

Research Article

Fault Diagnosis of Complex Industrial Process Using KICA and Sparse SVM

Jie Xu,¹ Jin Zhao,¹ Baoping Ma,¹ and Shousong Hu²

¹ School of Electrical and Automation Engineering, Nanjing Normal University, Nanjing, Jiangsu 210042, China

² College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China

Correspondence should be addressed to Jie Xu; 63038@njnu.edu.cn

Received 11 January 2013; Accepted 11 March 2013

Academic Editor: Engang Tian

Copyright © 2013 Jie Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

New approaches are proposed for complex industrial process monitoring and fault diagnosis based on kernel independent component analysis (KICA) and sparse support vector machine (SVM). The KICA method is a two-phase algorithm: whitened kernel principal component analysis (KPCA). The data are firstly mapped into high-dimensional feature subspace. Then, the ICA algorithm seeks the projection directions in the KPCA whitened space. Performance monitoring is implemented through constructing the statistical index and control limit in the feature space. If the statistical indexes exceed the predefined control limit, a fault may have occurred. Then, the nonlinear score vectors are calculated and fed into the sparse SVM to identify the faults. The proposed method is applied to the simulation of Tennessee Eastman (TE) chemical process. The simulation results show that the proposed method can identify various types of faults accurately and rapidly.

1. Introduction

The demand for safe operation in the industry has propelled research into fault detection and diagnosis methods. Based on principal component analysis (PCA), independent component analysis (ICA), partial least squares (PLSs), and canonical correlation analysis (CVA), the statistical method can represent high-dimensional process data in a reduced dimension, which contains most of the variance of the original data. These methods have been used in various applications [1–4].

ICA is a statistical approach that has the potential ability for blind source separation (BSS) without the prior information about the mixtures under the source signals that are statistically independent [5]. However, ICA fails to separate the nonlinearly mixed source due to its intrinsic linearity. One new nonlinear ICA technique for tackling the nonlinear problem has been in development, which is called KICA [6]. KICA can efficiently compute ICs in high-dimensional feature spaces using the kernel matrix K . However, KICA algorithm is difficult to diagnose faults since the mapping function is unknown.

Sparse SVM is a learning strategy based on the Newton method and conjugate gradient method. It is proposed in this paper to diagnose faults, which is able to accelerate not only the model selection process but also the training speed.

This paper is organized as follows. In Section 2, the kernel ICA algorithm is introduced. In Section 3, sparse SVM is proposed. The experimental results of this methods proposed in this paper are also presented in Section 4. Finally, our work of this paper is summarized in the last section.

2. KICA Algorithm

Let $x_1, x_2, \dots, x_m \in R^m$ be the n training samples for kernel PCA learning. By the nonlinear mapping $\Phi : x \in R^m \rightarrow z \in R^n$, the measure inputs are extended into the hyperdimensional feature space. The mapping of x_i is simply noted as $\Phi(x_i)$. The sample covariance in the feature space is given by

$$S^F = \frac{1}{N} \sum_{i=1}^N \Phi(x_i) \Phi(x_i)^T. \quad (1)$$

Let $Q = [\Phi(x_1), \dots, \Phi(x_n)]$, then S^F can be expressed by $S^F = (1/N)QQ^T$. Let us form the Gram matrix $K = QQ^T$. The matrix is an $N \times N$ matrix, and its elements are as follows:

$$[K]_{ij} = K_{ij} = \langle \Phi(x_i) \cdot \Phi(x_j) \rangle = K(x_i, x_j). \quad (2)$$

We can center the Gram matrix K by

$$\begin{aligned} \tilde{K} &= K - E_N K - K E_N + E_N K E_N, \\ E_N &= \frac{1}{N} \begin{bmatrix} 1 & \dots & 1 \\ \vdots & & \vdots \\ 1 & \dots & 1 \end{bmatrix} \in R^{N \times N}. \end{aligned} \quad (3)$$

Normalized \tilde{K} using the following formula:

$$\hat{K} = \frac{\tilde{K}}{\text{trace}(\tilde{K})/N}. \quad (4)$$

Calculate the eigenvectors $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_d$ of K corresponding to d largest positive eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$. Then, the d largest positive eigenvalues of S^F are $\lambda_1/N, \lambda_2/N, \dots, \lambda_d/N$, and the associated orthonormal eigenvectors v_1, v_2, \dots, v_d can be expressed by

$$v_i = \frac{1}{\sqrt{\lambda_i}} Q \alpha_i \quad (i = 1, 2, \dots, d). \quad (5)$$

Denote $H = [\alpha_1, \alpha_2, \dots, \alpha_d]$, $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$ and its matrix form is $V = QH\Lambda^{-1/2}$.

The mapped data in feature space can be whitened by the following transformation:

$$y = G^T \Phi(x). \quad (6)$$

Specifically

$$\begin{aligned} y &= G^T \Phi(x) = \sqrt{N} \Lambda^{-1} H^T Q^T \Phi(x) \\ &= \sqrt{N} \Lambda^{-1} H^T Q^T [\Phi(x_1), \dots, \Phi(x_N)]^T \Phi(x) \\ &= \sqrt{N} \Lambda^{-1} H^T [\tilde{K}(x_1, x), \dots, \tilde{K}(x_N, x)]^T \\ &= \sqrt{N} \Lambda^{-1} H^T \tilde{K}, \end{aligned} \quad (7)$$

where $H = [\alpha_1, \alpha_2, \dots, \alpha_d]$, $\Lambda = [\lambda_1, \lambda_2, \dots, \lambda_d]$.

Centering and whitening are carried out in the KPCA feature space. We designate the data after preprocessing as X_{normal} , W , as well as S_{normal} are obtained from the fast ICA algorithm in the feature space. Data dimension can be reduced by selecting a few rows of W based on the assumption that the rows with the largest sum of squares coefficient have the largest effect on the variation of S . The detailed algorithm for fast ICA can be easily found in the literatures [7]. The selected a row of W constitutes a matrix W_d (dominant part of W), and the remaining rows of W constitute a matrix W_e .

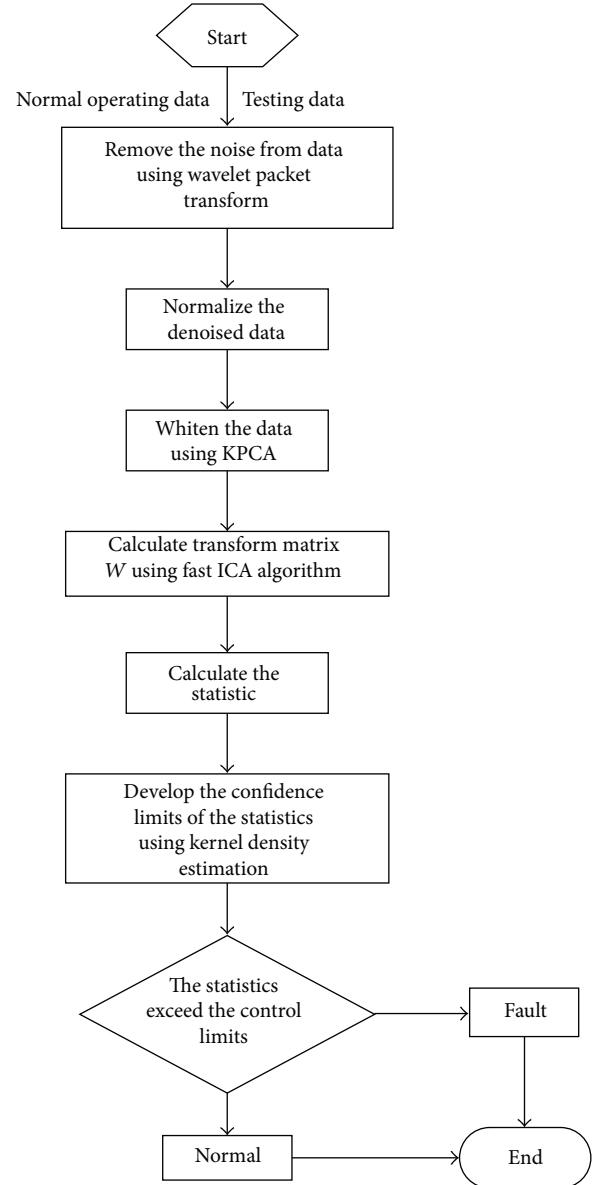


FIGURE 1: Fault detection algorithm flowchart.

For the new sampling data $X_{\text{new}}(k)$, independent vector that corresponds to the principal component space and residual space is $\hat{S}_{\text{new } d}(k) = W_d X_{\text{new}}(k)$, $\hat{S}_{\text{new } e}(k) = W_e X_{\text{new}}(k)$, respectively [8].

We defined three process monitoring statistics [8]:

$$\begin{aligned} U^2(k) &= \hat{S}_{\text{new } d}(k)^T \hat{S}_{\text{new } d}(k), \\ \text{SPE}(k) &= e(k)^T e(k) = (x(k) - \hat{x}(k))^T (x(k) - \hat{x}(k)), \\ U_e^2(k) &= \hat{S}_{\text{new } e}(k)^T \hat{S}_{\text{new } e}(k). \end{aligned} \quad (8)$$

And their confidence limits can be obtained by kernel density estimation. For more details about kernel density estimation, it can be found in the literature [9].

Specific implementation steps are shown in Figure 1.

3. Sparse SVM

Consider the second-order norm soft margin constraints; thus obtain the following optimization problem:

$$\begin{aligned} \min_{w, \xi} \quad & \langle w, w \rangle + C \sum_{i=1}^n \xi_i^2 \\ \text{s.t.} \quad & y_i (\langle w, \Phi(x_i) \rangle + b) \geq 1 - \xi_i, \quad \xi_i \geq 0. \end{aligned} \quad (9)$$

Consider (9) of the dual problem:

$$\begin{aligned} \max_{\alpha, \alpha^*} \quad & - \sum_{i,j=1}^n y_i y_j \alpha_i \alpha_j \left(k(x_i, x_j) + \frac{\delta_{i,j}}{C} \right) + 2 \sum_{i=1}^n \alpha_i \\ \text{s.t.} \quad & \sum_{i=1}^n y_i \alpha_i = 0, \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, n. \end{aligned} \quad (10)$$

The optimization problem can be written as an unconstrained one [10, 11]:

$$\min_w w^T w + C \sum_{i=1}^L L(y_i, w^T \Phi(x_i) + b). \quad (11)$$

Let us now consider nonlinear SVMs with a kernel function k and an associated the reproducing kernel Hilbert space H . The optimization problem (11) becomes

$$\min_{f \in H} \lambda \|f\|_H^2 + \sum_{i=1}^n L(y_i, f(x_i)). \quad (12)$$

We have made a change of variable by introducing the regularization parameter $\lambda = 1/C$. We have also dropped the offset b for the sake of simplicity.

Suppose now that the loss function L is differentiable with respect to its second argument. Using the reproducing property, (12) becomes

$$\min_{\beta} \lambda \beta^T K \beta + \sum_{i=1}^n L(y_i, K_i^T \beta). \quad (13)$$

For a given value of the vector β , we say that a point x_i is a support vector if $y_i f(x_i) < 1$ and if the loss on this point is nonzero. Let us reorder the training points such that the first n_{sv} points are support vectors. Finally, let I_{sv} be the $n \times n$ diagonal matrix with the first n_{sv} entries being 1 and the others 0. The gradient of (13) with respect to β is

$$\begin{aligned} \nabla &= 2\lambda K \beta + 2 \sum_{i=1}^{n_{sv}} K_i y_i (y_i K_i^T \beta - 1) \\ &= 2(\lambda \beta + K I^{sv} (K \beta - y)). \end{aligned} \quad (14)$$

And the Hessian is $H = 2(\lambda K + K I^{sv} K)$.

Each Newton step consists of the following update: $\beta \leftarrow \beta - \gamma H^{-1} \nabla$, where the step size is found by one-dimensional search method.

```
Function:  $\beta = \text{PrimalSVM}_{\text{ENM}}(K, y, \lambda)$ 
 $sv \leftarrow \{1, \dots, n\}; \beta_{\text{old}} = 0$ 
Repeat
 $\beta = 0;$ 
 $\beta_{sv} = (\lambda I_{n_{sv}} + K_{sv,sv})^{-1} y_{sv};$ 
 $step = \beta - \beta_{\text{old}};$ 
 $\gamma = 1;$ 
  While  $\Omega(\beta + \gamma \cdot step) > \Omega(\beta) + \mu \gamma \cdot (\nabla^T \cdot step)$ 
     $\gamma \leftarrow \rho \cdot \gamma;$ 
  End
   $\beta \leftarrow \beta + \gamma \cdot step;$ 
   $\beta_{\text{old}} \leftarrow \beta;$ 
   $sv \leftarrow i, i \text{ satisfy } y_i (K_i^T \beta) < 1;$ 
Until  $sv$  has not changed.
Where  $\mu \in (0, 0.5), \rho \in (0, 1)$ .
```

ALGORITHM 1: SVM based on the modified Newton optimization.

Combining (14) and H as $\nabla = H\beta - 2KI^{sv}y$, we find that after the update:

$$\beta = (\lambda K + KI^{sv}K)^{-1} \cdot KI^{sv}y = (\lambda I_n + I^{sv}K)^{-1} I^{sv}y. \quad (15)$$

Using the fact that the lower left block $\lambda I_n + I^{sv}K$ is 0, the update (15) turns out to be $\beta_{sv} = (\lambda I_{n_{sv}} + K_{sv,sv})^{-1} y_{sv}$.

Therefore, the algorithm of SVM based on the modified Newton optimization is performed as shown in Algorithm 1.

We now consider the optimization parameter is $[b, \beta^T]$. The augmented Hessian is

$$H = S \begin{bmatrix} 1^T I^{sv} 1 & 1^T K I^{sv} \\ K I^{sv} 1 & \lambda K + K I^{sv} K \end{bmatrix}, \quad (16)$$

where 1 should be understood as a vector of all 1.

We still use the former method of choice to choose the right ‘‘basis functions’’ and realize the purpose of building sparse SVM. Assume that the choice of ‘‘basis functions’’ collection is J ; then the selection of the next ‘‘basis function’’ j is equivalent to ensuring that β_j is constant and optimizing the following questions:

$$\min \lambda \begin{bmatrix} \beta_j^T & \beta_j \end{bmatrix} \begin{bmatrix} K_{JJ} & K_{Jj} \\ K_{jJ} & K_{jj} \end{bmatrix} \begin{bmatrix} \beta_j \\ \beta_j \end{bmatrix} + \sum_{i \in sv} (1 - y_i (K_{ij} \beta_j + K_{ij} \beta_j))^2. \quad (17)$$

Through the introduction of the Cholesky decomposition and the corresponding rank 1 update strategy, based on unconstrained optimization SpSVM with the following steps to achieve the following:

- (1) update matrix $\lambda I_{n_{sv}} + K_{sv}$ using the Cholesky decomposition;
- (2) optimize β_j using Newton’s method (Algorithm 1). If sv changes, update matrix $\lambda I_{n_{sv}} + K_{sv}$ using rank 1 update strategy;
- (3) if the ‘‘basis function’’ collection J is equal to the size of preset value, end of the program, otherwise, turn to step 4;

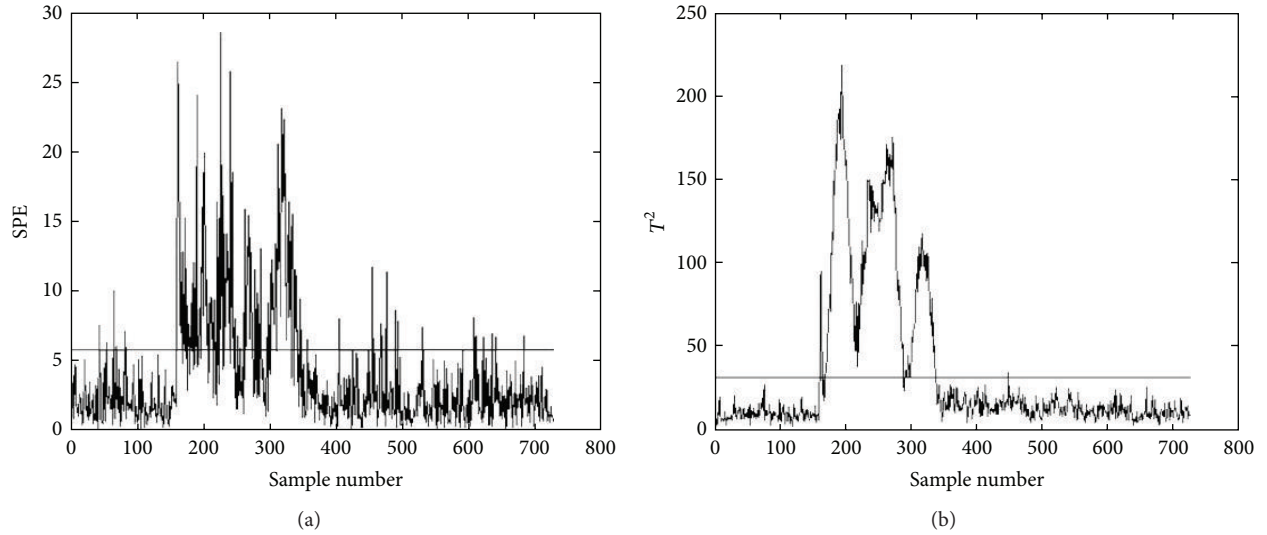


FIGURE 2: KPCA monitoring for fault 5.

TABLE 1: Results of fault identification.

ν	Diagnostic precision			n_{sv}		
	SVM _{IVR}	MKL _{DT}	SpSVM _{IVR}	SVM _{IVR}	MKL _{DT}	SpSVM _{IVR}
0	100%	100%	100%	97	40	19
0.02	99.2%	98.1%	98.9%	130	48	29
0.05	98.8%	96.2%	98.2%	165	61	46

- (4) select the next “basis function” $j : J \leftarrow J + j$, transferred to step 1.

4. Experimental Results

The proposed method is applied to the simulation of the Tennessee Eastman chemical process to achieve fault detection and fault diagnosis. SVM training involved in the QP and the QCQP problem can be by optimizing the package MOSEK method based on the interior point, and the SVM training data are preprocessing.

The process has 12 manipulated variables, 22 continuous process measurements, and 19 composition measurements sampled less frequently. The process contains the Gaussian noise. TE process includes 20 predefined failure modes, representing a step, random changes, slow drift, and other fault type, its process flow diagram and detailed process, see the text [12]. We use the control strategy in paper [13] to conduct closed-loop control. Simulation data for a 960×52 matrix are obtained as follows: 52 represent the number of process variables, and 960 represent the sampling points of each variable. The simulation is running 48 hours. The failure was in the introduction at the first 8 hours.

In order to verify the effectiveness of the KICA monitoring algorithms, KPCA and KICA were applied to detect fault 5. The monitoring charts of KPCA for fault 5 are shown in Figure 2. The figures indicate that T^2 and SPE statistics can also detect the happening of faults, but there will be false positives and false negatives in some sampling points. The

monitoring charts of KICA for fault 5 are shown in Figure 3. In contrast to KPCA, it can be seen that the U^2 , U_e^2 , and SPE statistics all respond strongly to the faults.

If the statistics exceed the control limits, it indicates a failure occurrence. We select 52 variables for the condition attributes and select 15 groups for the decision attribute, respectively, on behalf of 15 different types of failures. The training set is composed of 600 data, 40 samples data of each category. Test set is composed of 1200 data, 80 samples data of each category. We use 1-a.-r-mix strategy to achieve the multiclass classification SVM. The factor C is then set to 1000. The size of the alternative set κ in SpSVM is 20. Specific diagnostic results are shown in Table 1.

5. Conclusions

In this paper, KICA and sparse SVM are proposed for the complex industrial process monitoring. KICA can efficiently compute ICs in high-dimensional feature spaces by means of integral operations. The method using sparse SVM can reduce the number of support vectors without reducing the classification accuracy. The application results show the effectiveness of proposed method.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. 60234010, the

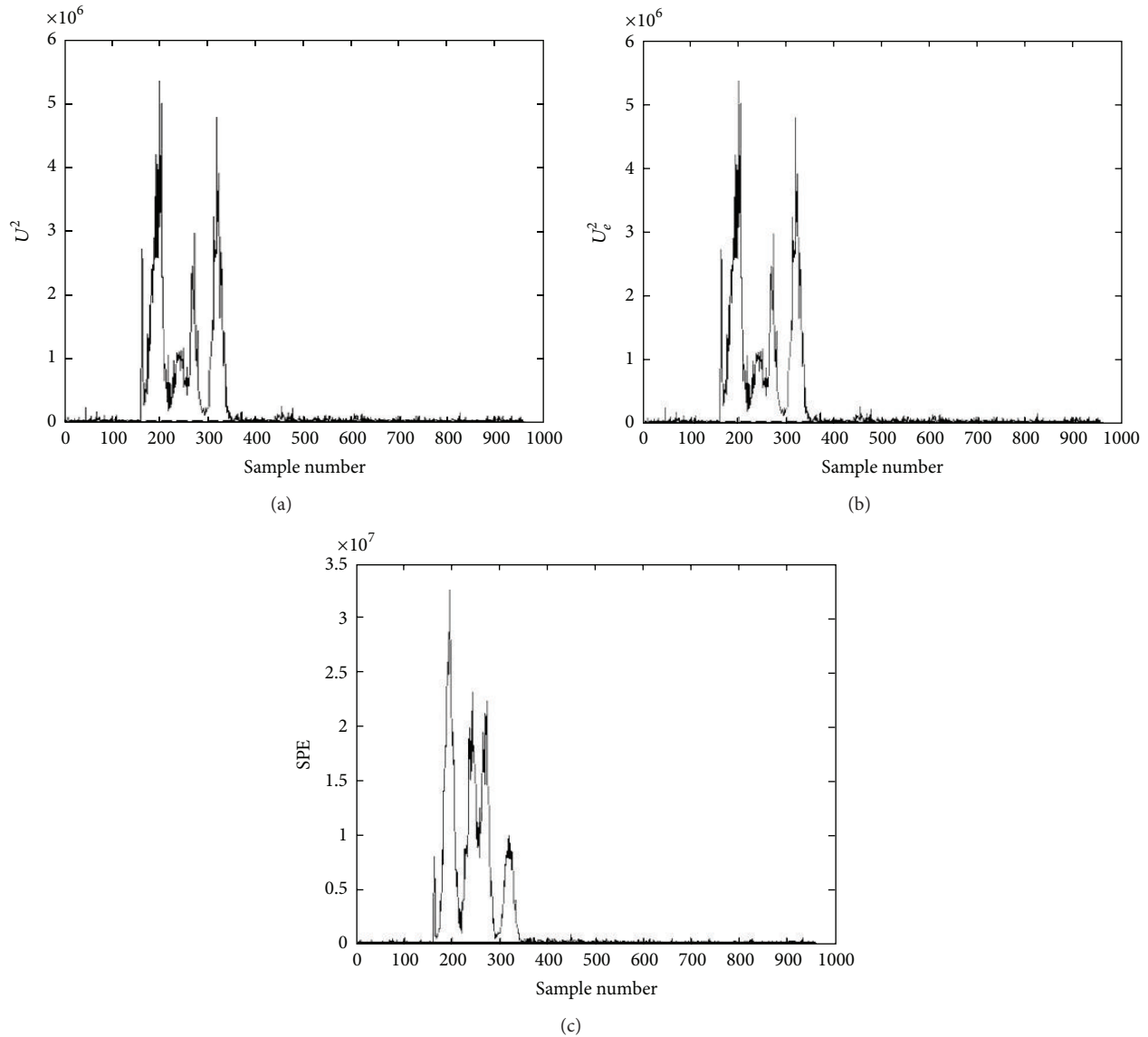


FIGURE 3: KICA monitoring for fault 5.

National Natural Science Youth Foundation under Grant no. 61203092, and Jiangsu Province Higher Education Natural Science Basic Research surface project under Grant no. 11KJB510007.

References

- [1] J. E. Jackson and G. S. Mudholkar, "Control Procedures for residuals associated with principal component analysis," *Technometrics*, vol. 21, no. 3, pp. 341–349, 1979.
- [2] M. Kano, S. Tanaka, S. Hasebe, I. Hashimoto, and H. Ohno, "Monitoring independent components for fault detection," *AIChE Journal*, vol. 49, no. 4, pp. 969–976, 2003.
- [3] S. J. Zhao, J. Zhang, and Y. M. Xu, "Performance monitoring of processes with multiple operating modes through multiple PLS models," *Journal of Process Control*, vol. 16, no. 7, pp. 763–772, 2006.
- [4] N. Mehranbod, M. Soroush, and C. Panjapornpon, "A method of sensor fault detection and identification," *Journal of Process Control*, vol. 15, no. 3, pp. 321–339, 2005.
- [5] P. Comon, "Independent component analysis, A new concept?" *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.
- [6] J. Yang, X. Gao, D. Zhang, and J. Y. Yang, "Kernel ICA: an alternative formulation and its application to face recognition," *Pattern Recognition*, vol. 38, no. 10, pp. 1784–1787, 2005.
- [7] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.
- [8] J. M. Lee, C. Yoo, and I. B. Lee, "Statistical process monitoring with independent component analysis," *Journal of Process Control*, vol. 14, no. 5, pp. 467–485, 2004.
- [9] M. P. Wand and M. C. Jones, *Kernel Smoothing*, Chapman and Hall, London, UK, 1995.
- [10] S. S. Keerthi, O. Chapelle, and D. DeCoste, "Building support vector machines with reduced classifier complexity," *Journal of Machine Learning Research*, vol. 7, pp. 1493–1515, 2006.

- [11] O. Chapelle, "Training a support vector machine in the primal," *Neural Computation*, vol. 19, no. 5, pp. 1155–1178, 2007.
- [12] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*, Springer, London, UK, 2001.
- [13] J. H. Chen and C. M. Liao, "Dynamic process fault monitoring based on neural network and PCA," *Journal of Process Control*, vol. 12, no. 2, pp. 277–289, 2002.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

