

Process Systems Engineering and Process Safety

High dimension feature extraction based visualized SOM fault diagnosis method and its application in *p*-xylene oxidation process☆

Ying Tian, Wenli Du*, Feng Qian

Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China

ARTICLE INFO

Article history:

Received 7 September 2014

Received in revised form 19 January 2015

Accepted 26 January 2015

Available online 27 March 2015

Keywords:

Self-organizing map

Local tangent space alignment

Fault diagnosis

Visualization

p-xylene oxidation

ABSTRACT

Purified terephthalic acid (PTA) is an important chemical raw material. *p*-xylene (PX) is transformed to terephthalic acid (TA) through oxidation process and TA is refined to produce PTA. The PX oxidation reaction is a complex process involving three-phase reaction of gas, liquid and solid. To monitor the process and to improve the product quality, as well as to visualize the fault type clearly, a fault diagnosis method based on self-organizing map (SOM) and high dimensional feature extraction method, local tangent space alignment (LTSA), is proposed. In this method, LTSA can reduce the dimension and keep the topology information simultaneously, and SOM distinguishes various states on the output map. Monitoring results of PX oxidation reaction process indicate that the LTSA-SOM can well detect and visualize the fault type.

© 2015 The Chemical Industry and Engineering Society of China, and Chemical Industry Press. All rights reserved.

1. Introduction

Purified terephthalic acid (PTA) is used to produce polyester, including polyester fiber, polyethylene terephthalate bottle resin, and polyester film. Because of the huge demand for polyester, production of PTA increases dramatically, in which the oxidation of *p*-xylene (PX) is a critical step [1]. Since a slight fault may cause a huge loss because of multiple connections among the components, it is important to find an effective method to detect the fault timely and monitor the PX oxidation reaction process.

Over the past decades, many methods have been used to monitor the process, among which the most popular approach is multivariate statistical process monitoring (MSPM). Meanwhile, visualization methods have been developed and their superiority is to transfer data from a numeric form to a graphic form, which facilitates people understanding. Currently, how to use visualization methods as well as MSPM to identify fault information is a research focus.

To visualize the high-dimensional data, the self-organizing map (SOM) based methodology has been developed by Kohonen [2]. As an unsupervised neural network, SOM preserves the original topological

structure of data effectively, projects high-dimensional data to a two-dimensional grid, and achieves intuitive visualization. Some SOM based methods have been applied ever since, such as in improving geometric quality parameters of hot rolled strips [3], classifying wall temperatures of a blast furnace [4], visualizing flotation process disturbances [5], monitoring a continuous pulp digester [6], classifying multistate operations [7,8], and detecting industrial process data [9].

Although SOM can reduce the dimensionality of data and preserve the topology, it is sometimes difficult to distinguish fault clusters, especially in complex chemical processes such as PX oxidation with nonlinear and relevant variables. Therefore, before SOM training, a certain feature extraction method should be adopted to distinguish fault clusters.

The key for feature extraction and dimension reduction is to preserve the underlying local geometrical information of raw high-dimensional data and reduce insignificant dimensions. For nonlinear original data, traditional feature extraction methods such as principle component analysis (PCA) [10] cannot well preserve the geometrical information when conducting dimension reduction, since PCA may map close points in high-dimension space into distant points in low-dimension space. To overcome this difficulty, lots of methods such as multidimensional scaling [11], auto-encoder neural networks [12], locally linear embedding [13], and mixtures of linear model [14] have been proposed. Recently, artificial immune system based methods have been proposed for chemical process fault diagnosis [15,16]. Moreover, some dimension reduction methods such as correlative component analysis (CCA) [17] and Fisher discriminant analysis [18] have been integrated with SOM for fault diagnosis. However, these methods still have some problems, such as huge calculating time and lots of

☆ Supported by the Major State Basic Research Development Program of China (2012CB720500), the National Natural Science Foundation of China (61333010 21276078), the National Science Fund for Outstanding Young Scholars (61222303), the Fundamental Research Funds for the Central Universities, Shanghai Rising-Star Program (13QH1401200), the Program for New Century Excellent Talents in University (NCET-10-0885) and Shanghai R&D Platform Construction Program (13DZ2295300).

* Corresponding author.

E-mail address: wldu@ecust.edu.cn (W. Du).

parameters to be decided. Zhang and Zha have proposed a neighborhood based dimension reduction method, local tangent space alignment (LTSA) [19], in which the tangent spaces are learned by fitting an affine subspace in the neighborhood of each data point and the local geometry of underlying manifold is represented by these tangent spaces. Compare to some other manifold learning methods, LTSA has less parameters and less calculating load. Therefore, LTSA can effectively keep the topology information and reduce the calculating time while reducing the dimension.

In this paper, we intend to visualize the fault diagnosis of PX oxidation process by combining SOM and LTSA. Based on the original data, LTSA extracts low dimensional submanifold. With the extracted feature, SOM can distinguish clusters more clearly. Therefore, a high accuracy of fault diagnosis is achieved in process monitoring. The implementation of LTSA–SOM based fault diagnosis for the PX oxidation reaction process is presented.

2. PX Oxidation Reaction Process

2.1. Description of process

The PX oxidation reaction process is used for production of PTA, including one reactor, four condensers, and one reflux drum. The lower part of the reactor is a bubbling reactor, with air bubbles from its bottom and stirring slowly by a scraper at the bottom. The upper part of the reactor is for dehydration. At high pressure and temperature, PX, acetic acid (solvent), cobalt acetate, manganese acetate (catalyst), tetrabromoethane (accelerator), and compressed air are sent into the oxidation reactor to produce TA, which is in a state of slurry and is discharged continuously into a slurry tank. Then crystallization, separation and drying are adopted to obtain TA powder. The reaction heat is removed from the top of the reactor by evaporating acetic acid solvent and water. The steam and gas (acetic acid, water, gas) excluded from the top of the reactor are cooled in four condensers and separated in the reflux drum. The condensed acetic acid is sent back to the system by reflux drum and reflux pump, and the remaining is used for removing some by-products in the reaction. The oxidation process is followed by sequential purification units for production of PTA. In this study, only the PX oxidation process is taken into consideration. The simplified flow chart of the PTA oxidation process is shown in Fig. 1 [20–22].

2.2. Monitoring variables and fault types in the process

The mechanism of PX oxidation reaction is complex, with long process, complicated operating units, and related variables. In the process, many variables will influence the yield and quality of TA. Based on investigations, we decide the variables with significant influence on the final production as measurement variables (Table 1) and the fault events of PTA equipment (Table 2). For fault diagnosis and monitoring, LTSA–SOM is proposed for the PX oxidation process.

Table 1
Process measurement variables (1 bar = 10^5 Pa)

Variables	Measurement	Standard value
1	PX feed	24575 kg·h ⁻¹
2	FC1101 temperature	25 °C
3	HAC feed	101670 kg·h ⁻¹
4	H ₂ O feed	10223 kg·h ⁻¹
5	CO feed	56 kg·h ⁻¹
6	MN feed	30 kg·h ⁻¹
7	BR feed	100 kg·h ⁻¹
8	FC1102 temperature	25 °C
9	Air feed	114324 kg·h ⁻¹
10	FC1109 temperature	45 °C
11	TA product	36462 kg·h ⁻¹
12	Concentration of 4CBA in Product	124.5 kg·h ⁻¹
13	HAC in product	89409 kg·h ⁻¹
14	Total volume flow of outflow	155.2 m ³ ·h ⁻¹
15	Concentration of CO ₂ in off gas	1.4%
16	Concentration of CO(V) in off gas	0.45%
17	Concentration of O ₂ in off gas	2.9%
18	Reactor temperature	186.28 °C
19	Reactor pressure	11.1458 bar
20	Reactor level	10.96 m
21	Crystallizer temperature	40 °C
22	Crystallizer pressure	10.9431 bar
23	Crystallizer level	1.25 m

3. LTSA and SOM

3.1. LTSA algorithm

In feature extraction and dimension reduction, it is important to preserve the local geometrical information of original data while reducing

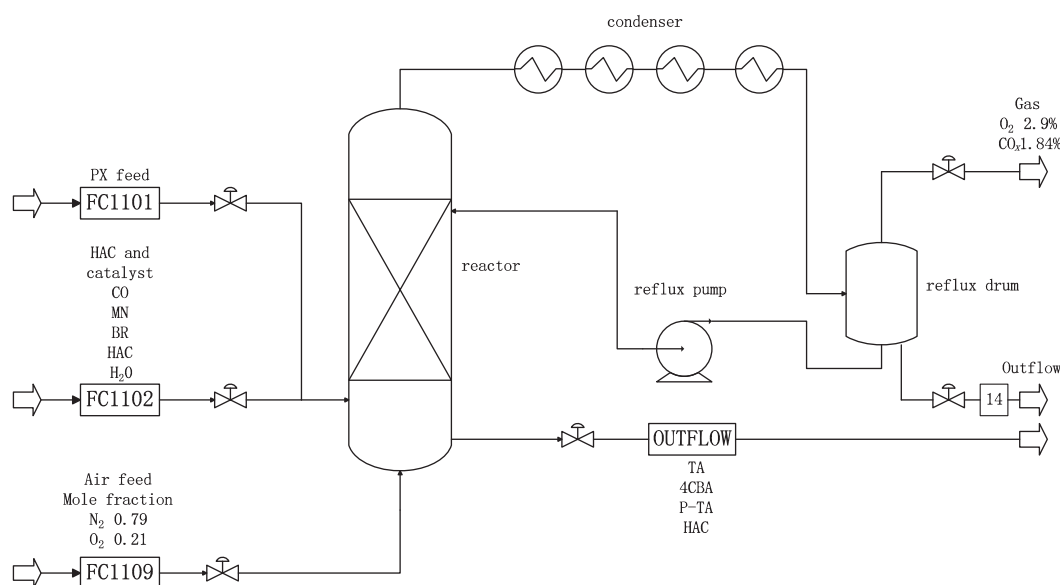


Fig. 1. Simplified flow-chart of PX oxidation process.

Table 2
Process faults

Fault type	Description	Fault value
1	Step change of PX feed	17202.5 kg·h ⁻¹
2	Step change of HAC feed	71169 kg·h ⁻¹
3	Step change of H ₂ O feed	7156.1 kg·h ⁻¹
4	Step change of air feed	108607.8 kg·h ⁻¹
5	Step change of PX feed temperature	35 °C
6	Step change of air feed temperature	55 °C
7	Step change of FC1102 temperature	17.5 °C
8	Sticking of B1 valve	25%
9	Sticking of condenser valve	25%

dimensions. However, the data in a complex chemical industry are usually nonlinear and related. Traditional linear feature extraction methods such as PCA will fail to well preserve its geometrical information when unfolding the nonlinear manifold. Unlike PCA, LTSA is a neighborhood based feature extraction and dimension reduction method. LTSA analyzes the nonlinear structure through original data points and relationships between original data points and their neighbors, so it reserves the local geometry in low dimensional space. LTSA takes a collection of N M -dimensional points $\mathbf{X}_{\text{origin}}$ as the input of the algorithm, then constructs k local nearest neighbors for each input point and produces N d -dimensional embedding. The details are as follows.

- (1) For each point $\mathbf{x}_i$ ($i = 1, 2, \dots, N$), find its k local nearest neighbors $\mathbf{x}_{ij}$ ($j = 1, 2, \dots, k$). Set $\mathbf{X}_i = (\mathbf{x}_{i1}, \mathbf{x}_{i2}, \dots, \mathbf{x}_{ik})$.
- (2) For each point $\mathbf{x}_i$ ($i = 1, 2, \dots, N$), extract local information by computing the d largest eigenvectors $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_d$ of the correlation matrix $(\mathbf{X}_i - \bar{\mathbf{x}}_i \mathbf{e}^T)(\mathbf{X}_i - \bar{\mathbf{x}}_i \mathbf{e}^T)^T$ and set

$$\mathbf{G}_i = [\mathbf{e}/\sqrt{k}, \mathbf{g}_1, \dots, \mathbf{g}_d] \quad (1)$$

where $\mathbf{e}$ is the k -dimensional vector that all elements equal one, $\bar{\mathbf{x}}_i = \frac{1}{k} \sum_j \mathbf{x}_{ij}$, $\bar{\mathbf{x}}_i \in R^{M \times 1}$.

- (3) Denote the set of indices for the k nearest neighbors of $\mathbf{x}_i$ as $I_i = \{i_1, i_2, \dots, i_k\}$. Construct alignment matrix $\mathbf{B}$ by local summation,

$$\mathbf{B}(I_i, I_i) \leftarrow \mathbf{B}(I_i, I_i) + \mathbf{I} - \mathbf{G}_i \mathbf{G}_i^T, i = 1, \dots, N \quad (2)$$

where initial $\mathbf{B} = 0$ and $\mathbf{I} \in R^{k \times k}$ is the unit matrix. After the iteration for all $\mathbf{x}_i$, we obtain $\mathbf{B}$.

- (4) Compute the $d + 1$ smallest eigenvectors of $\mathbf{B}$, pick up the eigenvector matrix $[\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{d+1}]$ corresponding to the 2nd to $d + 1$ th smallest eigenvalues, and set the global coordinates $\mathbf{T} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{d+1}]$.

In this algorithm, we cannot obtain an analytic form of the mapping function. Therefore, for each new sample, relearning is needed to get its low dimensional embedding. In order to solve this problem, the mapping base vector of LTSA is obtained through Eq. (3).

$$\mathbf{MAP} = \mathbf{X}_{\text{origin}}^{-1} \times \mathbf{T} \quad (3)$$

With the base vectors of the projection space obtained, the new sample point is mapped to a low dimensional linear space through the base vector directly.

$$\mathbf{X}_{\text{reduce}} = \mathbf{X}_{\text{new}} \cdot \mathbf{MAP} \quad (4)$$

To illustrate the advance of LTSA, we test the algorithm on the Iris dataset and compare the effectiveness and computation time of LTSA and CCA. The dataset contains 150 samples, belonging to three parts of a plant. Fig. 2 shows that the three classes are separated effectively with both methods. Table 3 shows that the average computation time of CCA is significantly longer than that of LTSA. Therefore, LTSA presents better performance and is adopted here.

Table 3

Computation time for processing Iris dataset

	CCA	LTSA
Average computer time /s	2.21	1.37

3.2. Self-organizing map

As an unsupervised neural network, SOM can project high-dimensional data into a two-dimensional grid, so it has the ability for data visualization. Moreover, with SOM mapping, close input data points still locate in the nearby neurons on the output map, so SOM can be used as clustering tool.

For a dataset $\mathbf{Y} \in R^{N \times M}$, we have $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{iM}]$ ($i = 1, 2, \dots, N$), where N is the total number of input samples and M is the dimension of samples. SOM takes the dataset of $\mathbf{Y}$ as input and takes an ordered collection of neurons as output. Each neuron j in output layer has a reference

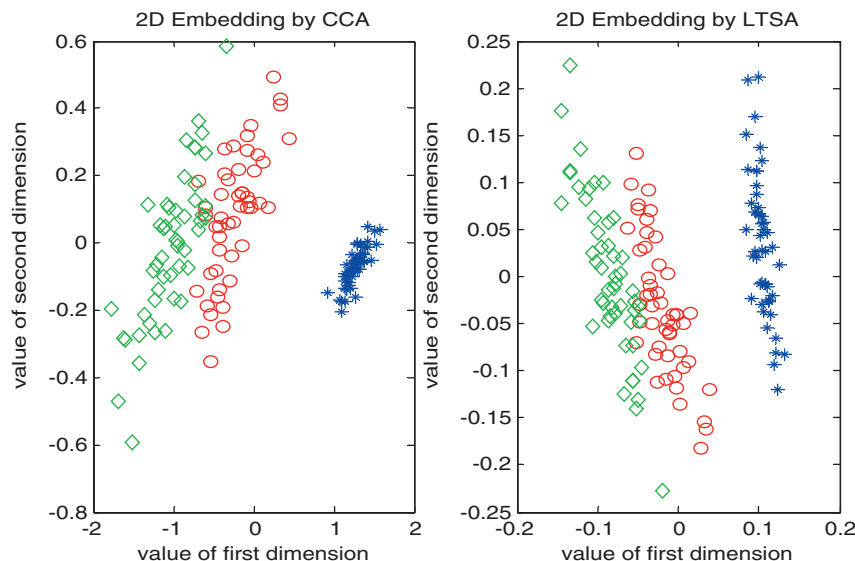


Fig. 2. Iris dataset processed by CCA and LTSA. Symbols: different samples.

vector $\omega_j = [\omega_{j1}, \omega_{j2}, \dots, \omega_{jM}]$ ($j = 1, 2, \dots, J$) to connect the input layer, where J is the total number of neurons.

These neurons need training to represent and visualize $\mathbf{Y}$. Initially, let each ω_j be assigned a random vector. When a sample $\mathbf{y}_i \in \mathbf{Y}$ is obtained, calculate the distance between the sample and the reference vector. The neuron whose reference vector has the smallest difference from $\mathbf{y}_i$ is defined as the winner neuron and is called as the best matching unit.

$$b_i = \arg \min \|\mathbf{y}_i - \omega_j(t)\| \quad \text{for } j = 1, 2, \dots, J \quad (5)$$

where the Euclidean distance is usually used as the criterion.

After the selection of winning neuron b_i , the reference vector of the winning neuron and the reference vectors of neighboring neurons are updated. The updated reference vectors of the neurons in the SOM are changed to make them more similar to the input vector. The update function for reference vectors is

$$\omega_j(t+1) = \omega_j(t) + \alpha(t)h_{bj}(t)\|\mathbf{y}_i - \omega_j(t)\| \quad (6)$$

where $\alpha(t)$ is the learning rate parameter and $h_{bj}(t)$ is the neighborhood function.

Let $\mathbf{r}_{bi}$ be the position of the winning neuron on the output map, where $\mathbf{r}_{bi} \in R^2$. All neurons whose distance from the b_i th neuron less than a certain value can be considered as its neighbor neurons. A popular choice of neighborhood function is a Gaussian function

$$h_{bj}(t) = \exp\left(-\frac{\|\mathbf{r}_{bi} - \mathbf{r}_j\|^2}{2\sigma^2(t)}\right) \quad (7)$$

where $\sigma(t)$ is the neighborhood width of the neurons.

For convergence, $\sigma(t)$ and $\alpha(t)$ are given a large value and then decrease gradually with t . As $t \rightarrow \infty$, $\alpha(t) \rightarrow 0$ and $\sigma(t)$ approaches a small value.

The training ends when the number of iterations reaches certain value or no noticeable changes in the map. The neurons on the output map are labeled with their corresponding input names.

4. Case Study: Diagnosis of PX Oxidation Reaction

Based on the general kinetic model of PX oxidation reaction, HAC combustion model, and PX combustion model, the entire PX oxidation process is built in Aspen plus by Qian *et al.* [22]. With the model running directly, the normal data can be collected. In order to obtain the fault data, step changes are added to related variables, then the process is run and corresponding fault data are gathered. For instance, for collecting fault 1 data, the feed of PX is changed from $24575 \text{ kg} \cdot \text{h}^{-1}$ to $17202.5 \text{ kg} \cdot \text{h}^{-1}$ and the whole process is run. Since the process variables are related with each other, lots of variables are affected. The details of LTSA-SOM based fault diagnosis for PX oxidation are as follows.

4.1. In training stage

- (1) Obtain 48 normal observations from the model of PX oxidation process in Aspen plus and normalize the data. According to Section 3.1, obtain base vectors of LTSA projection space.
- (2) Select 48 fault observations from fault 1 to fault 9 separately and 48 normal observations to construct original training dataset containing normal and fault data (totally 480 samples). Normalize the data using the mean and variance of normal stage and define the normalized dataset as $\mathbf{X}$.
- (3) Reduce the dimension of $\mathbf{X}$ using the base vector in the mapping space calculated in step (1) and define the treated data as $\mathbf{Y}$.
- (4) Choose a vector of $\mathbf{Y}$ as the input vector of SOM, set initial parameters, calculate distances between the input vector and reference vectors of all neurons, and find the winning neurons. Update the

reference vectors of the neurons within the neighborhood of the winning neuron. Choose another input vector until all samples finish. Update the learning rate and neighborhood function.

Repeat the process until no noticeable change occurs in the map.

- (5) Label the names of fault categories to the output neurons.

4.2. In testing stage

- (1) 40 observations are gathered separately from normal process as well as fault 1 to fault 9 to construct test data. Normalize the data and analyze the data by the base vectors, and name the treated data as $\mathbf{Y}_{\text{test}}$.
- (2) Send $\mathbf{Y}_{\text{test}}$ into the trained SOM network and seek the best matching unit according to the shortest distance between the test data vectors and reference vectors.

To evaluate the performance of LTSA-SOM in process monitoring and fault diagnosis, it is compared with SOM and PCA-SOM. Fig. 3 is the mapping result for simple SOM. The training data are divided into several clusters in output map (a). In the U-matrix (b), areas with homogeneous blue in SOM map are clustered within small distances, while the light color is the borders. With the change of color, the U-matrix can tell clusters clearly. However, the data of fault 3 are divided into two separated clusters, caused by the step change of H_2O feed. When the fault occurs, the change rate for the measurement of H_2O feed reaches 30%. Three neurons of fault 3 are far from others, and the cluster of fault 7 lies between the three neurons and other neurons of fault 3, which destroys the blocking of SOM. When new test data are projected into the region around faults 3 and 7, it is difficult to identify the exact cluster for the new test data.

Considering the disadvantage of simple SOM, we hope to process input data beforehand to improve the performance. Here, we first use PCA-SOM. PCA is the most popular method to extract feature and reduce dimension. In this study, training data are same as those used in SOM training. The number of principal components is determined by the ratio of variance contribution and it is set as 85% in this paper. The dimension of the training data reduces from 22 to 13. The training result is shown in Fig. 4. Fig. 4(a) shows that the training data are not clustered very well. For example, faults 6 and 7 are very close to each other. The U-matrix (b) shows that there is no border between these two faults. Thus the performance of PCA-SOM is improved little because PCA only extracts variance information from training data and cannot reserve the topological information of the data.

Finally, LTSA-SOM is applied to process the original data. With LTSA feature extraction, original 22-dimensional data are reduced into 11-dimensional data. Then the new input data are sent into the map for training. The result is shown in Fig. 5. Only one neuron of fault 7 mixes with one neuron of fault 6, and all other data are separated in right clusters. It proves that LTSA plays an important role in classification.

When the SOM output is obtained with training data, it can be used as a reference model for test data. Test data will find the best matching unit automatically.

Fig. 6 is the trajectory for the whole process of fault 6 that contains both normal and fault data. Fault 6 is caused by the step change of air feed temperature. It will influence the measurement value of FC1109 temperature (variable 10) as well as nearly all concentrations of substances in product and in off gas. In Fig. 6, we can see the process clearly. The operation is in normal area at first (a), and then leaves the normal region and extends to the area of fault 6 (b)–(c). Therefore, the trajectory can be used to monitor real-time fault processes. It is to be noticed that with the trained SOM for test data, we do not need to project the test data into the exact neurons that have the training data but just map the test data into the related region, as shown in the map result. When the test data are projected into the right region, the test data are identified appropriately. In this example, the test data are mapped

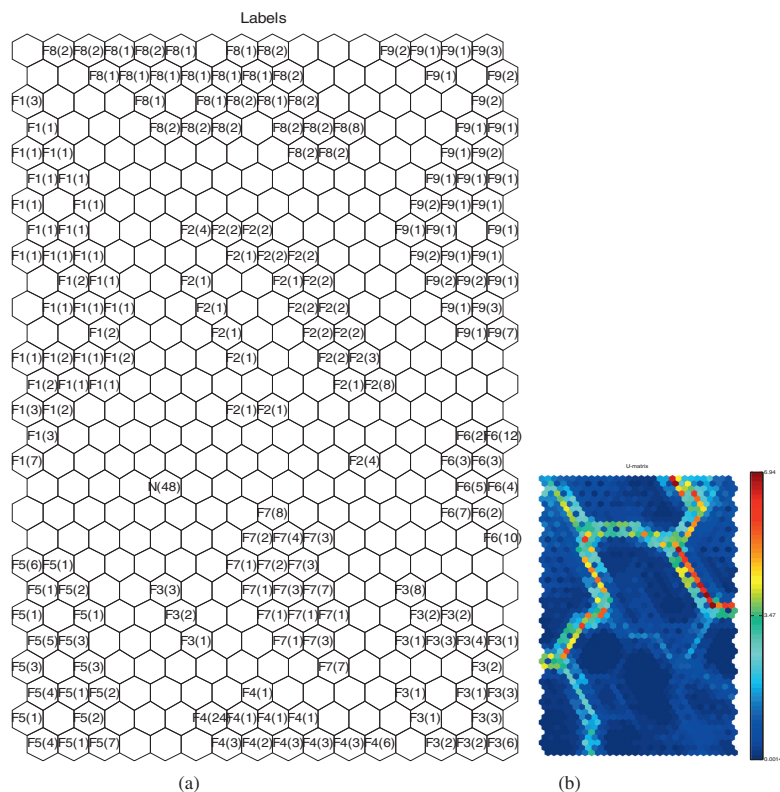
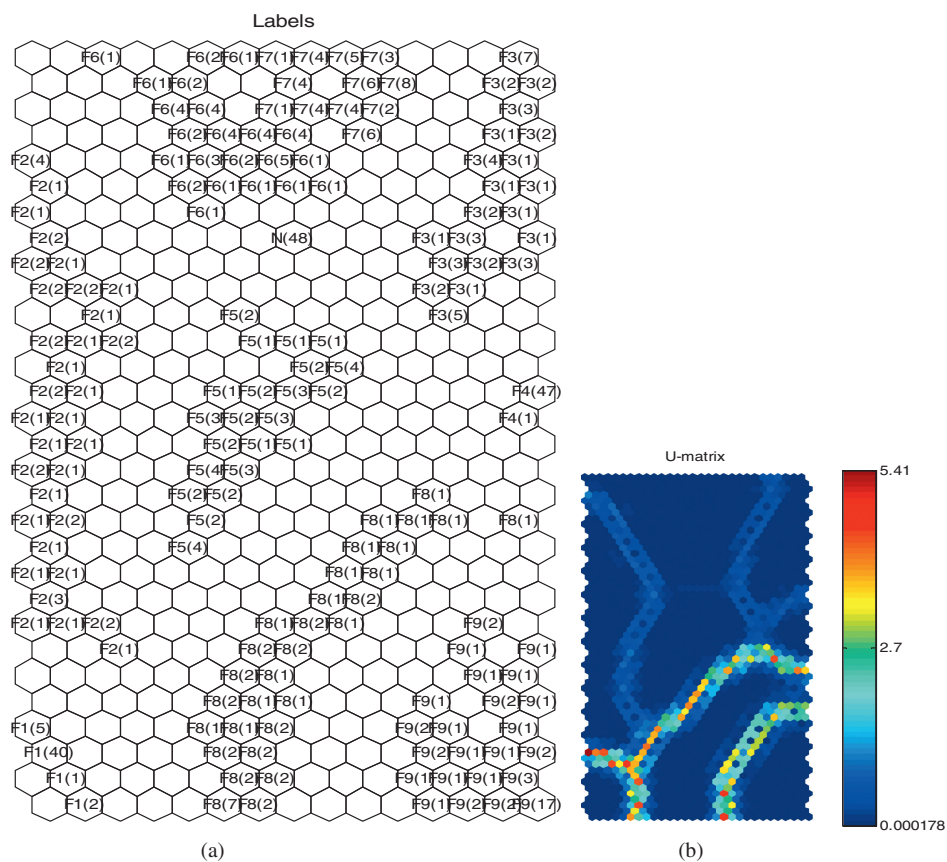


Fig. 3. Mapping result (a) and U-matrix (b) of training data for normal process and faults 1–9 based on SOM. N: normal status; F1–F9: faults 1–9; number in bracket: number of observations mapped into certain neuron.



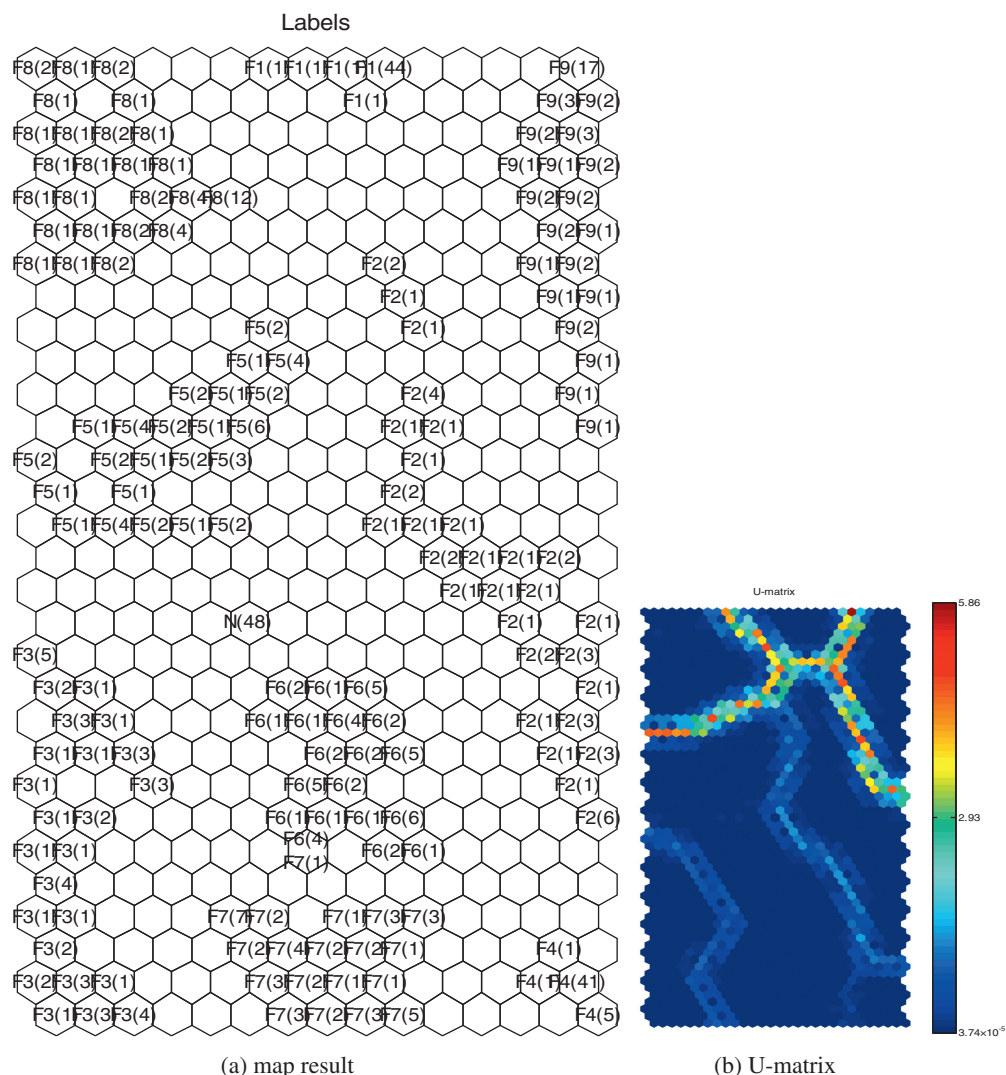


Fig. 5. Mapping result (a) and U-matrix (b) of training data for normal process and faults 1–9 based on LTSA-SOM.

into the blank neuron. Since the blank neuron is in the region of fault 6, the data belong to fault 6. We also notice that one data point is projected into the neuron containing faults 6 and 7. In this situation, the fault data may be from fault 6 or fault 7.

When the magnitude of new test fault data is different from the original training fault data, whether the original training SOM can identify the new test data depends on the situation.

- (1) Fault 1 is the step change of PX feed. The normal value is $24575 \text{ kg} \cdot \text{h}^{-1}$. In training stage, the PX feed changes to $17202.5 \text{ kg} \cdot \text{h}^{-1}$, and in test stage, it changes to $12288 \text{ kg} \cdot \text{h}^{-1}$. Fig. 7(a) is the test result of new test data for fault 1. The new data are mapped into the region of fault 1, because the classification index of SOM is the distance. When the change range between new test data and original training data is not large, the distance between them is still the nearest one, so it can be classified into the original region and identified appropriately.
- (2) Fault 3 is the step change of H_2O feed. The normal value is $10223 \text{ kg} \cdot \text{h}^{-1}$. In training stage, H_2O feed changes to $716.1 \text{ kg} \cdot \text{h}^{-1}$, and in test stage, it changes to $500 \text{ kg} \cdot \text{h}^{-1}$. It is a very large change. In this situation, the new test data has the nearest distance with an unclassified neuron, and the new test data cannot be identified correctly, as shown in Fig. 7(b).
- (3) Fault 6 is the step change of air feed temperature. The normal value is 45°C . In training stage, the air feed temperature changes

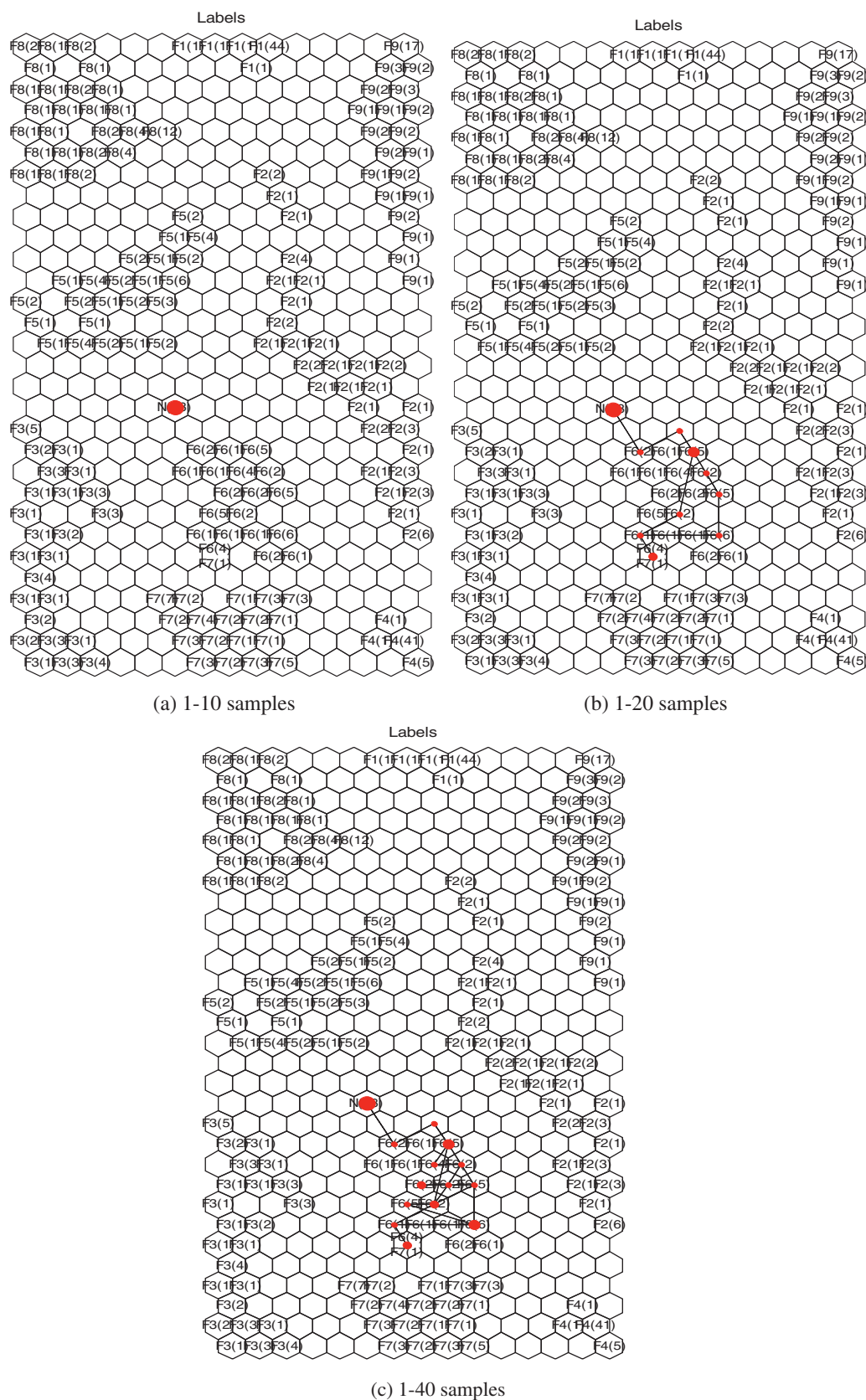
to 55°C , and in the test stage, it changes to 30°C . It is not a large change, but the original training SOM cannot identify the fault accurately, as shown in Fig. 7(c). Because the distance between faults 6 and 7 is small, and a small magnitude change of new fault 6 test data makes it closer to the region of fault 7.

All in all, when the magnitude of new test fault data is different from that of training one, if the change range is small and the distances between original training data and other original training data are large, we can obtain accurate classified results; but if the change range is large or these distances are small, we cannot obtain the right result.

It should be noticed that in a real chemical process, for a certain fault, the fault magnitude usually does not have a very large change. Therefore, the original training SOM can be used.

5. Conclusions

We propose LTSA-SOM to extract the high dimensional data feature and diagnose a PX oxidation reaction process. The measured variables and fault events of PX oxidation process are determined. In the algorithm LTSA-SOM, LTSA can extract feature and reduce dimension while preserving the local geometry in low dimensional space and SOM



diagnosis and it can monitor the trajectory of whole process timely. Therefore, the proposed LTSA-SOM method is efficient and suitable for fault diagnosis and real time monitoring of PX oxidation process.

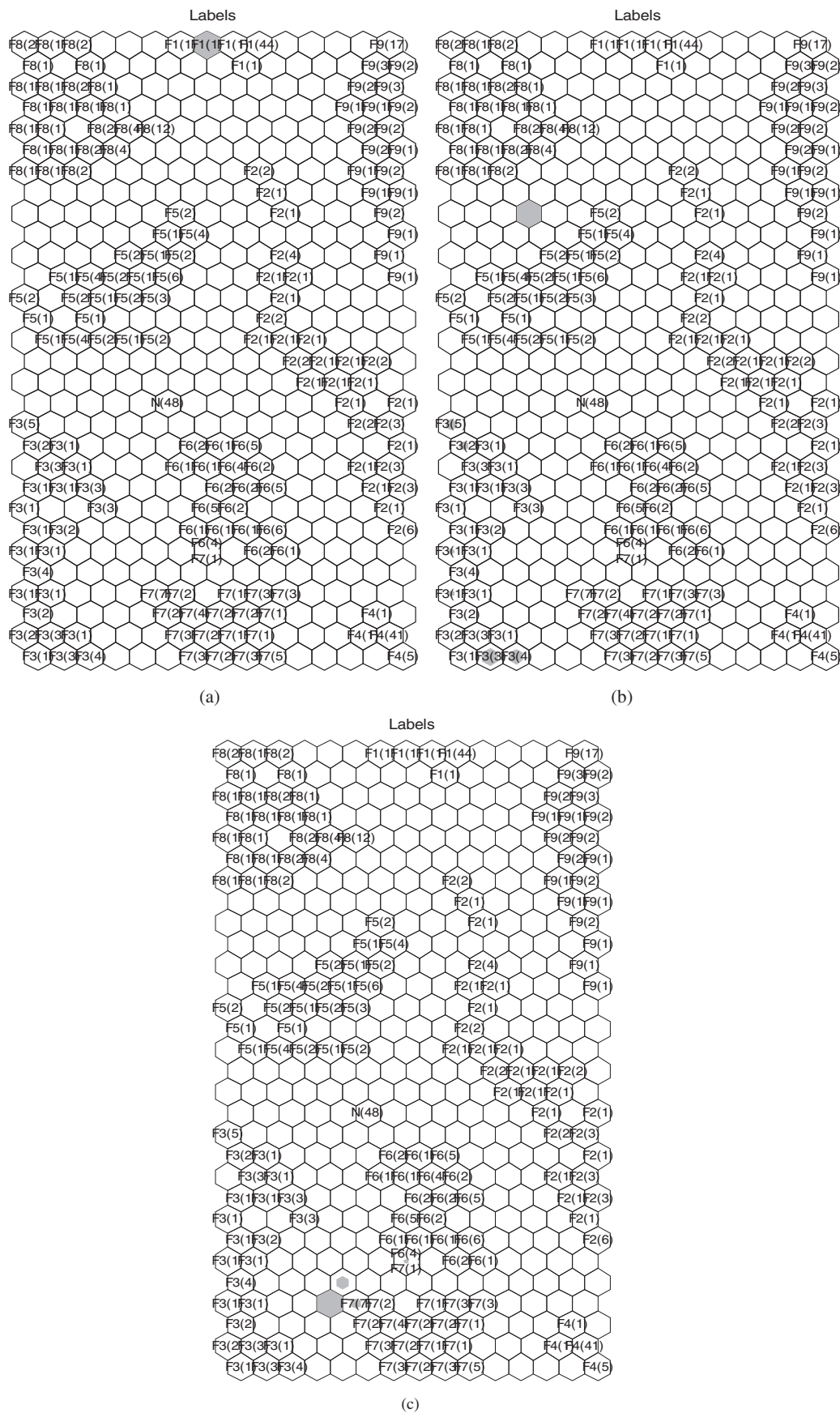


Fig. 7. Mapping results of new test data for faults 1 (a), 3 (b), and 6 (c).

References

- [1] A. Cincotti, R. Orru, A. Broi, G. Cao, Effect of catalyst concentration and simulation of precipitation processes on liquid-phase catalytic oxidation of p-xylene to terephthalic acid, *Chem. Eng. Sci.* 52 (21) (1997) 4205–4213.
- [2] T. Kohonen, The self-organizing map, *Proc. IEEE* 78 (9) (1990) 1464–1480.
- [3] L. Cser, A.S. Korhonen, P. Mantyla, O. Simula, Data mining in improving the geometric quality parameters of hot rolled strips, Proceedings of the 1999 International Conference on Quality Manufacturing, 1999 (Stellenbosch, South Africa).
- [4] H. Saxén, L. Lassus, M. Seppänen, T. Karjalainen, Pattern recognition and classification of blast furnace wall temperatures, *Ironmak. Steelmak.* 27 (3) (2000) 207–211.
- [5] J.S. Van Deventer, D.W. Moolman, C. Aldrich, Visualisation of plant disturbances using self-organising maps, *Comput. Chem. Eng.* 20 (1996) S1095–S1100.
- [6] J. Ahola, E. Alhoniemi, O. Simula, Monitoring industrial processes using the self-organizing map, *Soft Computing Methods in Industrial Applications*, 1999. SMCia/99. Proceedings of the 1999 IEEE Midnight-Sun Workshop on: IEEE, 1999.
- [7] Y.S. Ng, R. Srinivasan, Multivariate temporal data analysis using self-organizing maps. 1. Training methodology for effective visualization of multistate operations, *Ind. Eng. Chem. Res.* 47 (20) (2008) 7744–7757.
- [8] Y.S. Ng, R. Srinivasan, Multivariate temporal data analysis using self-organizing maps. 2. Monitoring and diagnosis of multistate operations, *Ind. Eng. Chem. Res.* 47 (20) (2008) 7758–7771.
- [9] F. Corona, M. Mulas, R. Baratti, J.A. Romagnoli, On the topological modeling and analysis of industrial process data using the SOM, *Comput. Chem. Eng.* 34 (12) (2010) 2022–2032.
- [10] Z. Ge, C. Yang, Z. Song, Improved kernel PCA-based monitoring approach for nonlinear processes, *Chem. Eng. Sci.* 64 (9) (2009) 2245–2255.
- [11] I. Borg, P. Groenen, *Modern Multidimensional Scaling*, Springer, NY, 1997.
- [12] D. DeMers, G. Cottrell, Non-linear dimensionality reduction, *Adv. Neural Inf. Process. Syst.* (1993) 580–580.
- [13] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science* 290 (5500) (2000) 2323–2326.
- [14] M.E. Tipping, C.M. Bishop, Mixtures of probabilistic principal component analyzers, *Neural Comput.* 11 (2) (1999) 443–482.
- [15] Y. Dai, J. Zhao, Fault diagnosis of batch chemical processes using a dynamic time warping (DTW)-based artificial immune system, *Ind. Eng. Chem. Res.* 50 (8) (2011) 4534–4544.
- [16] K. Ghosh, R. Srinivasan, Immune-system-inspired approach to process monitoring and fault diagnosis, *Ind. Eng. Chem. Res.* 50 (3) (2010) 1637–1651.
- [17] X. Chen, X. Yan, Using improved self-organizing map for fault diagnosis in chemical industry process, *Chem. Eng. Res. Des.* 90 (12) (2012) 2262–2277.
- [18] X. Chen, X. Yan, Fault diagnosis in chemical process based on self-organizing map integrated with fisher discriminant analysis, *Chin. J. Chem. Eng.* 21 (4) (2013) 382–387.
- [19] Z. Zhang, H. Zha, Principal manifolds and nonlinear dimensionality reduction via tangent space alignment, *J. Shanghai Univ. Engl. Ed.* 8 (4) (2004) 406–424.
- [20] L. Tao, X. Kong, W. Zhong, F. Qian, Modified self-adaptive immune genetic algorithm for optimization of combustion side reaction of p-xylene oxidation, *Chin. J. Chem. Eng.* 20 (6) (2012) 1047–1052.
- [21] B. Xu, R. Qi, W. Zhong, F. Qian, Optimization of p-xylene oxidation reaction process based on self-adaptive multi-objective differential evolution, *Chemom. Intell. Lab. Syst.* 127 (2013) 55–62.
- [22] F. Qian, L. Tao, W. Sun, W. Du, Development of a free radical kinetic model for industrial oxidation of p-xylene based on artificial neural network and adaptive immune genetic algorithm, *Ind. Eng. Chem. Res.* 51 (8) (2012) 3229–3237.