Hindawi
Journal of Control Science and Engineering
Volume 2017, Article ID 1812989, 8 pages
https://doi.org/10.1155/2017/1812989

Research Article

Subspace Method Aided Data-Driven Fault Detection Based on Principal Component Analysis

Lingling Ma and Xiangshun Li

Wuhan University of Technology, Wuhan, China

Correspondence should be addressed to Xiangshun Li; lixiangshun@whut.edu.cn

Received 17 June 2017; Revised 5 September 2017; Accepted 22 October 2017; Published 12 November 2017

Academic Editor: Abdul-Qayyum Khan

The model-based fault detection technique, which needs to identify the system models, has been well established. The objective of this paper is to develop an alternative procedure instead of identifying the system models. In this paper, subspace method aided data-driven fault detection based on principal component analysis (PCA) is proposed. The basic idea is to use PCA to identify the system observability matrices from input and output data and construct residual generators. The advantage of the proposed method is that we just need to identify the parameterized matrices related to residuals rather than the system models, which reduces the computational steps of the system. The proposed approach is illustrated by a simulation study on the Tennessee Eastman process.

1. Introduction

The safety, stability, and efficiency of dynamic systems have always been a matter of great concern in the field of complex industrial processes. Fault detection plays an extremely important role in improving the safety of processes and attracts more and more attention in the field of process monitoring systems.

During the past three decades, a model-based fault detection technique for linear time invariant (LTI) systems has been well established [1–4]. But in some complex systems and large-scale industries, it is difficult to obtain accurate mathematical models. With the development of the computer and information industry, Big Data has attracted more and more attention. Actually, industrial processes have large amounts of operating data which contain abundant information about the system. Subspace model identification which uses process data to identify system models comes into being under this kind of background. There are several methods of subspace algorithms, such as MOESP [5], N4SID [6], and CVA [7]. In a typical SMI, the identification procedure comprises two steps. Firstly, the extended observability matrix Γ_f and a block triangular Toeplitz matrix H_f are identified from input and output data. Then, A, B, C, and D are calculated from the identified observability matrix and the Toeplitz matrix. Fault detection using this method usually comprises two steps: (1) system identification and (2) model-based fault detection.

Residual generation and residual evaluation are essential for model-based fault detection. Fault detection can be achieved if residuals are obtained. Parity space approach (PSA) [8] is the simplest and most widely used among model-based fault detection methods. We noticed that there is a close bond between the two methods by comparing PSA and SMI. The residual could be calculated by some parameterized matrices related to Γ_f and H_f. So, it is feasible that we only need to use the first step of SMI to identify parameterized matrices instead of the system models, which reduces the calculation steps. To the best of our knowledge, Ding and his coworkers subtly associated SMI with fault detection and proposed a subspace aided approach [9–11]. But most of these SMI methods are biased under the errors-invariables (EIV) situation. Wang and Qin [12] proposed a subspace identification approach based on PCA that gives consistent model estimates under the EIV situation. Inspired by their work, we propose subspace method aided data-driven fault detection based on PCA in this paper. Figure 1 shows the difference between the proposed approach and the classic model-based approach.

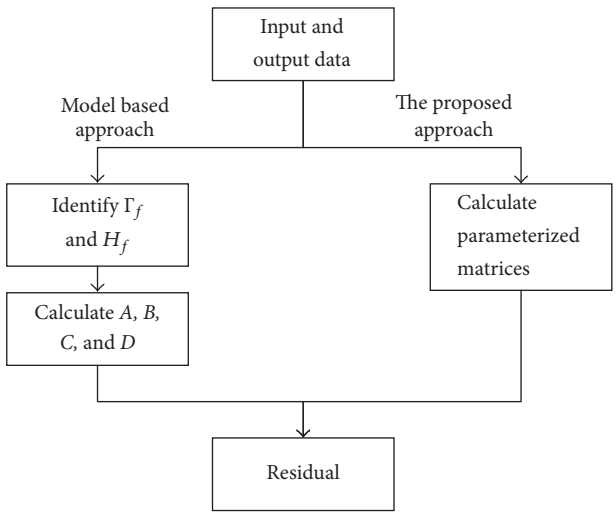

Figure 1: Model-based approach and the proposed approach.

In this paper, the main contribution of this proposed approach lies in the direct identification of the parameterized matrices by PCA. As long as these matrices related to residuals are identified, we can construct residual generators. The proposed approach is illustrated by a simulation study on the Tennessee Eastman process.

The rest of the paper is organized as follows. Section 2 gives the preliminaries and problem formulation. The identification of parameterized matrices is presented in Section 3. Section 4 illustrates a simulation study on the Tennessee Eastman process. Finally, the conclusions are presented in Section 5.

2. Preliminaries and Problem Formulation

2.1. Process Descriptions. The form of the state space representation of a discrete time LTI system is given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu^*(k) + p(k) \\ y^*(k) &= Cx(k) + Du^*(k), \end{aligned} \tag{1}$$

where $x \in R^n$, $u^*(k) \in R^l$, $y^*(k) \in R^m$, and $p(k) \in R^n$ denote the vectors of the state variables, noise-free inputs, noise-free outputs, and process noise, respectively. The available input measurements $u(k)$ and output measurements $y(k)$ are described by [13]

$$\begin{aligned} u(k) &= u^*(k) + o(k) \\ y(k) &= y^*(k) + v(k), \end{aligned} \tag{2}$$

where $o(k) \in R^l$ and $v(k) \in R^m$ represent the input and output noise. The following assumptions are introduced:

(i) The system is observable and controllable.

(ii) $o(k)$, $v(k)$, and $p(k)$ are assumed to be zero-mean, normally distributed white noise; that is,

$$\begin{aligned} &E\left\{\begin{bmatrix} p(k) \\ o(k) \\ v(k) \end{bmatrix}\begin{bmatrix} p(j)^T & o(j)^T & v(j)^T \end{bmatrix}\right\} \\ &= \begin{bmatrix} R_{pp} & R_{po} & R_{pv} \\ R_{po}^T & R_{oo} & R_{ov} \\ R_{pv}^T & R_{ov}^T & R_{vv} \end{bmatrix}\delta_{kj} \quad \delta_{kj} = \begin{cases} 1, & k = j \\ 0, & k \neq j. \end{cases} \end{aligned} \tag{3}$$

(iii) All the noise is assumed to be independent of the past noise-free input $u^*(k)$ and initial state $x(1)$. They satisfy

$$E\left\{u^*(k)\begin{bmatrix} p(j) \\ o(j) \\ v(j) \end{bmatrix}^T\right\} = 0, \quad \text{for } j \geqslant k. \tag{4}$$

(iv) System matrices A, B, C, and D and system order n are unknown.

We define the following vectors and Hankel matrices. The subscripts p and f represent the past and the future, where $p \geqslant f \geqslant n$.

$$u_p(k) = \begin{bmatrix} u(k-p) \\ u(k-p+1) \\ \vdots \\ u(k-1) \end{bmatrix} \in R^{lp},$$

$$u_f(k) = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+f-1) \end{bmatrix} \in R^{lf},$$

$$y_p(k) = \begin{bmatrix} y(k-p) \\ y(k-p+1) \\ \vdots \\ y(k-1) \end{bmatrix} \in R^{mp},$$

$$y_f(k) = \begin{bmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+f-1) \end{bmatrix} \in R^{mf},$$

$$
\begin{aligned}
U_p &= \left[u_p(k)\, u_p(k+1) \cdots u_p(k+N-1)\right] \in R^{lp\times N},\\
U_f &= \left[u_f(k)\, u_f(k+1) \cdots u_f(k+N-1)\right]\\
&\quad \in R^{lf\times N},\\
Y_p &= \left[y_p(k)\, y_p(k+1) \cdots y_p(k+N-1)\right]\\
&\quad \in R^{mp\times N},\\
Y_f &= \left[y_f(k)\, y_f(k+1) \cdots y_f(k+N-1)\right]\\
&\quad \in R^{mf\times N},\\
X(k) &= \left[x(k)\, x(k+1) \cdots x(k+N-1)\right] \in R^{n\times N},\\
Z_p &= \begin{bmatrix} Y_p \\ U_p \end{bmatrix} \in R^{(lp+mp)\times N},\\
Z_f &= \begin{bmatrix} Y_f \\ U_f \end{bmatrix} \in R^{(lf+mf)\times N}.
\end{aligned}
\tag{5}
$$

By iterating (1) and (2), we can get

$$
\begin{aligned}
y_f(k) = \Gamma_f x(k) + H_f u_f(k) - H_f v_f(k) + G_f p_f(k)\\
+ o_f(k).
\end{aligned}
\tag{6}
$$

The vectors $v_f(k) \in R^{lf}$, $o_f(k) \in R^{mf}$, $p_f(k) \in R^{nf}$, and $u_f(k) \in R^{lf}$ have similar structures to $y_f(k)$. If we use Hankel matrix instead of the vectors to describe the system, (6) can be written as

$$
\begin{aligned}
Y_f &= \Gamma_f X(k) + H_f U_f + E_f\\
E_f &= -H_f V_f + G_f P_f + O_f.
\end{aligned}
\tag{7}
$$

The matrices V_f and O_f are defined similarly to Y_f.

$$
\begin{aligned}
\Gamma_f &= \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix} \in R^{mf\times n},\\
H_f &= \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ CA^{f-2}B & CA^{f-3}B & \cdots & D \end{bmatrix} \in R^{mf\times lf},\\
G_f &= \begin{bmatrix} 0 & 0 & \cdots & 0 \\ C & 0 & \cdots & 0 \\ \vdots & \cdots & \ddots & \vdots \\ CA^{f-2} & CA^{f-3} & \cdots & 0 \end{bmatrix} \in R^{mf\times nf}.
\end{aligned}
\tag{8}
$$

2.2. Residual Generation and Evaluation. Residual generation and residual evaluation are essential to design a fault detection system. The main idea of this paper is to generate residual from input and output data. How to generate and evaluate residual is introduced in the following section.

Moving $H_f U_f$ to the left of the equation, (7) can be rewritten as

$$
Y_f - H_f U_f = \Gamma_f X(k) + E_f. \tag{9}
$$

Taking $\Gamma_f^{\perp} \in R^{mf\times(mf-n)}$ as the orthogonal complement of Γ_f, we can obtain

$$
\left(\Gamma_f^{\perp}\right)^T \Gamma_f = 0. \tag{10}
$$

Since $X(k)$ is unknown, we multiply $(\Gamma_f^{\perp})^T$ on both sides of the equation to eliminate the effect of the unknown state [14]. Then, we can get

$$
\left(\Gamma_f^{\perp}\right)^T \left(Y_f - H_f U_f\right) = \left(\Gamma_f^{\perp}\right)^T E_f. \tag{11}
$$

So, the residual signal

$$
R = \left(\Gamma_f^{\perp}\right)^T \left(Y_f - H_f U_f\right) = \left(\Gamma_f^{\perp}\right)^T Y_f - \left(\Gamma_f^{\perp}\right)^T H_f U_f. \tag{12}
$$

We noticed that the residual is closely related to the extended observability Γ_f and a block triangular Toeplitz matrix H_f. Therefore, if we want to obtain the residual, the key step is to identify $(\Gamma_f^{\perp})^T$ and $-(\Gamma_f^{\perp})^T H_f$. In Section 3, how to identify these parameterized matrices will be introduced in detail.

Once these parameterized matrices are identified, we can get residual signal. Then, the fault detection can be completed by residual evaluation. Let $r(k)$ be the residual vector at the instant k, defining the testing statistic [15]

$$
J = r(k)^T \Sigma_{\text{res}}^{-1} r(k) \quad \Sigma_{\text{res}} = \frac{RR^T}{(N-1)}, \tag{13}
$$

whereΣ_{res} is the covariance matrix of the residual matrix in the fault-free case. Note that when there is no fault, $J \sim \chi^2(h)$, where $\chi^2(h)$ is the chi-squared distribution with h degrees of freedom. We set $J_{\text{th}} = \chi_\alpha^2(h)$ as threshold and define the detection logic as

$$
J = \begin{cases} < J_{\text{th}}, & \text{no fault} \\ > J_{\text{th}}, & \text{a fault is detected.} \end{cases} \tag{14}
$$

Besides the chi-squared distribution, kernel density estimation is also utilized to determine the threshold [16]. It is assumed that the measurements follow a Gaussian distribution. The threshold J_{th} can be calculated by $P(J < J_{\text{th}}(\alpha)) = \alpha$, where α is confidence level.

$$
\begin{aligned}
P(x < a) &= \int_{-\infty}^{a} p(x)\, dx = \alpha\\
p(x) &= \frac{1}{Nh} \sum_{k=1}^{N} K\left(\frac{x - x_k}{h}\right),
\end{aligned}
\tag{15}
$$

where x_k is the kth sample of x, N is the number of samples, h is a smoothing parameter, and $K(\cdot)$ is a kernel function.

Given a proper confidence level, we can get the threshold J_{th}. In this paper, we set $\alpha = 0.95$. After the statistics and threshold are deduced, the fault detection can be realized according to the above detection logic.

3. The Identification of Parameterized Matrices

Based on the knowledge that has been introduced in the previous sections, we know that the identification of matrices related to residual is the key step in this approach. In this section, we discuss the issue of how to identify these matrices. Wang and Qin proposed a subspace identification approach based on PCA. We will briefly introduce it. When the process is corrupted by process noise and measurement simultaneously, they use instrumental variables Z_p to remove the noise, because the past data Z_p is uncorrelated with future noise. According to these assumptions, we can get

$$\begin{aligned}&\lim_{N\to\infty}\frac{1}{N}E_f Z_p^T\\ &\quad=\lim_{N\to\infty}\frac{1}{N}\left(-H_f V_f Z_p^T+G_f P_f Z_p^T+O_f Z_p^T\right)=0.\end{aligned}\tag{16}$$

If we utilize Z_f to represent residual instead of Y_f and U_f, (11) can be expressed as

$$\left(\Gamma_f^{\perp}\right)^T\left[I\mid -H_f\right]Z_f=\left(\Gamma_f^{\perp}\right)^T E_f.\tag{17}$$

Multiplying $(1/N)Z_p^T$ on both sides of the equation, we can get

$$\frac{1}{N}\left(\Gamma_f^{\perp}\right)^T\left[I\mid -H_f\right]Z_f Z_p^T=\frac{1}{N}\left(\Gamma_f^{\perp}\right)^T E_f Z_p^T.\tag{18}$$

From (16), we know

$$\frac{1}{N}\left(\Gamma_f^{\perp}\right)^T\left[I\mid -H_f\right]Z_f Z_p^T=0.\tag{19}$$

Equation (19) indicates that $(1/N)Z_f Z_p^T$ has zero scores. From the rank condition

$$\operatorname{rank}\left(\lim_{N\to\infty}\frac{1}{N}E_f Z_p^T\right)=lf+n,\tag{20}$$

we know that when $N\to\infty$, $(1/N)Z_f Z_p^T$ has $mf-n$ zero scores. Performing PCA on the process data,

$$\lim_{N\to\infty}\frac{1}{N}Z_f Z_p^T=PT^T+\widetilde{P}\widetilde{T}^T.\tag{21}$$

When $N\to\infty$,

$$\begin{bmatrix}\Gamma_f^{\perp}\\ -H_f^T\Gamma_f^{\perp}\end{bmatrix}=\widetilde{P}M,\quad \widetilde{P}=\begin{bmatrix}\widetilde{P}_y\\ \widetilde{P}_u\end{bmatrix},\tag{22}$$

where $M\in R^{(mf-n)(mf-n)}$ is a nonsingular matrix, $\widetilde{P}_y$ is the first mf rows of $\widetilde{P}$, and $\widetilde{P}_u$ is the last lf rows of $\widetilde{P}$.

(1) Generate Z_f and Z_p with fault-free data
(2) Perform PCA on $(1/N)Z_f Z_p^T$
(3) Calculate $\Gamma_f^{\perp}$ and $-H_f^T\Gamma_f^{\perp}$ according to (22)
(4) Construct residual R according to (12)
(5) Calculate $\Sigma_{\text{res}}=RR^T/(N-1)$ in the fault-free case
(6) Generate Y_p and U_p with on-line operating data
(7) Construct residual R according to (12), then calculate the testing statistic J by (13)
(8) Determine the threshold J_{th}
(9) Evaluate the residual according to the detection logic

Algorithm 1: SMI-PCA fault detection.

The nonsingular matrix M is unit matrix in our paper. The matrices $\Gamma_f^{\perp}$ and $-H_f^T\Gamma_f^{\perp}$ can be obtained according to (22), and then the residual can be generated. The algorithm from data to realize fault detection can be summarized as the steps shown in Algorithm 1.

4. Simulation Study on the Tennessee Eastman Process

In this section, we apply the proposed approach to Tennessee Eastman (TE) process. Three indices are used—fault detection rate (FDR), false alarm rate (FAR) [17], and fault detection time (FDT)—to demonstrate the efficiency of the proposed approach. The fault detection time is the first time instance with the testing statistic J above the thresholds J_{th}.

$$\begin{aligned}\text{FDR}&=\frac{\text{No. of samples}\left(J>J_{\text{th}}\mid f\neq 0\right)}{\text{total samples}\left(f\neq 0\right)}\times 100\\ \text{FAR}&=\frac{\text{No. of samples}\left(J>J_{\text{th}}\mid f=0\right)}{\text{total samples}\left(f=0\right)}\times 100.\end{aligned}\tag{23}$$

4.1. TE Process. TE process model is a realistic simulation program of a chemical process that is widely accepted as a benchmark for control monitoring studies [18]. The TE model that we used was downloaded from http://depts.washington.edu/control/LARRY/TE. The flow diagram of the process is shown in Figure 2. The process contains five major units named reactor, condenser, compressor, separator, and stripper.

There are 53 measurements, 12 of which are manipulated variables and 41 are process variables. The 12 manipulated variables are listed in Table 1. In this paper, the other 9 manipulated variables except for XMV(5), XMV(9), and XMV(12) are taken as inputs. The dimension of the TE dataset is very large. To overcome the curse of dimensionality, the 41 process variables are divided into eight blocks [19]. We choose the input feed block as outputs, which are shown in Table 2. Table 3 shows the 20 process faults of TE process.

In our simulation, these data are acquired in the case of Mode 1. The operating time is set to be 36 h. The total number of samples is $N=3601$. The faults are introduced at 8 h, which

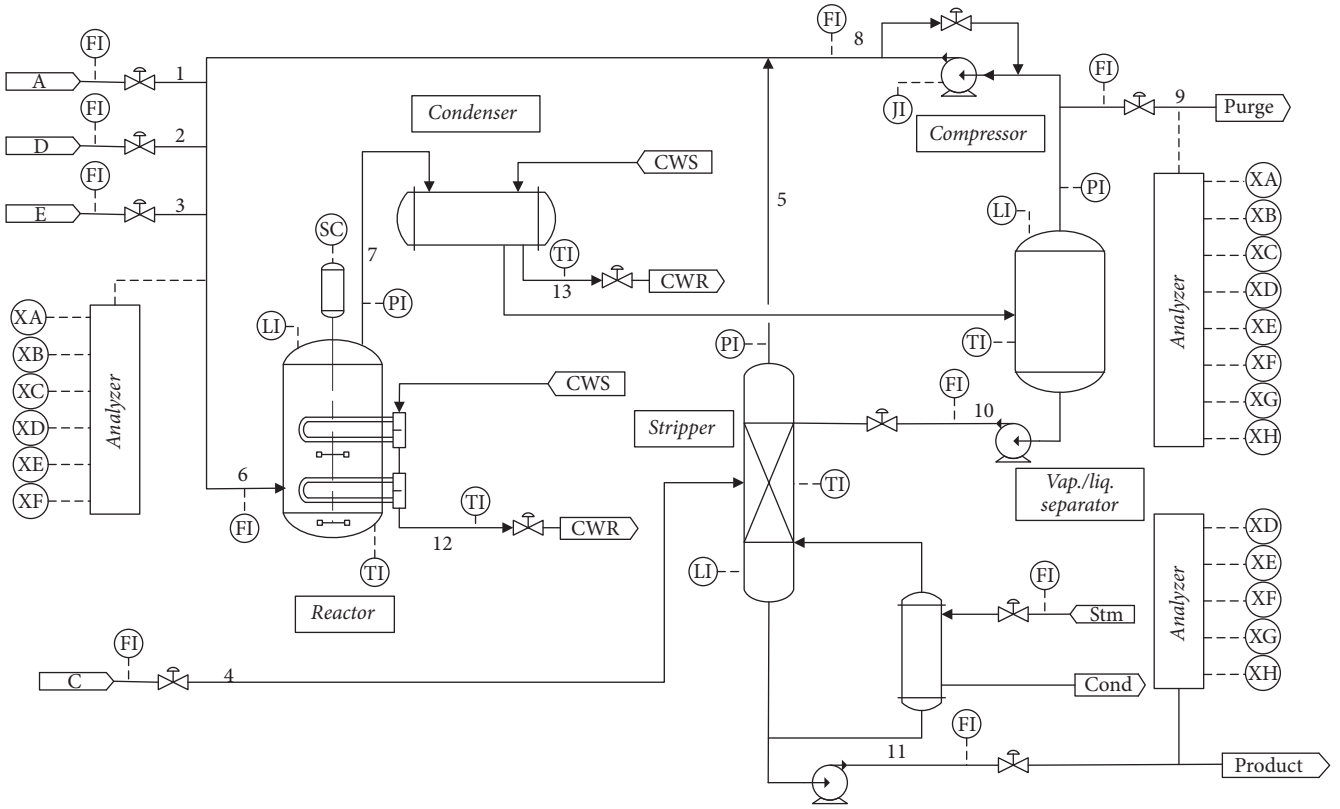

Figure 2: The Tennessee Eastman process.

Table 1: Process manipulated variables.

Variable name	Variable number
D feed flow	XMV(1)
E feed flow	XMV(2)
A feed flow	XMV(3)
A and C feed flow	XMV(4)
Compressor recycle valve	XMV(5)
Purge valve	XMV(6)
Separator pot liquid flow	XMV(7)
Stripper liquid product flow	XMV(8)
Stripper steam valve	XMV(9)
Reactor cooling water flow	XMV(10)
Condenser cooling water flow	XMV(11)
Agitator speed	XMV(12)

means that the fault occurs after 800 samples. p and f are both set as 10.

4.2. Simulation Results. The process is simulated with 20 different faults. For the sake of simplicity, we only take some typical faults to show. The first type of fault is step change in A and C feed ratio (IDV1). Figure 3 shows the testing statistic J based on residual vectors for input feed block. The blue line is the testing statistic J and the threshold J_{th} is shown by a red line. The second type of fault is a random variation change in A, B, and C feed composition (IDV8). The simulated result is

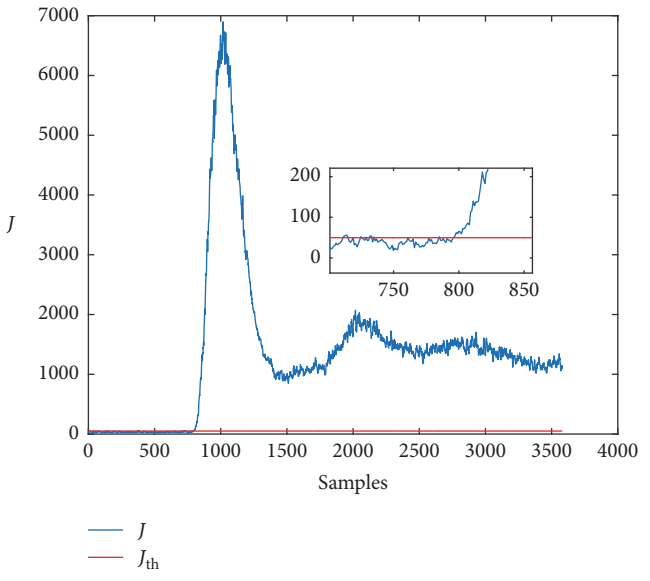

Figure 3: Detection of IDV(1).

shown in Figure 4. Figure 5 shows the testing statistic of the third type of fault which is a sticking fault in reactor cooling water valve (IDV14). The last fault (IDV17) shown in Figure 6 is an unknown fault. The type and process variables of fault are not known.

Table 2: Input feed block process measurements.

Variable name	Variable number
A feed (stream1)	XMEAS(1)
D feed (stream1)	XMEAS(2)
E feed (stream1)	XMEAS(3)
A and C feed	XMEAS(4)

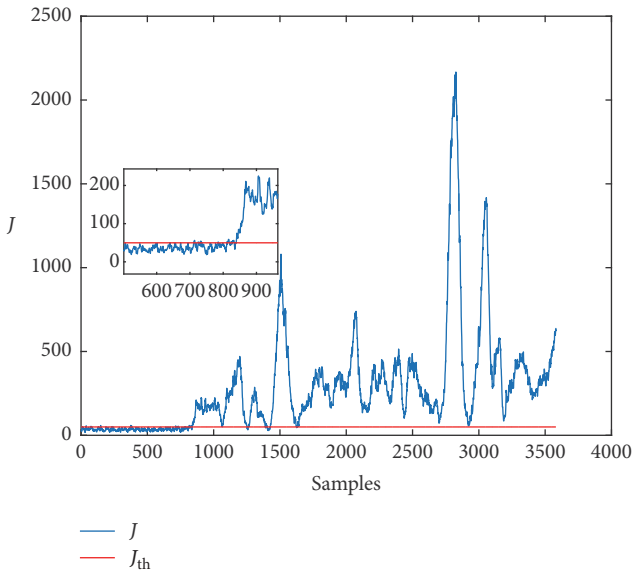

Figure 4: Detection of IDV(8).

Based on the fault detection method, if the testing statistics exceed the threshold, the fault is detected. We can see from Figures 3–6 that all the testing statistics J exceed the threshold J_{th} after 800 samples. The four faults have been detected.

In addition to the simulation results, the FDR, FAR, and FDT shown in Table 4 are also obtained to demonstrate the efficiency. Besides the comparison between SMI-PCA and SMI-PCA KDE, we also compare them with the classic PCA in order to better reflect the advantages of this proposed method. According to the study of Mahadevan and Shah [20], the FDR of T^2 and Q for PCA are listed in the table. The maximum fault detection rate has been highlighted in boldface. We can see that most of the fault detection rate of our proposed method is higher than PCA. The fault detection rate of using kernel density estimation to calculate the threshold is slightly higher than the chi-squared distribution. Most of the faults can be detected well and they have high FDRS and low FARS. But for some faults, that is, IDV(3), (5), (9), and (15-16), the efficiency of fault detection is very poor. This may be caused by a very small change in the variables, which is also a common problem in the TE process fault detection. There is no meaning to calculate the fault detection time if the faults cannot be detected, so the fault detection time is indicated with "$*$" in the table. The fault detection time of SMI-PCA and SMI-PCA KDE is the same because of their similar detection rate.

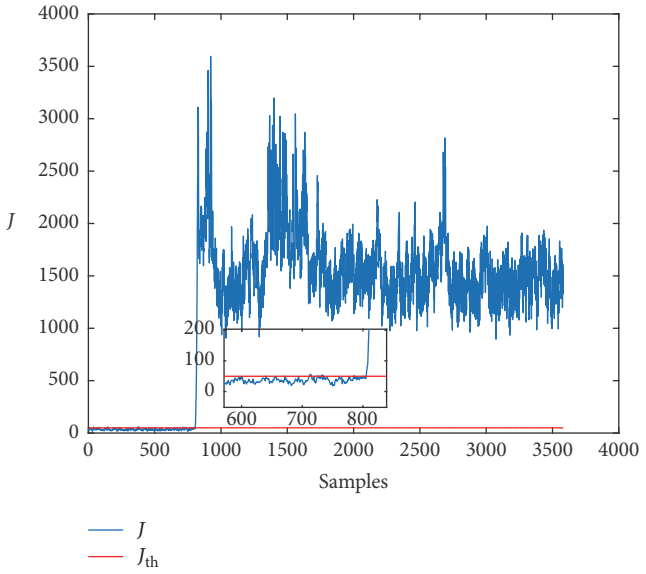

Figure 5: Detection of IDV(14).

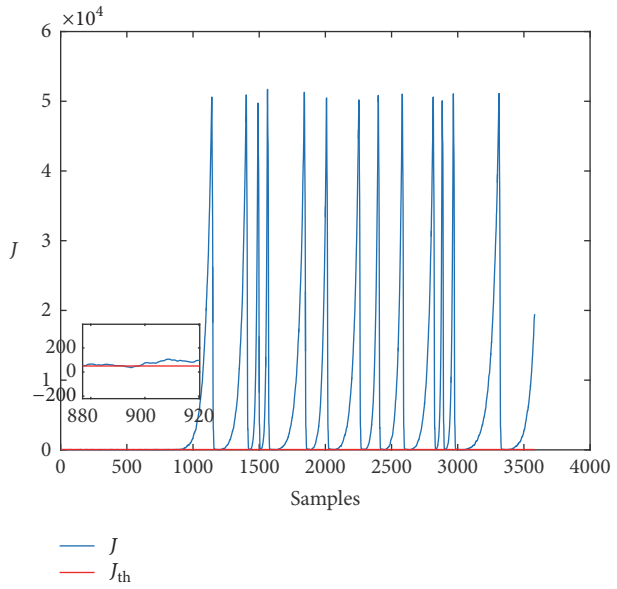

Figure 6: Detection of IDV(17).

5. Conclusions

In this paper, subspace method aided data-driven fault detection based on PCA has been presented. This method is to identify the parameterized matrices using PCA and construct residual generators with the input and output data. A simulation study on the TE process demonstrates the availability of this method. It indicates that the method proposed in this paper has better effects than PCA on fault detection rate and is suitable for linear systems which are observable and controllable. Moreover, the problem of threshold is also discussed in this paper. The fault detection rate of using kernel density estimation is slightly superior to the chi-squared distribution.

Table 3: Process faults.

Variable number	Process variable	Type
IDV(0)	Normal operation	Step
IDV(1)	A/C feed ratio, B composition constant	Step
IDV(2)	B composition, A/C ratio constant	Step
IDV(3)	D feed temperature	Step
IDV(4)	Reactor cooling water inlet temperature	Step
IDV(5)	Condenser cooling water inlet temperature	Step
IDV(6)	A feed loss	Step
IDV(7)	C header pressure loss-reduced availability	Step
IDV(8)	A, B, C feed composition	Random variation
IDV(9)	D feed temperature	Random variation
IDV(10)	C feed temperature	Random variation
IDV(11)	Reactor cooling water inlet temperature	Random variation
IDV(12)	Condenser cooling water inlet temperature	Random variation
IDV(13)	Reaction kinetics	Random variation
IDV(14)	Reactor cooling water valve	Sticking
IDV(15)	Condenser cooling water valve	Sticking
IDV(16)	Unknown	Unknown
IDV(17)	Unknown	Unknown
IDV(18)	Unknown	Unknown
IDV(19)	Unknown	Unknown
IDV(20)	Unknown	Unknown

Table 4: TE process: fault detection for test data (%).

Fault	PCA		SMI-PCA			SMI-PCA KDE		
	FDR(T^2)	FDR(Q)	FDR	FAR	FDT	FDR	FAR	FDT
IDV(1)	99.2	99.8	**100**	4.25	800	**100**	5.63	800
IDV(2)	98.0	98.6	99.46	3.88	800	**99.5**	5.25	800
IDV(3)	0	0	11.97	3.75	*	**14.52**	5.12	*
IDV(4)	4.4	96.2	**100**	4.62	800	**100**	6	800
IDV(5)	22.5	**25.4**	14.92	4.37	*	17.43	5.75	*
IDV(6)	98.9	100	**100**	4.62	800	**100**	6.12	800
IDV(7)	91.5	**100**	**100**	4.75	800	**100**	6.12	800
IDV(8)	96.6	97.6	97.70	3.62	811	**97.81**	5.12	811
IDV(9)	0	0	**12.4**	3.62	*	14.49	5.12	*
IDV(10)	33.4	34.1	81.56	3.62	811	**82.82**	5.12	811
IDV(11)	20.6	64.4	98.67	3.62	811	**98.78**	5.12	811
IDV(12)	97.1	**97.5**	43.35	3.62	811	46.76	5.12	811
IDV(13)	94.0	95.5	98.81	3.62	811	**98.85**	5.12	811
IDV(14)	84.2	**100**	99.82	**3.62**	801	99.86	5	801
IDV(15)	0	0	5.18	3.62	*	**6.25**	5.12	*
IDV(16)	16.6	**24.5**	4.46	3.62	*	5.9	5.12	*
IDV(17)	74.1	89.2	91.41	3.62	811	**91.88**	5.12	811
IDV(18)	88.7	**89.9**	30.63	3.62	811	32.49	5.12	811
IDV(19)	0.4	12.7	**99.75**	3.75	801	**99.75**	5.12	801
IDV(20)	26.4	43	89.25	3.62	811	**90.08**	5.12	811

Compared with the traditional model-based fault detection method, prior knowledge of model mechanisms is not needed and only the parameterized matrices related to residuals rather than the system models need to be identified.

Conflicts of Interest

The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

This work is financially supported by the Fundamental Research Funds for the Central Universities (WUT:2017II41GX).

References

[1] J. Chen and R. J. Patton, "Fault diagnosis of non-linear dynamic systems," in *Robust Model-Based Fault Diagnosis for Dynamic Systems*, vol. 3 of *The International Series on Asian Studies in Computer and Information Science*, pp. 251–295, Springer, Boston, Mass, USA, 1999.

[2] S. X. Ding, "Fault identification schemes," in *Model-Based Fault Diagnosis Techniques*, Advances in Industrial Control, pp. 441–469, Springer, London,uk, 2013.

[3] R. Isermann, "Model-based fault-detection and diagnosis—status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71–85, 2005.

[4] J. Cai, H. Ferdowsi, and J. Sarangapani, "Model-based fault detection, estimation, and prediction for a class of linear distributed parameter systems," *Automatica*, vol. 66, pp. 122–131, 2016.

[5] M. Verhaegen and P. Dewilde, "Subspace model identification. I. The output-error state-space model identification class of algorithms," *International Journal of Control*, vol. 56, no. 5, pp. 1187–1210, 1992.

[6] P. van Overschee and B. de Moor, "N4SID: subspace algorithms for the identification of combined deterministic-stochastic systems," *Automatica*, vol. 30, no. 1, pp. 75–93, 1994.

[7] W. E. Larimore, "Canonical variate analysis in identification, filtering, and adaptive control," in *Proceedings of the 29th IEEE Conference on Decision and Control*, pp. 596–604, December 1990.

[8] E. Y. Chow and A. S. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Transactions on Automatic Control*, vol. 29, no. 7, pp. 603–614, 1984.

[9] S. X. Ding, P. Zhang, A. Naik, E. L. Ding, and B. Huang, "Subspace method aided data-driven design of fault detection and isolation systems," *Journal of Process Control*, vol. 19, no. 9, pp. 1496–1510, 2009.

[10] Y. Wang, G. Ma, S. X. Ding, and C. Li, "Subspace aided data-driven design of robust fault detection and isolation systems," *Automatica*, vol. 47, no. 11, pp. 2474–2480, 2011.

[11] S. X. Ding, "Data-driven design of monitoring and diagnosis systems for dynamic processes: a review of subspace technique based schemes and some recent results," *Journal of Process Control*, vol. 24, no. 2, pp. 431–449, 2014.

[12] J. Wang and S. J. Qin, "A new subspace identification approach based on principal component analysis," *Journal of Process Control*, vol. 12, no. 8, pp. 841–855, 2002.

[13] C. T. Chou and M. Verhaegen, "Subspace algorithms for the identification of multivariable dynamic errors-in-variables models," *Automatica*, vol. 33, no. 10, pp. 1857–1869, 1997.

[14] Z. Chen and H. Fang, "A data-driven approach of fault detection for LTI systems," in *Proceedings of the 32nd Chinese Control Conference (CCC '13)*, pp. 6174–6179, Xi'an, China, July 2013.

[15] Z. Chen, H. Fang, and Y. Chang, "Weighted data-driven fault detection and isolation: a subspace-based approach and algorithms," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 5, pp. 3290–3298, 2016.

[16] C. Ruiz-Cárcel, Y. Cao, D. Mba, L. Lao, and R. T. Samuel, "Statistical process monitoring of a multiphase flow facility," *Control Engineering Practice*, vol. 42, pp. 74–88, 2015.

[17] S. Yin, S. X. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.

[18] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[19] T. Kourti and J. F. MacGregor, "Process analysis, monitoring and diagnosis, using multivariate projection methods," *Chemometrics and Intelligent Laboratory Systems*, vol. 28, no. 1, pp. 3–21, 1995.

[20] S. Mahadevan and S. L. Shah, "Fault detection and diagnosis in process data using one-class support vector machines," *Journal of Process Control*, vol. 19, no. 10, pp. 1627–1639, 2009.