professor in Intelligent Systems and Signal

Processing. Dr. Chen's research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, intelligent control system design, evolutionary computation methods and optimization. He has published over 500 research papers.

Dr. Chen is a Fellow of IEEE and a Fellow of IET. He is a Distinguished Adjunct Professor at the King Abdulaziz University, Jeddah, Saudi Arabia. He is an ISI highly cited researcher in the engineering category (March 2004). He is elected as a Fellow of the United Kingdom Royal Academy of Engineering.



**Xuemin Tian** received his Bachelor of Engineering from Huadong Petroleum Institute, Dongying, China, in January 1982, and his M.S. degree from Beijing University of Petroleum, Beijing, China, in June 1994. From September 2001 to June 2002, he served as a visiting professor at Centre of Process Control, University of California in Santa Barbara.

He is a professor of Process Control at China University of Petroleum (Hua Dong). Professor Tian's research interests are in modeling, advanced process control and optimization for petrol-chemical processes as well as fault detection and diagnosis, and process monitoring.