Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

A Canonical Variate Analysis based Process Monitoring Scheme and Benchmark Study

Zhiwen Chen*, Kai Zhang*, Haiyang Hao*, Steven X. Ding*
Minjia Krueger*, Zhangming He*, **

* *Institute for Automatic Control and Complex Systems, University of Duisburg-Essen, Bismarckstrasse 81 BB, 47057 Duisburg, Germany (e-mail: zhiwen.chen@uni-due.de).*
** *College of Science, National University of Defense Tech., Yanwachi 47, 410073 Changsha, China*

Abstract: Principal component analysis (PCA) and Partial least square (PLS) are powerful multivariate statistical tools that have been successfully applied for process monitoring. They are efficient in dimension reduction and are suitable for processing large amount of data. Nevertheless, their application scope is restricted to static processes where the dynamics are ignored. In order to achieve improved monitoring performance for dynamic processes, in this paper, we propose an effective dynamic monitoring scheme based on the canonical variate analysis (CVA) technique. Different from the standard PCA- and PLS-based techniques which rely on mean-extraction for residual generation, the proposed CVA-based scheme takes process dynamics into account as well. The properties of all three methods are then compared in detail and finally, the improvements of the proposed method are demonstrated on the well-accepted Tennessee Eastman benchmark process.

Keywords: Canonical variate analysis, principal component analysis, dynamic process monitoring, fault detection.

1. INTRODUCTION

Process monitoring and fault diagnosis techniques have been applied in many areas, such as steel industrial, chemical engineering, electronic engineering and so on. As a major part of those methods, multivariate statistical process monitoring methods (MSPM) have attracted much attention, like principal component analysis (PCA), partial least square (PLS) and canonical variate analysis (CVA) [MacGregor et al., 1995, Qin, 2003, Ding et al., 2013]. PCA is characterized with simplicity and efficiency in processing huge amount of correlated process data. Typically, a few of principal components are extracted from highly correlated process data, and proper test statistics are established for process monitoring purpose. When using two blocks of data, PLS aims to find combinations of variables that are highly correlated. Meanwhile, it selects those linear combinations in a way that eliminates redundancies in the data blocks and defines a new set of variables in each block, which are uncorrelated. A limitation of PCA- and PLS-based approaches is that they rely on static property and *a prior* assumption is that the observations are uncorrelated in time [Hu et al., 2012, Yin et al., 2012]. By using time-lagged variables to develop dynamic model, Ku et al. [1995] proposed a dynamic PCA to deal with this issue.

The CVA technique was originally developed by Hotelling [1933], it was called canonical correlation analysis. Thanks to Akaike and Larimore's work, CVA method has become a main tool in system identification [Akaike, 1974, Larimore, 1983]. A model identified employing CVA can develop several methods for process monitoring. Negiz and Cinar [1997] proposed a T^2 statistic based on estimated state variables for a milk pasteurization process. Russell et al. [2000] studied a residual generation based on state variables to measure the variations outside the state space. In order to detect process changes, Juricek et al. [2004] proposed a local approach based on CVA. Further applications of CVA for process monitoring can be found in [Simoglou et al., 2002, Odiowei et al., 2010]. We notice that little attentions have been focused on the residual generation based on canonical pair. The objective of this paper is to address residual generation based on canonical pair for dynamic process monitoring. According to the generated residual vector, the complete process monitoring scheme is designed. In order to verify that the proposed CVA-based scheme can provide improved performance, a comparison between the PCA, PLS and CVA monitoring schemes is carried out.

The rest of this paper is organized as follows. In Section 2, some preliminaries on static process monitoring are firstly given. Motivated by the state of the art and industrial requirements, we then formulated the problem to be solved. In Section 3, the CVA-based process description and monitoring are proposed. A brief comparison between PCA, PLS and CVA is also done. The comparison results among them on Tennessee Eastman process are shown in Section 4. The paper ends with conclusions in Section 5.

[1] Zhiwen Chen, Kai Zhang and Zhangming He appreciate the founding support from China Scholarship Council(CSC); Chen also appreciates the partial support from the Natural Science Foundation of China (No.61273159).

2. PRELIMINARIES AND PROBLEM FORMULATION

2.1 Preliminaries

For monitoring of static process, PCA and PLS are widely implemented methods. PCA is an optimal dimensionality reduction method that captures the significant variability information in the original data set. Given N observations of m measurement variables form a training data matrix $X \in \mathbf{R}^{m \times N}$. By normalizing the collected data matrix X to zero mean and unit variance, a sampled covariance matrix R is constructed, and by means of singular value decomposition (SVD), the singular values and the corresponding eigenvector can be given as

$$R \approx \frac{1}{N-1} X X^T = P \Lambda P^T, \Lambda = diag(\lambda_1, \ldots, \lambda_m), \quad (1)$$

where N is large. λ_i $(i = 1, \ldots, m)$ are singular values in descending order, P is the eigenvector matrix.

The number of principal components (PCs) l is determined by proper criterion and P, Λ are divided into

$$\Lambda = \begin{bmatrix} \Lambda_{pc} & 0 \\ 0 & \Lambda_{res} \end{bmatrix}, \begin{array}{l} \Lambda_{pc} = diag(\lambda_1, \ldots, \lambda_l), \\ \Lambda_{res} = diag(\lambda_{l+1}, \ldots, \lambda_m), \end{array}$$

$$P = [P_{pc} P_{res}], P_{pc} \in \mathbf{R}^{m \times l}, P_{res} \in \mathbf{R}^{m \times (m-l)}.$$

Based on it, the SPE and T^2_{PCA} statistics

$$SPE = x^T \left(I - P_{pc} P_{pc}^T\right) x = x^T P_{res} P_{res}^T x, \quad (2)$$

$$T^2_{PCA} = x^T P_{pc} \Lambda_{pc}^{-1} P_{pc}^T x, \quad (3)$$

are usually used for process monitoring. Detailed description can be found in [Yin et al., 2012].

Suppose that the process under consideration have normalized quality and process variables data sets $Z \in \mathbf{R}^{n \times N}$ and $Y \in \mathbf{R}^{m \times N}$, respectively $(1 < n < m)$. The core idea of PLS is to

set $Y_1 = Y$ and recursively compute for $i = 1, \cdots, \gamma$

$$w_i^* = \arg \max_{\|w_i\|=1} \left\| w_i^T Y_i Z^T \right\|_E, \quad (4)$$

$$t_i = w_i^T Y_i, p_i = \frac{Y_i t_i^T}{\|t_i\|_E^2}, \quad (5)$$

$$r_i = \begin{cases} w_1, i = 1 \\ \prod_{j=1}^{i-1} \left(I - w_j p_j^T\right) w_i, i > 1 \end{cases}, q_i = \frac{Z t_i^T}{\|t_i\|_E^2}, \quad (6)$$

$$Y_{i+1} = Y_i - p_i t_i^T, \quad (7)$$

where γ is the number of latent variables (LVs) and determined by cross-validation.

and matrices P, Q, R and T are formed. The correlation model given by standard PLS algorithm is

$$\begin{array}{c} Y = T P^T + E, \\ Z = T Q^T + F = Y M + F, M = R Q^T. \end{array} \quad (8)$$

Based on it, the SPE_{PLS} and T^2_{PLS} statistics

$$SPE_{PLS} = \left\| \left(I - P R^T\right) y \right\|_E^2, \quad (9)$$

$$T^2_{PLS} = y^T R \left(\frac{T^T T}{N-1}\right)^{-1} R^T y, \quad (10)$$

are preferred for process monitoring. Detailed description can be found in [Yin et al., 2012].

2.2 Problem formulation

From the preliminaries, it is evident that both PCA and PLS techniques are restricted to static process monitoring. They are inappropriate to monitor dynamic process due to autocorrelation exists in measurements. In this paper, we focus on monitoring dynamic process in steady state. To this end, a stochastic state space representation is considered

$$x(k+1) = Ax(k) + Bu(k) + w(k), \quad (11)$$

$$y(k) = Cx(k) + v(k), \quad (12)$$

where $x \in \mathbf{R}^l$ is the state vector, $u \in \mathbf{R}^m$ is the input vector and $y \in \mathbf{R}^n$ is the output vector, and the system is subject to two uncorrelated, white noises $w \in \mathbf{R}^l$ and $v \in \mathbf{R}^n$. Matrices A, B, C are real constant matrices with appropriately dimension.

The major results of this paper are to:

- design a residual generation scheme for the above dynamic process where
  - the effect of inputs has been eliminated.
- monitor the process by
  - establish a residual evaluation function and
  - determine a proper threshold.

3. A CVA-BASED PROCESS MONITORING SCHEME

3.1 Process description with CVA

Motivated by the state variable analysis by Larimore [1983], We assume that there exist mappings P, M. So that

$$X = P^T Z_p, \quad (13)$$

$$M Z_f = Q X + E, \quad (14)$$

where

$$Z_p(k) = \left[y^T(k-s), \ldots, y^T(k-1), u^T(k-s), \ldots, u^T(k-1)\right]^T,$$
$$Z_f(k) = \left[y^T(k), \ldots, y^T(k+s_f), u^T(k), \ldots, u^T(k+s_f)\right]^T.$$

$Z_p \in \mathbf{R}^{\omega \times N}$, $Z_f \in \mathbf{R}^{\eta \times N}$ are two stacked process data sets, and collected from the process output vector (sensors) and input vector (actuators) in a time interval which is divided into the 'past' and 'future' periods. E denotes the white noise and X denotes the sample set of the process canonical correlation variables (state variables). s and s_f are the number of lags, we determine them by the method mentioned in [Odiowei et al., 2010].

Equivalently, the model (11)-(12) can be written as

$$x(k+1) = A_K x(k) + Bu(k) + Ky(k), A_K = A - KC, \quad (15)$$

$$y(k) = Cx(k) + e(k), \quad (16)$$

for some K that ensures the eigenvalues of A_K are all located in the unit circle to make the system stable, $e(k)$ is the so-called innovation sequence. It is easily obtained from (15) that

$$x(k) = A_K^s x(k-s) + \sum_{i=1}^{s} A_K^{i-1} Bu(k-i) + \sum_{i=1}^{s} A_K^{i-1} Ky(k-i), \quad (17)$$

and thus for a large s

$$x(k) \approx P^T z_p, P^T = [P_y \ P_u], \quad (18)$$

where
$P_y = \left[A_K^{s-1}K \ \ldots \ A_K K \ K\right], P_u = \left[A_K^{s-1}B \ \ldots \ A_K B \ B\right]$.
The 'past' process measurements $z_p(k)$ includes the process input and output data in the time period $[k-s, k-1]$.

Let the future output

$$y_{s_f}(k) = \left[y^T(k), y^T(k+1), \ldots, y^T(k+s_f)\right]^T$$

which removes the futre input, is a part of $Z_f(k)$. The future input vector $u_{s_f}(k)$ and future noise vector $e_{s_f}(k)$ are similar in form as $y_{s_f}(k)$.

In order to derive the model (14), the representation of (15) can be changed as

$$x(k+1) = Ax(k) + Bu(k) + Ke(k).$$

Then, we have

$$y_{s_f}(k) = \Gamma_{s_f} x(k) + H_{u,s_f} u_{s_f}(k) + H_{e,s_f} e_{s_f}(k), \quad (19)$$

where

$$\Gamma_{s_f} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{s_f} \end{bmatrix}^T, H_{u,s_f} = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ CB & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s_f-1}B & \ldots & CB & 0 \end{bmatrix},$$

$$H_{e,s_f} = \begin{bmatrix} I & 0 & \ldots & 0 \\ CK & I & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ CA^{s_f-1}K & \ldots & CK & I \end{bmatrix}.$$

(19) can be further written as

$$Mz_f(k) = Qx(k) + E,$$

where

$$z_f(k) = \left[y_{s_f}(k) \ u_{s_f}(k)\right],$$
$$M = \left[I \ -H_{u,s_f}\right], Q = \Gamma_{s_f}, E = H_{e,s_f} e_f(k),$$

s_f is some (large) integer. $z_f(k)$ is composed of the 'future' process data in the time period $[k, k+s_f]$.

In the context of state space representations, the state vector given by (13) can be different from the one defined in (11). In that case, there exists a regular state transformation.

3.2 CVA-based process monitoring

We assume that

$$\begin{pmatrix} z_p \\ z_f \end{pmatrix} \sim N\left(\begin{bmatrix} \varepsilon(z_p) \\ \varepsilon(z_f) \end{bmatrix}, \begin{bmatrix} \Sigma_{zp} & \Sigma_{z_p z_f} \\ \Sigma_{z_f z_p} & \Sigma_{z_f} \end{bmatrix}\right),$$

where $\varepsilon(\cdot)$ represents the expectation operator, Σ_{z_p} and Σ_{z_f} are covariance of the 'past' and 'future' process data, respectively. $\Sigma_{z_p z_f}$ is the cross variance. We suppose that Z_p, Z_f are normalized data sets. As analysed by Hotelling [1933], do an SVD

$$\begin{aligned} &(\tfrac{Z_p Z_p^T}{N-1})^{-1/2}(\tfrac{Z_p Z_f^T}{N-1})(\tfrac{Z_f Z_f^T}{N-1})^{-1/2} = \\ &(Z_p Z_p^T)^{-1/2}(Z_p Z_f^T)(Z_f Z_f^T)^{-1/2} = U\Sigma V^T, \\ &\Sigma = \begin{bmatrix} diag(\delta_1, \ldots, \delta_l) & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned} \quad (20)$$

The diagonal matrix Σ contains the singular values in a descending order. The ratio of cumulative singular values to the sum of all the singular values (CPV) is employed to determine l, which is the order of the system [Negiz and Cinar, 1997]. The state variables are linear combinations of the mapping matrix P, which is given as:

$$P = (Z_p Z_p^T)^{-1/2} U_l, L = (Z_f Z_f^T)^{-1/2} V_l, \quad (21)$$

where U_l consists the first l columns of U. The matrix P and L satisfy

$$P^T Z_p Z_p^T P = I, L^T Z_f Z_f^T L = I.$$

Motivated by Juricek et al. [2004], we applied a projection method to remove the effects of future inputs. The normalization is

$$\bar{Y}_f = Y_f - (Y_f U_f^T)(U_f U_f^T)^{-1} U_f, \quad (22)$$

where Y_f, U_f are the sub-matrices of Z_f. After data normalization, $\bar{Y}_f \in \mathbf{R}^{\tau \times N}$ represents N normalized samples of the output data

$$\bar{y}_f(k) = \left[\, \bar{y}^T(k), \ldots, \bar{y}^T(k+s_f) \,\right]^T \in \mathbf{R}^\tau, \tau = (s_f+1)n.$$

It is well known that in process monitoring that the state variables reflect the process operation and behavior and therefore, process engineers can extract real-time and significant information from it.

The idea behind the fault detection scheme is sufficient utilization of the relations amongst the canonical correlation vectors. Let Z_P and $\bar{Y}_f$ be normalized, recall that by equ(20-21), it is obvious that

$$\begin{aligned} &P^T Z_p \bar{Y}_f^T L = \Sigma, \\ &P = (Z_p Z_p^T)^{-\frac{1}{2}} U_l, L = (\bar{Y}_f \bar{Y}_f^T)^{-\frac{1}{2}} V_l, \end{aligned} \quad (23)$$

where P shares the same symbol as in (13).

A standard method for monitoring process using state-space models is based on the statistical properties of output residual. We define a fair intuitive residual sequence. Hence, (23) leads to

$$\Sigma P^T Z_p = L^T \bar{Y}_f. \quad (24)$$

Based on this relation, we define a residual vector

$$r(k) = L^T \bar{y}_f(k+s_f+1) - \Sigma P^T z_{s-1}(k) \in \mathbf{R}, \quad (25)$$

for building a test statistic. Note that the covariance matrix of $r(k)$ can be estimated by

$$\begin{aligned} &(L^T \bar{Y}_f - \Sigma P^T Z_p)(L^T \bar{Y}_f - \Sigma P^T Z_p)^T \\ &= L^T \bar{Y}_f \bar{Y}_f^T L + \Sigma^2 P^T Z_p Z_p^T P - 2\Sigma P^T Z_p \bar{Y}_f^T L \\ &= I - \Sigma^2. \end{aligned} \quad (26)$$

We assumed the 'past' and 'future' process data are normally distributed, the canonical variate residual should also have a multivariable normal distribution with covariance matrix given in (26).The variations in the state space can be monitored by employing the T_r^2 statistic

$$T_r^2 = r^T(k)(I - \Sigma^2)^{-1} r(k). \quad (27)$$

The T_r^2 statistic is a combination of each sample in every dimension with weighted coefficient and compared with an upper limit (threshold). The threshold J_{th,T^2} can be determined by specifying the type I error probability, α, of the current T_r^2 value exceeding the threshold. And the threshold is given as

$$J_{th,T^2} = \frac{l(N^2-1)}{N(N-l)} F_\alpha(l, N-l), \quad (28)$$

with a given significant level α to detect variations (faults) inside the residual space.

Table 1. A brief comparison among CVA, PCA and PLS methods

Method	Numerical robustness	Main applications	Computation cost
PCA	High	Large-scale static process	Low: one SVD on $m \times m$ matrix
PLS	High	Static process,KPI-related monitoring	Medium: γ times SVD on $m \times m$ matrix
CVA	Low	Steady dynamic process	Medium: one SVD on $(m+n)s \times (s_f+1)n$ matrix

The design steps of the process monitoring scheme consisting of off-line training and on-line monitoring, is summarized in the following algorithm.

Off-line training

- *Normalize the process data, which delivers* $Z_p, \bar{Y}_f$.
- *Determine the number of lags* s *and* s_f.
- *Do an SVD*
$$(Z_p Z_p^T)^{-1/2}(Z_p \bar{Y}_f^T)(\bar{Y}_f \bar{Y}_f^T)^{-1/2} = U \Sigma V^T.$$
- *Determine the system order* l *and form* P, L *according to (23).*
- *Compute* $(I-\Sigma^2)^{-1}$ *and build the test statistic (28).*
- *Determine the corresponding threshold.*

On-line monitoring

- *Collect new data for monitoring and normalization.*
- *Calculate current states and residual sequence.*
- *Calculate the* T_r^2 *metric according to (27).*
- *Make a decision according to the detection logic.*
$$T_r^2 > J_{th,T^2} \Rightarrow \text{faulty, otherwise fault free} \quad (29)$$

3.3 A comparison among CVA, PCA and PLS methods

In this section, we only focus on the discussion upon applying PCA, PLS and CVA to fault detection issue. Previously, we need to claim two requirements: 1) All the three methods assume the observation signals follow multivariate normal distribution; 2) Observations of interest are normalized.

Table 1 indicates a brief comparison among CVA, PCA and PLS methods, in which numerical robustness, main applications and computation cost are mainly considered. In PCA fault detection framework, one SVD of the sample covariance matrix is the core of this method, which is a stable numerical solution for matrix decomposition, this is equivalent to remove the correlation between variables. In PCA, the measurement subspaces is decomposed into principal space and residual space, then two test statistics T_{PCA}^2 and SPE (squared prediction error) are defined.

Furthermore, standard PLS algorithm possesses high numerical robustness, because it avoids directly compute the inverse covariance matrix by recursively computing latent variables. As the correlation model (4) described, the original idea behind PLS fault detection is to identify combinations of variables that are highly correlated by using covariance information. In practice, product quality variables are usually difficult to measure or delay-sampling compare with process variable, so the quality variable will not be directly used in on-line monitoring as shown in (9) and (10). When taking quality variable into account, PLS is preferred as aiming to detect the faults in process variables that are mostly related to quality variable or key performance indicator (KPI) [Ding et al., 2013].

One major limitation of PCA and PLS is that they do not take autocorrelation into account. The CVA-based method can efficiently solve this problem by using the state-space representation. However, the CVA-based method has a low numerical robustness, due to the inverse of covariance matrix $Z_P Z_P^T$ and $\bar{Y}_f \bar{Y}_f^T$. The basic idea of CVA is to construct state variables directly from past variable Z_p and future variable Z_f. The estimated state variable is treated as a linear combination of past variable, like $X = P^T Z_P$. Several statistics have been developed by Russell et al. [2000], Juricek et al. [2004], all have obtained efficient monitoring results. The proposed CVA method developed on a residual generation based on canonical variable pair. Due to the orthogonality of canonical pair, the residual vector $r(k)$ possesses good statistical property, and the T_r^2 statistic can efficiently monitor the steady dynamic process.

4. BENCHMARK STUDY

Tennessee Eastman Process (TEP) benchmark is a well-accepted platform designed to simulate the real chemical producing process. It has been extensively employed to show the applicability of various academic approaches. It could be universally found of its technical description in a large number of publications [Russell et al., 2000, Yin et al., 2012]. In this section, we will demonstrate the application of CVA to fault detection on the TEP simulator developed by Richer et al. [1996]. The simulator is online available at the website [2]. There are six operating modes defined in [Downs and Vogel, 1993] for the TEP. In this application study, the process is running under mode one and the decentralized control strategy developed in [Richer et al., 1996] is adopted. The sampling time is set to be three minutes and the simulation time 48 hours, which generates 960 samples of data for each scenario. All the 20 faults defined by Downs and Vogel could be realized in the TEP simulator. The whole data set consisting of 52 variables with 960 samples may cause tremendous computation cost and memory problem, thus it is of great benefit to divide them into several groups, and each group covers explicit input-output information. The group-divided results could be seen in [Ding et al., 2009]. In this study, the manipulated variables (XMV 1-11) are applied as the inputs, two blocks of outputs are designed in Table 2. To demonstrate the effectiveness of the new approach, the well-applied PCA and PLS based methods are taken into account as the comparison objects.

In the training phase, two types of input-output patterns are trained with CVA. The needed parameters are determined, see Table 3. $s_f = s$ is predefined a priori, and the process order n could be achieved imitating PCA-like manner, see reference [Odiowei et al., 2010]. Finally, threshold-setting comes from popular T^2 based approach with a significance level 0.05.

[2] http://depts.washington.edu/control/LARRY/TE/download.html

In the first scenario, the block one output variables (Reactor feed analysis) are considered. Taking fault 12 as the first case study, Fig. 2 sketches the PCA-based detection results, from which both two detection index could not function well for this fault. Turn to PLS based detection, see in Fig. 2, it appears to be the same terrible like PCA-based one. However, CVA-based result, see in Fig. 3, behaves good, as it significantly improve the detection rate.

Table 2. Selected Outputs

Block name	Variable name	variable number
Reactor feed analysis	Component A	XMEAS(23)
	Component B	XMEAS(24)
	Component C	XMEAS(25)
	Component D	XMEAS(26)
	Component E	XMEAS(27)
	Component F	XMEAS(28)
Product analysis	Component D	XMEAS(37)
	Component E	XMEAS(38)
	Component F	XMEAS(39)
	Component G	XMEAS(40)
	Component H	XMEAS(41)

Table 3. Parameters in training phase

No.	m	l	s_f	s	N	n
1	6	11	12	12	468	59
2	5	11	17	17	463	70

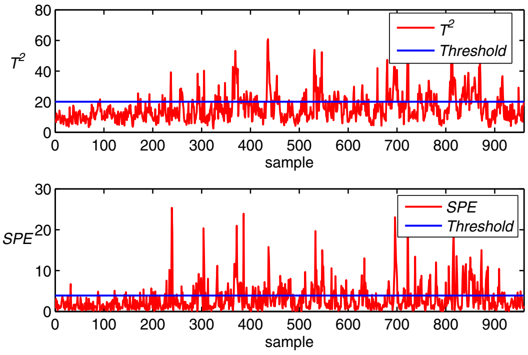

Fig. 1. PCA-based detection for fault 12 with group 1 output variables

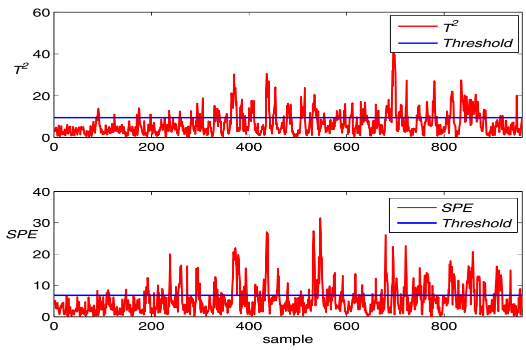

Fig. 2. PLS-based detection for fault 12 with group 1 output variables

Associate with the second scenario, the block two outputs (Product analysis) are utilized instead. Fig. 4 plots PCA's

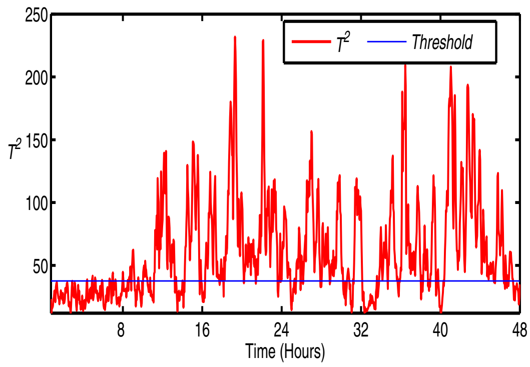

Fig. 3. CVA-based detection for fault 12 with group 1 output variables

performance, of which functionality is yet not improved compared with the first scenario. So do in PLS-based results in Fig. 5. Similarly, it has be considerably enhanced by the present CVA-based approach, referred in Fig. 6. CVA-based method proposed in this paper, has strongly improved the detection result.

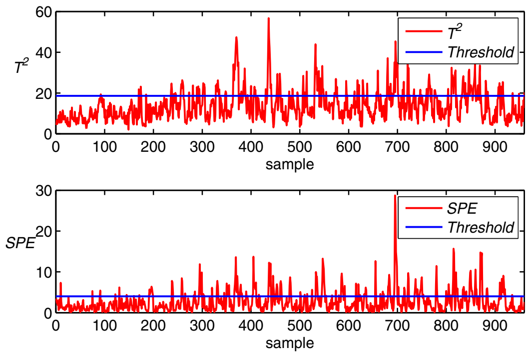

Fig. 4. PCA-based detection for fault 12 with group 2 output variables

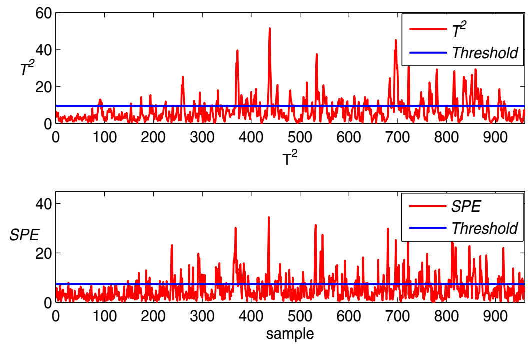

Fig. 5. PLS-based detection for fault 12 with group 2 output variables

Finally, with respect to all well-defined 20 faults, the fault detection rates are summarized for PCA-, PLS- and CVA-based methodologies, and are shown in Table 4. Referring the figures, it can evidently draw that the proposed approach overweigh the PCA- and PLS-based methods.

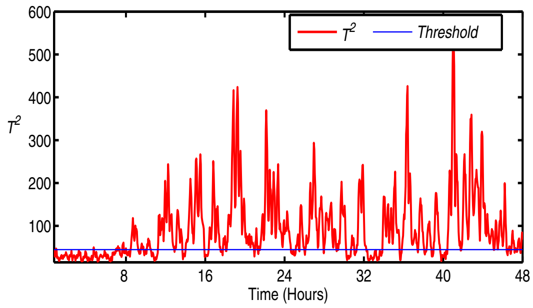

Fig. 6. CVA-based detection for fault 12 with group 2 output variables

Table 4. Fault detection rate comparison between CVA, PCA and PLS

No.	Output block 1			Output block 2		
	CVA	PCA	PLS	CVA	PCA	PLS
1	0.9963	0.9975	0.9975	0.9963	0.9988	0.9988
2	0.9950	0.9900	0.9990	0.9900	0.9875	0.9900
3	0.1061	0.0437	0.1336	0.1350	0.0649	0.1311
4	0.9988	0.9988	0.9988	0.9988	0.9988	0.9988
5	0.1911	0.0587	0.1286	0.0849	0.0624	0.1236
6	0.9988	0.9988	0.9988	1	1	1
7	0.9875	0.9900	0.9988	0.9988	0.9988	0.9988
8	0.9990	0.9913	0.9913	0.9925	0.9900	0.9913
9	0.1511	0.0924	0.1760	0.0974	0.0836	0.1686
10	0.6729	0.8402	0.8302	0.9438	0.8277	0.8165
11	0.9838	0.9600	0.9626	0.9900	0.9413	0.9663
12	0.6954	0.4657	0.4382	0.8190	0.4145	0.4507
13	0.9888	0.9863	0.9875	0.9913	0.9838	0.9863
14	0.9950	0.9026	0.9064	0.9950	0.8739	0.9114
15	0.1111	0.0737	0.1136	0.1099	0.0524	0.1061
16	0.0936	0.0437	0.1049	0.0424	0.0549	0.0999
17	0.9650	0.9164	0.9213	0.9650	0.8801	0.9164
18	0.9613	0.8677	0.3146	0.7203	0.2797	0.2884
19	0.9950	0.9963	0.9975	0.9950	0.9888	0.9775
20	0.9863	0.9818	0.9800	0.9913	0.9738	0.9775

5. CONCLUSIONS

In this paper,we have proposed a CVA-based method for dynamic process monitoring by constructing residual with canonical variate. Different from the standard PCA- and PLS-based techniques which rely on mean-extraction for residual generation, the proposed CVA-based scheme takes process dynamics into account. In addition, the proposed scheme is tested on an industrial benchmark of TEP and compared with the standard PCA and PLS methods. As for the dynamic TE simulation, the proposed method shows better performance than PCA and PLS in effectiveness and superiority. Note that the fault detection scheme is no need to identify the state-space matrices, only need to find the canonical correlation defined by P and L. The detection delay is inevitable, because this scheme uses a immediate relation between output and input process data. As for this, comparison study the proposed schemes with other scheme based on identified state-space in the aspect of fault detection delay will be further extended.

REFERENCES

H. Akaike. Stochastic theory of minimal realization. *IEEE Trans.Automat. Contr.*, AC-19, 6:667-674, 1974.

S.X. Ding, P. Zhang, A. Naik, E.L. Ding, B. Huang. Subspace method aided data-driven design of fault detection and isolation systems. *Journal of Process Control*, 19:1496-1510, 2009.

S.X. Ding, S. Yin, K.X. Peng, H.Y. Hao, B. Shen. A novel scheme for key performance indicator predition and diagnosis with application to an industrial hot strip mill. *IEEE Transactions on Industrial Informatics*, 9(4):2239-2247, 2013.

J.J. Downs and E.F. Vogel. A plant-wide industrial process control problem. *Computers and Chemical Engineering*, 17:245-255, 1993.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *J.Educ.Psychol.*, 24:417-498, 1933.

Z.K. Hu, Z.W. Chen, C.C. Hua, W.H. Gui, C.H. Yang and S.X. Ding. A simplified recursive dynamic pca based monitoring scheme for imperal smelting process. *International Journal of Innovative Computing, Information and Control*, 8(4):2551-2561, 2012.

B.C. Juricek, D.E. Seborg and W.E. Larimore. Fault detection using canonical variate analysis. *Ind.Eng.Chem.Res.*, 43:458-474, 2004.

W.F Ku, R.H. Storer and C. Georgakis. Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics & intelligent laboratory systems*, 30(1):179-19, 1995.

W.E. Larimore. System identification, reduced order filtering and modeling via canonical variate analysis. *Proc. Amer. Control Conf.*, 445-451, June 1983.

W.E. Larimore. Statistical optimality and canonical variate analysis sytem identification. *Signal Processing*, 52:131-144, 1996.

J. MacGregor, T. Kourti. Statistical process control of multivariate processes. *Control Engineering Practice*, 3:403-414, 1995.

A. Negiz, A. Cinar. Statistical monitoring of multivariable dynamic process with state space models. *AIChE Journal*, 8:2002-2020, 1997.

N.L. Ricker. Decentralized control of the Tennessee Eastman Challenge Process. *Journal of Process Control*, 6(4):205-221, 1996.

P. Odiowei, and Y. Cao. Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations. *IEEE Transactions on Industry Informatics*, 6:36-45, 2010.

S.J. Qin. Statistical process monitoring: Basics and beyond. *Journal of Chemometrics*, 17:480-502, 2003.

R. Shi, J.F. MacGregor. Modeling of dynamic systems using latent variable and subspace methods. *Journal of Chemometrics*, 14(5-6):423-439, 2000.

E. Russell, L.H. Chiang, and R.D. Braatz. Data-driven methods for fault detection and diagnosis in chemical processes. *Springer-Verlag, 1st ed.*, 2000.

A.Simoglou, E.B. Martin and A.J. Morris. Statistical performance monitoring of dynamic multivariate processes using state space modelling. *Computers and Chemical Engineering*, 26:909-920, 2002.

S. Yin, S.X. Ding, A. Haghani, H.Y. 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*, 22:1567-1581, 2012.