Modified kernel principal component analysis based on local structure analysis and its application to nonlinear process fault diagnosis

Xiaogang Deng [a], Xuemin Tian [a], Sheng Chen [b,c],*

[a] College of Information and Control Engineering, China University of Petroleum (East China), Qingdao, Shandong 266555, China
[b] Electronics and Computer Science, Faculty of Physical and Applied Sciences, University of Southampton, Southampton SO17 1BJ, UK
[c] Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia

ARTICLE INFO

ABSTRACT

Traditional kernel principal component analysis (KPCA) concentrates on the global structure analysis of data sets but omits the local information which is also important for process monitoring and fault diagnosis. In this paper, a modified KPCA, referred to as the local KPCA (LKPCA), is proposed based on local structure analysis for nonlinear process fault diagnosis. In order to extract data feature better, local structure analysis is integrated within the KPCA, and this results in a new optimisation objective which naturally involves both global and local structure information. With the application of usual kernel trick, the optimisation problem is transformed into a generalised eigenvalue decomposition on the kernel matrix. For the purpose of fault detection, two monitoring statistics, known as the T^2 and Q statistics, are built based on the LKPCA model and confidence limit is computed by kernel density estimation. In order to identify fault variables, contribution plots for monitoring statistics are constructed based on the idea of sensitivity analysis to locate the fault variables. Simulation using the Tennessee Eastman benchmark process shows that the proposed method outperforms the traditional KPCA, in terms of fault detection performance. The results obtained also demonstrate the potential of the proposed fault identification approach.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The demands for improving product quality and ensuring process safety have stimulated the recent development of fault diagnosis techniques. As large amounts of data are available in modern process industry, data-driven methods based on the statistical process control theory have been one of the most fascinating topics in the process fault diagnosis field. Principal component analysis (PCA) is a classical data-driven multivariate statistical method which has attracted much attention from researchers [1–3]. However, PCA is a linear projection method, which cannot effectively capture the nonlinear features existing in real industrial processes. In order to cope with this problem, many modified nonlinear PCA methods have been developed. Krammer [4] first studied a nonlinear PCA based on an auto-associative neural network. Dong and MacAvoy [5] proposed the principal curve method as a nonlinear generalisation of linear PCA. Hiden et al. [6] suggested a nonlinear PCA using genetic programming, while Geng and Zhu [7] presented an adaptive nonlinear PCA based on an improved input training neural network.

More recently, the kernel PCA (KPCA) method has gained considerable interests in various research fields. KPCA was firstly proposed in [8], which applies a kernel function to compute the nonlinear principal components. This method has been applied to process fault detection and diagnosis. Lee and co-authors [9,10] proposed the KPCA-based T^2 and Q statistics for the fault detection of continuous and batch processes. Cho et al. [11] and Choi et al. [12] formulated two fault identification strategies using KPCA. In order to analyse multi-scale data, multi-scale KPCA methods were studied by combining the wavelet analysis and ensemble empirical mode decomposition [13–16]. To improve the computation efficiency, Tian et al. [17] and Cui et al. [18] used feature sample selection to reduce the computational complexity of calculating the kernel matrix in KPCA. Nguyen and Golinval [19] applied the KPCA to detect mechanical system faults by comparing the subspace angle between a reference and the current state. Considering the dynamic property of process data, Jia and co-authors [20] developed a dynamic KPCA method by integrating the kernel PCA and ARMAX time series model, while Zhang et al. [21] discussed a multi-block KPCA for large-scale processes. For effective monitoring of nonlinear and nonstationary processes, the work [22] proposed a variable window adaptive KPCA based on a fast block adaptation. Cao et al. [23] formulated the modularity of kernel methods for chemical data modelling, while Fu et al. [24] applied KPCA to capture the latent structure of training data for building a two-step nonlinear classification algorithm. It can be seen

* Corresponding author at: Electronics and Computer Science, Faculty of Physical and Applied Sciences, University of Southampton, Southampton SO17 1BJ, UK.
E-mail address: sqc@ecs.soton.ac.uk (S. Chen).

that KPCA offers a promising tool for nonlinear process fault detection and diagnosis.

The basic idea of KPCA is to map the input space onto a high-dimensional feature space via a nonlinear mapping and then to project the data along the directions of maximal variances in feature space. According to this principle, KPCA may be viewed as a global structure analysis technique because it does not consider the inner relationship among different data points. In other words, KPCA is a nonlinear data dimension reduction method which focuses on the global structure information and ignores the detailed local structure information in a data set. However, the local structure information is also important for data mining and feature extraction. Recently, local structure analysis methods, such as the locally linear embedding (LLE) [25], Laplacian eigenmaps (LE) [26], and locality preserving projections (LPP) [27,28], have been proposed in the study of manifold learning and have been proven to be powerful for data mining and process monitoring. Specifically, Zhang et al. [29] presented an LLE-based sensor fault detection method, while Li and Zhang [30] proposed to use the supervised LLE projection for machinery fault diagnosis. Jiang and co-authors [31,32] applied the modified LE method for fault pattern classification, while Hu and Yuan [33] proposed the multiway LPP (MLPP) method for batch process monitoring and demonstrated that the MLPP outperforms the conventional multiway PCA. Shao et al. [34] introduced a nonlinear fault diagnosis based on the generalized LPP method which imposes orthogonality constraints on the projection vectors, while Yu [35,36] used the LPP for bearing performance degradation assessment, combing with an exponential weighted moving average statistic and Gaussian mixture models, respectively. Furthermore, the work [37] proposed a global–local structure analysis model, combining the advantages of LPP and PCA. However, this method is inherently a linear transformation and it does not consider the nonlinearity of industrial processes. It can be seen that, although there exist a large number of studies on local structure analysis, no nonlinear monitoring method has been proposed to consider both the global and local data information analysis.

Motivated by the above discussion, we propose a modified KPCA method for nonlinear process fault diagnosis by introducing local structure analysis. The resulting modified KPCA, referred to as the local KPCA (LKPCA), performs both the global and local data structure analysis simultaneously. The main difference between the LKPCA and the standard KPCA lies in the design of the LKPCA optimisation objective which introduces local data structure analysis into the global optimisation of KPCA. With the application of usual kernel trick, the LKPCA optimisation is solved by the generalised eigenvalue decomposition of kernel matrix, and two monitoring statistics are built based on the proposed LKPCA model for fault detection. To tackle the challenging problem of nonlinear data-driven fault identification, we propose a novel construction of the contribution plot for the LKPCA to identify fault variables. Simulation results obtained on the Tennessee Eastman benchmark process demonstrate that the proposed LKPCA method performs better than the standard KPCA method, in terms of fault detection. The applicability of the proposed new fault identification scheme is also demonstrated in the simulation study. The reminder of this paper is organised as follows. We provide a brief review of the standard KPCA in Section 2. The proposed LKPCA method is detailed in Section 3. In Section 4, we present the process monitoring strategy using the proposed LKPCA, while the simulation study using the Tennessee Eastman benchmark process is provided in Section 5. Our conclusions are offered in Section 6.

2. Kernel principal component analysis

KPCA [8] is an unsupervised learning method, which first maps the data in the original input space onto a high-dimensional feature space

via a nonlinear mapping and then executes linear PCA in the resulting feature space. Specifically, denote $X_n = [x_1 \ x_2 \cdots x_n]^T \in \mathbb{R}^{n \times m}$ as the training data matrix, with the n samples of the process data vectors $x_i \in \mathbb{R}^{m \times 1}$, $1 \leq i \leq n$. A nonlinear mapping $\phi : x \in \mathbb{R}^{m \times 1} \rightarrow \phi(x) \in \mathcal{F}$ maps the training data onto a high-dimensional feature space $\mathcal{F}$. Linear PCA is then executed in $\mathcal{F}$ with the objective of finding a projection vector f so that the linear transformation

$$t = (\phi(x))^T f \tag{1}$$

has the maximum variance for the training data X_n. Assume that the feature vector $\phi(x_i)$ has been mean centred and variance scaled. Further denote $t_i = (\phi(x_i))^T f$. Then, this optimisation task can be formulated as

$$
\begin{aligned}
max \quad J_{KPCA}(f) &= max \frac{1}{n-1} \sum_{i=1}^{n} t_i^2 \\
&= max \frac{1}{n-1} \sum_{i=1}^{n} \left((\phi(x_i))^T f \right)^2,
\end{aligned} \tag{2}
$$
$$s.t. \quad f^T f = 1.$$

The projection vector f is also called the loading vector and $t_n = [t_1 \ t_2 \cdots t_n]^T$ the score vector of the training data matrix X_n. It is well known that there exist the coefficients $\alpha_j, 1 \leq j \leq n$, such that the loading vector takes the form

$$f = \sum_{j=1}^{n} \alpha_j \phi(x_j). \tag{3}$$

Substituting Eq. (3) into Eq. (2) leads to the optimisation problem

$$
\begin{aligned}
max \quad &J_{KPCA}(\alpha) \\
&= max \frac{1}{n-1} \sum_{i=1}^{n} \left((\phi(x_i))^T \sum_{j=1}^{n} \alpha_j \phi(x_j) \right)^2,
\end{aligned} \tag{4}
$$
$$s.t. \quad \left(\sum_{i=1}^{n} \alpha_i \phi(x_i) \right)^T \left(\sum_{i=1}^{n} \alpha_i \phi(x_i) \right) = 1,$$

where $\alpha = [\alpha_1 \ \alpha_2 \cdots \alpha_n]^T \in \mathbb{R}^{n \times 1}$.

To avoid the difficulty of explicitly defining the nonlinear high-dimensional mapping, the usual kernel trick is applied. With the introduction of a kernel function $k(x, x_j) = (\phi(x))^T \phi(x_j) = k(x_j, x)$, which is symmetric, the optimisation (4) can be expressed as

$$max J_{KPCA}(\alpha) = max \frac{1}{n-1} \sum_{i=1}^{n} \left(\sum_{j=1}^{n} \alpha_j k(x_i, x_j) \right)^2, \tag{5}$$
$$s.t. \quad \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) = 1.$$

There exist a number of kernel functions that can be utilised for this purpose, including polynomial kernel, sigmoid kernel and radial basis kernel. By defining the kernel matrix K whose (i,j)-th element is $[K]_{i,j} = k(x_i, x_j)$, we can further express Eq. (5) as

$$max J_{KPCA}(\alpha) = max \frac{1}{n-1} \alpha^T K K \alpha, \tag{6}$$
$$s.t. \quad \alpha^T K \alpha = 1.$$

The optimal coefficient vector α can be obtained by solving the eigenvector problem:

$$KK\alpha = (n-1)\lambda K\alpha, \tag{7}$$

with the normalised eigenvectors that satisfy $\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\alpha} = 1$. The solutions of Eq. (7) can be found by solving the equivalent problem:

$$\boldsymbol{K}\boldsymbol{\alpha} = (n-1)\lambda\boldsymbol{\alpha}. \tag{8}$$

Solving the eigenvalue problem (8) yields the $\widetilde{n}$ nonzero eigenvalues $\lambda^{1)} \geq \lambda^{2)} \geq \cdots \geq \lambda^{\widetilde{n})} > 0$ with the corresponding eigenvectors $\boldsymbol{\alpha}^{i)}$, $1 \leq i \leq \widetilde{n}$, where all the eigenvectors have been normalised to satisfy $(\boldsymbol{\alpha}^{i)})^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\alpha}^{i)} = 1$, and the $n - \widetilde{n}$ zero eigenvalues $\lambda^{\widetilde{n}+1)} = \cdots = \lambda^{n)} = 0$. The dimensionality of the problem can be reduced by only retaining the first $\widetilde{n}_s (\leq \widetilde{n})$ eigenvectors.

After the KCPA training, the test scores $\bar{t}_i$, $1 \leq i \leq \widetilde{n}$ or $1 \leq i \leq \widetilde{n}_s$, for the new data $\boldsymbol{x}$ can be extracted by projecting $\boldsymbol{\phi}(\boldsymbol{x})$ onto the i-th loading vector $\boldsymbol{f}^{i)} = \sum_{j=1}^{n} \alpha_j^{i)}\boldsymbol{\phi}(\boldsymbol{x}_j)$ in the feature space:

$$\bar{t}_i = (\boldsymbol{\phi}(\boldsymbol{x}))^{\mathrm{T}}\boldsymbol{f}^{i)} = \sum_{j=1}^{n} \alpha_j^{i)}(\boldsymbol{\phi}(\boldsymbol{x}))^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_j) = \boldsymbol{k}_{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{\alpha}^{i)}, \tag{9}$$

where $\boldsymbol{k}_{\boldsymbol{x}} = [k(\boldsymbol{x},\boldsymbol{x}_1)k(\boldsymbol{x},\boldsymbol{x}_2)\cdots k(\boldsymbol{x},\boldsymbol{x}_n)]^{\mathrm{T}} \in \mathbb{R}^{n \times 1}$ and $\boldsymbol{\alpha}^{i)} = [\alpha_1^{i)}\alpha_2^{i)} \cdots \alpha_n^{i)}]^{\mathrm{T}}$.

3. Local kernel principal component analysis

It is clear that the KPCA with the objective of explaining the maximum variance of the training data does not consider any inner relationship among neighbourhood data points. Therefore, KPCA can be viewed as a global data structure analysis technique. As discussed in the Introduction section, many manifold learning methods, such as the LLE [25], the LE [26] and the LPP [27,28], have been proposed for local structure analysis. A common similarity among these methods is that they all analyse the inner relationships of data points and aim at preserving the local neighbourhood structure during data transformation and feature extraction. This motivates us to introduce local structure information into the KPCA optimisation.

3.1. Local structure analysis

The high-dimensional data $\boldsymbol{\phi}(\boldsymbol{x})$ typically lies on a low-dimensional manifold embedded in the ambient space, and a local structure analysis is to find the optimal linear approximation (1) that makes the neighbouring points to stay as close together as possible. In other words, if $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and $\boldsymbol{\phi}(\boldsymbol{x}_j)$ are close, then $t_i = (\boldsymbol{\phi}(\boldsymbol{x}_i))^{\mathrm{T}}\boldsymbol{f}$ and $t_j = (\boldsymbol{\phi}(\boldsymbol{x}_j))^{\mathrm{T}}\boldsymbol{f}$ should be close as well. A reasonable criterion for choosing the projection vector $\mathbf{f}$ is therefore to minimise the following objective function $J_{\mathrm{LSA}}(\boldsymbol{f})$ under the appropriate constraints [27]

$$minJ_{\mathrm{LSA}}(\boldsymbol{f}) = min\sum_{i=1}^{n}\sum_{j=1}^{n}\left(t_i - t_j\right)^2 w_{i,j}$$
$$= min\sum_{i=1}^{n}\sum_{j=1}^{n}\left((\boldsymbol{\phi}(\boldsymbol{x}_i))^{\mathrm{T}}\boldsymbol{f} - \left(\boldsymbol{\phi}\left(\boldsymbol{x}_j\right)\right)^{\mathrm{T}}\boldsymbol{f}\right)^2 w_{i,j}, \tag{10}$$

where $w_{i,j}$ is a weighting parameter which incurs a heavy penalty if the neighbouring points $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and $\boldsymbol{\phi}(\boldsymbol{x}_j)$ are mapped far apart.

The weight parameters $w_{i,j}$ represent the neighbourhood relationships between different data points, and they may be determined based on the adjacency graph. In the adjacency graph, if nodes $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and $\boldsymbol{\phi}(\boldsymbol{x}_j)$ are neighbouring points, they are connected by an edge and the corresponding weight value on this edge is $w_{i,j}$. By contrast, if these two nodes are not neighbouring points, there is no edge between them and the weight value is set to zero. The definition of "neighbourhood" can be expressed in two ways: K-nearest neighbours (KNN) and ε neighbours. The former puts a neighbourhood edge between $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and $\boldsymbol{\phi}(\boldsymbol{x}_j)$ if $\boldsymbol{\phi}(\boldsymbol{x}_i)$ is among the K nearest neighbours of $\boldsymbol{\phi}(\boldsymbol{x}_j)$ or $\boldsymbol{\phi}(\boldsymbol{x}_j)$ is among the K nearest neighbours of $\boldsymbol{\phi}(\boldsymbol{x}_i)$,

while the latter connects an edge between $\boldsymbol{\phi}(\boldsymbol{x}_i)$ and $\boldsymbol{\phi}(\boldsymbol{x}_j)$ if $\|\boldsymbol{\phi}(\boldsymbol{x}_i) - \boldsymbol{\phi}(\boldsymbol{x}_j)\|^2 \leq \varepsilon$, where ε is a neighbourhood relationship threshold. Again, the kernel trick allows us to evaluate the Euclidean distance in the feature space without the need to explicitly specify the nonlinear mapping $\boldsymbol{\phi} : \boldsymbol{x} \in \mathbb{R}^{m \times 1} \rightarrow \boldsymbol{\phi}(\boldsymbol{x}) \in \mathcal{F}$. In fact,

$$\left\|\boldsymbol{\phi}(\boldsymbol{x}_i) - \boldsymbol{\phi}\left(\boldsymbol{x}_j\right)\right\|^2 = \left(\boldsymbol{x}_i - \boldsymbol{x}_j\right)^{\mathrm{T}}\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right)$$
$$= k(\boldsymbol{x}_i, \boldsymbol{x}_i) - k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) - k\left(\boldsymbol{x}_j, \boldsymbol{x}_i\right) + k\left(\boldsymbol{x}_j, \boldsymbol{x}_j\right)$$

which is only related to the kernel function $k(\bullet, \bullet)$. With the neighbourhood information from the adjacency graph, a simple way to select weighting parameters is given by

$$w_{i,j} = \begin{cases} 1, & \text{if } \boldsymbol{\phi}(\boldsymbol{x}_i) \text{ and } \boldsymbol{\phi}\left(\boldsymbol{x}_j\right) \text{ are connected,} \\ 0, & \text{if } \boldsymbol{\phi}(\boldsymbol{x}_i) \text{ and } \boldsymbol{\phi}\left(\boldsymbol{x}_j\right) \text{ are not connected.} \end{cases} \tag{11}$$

An alternative way of calculating $w_{i,j}$ is based on the heat kernel [27].

By defining $d_i = \sum_{j=1}^{n} w_{i,j}$, the optimisation (10) can be re-arranged as

$$minJ_{\mathrm{LSA}}(\boldsymbol{f}) = min\left\{\sum_{i=1}^{n}\boldsymbol{f}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_i)d_i(\boldsymbol{\phi}(\boldsymbol{x}_i))^{\mathrm{T}}\boldsymbol{f} - \sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{f}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_i)w_{i,j}\left(\boldsymbol{\phi}\left(\boldsymbol{x}_j\right)\right)^{\mathrm{T}}\boldsymbol{f}\right\}. \tag{12}$$

Further denote the nonlinear mapping matrix of $\boldsymbol{X}_n$ as $\Phi(\boldsymbol{X}_n) = [\boldsymbol{\phi}(\boldsymbol{x}_1)\boldsymbol{\phi}(\boldsymbol{x}_2) \cdots \boldsymbol{\phi}(\boldsymbol{x}_n)]^{\mathrm{T}}$, the diagonal matrix $\boldsymbol{D} = \mathrm{diag}\{d_1, d_2, \cdots d_n\}$, and the weighting matrix $\boldsymbol{W}$ whose (i, j)-th element is $[\boldsymbol{W}]_{i,j} = w_{i,j}$. The optimisation (12) can be expressed as

$$minJ_{\mathrm{LSA}}(\boldsymbol{f}) = min\,\boldsymbol{f}^{\mathrm{T}}(\Phi(\boldsymbol{X}_n))^{\mathrm{T}}(\boldsymbol{D} - \boldsymbol{W})\Phi(\boldsymbol{X}_n)\boldsymbol{f}$$
$$= min\,\boldsymbol{f}^{\mathrm{T}}(\Phi(\boldsymbol{X}_n))^{\mathrm{T}}\boldsymbol{L}\Phi(\boldsymbol{X}_n)\boldsymbol{f}, \tag{13}$$

where the matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ is also called the Laplacian matrix. As in the KPCA, the projection vector $\boldsymbol{f}$ takes the form (3) based on the training samples. Substituting $\boldsymbol{f} = \sum_{j=1}^{n} \alpha_j\boldsymbol{\phi}\left(\boldsymbol{x}_j\right) = (\Phi(\boldsymbol{X}_n))^{\mathrm{T}}\boldsymbol{\alpha}$ into the optimisation objective in Eq. (13) for local structure analysis leads to

$$minJ_{\mathrm{LSA}}(\boldsymbol{\alpha}) = min\left\{\boldsymbol{\alpha}^{\mathrm{T}}\Phi(\boldsymbol{X}_n)(\Phi(\boldsymbol{X}_n))^{\mathrm{T}}\boldsymbol{L}\Phi(\boldsymbol{X}_n)(\Phi(\boldsymbol{X}_n))^{\mathrm{T}}\boldsymbol{\alpha}\right\}. \tag{14}$$

Noting that the kernel matrix $\boldsymbol{K} = \Phi(\boldsymbol{X}_n)(\Phi(\boldsymbol{X}_n))^{\mathrm{T}}$, the above optimisation becomes

$$min\,J_{\mathrm{LSA}}(\boldsymbol{\alpha}) = min\,\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{\alpha}. \tag{15}$$

3.2. Modified KPCA based on local structure analysis

It can be seen that the standard KPCA optimisation (6) aims at the global data structure information as it maximally explains the training data variance, while the optimisation (15) only aims at the local structure information as it preserves local neighbourhood relationships. For data analysis and process monitoring application, both local and global data information are useful, because the global structure defines the outer shape of the process dataset while the local structure provides its inner organisation [37]. Our proposed LKPCA approach incorporates local structure analysis naturally into the KPCA. In order to extract the maximal training data variance as well as to preserve the local data structure between neighbouring data points, the optimisation objective of the LKPCA is to maximise $\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{K}\boldsymbol{\alpha}$ (for global variance extraction) and

to minimise $\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{KLK\alpha}$ (for local structure preserving), simultaneously. In other words, the goal of the proposed LKPCA is to solve the following optimisation problem

$$\max J_{\mathrm{LKPCA}}(\boldsymbol{\alpha}) = \max \frac{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{KK\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{KLK\alpha}}, \tag{16}$$
$$\text{s.t. } \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K\alpha} = 1.$$

The optimal coefficient vector $\boldsymbol{\alpha}$ can be obtained by solving the generalised eigenvector problem

$$\boldsymbol{KK\alpha} = \lambda\boldsymbol{KLK\alpha}. \tag{17}$$

In order to ensure a nonsingular problem, the regularisation method [34] can be used by substituting $\boldsymbol{KLK}$ with $\boldsymbol{KLK} + \delta\boldsymbol{I}_n$ in Eq. (17), where δ is a small positive regularisation parameter and $\mathbf{I}_n$ denotes the $n \times n$ identity matrix. A set of the $\tilde{n}$ normalised eigenvectors $\boldsymbol{\alpha}^{1)}, \boldsymbol{\alpha}^{2)}, \cdots, \boldsymbol{\alpha}^{n)}$ related to the $\tilde{n}$ nonzero eigenvalues can be obtained by solving (17), which meet $(\boldsymbol{\alpha}^{i)})^{\mathrm{T}}\boldsymbol{K\alpha}^{i)} = 1$, $1 \leq i \leq \tilde{n}$. Again, to reduce the dimensionality of the problem, we may only retain the first $\tilde{n}_s$ eigenvectors corresponding to the $\tilde{n}_s$ most significant eigenvalues. The test scores of a test vector $\boldsymbol{\phi}(\boldsymbol{x})$ are then extracted by projecting $\boldsymbol{\phi}(\boldsymbol{x})$ onto the eigenvectors $\boldsymbol{f}^{i)} = \sum_{j=1}^{n}\alpha_j^{i)}\boldsymbol{\phi}(\boldsymbol{x}_j)$, yielding for $1 \leq i \leq \tilde{n}$ or $1 \leq i \leq \tilde{n}_s$

$$\bar{t}_i = \sum_{j=1}^{n} \alpha_j^{i)} k(\boldsymbol{x}, \boldsymbol{x}_j) = \boldsymbol{k}_{\boldsymbol{x}}^{\mathrm{T}}\boldsymbol{\alpha}^{i)}. \tag{18}$$

4. Process monitoring strategy based on the LKPCA

Like any process monitoring scheme, the process monitoring based on the LKPCA includes two parts: fault detection and fault identification. Fault detection is to detect if any fault occurs using the LKPCA based monitoring statistics. When a fault is detected, fault identification is activated to identify which process variable is related to the fault.

4.1. Fault detection based on the LKPCA

The standard KPCA-based monitoring often uses two monitoring statistics, known as the T^2 and Q statistics [9,10]. Similarly, these two monitoring statistics can be built for the LKPCA-based fault detection and process monitoring. Specifically, the T^2 statistic is built to measure the dominating data variation in the principal component space, and is defined as

$$T^2 = \boldsymbol{t}_{\tilde{n}_s}^{\mathrm{T}} \boldsymbol{S}^{-1} \boldsymbol{t}_{\tilde{n}_s}, \tag{19}$$

where the dominating test score vector $\boldsymbol{t}_{\tilde{n}_s} = \left[\bar{t}_1 \bar{t}_2 \cdots \bar{t}_{\tilde{n}_s}\right]^{\mathrm{T}}$, corresponding to the first $\tilde{n}_s$ significant eigenvalues, is calculated as in Eq. (18), while $\mathbf{S}$ is the covariance matrix of the first $\tilde{n}_s$ training score vectors calculated on the training data set. By contrast, the Q statistic is used to measure the data variability in the residual space, and is defined as

$$Q = \sum_{i=1}^{\tilde{n}} (\bar{t}_i)^2 - \sum_{i=1}^{\tilde{n}_s} (\bar{t}_i)^2 = \boldsymbol{t}_{\tilde{n}}^{\mathrm{T}}\boldsymbol{t}_{\tilde{n}} - \boldsymbol{t}_{\tilde{n}_s}^{\mathrm{T}}\boldsymbol{t}_{\tilde{n}_s}, \tag{20}$$

where the test score vector $\bar{\boldsymbol{t}}_{\tilde{n}} = \left[\bar{t}_1\bar{t}_2 \cdots \bar{t}_{\tilde{n}}\right]^{\mathrm{T}}$, corresponding to all the $\tilde{n}$ nonzero eigenvalues, is also obtained using Eq. (18) and represents all the test scores.

After the monitoring statistics are obtained, the confidence limit is calculated to determine whether the process is in normal operating regions. Typical confidence limits for the KPCA-based monitoring statistics are obtained using the F and weighted χ^2 distributions [10]. However, these two distributions may not be suitable for the LKPCA-based scheme, because of the introduction of local structure analysis. We

propose to use the well-known kernel density estimation method [17,38–40] to estimate the distribution of a LKPCA-based monitoring statistic variable, which will enable us to calculate the confidence limits of the LKPCA-based monitoring statistics. Assume that the monitoring statistic variable y is governed by an unknown probability density function (PDF). Given the sample data set $\{y_l\}_{i=l}^{L}$ of the variable y, the kernel estimator for its PDF is given by [39,40]

$$\hat{p}(y) = \frac{1}{hL}\sum_{l=1}^{L} \varphi\left(\frac{y-y_l}{h}\right), \tag{21}$$

where h is known as the kernel width, and $\varphi(\bullet)$ denotes the chosen kernel function. Typically the Gaussian kernel function $\varphi(u) = e^{-u^2/2}/\sqrt{2\pi}$ is used. The value $\tilde{y}$, which is defined by

$$\int_{-\infty}^{\tilde{y}} \hat{p}(u)du = 0.95, \tag{22}$$

provides the 95% confidence limit.

The fault detection procedure includes two stages: off-line modelling and on-line detection. In the off-line modelling stage, the normal operating model is developed by the LKPCA and the confidence limits of the monitoring statistics are determined. During the on-line detection stage, new observed data is collected and the on-line monitoring statistics are calculated to determine whether the process is under the normal operation condition. The detailed LKPCA-based fault detection procedure can now be summarised.

The off-line modelling:

1) Acquire the normal operating data matrix $\boldsymbol{X}_L = [\boldsymbol{x}_1 \ \boldsymbol{x}_2 \cdots \boldsymbol{x}_L]^{\mathrm{T}} \in \mathbb{R}^{L \times m}$, where $L \geq n$ and $\boldsymbol{x}_i = [x_{1,i} \ x_{2,i} \cdots x_{m,i}]^{\mathrm{T}}$. Use a subset $\boldsymbol{X}_n \subseteq \boldsymbol{X}_L$, where $\boldsymbol{X}_n = [\boldsymbol{x}_1 \boldsymbol{x}_2 \cdots \boldsymbol{x}_n]^{\mathrm{T}} \in \mathbb{R}^{n \times m}$, to compute the sample mean and variance for each variable

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{j,i}, 1 \leq j \leq m, \tag{23}$$

$$\sigma_j^2 = \frac{1}{n-1}\sum_{i=1}^{n} \left(x_{j,i} - \bar{x}_j\right)^2, 1 \leq j \leq m. \tag{24}$$

Then, normalise the training data $\boldsymbol{x}_i$ for $1 \leq i \leq L$ using the means $\bar{\boldsymbol{x}} = [\bar{x}_1 \bar{x}_2 \cdots \bar{x}_m]^{\mathrm{T}}$ and the variances $\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_m^2\}$ according to

$$\boldsymbol{x}_i^{\mathrm{c-s}} = \mathrm{diag}\left\{\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \cdots, \frac{1}{\sigma_m}\right\}(\boldsymbol{x}_i - \bar{\boldsymbol{x}}). \tag{25}$$

2) Compute the $n \times n$ kernel matrix $\boldsymbol{K}$ based on $\boldsymbol{X}_n^{\mathrm{c-s}} = [\boldsymbol{x}_1^{\mathrm{c-s}} \boldsymbol{x}_2^{\mathrm{c-s}} \cdots \boldsymbol{x}_n^{\mathrm{c-s}}]^{\mathrm{T}}$, and then doubly centre as well as scale $\boldsymbol{K}$ according to[1]

$$\boldsymbol{K}^{\mathrm{c}} = \boldsymbol{K} - \boldsymbol{KZ}_n - \boldsymbol{Z}_n\boldsymbol{K} + \boldsymbol{Z}_n\boldsymbol{KZ}_n, \tag{26}$$

$$\tilde{\boldsymbol{K}}^{\mathrm{c-s}} = \frac{\tilde{\boldsymbol{K}}^{\mathrm{c}}}{\mathrm{trace}\left(\tilde{\boldsymbol{K}}^{\mathrm{c}}\right)/(n-1)}, \tag{27}$$

where $\boldsymbol{Z}_n$ is the $n \times n$ matrix whose elements are all equal to $1/n$.
3) Solve the generalised eigenvalue problem by replacing $\boldsymbol{K}$ with $\tilde{\boldsymbol{K}}^{\mathrm{c-s}}$ in Eq. (16) to obtain the $\tilde{n}$ nonzero eigenvalues and the corresponding eigenvectors.
4) Calculate the monitoring statistics, $T^2(l)$ and $Q(l)$, for each normal operating data $\boldsymbol{x}_l^{\mathrm{c-s}}$, $1 \leq l \leq L$, according to Eqs. (18) to (20), and establish the control limits for the monitoring statistics, T^2 and Q, using the kernel density estimators (21) and (22) based on the sample data $\{T^2(l)\}_{l=1}^{L}$ and $\{Q(l)\}_{l=1}^{L}$.

---

[1] The kernel matrix has to be centred in both row and column to conserve its symmetry as well as scaled. This is to ensure that the implicitly mapped data $\boldsymbol{\phi}(\boldsymbol{x}_i)$ in the feature space is mean centred and variance scaled [10].

The on-line detection:

1) Obtain the new data $\mathbf{x}_{\text{new}}$, and normalise it with the n-sample training means and variances of the m variables, as in Eq. (25) of the off-line modelling, to obtain $\mathbf{x}_{\text{new}}^{c-s}$.

2) Compute the test kernel vector $\mathbf{k}_{\mathbf{x}_{\text{new}}} = [k(\mathbf{x}_{\text{new}}^{c-s}, \mathbf{x}_1^{c-s}) k(\mathbf{x}_{\text{new}}^{c-s}, \mathbf{x}_2^{c-s}) \cdots k(\mathbf{x}_{\text{new}}^{c-s}, \mathbf{x}_n^{c-s})]^{\mathrm{T}}$, and apply the double centering as well as scaling to it

$$\widetilde{\mathbf{k}}_{\mathbf{x}_{\text{new}}}^{c} = \mathbf{k}_{\mathbf{x}_{\text{new}}} - \mathbf{Z}_n \mathbf{k}_{\mathbf{x}_{\text{new}}} - \mathbf{K}\mathbf{z}_n + \mathbf{Z}_n \mathbf{K}\mathbf{z}_n, \tag{28}$$

$$\frac{\widetilde{\mathbf{k}}_{\mathbf{x}_{\text{new}}}^{c-s} = \widetilde{\mathbf{k}}_{\mathbf{x}_{\text{new}}}^{c}}{\text{trace}\left(\widetilde{\mathbf{K}}^c\right)/(n-1)}, \tag{29}$$

where $\mathbf{z}_n$ is the $n \times 1$ vector whose elements are all equal to $1/n$, while $\mathbf{Z}_n$, $\mathbf{K}$ and $\widetilde{\mathbf{K}}^c$ are the matrices given in Eq. (26) of the off-line modelling.

3) Compute the new scores according to (18) by replacing $\mathbf{k}_{\mathbf{x}}$ with $\widetilde{\mathbf{k}}_{\mathbf{x}_{\text{new}}}^{c-s}$, and obtain the two monitoring statistics, T_{new}^2 and Q_{new}, according to Eqs. (19) and (20), respectively.

4) The two statistics of the new observed data, T_{new}^2 and Q_{new}, are compared with the corresponding confidence limits. If the upper control limits are exceeded, an abnormal behaviour of the process is detected.

4.2. Fault variable identification using the LKPCA-based contribution plot

As in any fault detection scheme, the LKPCA-based monitoring chart only indicates the deviation of the process from normal operating conditions but does not provide information on which process variable is the root cause of the fault. Once a fault is detected by the monitoring statistics, it is the task of fault identification to identify fault variables and to locate root causes, which is a challenging task in fault diagnosis. In the previous works [41,42], contribution plot, which identifies the potential fault variables by calculating their contributions to monitoring statistics, has displayed certain effectiveness to identify fault variable for the linear PCA method. In the contribution plot, the variable with the greatest contribution value to the monitoring statistic usually is the fault source. However, the traditional linear contribution plot cannot be directly generalised to a kernel method because of the use of implicit nonlinear transformation [9]. Although some methods have been developed based on reconstruction error calculation [12] and virtual scale factor analysis [11], fault variable identification and source diagnosis is still an unsolved open problem for nonlinear data-driven fault diagnosis.

In order to identify fault variables for the LKPCA-based diagnosis, we present a LKPCA contribution plot method which is easy to implement. The proposed method is inspired by the sensitivity analysis [43], which calculates the rates of change in the system output variables, resulting from small perturbations in the problem parameters. For the observed vector $\mathbf{x} = [x_1 \ x_2, \dots x_m]^{\mathrm{T}}$ with the m variables, we design the contributions of the i-th variable x_i to the two monitoring statistics, T^2 and Q, as

$$CT_i^2 = x_i \frac{\partial T^2}{\partial x_i}, \tag{30}$$

$$CQ_i = x_i \frac{\partial Q}{\partial x_i}. \tag{31}$$

Combining Eq. (18) with Eqs. (19) and (20), respectively, we can express T^2 and Q as

$$T^2 = \mathbf{k}_{\mathbf{x}}^{\mathrm{T}} \mathbf{A}_{n_s} \mathbf{S}^{-1} \mathbf{A}_{n_s}^{\mathrm{T}} \mathbf{k}_{\mathbf{x}}, \tag{32}$$

$$Q = \mathbf{k}_{\mathbf{x}}^{\mathrm{T}} \left(\mathbf{A}_n \mathbf{A}_n^{\mathrm{T}} - \mathbf{A}_{n_s} \mathbf{A}_{n_s}^{\mathrm{T}}\right) \mathbf{k}_{\mathbf{x}}, \tag{33}$$

where $\mathbf{A}_q = [\boldsymbol{\alpha}^{(1)} \boldsymbol{\alpha}^{(2)} \cdots \boldsymbol{\alpha}^{(q)}]$, and $\mathbf{k}_{\mathbf{x}}$ has been properly centred and scaled according to Eqs. (28) and (29). Thus, the contributions of x_i to T^2 and Q are given respectively by

$$CT_i^2 = 2x_i \frac{\partial \mathbf{k}_{\mathbf{x}}^{\mathrm{T}}}{\partial x_i} \mathbf{A}_{n_s} \mathbf{S}^{-1} \mathbf{A}_{n_s}^{\mathrm{T}} \mathbf{k}_{\mathbf{x}}, \tag{34}$$

$$CQ_i = 2x_i \frac{\partial \mathbf{k}_{\mathbf{x}}^{\mathrm{T}}}{\partial x_i} \left(\mathbf{A}_n \mathbf{A}_n^{\mathrm{T}} - \mathbf{A}_{n_s} \mathbf{A}_{n_s}^{\mathrm{T}}\right) \mathbf{k}_{\mathbf{x}}. \tag{35}$$

We now discuss a few important issues related to the use of contribution plot. Firstly, we adopt the relative contribution values of each variable with respect to some normal operating condition. More specifically, let $CT_{i,\text{normal}}^2$ and $CQ_{i,\text{normal}}$ denote the contribution values of the i-th process variable under the normal operation condition considered. Further define the mean and standard deviation operations, mean($\bullet$) and std($\bullet$), as performing over the whole training data set $\mathbf{X}_L$. Then, the relative contributions, whose expressions are given by

$$RCT_i^2 = \frac{CT_i^2 - \text{mean}\left(CT_{i,\text{normal}}^2\right)}{\text{std}\left(CT_{i,\text{normal}}^2\right)}, \tag{36}$$

$$RCQ_i = \frac{CQ_i - \text{mean}\left(CQ_{i,\text{normal}}\right)}{\text{std}\left(CQ_{i,\text{normal}}\right)}, \tag{37}$$

provide more accurate fault isolation than the original contribution plot. Secondly, the contributions of a process variable, as computed in (36) and (37), can be positive or negative. The sign of a contribution value is not important, and it is the absolute value that reflects the influence of the process variable. Therefore, we use $|RCT_i^2|$ and $|RCQ_i|$ in contribution graph. Thirdly, the contributions, $|RCT_i^2|$ and $|RCQ_i|$, calculated using one sample $\mathbf{x}$ are stochastic, which are influenced by many uncertainties, such as the measurement noise. Therefore, in practical implementation, we always use $\overline{|RCT_i^2|}$ and $\overline{|RCQ_i|}$, which are the average values of $|RCT_i^2|$ and $|RCQ_i|$ over several fault data samples, in the contribution plot.

5. Simulation study

The proposed process monitoring and fault diagnosis strategy based on the LKPCA was tested on the well-known Tennessee Eastman (TE) process [41], and its achievable performance was compared with that of the benchmark scheme based on the standard KPCA.

5.1. Process description

The TE process was first introduced by Downs and Vogel [44] which has since been widely used as a benchmark process for comparison of various process monitoring strategies [18,22,37,44–46]. This process is from a realistic, standard model of an industrial plant-wide chemical operation and consists of five major units: a reactor, a condenser, a compressor, a separator, and a stripper. A flowchart of the TE process is illustrated in Fig. 1. The process contains 52 monitoring variables which include 11 manipulated variables (MVs), 22 continuous process measurements, and 19 composition measurements. All the 52 variables are listed in Table 1. A TE simulator coded by FORTRAN is provided in http://brahms.scs.uiuc.edu and allows 21 pre-programmed major process faults. All these 21 faults are listed in Table 2. The TE simulator produces 960 observations for normal and each fault operation modes with a sampling interval of 3 min. The data sets for the 21 fault operations as well as the normal operation can be downloaded from the above website. All faults are introduced into the process at the 160-th sample. More detailed description of the TE process can be found in [41].
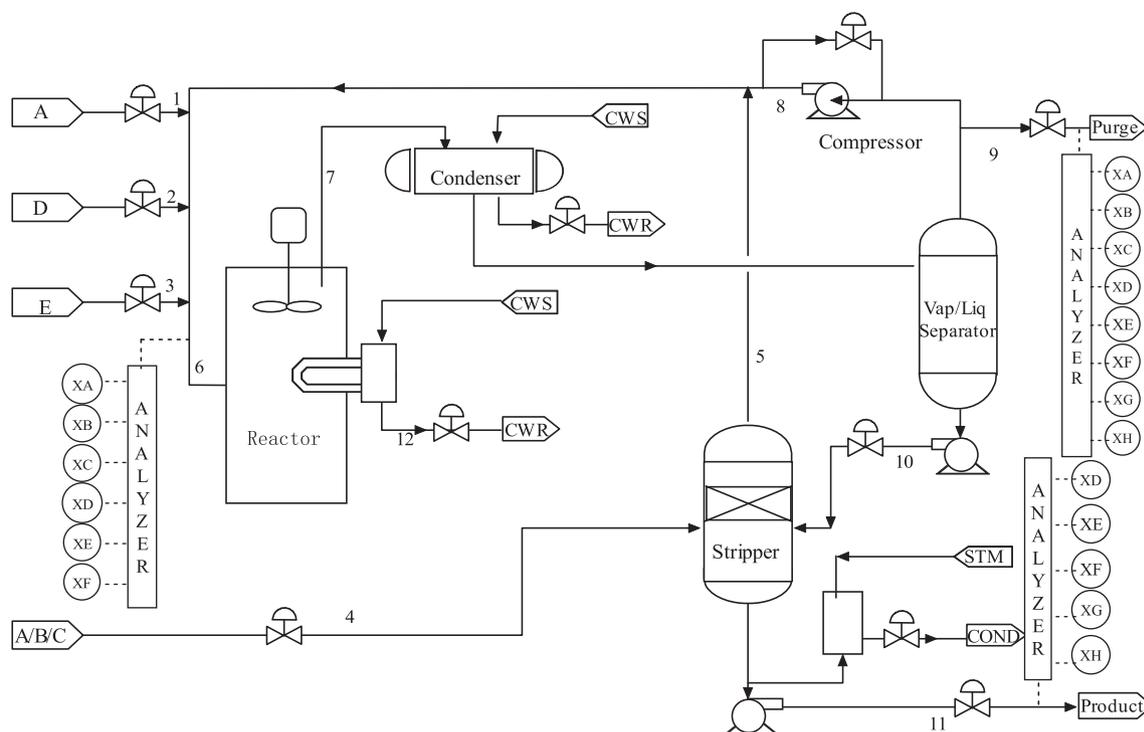
Fig. 1. Flowchart of Tennessee Eastman process.

5.2. Results and discussion

The first 300 samples ($n = 300$) in the process normal-operation data set were used for model training, and all the 960 normal-operation data samples ($L = 960$) were used to test the model performance under the normal operation as well as to build the confidence limits for monitoring statistics. For the purpose of convenient comparison, all monitoring statistics were divided by their respective confidence limits so that alarm limits in all plots were equal to 1. Two criteria were used to compare fault detection performance, which are

Table 1
List of the monitoring variables in Tennessee Eastman process.

No.	Process variable
1	A feed (stream 1)
2	D feed (stream 2)
3	E feed (stream 3)
4	Total feed (stream 4)
5	Recycle flow (stream 8)
6	Reactor feed rate (Stream 6)
7	Reactor pressure
8	Reactor level
9	Reactor temperature
10	Purge rate (stream 9)
11	Product separator temperature
12	Product separator level
13	Product separator pressure
14	Product separator underflow (stream 10)
15	Stripper level
16	Stripper pressure
17	Stripper underflow (stream 11)
18	Stripper temperature
19	Stripper steam flow
20	Compressor work
21	Reactor cooling water outlet temperature
22	Separator cooling water outlet temperature
23–28	Components A, B, C, D, E and F in stream 6
29–36	Components A, B, C, D, E, F, G and H in stream 9
37–41	Components D, E, F, G and H in stream 11
42	MV for D feed flow (stream 2)
43	MV for E feed flow (stream 3)
44	MV for A feed flow (stream 1)
45	MV for total feed flow (stream 4)
46	MV for compressor recycle valve
47	MV for purge valve (stream 9)
48	MV for separator pot liquid flow (stream 10)
49	MV for stripper liquid prod flow (stream 11)
50	MV for stripper steam valve
51	MV for reactor cooling water flow
52	MV for condenser cooling water flow

Table 2
Tennessee Eastman process fault patterns.

Fault ID	Process variable	Fault type
IDV(1)	A/C feed ratio, B composition constant (stream 4)	Step
IDV(2)	B composition, A/C feed ratio constant (stream 4)	Step
IDV(3)	D feed temperature (stream 2)	Step
IDV(4)	Reactor cooling water inlet temperature	Step
IDV(5)	Condenser cooling water inlet temperature	Step
IDV(6)	A feed loss (stream 1)	Step
IDV(7)	C header pressure loss-reduced availability (stream 4)	Step
IDV(8)	A, B and C feed compositions (stream 4)	Random variation
IDV(9)	D feed temperature (stream 2)	Random variation
IDV(10)	C feed temperature (stream 4)	Random variation
IDV(11)	Reactor cooling water inlet temperature	Random variation
IDV(12)	Condenser cooling water inlet temperature	Random variation
IDV(13)	Reaction kinetics	Slow shift
IDV(14)	Reactor cooling water valve	Sticking
IDV(15)	Condenser cooling water valve	Sticking
IDV(16)–(20)	Unknown	Unknown
IDV(21)	Valve position constant (stream 4)	Constant position
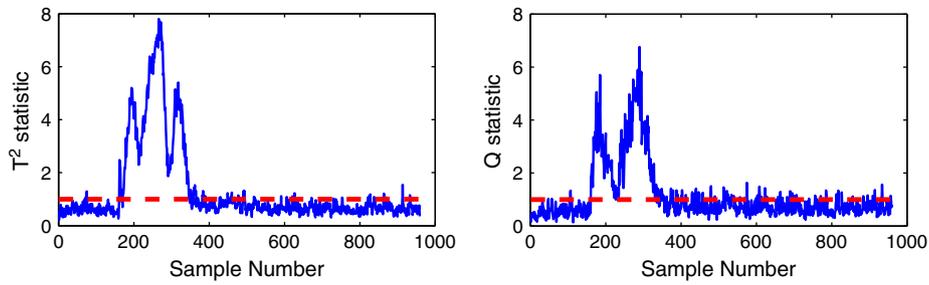
Fig. 2. On-line monitoring charts of the KPCA under Fault IDV(5).

fault detection time and fault alarming rate. Fault detection time is defined as the first sample number after previous eight consecutive samples have exceeded the confidence limit, while fault alarming rate is defined as the percentage of the alarming samples in all the fault samples.

When applying the KPCA and LKPCA models to fault diagnosis, the dimension of the feature space should be determined carefully. Theoretically, the number of nonzero eigenvalues $\tilde{n}$ is equal to the dimension of the feature space. However, if the number of model training samples n is very large, the number of nonzero eigenvalues may also be very large. According to [12], the dimension of the feature space can be determined as the smallest number of the ordered eigenvalues whose cumulative sum is above 99.99% of the sum of all the eigenvalues. In the simulation, we set $\tilde{n}$ to this smallest number. The Gaussian kernel function

$$k(\boldsymbol{x}, \boldsymbol{y}) = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|^2}{2\sigma_k^2}}, \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^{m \times 1}, \tag{38}$$

where σ_k is the kernel width parameter, is a popular choice of kernel function, and was used in this study. The kernel width σ_k can be determined via cross validation. Alternatively, σ_k is computed according to the number of the process variables m in some studies of KPCA [11,12]. In the simulation, we set $\sigma_k = 20\,m$ for both the KPCA and LKPCA models. Thirdly, it is important to select the number of the dominant principal components $\tilde{n}_s$ properly. Existing methods of selecting $\tilde{n}_s$ include cumulative percent eigenvalue, SCREE test, parallel analysis and reconstruction error criterion [41], but there is no consensus as which technique is best. In the simulation, we simply set $\tilde{n}_s$ to the number of the dominant principal components whose cumulative eigenvalue sum counted to above 90% of the sum of all the eigenvalues. With this technique, the values of $\tilde{n}_s$ for the KPCA and LKPCA were found to be 30 and 42, respectively, using the normal operation data set. Lastly, the neighbourhood relationship parameter influences the performance of the LKPCA, and should be chosen appropriately. In our simulation study, the KNN method was used to construct the adjacency graph and the value of K was set to 5. The influence of different K values on the detection performance of the LKPCA model is further discussed in Subsection 5.3.
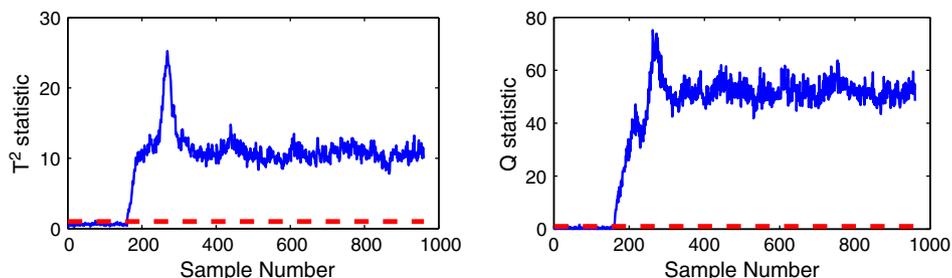
5.2.1. Fault detection performance

The monitoring results for three fault cases, IDV(5), IDV(10) and IDV(19), are first used to demonstrate the effectiveness of the proposed LKPCA-based strategy. Fault IDV(5) is a step change in the condenser cooling water inlet temperature. The monitoring results obtained using the KPCA method under Fault IDV(5) are shown in Fig. 2, where the T^2 and Q monitoring statistics are plotted as solid curves, while the confidence limits are represented by dashed lines. The Q statistic of the KPCA model detects Fault IDV(5) at the 161-th sample and its T^2 statistic detects the fault at the 171-th sample. However, these two statistics return to below their confidence limits after around the 348-th sample. This is because there exists a feedback control loop for the condenser cooling water outlet temperature, and the feedback control attempts to compensate the step disturbance in the condenser cooling water inlet temperature. The KPCA-based monitoring statistics shown in Fig. 2, therefore, give a mistaken information that the fault has disappeared after the 348-th sample, while in fact, this fault does not disappear but is only hidden. With the application of the LKPCA, the resulting monitoring charts are plotted in Fig. 3, where it can be seen that both the LKPCA-based T^2 and Q statistics exceed their thresholds from the 161-th sample. Moreover, the LKPCA-based monitoring statistics correctly indicate the existence of a fault throughout the whole fault operation period, and this clearly demonstrates that the LKPCA-based method is more effective than the KPCA for detecting Fault IDV(5).

The detection results obtained by the KPCA and LKPCA models under Fault IDV(10) are plotted in Figs. 4 and 5, respectively. Fault IDV(10) involves the random variation of C feed temperature. From Fig. 4, we can see that Fault IDV(10) is detected by the KPCA-based T^2 and Q statistics at the 244-th sample and 208-th sample, respectively. By comparison, the LKPCA performs better and its two statistics detect the fault at the same 184-th sample. From the detection results under Fault IDV(10), it can be seen that the LKPCA is more sensitive than the KPCA in fault detection.

Due to the proprietary reason, Fault patterns IDV(16) to IDV(20) were not disclosed, and only the fault data sets were provided. When the KPCA was applied under Fault IDV(19), the results are shown in Fig. 6, where it can be seen that the T^2 and Q monitoring statistics vary around the confidence limits with the poor fault alarming rates of 15.88% and 24.75%, respectively. Clearly, the KPCA-based method

Fig. 3. On-line monitoring charts of the LKPCA under Fault IDV(5).
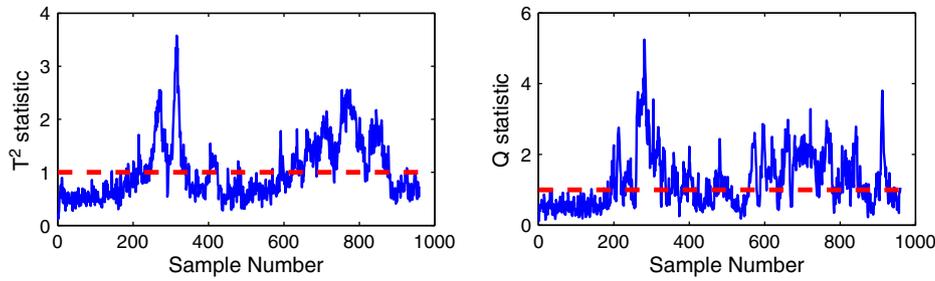
Fig. 4. On-line monitoring charts of the KPCA under Fault IDV(10).
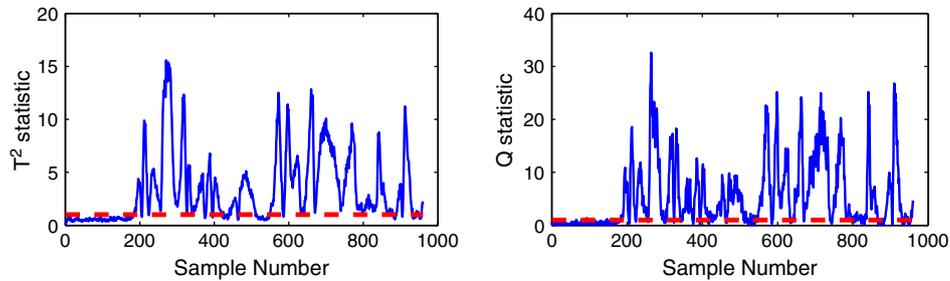
Fig. 5. On-line monitoring charts of the LKPCA under Fault IDV(10).
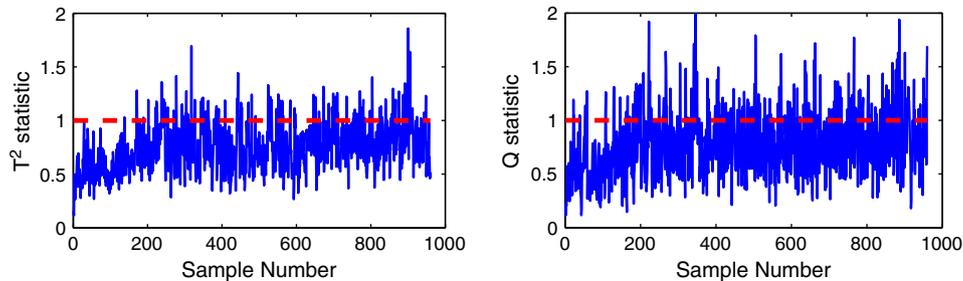
Fig. 6. On-line monitoring charts of the KPCA under Fault IDV(19).

cannot detect Fault IDV(19) confidently. In comparison, the monitoring charts obtained by the LKPCA method under Fault IDV(19), depicted in Fig. 7, show that the LKPCA-based T^2 and Q statistics clearly detect Fault IDV(19) at the 192-th sample and the 170-th sample, respectively, with the corresponding fault alarming rates of 68.75% and 72.63%. This again demonstrates the superiority of the LKPCA-based fault detection method over the standard KPCA-based one.

Next we summarise the fault detection performance for all the 21 fault cases in Tables 3 and 4, in terms of fault alarming rate and fault detection time, respectively. In terms of achievable fault alarming rates given in Table 3, we observe that both the KPCA-based and LKPCA-

based schemes provide similar good fault detection performance for Faults IDV(1), IDV(2), IDV(4), IDV(6), IDV(7), IDV(8), IDV(12), IDV(13), IDV(14), and IDV(18). However, the two methods both perform poorly for Faults IDV(3), IDV(9), and IDV(15). The previous works [15,41] have also found that these faults prove to be difficult for data-driven detection methods because there are no observable changes in the mean or the variance of these fault data sets. For Faults IDV(5), IDV(10), IDV(11), IDV(16), IDV(17), IDV(19), IDV(20), and IDV(21), the proposed LKPCA method improves the detection performance over the standard KPCA method considerably. For these eight fault patterns, the average fault alarming rates achieved by the KPCA
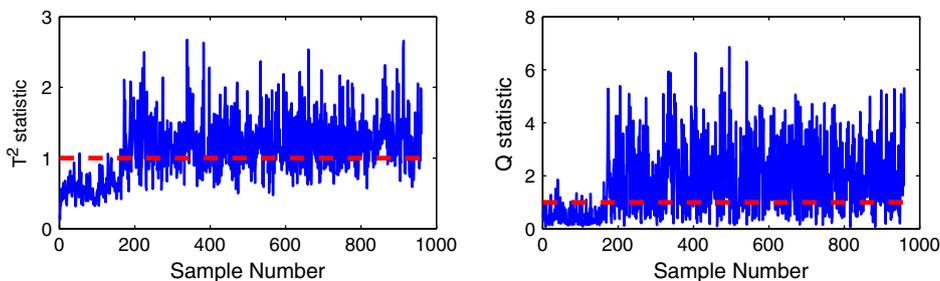
Fig. 7. On-line monitoring charts of the LKPCA under Fault IDV(19).

Table 3
Comparison of the fault alarming rates (%) achieved by the KPCA and LKPCA methods.

Fault ID	KPCA		LKPCA	
	T^2	Q	T^2	Q
IDV(1)	99.38	99.75	99.75	99.63
IDV(2)	98.75	98.50	98.75	98.75
IDV(3)	6.63	8.25	5.50	9.38
IDV(4)	64.38	100.00	99.13	89.88
IDV(5)	30.88	35.50	100.00	100.00
IDV(6)	99.25	100.00	100.00	100.00
IDV(7)	100.00	100.00	100.00	92.25
IDV(8)	97.50	98.00	98.00	96.63
IDV(9)	7.13	7.13	3.88	9.00
IDV(10)	49.50	61.25	85.00	82.75
IDV(11)	60.25	63.50	72.75	56.25
IDV(12)	98.75	97.13	99.63	99.75
IDV(13)	94.63	95.63	95.50	95.63
IDV(14)	99.88	99.88	100.00	99.88
IDV(15)	9.88	9.50	9.50	10.00
IDV(16)	31.63	59.75	88.00	80.75
IDV(17)	81.50	95.00	91.38	97.00
IDV(18)	89.38	90.75	90.00	90.63
IDV(19)	15.88	24.75	68.75	72.63
IDV(20)	42.25	56.75	73.63	80.88
IDV(21)	48.75	53.50	59.63	47.13

Table 4
Comparison of the fault detection time (sample number) achieved by the KPCA and LKPCA methods.

Fault ID	KPCA		LKPCA	
	T^2	Q	T^2	Q
IDV(1)	167	163	163	166
IDV(2)	171	173	171	175
IDV(3)	–	–	–	–
IDV(4)	272	161	161	163
IDV(5)	171	161	161	161
IDV(6)	167	161	161	161
IDV(7)	161	161	161	162
IDV(8)	183	177	177	182
IDV(9)	161	–	–	–
IDV(10)	244	208	184	184
IDV(11)	207	167	167	216
IDV(12)	167	163	163	163
IDV(13)	206	197	197	203
IDV(14)	161	162	161	162
IDV(15)	935	–	911	–
IDV(16)	356	175	167	169
IDV(17)	189	182	185	182
IDV(18)	249	244	244	244
IDV(19)	–	–	192	170
IDV(20)	247	245	231	227
IDV(21)	435	409	399	666

Note: – indicates that this fault cannot be detected with the monitoring statistic considered.

and LKPCA are compared in Fig. 8(a), while the average fault alarming rates of the two methods over all the 21 fault patterns are compared in Fig. 8(b). The results of Fig. 8 clearly demonstrate that the LKPCA based method on average produces higher fault alarming rate and can detect faults more effectively, in comparison with the standard KPCA method. From Table 4, it can be seen that the LKPCA based method has shorter fault detection times for Faults IDV(10), IDV(15), IDV(16), IDV(19), IDV(20), and IDV(21), than the KPCA based method. For the other fault patterns except for Fault IDV(9), both the LKPCA and KPCA schemes achieve similar detection performance. Thus, it can be concluded that the LKPCA based monitoring scheme performs better than the KPCA based one for the TE process.

5.2.2. Fault identification performance

After a fault is detected, it is important to identify the process variables that cause the fault and potentially to help removing the fault. It should be emphasised that nonlinear data-driven fault variable identification is an unsolved open problem. We investigated the potential of the new LKPCA-based contribution plot for fault identification using Faults IDV(4) and IDV(6) as two examples.

The LKPCA-based monitoring charts under Fault IDV(4), depicted in Fig. 9, indicated that the T^2 statistic detected this fault at the 161-th sample. The LKPCA-based T^2 and Q contribution plots are illustrated in Fig. 10 for the fault variable identification, where the contribution value was averaged over the 161-th to 165-th samples. From Fig. 10, it can be seen that the largest contribution to both the T^2 and Q contribution plots comes from the No.51 variable which corresponds to the reactor cooling water flow valve. With the help of this indication, operator can quickly inspect the trend of the reactor cooling water flow valve which is given in Fig. 11. From Fig. 11, we can see a clear change in the reactor cooling water flow valve which involves the control loop for the reactor temperature. Operator can further check the related variables, such as the reactor cooling water inlet temperature and reactor cooling water inlet flow, that may cause this change. In fact, Fault IDV(4) results from the reactor cooling water inlet temperature. Because the reactor cooling water inlet temperature is not included in the monitoring variable set, the contribution plots correctly indicate the related reactor cooling water flow valve as the fault variable. Therefore, the contribution plots of Fig. 10 provide a correct and effective diagnosis of Fault IDV(4).

Fault IDV(6) involves the component A feed loss in stream 1. This fault was detected quickly at the 161-th sample by the LKPCA-based T^2 and Q charts, as shown in Fig. 12. Average contribution values over the 161-th to 165-th samples are plotted in Fig. 13. Both the No.1 and No.44 variables show the highest contribution to this fault. From Table 1, we can see that the No.1 variable is the component A feed in stream 1 and the No.44 variable is the component A feed flow valve in stream 1. Obviously these two variables are connected closely and we can conclude that the fault variable is closely linked to the component
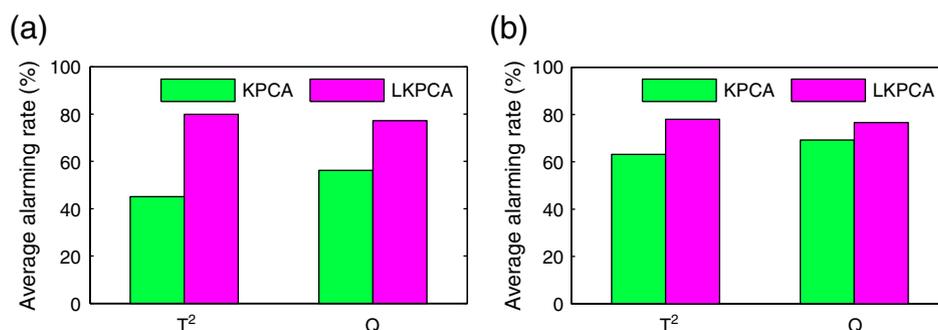
Fig. 8. Comparison of the average fault alarming rates achieved by the KPCA and LKPCA schemes: (a) average over Faults IDV(5), IDV(10), IDV(11), IDV(16), IDV(17), IDV(19), IDV(20) and IDV(21), and (b) average over all the 21 fault cases.
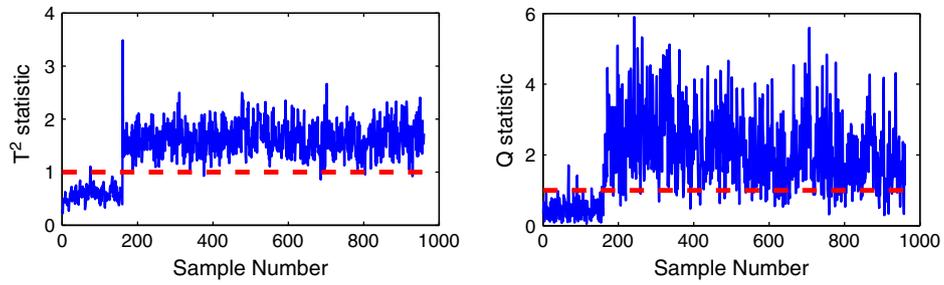
Fig. 9. On-line monitoring charts of the LKPCA under Fault IDV(4).
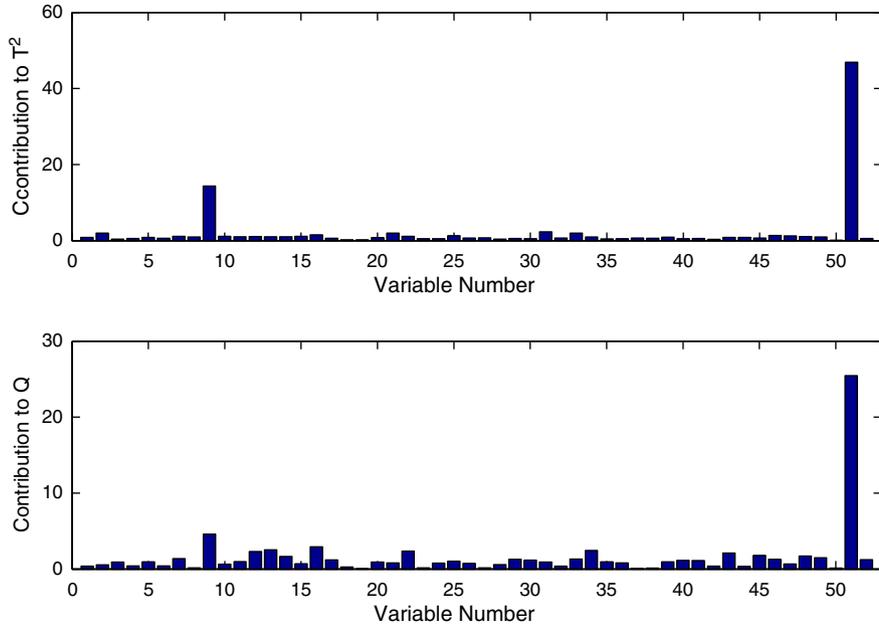
Fig. 10. Fault identification using the LKPCA-based T^2 and Q contribution plots under Fault IDV(4).
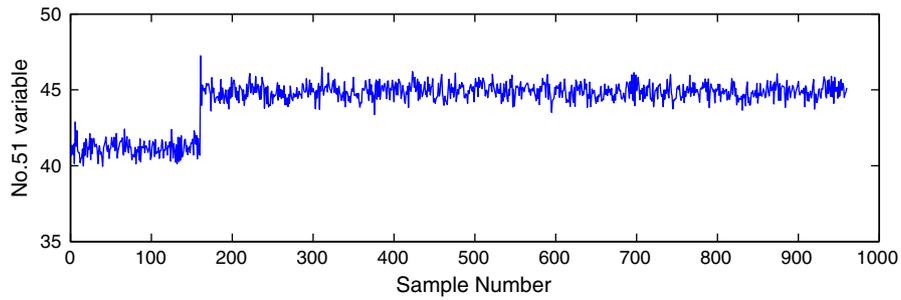
Fig. 11. The trend of No.51 variable under Fault IDV(4).
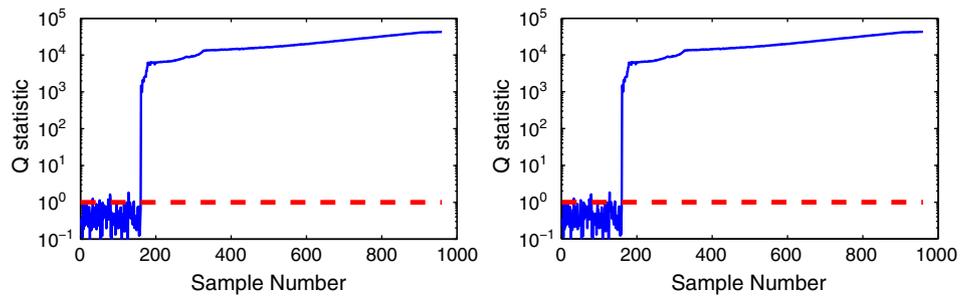
Fig. 12. On-line monitoring charts of the LKPCA under Fault IDV(6).
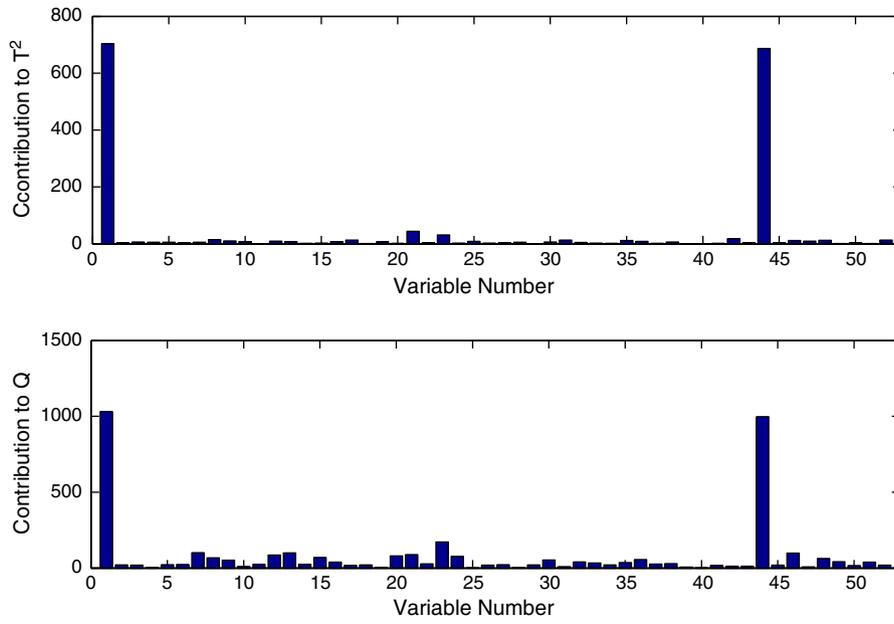
Fig. 13. Fault identification using the LKPCA-based T^2 and Q contribution plots under Fault IDV(6).
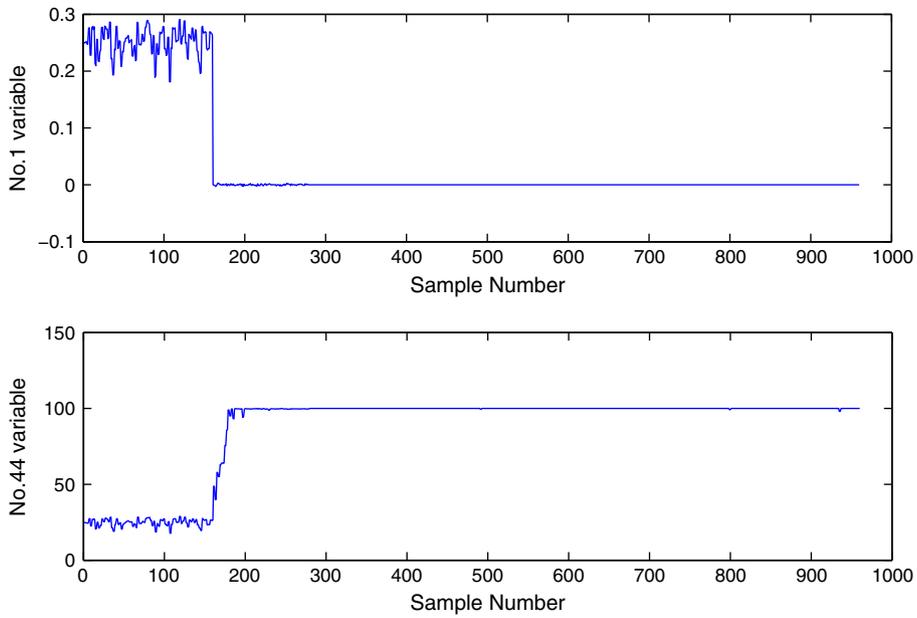
Fig. 14. The trends of No.1 and No.44 variables under Fault IDV(6).
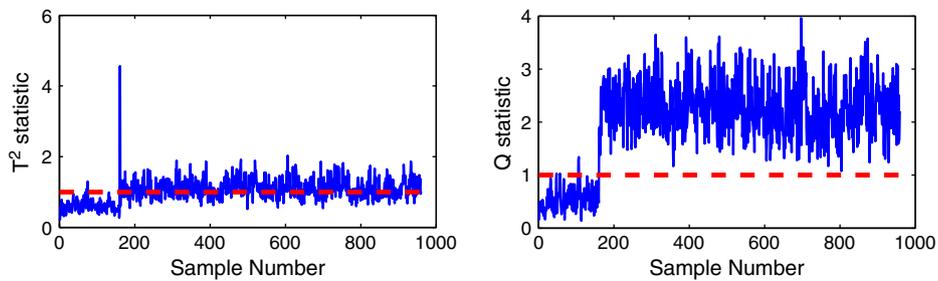
Fig. 15. On-line monitoring charts of the KPCA under Fault IDV(4).
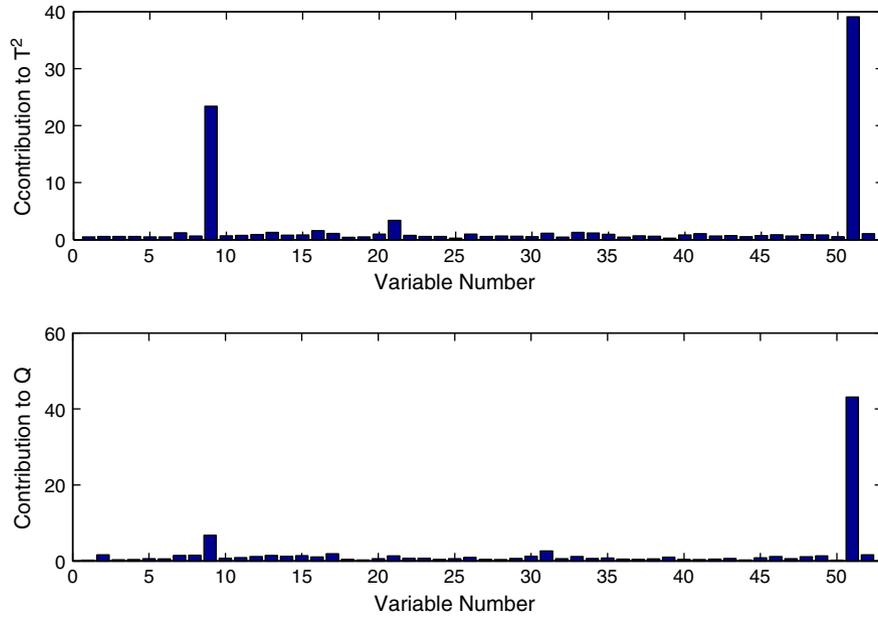
Fig. 16. Fault identification using the KPCA-based T^2 and Q contribution plots under Fault IDV(4).
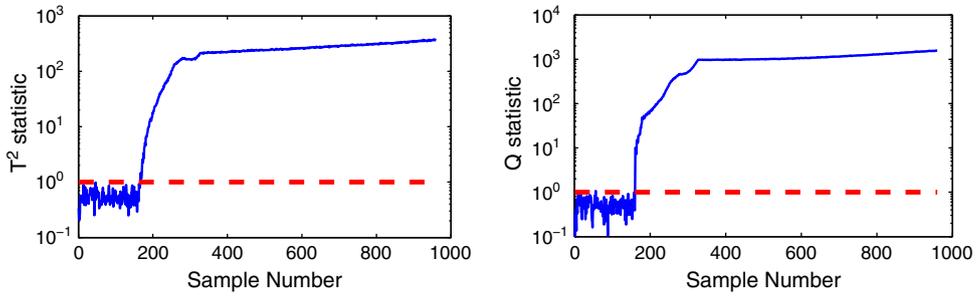
Fig. 17. On-line monitoring charts of the KPCA under Fault IDV(6).
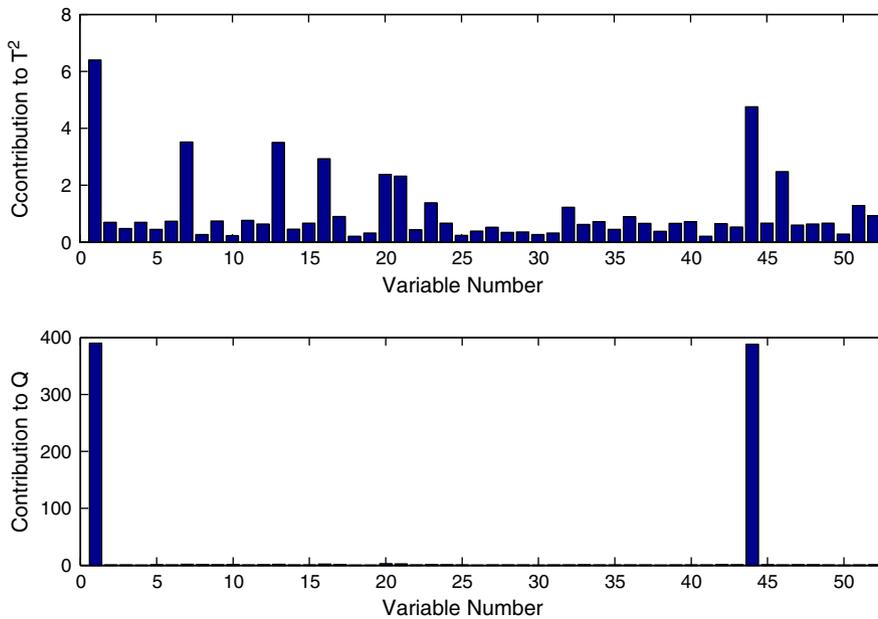
Fig. 18. Fault identification using the KPCA-based T^2 and Q contribution plots under Fault IDV(6).

A feed. The trends of both the No.1 and No.44 variables are plotted in Fig. 14. It is clear that the fault identification results correctly reflect the changes of these two variables, which is consistent with the real fault source.

As mentioned in the Introduction section, two fault identification methods were proposed in [11,12] for the KPCA based fault diagnosis. The idea presented in [12] is to use variable reconstruction error to identify fault variable, and this method needs the recursive iteration for the reconstructed value of each sample. The method proposed in [11] measures the contribution of each variable using a virtual scale factor method which involves the calculation of a matrix trace for each sample. These two methods are very complex and time-consuming. By contrast, our new contribution plot approach proposed in Subsection 4.2 is computationally very simple and extremely easy to implement. Moreover, our contribution plot method is very general, and is equally applicable to the KPCA-based scheme. To demonstrate this, we also applied our method to calculate the KPCA-based T^2 and Q contribution plots for fault identification of Faults IDV(4) and IDV(6).

The KPCA-based monitoring charts under Fault IDV(4) are depicted in Fig. 15, where it can be seen that the KPCA-based Q statistic detected this fault at the 161-th sample. Our contribution plot method of Subsection 4.2 was then used to calculate the KPCA-based T^2 and Q contribution plots, which are shown in Fig. 16, where the average contribution value over the 161-th to 165-th samples was used. The KPCA-based monitoring charts under Fault IDV(6) given in Fig. 17 shows that the KPCA-based Q statistic indicated a fault at the 161-th sample. The KPCA-based T^2 and Q contribution plots, averaged over the 161-th to 165-th samples, were then constructed and shown in Fig. 18. The fault identification results of Figs. 16 and 18 are similar to those of Figs. 10 and 13. More specifically, the quality of the LKPCA-based T^2 contribution plot is slightly better than that of the KPCA-based T^2 contribution plot in these two cases. These results confirm that our contribution plot approach for fault identification is a general one, suitable for both the KPCA and LKPCA.

Although the above simulation results have demonstrated the potential of our contribution plot technique in locating fault variables, the power of this technique should not be overstated. For complex fault cases, it is likely that many process variables will be involved and these variables may interact in complicated ways. Therefore, many variables may exhibit high contribution values, and it may become difficult to identify real fault variables using the proposed contribution plots. Clearly, further study is warranted to refine the proposed contribution plot approach or to develop alternatives for effectively locating the fault source in complex fault modes.
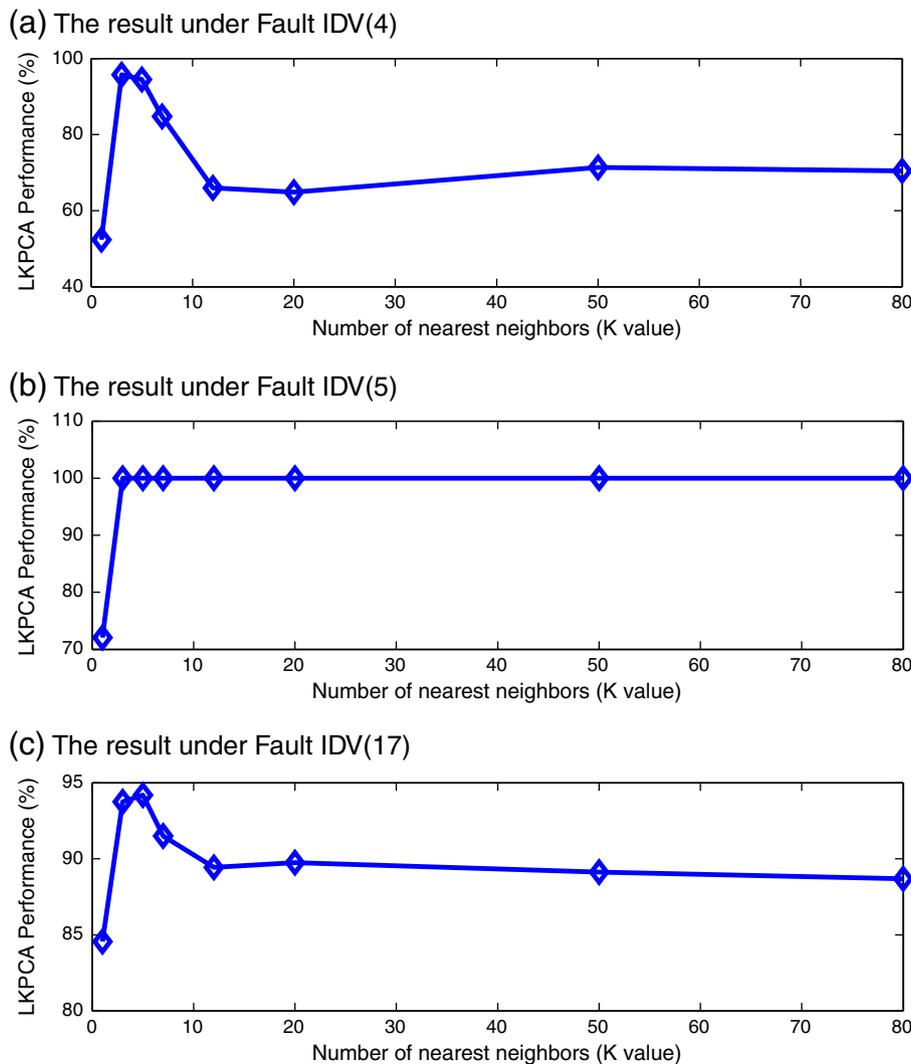
Fig. 19. The LKPCA-based monitoring performance in terms of the average fault alarming rate of T^2 and Q statistics under three fault cases.
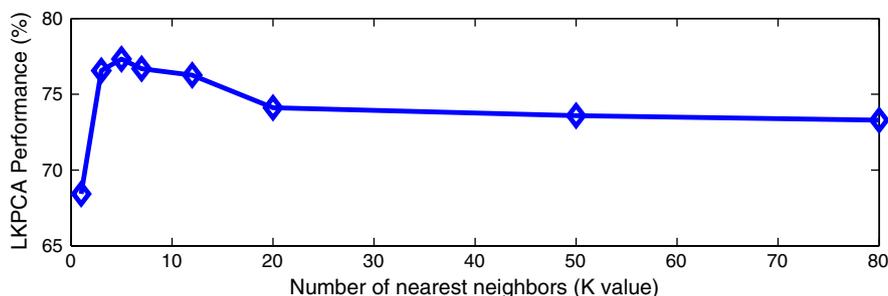
Fig. 20. The LKPCA-based monitoring performance in terms of the average fault alarming rate of T^2 and Q statistics, averaged over all the 21 fault cases.

5.3. Selection of K value for adjacency graph

In the LKPCA-based monitoring, the value of K is an important parameter in determining the weightings $w_{i,j}$ in the optimisation objective function of local structure analysis. If K is set to zero or a too small value, the LKPCA optimisation process will not consider the local structure analysis or will consider the local structure of the data set insufficiently. By contrast, if K is set to a too large value, the LKPCA optimisation process will over-emphasise the importance of the local structure in the data set. Clearly, there exists an optimal choice of K value, which is problem dependant.

In our simulation study of the TE process, we chose $K = 5$ empirically. To elaborate further, this value was determined after an experiment of investigating the influence of K to the LKPCA-based monitoring performance, in terms of the average fault alarming rate of T^2 and Q statistics. In particular, the value of K was set to 1, 3, 5, 7, 12, 20, 50 and 80, and the average fault alarming rates of T^2 and Q statistics obtained under Faults IDV(4), IDV(5) and IDV(17) are shown in Fig. 19(a),(b) and (c), respectively. It can be seen that the best K values were $K = 3$, $K \geq 3$ and $K = 5$ under Faults IDV(4), IDV(5) and IDV(17), respectively. The average LKPCA-based monitoring performance over all the 21 fault cases was further investigated in Fig. 20, where it can be seen that the best choice of K was $K = 5$ on average.

6. Conclusions

A modified kernel principal component analysis, referred to as the LKPCA, has been proposed for process monitoring and fault diagnosis. Our novel contribution has been to integrate the local structure analysis, which is also important for process monitoring and fault diagnosis, into the global optimisation of the standard KPCA technique naturally. Monitoring statistics based on the proposed LKPCA method have been derived for fault detection. Extensive simulation results obtained on the Tennessee Eastman benchmark process have demonstrated that the proposed LKPCA method outperforms the standard KPCA method significantly, in terms of fault detection performance. Furthermore, a contribution plot technique has been developed based on sensitivity analysis of the LKPCA monitoring statistics to identify fault variables. This fault identification technique is computationally very simple and easy to implement, and its potential in locating fault source has been demonstrated in the simulation study. Moreover, the proposed contribution plot approach has been shown to be a general method, equally applicable to the standard KPCA based fault diagnosis.

Fault source diagnosis is the most difficult and challenging problem in nonlinear data-driven fault diagnosis, and identifying fault variables in complex fault modes remains an unsolved open problem. Future study is warranted to further enhance the proposed contribution plot approach and to develop more effective fault identification techniques. In particular, it is worth investigating an alternative fault identification approach based on the method of [47,48].

References

[1] V. Venkatasubramanian, R. Rengaswamy, S.N. Kavuri, K. Yin, A review of process fault detection and diagnosis Part III: process history based methods, Computers and Chemical Engineering 27 (3) (March 2003) 327–346.
[2] G.A. Cherry, S.J. Qin, Multiblock principal component analysis based on a combined index for semiconductor fault detection and diagnosis, IEEE Transactions on Semiconductor Manufacturing 19 (2) (May 2006) 159–172.
[3] R. Dunia, S.J. Qin, Joint diagnosis of process and sensor faults using principal component analysis, Control Engineering Practice 6 (4) (April 1998) 457–469.
[4] M.A. Krammer, Nonlinear principal component analysis using autoassociative neural networks, AICHE Journal 37 (2) (1991) 233–243.
[5] D. Dong, T.J. McAvoy, Nonlinear principal component analysis — based on principal curves and neural networks, Computers and Chemical Engineering 20 (1) (Jan. 1996) 65–78.
[6] H.G. Hiden, M.J. Willis, M.T. Tham, G.A. Montague, Non-linear principal components analysis using genetic programming, Computers and Chemical Engineering 23 (3) (Feb. 1999) 413–425.
[7] Z.-Q. Geng, Q.-X. Zhu, Multiscale nonlinear principal component analysis (NLPCA) and its application for chemical process monitoring, Industrial and Engineering Chemistry Research 44 (10) (2005) 3585–3593.
[8] B. Schölkpof, A.J. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation 10 (5) (July 1998) 1299–1319.
[9] J.-M. Lee, C.-K. Yoo, S.W. Choi, P.A. Vanrolleghem, I.-B. Lee, Nonlinear process monitoring using kernel principal component analysis, Chemical Engineering Science 59 (1) (Jan. 2004) 223–234.
[10] J.-M. Lee, C.-K. Yoo, I.-B. Lee, Fault detection of batch processes using multiway kernel principal component analysis, Computers and Chemical Engineering 28 (9) (Aug. 2004) 1837–1847.
[11] J.-H. Cho, J.-M. Lee, S.W. Choi, D.-K. Lee, I.-B. Lee, Fault identification for process monitoring using kernel principal component analysis, Chemical Engineering Science 60 (1) (Jan. 2005) 279–288.
[12] S.W. Choi, C.-K. Lee, J.-M. Lee, J.H. Park, I.-B. Lee, Fault detection and identification of nonlinear processes based on kernel PCA, Chemometrics and Intelligent Laboratory Systems 75 (1) (Jan. 2005) 55–67.
[13] X.-G. Deng, X.-M. Tian, Multivariate statistical process monitoring using multi-scale kernel principal component analysis, Proc. 6th IFAC Symp. Fault Detection, Supervision and Safety of Technical Processes (Beijing, China), Aug.29–Sept.1, 2006, 2006, pp. 108–112.
[14] S.W. Choi, J. Morris, I.-B. Lee, Nonlinear multiscale modeling for fault detection and identification, Chemical Engineering Science 63 (8) (April 2008) 2252–2266.
[15] Y. Zhang, C. Ma, Fault diagnosis of nonlinear processes using multiscale KPCA and multiscale KPLS, Chemical Engineering Science 66 (1) (Jan. 2011) 64–72.
[16] M. Zvokelj, S. Zupan, I. Prebil, Non-linear multivariate and multiscale monitoring and signal denoising strategy using kernel principal component analysis combined with ensemble empirical mode decomposition method, Mechanical Systems and Signal Processing 25 (7) (Oct. 2011) 2631–2653.
[17] X.-M. Tian, X.-L. Zhang, X.-G. Deng, S. Chen, Multiway kernel independent component analysis based on feature samples for batch process monitoring, Neurocomputing 72 (7-9) (March 2009) 1584–1596.
[18] P. Cui, J. Li, G. Wang, Improved kernel principal component analysis for fault detection, Expert Systems with Applications 34 (2) (Feb. 2008) 1210–1219.
[19] V.H. Nguyen, J.-C. Golinval, Fault detection based on kernel principal component analysis, Engineering Structures 32 (11) (Nov. 2010) 3683–3691.
[20] M. Jia, F. Chu, F. Wang, W. Wang, On-line batch process monitoring using batch dynamic kernel principal component analysis, Chemometrics and Intelligent Laboratory Systems 101 (2) (April 2010) 110–122.

[21] Y.-W. Zhang, H. Zhou, S.J. Qin, Decentralized fault diagnosis of large scale processes using multiblock kernel principal component analysis, Act Automatica Sinica 36 (4) (April 2010) 593–597.

[22] I.B. Khediri, M. Limam, C. Weihs, Variable window adaptive kernel principal component analysis for nonlinear nonstationary process monitoring, Computers and Industrial Engineering 61 (3) (Oct. 2011) 437–446.

[23] D.-S. Cao, Y.-Z. Liang, Q.-S. Xu, Q.-N. Hu, L.-X. Zhang, G.-H. Fu, Exploring nonlinear relationships in chemical data using kernel-based methods, Chemometrics and Intelligent Laboratory Systems 107 (1) (May 2011) 106–115.

[24] G.-H. Fu, D.-S. Cao, Q.-S. Xu, H.-D. Li, Y.-Z. Liang, Combination of kernel PCA and linear support vector machine for modeling a nonlinear relationship between bioactivity and molecular descriptors, Journal of Chemometrics 25 (2) (Feb. 2011) 92–99.

[25] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (Dec. 2000) 2323–2326.

[26] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensional reduction and data representation, Neural Computation 15 (6) (June 2003) 1373–1396.

[27] X. He, P. Niyogi, Locality preserving projections, in: S. Thrun, L.K. Saul, B. Schölkpf (Eds.), Advances in Neural Information Processing Systems, 16, MIT Press, 2004, pp. 153–160.

[28] X. He, D. Cai, W. Min, Statistical and computational analysis of locality preserving projection, Proc. 22nd Int. Conf. Machine Learning (Bonn, Germany), Aug. 7–11, 2005, 2005, pp. 281–288.

[29] W. Zhang, B. Li, W. Zhou, A LLE-based approach to sensor fault detection, Proc. Int. Joint Conf. Neural Networks (Hong Kong, China), June 1–8, 2008, 2008, pp. 2425–2429.

[30] B. Li, Y. Zhang, Supervised locally linear embedding projection (SLLEP) for machinery fault diagnosis, Mechanical Systems and Signal Processing 25 (8) (Nov. 2011) 3125–3134.

[31] Q. Jiang, M. Jia, J. Hu, F. Xu, Machinery fault diagnosis using supervised manifold learning, Mechanical Systems and Signal Processing 23 (7) (Oct. 2009) 2301–2311.

[32] Q. Jiang, M. Jia, J. Hu, F. Xu, Modified Laplacian eigenmap method for fault diagnosis, Chinese Journal of Mechanical Engineering 21 (3) (2008) 90–93.

[33] K. Hu, J. Yuan, Multivariate statistical process control based on multiway locality preserving projections, Journal of Process Control 18 (7–8) (2008) 797–807.

[34] J.-D. Shao, G. Rong, J.M. Lee, Generalized orthogonal locality preserving projections for nonlinear fault detection and diagnosis, Chemometrics and Intelligent Laboratory Systems 96 (1) (March 2009) 75–83.

[35] J.-B. Yu, Bearing performance degradation assessment using locality preserving projections, Expert Systems with Applications 38 (6) (June 2011) 7440–7450.

[36] J.-B. Yu, Bearing performance degradation assessment using locality preserving projections and Gaussian mixture models, Mechanical Systems and Signal Processing 25 (7) (Oct. 2011) 257–2588.

[37] M. Zhang, Z. Ge, Z. Song, R. Fu, Global–local structure analysis model and its application for fault detection and identification, Industrial and Engineering Chemistry Research 50 (11) (April 2011) 6837–6848.

[38] J.-M. Lee, C.K. Yoo, I.-B. Lee, Statistical process monitoring with independent component analysis, Journal of Process Control 14 (5) (Aug. 2004) 467–485.

[39] E. Parzen, On estimation of a probability density function and mode, Annals of Mathematical Statistics 33 (3) (Sept. 1962) 1066–1076.

[40] B.W. Silverman, Density Estimation, Chapman & Hall, London, 1996.

[41] L.H. Chiang, E.L. Russell, R.D. Braatz, Fault Detection and Diagnosis in Industrial Systems, Springer-Verlag, London, 2001.

[42] J.A. Westerhuis, S.P. Gurden, A.K. Smilde, Generalized contribution plots in multivariate statistical process monitoring, Chemometrics and Intelligent Laboratory Systems 51 (1) (May 2000) 95–114.

[43] L. Petzold, S. Li, Y. Cao, R. Serban, Sensitivity analysis of differential-algebraic equations and partial differential equations, Computers and Chemical Engineering 30 (10–12) (Sept. 2006) 1553–1559.

[44] J.J. Downs, E.F. Vogel, A plant-wide industrial process control problem, Computers and Chemical Engineering 17 (3) (March 1993) 245–255.

[45] N. Lu, F. Wang, F. Gao, Combination method of principal component and wavelet analysis for multivariate process monitoring and fault diagnosis, Industrial and Engineering Chemistry Research 42 (18) (July 2003) 4198–4207.

[46] J. Lee, B. Kang, S.-H. Kang, Integrating independent component analysis and local outlier factor for plant-wide process monitoring, Journal of Process Control 21 (7) (July 2011) 1011–1021.

[47] G.J. Postma, P.W.T. Krooshof, L.M.C. Buydens, Opening the kernel of kernel partial least squares and support vector machines, Analytica Chimica Acta 705 (1–2) (Oct. 2011) 123–134.

[48] P.W.T. Krooshof, B. Ustün, G.J. Postma, L.M.C. Buydens, Visualization and recovery of the (bio)chemical interesting variables in data analysis with support vector machine classification, Analytical Chemistry 82 (16) (Aug. 2010) 7000–7007.