Nonlinear Dynamic Process Monitoring Using Canonical Variate Kernel Analysis

Simin Li 1,2,†, Shuang-hua Yang 1,2,*,† and Yi Cao 1,2,†

Abstract: Most industrial systems today are nonlinear and dynamic. Traditional fault detection techniques show their limits because they can hardly extract both nonlinear and dynamic features simultaneously. Canonical variate analysis (CVA) shows its excellent monitoring performance in fault detection for dynamic processes but is not applicable to nonlinear processes. Inspired by the CVA method, a novel nonlinear dynamic process monitoring method, namely, the “canonical variate kernel analysis” (CVKA), is proposed in this work. The way to extract nonlinear features is different from a traditional kernel canonical variate analysis (KCVA). In a sequential structure, the new approach firstly extracts the linear dynamic features from the data through the CVA method, followed by a kernel principal component analysis to extract nonlinear features from the CVA residual space. The new CVKA method is then applied to a TE process case study, proving the excellent performance of CVKA compared to other common approaches in dynamic nonlinear process monitoring for TE-like processes.

Keywords: fault detection; CVA; nonlinear dynamic process; PCA

1. Introduction

Process monitoring and fault detection have become increasingly important to guarantee the normal operation of production, whose task are to determine if there is a fault occurred. As industrial systems become more complex, it becomes increasingly difficult to establish accurate mathematical models and acquire sufficient empirical knowledge. Therefore, techniques based on first-principle models and expert experience are not widely persuasive. Data-driven techniques have been proved to be most effective in practice, with which complex mechanisms and knowledge are not required, and faults can be detected from the change of data.

Multivariate statistical process monitoring (MSPM) techniques [1] such as principal component analysis (PCA) [2] and canonical variate analysis (CVA) [3] can reduce the noise of the original data without losing the most basic information. Low-dimensional data not only help to reduce computation/training time, but also remove redundant features to resolve multicollinearity issues.

Among all the MSPM techniques, the PCA-based techniques is extensively used for fault detection. The main idea is to represent the original space with several orthogonal features, which are called principal components. It is the most direct and simplest dimensionality reduction method. PCA has an assumption of a linear and static system. However, real systems are always nonlinear and dynamic. To cope with the nonlinear problem, Kramer [4] trained a feed-forward neural network to realize a nonlinear mapping named nonlinear PCA (NLPCA). Lee et al. [5] developed kernel PCA (KPCA) for nonlinear process, which extracted nonlinear principal components using kernel functions. For dynamic processes, Ku et al. [6] applied a “time lag shift” method to detect dynamic processes...
which they called dynamic PCA (DPCA). To extract deep features, Deng et al. [7] designed a multilayer PCA model called deep PCA (DePCA). Moreover, Chen et al. [8] applied deep PCA to an electrical drive system for incipient fault detection.

The main limitation of PCA-based methods is that they are based on the fact that measurement variables are time-independent [9]. However, in reality, dynamic industrial processes account for most processes. Different from PCA, CVA takes time correlation into consideration [10]. Although DPCA can detect some dynamic information, Russell et al. [11] extensively tested the effectiveness of DPCA and CVA in process monitoring for dynamic industrial processes. The finding was that CVA performed better both in accuracy and sensitivity. Therefore, CVA seems to be a better choice for dynamic process monitoring.

CVA works by maximizing the correlation between past and future vectors [12], and the CVs can maximally explain the time-related features in these vectors. To address the non-Gaussian issue caused by the nonlinearities of industrial processes, Odiowei et al. [13] combined CVA with upper control limit (UCL), where UCL was directly obtained by a kernel density estimation (KDE) without the Gaussian assumption. Moreover, Samuel et al. [14] performed a CVA with KDE in the KPCA kernel space, which was named kernel canonical variate analysis (KCVA). CVA is not sensitive enough for monitoring incipient faults; Pilario et al. [15] proposed a new CVA dissimilarity-based index and the new method was called canonical variate dissimilarity analysis (CVDA). Conventional single kernels only have good interpolation or extrapolation ability. To overcome this drawback, Pilario et al. [16] used a mixed kernel strategy to keep both good interpolation and extrapolation ability which they called mixed-kernel CVDA (MK-CVDA).

Most industrial systems today are nonlinear and dynamic. However, traditional fault detection method can only address at most one of these features, so the detection performance is not ideal. To cope with this problem, some experts have proposed fault detection methods for nonlinear dynamical systems. Choi et al. [17] proposed dynamic kernel PCA (DKPCA), which expressed nonlinearity with kernel functions and also described the dynamics in a time-expanded way. In a previous work, we combined DPCA and KPCA in a deep model to capture different features [18], which was named deep DPCA (DeDPCA). Guo et al. [19] combined dynamic inner PCA, PCA and KPCA in a sequential structure to deal with the dynamical nonlinear problem. The aforementioned KCVA is also a nonlinear dynamic process monitoring method. Nevertheless, the way the DKPCA describes a nonlinear dynamical system is not effective and it may lose some significant features in the system. The information fusion of different layers in DeDPCA may cause too high a false alarm rate. KCVA only focuses on the dynamics of the principal space but ignores the residual space. Therefore, the monitoring performance of existing nonlinear dynamic fault detection methods is not good enough and a more effective fault detection method is needed.

Current industrial systems always contain linear and nonlinear relationships. Extracting only one type of feature can not fully represent the system. Considering both linear and nonlinear features is better. Inspired by it, a novel hybrid linear–nonlinear dynamic statistical modelling technique is proposed. The main idea is that linear dynamic features are extracted first using the CVA method and nonlinear features are then extracted from the CVA residual space. The CVs of CVA contain most linear dynamic features, leaving some nonlinear features and noise in the residuals. Therefore, nonlinear features are further extracted using nonlinear methods in the CVA residual space. Both features can be fully used for process monitoring. Kernel-based methods and neural networks both can be used to handle the nonlinear problem [20,21]. Kernel-based methods require a low computational complexity and have good nonlinear estimation capabilities if the parameters are appropriate. Hence, KPCA is adopted to extract the nonlinear features from the obtained CVs. Moreover, the $T^2$ and $Q$ statistics are together applied for fault decision and $T^2$ based on different features are fused together to decide the system status. The new method is named canonical variate kernel analysis (CVKA).
There are some contributions in this paper. Firstly, a new hybrid statistical model structure has been presented by combining CVA with KPCA in a sequential structure. Linear dynamic and nonlinear features are both extracted for nonlinear dynamic process monitoring. Secondly, the fusion of Hotelling’s $T^2$ reduces the number of detection metrics and therefore reduces the false alarm rate. Finally, an improved nonlinear dynamic fault detection performance on a TE plant has been attained over existing nonlinear dynamic methods.

The work is organized as follows. In Section 2, the basic idea of CVA-based dynamic fault detection is described. In Section 3, the detailed information of the CVKA method is presented. In Section 4, a case study of the TE process is used to demonstrate the effectiveness of the CVKA method. Section 5 discusses the merits and limitations. Section 6 summarizes this work.

2. Brief Review of the CVA Method

Assume $X \in \mathbb{R}^{n \times m}$ is a matrix which contains $n$ samples. Each sample consists of $m$ variables. $x_k$ is the $k$th row vector of $X$. Denote $x_k = [u_k, y_k]$, where $u_k$ includes input variables and $y_k$ includes output variables. Therefore, in the first feature layer, the $k$th past and $k$th future row vectors of $X$ can be obtained, respectively, as follows:

$$x_{p,k} = \begin{bmatrix} u_{k-1} & u_{k-2} & \cdots & u_{k-q} & y_{k-1} & y_{k-2} & \cdots & y_{k-q} \end{bmatrix}$$

$$= \begin{bmatrix} x_{k-1} & x_{k-2} & \cdots & x_{k-q} \end{bmatrix}$$

$$x_{f,k} = \begin{bmatrix} u_k & u_{k+1} & \cdots & u_{k+q-1} \end{bmatrix}$$

(1)

where $q$ is the past and future measurements. $q$ needs to be as small as possible while capturing the major autocorrelations within the data. $x_{p,k}$ and $x_{f,k}$ are normalized for further study as follows:

$$\tilde{x}_{p,k} = (x_{p,k} - \mu_p) / \sigma_p$$

$$\tilde{x}_{f,k} = (x_{f,k} - \mu_f) / \sigma_f$$

(3)

(4)

where $\mu_p$ and $\mu_f$ are sample means, and $\sigma_p$ and $\sigma_f$ are sample standard deviations.

The past observation matrices $X_p$ and the future observation matrices $X_f$ can be defined as

$$X_p = \begin{bmatrix} \tilde{x}_{p,q+1} & \tilde{x}_{p,q+2} & \cdots & \tilde{x}_{p,q+M} \end{bmatrix}$$

$$X_f = \begin{bmatrix} \tilde{x}_{f,q+1} & \tilde{x}_{f,q+2} & \cdots & \tilde{x}_{f,q+M} \end{bmatrix}$$

(5)

(6)

where $M = n - 2q + 1$.

The covariance matrix $\Sigma_{pp}$ can be estimated as follows:

$$\Sigma_{pp} = \frac{1}{M-1} X_p X_p^T$$

(7)

Moreover, covariance matrix $\Sigma_{ff}$ and cross-covariance matrix $\Sigma_{pf}$ can be expressed in a similar way.

The canonical variables are obtained by applying an SVD to $H$

$$H = \Sigma_{ff}^{-\frac{1}{2}} \Sigma_{pf} \Sigma_{pp}^{-\frac{1}{2}} = U W V^T$$

(8)

where $U$ is the past mapping matrices, $V$ is the future mapping matrices and $W$ is a diagonal matrix. The numbers on the diagonal represent singular values, which is the degree of correlation between column pairs of $U$ and $V$. $W = \text{diag}(\sigma_1, \sigma_2, \cdots, \sigma_r, 0, \cdots, 0)$, where $r_1$ is the rank of $H$ and $1 \geq \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{r_1}$. Thus, the data can be divided into the
state variables \((z_k \in \mathbb{R}^{1 \times r_1})\) and the residual canonical variates \((e_k \in \mathbb{R}^{1 \times (qm - r_1)})\). The state space contains the most linear dynamic features, and the residual space contains the most nonlinear features and noises.

\[
c_k = [z_k \ e_k]
\]  

The state variables \(z_k\) is a linear combination of \(\tilde{x}_{p,k}\), \(z_k = J_x \tilde{x}_{p,k}\), where \(J_x = V_x^T \Sigma_{pp}^{-1} V_x\) with \(V_x\) consisting of the first \(r_1\) columns of \(V\). The residual canonical variates are \(e_k = F_y \tilde{x}_{p,k}\), where \(F_y = V_y^T \Sigma_{pp}^{-1} V_y\) with \(V_y\) consisting of the remaining \(qm - r_1\) columns of \(V\) defined in Equation (8).

The change in state space and residual space can be monitored separately by the \(T^2\) and \(Q\) statistics. They are computed as

\[
T_k^2 = z_k^* z_k^T, \quad (10)
\]

\[
Q_k = e_k^* e_k^T. \quad (11)
\]

Since measurements are always nonlinear or non-Gaussian-distributed [22], determining the upper control limits using a Gaussian assumption is not preferable. Kernel density estimation (KDE) [13] performs well in estimating real probability density functions (PDFs).

The PDF \(\hat{p}(x)\) estimated by KDE is defined as

\[
\hat{p}(x) = \frac{1}{Mh} \sum_{k=1}^{M} K(\frac{x - x_k}{h})
\]  

(12)

where \(K(\cdot)\) is a kernel function, \(x_k\) is the sample of \(x\), and \(h\) is a smoothing parameter called bandwidth.

One popular kernel function can be written in this form

\[
K(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2).
\]  

(13)

According to the real PDF, the control limits \(T^2_{\text{lim}}\) and \(Q_{\text{lim}}\) can be calculated by Equations (14) and (15).

\[
P(T^2 \leq T^2_{\text{lim}}) = \int_{-\infty}^{T^2_{\text{lim}}} p(T^2) d(T^2) = \alpha, \quad (14)
\]

\[
P(Q \leq Q_{\text{lim}}) = \int_{-\infty}^{Q_{\text{lim}}} p(Q) d(Q) = \alpha. \quad (15)
\]

In our proposed CVKA method, KDE is still used to determine upper control limits.

For online monitoring, the \(T^2\) and \(Q\) statistics at each sampling instant can be obtained by Equations (14) and (15). If one of \(T^2\) or \(Q\) exceeds its upper control limit \(T^2_{\text{lim}}\) or \(Q_{\text{lim}}\), a fault is detected.

3. Proposed CVKA Method

CVA has shown its superiority for dynamic process monitoring [11]. However, it can only extract linear features for linear processes. Nonlinear features usually occur in the residuals of the linear model [10] and nonlinear features cannot be distinguished from noise in the residual. Therefore, CVA performs poorly in nonlinear dynamical systems, showing a low detectability for small faults. To extend CVA to nonlinear process, it is worthwhile to further analyse the residuals through nonlinear feature extraction techniques. The proposed CVKA model extracts dynamic features and nonlinear features by integrating CVA and KPCA as shown in Figure 1. Since the linear canonical variables extracted from the original data and the nonlinear PCs extracted from the CVA residual data both can be evaluated by \(T^2\) index, we can fuse them together to reduce the value of the statistic and thus reduce the false alarm rate.
Figure 1. Schematic of CVKA-based process monitoring.

3.1. Construction of CVKA Model

From Figure 1, the linear canonical variables and CVA residual data are obtained through the CVA model. Then, the CVA residual data $\epsilon_k$ are further investigated to dig the nonlinearity of the process. Given a dataset $\epsilon_k \in \mathbb{R}^{1 \times (qm-r_1)}$, $k = 1, \ldots, M$, a nonlinear mapping $\Phi : \mathbb{R}^{1 \times (qm-r_1)} \rightarrow F$ maps $\epsilon$ from the original space to a high-dimensional linear space $F$. The corresponding covariance matrix is calculated as

$$C^F = \frac{1}{M} \sum_{j=1}^{M} \Phi(\epsilon_j)\Phi(\epsilon_j)^T$$  \hspace{1cm} (16)

where $\Phi(\epsilon)$ has mean 0 and variance 1. $C^F$ needs to be diagonalized. This leads to an eigenvalue solving problem

$$\lambda p = C^F p$$  \hspace{1cm} (17)

where $\lambda$ is the eigenvalue of $C^F$, $p$ is the eigenvector. $\lambda$ is a positive number and an eigenvector is a nonzero vector.

Noting that $p$ is a linear combination of $\Phi(\epsilon)$, we have

$$p = \sum_{i=1}^{M} a_i \Phi(\epsilon_i).$$  \hspace{1cm} (18)

Multiplying both sides of Equation (19) by $\Phi(\epsilon_k)$ gives

$$\lambda \langle \Phi(\epsilon_k), p \rangle = \langle \Phi(\epsilon_k), C^F p \rangle.$$  \hspace{1cm} (19)

Substituting Equations (16) and (18) in (19), we have

$$\lambda \sum_{i=1}^{M} a_i \langle \Phi(\epsilon_k), \Phi(\epsilon_i) \rangle = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{M} a_i \Phi(\epsilon_k), \Phi(\epsilon_j) \rangle \langle \Phi(\epsilon_j), \Phi(\epsilon_i) \rangle.$$  \hspace{1cm} (20)

The kernel trick can be applied to $\Phi(\epsilon)$ to construct an $M \times M$ kernel matrix

$$K_{ij} = \langle \Phi(\epsilon_i), \Phi(\epsilon_j) \rangle = k(\epsilon_i, \epsilon_j), \quad i, j = 1, \ldots, M.$$  \hspace{1cm} (21)

$K$ can be centred by this way

$$K_c = K - I_M K - K I_M + I_M K I_M$$  \hspace{1cm} (22)
where $I_M$ is an $M \times M$ matrix where each element is $\frac{1}{M}$. Then, Equation (20) can be rewritten as

$$\lambda \sum_{i=1}^{M} a_iK_{ki} = \frac{1}{M} \sum_{i=1}^{M} \alpha_i \sum_{j=1}^{M} K_{ij}K_{ji}$$

(23)

which can be expressed in this form

$$M\lambda \alpha = K_r \alpha$$

(24)

resulting in the projection vector $\alpha_i \in \mathbb{R}^M$ and the corresponding nonlinear score vector $t_i = K_r \alpha_i \in \mathbb{R}^{1 \times r_2}$, $1 \leq i \leq M$ and $r_2$ is the number of PCs.

A typical kernel function can be denoted as

$$k(x_i, x_j) = \exp(-\frac{||x_i - x_j||^2}{h})$$

(25)

Therefore, the $T^2$ and $Q$ indices can be obtained as

$$T^2_k = s_k^T s_k$$

$$Q_k = ||\Phi(t_k) - \hat{\Phi}(t_k)||^2 = \Sigma_{i=1}^{M} t_i^2 - \Sigma_{i=1}^{r_2} t_i^2$$

(26)

(27)

where $s_k = [z_k^T, t_k^T]$. $\Phi(t_k)$ is the reconstruction of vector $\Phi(t_k)$.

As mentioned above, similar to the CVA algorithm, KDE was also adopted to estimate the PDF of $T^2$ and $Q$ and calculate the control limits $T^2_{\text{lim}}$ and $Q_{\text{lim}}$.

For online monitoring, the $T^2$ and $Q$ statistics are calculated using Equations (26) and (27). For CVKA-based fault detection, if one of $T^2$ or $Q$ exceeds its upper control limit $T^2_{\text{lim}}$ or $Q_{\text{lim}}$, a fault is detected.

3.2. Summary of the Proposed CVKA Scheme

The CVKA monitoring procedure consists of two steps. In offline training, the normal data are analysed by the CVKA technique to get the mapping vectors and compute the control limits. Online monitoring uses the sample’s continuous collection to check if a fault has occurred. Figure 2 shows the algorithm flowchart. Detailed steps are summarized in Algorithm 1.

**Algorithm 1** Detailed steps of CVKA-based fault detection

**Offline training:**
- **Step1.** Collect the normal data $X$ and compute the past and future data series using (1) and (2);
- **Step2.** Compute the Hankel matrices and perform an SVD on the scaled Hankel matrix from (7) and (8);
- **Step3.** Determine the canonical variables according to (9);
- **Step4.** Construct the kernel matrix from the canonical variables and determine the principal variables and residuals;
- **Step5.** Compute the monitoring indices using (26) and (27) and their control limits using (14) and (15), respectively;

**Online monitoring:**
- **Step6.** Acquire test data and construct the past and future vectors;
- **Step7.** Compute the canonical variables and project them to the kernel space as for the training data;
- **Step8.** Compute $T^2$ and $Q$ of the test data using (26) and (27);
- **Step9.** Judge if a fault occurs by comparing $T^2$ and $Q$ with their respective control limits.
4. Case Study

The CVKA method was compared with the DKPCA, DeDPCA and KCVA methods by applying them to a benchmark TE plant. Firstly, CVKA and DKPCA were compared to see how CVKA could improve the detection performance by considering the linear and nonlinear features in a sequential structure. Then, CVKA was compared with DeDPCA
to show the superiority of the state-space-based feature extraction method for dynamic process monitoring. Finally, CVKA was compared with KCVA to prove the necessity of further analysing the CVA residual space through nonlinear feature extraction methods.

4.1. Overview

The TE process [23,24] is the benchmark process shown in Figure 3. The TE production process consists mainly in the reaction of four gaseous materials, producing two products and a by-product. There are 52 variables in the whole TE process. A total of 20 faults are used for verification, including 4 types of different faults, respectively, step, random, slow drift and sticking. Moreover, there are five faults whose fault types are unknown. The corresponding data can be found in [25]. Training datasets and testing datasets all consist of 960 samples. The fault dataset contains 160 normal samples and 800 fault samples.

![Graphical description of the TE plant.](image)

4.2. Implementation Details

For the DKPCA, DeDPCA, KCVA and CVKA methods, the Gaussian kernel function was chosen and the kernel width was empirically set to $\sigma = 50$. For DeDPCA and DKPCA, the calculation results of time lag $k$ was two using Ku’s theory [6]. For KCVA and CVKA, the autocorrelations in the data became insignificant when the number $q$ of past and future measurements was five. A threshold of 90% was set for calculating the number of PCs and CVs. $T^2$ and $Q$ were used jointly. The KDE method was used to calculate the control limits. Before applying DKPCA, DeDPCA, KCVA and CVKA, all data were standardized to avoid dimensional effects.

The fault detection rate (FDR), the fault detection time (FDT) and the false alarm rate (FAR) were used to evaluate the performance. The FDR is the probability of detecting a fault under system fault conditions and measures the accuracy of a fault detection method. The FDT is the time it takes to first discover a fault. The FAR is the probability of falsely detecting a fault under normal system conditions and measures the robustness of a fault detection method.

4.3. Results and Discussion

To appreciate the performance of CVKA, the detection processes for F18 were firstly compared. F18 is an unknown fault. Figure 4 shows the fault detection results using different methods to monitor the TE process. The control limits are drawn with a red dashed line, while the monitoring statistics are drawn with a black solid line. For DKPCA,
it can be seen that the fault was not obvious at first and the $T^2$ and Q statistics took 76 sample intervals to first detect that fault. The fault could be identified after that time and the total FDR was 90.63%. Moreover, as time went by, the data shift in the residual space was more obvious than that in the principal space. For DeDPCA, the fault detection rate was 90.88% and the fault detection time was 31 sample intervals. It was better than DKPCA, which reflected the advantages of a hierarchical feature extraction. For KCVA, the fault detection rate was 90.88% and the fault detection time was 24 sample intervals after introducing the fault, earlier than that of DKPCA and DeDPCA. It was because the principal kernel space was further analysed using CVA to extract dynamic features. However, ignoring the residual kernel space may lose some critical linear information, leading to a high false alarm rate. Figure 4d confirms that CVKA outperformed DKPCA, KCVA and DeDPCA. Specifically, most linear and nonlinear features were reflected in the $T^2$ index and the total FDR was 91.38%, larger than that of DKPCA, KCVA and DeDPCA. The fault was first detected at the 163th sample, much earlier than that of any other methods. By comparing these results, it is apparent that CVKA is more suitable than DKPCA, KCVA and DeDPCA for TE-like nonlinear dynamic process monitoring.

Figure 4. Monitoring results for TE fault 18. (a) DKPCA. (b) DeDPCA. (c) KCVA. (d) CVKA.

Table 1 lists all the monitoring results of the 20 TE faults using DKPCA, DeDPCA, KCVA and CVKA. According to the difficulty of detection, the faults can be divided into three categories. Faults 1, 2, 4, 7, 8, 11, 12, 14 and 18 form the first category. These faults are obvious and can be easily detected using DKPCA. Faults 5, 6, 10, 13, 16, 17, 19 and 20 form the second category. These faults are not so obvious but still can be detected most of the time using DKPCA. Faults 3, 9 and 15 form the last category. These faults cause little change to the system and are difficult to detect using any fault detection method.

The CVKA method seemed to be most sensitive to all fault conditions in terms of FDR and FDT. CVKA had the largest FDRs for all faults except faults 8 and 10 in category one and category two, which means that CVKA could detect nearly all faults with a high accuracy. Moreover, CVKA obtained the smallest FDTs for most faults, indicating that DeCVDA could always detect faults earlier than any other methods. All methods kept a
relatively low FAR except KCVA, and the difference between the other three methods was not particularly obvious. CVKA obtained the largest average FDR, smallest average FDT and a relatively low average FAR, meaning that CVKA was more efficient. KCVA obtained the worst performance possibly because it only focused on principal kernel features but neglected the information in the residual subspace when doing kernel mapping, which also illustrated the necessity of further analysing the residual space through nonlinear feature extraction methods. The performance of DeDPCA was slightly better than that of DKPCA. Both methods are PCA-based methods. The only difference is that DKPCA extracts dynamic features and nonlinear features at the same time but DeDPCA extracts dynamic features and nonlinear features in different layers. It can be concluded that a layerwise feature extraction structure has the advantage of extracting more appropriate features. CVKA is generally better than DeDPCA, indicating that a state-space-based feature extraction CVA method is more suitable for extracting dynamic features.

Table 1. FDRs, FDTs and FARs of 20 TE faults using DKPCA, KCVA, DeDPCA and CVKA.

<table>
<thead>
<tr>
<th>Fault</th>
<th>FDR (%)</th>
<th>FDT (Sample)</th>
<th>FAR (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DKPCA</td>
<td>DeDPCA</td>
<td>KCVA</td>
</tr>
<tr>
<td>1</td>
<td>99.75</td>
<td>99.25</td>
<td>99.63</td>
</tr>
<tr>
<td>2</td>
<td>98.50</td>
<td>99.25</td>
<td>98.75</td>
</tr>
<tr>
<td>4</td>
<td>100.00</td>
<td>100.00</td>
<td>97.75</td>
</tr>
<tr>
<td>7</td>
<td>100.00</td>
<td>100.00</td>
<td>98.75</td>
</tr>
<tr>
<td>8</td>
<td>100.00</td>
<td>99.00</td>
<td>97.63</td>
</tr>
<tr>
<td>11</td>
<td>90.50</td>
<td>93.88</td>
<td>92.38</td>
</tr>
<tr>
<td>12</td>
<td>99.25</td>
<td>100.00</td>
<td>95.00</td>
</tr>
<tr>
<td>14</td>
<td>100.00</td>
<td>100.00</td>
<td>97.75</td>
</tr>
<tr>
<td>18</td>
<td>90.63</td>
<td>90.88</td>
<td>90.88</td>
</tr>
<tr>
<td>mean1</td>
<td>97.63</td>
<td>87.00</td>
<td>96.95</td>
</tr>
<tr>
<td>5</td>
<td>30.38</td>
<td>100.00</td>
<td>40.75</td>
</tr>
<tr>
<td>6</td>
<td>100.00</td>
<td>100.00</td>
<td>7.13</td>
</tr>
<tr>
<td>10</td>
<td>60.88</td>
<td>94.13</td>
<td>41.00</td>
</tr>
<tr>
<td>13</td>
<td>95.38</td>
<td>96.13</td>
<td>85.38</td>
</tr>
<tr>
<td>16</td>
<td>54.88</td>
<td>96.75</td>
<td>29.63</td>
</tr>
<tr>
<td>17</td>
<td>97.63</td>
<td>97.75</td>
<td>88.88</td>
</tr>
<tr>
<td>19</td>
<td>51.00</td>
<td>99.63</td>
<td>74.50</td>
</tr>
<tr>
<td>20</td>
<td>63.63</td>
<td>92.00</td>
<td>59.38</td>
</tr>
<tr>
<td>mean2</td>
<td>62.92</td>
<td>97.05</td>
<td>53.33</td>
</tr>
</tbody>
</table>

From the detection results in terms of the FDRs, FDTs and FARs of DKPCA, DeDPCA, KCVA and CVKA, we can find that for most conditions, CVKA achieved a higher FDR, an earlier FDT and a relatively low FAR, which means that the proposed CVKA technique was able to find faults earlier and more accurately and comprehensively. Hence, for a dynamic nonlinear system similar to the TE plant, CVKA is a more efficient method than others.

5. Discussion

5.1. Outcomes

A new nonlinear dynamic process monitoring method was proposed based on the multilayer model. The new model consisted of two layers, which were a CVA layer and a KPCA layer. The first CVA layer explained the lineairities and dynamics and the second KPCA layer explained the nonlinearities. A TE plant was used as a case study for verification. The results showed that CVKA attained a higher FDR, an earlier FDT than DKPCA, DeDPCA and KCVA, while keeping a relatively low FAR.
5.2. Limitations

One limitation is that in a hierarchical statistical model it is not easy to get the contribution of each original variable to the fault. The number of mappings increases the difficulty of the contribution calculation. Therefore, the interpretation is not obvious, and it is hard to find out the cause of the failure.

Another limitation is that when the CVA algorithm constructs the historical matrix and the future matrix, the dimensionality of the data is greatly increased, and the subsequent dimensionality reduction processing effect is not ideal. It may increase the computational complexity.

5.3. Further Research

Future work may include selecting optimum parameters, such as the type of kernel functions, how many PCs or CVs should be retained and so on. Moreover, computing the contributions of each variable to a fault may be one potential future work.

6. Conclusions

In this paper, the significance of nonlinear dynamic fault detection in industrial processes was emphasized and the shortcomings of existing nonlinear dynamic MSPM techniques were analysed. To solve this problem, a novel nonlinear dynamic process monitoring technique was proposed in which KPCA was applied to the CVA residual space, called CVKA. The new method could extract dynamic linear and nonlinear features efficiently. A kernel-based method was applied to avoid a complex neural network optimization. The fusion of $T^2$ based on different features reduced the false alarm rate. Using a TE case study designed to simulate 20 faults, the CVKA method was shown to be superior to DKPCA, KCVA and DeDPCA methods in nonlinear dynamic process monitoring for TE-like systems.

Author Contributions: Conceptualization, Y.C. and S.L.; methodology, Y.C. and S.L.; software, S.L.; validation, S.L.; formal analysis, S.-h.Y., Y.C. and S.L.; investigation, S.L.; resources, S.-h.Y. and Y.C.; data curation, S.L.; writing—original draft preparation, S.L.; writing—review and editing, S.-h.Y. and Y.C.; visualization, S.L.; supervision, S.-h.Y. and Y.C.; project administration, S.-h.Y.; funding acquisition, S.-h.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.