

Article

# Integrated Diagnostic Framework for Process and Sensor Faults in Chemical Industry

Jiaxin Zhang <sup>1,2</sup>, Wenjia Luo <sup>2</sup>  and Yiyang Dai <sup>1,\*</sup> <sup>1</sup> School of Chemical Engineering, Sichuan University, Chengdu 610065, China; 201821000250@stu.swpu.edu.cn<sup>2</sup> School of Chemistry and Chemical Engineering, Southwest Petroleum University, Chengdu 610500, China; luowenjia@swpu.edu.cn

\* Correspondence: daiyy@scu.edu.cn

**Abstract:** This study considers the problem of distinguishing between process and sensor faults in nonlinear chemical processes. An integrated fault diagnosis framework is proposed to distinguish chemical process sensor faults from process faults. The key idea of the framework is to embed the cycle temporal algorithm into the dynamic kernel principal component analysis to improve the fault detection speed and accuracy. It is combined with the fault diagnosis method based on the reconstruction-based contribution graph to diagnose the fault variables and then distinguish the two fault types according to their characteristics. Finally, the integrated fault diagnosis framework is applied to the Tennessee Eastman process and acid gas absorption process, and its effectiveness is proved.

**Keywords:** process and sensor fault; cycle temporal algorithm; dynamic kernel principal component analysis; reconstruction-based contribution; integrated diagnostic framework



**Citation:** Zhang, J.; Luo, W.; Dai, Y. Integrated Diagnostic Framework for Process and Sensor Faults in Chemical Industry. *Sensors* **2021**, *21*, 822. <https://doi.org/10.3390/s21030822>

Academic Editors: Giovanni Betta and Miguel Delgado-Prieto  
Received: 2 December 2020  
Accepted: 22 January 2021  
Published: 26 January 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The main goal of the chemical industry is to improve the efficiency and accuracy of manufacturing facilities. As the automation level of the chemical industry continues to improve, the scale of production increases and the complexity of the system grows continually. The probability of failure and fault in chemical system also increases. Compared with the permanent termination of the system caused by failure, the system offset caused by fault can be detected in time, and then corresponding measures can be taken to avoid accidents [1]. Therefore, distinguishing the type of faults in the chemical process is the key to reducing operator errors and ensuring system safety and reliability.

In recent years, many fault diagnosis methods for chemical processes have emerged. These methods diagnose different faults [2–5]. However, no distinction is made between the types of faults themselves. In an actual chemical process, faults can be divided into process faults and sensor faults [6]. Process faults are characterised by multivariate coordination. The occurrence of process fault means that the operating state of the system deviates from the normal value. Sensor fault has variable independence, and the fault variable is unique. The sensor fault interferes with the stability of the system and affects the judgment of the operator, which may lead to fault. Therefore, it is very important to diagnose process faults and sensor faults in modern industrial processes [7–9].

The fault diagnosis method is the key to ensure safe and effective operation of the process [10]. Generally, fault diagnosis methods can be divided into three categories: model-based [11], knowledge-based [12], and data-based [13] methods. Among them, the principal component analysis (PCA) in data-based methods is widely used for process and sensor fault diagnosis in chemical processes [14,15]. Ku et al., Lee et al., and Yang et al. successively proposed the dynamic principal component analysis (DPCA) [16], kernel principal component analysis (KPCA) [17], and dynamic kernel principal component analysis (DKPCA) [18] for process fault diagnosis. Qin et al. [19] improved the DPCA

method and proposed a comprehensive model for sensor fault diagnosis. Wang et al. [20] put forward a new strategy based on PCA for sensor fault diagnosis. Li et al. [21] proposed a sensor fault diagnosis method based on density-based clustering and PCA. When using various PCA models for fault diagnosis, the diagnosis space is divided into two subspaces called principal subspace and residual subspace.  $T^2$  statistics and squared prediction error (SPE) statistics are constructed to characterise the mean and variance information, respectively, of the processes in the two subspaces [22]. In the process of fault detection using  $T^2$  and SPE statistics, the following four types of monitoring results may appear: (1) neither of the statistics exceeds the control limit; (2)  $T^2$  statistics do not exceed the limit, but SPE statistics exceed the limit; (3)  $T^2$  statistics exceed the limit; and SPE statistics exceed the limit. (4)  $T^2$  statistics and SPE statistics both exceed the limit. When the SPE statistics change significantly (results (2) and (4)), it means that the normal operating model represented by the PCA model is destroyed, and a process fault or sensor fault has occurred. When the  $T^2$  statistic changes significantly and the SPE statistic does not change (result (3)), it indicates that the relationship between the variables is approximately satisfied, but the system has undergone some transformation, which may be due to a change in operating conditions or a system fault. Result (1) indicates that the system is normal [23]. According to results (2) and (4), this study makes a detailed distinction between process faults and sensor faults in chemical processes.

Chemical processes are characterised by large amounts of data and complex calculations. The abovementioned improved methods also have this limitation. To detect the occurrence of faults more rapidly and accurately, the PCA method has been improved accordingly. In temporal logic algorithms [24], the temporal logic can qualitatively describe the behaviour of the system over time, because it has a large number of operators such as 'always' and 'final'. Clarke et al. [25] proposed a computational temporal logic to describe the behaviour of concurrent systems. For chemical processes, a cycle temporal algorithm (CTA) was proposed based on traditional temporal logic to solve a large amount of chemical data and the problem of low calculation accuracy. By combining it with the DKPCA method, it can capture most of the fault information in the system and improve the detection accuracy. the operator can respond to the fault timely and effectively. The fault type is further distinguished through the fault identification step.

When the system detects that a fault has occurred, it needs to locate the current fault location and fault variables, and then determine whether the fault is a process fault or a sensor fault. Currently, the most widely used method for fault variable location is the contribution graph [26]. The method is based on quantifying the contribution of each process variable to a single principal component score. The contribution of each process variable to the main component in the out-of-control state is added, and it is called variable contribution [27,28]. In recent years, many variable identification methods have been reported [29–32]. This study uses the reconstruction-based contribution (RBC) graph method to reconstruct the faulty variables, emphasises the cross-correlation between variables, and clearly shows the relationship between them. The variable characteristics of different faults are used to determine whether the fault type is a process fault or a sensor fault, so as to provide technical support for the next operation of the subsequent operator.

The current chemical process fault diagnosis methods for process and sensor faults are independent of each other, Sensor faults on non-control loops will not directly cause accidents, but will affect operator judgment and the accuracy of process fault diagnosis. Therefore, it is necessary to distinguish between sensor faults and process faults. A process fault is when the process itself deviates from the normal state, and the sensor fault process itself does not deviate from the normal state, but the sensor index deviates from the true value. Sensor faults on non-control loops will not directly cause accidents, which will affect operator judgment and the accuracy of process fault diagnosis. There is no corresponding integrated model to distinguish process and sensor faults in detail. When a sensor fault occurs in a chemical process, it is regarded as a process fault by default, which could cause operator errors and accidents. Therefore, it is particularly important to distinguish between

process faults caused by real fault fluctuations and sensor faults indicated by false data. This paper proposes a CTA to improve the calculation speed and eliminate the redundancy problem caused by big data calculation. The CTA combined with DKPCA improves the fault detection speed and accuracy. Combined with the fault identification model based on the RBC graph, it emphasises the strong correlation of variables while extracting fault information. Finally, an integrated diagnostic framework for distinguishing sensor and process faults in a chemical process is obtained. The framework has the advantages of fast detection speed, high detection accuracy, and accurate fault variables. Its application to the Tennessee Eastman (TE) process and acid gas absorption process proves the effectiveness of the proposed integrated fault diagnosis framework.

The remainder of this paper is structured as follows. In Section 2, the CTA, DKPCA method, and RBC graph are introduced. In Section 3, the method proposed in the previous section is used to build an integrated fault diagnosis framework to distinguish between process and sensor faults. In Section 4, The TE process is compared with other fault detection methods to prove the advantages of the proposed method, and then four faults in the TE process and three faults in the acid gas absorption process are selected to demonstrate the effectiveness of the integrated fault diagnosis framework in distinguishing process and sensor faults. In Section 5, we summarise our work.

## 2. Fault Diagnosis Methods

### 2.1. Dynamic Kernel PCA

Generally, the PCA can only be effectively performed in the observation set of linear and steady-state changes. When the data change nonlinearly and dynamically, the PCA method can be converted to changes in linear data after mapping. However, the analysis method based on the kernel function does not need to calculate the eigenvector as the PCA method but to convert it into the eigenvalue and eigenvector of the kernel matrix. Thus, it avoids the calculation for obtaining the eigenvector in the high-dimensional space and converting it into projection, solving the linear combination of kernel functions, and by capturing the data dynamic matrix [23–35], it also solves the dynamic matching problem of the PCA model. Hence, the calculation is greatly simplified.

Assuming that the normal data set  $\mathbf{X}$  contains  $m$  variables, and each variable has  $n$  observations, the vectors at time  $t$  and augmented matrix  $\mathbf{X}(s)$  containing the observations at the previous  $s$  time to reflect the relationship between the variables' dynamic relationship.

$$\mathbf{X}(s) = \begin{bmatrix} x_t^T & \cdots & x_{t-s}^T \\ \vdots & \ddots & \vdots \\ x_{t+s-n}^T & \cdots & x_{t-n}^T \end{bmatrix} \quad (1)$$

Then, the dynamic matrix  $\mathbf{X}(s)$  is used to establish the dynamic principal component model through the PCA method, and then the dynamic characteristics can be analysed.

$$\mathbf{X}(s) = \mathbf{TP}^T + \mathbf{E} \quad (2)$$

where  $\mathbf{T}$  is the score matrix,  $\mathbf{P}$  is the load matrix, and  $\mathbf{E}$  is the residual matrix, which is the projection of the sample in the residual space.

For nonlinear problems, the principle of the KPCA is to map the input data to the high-dimensional feature space through the inner product kernel function  $\Phi$  to perform the PCA, thereby turning the two linearly inseparable points in the low-dimensional space into linearly separable ones. After nonlinear mapping, the observation vector  $x_i(t : t - s)$  is mapped to  $\Phi_i(t : t - s)$ , and the dynamic data augmentation matrix  $\mathbf{X}(s)$  is mapped to  $\Phi(s)$ .

Then, the covariance matrix of the feature space can be expressed as

$$\mathbf{C}^F = \frac{1}{N} \sum_{i=1}^N \phi_i(t:t-s) \phi_i^T(t:t-s) \quad (3)$$

Suppose the eigenvalue  $\lambda$  corresponding to the matrix  $\mathbf{C}^F$ , the eigenvector  $\nu$ , and the coefficient  $\alpha_i (i = 1, 2, \dots, N)$  such that

$$\nu = \sum_{i=1}^N \alpha_i \phi_i(t:t-s) \quad (4)$$

The corresponding characteristic equation is

$$\begin{aligned} \lambda \nu &= \mathbf{C}^F \nu = \left( \frac{1}{N} \sum_{i=1}^N \phi_i(t:t-s) \phi_i^T(t:t-s) \right) \nu \\ &= \frac{1}{N} \sum_{i=1}^N \langle \phi_i(t:t-s), \nu \rangle \phi_i(t:t-s) \end{aligned} \quad (5)$$

$$\lambda \langle \phi_k(t:t-s), \nu \rangle = \langle \phi_k(t:t-s), \mathbf{C}^F \nu \rangle \quad (6)$$

Equations (4) and (6) are combined to obtain:

$$\lambda \sum_{i=1}^N \alpha_i^k \langle \phi_k(t:t-s), \phi_i(t:t-s) \rangle = \frac{1}{N} \alpha_i^k \langle \phi_k(t:t-s), \sum_{j=1}^N \phi_j(t:t-s) \rangle \langle \phi_j(t:t-s), \phi_i(t:t-s) \rangle \quad (7)$$

By defining the kernel matrix  $\mathbf{K} \in \mathbf{R}^{N \times N}$ ,  $[\mathbf{K}_{ij}] = \mathbf{K}_{ij} = \langle \phi_j(t:t-s), \phi_i(t:t-s) \rangle$ , from the above formula, the feature vector can be obtained as

$$\lambda N \alpha = \mathbf{K}_{ij} \alpha \quad (8)$$

where  $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$ .

Before adopting the PCA on the feature space  $F$ , we first standardise the data, that is, replace  $\mathbf{K}$  with the following equation:

$$\mathbf{K} = \mathbf{K} - \mathbf{I}_N \mathbf{K} - \mathbf{K} \mathbf{I}_N + \mathbf{I}_N \mathbf{K} \mathbf{I}_N \quad (9)$$

where  $\mathbf{I}_N$  is equal to  $\frac{1}{N}$  multiplied by an  $N \times N$  identity matrix  $\mathbf{E} \in \mathbf{R}^{N \times N}$ . Therefore, the PCA in the feature space  $F$  is equivalent to solving the eigenvalue of Equation (6).

Combining Equations (8) and (4), the eigenvector  $\alpha$  of the kernel matrix  $\mathbf{K}$  can be derived from the eigenvector  $\nu$  of the matrix  $\mathbf{C}^F$ , and it satisfies

$$\langle \nu_k, \nu_k \rangle = 1 \quad (10)$$

where  $k = 1, 2, \dots, p$ .  $p$  is the number of principals.

By calculating the projection of the mapping data on the feature vector  $\nu_k$ , we find the principal component:  $t_k = \langle \nu_k, \phi(t:t-s) \rangle = \sum_{i=1}^N \alpha_i^k \langle \phi_i(t:t-s), \phi(t:t-s) \rangle = \sum_{i=1}^N \alpha_i^k \mathbf{K}_{ij}$ .

To solve the eigenvalues of Equation (8), we use  $t_k$  to calculate the feature space principal vector in the input space and introduce the kernel function  $K = \exp(-\|x - y\|^2 / 2\sigma^2)$  into the feature space to avoid directly calculating the nonlinear mapping.

## 2.2. Cycle Temporal Algorithm (CTA)

In this section, a new method, the CTA, is proposed, which extends the qualitative trend analysis (QTA) method [36] to temporal constraints, extracts based on temporal series, and calculates the difference between the temporal threshold and the linear fitting error relationship. The linear correlation between variables is preserved, and the principal component is obtained by the subsequent DKPCA method. The correlation saves most

of the process differences. Each principal component is a linear combination of all the variables. The data are cyclically segmented and merged, which reduces the amount of calculation, increases the calculation speed, and eliminates the redundancy generated by the calculation of the large data matrix.

The temporal logic equation is summarised and defined as follows:

$$\varphi | T | \overline{G}_{[b,e]} \varphi | \overline{F}_{[b,e]} \varphi | \neg \varphi | \varphi_1 \wedge \varphi_2 | \varphi_1 \overline{U}_{[b,e]} \varphi_2, \quad (11)$$

where T is 'always true', 'negation' ( $\neg$ ), and 'combination' ( $\wedge$ ), which are standard Boolean join operators. The time series operators  $\overline{G}_{[b,e]}$ ,  $\overline{F}_{[b,e]}$ , and  $\overline{U}_{[b,e]}$  represent the derived 'always', the derived 'final', and the derived 'until', and  $[b, e]$  represents the time interval satisfying  $b \leq e$ . This study defines  $u(x[\tau]) := f(x[\tau]) \leq 0$  and  $f(x[\tau]) = (x[\tau] - k\tau - c)^2 - \delta_e$ .  $\delta_e$  is the preset threshold. Where  $\delta_e = 4\sigma^2$ ,  $\sigma$  is the standard deviation of the data sequence.  $x[\tau]$  represents the data point at time  $\tau$ .  $k$  and  $c$  are constant parameters.

To eliminate the problem of slow calculation and calculation redundancy in the state of big data, the parameters and thresholds are extracted on the basis of temporal logic, and the data are processed through cycle segmentation and cycle merge. The specific process of the CTA is shown in Figure 1.

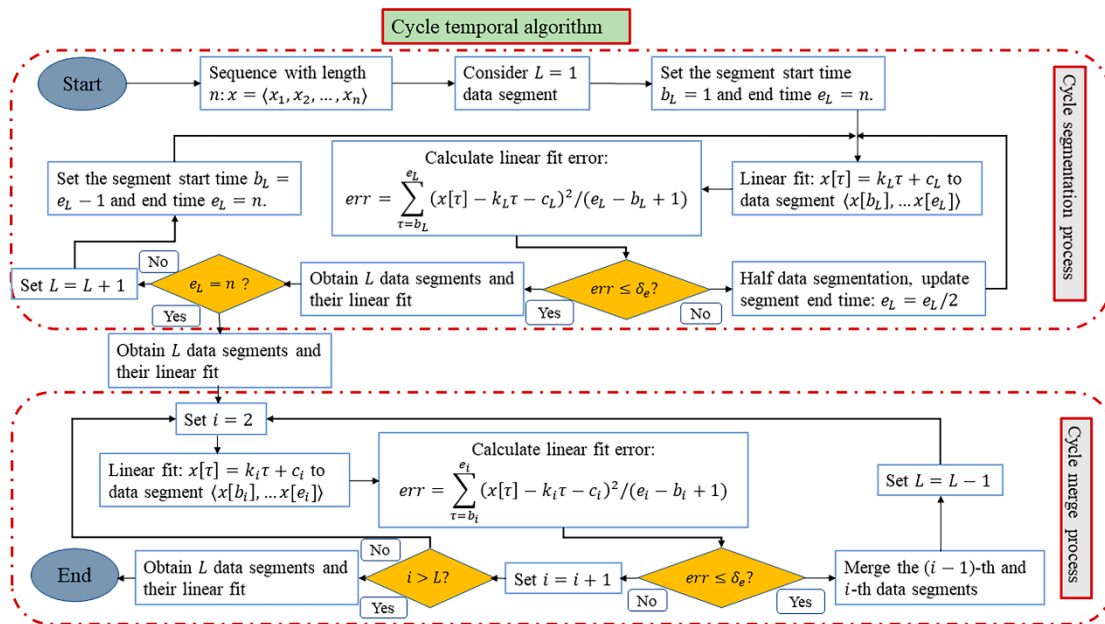


Figure 1. Cycle temporal algorithm (CTA) process.

The CTA process is divided into a cycle segmentation process and a cycle merge process. For the cycle segmentation process, according to the entire data sequence  $x = \langle x_1, x_2, \dots, x_n \rangle$ ,  $L = 1$  is set as the data segment start, the segment start point  $b_L$  and the end point  $e_L$  are set, and a linear fitting is performed on the segment. The fitting equation is  $\hat{x}[\tau] = k_L\tau + c_L$ , where  $k_L$  and  $c_L$  represent the slope and the y-axis intercept, respectively. If the linear fitting error  $err$  is greater than the extracted threshold  $\delta_e$ , the sequence  $x$  is halved. The linear fitting of the first half of the sequence is repeated until the linear fitting error is not greater than the predetermined error threshold  $\delta_e$ . First, it is determined whether the end of the segment reaches the end point, and then, it is

determined whether the data are divided into  $L$  segments. The linear fitting calculation is expressed in Equation (12).

$$err = \sum_{L=b_L}^{e_L} (x[\tau] - k_L - c_L)^2 / (e_L - b_L + 1) \quad (12)$$

For the cycle merge process, the obtained  $L$  segment data sequence is used as the input into the cycle merge process. First, the data segment is counted by  $i = 2$ , and the multi-segment data are linearly fitted. Contrary to the cycle segmentation part, when the linear fitting error  $err$  is less than the threshold  $\delta_e$ , merge the  $i$ -th sequence with the  $i - 1$ th sequence, then subtract 1 in segments, and merge in turn until the linear fitting error is greater than the threshold  $\delta_e$ . It is judged whether  $i$  is greater than  $L$ , and if it is, the merge ends.

Based on the above segmented fitting method, cycle segmentation and cycle merge are performed for a given data sequence. A sequence of quaternions of degree  $L$  can be obtained.

$$T\langle (k_1, c_1, b_1, e_1), (k_2, c_2, b_2, e_2), \dots, (k_L, c_L, b_L, e_L) \rangle \quad (13)$$

In each quadruple  $(k_i, c_i, b_i, e_i) (i \in [1, L])$ ,  $k_i$  and  $c_i$  are the slope and  $y$ -axis intercept of the  $i$ -th segment of linear fitting, respectively. The integers  $b_i$  and  $e_i$  represent the start and end times of the data segment, respectively, and the corresponding atomic predicate is expressed in Equation (14).

$$u_i((x[\tau] - k_i\tau - c_i)^2 - \delta_e \leq 0) \quad (14)$$

Based on the above, the temporal logic equation is reconstructed as:

$$\varphi\varphi_1U_{[b_1,e_1]}\varphi_2U_{[b_2,e_2]}\dots\varphi_{L-1}U_{[b_{L-1},e_{L-1}]}\varphi_L \quad (15)$$

Among them, if  $i \in [1, L - 1]$ , then  $\varphi_i u_i$ .

Eventually, the CTA is formed. The CTA combines the temporal logic to define the principle of the segmented cycle and determines the size of the cycle, the error threshold, and the final segment starting point and end point. The cycle segmentation of the model data is completed to reduce the dimensionality of the matrix, improving the calculation accuracy and eliminating the calculation redundancy.

### 2.3. RBC Graph

The numerical method of the RBC graph is used to identify the variable contribution rate.

Fault variable reconstruction is used to realise the accurate identification and separation of process and sensor faults by eliminating the influence of observed variables in the fault subspace.

When the system detects a fault  $f_i$ , the observation vector is  $x = [x_1, x_2, \dots, x_m]^T$ , where  $i = 1, 2, \dots, m$ , and  $m$  represent the number of process variable observations. The sampling time is omitted here, but the nature of the problem is not changed. If the process vector is reconstructed along the  $\zeta_i^T$  direction, the corresponding relationship is

$$z_i = x - \zeta_i^T f_i \quad (16)$$

where  $z_i$  represents the original value of the process view vector that is not affected by the fault, and  $\Phi$  is defined as a positive definite symmetric matrix. Then, the joint index  $index^2(z_i)$  for process monitoring can be obtained:

$$index^2(z_i) = z_i^T \Phi z_i = \|z_i\|_{\Phi}^2 = \|x - \zeta_i^T f_i\|_{\Phi}^2 \quad (17)$$

When the reconstruction vector  $z_i$  is infinitely close to or equal to the original value of the observation vector during the normal operation of the process, the joint index in the above equation reaches the minimum, indicating that this is the optimal reconstruction. Therefore, according to Equation (17), the first derivative of the fault parameter is calculated and set to be equal to 0, so that the optimal fault parameter is

$$f_i = \left( \zeta_i \Phi \zeta_i^T \right)^{-1} \zeta_i \Phi x \quad (18)$$

From Equations (17) and (18), it can be deduced that the reconstruction contribution of the variable  $x_i$  to the joint index is

$$RBC_i^{index} = \|\zeta_i^T f_i\|_{\Phi}^2 = \|\zeta_i^T \left( \zeta_i \Phi \zeta_i^T \right)^{-1} \zeta_i \Phi x\|_{\Phi}^2 = x^T \Phi \zeta_i^T \left( \zeta_i \Phi \zeta_i^T \right)^{-1} \zeta_i \Phi x \quad (19)$$

Substituting the relationship  $\Phi = L\Lambda L^T$  into the above equation to restore, the reconstruction contribution of the variable  $x_i$  to the joint index is obtained as:

$$RBC_i = x^T L\Lambda L^T \zeta_i^T \left( \zeta_i L\Lambda L^T \zeta_i^T \right)^{-1} \zeta_i L\Lambda L^T x = \frac{(\zeta_i L\Lambda L^T x)^2}{(L\Lambda L^T)_i} \quad (20)$$

where  $(L\Lambda L^T)_i$  represents the  $i$ -th diagonal component of matrix  $L\Lambda L^T$ .

### 3. Integrated Fault Diagnosis Framework

Modern chemical process faults are mainly divided into process faults and sensor faults. Process faults are caused by large deviations in the system, and they are real faults in the system. Sensor faults are caused by sensor faults of the detection system, and the data are falsely indicated. The system remains normal. Therefore, a process fault is a multi-variable coordination, and a sensor fault is a false indication of  $\zeta_i$  of a single measured variable. To further distinguish the types of faults when faults occur, this paper proposes an integrated fault diagnosis framework to distinguish the two types of faults in detail. The integrated fault diagnosis framework is shown in Figure 2.

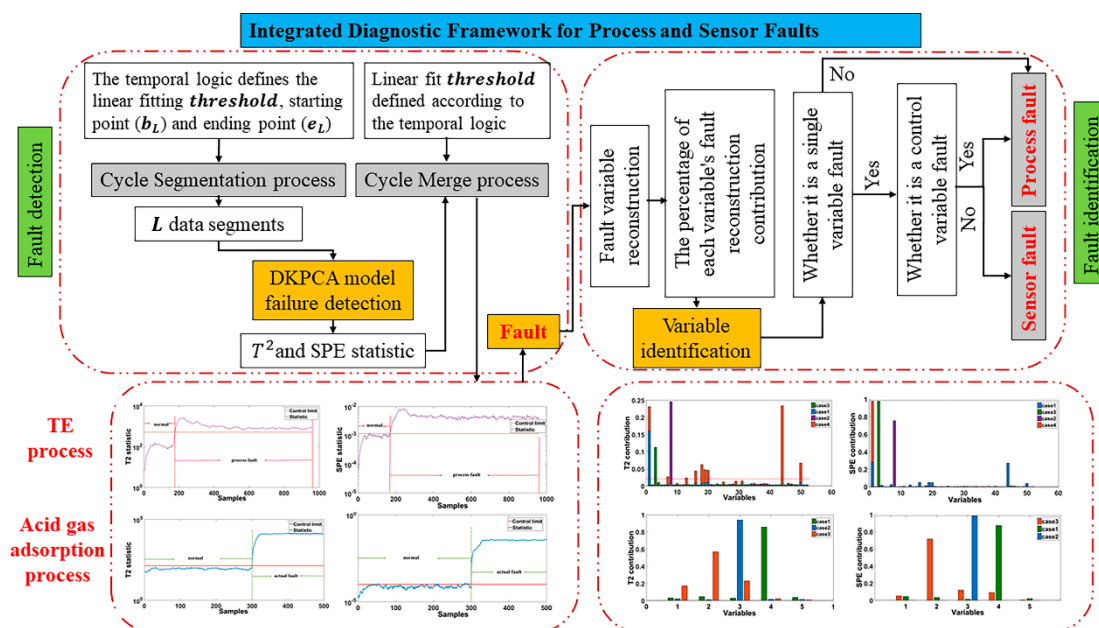


Figure 2. Integrated diagnosis framework for process and sensor faults.

The integrated fault diagnosis framework for process and sensor faults is mainly composed of two parts: fault detection and fault identification. Through the fault detection step, various faults in the system are accurately and rapidly detected. Fault identification involves identifying specific fault variables by reconstructing the variable diagram method for the detected fault and distinguishing the fault types by using the process and sensor fault characteristics.

### 3.1. Fault Detection

The fault detection part of the integrated fault diagnosis framework combines the CTA and DKPCA models. Using the cycle segmentation and cycle merge characteristics of the CTA model, the detected data are finely divided and calculated, which minimises a series of external influences such as noise, and the segmented calculation also speeds up the detection speed and accuracy. The fault information is saved to the greatest extent. The framework detects system faults timely and effectively.

First, the detailed process of cycle segmentation is shown in Figure 3.

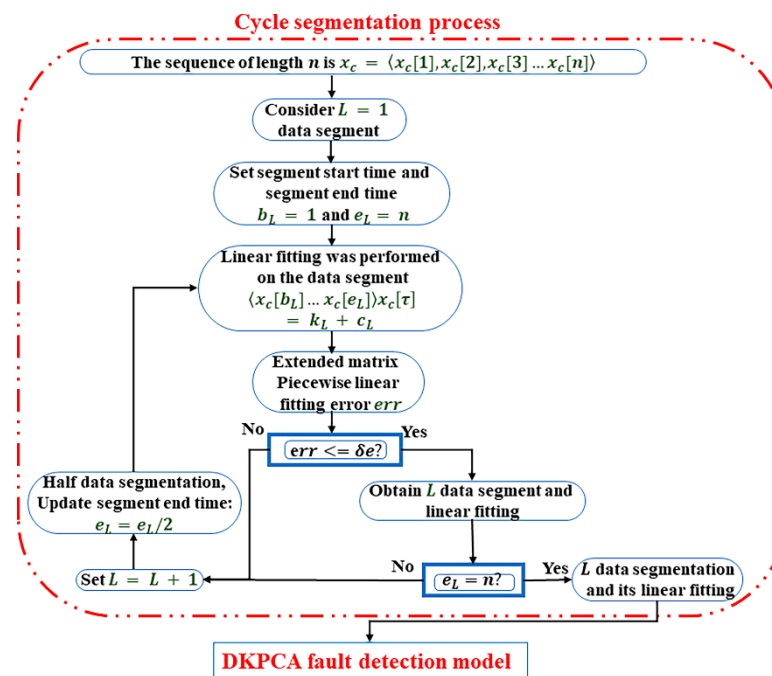


Figure 3. Cycle segmentation process of fault detection.

- (1) For the process data set, first define a segment as the first segment, that is,  $L = 1$ .
- (2) Extract the optimal threshold  $\delta_e$  from the temporal logic equation for the first segment of the process data set that has been determined, and define the start time  $b_1$  and end time  $e_1$  of the segment.
- (3) Perform piecewise data fitting for segment  $L = 1$ . The fitting equation is shown in Equation (10). Calculate the linear piecewise fitting error  $err$ , and determine whether the linear fitting error reaches the threshold  $\delta_e$ . If  $err$  is less than  $\delta_e$ , the segmentation stops and the current segment number is the optimal number of segments. If  $err$  is greater than  $\delta_e$ , go to step 4.
- (4) Divide the data that do not meet the threshold requirement,  $L = L + 1$ , define the segment end time  $e_L = e_L / 2$  for the new data segment, and repeat step 3 for linear fitting. Cycle segmentation calculates the linear fitting error  $err$ , until  $err$  is less than the threshold  $\delta_e$ , and enters the next judgment.
- (5) Judge in the data segment that reaches the threshold  $\delta_e$  whether the end point of the current segment reaches the end  $n$  point of the initial process data; if not, reset the start



$b_L = e_L - 1$  and end points  $e_L = n$  of the unsegmented part, and enter again the step 3 of the cycle segmentation part. When the end point is reached, the cycle segmentation process ends.

After obtaining the best segmentation data of the  $L$  segment, the segmentation data are used as the input into the DKPCA fault detection model. The fault detection process is shown in Figure 4.

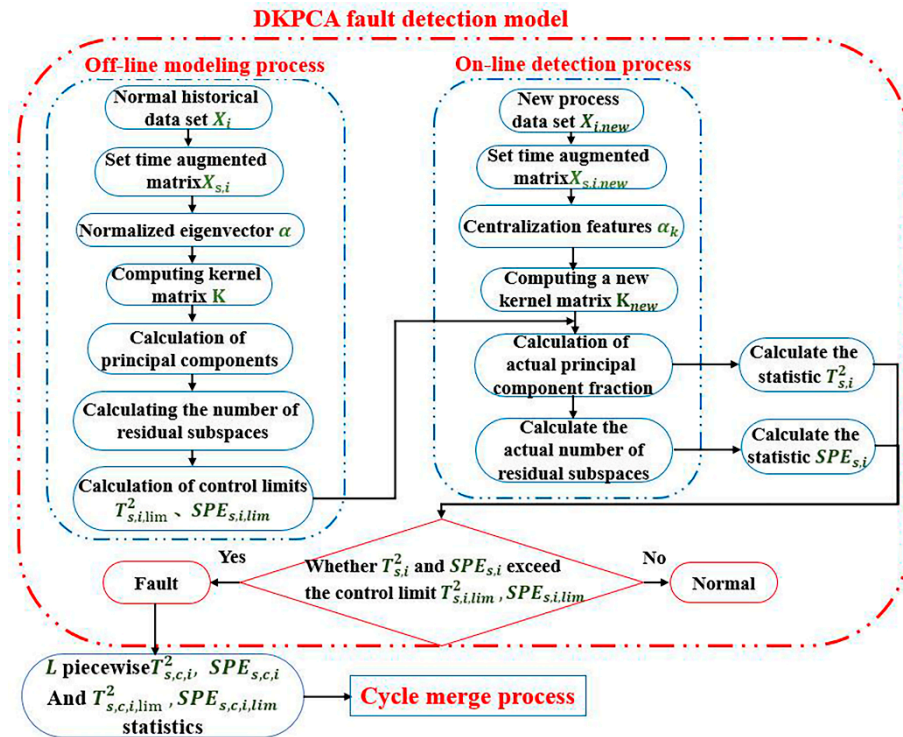


Figure 4. DKPCA model calculation process.

The DKPCA model calculation is divided into offline modelling process and online detection process.

Offline modelling process:

(1) The offline modelling part collects the normal segmentation data set  $X_i$ , sets the superposition time  $t$  and the previous time  $s$  of each data set, and obtains the dynamic augmentation matrix  $X_{s,i}$ .

(2) Standardise feature vector  $\alpha$ ; the feature vector is calculated using Equation (8).

(3) Calculate the piecewise kernel matrix  $K$ , where the kernel function is selected as the Gaussian kernel function  $K = \exp(-\|x - y\|^2 / 2\sigma^2)$ .

(4) Calculate the number of principal spaces and the number of residual subspaces according to the cumulative percentage of variance (CPV) method. The equation for calculating the cumulative variance percentage is as follows:

$$CPV = \frac{\sum_{i=1}^N \lambda_i}{\sum_{i=1}^J \lambda_j} \quad (21)$$

where  $\lambda$  is the characteristic value of covariance,  $J$  is the number of variables, and  $N$  is the number of principal components.

(5) Calculate the standard control limit of the no-fault condition  $SPE_{i,lim}, T_{i,lim}^2$ , as shown in Equations (22) and (23), and enter step 5 of online detection after the calculation is completed.

$$SPE_{s,i,lim} = \frac{v_i}{2m_i} \chi_{\frac{2m_i}{v_i, \alpha}}^2 \quad (22)$$

$$T_{s,i,lim}^2 = \frac{l_i(J_i^2 - 1)}{J_i(J_i - 1)} F_{l_i, J_i - l_i, \alpha} \quad (23)$$

where  $v_i$  and  $m_i^2$  respectively represent the mean and variance based on  $\mathbf{X}_{s,i}$ .

Online detection process:

- (1) Enter the new process data set  $\mathbf{X}_{i,new}$  that needs to be detected. In the same way as step 1 of the offline stage, set the vector of the superimposition time  $t$  and the previous time  $s$  to obtain the dynamic augmentation matrix  $\mathbf{X}_{s,i,new}$ .
- (2) The obtained dynamic augmentation matrix is used to centralise the feature vector.
- (3) Calculate the new kernel matrix  $\mathbf{K}_{new}$  by segmenting the centralised data. The kernel function used is consistent with the offline phase, which is a Gaussian kernel function.
- (4) Calculate the number of principal spaces using the CPV method.
- (5) Calculate nonlinear components and principal component scores. The equation is as follows:  $t_{k,new} = \langle v_k, \phi_{new}(t : t - s) \rangle = \sum_{i=1}^N \alpha_i^K \langle \phi_i(t : t - s), \phi(t : t - s) \rangle = \sum_{i=1}^N \alpha_i^K \mathbf{K}_{new}$ .
- (6) Calculate the new data statistics  $T_{s,i}^2$  and  $SPE_{s,i}$  based on the reference control limits  $T_{s,i,lim}^2$  and  $SPE_{s,i,lim}$  offline non-fault conditions, as shown in Equations (24) and (25). Verify whether  $T_{s,i}^2$  or  $SPE_{s,i}$  statistics exceed the control limit  $T_{s,i,lim}^2$ ,  $SPE_{s,i,lim}$ . If they exceed the control limit, it indicates that the system has a fault condition, and then enter the fault identification stage. Otherwise, it is determined to be normal, and return to step 1 to continue testing another new set of process data.

$$SPE_{s,i} = (z^i)^T (\mathbf{I}_{J_i} - \hat{\mathbf{P}} (\hat{\mathbf{P}})^T) z^i \leq SPE_{s,i,lim} \quad (24)$$

$$T_{s,i}^2 = (z^i)^T \hat{\mathbf{P}} (\hat{\Lambda}^i)^{-1} (\hat{\mathbf{P}})^T z^i \leq T_{s,i,lim}^2 \quad (25)$$

where  $\hat{\Lambda}^i = (\mathbf{T}_n^i)^T \mathbf{T}_n^i / (N_n - 1)$ , and  $\hat{\mathbf{P}} (J_i \times l_i)$  and  $\mathbf{T}_n^i (N_n \times l_i)$  are the load matrix and score matrix of the  $i$ -th DKPCA model, respectively.  $\mathbf{I}_{J_i} (J_i \times J_i)$  is the unit matrix. In addition, let the column vector  $z (J \times 1)$  represent the current observation data point and divide  $z$  them into the corresponding  $L$  blocks based on the  $L$  variable block in the previous section, which is  $z^i (J^i \times 1) (i = 1, 2, \dots, L)$ .

After detecting the fault of the segmented data, the segmented data are used as the input into the cycle merge process for merging, so as to determine the confirmation of the fault of the entire process.

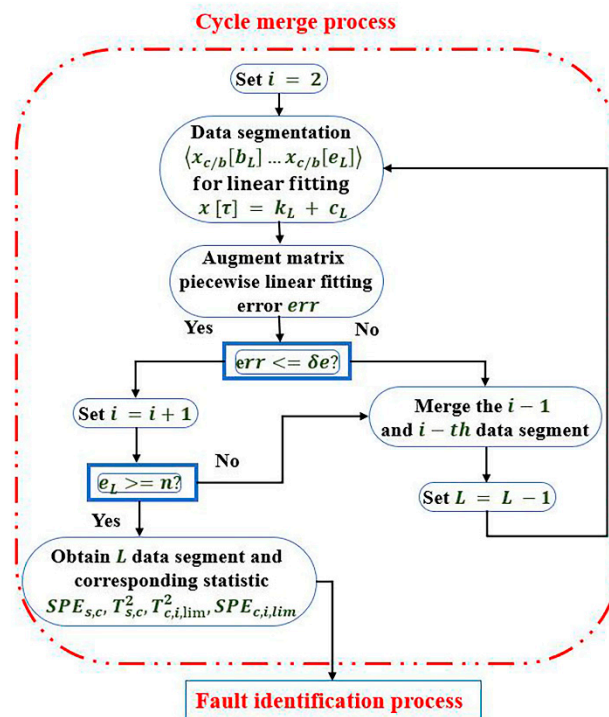
The detailed process of cycle merge is shown in Figure 5.

(1) The  $L$  segment  $T_i^2$  and  $SPE_i$  statistics obtained after the DKPCA model calculation part is completed are used as the input into the cycle merging stage.

(2) Set  $i = 2$ , merge according to the reverse step of cycle segmentation, perform linear fitting on the first two segments, calculate the linear fitting error  $err$ , and determine whether the linear fitting error reaches the obtained threshold  $\delta_e$ .

(3) If  $err$  is greater than the threshold  $\delta_e$ , go to step 4. If the threshold  $\delta_e$  is greater than  $err$ , merge the  $i$ -th and  $i - 1$ th segments, and the total length of the data segment at this time is  $L - 1$ . Return to the stage of step 2 to merge, the merged data and the next segment data are reset to the first and second segments, and the merge is repeated.

(4) When  $err$  is greater than the threshold  $\delta_e$ , it is judged whether the number of merged data segments  $L$  is less than  $i$ . If it is smaller than  $i$ , the cycle ends, and the final merged data segment is obtained. If  $L$  is greater than  $i$ , then continue to merge until  $L$  is greater than  $i$ . The cycle merge ends.



**Figure 5.** Cycle merge process of fault detection.

After completing the cycle merge process, the fault detection part of the integrated fault diagnosis model is completed, and the two statistics  $T_{s,i}^2$  and  $SPE_{s,i}$  detected are determined. If  $T_{s,i}^2$  and  $SPE_{s,i}$  exceed the calculated control limits  $T_{s,i,lim}^2$  and  $SPE_{s,i,lim}$ , the system is considered to be faulty, and the fault identification part of the next part is further analysed.

### 3.2. Fault Identification

After the fault detection part of the integrated fault diagnosis model is completed, the fault variables in the process are identified. In this study, the RBC model is used to reconstruct and identify the fault variables, and finally, the process and sensor faults are distinguished in detail.

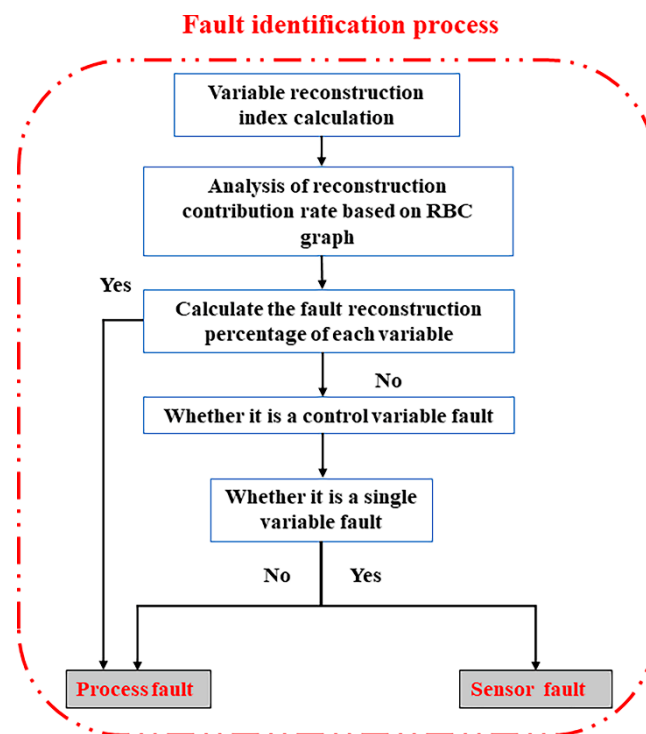
The fault identification process is shown in Figure 6.

(1) When the system detects a fault, it is used as the input to the fault identification part. The fault variable is first reconstructed using the principle of variable reconstruction, and the reconstruction index  $index^2(z_i)$  is calculated.

(2) Optimise the fault parameter  $f_i$  to capture most of the fault information.

(3) Calculate the contribution rate of variable reconstruction and calculate the percentage of the current variable contribution rate according to the multivariate statistical method. The final contribution percentage of the reconstruction variables is obtained.

According to the obtained variable reconstruction percentage, the current fault is distinguished as a process or sensor fault in detail. Process faults are generated internally by the system. They arise from a state deviation of the system control variables, which then spreads to other measured variables, and finally leads to faults. A sensor fault refers to the fault of the sensor of a measured variable, which causes false indications of the data, but the system itself does not fail. However, the operator mistakenly believes that the variable has a fault and adjusts the variable setting value, which will cause the system to fail and eventually evolve into a process fault. Therefore, it can be observed that the sensor fault is a single-variable data offset, whereas the process fault is a multi-variable coordination.



**Figure 6.** Fault identification process based on the RBC graph.

When reconstructing the contribution graph model to determine the fault variables, the possible fault variables are judged:

The RBC graph shows that a single variable exceeds the average fault contribution rate. Go to Step 2 for further judgment. If the RBC graph shows that multiple measured variables and a single control variable exceed the average fault contribution rate, it can be determined as a process fault. The corresponding control variable is the root cause of the fault.

For a situation that shows that a single variable has a fault, it is necessary to determine whether the variable is a controlled variable or a measured variable. If the fault variable is a controlled variable, it is determined that the fault is a process fault caused by a single controlled variable. If the fault variable is a measured variable, it is determined that the fault is a fault in the sensor where the variable is located, that is, a false indication of sensor data.

Based on the above detailed fault judgments, the integrated fault diagnosis model for chemical process sensors and process faults can distinguish the process or sensor faults that have occurred, even on the basis of detecting and identifying faults. It provides a technical basis for subsequent operations and helps prevent subsequent faults caused by maloperation.

#### 4. Case Studies

##### 4.1. Tennessee Eastman Process

The TE process was created by Downs and Vogel in 1993 [37] and is widely cited as a benchmark for studies in control and fault diagnosis. The flow chart of the TE process is presented in Figure 7.

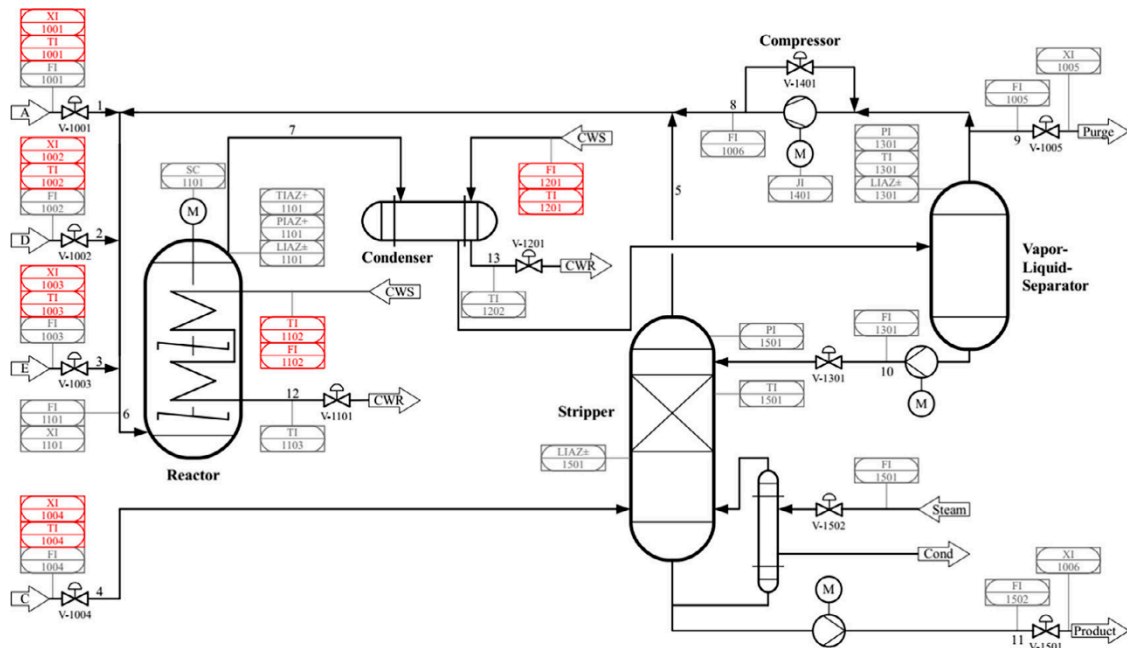


Figure 7. The Tennessee Eastman (TE) process.

The TE process includes 41 measurement variables and 11 manipulation variables. There is a strong correlation between 11 manipulation variables and their related measurement variables, while the correlation between measurement variables is relatively weak. Therefore, this process can be used as the basic process to study the correlation between variables. It includes a normal state and 21 fault states. Each state contains the training data and test data. Among them, there are 960 test data. The fault is introduced in the 160th sample. There are 21 faults in the TE process, of which 16 are known, and the remaining 5 are unknown. Fault modes 1–7 are caused by abnormal steps of some process variables, faults 8–12 are caused by random changes in some process variables; fault 13 is due to changes in reaction dynamics; and faults 14, 15, and 21 are caused by the valve being fixed in a specific position. The 21 faults are listed in Table 1.

For the TE process data, several different faults are set to distinguish the process and sensor faults. Before fault diagnosis, the basic parameters of the DKPCA model are set. The number of main components is determined by the standard that the CPV > 85% determine. The kernel width of the radial basis function was set to 800, and the confidence level was set to 95%.

Before distinguishing sensor and process faults, the advantages of the proposed model in fault detection are expressed. The number of principal components obtained by the DKPCA method is 28 for 52 variables of TE process, and the number of principal components obtained by proposed model is 26. It can be seen that the principal component obtained by proposed model includes more effective information, while there is invalid information in the principal component obtained by DKPCA method. The proposed model is more accurate in information acquisition.

In order to further quantify the accuracy and low detection delay of the proposed model, we measure its monitoring performance by fault detection rate (FDR) and time delay (TD). The Equations of FDR and TD are as follows:

$$FDR = \frac{(\text{control chart} > \text{control limit} | F \neq 0)}{\text{pre-set fault}} \times 100\% \quad (26)$$

$$TD = t_d - t_0 \quad (27)$$

where  $t_d$  is the fault detection time, and  $t_0$  is the fault occurrence time.

**Table 1.** Faults for the Tennessee Eastman (TE) process.

Fault Number	Description	Type
01	A/C feed ratio, B composition constant (Stream4)	Step
02	B composition, A/C ratio constant (Stream4)	Step
03	D feed temperature (Stream2)	Step
04	Reactor cooling water inlet temperature	Step
05	Condenser cooling water inlet temperature	Step
06	A feed loss (Stream1)	Step
07	C header pressure loss—reduced availability (Stream4)	Step
08	A, B, C feed composition (Stream4)	Random
09	D feed temperature (Stream2)	Random
10	C feed temperature (Stream4)	Random
11	Reactor cooling water inlet temperature	Random
12	Condenser cooling water inlet temperature	Random
13	Reaction kinetics	Slow drift
14	Reactor cooling water valve	Sticking
15	Condenser cooling water valve	Sticking
16–20	Unknown	Unknown
21	Valve position constant (Stream 4)	Constant position

The comparison results of FDR and TD statistics between CTA-DKPCA method and other methods are shown in Table 2

**Table 2.** Comparison of the FDR and TD of KPCA, 2-CLASS SVM [34], and the methods of this study in the TE process ((A) KPCA method, (B) 2-CLASS SVM method [38], (C) This study).

Fault No	KPCA		2-CLASS SVM		This Study	
	FDR (%)	TD (s)	FDR (%)	TD (s)	FDR (%)	TD (s)
1	83.0	20.0	<b>99.9</b>	6.0	99.2	<b>3.0</b>
2	90.0	78.0	97.8	57.0	<b>98.5</b>	<b>5.0</b>
4	75.0	26.0	100.0	3.0	<b>100.0</b>	<b>3.0</b>
5	80.0	18.0	99.9	6.0	<b>100.0</b>	<b>4.0</b>
6	77.0	10.0	<b>100.0</b>	3.0	98.4	<b>3.0</b>
7	75.0	14.0	<b>100.0</b>	3.0	99.1	<b>3.0</b>
8	72.0	112.0	95.8	60.0	<b>100.0</b>	<b>10.0</b>
10	68.0	40.0	85.8	12.0	<b>100.0</b>	<b>6.0</b>
11	81.0	25.0	<b>96.6</b>	3.0	92.0	<b>3.0</b>
12	59.0	36.0	<b>100.0</b>	3.0	99.0	<b>2.0</b>
13	75.0	259.0	91.9	153.0	<b>92.0</b>	<b>30.0</b>
14	55.0	21.0	<b>100.0</b>	3.0	99.5	<b>3.0</b>
16	66.0	29.0	<b>96.9</b>	<b>3.0</b>	91.2	4.0
17	71.0	99.0	92.9	72.0	<b>98.9</b>	<b>3.0</b>
18	81.0	378.0	90.0	231.0	<b>93.5</b>	<b>4.0</b>
19	52.0	45.0	88.5	3.0	<b>92.5</b>	<b>3.0</b>
20	70.0	77.0	85.0	45.0	<b>91.5</b>	<b>5.0</b>
21	51.0	12.0	100.0	3.0	<b>100.0</b>	<b>3.0</b>
<b>Mean</b>	71.2	71.61	95.6	37.17	<b>96.96</b>	<b>5.39</b>

The results of TE process fault detection show that the proposed model improves the accuracy of fault detection and the delay of fault detection. After proving the advantages of the model for fault detection, the proposed integrated fault diagnosis framework is used to divide the possible faults in industry in detail.

This study sets four different types of fault situations to distinguish these faults as process faults or sensor faults. The selection and setting of the cases are listed in Table 3.

**Table 3.** Four faults in TE process.

Case Number	Fault Situation	Description
1	Single variable data drift	Variable 1, data points 0–960: set the drift ratio of 0.1325 to the data
2	Large-scale jitter of process data	Variable 8, data points 150–250, 400–550, and 700–850: set jitter up by 50%
3	Single variable data step	Variable 3, set a step fault at data point 300
4	Actual fault	TE process fault 1

The first case introduces a drift fault with a drift ratio of 0.1325 at the 0 data point of variable 1: material A flow. The second case is in variable 8: in the ranges of 150–250, 400–550, and 700–850, the reactor liquid level is increased by 50%. The third case is the fault to introduce the data step at the 300th data point of the variable 3: E logistics flow. The last case is TE process simulation fault 1. Use the integrated fault diagnosis framework to diagnose the above 4 types of faults. The result of the integrated fault diagnosis is shown in Figure 5.

Figure 8a–d shows the fault detection results of the CTA-DKPCA model in the principal component subspace ( $T^2$ ) and residual subspace (SPE), respectively, where (e) is the fault identification results of principal component subspace and residual subspace based on RBC. The  $T^2$  statistic represents the macro situation of the system, that is, when there is a fault, the system can reflect the observable fault. SPE statistics represent the deviation from the normal situation in the system, that is, when the deviation occurs in the system, the normal error situation will be displayed in SPE statistics. So we can see that for the detection sensitivity of the residual subspace, the residual subspace represented by the SPE statistic is higher than the principal subspace represented by  $T^2$  statistic. For the contribution rate of the two subspaces, the principal component space shows the macro fault condition of the system itself, and the residual subspace shows the error of the system, so the  $T^2$  contribution rate is less than the SPE contribution rate. The error variable is observed through the SPE contribution rate, and then the  $T^2$  contribution rate is determined as the main macro fault variable of the system.

According to the integrated fault diagnosis framework, we can obtain the synergetic characteristics of the process fault accompanying variables. The variables of sensor fault are independent of each other. The process fault caused by the fault of the control variable causes the related measurement variables to fail together. Owing to the false indication of the sensor data, the sensor fault has no transferability between the sensors, and the system remains normal.

When the drift fault ratio of case 1 is set to be small, the sensor monitoring has slow data drift, and it is determined that a fault has occurred when the tolerance limit of the sensor is reached. Therefore, it takes longer to detect the system fault with the principal component space  $T^2$  statistic than with the residual subspace SPE statistic to detect the error in the system. From Figure 8e, it can be judged that only when variable 1 exceeds the average fault contribution of the 52 variables, and variable 1 is a measured variable, it is determined that case 1 is a sensor fault with unreliable sensor data based on the fault diagnosis result.

For case 2, the set system fluctuates. The integrated fault diagnosis result shows that the  $T^2$  statistic does not exceed the limit, the principal component space has no fault, and the residual subspace represented by SPE detects the fault in the fluctuation interval. Therefore, it shows that there is an error in the system. The fault identification result also shows that only variable 8 exceeds the control limit, and variable 8 is a measured variable. Therefore, it can be determined that case 2 is a false indication of data fluctuation of the sensor where variable 8 is located. The system remains normal.

Case 3 is a step fault, which is an instantaneous fault, and a single variable in the system is instantly amplified. The existence of a step fault can be detected immediately, and the fault identification result shows that the variable 3 single variable exceeds the limit. Other variables are normal. Thus, it is determined that the sensor where variable 3 is located has a transient fault, which generates the false indication of the data.

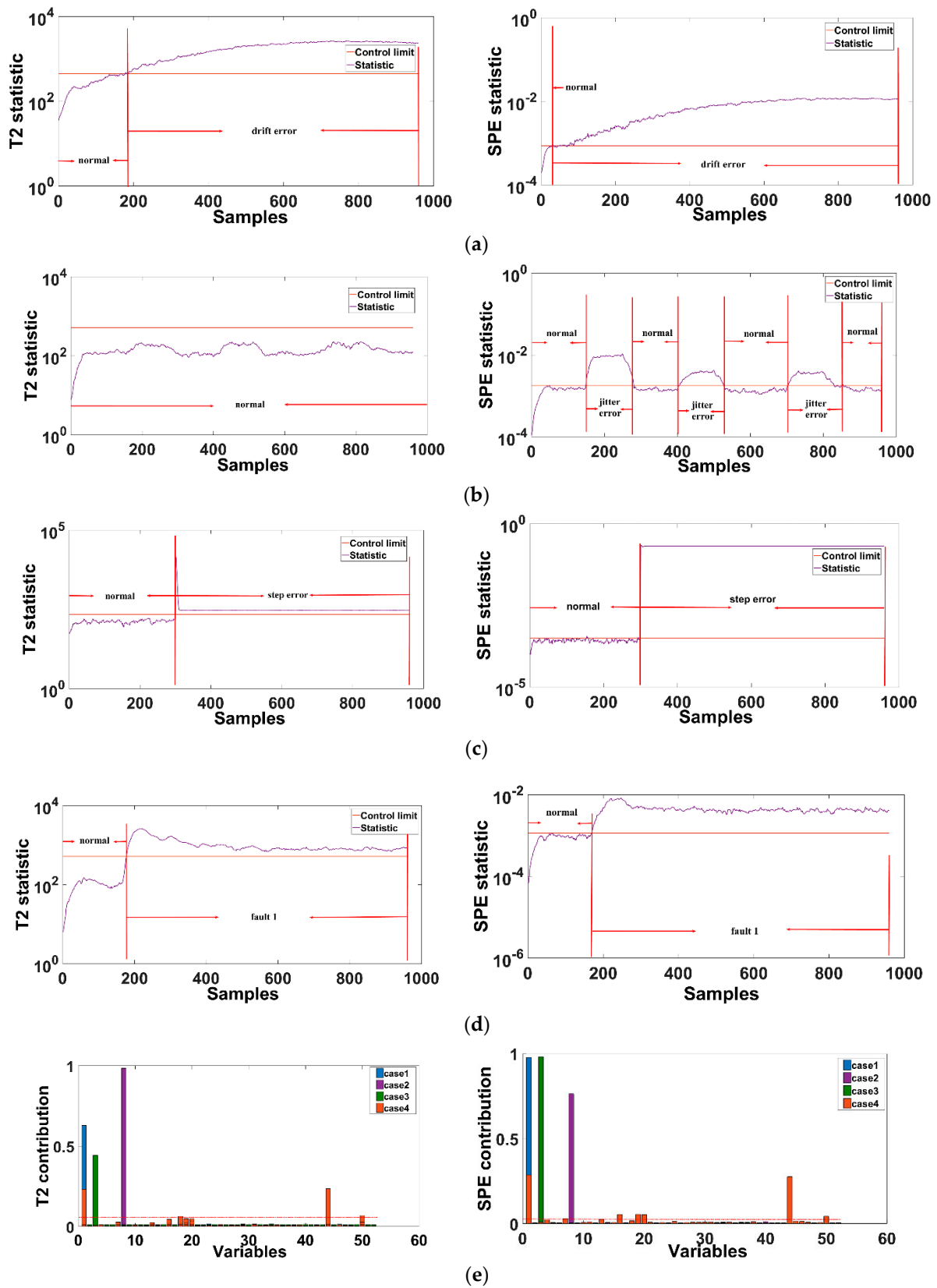


Figure 8. TE process fault diagnosis result. ((a)—case 1 fault detection result, (b)—case 2 fault detection result, (c)—case 3 fault detection result, (d)—case 4 fault detection result, (e)—fault identification result).



Case 4 is the simulation fault 1 of the TE process. The fault diagnosis results show that both  $T^2$  and SPE have detected the fault. The fault identification result shows that variables 1 and 44 exceed the average fault contribution rate. Variable 44 is the control variable: material A flow rate. Hence, it can be determined that case 4 is a process fault.

In addition to distinguishing the fault types of the system, the accuracy and speed of fault detection is also an important part of the fault diagnosis framework. Therefore, this study compares the FDR and TD of the four cases of the TE process with other PCA-derived algorithms. The results are presented in Table 4.

**Table 4.** Comparison with PCA, KPCA, DKPCA in FDR and TD.

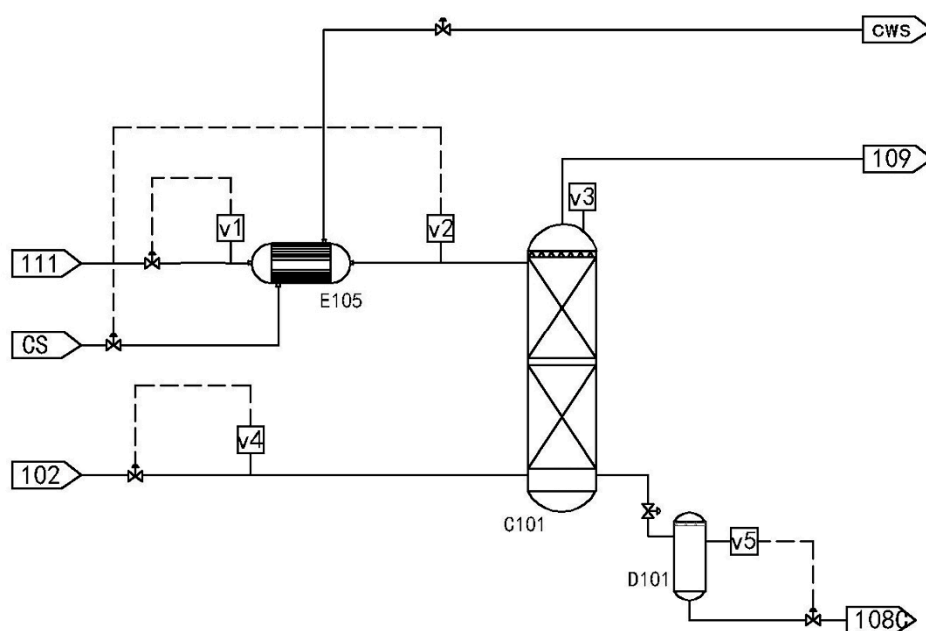
Case No	PCA		KPCA		DKPCA		This Study	
	FDR (%)	TD (s)	FDR (%)	TD (s)	FDR (%)	TD (s)	FDR (%)	TD (s)
1	48.6	278.0	67.6	55.0	72.5	50.0	79.9	25.0
2	16.5	125.0	78.5	65.0	89.2	25.0	91.3	5.0
3	83.0	12.0	95.0	10.0	98.0	2.0	100	0.0
4	31.0	35.0	83.0	20.0	91.0	11.0	99.2	3.0
Mean	44.78	112.5	81.03	37.5	87.68	22.0	92.6	8.25

The comparison results show that the fault detection method combined with CTA and DKPCA is far superior to other PCA algorithms in terms of FDR and TD. In terms of time delay, the fault detection algorithm proposed in this paper can detect faults faster than other algorithms. The average fault diagnosis rate and time delay are 92.6% and 8.25 s.

The results show that the integrated fault diagnosis framework proposed in this paper can clearly distinguish between process faults and sensor faults. The fault detection part has excellent performance in FDR and TD. Therefore, when a fault occurs, the operator can respond in time and guide the follow-up operation according to the fault identification part.

#### 4.2. Case Study of an Acid Gas Adsorption Process from Natural Gas

Methyldiethanolamine (MDEA) is often used as an absorbent in chemical processes to absorb acid gases. A flowchart of the typical absorption process is shown in Figure 9.



**Figure 9.** Acid gas adsorption process flow chart.

Stream 111 is the absorbent MDEA. It first exchanges heat with cooling water at room temperature through the heat exchanger E-105 and is then cooled to 21 °C, and then enters the absorption tower C-101 from the top. The raw material gas stream 102 enters the absorption tower C-101 from the bottom, flows counter currently with the absorbent MDEA, and absorbs acid gases (H<sub>2</sub>S and CO<sub>2</sub>) from the natural gas. The overhead gas of the absorption tower C-101 is natural gas containing a large amount of moisture, and then it enters the downstream dehydration system for further dehydration and purification to meet the national natural gas standards. The bottom product of the absorption tower C-101 is rich amine liquid containing acid gas. After heat exchange, the rich amine liquid enters the regeneration tower to resolve acid gas and regenerate the absorbent. According to the Piping and Instrument Diagram (P&ID) chart, the following variables: V1: absorbent MDEA volume flow rate, V2: absorption tower absorbent feed temperature, V3: absorption tower top pressure, V4: Natural gas feed flow, V5: Bottom liquid level height of absorption tower.

Three types of faults are set in the acid gas absorption process, including drift, stuck, and actual fault. The integrated fault diagnosis framework is used to distinguish the three types of faults in detail. Examples of the fault cases are listed in Table 5.

**Table 5.** Three faults in acid gas adsorption process.

Case Number	Fault Situation	Description
1	data drift	Set a slowly varying drift of the natural gas feed flow (V4) at data point 250
2	data stuck	Set a stuck of the absorption tower top pressure (V3) at data point 250
3	Fault 1	Heat exchanger inlet temperature rises to 31 °C at data point 300

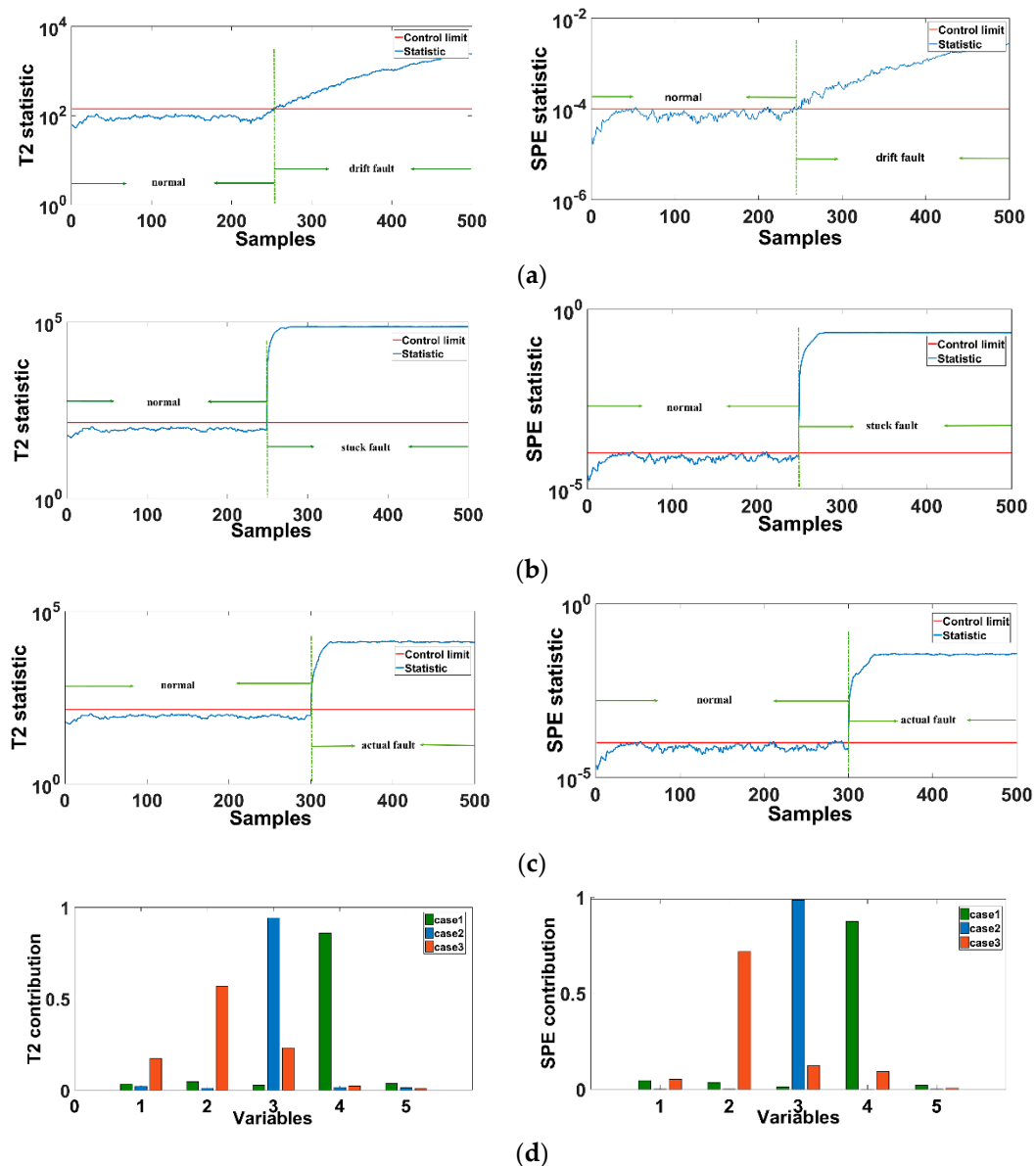
An acid gas absorption process is provided for the three different faults. In case 1, a graded drift of the gas flow rate is monitored at the sampling point 250. In case 2, the top pressure of the absorption tower is monitored and a stuck fault is set at the sampling point 250. In case 3, an actual fault is simulated by HYSYS, that is, the temperature of the heat exchanger inlet rises to 30 °C. The integrated fault diagnosis framework is used to detect and identify the three faults, and then distinguish the fault type as a process fault or a sensor fault. The fault diagnosis results are shown in Figure 7.

The diagnosis results for the acid gas absorption process can be obtained. For case 1, because the sensor sensitivity of the actual chemical process and the simulation process are different, the data drift of the actual process will be rapidly detected as the fault occurs, as shown in Figure 10d. The fault identification results show that variable 4 exceeds the average fault contribution rate of 20%, other variables are in a normal state, and variable 4 is the measurement variable of the condenser outlet temperature. Thus, case 1 is a sensor fault.

The actual chemical process often has a sudden stuck. The stuck is also an instantaneous fault; therefore, it will be detected instantaneously. The result of the fault variable also shows that the measured variable 3, absorption tower pressure, exceeds the average contribution rate, and other variables are normal. Hence, it is determined that case 2 is also a sensor fault.

For case 3, we introduced an actual fault of the acid gas absorption process and used the integrated fault diagnosis framework to detect and identify the fault. The detection results show that the fault can be accurately detected. The fault identification result shows that variables 1, 2, and 3 all exceed the control limit, indicating that the system has an offset condition. Therefore, it is determined that case 3 is a process fault.

In the acid gas absorption process, to determine the advantages of the proposed fault detection method in the actual chemical process, the two statistics of FDR and TD are compared with the PCA and its derivative algorithms. The comparison results are presented in Table 6.



**Figure 10.** Acid gas adsorption process fault diagnosis result ((a)—case 1 fault detection result, (b)—case 2 fault detection result, (c)—case 3 fault detection result, (d)—fault identification result).

**Table 6.** Comparison with PCA, KPCA, DKPCA in FDR and TD.

Case No	PCA		KPCA		DKPCA		This Study	
	FDR (%)	TD (s)	FDR (%)	TD (s)	FDR (%)	TD (s)	FDR (%)	TD (s)
1	56.1	24.0	77.0	15.0	90.1	5.0	99.2	2.0
2	78.9	12.0	89.0	8.0	98.2	4.0	100.0	0.0
3	65.5	34.0	80.0	19.0	89.5	6.0	100.0	0.0
Mean	66.83	23.33	82.0	14.0	92.6	5.0	99.73	0.67

The comparison results of various fault detection methods in the acid gas absorption case show that for actual chemical process faults, the fault detection part of the fault diagnosis framework proposed in this paper is far superior than the other algorithms in terms of fault diagnosis accuracy and speed. The FDR reached 99.73%. In cases 2 and 3 TD is 0, and the mean TD is 0.67 s, which proves that the proposed method has the best timeliness for fault detection.

Based on the above fault diagnosis results, it is shown that the CTA proposed in this paper optimises the speed and accuracy of fault detection and detects fault occurrence data points in time. The fault identification part based on the RBC graph can distinguish the chemical process sensor faults and process faults in detail. The group forms an integrated fault diagnosis framework suitable for modern chemical processes.

## 5. Conclusions

The distinction between sensor and process faults has always been an important part of chemical process fault diagnosis. This paper proposes an integrated fault diagnosis model that can effectively distinguish between sensor faults and process faults in modern chemical processes. First, a CTA is proposed to improve the calculation speed and accuracy of the model. The proposed algorithm is combined with the DKPCA for fault detection, and it is then combined with the RBC graph model to diagnose fault variables and comprehensively judge the process and sensor faults.

This study used a CTA combined with a DKPCA fault detection model to detect 18 faults in the TE process and verified its advantages in fault detection. Then, we selected four common fault types in the chemical process, including data drift, jitter, step, and actual fault. The fault detection model and the RBC fault diagnosis model were combined to distinguish the four types of faults as process faults or sensor faults in detail. Finally, based on variable correlation and fault transfer characteristics, the data drift, jitter, and step faults were determined as sensor faults, whereas the actual simulated fault was identified as process fault.

Finally, the integrated fault diagnosis model was applied to the acid gas absorption process to verify the effectiveness of the model in an actual chemical process. For the acid gas absorption process, three types of faults, namely, data drift, stuck, and actual faults, were selected to distinguish the fault types through the integrated fault diagnosis model. The results showed that data drift and sticking were sensor faults, and the actual fault was determined as the process fault. Based on the above results, this fault classification framework provides a strong foundation for the safety of chemical processes and ideas for follow-up research on chemical process fault diagnosis.

**Author Contributions:** Conceptualization, J.Z.; methodology, J.Z.; software, J.Z.; validation, J.Z., W.L. and Y.D.; writing—original draft preparation, J.Z.; writing—review and editing, W.L. and Y.D.; visualization, J.Z.; All authors have read and agreed to the published version of the manuscript.

**Funding:** The authors gratefully acknowledge the financial support from the National Natural Science Foundation of China (21706220).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data available on request due to restrictions eg privacy or ethical.

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

## References

1. Park, Y.J.; Fan, S.K.S.; Hsu, C.Y. A Review on Fault Detection and Process Diagnostics in Industrial Processes. *Processes* **2020**, *8*, 1123. [[CrossRef](#)]
2. Fan, S.-K.S.; Chang, Y.-J. An Integrated Advanced Process Control Framework Using Run-to-Run Control, Virtual Metrology and Fault Detection. *J. Process Control* **2013**, *23*, 933–942. [[CrossRef](#)]
3. Fei, H.; Chaojun, W. Fault Detection and Root Cause Analysis of a Batch Process via Novel Nonlinear Dissimilarity and Comparative Granger Causality Analysis. *Ind. Eng. Chem. Res.* **2019**, *58*, 21842–21854. [[CrossRef](#)]
4. He, F.; Wang, C.; Fan, S.-K.S. Nonlinear Fault Detection of Batch Processes Based on Functional Kernel Locality Preserving Projections. *Chemom. Intell. Lab. Syst.* **2018**, *183*, 79–89. [[CrossRef](#)]
5. Ahlam, M.; Madjid, F. Sensor and Component Fault Detection and Diagnosis for Hydraulic Machinery Integrating LSTM Autoencoder Detector and Diagnostic Classifiers. *Sensors* **2021**, *21*, 433.
6. Amin, M.T.; Khan, F.; Imtiaz, S.; Ahmed, S. Robust Process Monitoring Methodology for Detection and Diagnosis of Unobservable Faults. *Ind. Eng. Chem. Res.* **2019**, *58*, 19149–19165. [[CrossRef](#)]

7. Zhang, Z.M.; Li, X.; Jin, R. Data-Driven Anomaly Detection Approach for Time-Series Streaming Data. *Sensors* **2020**, *20*, 5646. [[CrossRef](#)]
8. Nguyen, N.P. Robust Fault Estimation Using the Intermediate Observer: Application to the Quadcopter. *Sensors* **2020**, *20*, 4917. [[CrossRef](#)]
9. Song, Q.; Wang, M. On the Accuracy of Fault Diagnosis for Rolling Element Bearings Using Improved DFA and Multi-Sensor Data Fusion Method. *Sensors* **2020**, *20*, 6465. [[CrossRef](#)]
10. Qin, S.J. Statistical Process Monitoring: Basics and Beyond. *J. Chemom.* **2003**, *17*, 480–502. [[CrossRef](#)]
11. Ding, S. Model-Based Fault Diagnosis in Dynamic Systems Using Identification Techniques. *Int. J. Robust Nonlinear Control* **2005**, *15*, 509–512. [[CrossRef](#)]
12. Li, W.; Li, H.; Gu, S.; Chen, T. Process Fault Diagnosis with Model-Based and Knowledge-Based Approaches: Advances and Opportunities. *Control Eng. Pract.* **2020**, *105*, 104637. [[CrossRef](#)]
13. Bounoua, W.; Benkara, A.B.; Kouadri, A.; Bakdi, A. Online Monitoring Scheme Using Principal Component Analysis through Kullback-Leibler Divergence Analysis Technique for Fault Detection. *Trans. Inst. Meas. Control* **2020**, *42*, 1225–1238. [[CrossRef](#)]
14. Yu, H.; Khan, F.; Vikram, G. An Alternative Formulation of PCA for Process Monitoring Using Distance Correlation. *Ind. Eng. Chem. Res.* **2016**, *55*, 656–669. [[CrossRef](#)]
15. Adedigba, S.A.; Khan, F.; Yang, M. Dynamic Failure Analysis of Process Systems Using Principal Component Analysis and Bayesian Network. *Ind. Eng. Chem. Res.* **2017**, *56*, 2094–2106. [[CrossRef](#)]
16. Ku, W.; Storer, R.H.; Georgakis, C. Disturbance Detection and Isolation by Dynamic Principal Component Analysis. *Chemom. Intell. Lab. Syst.* **1995**, *30*, 179–196. [[CrossRef](#)]
17. Lee, J.-M.; Yoo, C.; Choi, S.W.; Vanrolleghem, P.A.; Lee, I.-B. Nonlinear Process Monitoring Using Kernel Principal Component Analysis. *Chem. Eng. Sci.* **2004**, *59*, 223–234. [[CrossRef](#)]
18. Yang, Q.; Tian, F.; Wang, D. Nonlinear Dynamic Process Monitoring Based on Lifting Wavelets and Dynamic Kernel PCA. *Proc. World Congr. Intell. Control Autom.* **2010**, 5712–5716. [[CrossRef](#)]
19. Dunia, R.S.; Joe, Q. Joint Diagnosis of Process and Sensor Faults Using Principal Component Analysis. *Control Eng. Pract.* **1998**, *6*, 457–469. [[CrossRef](#)]
20. Wang, S.; Cui, J. Sensor-Fault Detection, Diagnosis and Estimation for Centrifugal Chiller Systems Using Principal-Component Analysis Method. *Appl. Energy* **2005**, *82*, 197–213. [[CrossRef](#)]
21. Li, G.; Hu, Y. Improved Sensor Fault Detection, Diagnosis and Estimation for Screw Chillers Using Density-Based Clustering and Principal Component Analysis. *Energy Build.* **2018**, *173*, 502–515. [[CrossRef](#)]
22. Chen, Q.; Kruger, U.; Meronk, M.; Leung, A. Synthesis of  $T^2$  and Q Statistics for Process Monitoring. *Control Eng. Pract.* **2004**, *12*, 745–755. [[CrossRef](#)]
23. Wang, H.; Song, Z.; Li, P. Fault Detection Behavior and Performance Analysis of Principal Component Analysis Based Process Monitoring Methods. *Ind. Eng. Chem. Res.* **2002**, *41*, 2455–2464. [[CrossRef](#)]
24. Kantaros, Y.; Zavlanos, M.M. A Temporal Logic Optimal Control Synthesis Algorithm for Large-Scale Multi-Robot Systems. *Int. J. Robot. Res.* **2020**, *39*, 812–836. [[CrossRef](#)]
25. Clarke, E.M.; Emerson, E.A. Design and Synthesis of Synchronization Skeletons Using Branching Time Temporal Logic. *Lect. Notes Comput. Sci.* **2008**, *5000*, 196–215.
26. Zhu, Q.X.; Luo, Y.; He, Y.L. Novel Distributed Alarm Visual Analysis Using Multicorrelation Block-Based PLS and Its Application to Online Root Cause Analysis. *Ind. Eng. Chem. Res.* **2019**, *58*, 20655–20666. [[CrossRef](#)]
27. Alcalá, C.F.; Qin, S.J. Reconstruction-Based Contribution for Process Monitoring with Kernel Principal Component Analysis. *Ind. Eng. Chem. Res.* **2010**, *49*, 7849–7857. [[CrossRef](#)]
28. Sun, H.; Zhang, S.; Zhao, C.; Gao, F.A. Sparse Reconstruction Strategy for Online Fault Diagnosis in Nonstationary Processes with No a Priori Fault Information. *Ind. Eng. Chem. Res.* **2017**, *56*, 6993–7008. [[CrossRef](#)]
29. Onel, M.; Kieslich, C.A.; Guzman, Y.A.; Floudas, C.A.; Pistikopoulos, E.N. Big Data Approach to Batch Process Monitoring: Simultaneous Fault Detection and Diagnosis Using Nonlinear Support Vector Machine-Based Feature Selection. *Comput. Chem. Eng.* **2018**, *116*, 503–520. [[CrossRef](#)]
30. Wang, H.; Chen, Y. A Robust Fault Detection and Diagnosis Strategy for Multiple Faults of VAV Air Handling Units. *Energy Build.* **2016**, *127*, 442–451. [[CrossRef](#)]
31. Wu, H.; Zhao, J. Fault Detection and Diagnosis Based on Transfer Learning for Multimode Chemical Processes. *Comput. Chem. Eng.* **2020**, *135*, 106731. [[CrossRef](#)]
32. Xu, C.; Zhao, S.; Liu, F. Sensor Fault Detection and Diagnosis in the Presence of Outliers. *Neurocomputing* **2019**, *349*, 156–163. [[CrossRef](#)]
33. Wang, Q.; Gao, Q.; Sun, G.; Ding, C. Double robust principal component analysis. *Neurocomputing* **2020**, *391*, 119–128. [[CrossRef](#)]
34. Kumar, A.; Bhattacharya, A.; Flores-Cerrillo, J. Data-driven process monitoring and fault analysis of reformer units in hydrogen plants: Industrial application and perspectives. *Comput. Chem. Eng.* **2020**, *136*, 106756. [[CrossRef](#)]
35. Liu, Z. Online Process Monitoring for Complex Systems with Dynamic Weighted Principal Component Analysis. *Chin. J. Chem. Eng.* **2016**, *6*, 775–786.
36. Maurya, M.R.; Rengaswamy, R.; Venkatasubramanian, V. Fault Diagnosis Using Dynamic Trend Analysis: A Review and Recent Developments. *Engineering Applications of Artificial Intelligence*. *Int. J. Intell. Real Time Autom.* **2007**, *20*, 133–146.

- 
37. Downs, J.J.; Vogel, E.F. A Plant-Wide Industrial Process Control Problem. *Comput. Chem. Eng.* **1993**, *17*, 245–255. [[CrossRef](#)]
  38. Onel, M.; Kieslich, C.N.; Pistikopoulos, E.N. A Nonlinear Support Vector Machine-Based Feature Selection Approach for Fault Detection and Diagnosis: Application to the Tennessee Eastman Process. *AIChE J.* **2019**, *65*, 992–1005. [[CrossRef](#)]