ELSEVIER

Chemical Engineering Science 59 (2004) 223–234

Chemical Engineering Science

www.elsevier.com/locate/ces

Nonlinear process monitoring using kernel principal component analysis

Jong-Min Lee[a], ChangKyoo Yoo[b], Sang Wook Choi[a], Peter A. Vanrolleghem[b], In-Beum Lee[a,*]

[a] *Department of Chemical Engineering, Pohang University of Science and Technology, San 31 Hyoja Dong, Pohang 790-784, South Korea*
[b] *BIOMATH, Ghent University, Coupure Links 653, B-9000 Gent, Belgium*

Received 9 May 2003; received in revised form 24 July 2003; accepted 5 September 2003

Abstract

In this paper, a new nonlinear process monitoring technique based on kernel principal component analysis (KPCA) is developed. KPCA has emerged in recent years as a promising method for tackling nonlinear systems. KPCA can efficiently compute principal components in high-dimensional feature spaces by means of integral operators and nonlinear kernel functions. The basic idea of KPCA is to first map the input space into a feature space via nonlinear mapping and then to compute the principal components in that feature space. In comparison to other nonlinear principal component analysis (PCA) techniques, KPCA requires only the solution of an eigenvalue problem and does not entail any nonlinear optimization. In addition, the number of principal components need not be specified prior to modeling. In this paper, a simple approach to calculating the squared prediction error (SPE) in the feature space is also suggested. Based on T^2 and SPE charts in the feature space, KPCA was applied to fault detection in two example systems: a simple multivariate process and the simulation benchmark of the biological wastewater treatment process. The proposed approach effectively captured the nonlinear relationship in the process variables and showed superior process monitoring performance compared to linear PCA.

Keywords: Kernel principal component analysis; Nonlinear dynamics; Fault detection; Systems engineering; Safety; Process control

1. Introduction

The monitoring of chemical and biological processes and the diagnosis of faults in those processes are very important aspects of process systems engineering because they are integral to the successful execution of planned operations and to improving process productivity. In recently designed industrial process plants, numerous variables are measured in various operating units, and these variables are recorded at many time points. The resulting data sets are highly correlated and are subject to considerable noise. In the absence of an appropriate method for processing such data, only limited information can be extracted and consequently plant operators have only a poor understanding of the process. This lack of understanding leads to unstable operation. However, if properly processed, the abundance of process data recorded in modern plants can provide a wealth of information, enabling plant operators to understand the status of the process and therefore to take appropriate actions when abnormalities are detected.

Traditionally, statistical process control (SPC) charts such as Shewhart, CUSUM and EWMA charts have been used to monitor processes and improve product quality. However, such univariate control charts show poor fault detection performance when applied to multivariate processes. This shortcoming of univariate control charts has led to the development of many process monitoring schemes that use multivariate statistical methods based on principal component analysis (PCA) and partial least squares (PLS) (Nomikos and MacGregor, 1995; Ku et al., 1995; Wise and Gallagher, 1996; Dong and McAvoy, 1996; Bakshi, 1998). PCA is the most widely used data-driven technique for process monitoring on account of its ability to handle high-dimensional, noisy, and highly correlated data by projecting the data onto a lower-dimensional subspace that contains most of the variance of the original data (Wise and Gallagher, 1996). By applying multivariate statistics to the lower-dimensional data representations produced by PCA, faults can be detected and diagnosed with greater proficiency. However, for some complicated cases in industrial chemical and biological processes with particularly nonlinear characteristics, PCA performs poorly due to its assumption that the process data are linear (Dong and McAvoy, 1996).

[*] Corresponding author. Tel.: +82-54-279-2274; fax: +82-54-279-3499.
E-mail address: iblee@postech.ac.kr (I.-B. Lee).

0009-2509/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ces.2003.09.012

To handle the problem posed by nonlinear data, Kramer (1991) developed a nonlinear PCA method based on auto-associative neural networks. However, the network proposed by Kramer is difficult to train because it has five layers. Moreover, it is difficult to determine the number of nodes in each layer. Dong and McAvoy (1996) developed a nonlinear PCA approach based on principal curves and neural networks. However, their principal curve algorithm assumes that the nonlinear function can be approximated by a linear combination of several univariate functions, and thus that it can be decomposed into a sum of functions of the individual variables. Such mappings can only be made for a limited class of nonlinear models, restricting the application of the principal curve algorithm to the identification of structures that exhibit additive-type behavior (Jia et al., 2001). Furthermore, a nonlinear optimization problem has to be solved to compute the principal curves and train the neural networks, and the number of principal components (PCs) must be specified in advance before training the neural networks. That is, if the number of PCs changes, the modeling procedure using the neural networks must be performed again. Alternative nonlinear PCA methods based on an input-training neural network (Jia et al., 2001) and on genetic programming (Hiden et al., 1999) have also been developed.

A new nonlinear PCA technique for tackling the nonlinear problem, called kernel PCA (KPCA), has been in development in recent years (Schölkopf et al., 1998; Mika et al., 1999; Romdhani et al., 1999). KPCA can efficiently compute PCs in high-dimensional feature spaces by means of integral operators and nonlinear kernel functions. The basic idea of KPCA is to first map the input space into a feature space via nonlinear mapping and then to compute the PCs in that feature space. For any given algorithm that can be expressed solely in terms of dot products (i.e., without explicit use of the variables themselves), this kernel method enables the construction of different nonlinear versions of the original algorithm (Christianini and Shawe-Taylor, 2000). Compared to other nonlinear methods, the main advantage of KPCA is that it does not involve nonlinear optimization (Schölkopf et al., 1998); it essentially requires only linear algebra, making it as simple as standard PCA. KPCA requires only the solution of an eigenvalue problem, and due to its ability to use different kernels, it can handle a wide range of nonlinearities. In addition, KPCA does not require that the number of components to be extracted be specified prior to modeling. Due to these merits, KPCA has shown better performance than linear PCA in feature extraction and classification in nonlinear systems (Schölkopf et al., 1998, 1999). However, the original KPCA method of Schölkopf et al. (1998) provides only nonlinear PCs and does not provide any method for reconstructing the data in the feature space. Thus, the direct application of KPCA to process monitoring is problematic because the monitoring chart of the squared prediction error (SPE) cannot be generated.

In this paper, we propose a new nonlinear process monitoring technique based on KPCA. A simple calculation of the SPE in the feature space is also suggested. The monitoring charts of T^2 and SPE are constructed in the feature space. The paper is organized as follows. The concept of KPCA is introduced in Section 2. In Section 3, the KPCA-based on-line monitoring strategy is presented. The superiority of process monitoring using KPCA is illustrated in Section 4 through two examples of a simple multivariate process and the wastewater simulation benchmark. Finally, we present our conclusions in Section 5.

2. KPCA

The key idea of KPCA is both intuitive and generic. In general, PCA can only be effectively performed on a set of observations that vary linearly. When the variations are nonlinear, the data can always be mapped into a higher-dimensional space in which they vary linearly. That is, according to Cover's theorem, the nonlinear data structure in the input space is more likely to be linear after high-dimensional nonlinear mapping (Haykin, 1999). This higher-dimensional linear space is referred to as the *feature space* (F). KPCA finds a computationally tractable solution through a simple kernel function that intrinsically constructs a nonlinear mapping from the input space to the feature space. As a result, KPCA performs a nonlinear PCA in the input space (Romdhani et al., 1999).

If a PCA is aimed at decoupling nonlinear correlations among a given set of data (with zero mean), $\mathbf{x}_k \in R^m$, $k = 1,\dots,N$ through diagonalizing their covariance matrix, the covariance can be expressed in a linear feature space F instead of the nonlinear input space, i.e.,

$$\mathbf{C}^F = \frac{1}{N}\sum_{j=1}^{N} \Phi(\mathbf{x}_j)\Phi(\mathbf{x}_j)^{\mathrm{T}}, \tag{1}$$

where it is assumed that $\sum_{k=1}^{N} \Phi(\mathbf{x}_k) = 0$, and $\Phi(\cdot)$ is a nonlinear mapping function that projects the input vectors from the input space to F. Note that the dimensionality of the feature space can be arbitrarily large or possibly infinite (Schölkopf et al., 1998). To diagonalize the covariance matrix, one has to solve the eigenvalue problem in the feature space

$$\lambda\mathbf{v} = \mathbf{C}^F\mathbf{v}, \tag{2}$$

where eigenvalues $\lambda \geqslant 0$ and $\mathbf{v} \in F \setminus \{\mathbf{0}\}$. The $\mathbf{v}$ along with the largest λ obtained by Eq. (2) become the first PC in F, and the $\mathbf{v}$ along with the smallest λ become the last PC. Here, $\mathbf{C}^F\mathbf{v}$ can be expressed as follows:

$$\mathbf{C}^F\mathbf{v} = \left(\frac{1}{N}\sum_{j=1}^{N}\Phi(\mathbf{x}_j)\Phi(\mathbf{x}_j)^{\mathrm{T}}\right)\mathbf{v}$$
$$= \frac{1}{N}\sum_{j=1}^{N}\langle\Phi(\mathbf{x}_j),\mathbf{v}\rangle\Phi(\mathbf{x}_j), \tag{3}$$

where $\langle\mathbf{x},\mathbf{y}\rangle$ denotes the dot product between $\mathbf{x}$ and $\mathbf{y}$. This implies that all solutions $\mathbf{v}$ with $\lambda \neq 0$ must lie in the span of $\Phi(\mathbf{x}_1),\ldots,\Phi(\mathbf{x}_N)$. Hence $\lambda\mathbf{v} = \mathbf{C}^F\mathbf{v}$ is equivalent to

$$\lambda\langle\Phi(\mathbf{x}_k),\mathbf{v}\rangle = \langle\Phi(\mathbf{x}_k),\mathbf{C}^F\mathbf{v}\rangle, \quad k = 1,\ldots,N \tag{4}$$

and there exist coefficients $\alpha_i (i = 1,\ldots,N)$ such that

$$\mathbf{v} = \sum_{i=1}^{N}\alpha_i\Phi(\mathbf{x}_i). \tag{5}$$

Combining Eqs. (4) and (5), we obtain

$$\lambda\sum_{i=1}^{N}\alpha_i\langle\Phi(\mathbf{x}_k),\Phi(\mathbf{x}_i)\rangle$$
$$=\frac{1}{N}\sum_{i=1}^{N}\alpha_i\left\langle\Phi(\mathbf{x}_k),\sum_{j=1}^{N}\Phi(\mathbf{x}_j)\right\rangle\langle\Phi(\mathbf{x}_j),\Phi(\mathbf{x}_i)\rangle \tag{6}$$

for all $k = 1,\ldots,N$. Note that the eigenvalue problem in Eq. (6) only involves dot products of mapped shape vectors in the feature space. In general, the mapping $\Phi(\cdot)$ may not always be computationally tractable, although it exists. However, it need not be explicitly computed; only dot products of two vectors in the feature space are needed.

Now, let us define an $N \times N$ matrix $\mathbf{K}$ by $[\mathbf{K}]_{ij} = K_{ij} = \langle\Phi(x_i),\Phi(x_j)\rangle$. Then the left-hand side of Eq. (6) can be expressed as

$$\lambda\sum_{i=1}^{N}\alpha_i\langle\Phi(\mathbf{x}_k),\Phi(\mathbf{x}_i)\rangle = \lambda\sum_{i=1}^{N}\alpha_i K_{ki}. \tag{7}$$

Since $k = 1,\ldots,N$, Eq. (7) becomes $\lambda\mathbf{K}\alpha$. The right-hand side of Eq. (6) can be expressed as

$$\frac{1}{N}\sum_{i=1}^{N}\alpha_i\left\langle\Phi(\mathbf{x}_k),\sum_{j=1}^{N}\Phi(\mathbf{x}_j)\right\rangle\langle\Phi(\mathbf{x}_j),\Phi(\mathbf{x}_i)\rangle$$
$$=\frac{1}{N}\sum_{i=1}^{N}\alpha_i\sum_{j=1}^{N}K_{kj}K_{ji}. \tag{8}$$

Since $k = 1,\ldots,N$, Eq. (8) becomes $(1/N)\mathbf{K}^2\alpha$. Combining Eqs. (7) and (8), we obtain

$$\lambda N\mathbf{K}\alpha = \mathbf{K}^2\alpha, \tag{9}$$

where $\alpha = [\alpha_1,\ldots,\alpha_N]^{\mathrm{T}}$. To find solutions of Eq. (9), we solve the eigenvalue problem

$$N\lambda\alpha = \mathbf{K}\alpha \tag{10}$$

for nonzero eigenvalues. A justification of this procedure is given in Schölkopf et al. (1998). Now, performing PCA in F is equivalent to resolving the eigen-problem of Eq. (10). This yields eigenvectors $\alpha_1,\alpha_2,\ldots,\alpha_N$ with eigenvalues $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_N$. The dimensionality of the problem can be reduced by retaining only the first p eigenvectors. We normalize $\alpha_1,\alpha_2,\ldots,\alpha_p$ by requiring that the corresponding vectors in F be normalized, i.e.,

$$\langle\mathbf{v}_k,\mathbf{v}_k\rangle = 1 \quad \text{for all } k = 1,\ldots,p. \tag{11}$$

Using $\mathbf{v}_k = \sum_{i=1}^{N}\alpha_i^k\Phi(\mathbf{x}_i)$, Eq. (11) leads to

$$1 = \left\langle\sum_{i=1}^{N}\alpha_i^k\Phi(\mathbf{x}_i),\sum_{j=1}^{N}\alpha_j^k\Phi(\mathbf{x}_j),\right\rangle$$
$$=\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i^k\alpha_j^k\langle\Phi(\mathbf{x}_i),\Phi(\mathbf{x}_j)\rangle$$
$$=\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i^k\alpha_j^k K_{ij} = \langle\alpha_k,\mathbf{K}\alpha_k\rangle = \lambda_k\langle\alpha_k,\alpha_k\rangle. \tag{12}$$

The PCs $\mathbf{t}$ of a test vector $\mathbf{x}$ are then extracted by projecting $\Phi(\mathbf{x})$ onto eigenvectors $\mathbf{v}_k$ in F, where $k = 1,\ldots,p$.

$$t_k = \langle\mathbf{v}_k,\Phi(\mathbf{x})\rangle = \sum_{i=1}^{N}\alpha_i^k\langle\Phi(\mathbf{x}_i),\Phi(\mathbf{x})\rangle. \tag{13}$$

To solve the eigenvalue problem of Eq. (10) and to project from the input space into the KPCA space using Eq. (13), one can avoid performing the nonlinear mappings and computing both the dot products in the feature space by introducing a *kernel function* of form $k(\mathbf{x},\mathbf{y}) = \langle\Phi(\mathbf{x}),\Phi(\mathbf{y})\rangle$ (Schölkopf et al., 1998; Romdhani et al., 1999).

There exist a number of kernel functions. According to Mercer's theorem of functional analysis, there exists a mapping into a space where a kernel function acts as a dot product if the kernel function is a continuous kernel of a positive integral operator. Hence, the requirement on the kernel function is that it satisfies Mercer's theorem (Christianini and Shawe-Taylor, 2000). Representative kernel functions are as follows:

Polynomial kernel:

$$k(\mathbf{x},\mathbf{y}) = \langle\mathbf{x},\mathbf{y}\rangle^d, \tag{14}$$

Sigmoid kernel:

$$k(\mathbf{x},\mathbf{y}) = \tanh(\beta_0\langle\mathbf{x},\mathbf{y}\rangle + \beta_1), \tag{15}$$

Radial basis kernel:

$$k(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{c}\right), \tag{16}$$

where d, β_0, β_1 and c are specified a priori by the user. The polynomial kernel and radial basis kernel always satisfy Mercer's theorem, whereas the sigmoid kernel satisfies it only for certain values of β_0 and β_1 (Haykin, 1999). These

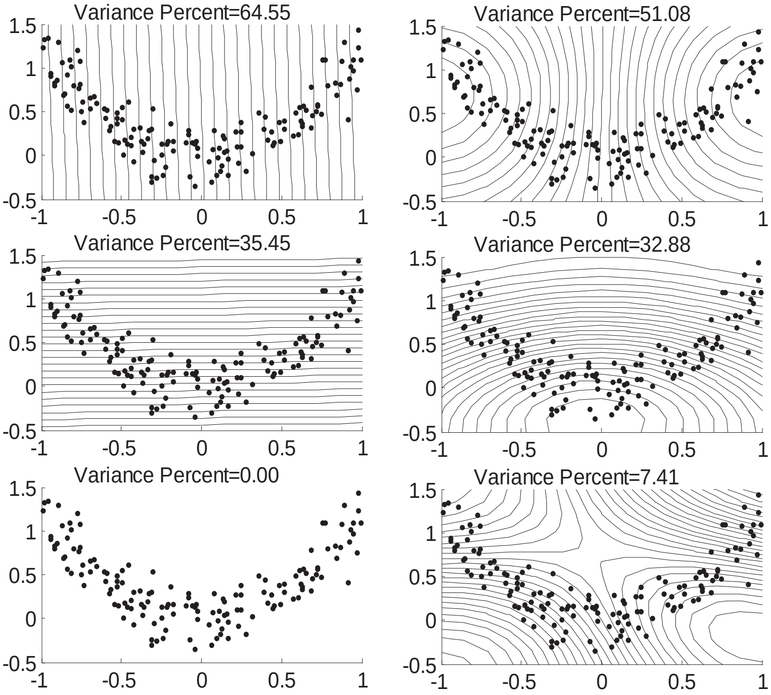

Fig. 1. Motivational example of KPCA (left column: linear PCA, right column: KPCA).

kernel functions provide a low-dimensional KPCA subspace that represents the distributions of the mapping of the training vectors in the feature space. A specific choice of kernel function implicitly determines the mapping Φ and the feature space F.

Before applying KPCA, mean centering in the high-dimensional space should be performed. This can be done by substituting the kernel matrix $\mathbf{K}$ with

$$\tilde{\mathbf{K}} = \mathbf{K} - \mathbf{1}_N\mathbf{K} - \mathbf{K}\mathbf{1}_N + \mathbf{1}_N\mathbf{K}\mathbf{1}_N, \tag{17}$$

where

$$\mathbf{1}_N = \frac{1}{N}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \in R^{N\times N}.$$

For further details, see the paper of Schölkopf et al. (1998).

2.1. Motivational example of KPCA

To obtain some insight into how KPCA behaves in the input space, we consider the following example (Schölkopf et al., 1998). A two-dimensional data set with 150 samples is generated in the following way: x-values have a uniform distribution in $[-1, 1]$, and y-values are generated using $y_i = 1.2x_i^2 + \xi$, where $\xi \sim N(0, 0.04)$. In this case, a radial basis kernel $k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/2)$ is used for KPCA. Fig. 1 shows the results obtained when linear PCA (left column) and KPCA (right column) are applied to this data set. As shown in Fig. 1, linear PCA results in only two nonzero eigenvalues, because the dimensionality of the input space is two. The first PC explains 64.55% of the data variance and the second PC captures the remaining variance. In contrast, KPCA permits the extraction of further components. Here only three components of KPCA are considered. The contour lines shown in each part of the figure (except for the zero eigenvalue in the case of linear PCA) represent constant principal values; that is, all points on a particular contour have the same principal value. In the case of linear PCA, the contour lines are orthogonal to the eigenvectors. Hence, in Fig. 1, the first and second eigenvectors correspond to the directions of the x- and y-axis, respectively. Because linear PCA produces straight contour lines, it cannot capture the nonlinear structure in the data. In contrast, the first PC of KPCA varies monotonically along the parabola that underlies the data (right column of Fig. 1). Consequently, KPCA produces contour lines of constant feature values which capture the nonlinear structure in the data better than linear PCA, although the eigenvectors cannot be drawn because they are in a higher-dimensional feature space (Schölkopf et al., 1998).

3. On-line monitoring strategy of KPCA

The simple motivational example in the previous section demonstrated the ability of KPCA to capture nonlinear structure in data that is missed by linear PCA. We now present a new monitoring method that exploits the merits of KPCA. The KPCA-based monitoring method is similar

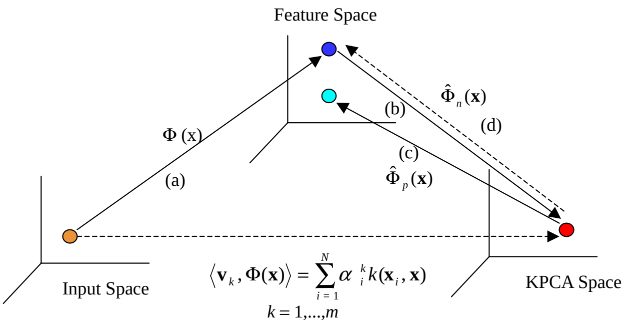
Feature Space
$\hat{\Phi}_n(\mathbf{x})$
(b)
(d)
Φ (x)
(c)
$\hat{\Phi}_p(\mathbf{x})$
(a)
Input Space
$\langle \mathbf{v}_k, \Phi(\mathbf{x}) \rangle = \sum_{i=1}^{N} \alpha_i^k k(\mathbf{x}_i, \mathbf{x})$
$k = 1,\ldots,m$
KPCA Space

Fig. 2. Conceptual diagram of KPCA.

to that using PCA in that Hotelling's T^2 statistic and the Q-statistic in the feature space can be interpreted in the same way.

A measure of the variation within the KPCA model is given by Hotelling's T^2 statistic. T^2 is the sum of the normalized squared scores, and is defined as

$$T^2 = [t_1, \ldots, t_p]\Lambda^{-1}[t_1, \ldots, t_p]^{\mathrm{T}}, \tag{18}$$

where t_k is obtained from Eq. (13) and $\mathbf{\Lambda}^{-1}$ is the diagonal matrix of the inverse of the eigenvalues associated with the retained PCs. The confidence limit for T^2 is obtained using the F-distribution:

$$T^2_{p,N,\alpha} \sim \frac{p(N-1)}{N-p} F_{p,N-p,\alpha}, \tag{19}$$

where N is the number of samples in the model and p is the number of PCs.

The measure of goodness of fit of a sample to the PCA model is the squared prediction error (SPE), also known as the Q statistic. However, the KPCA method of Schölkopf et al. (1998) provides only nonlinear PCs and does not provide any method for reconstructing the data in the feature space. Hence, construction of SPE monitoring charts is problematic in the KPCA method. In this paper, we propose a simple calculation of SPE in the feature space F. The conceptual framework of the KPCA method is shown schematically in Fig. 2 (Romdhani et al., 1999). First, KPCA performs a nonlinear mapping $\Phi(\cdot)$ from an input vector $\mathbf{x}$ to a high-dimensional feature space F (step (a)). Then, a linear PCA is performed in this feature space, which gives score values t_k in a lower p-dimensional KPCA space (step (b)). In order to reconstruct a feature vector $\Phi(\mathbf{x})$ from t_k, t_k is projected into the feature space via $\mathbf{v}_k$, giving a reconstructed feature vector $\hat{\Phi}_p(\mathbf{x}) = \sum_{k=1}^{p} t_k\mathbf{v}_k$ (step (c)). Then the SPE in the feature space is defined as $\mathrm{SPE} = \|\Phi(\mathbf{x}) - \hat{\Phi}_p(\mathbf{x})\|^2$. Here, $\Phi(\mathbf{x})$ is identical to $\hat{\Phi}_n(\mathbf{x}) = \sum_{k=1}^{n} t_k\mathbf{v}_k$ if $p = n$, where n is the number of nonzero eigenvalues generated from Eq. (10) among the total N eigenvalues (step (d)). Hence, the SPE proposed here is obtained using the equations:

$$\begin{aligned}
\mathrm{SPE} &= \|\Phi(\mathbf{x}) - \hat{\Phi}_p(\mathbf{x})\|^2 = \|\hat{\Phi}_n(\mathbf{x}) - \hat{\Phi}_p(\mathbf{x})\|^2 \\
&= \hat{\Phi}_n(\mathbf{x})^{\mathrm{T}}\hat{\Phi}_n(\mathbf{x}) - 2\hat{\Phi}_n(\mathbf{x})^{\mathrm{T}}\hat{\Phi}_p(\mathbf{x}) + \hat{\Phi}_p(\mathbf{x})^{\mathrm{T}}\hat{\Phi}_p(\mathbf{x}) \\
&= \sum_{j=1}^{n} t_j\mathbf{v}_j^{\mathrm{T}} \sum_{k=1}^{n} t_k\mathbf{v}_k - 2\sum_{j=1}^{n} t_j\mathbf{v}_j^{\mathrm{T}} \sum_{k=1}^{p} t_k\mathbf{v}_k \\
&\quad + \sum_{j=1}^{p} t_j\mathbf{v}_j^{\mathrm{T}} \sum_{k=1}^{p} t_k\mathbf{v}_k \\
&= \sum_{j=1}^{n} t_j^2 - 2\sum_{j=1}^{p} t_j^2 + \sum_{j=1}^{p} t_j^2 = \sum_{j=1}^{n} t_j^2 - \sum_{j=1}^{p} t_j^2,
\end{aligned} \tag{20}$$

where $\mathbf{v}_j^{\mathrm{T}}\mathbf{v}_k = 1$ when $j = k$, $\mathbf{v}_j^{\mathrm{T}}\mathbf{v}_k = 0$ otherwise.

The confidence limit for the SPE can be computed from its approximate distribution

$$\mathrm{SPE}_\alpha \sim g\chi_h^2. \tag{21}$$

The control limits for the SPE are based on Box's equation and are obtained by fitting a weighted χ^2-distribution to the reference distribution generated from normal operating condition data (Nomikos and MacGregor, 1995). In Eq. (21), g is a weighting parameter included to account for the magnitude of SPE and h accounts for the degrees of freedom. If a and b are the estimated mean and variance of the SPE, then g and h can be approximated by $g = b/2a$ and $h = 2a^2/b$. Note that the method of matching moments may go wrong when the number of observations is small and there are outliers in the data; hence, the reference normal data should be carefully selected and many reference observations should be used. Furthermore, the use of a χ^2-distribution implicitly assumes the errors follow a Gaussian distribution, which may not always be true in practice. However, because g and h are obtained directly from the moments of the sampling distribution of the normal operating condition data, the use of a weighted χ^2-distribution works well even in cases for which the errors do not follow a Gaussian distribution (Sprang et al., 2002).

3.1. *Outline of on-line KPCA monitoring*

3.1.1. *Developing the normal operating condition (NOC) model*

(1) Acquire normal operating data and normalize the data using the mean and standard deviation of each variable.

(2) Given a set of m-dimensional scaled normal operating data $\mathbf{x}_k \in R^m$, $k = 1, \ldots, N$, compute the kernel matrix $\mathbf{K} \in R^{N\times N}$ by $[\mathbf{K}]_{ij} = K_{ij} = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle = [k(\mathbf{x}_i, \mathbf{x}_j)]$.

(3) Carry out centering in the feature space for $\sum_{k=1}^{N} \tilde{\Phi}(\mathbf{x}_k) = 0$,

$$\tilde{\mathbf{K}} = \mathbf{K} - \mathbf{1}_N\mathbf{K} - \mathbf{K}\mathbf{1}_N + \mathbf{1}_N\mathbf{K}\mathbf{1}_N, \tag{22}$$

where,

$$\mathbf{1}_N = \frac{1}{N}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \in R^{N\times N}.$$

(4) Solve the eigenvalue problem $N\lambda\alpha = \tilde{\mathbf{K}}\alpha$ and normalize α_k such that $\langle \alpha_k, \alpha_k\rangle = 1/\lambda_k$.

(5) For normal operating data $\mathbf{x}$, extract a nonlinear component via

$$t_k = \langle \mathbf{v}_k, \tilde{\Phi}(\mathbf{x})\rangle = \sum_{i=1}^{N} \alpha_i^k \langle \tilde{\Phi}(\mathbf{x}_i), \tilde{\Phi}(\mathbf{x})\rangle = \sum_{i=1}^{N} \alpha_i^k \tilde{k}(\mathbf{x}_i, \mathbf{x}). \tag{23}$$

(6) Calculate the monitoring statistics (T^2 and SPE) of the normal operating data.

(7) Determine the control limits of the T^2 and SPE charts.

3.1.2. On-line monitoring

(1) Obtain new data for each sample and scale it with the mean and variance obtained at step 1 of the modeling procedure.

(2) Given the m-dimensional scaled test data $\mathbf{x}_t \in R^m$, compute the kernel vector $\mathbf{k}_t \in R^{1\times N}$ by $[\mathbf{k}_t]_j = [k_t(\mathbf{x}_t, \mathbf{x}_j)]$ where $\mathbf{x}_j$ is the normal operating data $\mathbf{x}_j \in R^m$, $j = 1, \ldots, N$.

(3) Mean center the test kernel vector $\mathbf{k}_t$ as follows:

$$\tilde{k}_t = \mathbf{k}_t - \mathbf{1}_t\mathbf{K} - \mathbf{k}_t\mathbf{1}_N + \mathbf{1}_t\mathbf{K}\mathbf{1}_N, \tag{24}$$

where $\mathbf{K}$ and $\mathbf{1}_N$ are obtained from step 2 of the modeling procedure and $\mathbf{1}_t = 1/N[1, \ldots, 1] \in R^{1\times N}$.

(4) For the test data $\mathbf{x}_t$, extract a nonlinear component via

$$t_k = \langle \mathbf{v}_k, \tilde{\Phi}(\mathbf{x}_t)\rangle = \sum_{i=1}^{N} \alpha_i^k \langle \tilde{\Phi}(\mathbf{x}_i), \tilde{\Phi}(\mathbf{x}_t)\rangle = \sum_{i=1}^{N} \alpha_i^k \tilde{k}_t(\mathbf{x}_i, \mathbf{x}_t). \tag{25}$$

(5) Calculate the monitoring statistics (T^2 and SPE) of the test data.

(6) Monitor whether T^2 or SPE exceeds its control limit calculated in the modeling procedure.

4. Simulation results

In this section, the monitoring results of PCA and KPCA are compared for two case studies. The proposed monitoring method was applied to fault detection in both a simple example and the simulation benchmark of the biological wastewater treatment process (WWTP). In this paper, a radial basis kernel function, $k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/c)$, is selected as the kernel function with $c = rm\sigma^2$, where r is a constant that is determined by consideration of the process to be monitored, m is the dimension of the input space, and σ^2 is the variance of the data (Mika et al., 1999). After testing the monitoring performance for various values of c, we found that $c = 10m\sigma^2$ is appropriate for monitoring processes with various faults. Although the value of c is dependent upon the system under study, we found that the radial basis kernel is the best for monitoring the nonlinear processes used as examples in the present work.

When designing the PCA model, we must determine the number of PCs. This number should be determined by considering both the curse of dimensionality and loss of data information. Several techniques exist for determining the number of PCs, none of which has emerged as the dominant technique (Chiang et al., 2001). These techniques include SCREE tests on the residual percent variance, the average eigenvalue approach, parallel analysis, cross-validation of prediction residual sum of squares (PRESS), Akaike Information Criterion (AIC), and the variance of the reconstruction error criterion (Valle et al., 1999). For linear PCA, we used a cross-validation method (Wold, 1978) based on PRESS to determine the number of PCs. For KPCA, we employed the cut-off method using the average eigenvalue to determine the number of PCs due to its simplicity and robustness. The cross-validation method may also be used for KPCA. Note that the magnitude of each eigenvalue reflects the variance of the corresponding PC. The average eigenvalue approach has proved quite popular as a criterion for choosing the number of PCs. This criterion accepts all eigenvalues with values above the average eigenvalue and rejects those below the average (Valle et al., 1999). The justification for this approach is that PCs contributing less than the average variable are insignificant. In general, the number of PCs selected for KPCA is larger than that for linear PCA because KPCA extracts the major PCs from the infinite high-dimensional feature space whereas linear PCA extracts the major PCs from the finite input-dimensional space. According to Schölkopf et al. (1999), KPCA has the potential to utilize more PCs to code structure rather than noise; hence, KPCA outperforms linear PCA in denoising if a sufficiently large number of PCs is used. In KPCA monitoring, the use of numerous PCs may cause type I error to increase. However, if the number of PCs in KPCA is chosen by an appropriate method, the level of type I error is acceptable for monitoring, as seen in the following examples.

4.1. A simple example

Consider the following system with three variables but only one factor, originally suggested by Dong and McAvoy (1996):

$$x_1 = t + e_1, \tag{26}$$

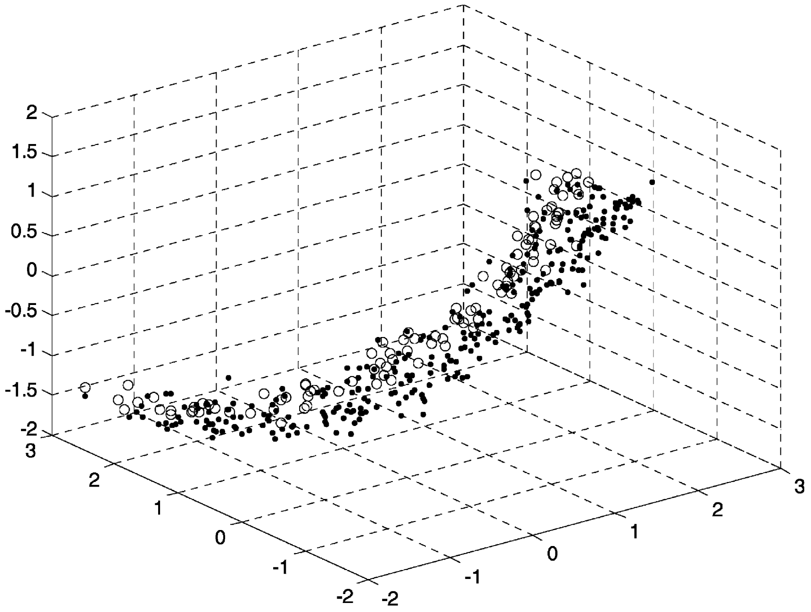

Fig. 3. Scaled data distribution of normal operating condition data (o) and disturbance 1 (●) (simple example).

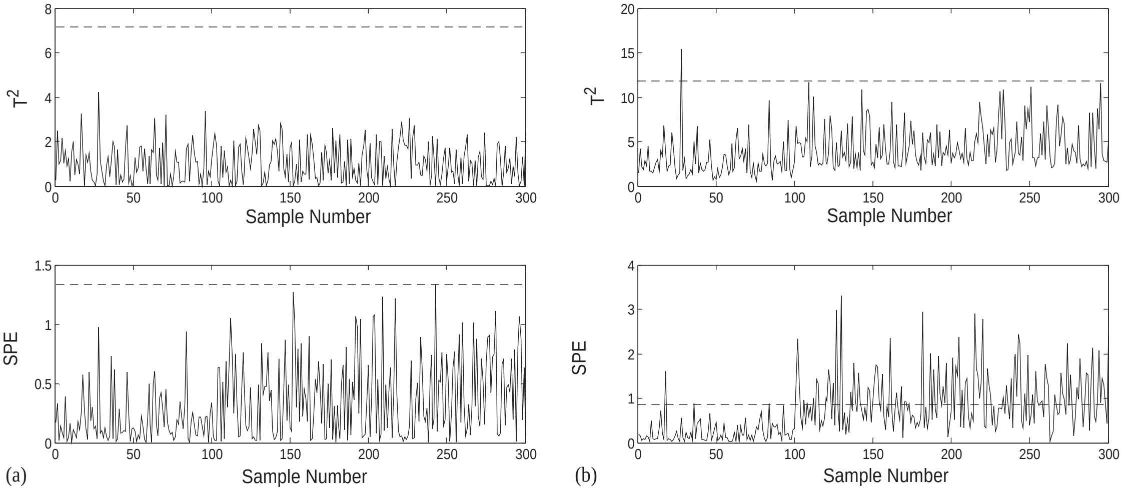

Fig. 4. (a) PCA monitoring charts, and (b) KPCA monitoring charts with disturbance 1 (simple example).

$$x_2 = t^2 - 3t + e_2, \tag{27}$$

$$x_3 = -t^3 + 3t^2 + e_3, \tag{28}$$

where e_1, e_2, and e_3 are independent noise variables $N(0, 0.01)$, and $t \in [0.01, 2]$. Normal data comprising 100 samples were generated according to these equations. These data were scaled to zero mean and unit variance. For this system, one PC was selected by cross-validation to model the linear PCA and three PCs were chosen to model KPCA by the average eigenvalue approach. Two sets of test data comprising 300 samples each were also generated. The following two disturbances were applied separately during generation of the test data sets:

Disturbance 1: A step change of x_2 by -0.4 was introduced starting from sample 101.

Disturbance 2: x_1 was linearly increased from sample 101 to 270 by adding $0.01(k - 100)$ to the x_1 value of each sample in this range, where k is the sample number.

The scaled data distribution of the normal operating condition data and the test data with disturbance 1 are plotted in Fig. 3. This figure clearly shows that this system is nonlinear and that it is difficult to identify the disturbance from normal operating data.

The T^2 and SPE charts for PCA monitoring of the process with disturbance 1 are shown in Fig. 4(a). The 99% confidence limits are also shown in this figure. It is evident from these charts that PCA does not detect disturbance 1; it

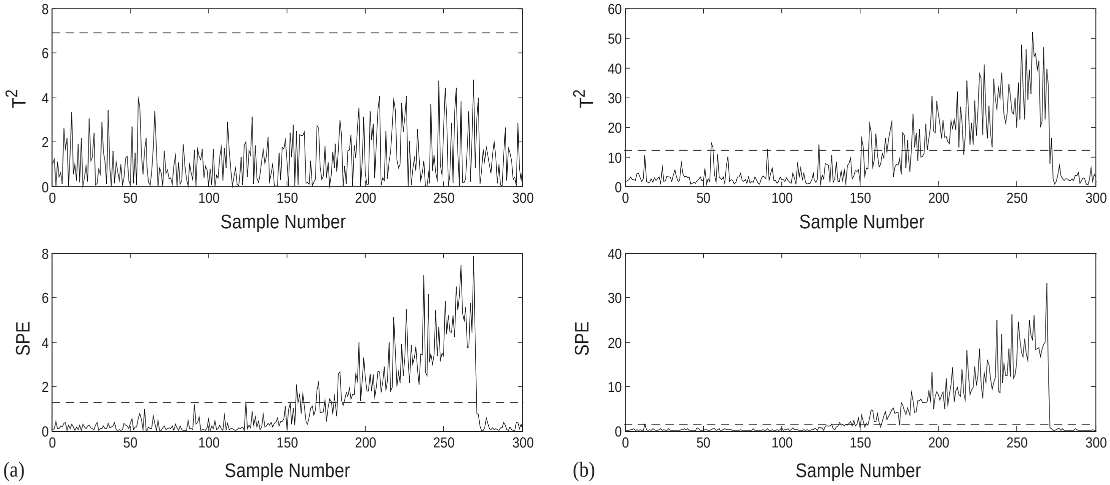

Fig. 5. (a) PCA monitoring charts, and (b) KPCA monitoring charts with disturbance 2 (simple example).

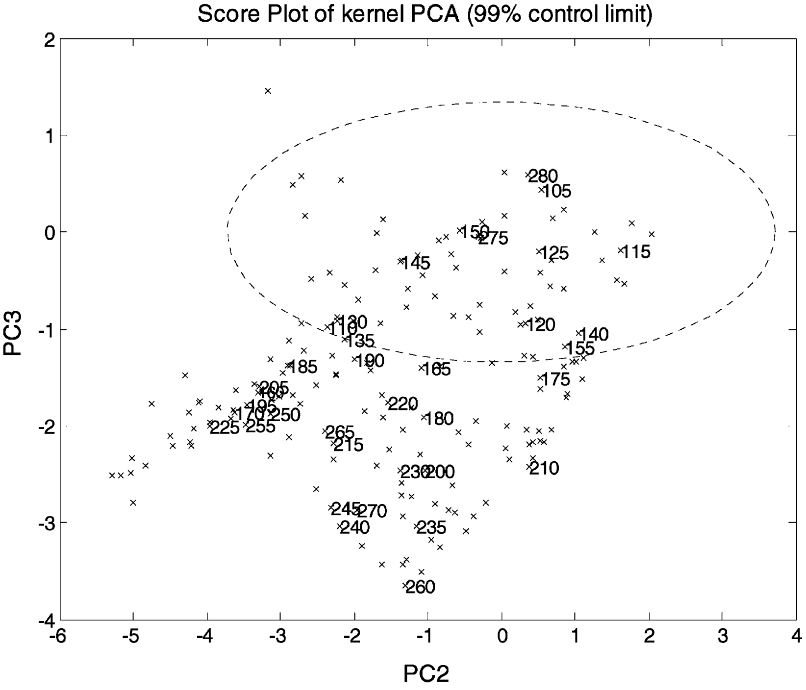

Fig. 6. Score plot (PC2 and PC3) of KPCA for the samples 101–280 with disturbance 2 (simple example).

only captures the dominant randomness. However, applying KPCA to the same process data gives the results presented in Fig. 4(b). KPCA shows relatively correct disturbance detection in comparison to PCA. In the SPE chart of Fig. 4(b), only one sample among the first 100 normal operating data exceeds the control limit, indicating that a $g\chi_h^2$ distribution provides a good approximation for the 99% control limit of the SPE chart. Overall, the level of type I errors is acceptable in both the T^2 and SPE charts of KPCA. The T^2 and SPE charts for PCA monitoring of the process with disturbance 2 are shown in Fig. 5(a). The SPE chart detects disturbance 2 from about sample 160 onwards whereas the T^2 chart does not detect any abnormalities. In addition, the SPE value rapidly decreases to normal operating condition levels after the disturbance is stopped at sample 270. In contrast to the PCA results, both the T^2 and SPE KPCA monitoring charts detect disturbance 2 (Fig. 5(b)). Moreover, the KPCA charts detect the disturbance earlier than does the SPE chart of PCA monitoring. Specifically, in the KPCA monitoring, abnormalities are detected from about sample 160 in the T^2 chart and from about sample 130 in the SPE chart, indicating that the KPCA method detects the disturbance 30 samples earlier than the PCA approach. In addition, both the T^2 and SPE values rapidly decrease to

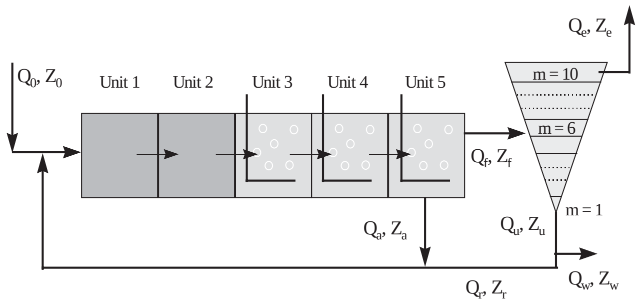

Fig. 7. Process layout for the simulation benchmark.

Table 1
Variables used in the monitoring of the benchmark model

No.	Symbol	Meaning
1	$S_{NH,in}$	Influent ammonium concentration
2	$\mathbf{Q}_{in}$	Influent flow rate
3	TSS_4	Total suspended solid (reactor 4)
4	$S_{O,3}$	Dissolved oxygen concentration (reactor 3)
5	$S_{O,4}$	Dissolved oxygen concentration (reactor 4)
6	K_La_5	Oxygen transfer coefficient (reactor 5)
7	$S_{NO,2}$	Nitrate concentration (reactor 2)

normal operating condition levels after the disturbance is stopped at sample 270. Fig. 6 shows a plot of the scores for PCs 2 and 3 calculated using KPCA for samples 101–280. This plot shows that the first deviation from normal operation appears at about sample 160 and propagates outside the normal operating region, reaching a maximum deviation at about sample 260. Then, the test data return to the normal operating region after about sample 271. This result shows that the extracted PCs of KPCA capture the nonlinear relationship in the process variables and push the fault data outside the normal operating region more efficiently than linear PCA.

4.2. Wastewater treatment process (WWTP)

The KPCA monitoring approach proposed here was also tested for its ability to detect small internal disturbances in simulated data obtained from a '*benchmark simulation*' of the WWTP (Spanjers et al., 1998). The activated sludge model no. 1 (ASM1) and a 10-layer settler model were used to simulate the biological reactions and the settling process, respectively. Fig. 7 shows a flow diagram of the modeled WWTP system. The plant is designed to treat an average flow of $20,000\ \mathrm{m}^3/\mathrm{day}$ with an average biodegradable chemical oxygen demand (COD) concentration of 300 mg/l. The plant consists of a five-compartment bioreactor ($6000\ \mathrm{m}^3$) and a secondary settler ($6000\ \mathrm{m}^3$). The first two compartments of the bioreactor are not aerated whereas the others are aerated. Ideal mixing is assumed to occur in all of the compartments, whereas the secondary settler is modelled with a one-dimensional series of 10 layers. For more information on this benchmark, refer to the website of the COST working group (http://www.ensic.u-nancy.fr/COSTWWTP).

Influent data and operation parameters developed by the working group on the benchmarking of wastewater treatment plants, COST 624, were used in the simulation (Spanjers et al., 1998). The training model was based on a normal operation period of 1 week of dry weather and a 2-week data set was used for validation. The sampling time was 15 min; hence each 24-h period consisted of 96 samples. The data used were the influent file and outputs with noise suggested by the benchmark. Among the many physical and biological variables in the benchmark, seven variables were selected to build the monitoring system (see Table 1). Among the selected variables, $S_{NH,in}$ and $S_{NO,2}$ are particularly important because they can be used to monitor the advanced biological nitrogen removal process, which is central to the benchmark simulation. The other variables (Q_{in}, TSS_5, $S_{O,3}$, $S_{O,4}$ and K_La_5) were chosen for on-line monitoring because they are routinely collected at the majority of wastewater treatment plants and provide information on the process status of the WWTP.

Two types of disturbance were tested using the proposed method: external disturbances and internal disturbances (Yoo et al., 2002). External disturbances are defined as those imposed upon the process from the outside, and are detectable when monitoring the influent characteristics. Internal disturbances are caused by changes within the process that affect the process behavior. For the external disturbance, two short-storm events were simulated, while a deterioration of nitrification was simulated as an internal disturbance. In the case of the storm events, both KPCA and linear PCA detected the disturbance well (data not shown). Below we concentrate on the internal disturbance to illustrate the superiority of KPCA over PCA. In the WWTP, deterioration of the nitrification rate can strongly affect the performance of the activated sludge; hence, its early detection is of importance. Although methods exist to measure the nitrification rate (e.g., by respirometry), none of these methods is straightforward (Vanrolleghem and Gillot, 2001). However, PCA or KPCA can be used to monitor the process because drifts in nitrification rate affect other process measurements. The internal disturbance was imposed by decreasing the nitrification rate in the biological reactor through a decrease in the specific growth rate of the autotrophs (μ_A). Two types of disturbance were considered: a step decrease and a linear decrease. In the step decrease case, at sample 288 the autotrophic growth rate was decreased rapidly from 0.5 to 0.44 day^{-1}. In the linear decrease case, from sample 288 the autotrophic growth rate was linearly decreased from 0.5 to 0.4 over 2 days (192 samples), after which the value of 0.4 was maintained until the end of the test data.

First, linear PCA was applied to the case of the step decrease of μ_A. The PCA model was able to capture most of the variability of the **X**-block using three PCs selected from

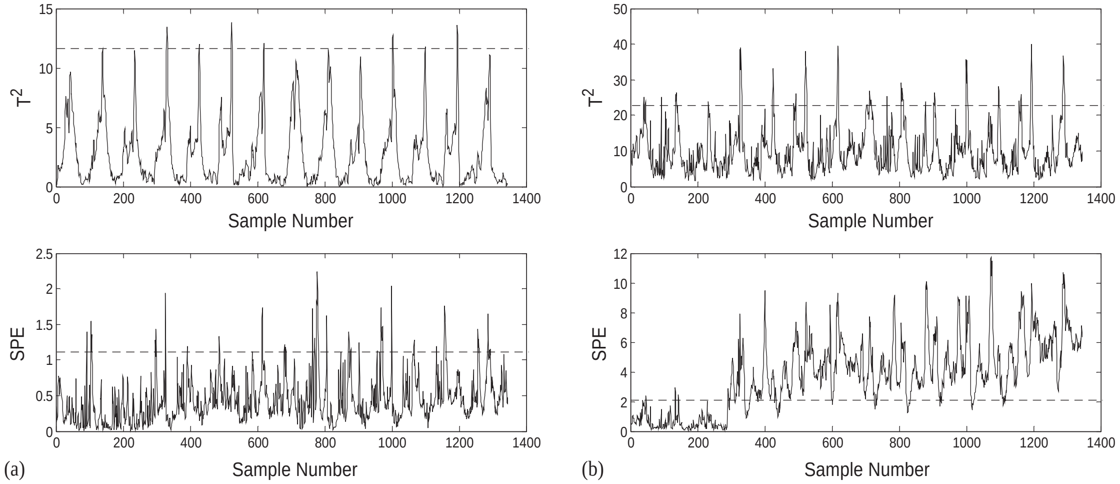

Fig. 8. (a) PCA monitoring charts, and (b) KPCA monitoring charts for the case of a step decrease in the nitrification rate (benchmark example).

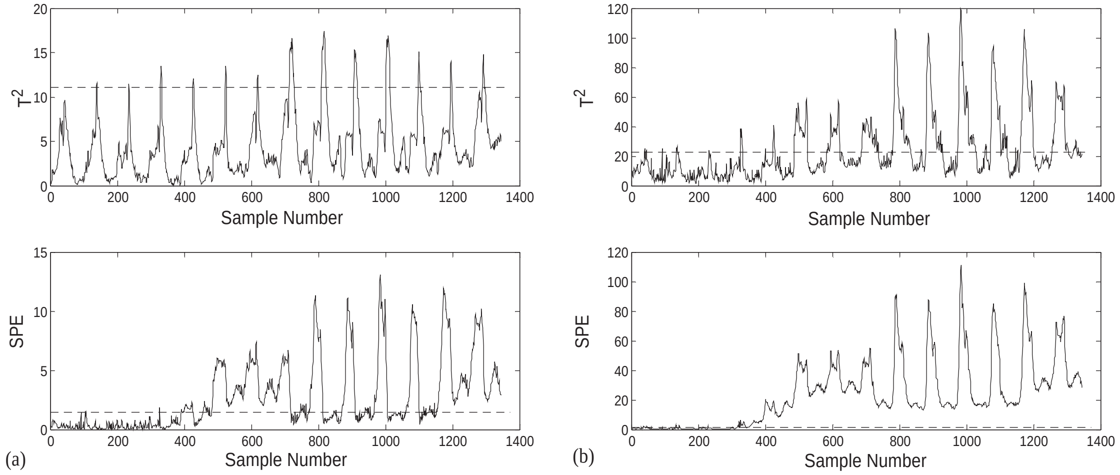

Fig. 9. (a) PCA monitoring charts, and (b) KPCA monitoring for the case of a linear decrease in the nitrification rate (benchmark example).

cross-validation. As shown in Fig. 8(a), the T^2 and SPE charts of the PCA method show periodic and nonstationary features originating from fluctuations in the influent load, which is characterized by strong diurnal changes in the flow rate and composition of the feed waste stream. However, it is clear from the T^2 and SPE charts that the PCA method with three PCs fails to detect the internal disturbance because the periodic and nonlinear features of the wastewater treatment plant dominate. In contrast to the PCA result, the SPE chart of the KPCA monitoring with nine PCs selected using the average eigenvalue approach (Fig. 8(b)) successfully detects the internal disturbance from sample 288 onwards, which represents almost 100% detection without a delay. In contrast, the T^2 chart detects the disturbance only at some samples.

Fig. 9(a) shows the monitoring results of linear PCA for the case of a linear decrease of μ_A. The T^2 chart of the PCA shows little evidence of the change, whereas the SPE chart shows a distinct change from about sample 390 (i.e., a delay of about 102 samples). Furthermore, even after sample 390, the SPE chart still falls below the 99% control limit at some samples even though the fault is still present. The KPCA monitoring charts for the same disturbance are shown in Fig. 9(b). The SPE chart of the KPCA detects the disturbance at about sample 318, approximately 72 samples earlier than linear PCA. Furthermore, after the disturbance is detected, the SPE values falls below the 99% control limit at very few samples. The T^2 chart of KPCA also indicates the presence of abnormalities, although it is less reliable than the SPE chart. Overall, the scores extracted from KPCA efficiently distinguish between the faulty and normal operating data.

The above two simulation examples demonstrate that KPCA can effectively capture the nonlinear relationship in process variables and that it gives better monitoring performance than PCA.

5. Conclusions

This paper proposes a new approach to process monitoring that uses KPCA to achieve multivariate statistical process control. KPCA can efficiently compute PCs in high-dimensional feature spaces by means of integral operators and nonlinear kernel functions. Compared to other nonlinear methods, KPCA has the following main advantages: (1) no nonlinear optimization is involved; (2) the calculations in KPCA are as simple as in standard PCA, and (3) the number of PCs need not be specified prior to modeling. In this paper, a simple calculation of the SPE in the feature space F is also suggested. The proposed monitoring method was applied to fault detection in both a simple multivariate process and the simulation benchmark of the biological WWTP. These examples demonstrated that the proposed approach can effectively capture nonlinear relationships in process variables and that, when used for process monitoring, it shows better performance than linear PCA.

The present work highlights the promise of the KPCA approach for process monitoring; however, KPCA has some problems that must be considered. The size of kernel matrix $\mathbf{K}$ becomes problematic when the number of samples becomes large. This can be solved by using a sparse approximation of the matrix $\mathbf{K}$, which still describes the leading eigenvectors sufficiently well (Smola and Schölkopf, 2000; Müller et al., 2001). The selection of the kernel function is crucial to the proposed method since the degree to which the nonlinear characteristic of a system is captured depends on this function; however, the general question of how to select the ideal kernel for a given monitoring process remains an open problem. In this paper, the radial basis kernel function ($k(\mathbf{x},\mathbf{y}) = \exp(-\|\mathbf{x}-\mathbf{y}\|^2/c)$) was considered. If this kernel function is to be used for process monitoring, the method for finding the optimal value of c should be clarified in future work. Furthermore, selection of the optimal number of PCs in the kernel space is also important. In comparison to linear PCA, nonlinear PCA based on neural networks has the disadvantage that it is difficult to compute the contributions of the original process variables because physically meaningful loading vectors cannot be found in the networks. KPCA also has the drawback that it is difficult to identify the potential source(s) of process faults in nonlinear situations because it is difficult or even impossible to find an inverse mapping function from the feature space to the original space. In the present work, we considered the performance of the KPCA monitoring method only from the viewpoint of fault detection. In future research, we will examine fault identification in KPCA monitoring, with one potential solution being the identification method proposed by Dunia et al. (1996).

Acknowledgements

This work was supported by a Grant No. (R01-2002-000-00007-0) from Korea Science & Engineering Foundation.and the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF).

References

Bakshi, B.R., 1998. Multiscale PCA with application to multivariate statistical process monitoring. A.I.Ch.E. Journal 44 (7), 1596–1610.

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

Christianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge university press, UK.

Dong, D., McAvoy, T.J., 1996. Nonlinear principal component analysis based on principal curves and neural networks. Computers and Chemical Engineering 20 (1), 65–78.

Dunia, R., Qin, S.J., Edgar, T.F., McAvoy, T.J., 1996. Identification of faulty sensors using principal component analysis. A.I.Ch.E. Journal 42 (10), 277–2812.

Haykin, S., 1999. Neural Networks. Prentice-Hall, Englewood Cliffs, NJ.

Hiden, H.G., Willis, M.J., Tham, M.T., Montague, G.A., 1999. Nonlinear principal components analysis using genetic programming. Computers and Chemical Engineering 23, 413–425.

Jia, F., Martin, E.B., Morris, A.J., 2001. Nonlinear principal components analysis with application to process fault detection. International Journal of Systems Science 31, 1473–1487.

Kramer, M.A., 1991. Nonlinear principal component analysis using autoassociateive neural networks. A.I.Ch.E. Journal 37 (2), 233–243.

Ku, W., Storer, R.H., Georgakis, C., 1995. Disturbance detection and isolation by dynamic principal component analysis. Chemometrics and Intelligent Laboratory Systems 30, 179–196.

Mika, S., Schölkopf, B., Smola, A.J., Müller, K.-R., Scholz, M., Rätsch, G., 1999. Kernel PCA and de-noising in feature spaces. Advances in Neural Information Processing Systems 11, 536–542.

Müller, K.-R., Mika, S., Rätsch, G., Tsuda, K., Schölkopf, B., 2001. An introduction to kernel-based learning algorithms. IEEE Transactions of Neural Networks 12 (2), 181–202.

Nomikos, P., MacGregor, J.F., 1995. Multivariate SPC charts for monitoring batch processes. Technometrics 37, 41–59.

Romdhani, S., Gong, S., Psarrou, A., 1999. A multi-view nonlinear active shape model using kernel PCA. Proceedings of BMVC, Nottingham, UK, pp. 483–492.

Schölkopf, B., Smola, A.J., Müller, K., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10 (5), 1299–1399.

Schölkopf, B., Mika, S., Burges, C.J.C., Knirsch, P., Müller, K.-R., Rätsch, G., Smola, A.J., 1999. Input space versus feature space in kernel-based methods. IEEE Transactions on Neural Networks 10 (5), 1000–1016.

Smola, A.J., Schölkopf, B., 2000. Sparse greedy matrix approximation for machine learning. Proceedings of ICML'00, San Francisco, pp. 911–918.

Spanjers, H., Vanrolleghem, P.A., Nguyen, K., Vanhooren, H., Patry, G.G., 1998. Towards a simulation-benchmark for evaluating respirometry-based control strategies. Water Science and Technology 37 (12), 219–226.

Sprang, E.N.M., Ramaker, H.-J., Westerhuis, J.A., Gurden, S.P., Smilde, A.K., 2002. Critical evaluation of approaches for on-line batch process monitoring. Chemical Engineering Science 57, 3979–3991.

Valle, S., Li, W., Qin, S.J., 1999. Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods. Industrial and Engineering Chemistry Research 38, 4389–4401.

Vanrolleghem, P.A., Gillot, S., 2001. Robustness and economic measures as control benchmark performance criteria. Water Science and Technology 45 (4–5), 117–126.

Wise, B.M., Gallagher, N.B., 1996. The process chemometrics approach to process monitoring and fault detection. Journal of Process Control 6 (6), 329–348.

Wold, S., 1978. Cross-validatory estimation of components in factor and principal components models. Technometrics 20, 397–405.

Yoo, C.K., Choi, S.W., Lee, I.-B., 2002. Dynamic monitoring method for multiscale fault detection and diagnosis in MSPC. Industrial and Engineering Chemistry Research 41, 4303–4317.