Multiway dynamic nonlinear global neighborhood preserving embedding method for monitoring batch process

Yongyong Hui\textsuperscript{1,2,3} and Xiaoqiang Zhao\textsuperscript{1,2,3}

Abstract
Aiming at the dynamic and nonlinear characteristics of batch process, a multiway dynamic nonlinear global neighborhood preserving embedding algorithm is proposed. For the nonlinear batch process monitoring, kernel mapping is widely used to eliminate nonlinearity by projecting the data into high-dimensional space, but the nonlinear relationships between batch process variables are limited by many physical constraints, and the infinite-order mapping is inefficient and redundant. Compared with the basic kernel mapping method which provides an infinite-order nonlinear mapping, the proposed method considers the dynamic and nonlinear characteristics with many physical constraints and preserves the global and local structures concurrently. First, the time-lagged window is used to remove the auto-correlation in time series of process variables. Second, a nonlinear method named constructive polynomial mapping is used to avoid unnecessary redundancy and reduce computational complexity. Third, the global neighborhood preserving embedding method is used to extract structures fully after the dynamic and nonlinear characteristics are processed. Finally, the effects of the proposed algorithm are demonstrated by a mathematical model and the penicillin fermentation process.

Keywords
Batch process monitoring, dynamic, nonlinear, global neighborhood preserving embedding

Introduction
Batch process mainly exists in food, semiconductor, chemical production, and so on. It is vital to develop approaches that ensure product quality and production safety requirements. As one of the popular monitoring methods, data-driven statistic process monitoring (SPM) is widely used in batch process.\textsuperscript{1–3}

Compared with the continuous process, the data of batch process have the structure of three-way data array. For the array, tensor and tucker models are used to deal with data directly.\textsuperscript{4–6} Some SPM methods unfold three-way array data to two-way array data, and then traditional SPM models of batch process are established,\textsuperscript{7–9} such as multiway principal component analysis (MPCA)\textsuperscript{8} and multiway partial least squares (MPLS).\textsuperscript{10,11} The methods unfolding three-way array are along the batch direction or variable direction, but the conventional linear monitoring methods cannot describe the underlying nonlinearity of process variables.\textsuperscript{12} For the nonlinear characteristic, the regular methods combine kernel mapping, which linearizes the relationship with high-dimensional projection, such as kernel principal component analysis (KPCA),\textsuperscript{13,14} kernel independent component analysis (KICA),\textsuperscript{15} and kernel Fisher discriminant analysis (KFDA).\textsuperscript{16} However, they only preserve global structure, and the neighborhood structure of data is neglected. The loss of neighborhood structure would inevitably influence the effect of process monitoring.

As one of the manifold learning algorithms, neighborhood preserving embedding (NPE) can preserve the local structure of data and has been widely used in processing monitoring.\textsuperscript{17–21} As a nonlinear extension of NPE,\textsuperscript{22} kernel NPE tries to preserve local topology relationship. But kernel NPE only extracts the neighborhood structure and ignores the global structure.\textsuperscript{23,24}
Therefore, to reveal the inherent properties of process data effectively, Tong and Yan proposed a method that considered the global–local structures. Hui and Zhao divided the batch process variables into related and independent variables and then used the corresponding method to realize process monitoring. Zhao and Tao proposed a tensor global–local model in batch process monitoring. Although the global and local structures were preserved in dimension reduction, in real industrial process, the nonlinear relationships between variables were often limited by physical constraints, resulting in kernel mapping being exorbitant. The type of kernel function determines the dimensions of feature space. The radial basis kernel is widely used to solve nonlinear relationships by providing a higher computational complexity; especially for batch process, it needs to unfold the three-array data into two-array data. After unfolding, the row would be larger than before, so if the kernel mapping is used in batch process, the computational complexity is very high and may be beyond the computer memory.

In the present work, multiway dynamic nonlinear global neighborhood preserving embedding (MDNGNPE) method is proposed for dynamic nonlinear batch process monitoring. Because the data at sampling time \( t \) are related to the data before and after sampling time \( t \), the time-lagged windows are used to remove the auto-correlation in time series of process variables. Then, for the nonlinear characteristic of process data, constructive polynomial mapping (CPM) is used for achieving nonlinear mapping and for considering the physical constraints. The mapped data samples are extracted to the global and local data structures using global neighborhood preserving embedding (GNPE). The MDNGNPE method is applied in a numerical process and the penicillin fermentation process to verify the monitoring effects.

NPE

The NPE preserves reconstruction neighbor relationships in projected low-dimensional subspace. For training dataset \( X \), NPE can extract a linear projection matrix \( A \) to project the dataset \( X \) into low-dimensional subspace. The NPE procedures can be expressed as follows:

1. Construct the adjacency graph: the neighbors are defined by \( k \)-nearest neighbors (knn). If \( x_g \) is one of the knn of \( x_q \), there is an edge between the \( q \)th and \( g \)th node; otherwise, there is no edge.
2. Calculate the weight matrix \( W \): the weight of node \( q \) to node \( g \) can be represented by \( W_{q,g} \); if there is no edge, \( W_{q,g} \) is set as 0. The weight can be computed as follows

\[
\Phi(W) = \min_{W} \sum_{g=1}^{n} \left\| x_q - \sum_{g=1}^{n} W_{q,g} x_g \right\|^2 \tag{1}
\]

where \( \sum_{q=1}^{n} W_{q,g} = 1 \), \( g = 1, 2, \ldots, m \), \( q_1, \ldots, q_k \), is determined by the adjacency graph.
3. Compute the projection: the projection matrix \( P \) can be obtained as follows

\[
\Phi(P) = \sum_{q=1}^{n} \left( y_q - \sum_{g=1}^{n} W_{q,g} y_g \right)^2 \tag{2}
\]

where \( y_q = x_q \), \( W_{q,g} = 1 \), \( g = 1, 2, \ldots, m \), \( q_1, \ldots, q_k \), is determined by the adjacency graph.

MDNGNPE method

Three-way array data unfolding

Batch process consists of batches, variables, and sampling points. Therefore, the data are expressed as \( X(I \times J \times Z) \) (\( I \), \( J \), and \( Z \) represent batches, variables, and sampling points, respectively). Here, we unfold \( X(I \times J \times Z) \) into \( X(I \times ZJ) \). Then, \( X(I \times ZJ) \) is rearranged as \( X(ZI \times J) \). This unfolding method reflects its dynamic characteristic over time. The detailed flowchart is shown in Figure 1.

![Figure 1. Hybrid unfolding of batch process data.](image)
Dynamic nonlinear global neighborhood preserving embedding method

The auto-correlation in time series of process variables widely exists in real industrial process. To remove auto-correlation, an efficient method is to utilize time-lagged data matrix. After batch process data are unfolded by hybrid approach, each variable is augmented on the original time series. The time-lagged data matrix is as follows:

\[
X_d = \begin{pmatrix}
X^T(1) & \ldots & X^T(d-1) & X^T(d) \\
\vdots & \ddots & \vdots \\
X^T(r-d+1) & \ldots & X^T(r-1) & X^T(r) \\
\vdots & \ddots & \vdots \\
X^T(R-d+1) & \ldots & X^T(R-1) & X^T(R)
\end{pmatrix}
\]  

(4)

where \( d \) is the window length, \( R \) is the total number of samples, and \( X(r) = [x_{1,r}, x_{2,r}, \ldots, x_{J,r}]^T \) is the measurement vector of the \( r \)th sampling time \((j = 1, 2, \ldots, J) \) is the measurement variable). The non-linear relationship is limited by some physical constraints. The kernel mapping is used to solve the nonlinear relationship between variables with an infinite-order mapping; this may cause redundancy, and the computational complexity is high.

CPM is introduced to employ nonlinear mapping. For training data \( X \in \mathbb{R}^{m \times n} \), after building time-lagged data matrix \( X_d \in \mathbb{R}^{m \times nd} \) \((nd = n \times d, d \) is the window length), GNPE is used to maintain the global–local structure.

For the NPE algorithm, the local features can be extracted by equation (5) after seeking the nearest \( k \) neighbor points

\[
J(p)_{local} = \min \sum_i \left\| y_i - \sum_j w_{ij} y_j \right\|^2 = \min Y^T(I - W)^T(I - W)Y = \min p^T M p
\]

(5)

where \( Y^T = p^T X \) and \( M = X(I - W)^T(I - W)X^T \); the constraint is \( p^T X X^T p = 1 \).

The global structure is preserved by seeking the maximum direction of variance, as shown in equation (6)

\[
J(p)_{global} = \max \sum^n_{i=1} \left\| y_i - \bar{y} \right\|^2 \quad = \max \sum^n_{i=1} p^T(x_i - \bar{x})(x_i - \bar{x})^T p \quad = \max p^T G p
\]

(6)

where \( \bar{x} = (\sum^n_{i=1} x_i) / n \) and \( G = (X - \bar{x})(X - \bar{x})^T \). Since the low-dimensional space \( Y \) and the original space \( X \) maintain the same maximum variance direction, the same global structures are preserved in the extracted low-dimensional space.

GNPE preserves both the global and local structures, and the optimal objective function is shown in equation (7)

\[
J_{GNPE} = \max_{y} J(y)_G = \max_{y} \frac{\sum^n_{i=1} \left\| y_i - \bar{y} \right\|^2}{\sum^n_{i=1} \left( y_i - \sum_j W_{ij} y_j \right)^2}
\]

(7)

Equation (7) can be converted to equation (8) by the projection relationship \( y_i^2 = p^T X_i \).

\[
J_{GNPE}(p) = \max \frac{p^T G p}{p^T M p}
\]

(8)

Equation (8) can be transformed to solve the generalized eigenvalue problem using equation (9)

\[
G p = \lambda M p
\]

(9)

where the projection matrix \( P \) can be achieved by the eigenvectors corresponding to the largest \( g \) eigenvalues.

The GNPE algorithm which maps time-lagged data matrix \( X_d \in \mathbb{R}^{m \times nd} \) into low-dimensional space \( Y(0) \in \mathbb{R}^{(m - d + 1) \times f} \) \((f < nd) \) can be expressed as follows

\[
Y(0) = P^T(0) X_d
\]

(10)

where \( P(0) \) is the transfer matrix obtained by the GNPE algorithm, and the dimensionality reduction structure \( Y(0) \in \mathbb{R}^{(m - d + 1) \times f} \) can be obtained by equation (10).

Then, \( Y(0) \) undergoes a second-order polynomial mapping to become the mapped data \( H(0) = [h_1(0)^T, h_2(0)^T, \ldots, h_d(0)^T, \ldots, h_{(m-d+1)(0)}^T]^T \)

\[
h_u(0) = [y_u(0),\ldots,y_u(0)^2,\ldots,(y_u(0))^2],
\]

\[
y_u(0),y_u(0),\ldots,y_u(0)^2,\ldots,(y_u(0))^2]
\]

(11)

where \( y_u(0) \) is the element of the obtained dimensionality reduction matrix \( Y(0)(u = 1, 2, \ldots, (m - d + 1)) \). The GNPE algorithm is used to preserve the global and local structures during the process of dimensionality reduction. Then, high-order nonlinear projections are obtained by the iterative method.

Given \( \tilde{H}(0) = [h_1(0)^T, h_2(0)^T, \ldots, h_{(m-d+1)(0)}^T]^T \), for \( k \geq 1 \), we use GNPE to extract the dimensionality reduction matrix \( Y(k) \) based on \( \tilde{H}(k-1) \)

\[
Y(k) = P^T(k) H(k - 1)
\]

(12)

\( Y(k) \) undergoes a second-order polynomial mapping

\[
h_u(k) = [y_u(k),\ldots,y_u(k)^2,\ldots,(y_u(k))^2],
\]

\[
y_u(k),y_u(k),\ldots,y_u(k)^2,\ldots,(y_u(k))^2]
\]

(13)

where \( H(k) = [h_1(k)^T, h_2(k)^T, \ldots, h_d(k)^T]^T \), \( y_u(k) \) is the element of \( Y(k)(u = 1, 2, \ldots, (m - d + 1), v = 1, 2, \ldots, f) \), and the number of eigenvectors is determined.
in the initialization and remains unchanged during every iteration of equation (13).

**Fault detection based on MDNGNPE**

**Fault detection**

After unfolding three-dimensional data matrix \(X(I \times J \times Z)\) into \(X(Z) I J\), we use the proposed MDNGNPE method to establish the statistic model.

For the order of nonlinear mapping, \(K\) is specified by the user. After each iteration, squared prediction error (SPE) and Hotelling’s \(T^2\) statistics can be calculated by equation (14)

\[
\text{SPE}_k(\tau) = (e^2_k(\tau))^T e^2_k(\tau) \\
T^2_k(\tau) = h^2_k(\tau)P(k)(A(k))^{-1}(P(k))^T h^2_k(\tau)^T
\]

where \(e^2_k(\tau) = (h^2_k(\tau) - h^2_k(\tau)B(k)P(k))^T\), \(h^2_k(\tau)\) can be obtained by the online data \(X_{\text{new}}\), \(P(k) = ((B(k))^T B(k))^{-1}B(k)\) is the transformation matrix of the \(k\)th step and \(k \in \{0, 1, \ldots, K\}\) is the iteration times, \(A(k) = (Y(k))^T Y(k)/(n-1)\), and \(n\) is the row of \(Y(k)\). \(H(k)\) is obtained using equations (10)–(13).

The monitoring statistics of MDNGNPE are given as follows

\[
\text{SPE}(\tau) = \left[ \sum_{k=0}^{K} \text{SPE}_k(\tau) \right]^{\frac{1}{2}}
\]

\[
T^2(\tau) = \left[ \sum_{k=0}^{K} T^2_k(\tau) \right]^{\frac{1}{2}}
\]

The SPE and \(T^2\) control limits can be determined by kernel density estimation (KDE). Here, the window length is chosen as 2 to remove the auto-correlation of process variables.

**Monitoring steps**

**Offline monitoring.**

- Step 1. Unfold three-way array data \(X(I \times J \times Z)\) into \(X(Z) I J\) and normalize;
- Step 2. Construct the time-lagged data matrix using equation (4);
- Step 3. CPM is introduced to employ nonlinear mapping;
- Step 4. GNPE is used to reduce dimensions after each iteration of CPM nonlinear mapping, and then the MDNGNPE method is obtained;
- Step 5. Calculate SPE and \(T^2\) statistics based on MDNGNPE;
- Step 6. Compute control limits of SPE and \(T^2\) using KDE.

**Online monitoring.**

- Step 1. Normalize the new batch data;
- Step 2. Construct the time-lagged data matrix for the new samples;
- Step 3. Obtain the nonlinear mapping of new time-lagged data matrix and then project into dimensional reduced subspace using GNPE;
- Step 4. Calculate SPE and \(T^2\) statistics for new samples;
- Step 5. Judge whether the SPE and \(T^2\) statistics exceed the control limits.

The monitoring flowchart based on the MDNGNPE method is shown in Figure 2.

**Simulation verification and analysis**

In order to verify the performance of MDNGNPE, two cases are selected for verification. The first one is a numerical process with dynamic nonlinear characteristic; the second one is a well-known penicillin fermentation process. In these two cases, the monitoring effects of MPCA, multiway neighborhood preserving embedding (MNPE), multiway global neighborhood preserving embedding (MGNE) and MDNGNPE are compared and analyzed.

**Numerical process**

A nonlinear dynamic numerical system is as follows

\[
q(s) = Aq(s-1) + B_l^2 (s-1) \\
l(s) = Cl(s-1) + Dv(s-1) \\
v(s) = q(s) + w(s)
\]

where \(q \in R^{3 \times 1}\), \(l \in R^{2 \times 1}\) is the correlation input, \(v(s)\) is the Gaussian distribution noise with 0 mean and 0.01 variance, and \(w(s)\) is the Gaussian distribution noise with 0 mean and 1 variance. \(A, B, C,\) and \(D\) are as follows

\[
A = \begin{bmatrix}
0.118 & -0.191 & 0.287 \\
0.847 & 0.264 & 0.943 \\
-0.333 & 0.514 & -0.217
\end{bmatrix},
B = \begin{bmatrix}
1 & 2 \\
3 & -4 \\
-2 & 1
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0.811 & -0.226 \\
0.477 & 0.415
\end{bmatrix},
D = \begin{bmatrix}
0.193 & 0.689 \\
-0.320 & -0.749
\end{bmatrix}
\]

The process monitoring variables are \(x(s) = [l_1(s), l_2(s), y_1(s), y_2(s), y_3(s)]\), and the normal operation dataset of 20 batches is generated, in which the small differences of variable correlation coefficients in each simulation are considered, and the whole duration is composed of 300 samples for each batch.

Furthermore, an abnormal case is added to test the effect of MDNGNPE: variable \(l_2\) is linearly increased by 0.08 \((s-150)\) from the 151st sampling point to the end (the whole duration is composed of 300 samples).

MPCA, MNPE, MGNE, and MDNGNPE are applied to monitor the numerical system. Monitoring charts are shown in Figures 3–6, respectively. Figure 3 shows the MPCA monitoring results, and the SPE and \(T^2\) statistics alarm the fault at the 251th and 223th sample, but the fault is added at the 151th sampling point,
due to which there is a large delay in fault detection. Figure 4 shows the SPE and $T^2$ statistics of MNPE; we can see that the fault is detected at the 241th and 240th sample, respectively, which causes a large delay in fault detection. Figure 5 shows the monitoring results of MGNPE; the SPE and $T^2$ statistics alarm the fault at the 241th and 232th sample, respectively, due to which there is a large delay. By contrast, the monitoring results of MDNGNPE are shown in Figure 6; the SPE statistic detects the fault at the 215th sample, and the $T^2$ statistic detects the fault at the 205th sample. It is obvious that MDNGNPE responds more quickly and detects the fault earlier than the other three methods. It shows that MDNGNPE has better monitoring performance than MPCA, MNPE, and MGNPE. This is due to the fact that the MDNGNPE method can fully extract the nonlinear dynamic characteristic of the process data.

**Penicillin fermentation process**

The benchmark simulation of penicillin production is a typical dynamic, nonlinear batch process. In this paper, batch process data were generated by the standard simulation platform Pensim 2.0 in the penicillin fermentation process. Pensim2.0 developed by Illinois Institute of Technology to study the typical batch process more conveniently can produce penicillin fermentation process data under different initial conditions and different working conditions for analysis and research. The production process of penicillin fermentation is shown in Figure 7.
The fermentation process lasts for a total of 400 h, and the sampling interval is 1 h. A total of 36 normal batches are obtained. The 10 measurement variables which can represent the main characteristics of process are chosen, so the normal data $X (36 \times 10 \times 400)$ can be obtained.
The fault detection capability is evaluated using six fault cases that involve different fault types and magnitudes. The details of fault cases are shown in Table 1. Tables 2 and 3 show the monitoring results. We can see from the tables that step fault 1 and step fault 3 can be detected by four methods. Although step fault 5 cannot be detected by these four methods immediately, the performance by the MDNGNPE method is better. There is a delay in fault detection because of the slow spread of the change in glucose substrate rate through the related variables. In contrast to step faults, the ramp faults are hard to detect because of slow changes in its variables. For ramp fault 2, ramp fault 4, and ramp fault 6, the MDNGNPE monitoring method first detects the fault and synthesizes fault detection rate (FDR) and false alarm rate (FAR), which can accurately distinguish normal and abnormal states.

Fault 2 is the ramp fault of aeration rate. Figures 8–11 show the monitoring results of MPCA, MNPE, MGNPE, and MDNGNPE methods under fault 2, respectively. Figure 8 shows the fault detection result of MPCA; the SPE and $T^2$ alarm the fault at the 300th and 308th sample, and it cannot detect the fault on time when fault 2 occurs; the false alarm occurs at the first 40 samples in $T^2$ statistics. Figure 9 shows the SPE and $T^2$ statistics of MNPE; the fault is detected at the 329th and 302th sample, due to which there is a large delay in fault detection; it also generates false alarm at first 40 sampling data in SPE and $T^2$ statistics. Figure 10 shows the monitoring results of MGNPE; SPE detects the fault at the 307th sample and $T^2$ detects the fault at the 259th sample; it also generates false alarm before 45 sampling point data of SPE. By contrast, the SPE and $T^2$ statistics of MDNGNPE in Figure 11 alarm the fault at the 182th and 198th sample, which gives early fault detection results and has higher FDR. Generally, MDNGNPE has better monitoring performance than MPCA, MNPE, and MGNPE.

The second test case is fault 6 in which the fault occurs at the 150th sample point and continues to the end. MPCA, MNPE, MGNPE, and MDNGNPE monitoring charts of fault 6 are shown in Figures 12–15, respectively. Figure 12 shows the fault detection result of MPCA; the SPE and $T^2$ alarm the fault at the 300th and 308th sample, and it cannot detect the fault on time when fault 2 occurs; the false alarm occurs at the first 40 samples in $T^2$ statistics. Figure 9 shows the SPE and $T^2$ statistics of MNPE; the fault is detected at the 329th and 302th sample, due to which there is a large delay in fault detection; it also generates false alarm at first 40 sampling data in SPE and $T^2$ statistics. Figure 10 shows the monitoring results of MGNPE; SPE detects the fault at the 307th sample and $T^2$ detects the fault at the 259th sample; it also generates false alarm before 45 sampling point data of SPE. By contrast, the SPE and $T^2$ statistics of MDNGNPE in Figure 11 alarm the fault at the 182th and 198th sample, which gives early fault detection results and has higher FDR. Generally, MDNGNPE has better monitoring performance than MPCA, MNPE, and MGNPE.

The second test case is fault 6 in which the fault occurs at the 150th sample point and continues to the end. MPCA, MNPE, MGNPE, and MDNGNPE monitoring charts of fault 6 are shown in Figures 12–15, respectively. Figure 12 shows the fault detection result of MPCA; the SPE and $T^2$ alarm the fault at the 300th and 308th sample, and it cannot detect the fault on time when fault 2 occurs; the false alarm occurs at the first 40 samples in $T^2$ statistics. Figure 9 shows the SPE and $T^2$ statistics of MNPE; the fault is detected at the 329th and 302th sample, due to which there is a large delay in fault detection; it also generates false alarm at first 40 sampling data in SPE and $T^2$ statistics. Figure 10 shows the monitoring results of MGNPE; SPE detects the fault at the 307th sample and $T^2$ detects the fault at the 259th sample; it also generates false alarm before 45 sampling point data of SPE. By contrast, the SPE and $T^2$ statistics of MDNGNPE in Figure 11 alarm the fault at the 182th and 198th sample, which gives early fault detection results and has higher FDR. Generally, MDNGNPE has better monitoring performance than MPCA, MNPE, and MGNPE.
of MPCA; the SPE alarms the fault at the 306th sample and \( T^2 \) alarms at the 355th sample; it cannot detect the fault on time, and the false alarm generates at first 40 samples in SPE. We can see from Figure 13 that the SPE and \( T^2 \) of MNPE alarm the fault at the 264th and 300th sample; it also generates false alarm at first 40 sample data in SPE and \( T^2 \). Figure 14 shows the monitoring chart of MGNPE; the SPE and \( T^2 \) alarm the fault at the 281th and 234th sample, due to which there is a large delay in fault alarm, and the false alarm occurs at first 40 sample data in SPE and \( T^2 \). By contrast, the SPE and \( T^2 \) of MDNGNPE in Figure 15 detect the fault earlier than MPCA, MNPE, and MGNPE, which alert the fault at the 241th and 213th sample, respectively. MDNGNPE outperforms MPCA, MNPE and MGNPE in terms of fault detection performance.

The comparison results of average FDR and FAR under the six fault batches are shown in Figure 16. The bar heights in different colors represent the average FDR and FAR of different algorithms. It can be seen from the figure that the average FDR of MDNGNPE is higher than MPCA, MNPE and MGNPE, and the average FAR of MDNGNPE is lower. By intuitive comparison, it further explains the superior monitoring.

### Table 2. Fault detection rate.

| Fault no. | MPCA | MNPE | MGNPE | MDNGNPE |
|-----------|------|------|-------|---------|
|           | SPE  | \( T^2 \) | SPE  | \( T^2 \) | SPE  | \( T^2 \) | SPE  | \( T^2 \) |
| 1         | 1.0000 | 0.5000 | 1.0000 | 0.6240 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| 2         | 0.5040 | 0.6440 | 0.5080 | 0.5880 | 0.5800 | 0.5400 | 0.7333 | 0.7952 |
| 3         | 1.0000 | 0.6520 | 1.0000 | 0.4720 | 0.4920 | 1.0000 | 0.8594 | 1.0000 |
| 4         | 0.2840 | 0.5280 | 0.4240 | 0.4640 | 0.4720 | 0.5080 | 0.3815 | 0.8273 |
| 5         | 0.3840 | 0.3560 | 0.3440 | 0.3960 | 0.5300 | 0.7600 | 0.6132 | 0.8305 |
| 6         | 0.5760 | 0.4240 | 0.6480 | 0.6320 | 0.5720 | 0.7360 | 0.7390 | 0.8434 |

MPCA: multiway principal component analysis; MNPE: multiway neighborhood preserving embedding; MGNPE: multiway global neighborhood preserving embedding; SPE: squared prediction error; MDNGNPE, multiway dynamic nonlinear global neighborhood preserving embedding.

### Table 3. False alarm rate.

| Fault no. | MPCA | MNPE | MGNPE | MDNGNPE |
|-----------|------|------|-------|---------|
|           | SPE  | \( T^2 \) | SPE  | \( T^2 \) | SPE  | \( T^2 \) | SPE  | \( T^2 \) |
| 1         | 0.0200 | 0.1400 | 0.0670 | 0.1400 | 0.0933 | 0.0133 | 0.0267 | 0.0267 |
| 2         | 0     | 0.1000 | 0.1200 | 0.0667 | 0.0933 | 0.0067 | 0.0467 | 0.0200 |
| 3         | 0.1400 | 0.1400 | 0.0667 | 0.1200 | 0.0933 | 0.0133 | 0.0200 | 0.0200 |
| 4         | 0.0133 | 0.1400 | 0.0667 | 0.1200 | 0.0933 | 0.0067 | 0.0200 | 0.0200 |
| 5         | 0.0133 | 0.1400 | 0.0667 | 0.1200 | 0.0800 | 0.0867 | 0.0467 | 0.0533 |
| 6         | 0.1133 | 0.0333 | 0.0867 | 0.1867 | 0.0933 | 0.1200 | 0.0467 | 0.0533 |

MPCA: multiway principal component analysis; MNPE: multiway neighborhood preserving embedding; MGNPE: multiway global neighborhood preserving embedding; SPE: squared prediction error; MDNGNPE, multiway dynamic nonlinear global neighborhood preserving embedding.

![Figure 8. Fault detection results of MPCA under fault 2.](image-url)
Figure 9. Fault detection results of MNPE under fault 2.

Figure 10. Fault detection results of MGNPE under fault 2.

Figure 11. Fault detection results of MDNGNPE under fault 2.
Figure 12. Fault detection results of MPCA under fault 6.

Figure 13. Fault detection results of MNPE under fault 6.

Figure 14. Fault detection results of MGNPE under fault 6.
effect of MDNGNPE over the MPCA, MNPE, and MGNPE.

**Conclusion**

In this paper, an MDNGNPE method is proposed. First, the time-lagged window is constructed to solve the problem of auto-correlation in time series. For nonlinear characteristic, an effective nonlinear mapping method is used to retain the nonlinear structure between process variables, which considers the physical limits when nonlinear mapping is employed. Second, after employing the time-lagged window and nonlinear mapping, GNPE is used to preserve global and local structures and reduce the complexity of nonlinear mapping. Finally, the simulation results obtained from the numerical simulation and the penicillin production process prove the effect of the proposed method. But the parameters in nonlinear mapping need to be set by experience. Therefore, we will build a mathematical model in the future to calculate specific parameters, instead of setting by experience.
Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was financially supported by the National Natural Science Foundation of China (Nos. 61763029, 61873116), the National Defense Basic Research Project of China (No. JCKY2018427C002), Industrial Support and Guidance Project of Colleges and Universities in Gansu Province (No. 2019C-05), Open Fund Project of Key Laboratory of Gansu Advanced Control for Industrial Processes (2019KFJJ05).

ORCID iD

Yongyong Hui https://orcid.org/0000-0001-5014-887X

References

1. Wang H and Yao M. Fault detection of batch processes based on multivariate functional kernel principal component analysis. *Chemometr Intell Lab* 2015; 149: 78–89.
2. Deng X, Tian X, Chen S, et al. Fault discriminant enhanced kernel principal component analysis incorporating prior fault information for monitoring nonlinear processes. *Chemometr Intell Lab* 2017; 162: 21–34.
3. Zhao H, Yao R, Xu L, et al. Study on a novel fault damage degree identification method using high-order differential mathematical morphology gradient spectrum entropy. *Entropy* 2018; 20(9): 682–699.
4. Zhao X and Wang T. Tensor dynamic neighborhood preserving embedding algorithm for fault diagnosis of batch process. *Chemometr Intell Lab* 2017; 162: 94–103.
5. Fanac-Z T H and Gama J. Tensor-based anomaly detection: an interdisciplinary survey. *Knowl-Based Syst* 2016; 98: 130–147.
6. Luo L, Bao S, Gao Z, et al. Batch process monitoring with GTucker2 model. *Ind Eng Chem Res* 2014; 53(39): 15101–15110.
7. Tong C, Lan T and Shi X. Fault detection and diagnosis of dynamic processes using weighted dynamic decen-tralized PCA approach. *Chemometr Intell Lab* 2017; 161: 34–42.
8. Monroy I, Villez K, Graels M, et al. Dynamic process monitoring and fault detection in a batch fermentation process: comparative performance assessment between MPCA and BDPCA. *Comput Aided Chem Eng* 2011; 29(2): 1371–1375.
9. Deng W, Xu J and Zhao H. An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem. *IEEE Access* 2019; 7: 20281–20292.
10. Wang X, Wang P, Gao X, et al. On-line quality prediction of batch processes using a new kernel multiway partial least squares method. *Chemometr Intell Lab* 2016; 158: 138–145.
11. Peng K, Li Q, Zhang K, et al. Quality-related process monitoring for dynamic non-Gaussian batch process with multi-phase using a new data-driven method. *Neurocomputing* 2016; 214: 317–328.
12. Zhao H, Sun M, Deng W, et al. A new feature extraction method based on EEMD and multi-scale fuzzy entropy for motor bearing. *Entropy* 2017; 19(1): 14–34.
13. Lee JM, Yoo CK, Sang WC, et al. Nonlinear process monitoring using kernel principal component analysis. *Chem Eng Sci* 2004; 59(1): 223–234.
14. Zhao H, Zheng J, Xu J, et al. Fault diagnosis method based on principal component analysis and broad learning system. *IEEE Access* 2019; 7: 99263–99272.
15. Tian X, Zhang X, Deng X, et al. Multiway kernel independent component analysis based on feature samples for batch process monitoring. *Neurocomputing* 2009; 72(7–9): 1584–1596.
16. Md Nor N, Hussein MA and Che Hassan CR. Fault diagnosis and classification framework using multi-scale classification based on kernel Fisher discriminant analysis for chemical process system. *Appl Soft Comput* 2017; 61(Supplement C): 959–972.
17. Song B, Tan S and Shi H. Time-space locality preserving coordination for multimode process monitoring. *Chemometr Intell Lab* 2016; 151: 190–200.
18. Su Z, Tang B, Ma J, et al. Fault diagnosis method based on incremental enhanced supervised locally linear embedding and adaptive nearest neighbor classifier. *Measurement* 2014; 48: 136–148.
19. Yuan X, Ge Z, Ye L, et al. Supervised neighborhood preserving embedding for feature extraction and its application for soft sensor modeling. *J Chemometr* 2016; 30(8): 430–441.
20. Miao A, Ge Z, Song Z, et al. Nonlocal structure constrained neighborhood preserving embedding model and its application for fault detection. *Chemometr Intell Lab* 2015; 142: 184–196.
21. Hui Y and Zhao X. Multi-phase batch process monitoring based on multiway weighted global neighborhood preserving embedding method. *J Process Contr* 2018; 69: 44–57.
22. Tao X, Ji H and Men J. Kernel neighborhood preserving embedding for classification. *J Electr* 2009; 26(3): 374–379.
23. Yang J, Zhang M, Shi H, et al. Dynamic learning on the manifold with constrained time information and its application for dynamic process monitoring. *Chemometr Intell Lab* 2017; 167: 179–189.
24. Xiao Z, Wang H and Zhou J. Robust dynamic process monitoring based on sparse representation preserving embedding. *J Process Contr* 2016; 40: 119–133.
25. Tong C and Yan X. Statistical process monitoring based on a multi-manifold projection algorithm. *Chemometr Intell Lab* 2014; 130(2): 20–28.
26. Hui Y and Zhao X. Batch process monitoring based on WGNPE-GSVDD related and independent variables. *Chinese J Chem Eng* 2018; 26(12): 2549–2561.
27. Zhao X and Tao W. Batch process fault diagnosis based on TGNPE algorithm. *CIESC J* 2016; 67(3): 1055–1062.
28. Yu H and Khan F. Improved latent variable models for nonlinear and dynamic process monitoring. *Chem Eng Sci* 2017; 168(Supplement C): 325–338.
29. Yang X and Yin S. Robust global identification and output estimation for LPV dual-rate systems subjected to random output time-delays. *IEEE T Ind Inform* 2017; 13(6): 2876–2885.
30. Yang X and Gao H. Multiple model approach to linear parameter varying time-delay system identification with EM algorithm. *J Franklin Inst* 2014; 351(12): 5565–5581.
31. Roweis S and Ghahramani Z. A unifying review of linear Gaussian models. *Neural Comput* 1999; 11(2): 305–345.
32. Wold S, Kettaneh N, Friden H, et al. Modelling and diagnostics of batch processes and analogous kinetic experiments. *Chemometr Intell Lab* 2008; 44(98): 331–340.
33. Martin EB and Morris AJ. Non-parametric confidence bounds for process performance monitoring charts. *J Process Contr* 1996; 6(6): 349–358.
34. Ku W, Storer RH and Georgakis C. Disturbance detection and isolation by dynamic principal component analysis. *Chemometr Intell Lab* 1995; 30(1): 179–196.
35. Zhang H, Tian X and Deng X. Batch process monitoring based on multiway global preserving kernel slow feature analysis. *IEEE Access* 2017; 5: 2696–2710.
36. Birol G, Ündey C and Çınar A. A modular simulation package for fed-batch fermentation: penicillin production. *Comput Chem Eng* 2002; 26(11): 1553–1565.