Chemometrics and Intelligent Laboratory Systems 51 (2000) 81–93

Chemometrics and intelligent laboratory systems

www.elsevier.com/locate/chemometrics

Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis

Evan L. Russell, Leo H. Chiang, Richard D. Braatz *

Large Scale Systems Research Laboratory, Department of Chemical Engineering, University of Illinois at Urbana-Champaign, 600 South Mathews Avenue, Box C-3, Urbana, IL 61801-3792, USA

Accepted 24 January 2000

Abstract

Principal component analysis (PCA) is a well-known data dimensionality technique that has been used to detect faults during the operation of industrial processes. Dynamic principal component analysis (DPCA) and canonical variate analysis (CVA) are data dimensionality techniques which take into account serial correlations, but their effectiveness in detecting faults in industrial processes has not been extensively tested. In this paper, score/state and residual space PCA, DPCA, and CVA are applied to the Tennessee Eastman process simulator, which was designed to simulate a wide variety of faults occurring in a chemical plant based on a facility at Eastman Chemical. This appears to be the first application of residual space CVA statistics for detecting faults in a large-scale process.

Statistics quantifying variations in the residual space were usually more sensitive but less robust to the faults than the statistics quantifying the variations in the score or state space. The statistics exhibiting a small missed detection rate tended to exhibit small detection delays and vice versa. A residual-based CVA statistic proposed in this paper gave the best overall sensitivity and promptness, but the initially proposed threshold for the statistic lacked robustness. This motivated increasing the threshold to achieve a specified missed detection rate. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Fault detection; Principal component analysis; Dynamic principal component analysis; Canonical variate analysis; Process monitoring; Chemometrics methods; Tennessee Eastman process; Multivariate statistics

1. Introduction

Large amounts of data are collected in many industrial processes. The task of fault detection is to use this data to determine when abnormal process behavior has occurred, whether associated with equipment failure, equipment wear, or extreme process faults. While techniques based on first-principles models have been around for more than two decades, their contribution to industrial practice has not been pervasive due to the substantial cost and time required to develop a sufficiently accurate process model for a complex chemical plant. The fault detection techniques that have dominated the literature for the past decade and have been most effective in practice are

* Corresponding author. Tel.: +1-217-333-5073; fax: +1-217-333-5052.

E-mail address: braatz@uiuc.edu (R.D. Braatz).

0169-7439/00/$ - see front matter
PII: S0169-7439(00)00058-7

based on models constructed almost entirely from process data.

The accuracy of detecting faults from data can be improved using data dimensionality reduction techniques, such as principal component analysis (PCA), dynamic principal component analysis (DPCA), and canonical variate analysis (CVA). The lower dimensional representations produced by these techniques can better generalize to new process data than representations using the entire dimensionality.

Academic and industrial process control engineers have applied PCA for abstracting structure from multidimensional chemical process data [1,2]. PCA determines the most accurate lower dimensional representation of the data, in terms of capturing the data directions that have the most variance. The resulting lower dimensional model has been used for detecting out-of-control status and for diagnosing faults leading to the abnormal process operation [3]. Several applications of PCA to real industrial data have been conducted at DuPont and other companies over the past 6 years, with much of the results available in various publications (for example, see Refs. [4,5] and citations therein).

PCA can be extended to take into account serial correlations in the data by augmenting each observation vector with the previous l observations. We will refer to this approach as *dynamic* PCA (DPCA), irrespective of how the number of lags are selected (the DPCA method of Ref. [6] is one implementation of this approach). CVA is a dimensionality reduction technique in multivariate statistical analysis involving the selection of pairs of variables from the inputs and outputs that maximize a correlation statistic [7]. Like DPCA, the method takes serial correlations into account during the dimensionality reduction procedure.

PCA has been used to detect faults from data collected from real chemical plants and computer simulations (the Tennessee Eastman process [3,5,6,8–13]). Applications of DPCA and CVA to chemical processes either in simulation or industry are much more limited [6,14–16]. The objective of this paper is to evaluate and compare the performance of PCA, DPCA, and CVA for detecting faults in a realistic chemical process simulation. In this comparison, a CVA-based residual space statistic is proposed (T_r^2) for use in fault detection. As will be seen later, the proposed statistic gave better overall sensitivity and promptness than the existing PCA, DPCA, and CVA statistics applied to the Tennessee Eastman process.

The paper is organized as follows. First, PCA and DPCA are briefly described. Then, the CVA statistical method and fault detection statistics are described. Finally, PCA, DPCA, and CVA are applied to data collected from the Tennessee Eastman process simulator. The sensitivity, promptness, and robustness of the statistics are compared.

2. Methods

2.1. PCA

2.1.1. Definition

PCA is an optimal dimensionality reduction technique in terms of capturing the variance of the data. PCA determines a set of orthogonal vectors, called *loading vectors*, which can be ordered by the amount of variance explained in the loading vector directions. Given n observations of m measurement variables stacked into a training data matrix $\mathbf{X}$, the loading vectors are calculated by computing the singularities of the optimization problem

$$\max_{v \neq 0} \frac{v^T \mathbf{X}^T \mathbf{X} v}{v^T v} \tag{1}$$

where $v \in \mathscr{R}^m$. The stationary points of Eq. (1) can be computed via the SVD

$$\frac{1}{\sqrt{n-1}} \mathbf{X} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \tag{2}$$

where $\mathbf{U} \in \mathscr{R}^{n \times n}$ and $\mathbf{V} \in \mathscr{R}^{m \times m}$ are unitary matrices and the matrix $\mathbf{\Sigma} \in \mathscr{R}^{n \times m}$ contains the nonnegative real *singular values* of decreasing magnitude ($\sigma_1 \geq \ldots \geq \sigma_{\min(m,n)} \geq 0$). The loading vectors are the orthonormal column vectors in the matrix $\mathbf{V}$, and the variance of the training set projected along the ith column of $\mathbf{V}$ is equal to σ_i^2.

2.1.2. Fault detection

Normal operations can be characterized by employing Hotelling's T^2 statistic [4]:

$$T^2 = \boldsymbol{x}^T P \mathbf{\Sigma}_a^{-2} P^T \boldsymbol{x} \tag{3}$$

where P includes the loading vectors associated with the a largest singular values, $\mathbf{\Sigma}_a$ contains the first a

rows and columns of $\boldsymbol{\Sigma}$, and $\boldsymbol{x}$ is an observation vector of dimension m. Given a number of loading vectors, a, to include in Eq. (3), the threshold can be calculated for the T^2 statistic using the probability distribution

$$T_\alpha^2 = \frac{(n^2-1)a}{n(n-a)} F_\alpha(a,n-a) \tag{4}$$

where $F_\alpha(a,n-a)$ is the upper $100\alpha\%$ critical point of the F-distribution with a and $n-a$ degrees of freedom. The T_2 statistic with Eq. (4) defines the normal process behavior, and an observation vector outside this region indicates that a fault has occurred.

The portion of the measurement space corresponding to the lowest $m-a$ singular values can be monitored by using the Q statistic developed by Jackson and Mudholkar [17]:

$$Q = r^T r, \qquad r = (I - PP^T)\boldsymbol{x} \tag{5}$$

The threshold for the Q statistic can be calculated from its approximate distribution [17]:

$$Q_\alpha = \theta_1 \left[\frac{h_0 c_\alpha \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right]^{1/h_0} \tag{6}$$

where $\theta_i = \sum_{j=a+1}^{n} \sigma_j^{2i}$, $h_0 = 1 - (2\theta_1\theta_3)/3\theta_2^2$, and c_α is the normal deviate corresponding to the $(1-\alpha)$ percentile.

2.1.3. Reduction order

A key step in a data dimensionality reduction technique is to determine the order of the reduction. There exist several techniques for determining the number of loading vectors, a, to maintain in the PCA [8,18]. Ku et al. [6] recommend the parallel analysis method, because it had the best performance in their applications. Parallel analysis determines the dimensionality of the PCA model by comparing the singular value profile to that obtained by assuming independent measurement variables. The dimension is determined by the point at which the two profiles cross. This approach is particularly attractive since it is intuitive and easy to automate.

2.1.4. DPCA

The previously discussed PCA monitoring methods implicitly assume that the observations at one time instant are statistically independent to observations at past time instances. For typical chemical processes, this assumption is valid only for long sampling intervals, e.g., 2 to 12 h. Shorter sampling intervals are desired for speedy fault detection, in which case a method taking into account the serial correlations in the data may provide improved fault detection. The PCA methods can be extended to take into account the serial correlations, by augmenting each observation vector with the previous l observations and stacking the data matrix in the following manner,

$$\mathbf{X}(l) = \begin{bmatrix} x_t^T & x_{t-1}^T & \cdots & x_{t-l}^T \\ x_{t-1}^T & x_{t-2}^T & \cdots & x_{t-l-1}^T \\ \vdots & \vdots & \ddots & \vdots \\ x_{t+l-n}^T & x_{t+l-n-1}^T & \cdots & x_{t-n}^T \end{bmatrix} \tag{7}$$

where $\boldsymbol{x}_t^T$ is the m-dimensional observation vector in the training set at time instance t. By performing PCA on the data matrix in Eq. (7), a multivariate autoregressive (AR) (ARX model if the process inputs are included) is extracted directly from the data [6,19]. The Q statistic is then the squared prediction error of the ARX model. If enough lags l are included in the data matrix, the Q statistic is statistically independent from one time instant to the next, and the threshold (6) is theoretically justified. This approach of applying PCA to Eq. (7) is referred to here as *dynamic* PCA (DPCA). To determine the number of lags l, autoregressive models with several different numbers of lags to the *training data* are fitted. The number of lags l is selected to be the lag minimizing the small sample AIC criterion [7]. Note that this method for automatically determining l is not the same as that used by the DPCA algorithm of Ku et al. [6]. The T_2 and Q statistics and their thresholds generalize directly to DPCA.

2.2. CVA

2.2.1. The CVA statistical method

CVA is a dimensionality reduction technique that is optimal in terms of maximizing a correlation statistic between two sets of variables. Given vectors of variables $\boldsymbol{x} \in \mathscr{R}^m$ and $\boldsymbol{y} \in \mathscr{R}^n$ with covariance matrices $\boldsymbol{\Sigma}_{xx}$ and $\boldsymbol{\Sigma}_{yy}$, and cross covariance matrix

$\mathbf{\Sigma}_{xy}$, there exist matrices $\mathbf{J} \in \mathscr{R}^{m \times m}$ and $\mathbf{L} \in \mathscr{R}^{n \times n}$ such that

$$\mathbf{J\Sigma}_{xx}\mathbf{J}^T = \mathbf{I}_{\overline{m}}, \quad \mathbf{L\Sigma}_{yy}\mathbf{L}^T = \mathbf{I}_{\overline{n}} \tag{8}$$

and

$$\mathbf{J\Sigma}_{xy}\mathbf{L}^T = \mathbf{D} = \mathrm{diag}(\gamma_1, \cdots, \gamma_r, 0, \cdots, 0), \tag{9}$$

where $\gamma_1 \geq \ldots \geq \gamma_r$, $\overline{m} = \mathrm{rank}(\mathbf{\Sigma}_{xx})$, $\overline{n} = \mathrm{rank}(\mathbf{\Sigma}_{yy})$, $\mathbf{D}$ contains the *canonical correlations* γ_i, and the notation $\mathbf{I}_k$ denotes the block diagonal matrix containing the $k \times k$ identity matrix for the first block and a zero matrix for the second block [7]. The vector of *canonical variables* $\boldsymbol{c} = \boldsymbol{Jx}$ contains a set of uncorrelated random variables and has the covariance matrix

$$\mathbf{\Sigma}_{cc} = \mathbf{J\Sigma}_{xx}\mathbf{J}^T = \mathbf{I}_{\overline{m}}, \tag{10}$$

and the vector of *canonical variables* $\boldsymbol{d} = \mathbf{Ly}$ contains a set of uncorrelated random variables and has the covariance matrix

$$\mathbf{\Sigma}_{dd} = \mathbf{L\Sigma}_{yy}\mathbf{L}^T = \mathbf{I}_{\overline{n}}. \tag{11}$$

The cross covariance matrix between $\boldsymbol{c}$ and $\boldsymbol{d}$ is diagonal (see (9)), and the two vectors are only pairwise correlated. The degree of the pairwise correlations are indicated and can be ordered by the canonical correlations γ_i. In the case where the covariance matrices are invertible (the usual case in practice), the projection matrices $\mathbf{J}$ and $\mathbf{L}$ and the matrix of canonical correlations $\mathbf{D}$ can be computed by solving the SVD:

$$\mathbf{\Sigma}_{xx}^{-1/2}\mathbf{\Sigma}_{xy}\mathbf{\Sigma}_{yy}^{-1/2} = \mathbf{U\Sigma V}^T \tag{12}$$

where $\mathbf{J} = \mathbf{U}^T\mathbf{\Sigma}_{xx}^{-1/2}$, $\mathbf{L} = \mathbf{V}^T\mathbf{\Sigma}_{yy}^{-1/2}$, and $\mathbf{D} = \mathbf{\Sigma}$ [20]. The weightings $\mathbf{\Sigma}_{xx}^{-1/2}$ and $\mathbf{\Sigma}_{yy}^{-1/2}$ ensure that the canonical variables are uncorrelated and have unit variance, and the matrices $\mathbf{U}^T$ and $\mathbf{V}^T$ rotate the canonical variables so that $\boldsymbol{c}$ and $\boldsymbol{d}$ are only pairwise correlated. The degree of the pairwise correlations are indicated by the singular values.

2.2.2. CVA state vector construction

While the CVA concept for multivariate statistical analysis was developed by Hotelling, it was not applied to system identification until Akaike's work on the ARMA model [7,21–23]. Given time series input data $u_t \in \mathscr{R}^{m_u}$ and output data $y_t \in \mathscr{R}^{m_y}$, the linear state space model is given by [24]

$$x_{t+1} = \Phi \boldsymbol{x}_t + Gu_t + w_t \tag{13}$$

$$y_t = H\boldsymbol{x}_t + Au_t + Bw_t + v_t \tag{14}$$

where $\boldsymbol{x}_t \in \mathscr{R}^k$ is a k-order state vector and w_t and v_t are white noise processes that are independent with covariance matrices $\mathbf{Q}$ and $\mathbf{R}$, respectively. It is assumed that the state space matrices and covariance matrices are constant. The term Bw_t in Eq. (14) allows the noise in the *output equation* (14) to be correlated with the noise in the *state equation* (13). Omitting the term Bw_t, typically done for many state space models, may result in a state order that is not minimal [20].

At a particular time instant $t = 1, \ldots, n$ the vector containing the information from the past is

$$\boldsymbol{p}_t = \left[y_{t-1}^T, y_{t-2}^T, \cdots, u_{t-1}^T, u_{t-2}^T, \cdots \right]^T, \tag{15}$$

and the vector containing the output information in the present and future is

$$\boldsymbol{f}_t = \left[y_t^T, y_{t+1}^T, \cdots \right]^T. \tag{16}$$

The projection matrices $\mathbf{J}$ and $\mathbf{L}$ and the matrix of canonical correlations $\mathbf{D}$ can be computed via the SVD by substituting the matrix $\mathbf{\Sigma}_{xx}$ with $\mathbf{\Sigma}_{pp}$, $\mathbf{\Sigma}_{yy}$ with $\mathbf{\Sigma}_{ff}$, and $\mathbf{\Sigma}_{xy}$ with $\mathbf{\Sigma}_{pf}$ in Eq. (12). Assuming the order of the state space model, k, is chosen to be greater than or equal to the order of the minimal state space realization of the actual system, the state vector ($\boldsymbol{x}_t$ in Eqs. (13) and (14)) can be extracted from the data using CVA states

$$\boldsymbol{x}_t = \mathbf{J}_k \boldsymbol{p}_t = \mathbf{U}_k^T \hat{\mathbf{\Sigma}}_{pp}^{-1/2} \boldsymbol{p}_t \tag{17}$$

where $\mathbf{U}_k$ contains the first k columns of $\mathbf{U}$ in Eq. (12) [7].

Since in practice there is a finite amount of data available, the vectors $\boldsymbol{p}_t$ and $\boldsymbol{f}_t$ are truncated as

$$\boldsymbol{p}_t = \left[y_{t-1}^T, y_{t-2}^T, \cdots, y_{t-l}^T, u_{t-1}^T, u_{t-2}^T, \cdots, u_{t-l}^T \right]^T \tag{18}$$

and

$$\boldsymbol{f}_t = \left[y_t^T, y_{t+1}^T, \cdots, y_{t+h}^T \right]^T \tag{19}$$

where l and h are the number of lags to include. Theoretically, the CVA algorithm does not suffer when $l = h > k$ (actually, l and h just need to be larger than the largest observability index [25]). However, the state order of the system is not known a priori. Here we fit autoregressive models with several different numbers of lags to the data via recursive least squares and select l as the lag minimizing the small sample AIC criterion [26]. This ensures that enough lags are used to capture the statistically significant information in the data.

Once the states are known, the state space matrices can be estimated by multiple linear regression [7]. Details are not given here since these matrices are not explicitly used in the fault detection algorithm.

2.2.3. Process monitoring measures

A process monitoring statistic based on quantifying the variations of the CVA states (17), has been investigated by Negiz and Cinar [15] on a milk pasteurization process. Assuming normal errors, the statistic

$$T_s^2 = \boldsymbol{p}_t^T \mathbf{J}_k^T \mathbf{J}_k \boldsymbol{p}_t, \tag{20}$$

follows the distribution

$$T_{s,\alpha}^2 = \frac{k(n^2-1)}{n(n-k)} F_\alpha(k, n-k). \tag{21}$$

The T_s^2 statistic measures the variations inside the state space, and the process faults can be detected by selecting a level of significance and computing the appropriate threshold using $T_{s,\alpha}^2$.

Motivated by the residual-based statistic used in PCA and DPCA (5), we propose to measure the variations outside the state space using the statistic

$$T_r^2 = \boldsymbol{p}_t^T \mathbf{J}_q^T \mathbf{J}_q \boldsymbol{p}_t \tag{22}$$

where J_q contains the last $q = l(m_u + m_y) - k$ rows of J in Eq. (8). Assuming normality, the statistic (22) follows the distribution

$$T_{r,\alpha}^2 = \frac{q(n^2-1)}{n(n-q)} F_\alpha(q, n-q). \tag{23}$$

A weakness of this approach is that T_r^2 can be overly sensitive due to the inversion of the small values of $\boldsymbol{\Sigma}_{pp}$ in Eq. (17) [17]. This can result in a high false alarm rate. To address this concern, the threshold should be adjusted before applying the statistics for process monitoring (see next section for an example).

The residual vector of the state space model in terms of the past p_t can be calculated

$$\boldsymbol{r}_t = \left(\mathbf{I} - \mathbf{J}_k^T \mathbf{J}_k\right) \boldsymbol{p}_t, \tag{24}$$

and the variation in the residual space can be monitored using a Q statistic similar to the (D)PCA approaches

$$Q = \boldsymbol{r}_t^T \boldsymbol{r}_t. \tag{25}$$

A violation of the T_s^2 statistic indicates that the states are out-of-control, and a violation of the T_r^2 or Q statistics indicates that the characteristic of the measurement noise has changed and/or new states have been created in the process. The flexibility of the state space model and the optimality of the CVA approach suggest that the CVA states will more accurately represent dynamic process changes compared to the scores using PCA or DPCA. Other CVA-based fault detection statistics are reported in the literature [16,27].

3. Application

The Tennessee Eastman process simulator was created by the Eastman Chemical Company to provide a realistic industrial process for evaluating process control and monitoring methods [28]. The Tennessee Eastman process simulator has been widely used by the process monitoring community as a source of data for comparing various approaches [3,5,6,8,12,14,29–35]. The test problem is based on an actual chemical process where the components, kinetics, and operating conditions were modified for proprietary reasons. A diagram of the process is contained in Fig. 1. The simulation code allows 21 preprogrammed major process upsets, as shown in Table 1. The plant-wide control structure recommended in Lyman and Georgakis [36] was implemented to generate the closed loop simulated process data for each fault. For detailed discussion on the control structures of the Tennessee Eastman process simulator, please refer to Refs. [37–40].

The training and testing data sets for each fault consisted of $n = 500$ and 960 observations, respectively. Each data set started with no faults, and the faults were introduced 1 and 8 simulation hours into

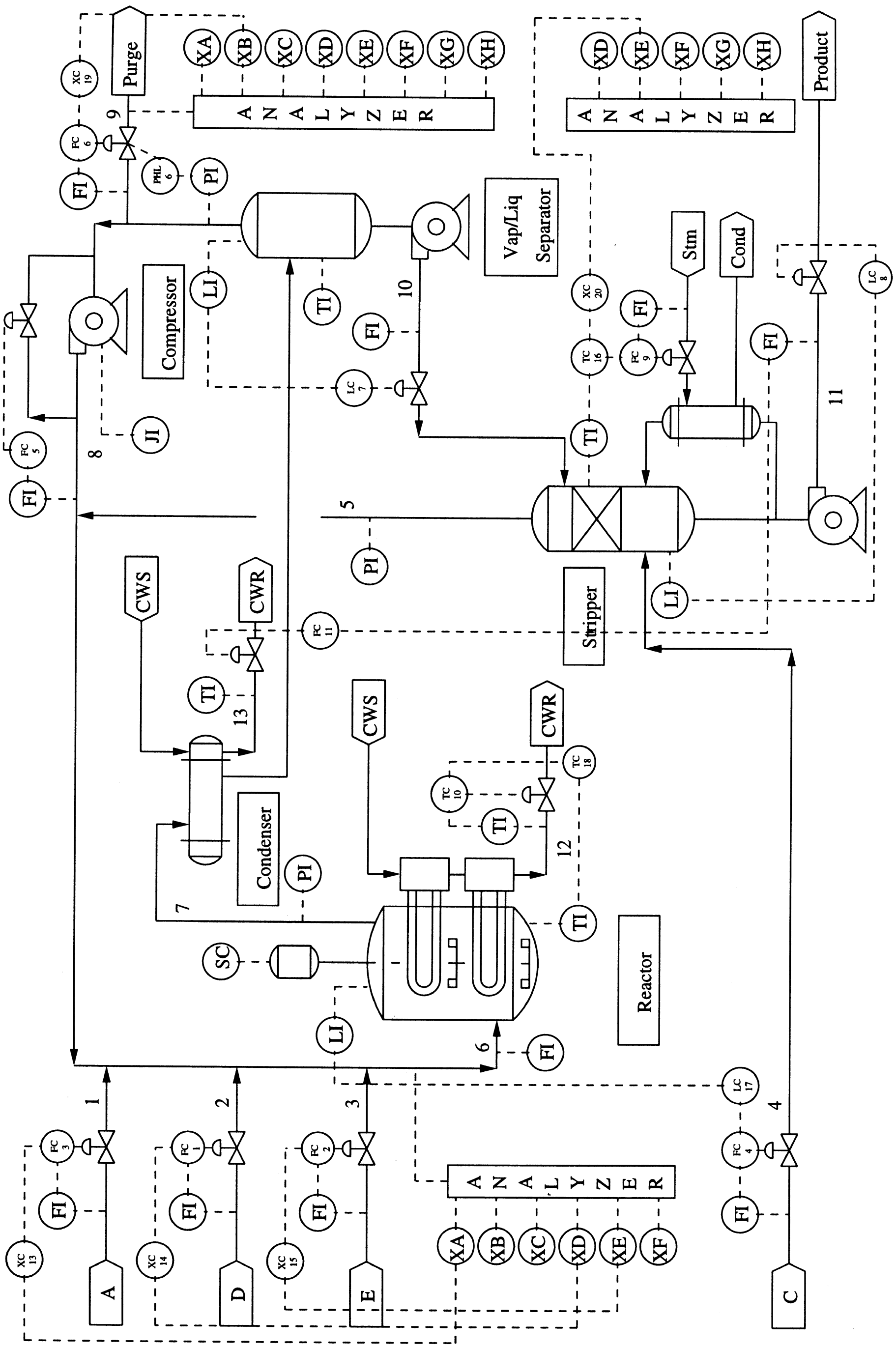

Fig. 1. A diagram of the Tennessee Eastman process simulator.

Table 1
Process faults for the Tennessee Eastman process simulator

Variable	Description	Type
IDV(1)	A/C feed ratio, B composition constant (Stream 4)	Step
IDV(2)	B composition, A/C ratio constant (Stream 4)	Step
IDV(3)	D feed temperature (Stream 2)	Step
IDV(4)	Reactor cooling water inlet temperature	Step
IDV(5)	Condenser cooling water inlet temperature	Step
IDV(6)	A feed loss (Stream 1)	Step
IDV(7)	C header pressure loss - reduced availability (Stream 4)	Step
IDV(8)	A, B, C feed composition (Stream 4)	Random variation
IDV(9)	D feed temperature (Stream 2)	Random variation
IDV(10)	C feed temperature (Stream 4)	Random variation
IDV(11)	Reactor cooling water inlet temperature	Random variation
IDV(12)	Condenser cooling water inlet temperature	Random variation
IDV(13)	Reaction kinetics	Slow drift
IDV(14)	Reactor cooling water valve	Sticking
IDV(15)	Condenser cooling water valve	Sticking
IDV(16)	Unknown	
IDV(17)	Unknown	
IDV(18)	Unknown	
IDV(19)	Unknown	
IDV(20)	Unknown	
IDV(21)	The valve for Stream 4 was fixed at the steady state position	Constant position

the run, respectively, for the training and testing data sets. All the manipulated and measurement variables except the agitation speed of the reactor's stirrer for a total of $m = 52$ variables were recorded. The data was sampled every 3 min, and the random seed was changed before the computation of the data set for each fault. Twenty-one testing sets were generated using the preprogrammed faults (Fault 1–21). In addition, one testing set (Fault 0) was generated with no faults.

Multiple faults occurring within the same time window are likely to happen for many industrial processes. The statistics for *detecting* a single fault are directly applicable for detecting multiple faults be-

Table 2
The lags and orders for the various models

Fault	ARX (l)	PCA (a)	DPCA (a)	CVA (k)
0	3	11	29	29
1	2	8	13	25
2	2	8	19	28
3	2	12	27	26
4	2	12	26	26
5	2	8	15	27
6	6	6	11	3
7	2	6	10	27
8	2	8	13	24
9	3	11	26	30
10	3	10	24	28
11	3	9	26	28
12	2	6	7	25
13	2	8	12	24
14	3	13	27	25
15	3	11	27	30
16	3	12	27	26
17	2	9	24	27
18	2	4	4	27
19	3	16	32	29
20	3	11	25	32
21	2	14	25	26

Table 3
False alarm rates for the training and testing sets

Method	Statistics	Training set	Testing set
PCA	T^2	0.002	0.014
PCA	Q	0.004	0.016
DPCA	T^2	0.002	0.006
DPCA	Q	0.004	0.281
CVA	T_s^2	0.013	0.083
CVA	T_r^2	0	0.126
CVA	Q	0.009	0.087

Table 4
Missed detection rates for the testing set

Fault	PCA T^2	PCA Q	DPCA T^2	DPCA Q	CVA T_s^2	CVA T_r^2	CVA Q
1	0.008	0.003	0.006	0.005	0.001	$\boxed{0}$	0.003
2	0.020	0.014	0.019	0.015	0.011	$\boxed{0.010}$	0.026
3	0.998	0.991	0.991	0.990	0.981	0.986	0.985
4	0.956	0.038	0.939	$\boxed{0}$	0.688	$\boxed{0}$	0.975
5	0.775	0.746	0.758	0.748	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$
6	0.011	$\boxed{0}$	0.013	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$
7	0.085	$\boxed{0}$	0.159	$\boxed{0}$	0.386	$\boxed{0}$	0.486
8	0.034	0.024	0.028	0.025	0.021	$\boxed{0.016}$	0.486
9	0.994	0.981	0.995	0.994	0.986	0.993	0.993
10	0.666	0.659	0.580	0.665	0.166	$\boxed{0.099}$	0.599
11	0.794	0.356	0.801	$\boxed{0.193}$	0.515	0.195	0.669
12	0.029	0.025	0.010	0.024	$\boxed{0}$	$\boxed{0}$	0.021
13	0.060	0.045	0.049	0.049	0.047	$\boxed{0.040}$	0.055
14	0.158	$\boxed{0}$	0.061	$\boxed{0}$	$\boxed{0}$	$\boxed{0}$	0.122
15	0.988	0.973	0.964	0.976	0.928	0.903	0.979
16	0.834	0.755	0.783	0.708	0.166	$\boxed{0.084}$	0.429
17	0.259	0.108	0.240	0.053	0.104	$\boxed{0.024}$	0.138
18	0.113	0.101	0.111	0.100	0.094	$\boxed{0.092}$	0.102
19	0.996	0.873	0.993	0.735	0.849	$\boxed{0.019}$	0.923
20	0.701	0.550	0.644	0.490	0.248	$\boxed{0.087}$	0.354
21	0.736	0.570	0.644	0.558	0.440	$\boxed{0.342}$	0.547

cause the thresholds in Eqs. (4), (6), (21), and (23) depend only on the data from the normal operating conditions (Fault 0). Multiple simultaneous faults should be detectable provided that their affects on the measured process variables do not cancel. The task of *diagnosing* multiple faults is rather challenging and the proficiencies of the fault diagnosis statistics depend on the nature of the combination of the faults [3].

All the data were autoscaled prior to the applications of PCA, DPCA, and CVA in order to avoid particular variables inappropriately dominating the dimensionality reduction procedure. The autoscaling consisted of two steps. The first step was to subtract each variable by its sample mean so that the objective is to capture the variation of the data from the mean. The second step was to divide each variable of the mean-centered data by its standard deviation. This scales each variable to unit variance.

The objective of a fault detection technique is that it is *robust* to data independent of the training set,

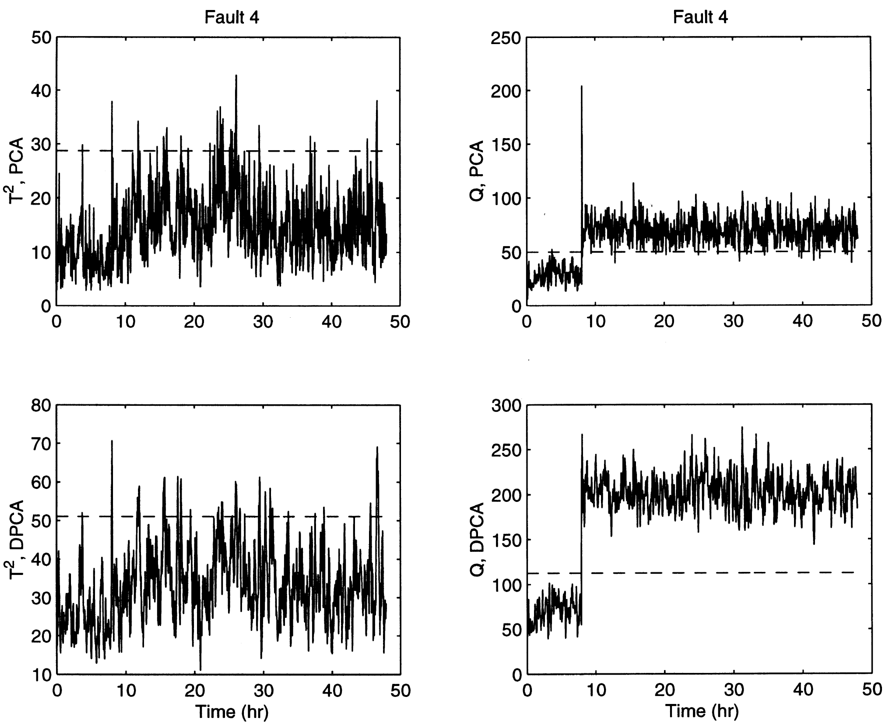

Fig. 2. The (D)PCA multivariate statistics for fault detection for Fault 4.

sensitive to all the possible faults of the process, and *prompt* to the detection of the faults. The robustness of each statistic was determined by calculating the false alarm rate during normal operating conditions for the testing set and comparing it against the level of significance upon which the threshold is based. The sensitivity of the fault detection techniques was quantified by calculating the missed detection rates for Faults 1–21 of the testing set. The promptness of the measures is based on the detection delays for Faults 1–21 of the testing set. The sensitivity and promptness of a statistic are often investigated in the literature. Interestingly, the robustness is often ignored in the literature when evaluating fault detection algorithms, although it is an important factor to consider in practice.

The orders determined for PCA, DPCA, and CVA and the number of lags l determined for DPCA and CVA are displayed in Table 2. Using a level of significance $\alpha = 0.01$, the false alarm rates of the training and testing sets were computed and tabulated in Table 3. The false alarm rates for the PCA and DPCA-based T^2 statistics are comparable in magnitude to $\alpha = 0.01$. The CVA-based statistics and the DPCA-based Q statistic resulted in relatively high false alarm rates for the testing set compared to the other multivariate statistics. The lack of robustness for T_s^2 and T_r^2 can be explained by the inversion of $\hat{\Sigma}_{pp}$ in Eq. (17). The high false alarm rates for the DPCA-based and CVA-based Q statistics may be due to a violation of the assumptions used to derive the threshold (6).

Although often done in the literature, it would not be fair to directly compare the fault detection statistics in terms of missed detection rates when they have such widely varying false alarm rates. In computing the missed detection rates for Faults 1–21 of the testing set, the threshold for each statistic was adjusted to the tenth highest value for the normal operating condition of the *testing* set. The adjusted thresholds correspond to a level of significance $\alpha = 0.01$ by considering the probability distributions of the statistics for the normal operating condition. For statistics that showed low false alarm rates, the adjustment only

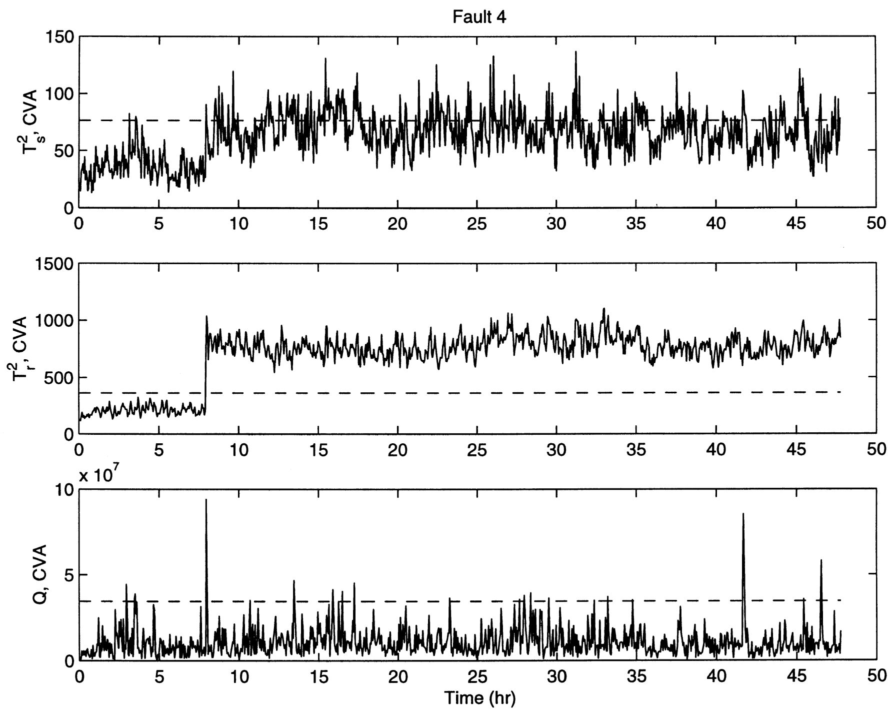

Fig. 3. The CVA multivariate statistics for fault detection for Fault 4.

shifted the thresholds slightly. For each statistic that showed a high false alarm rate, the adjustment increased the threshold by approximately 50%. Numerous simulation runs for the normal operating conditions confirmed that the adjusted thresholds indeed corresponded to a level of significance $\alpha = 0.01$. It was felt that this adjustment of thresholds provides a fairer basis for the comparison of the sensitivities of the statistics. For each statistic, the missed detection rates for all 21 faults were computed and tabulated in Table 4.

The missed detection rates for Faults 3, 9, and 15 are fairly high and nearly correspond to the level of significance for several of the statistics. No observable change in the mean or the variance could be detected by visually comparing the plots of each observation variable associated with Faults 3, 9, and 15 to the plots associated with Fault 0. Including the missed detection rates for these faults would skew the comparison of the methods, and therefore these faults will not be considered when comparing the methods.

The minimum missed detection rate achieved for each fault except Faults 3, 9, and 15 is contained in a box in Table 4. Except for the CVA Q-statistic, the statistics which quantify variations in the residual space were usually more sensitive to the faults than the statistics quantifying the variations in the score or state space. In other words, the faults usually created new states in the process rather than magnify the states during in-control operations. For example, consider Fault 4, where the missed detection rates for T_r^2 and the D(PCA)-based Q statistics are much smaller than T_s^2 and the (D)PCA-based T^2 statistics (see Table 4). The extent to which the multivariate statistics are sensitive to Fault 4 can be examined in Figs. 2 and 3.

The T_r^2 statistic produced smaller missed detection rates than the Q statistics for most faults. One of the reasons for this is that the CVA statistics had a more consistent response than the PCA statistics. This is illustrated by Fault 5, where the (D)PCA-based Q statistics dropped below the threshold 10 h after the

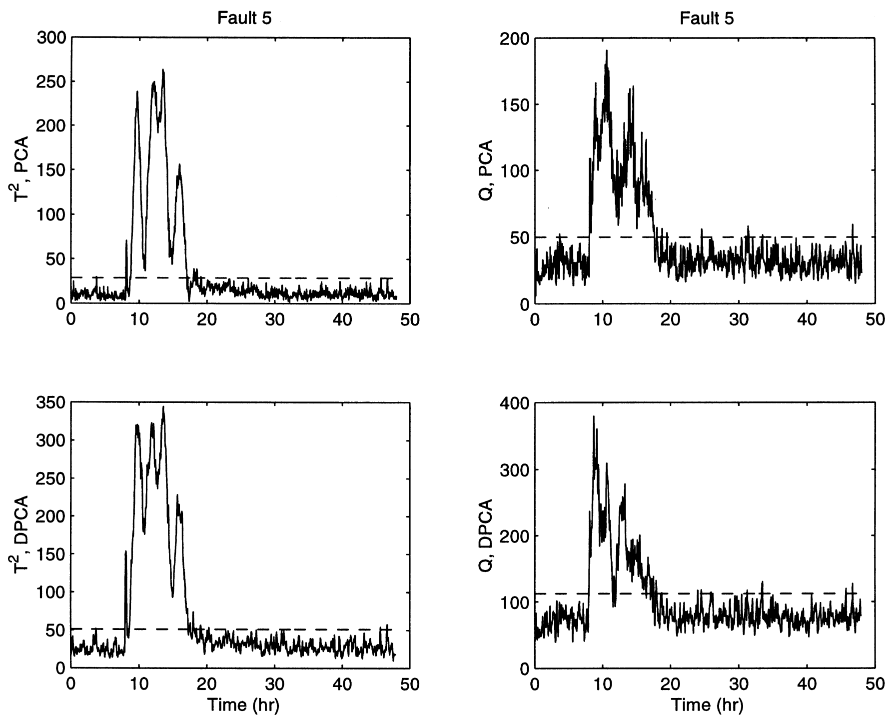

Fig. 4. The (D)PCA multivariate statistics for fault detection for Fault 5.

fault occurred whereas none of the CVA-statistics dropped below the threshold after the fault occurred (see Figs. 4 and 5). A process operator viewing the PCA-based Q statistics in Fig. 4 would likely concur that a fault entered the process and then corrected itself in less than 10 h. The T_r^2 and the other CVA-based statistics in Fig. 5 correctly show that the fault persists throughout the simulation time.

While the T_r^2 statistic with the rescaled threshold had the lowest missed detection rate for most faults for the Tennessee Eastman process, it should always be used as part of a fault detection strategy which has at least one other statistic. In particular, a fault that does not affect the states in the T_r^2 statistic will be invisible to this statistic.

Since false alarms are inevitable, an out-of-control value of a statistic can be the result of a fault or of a false alarm. In order to decrease the rate of false alarms, a fault can be indicated only when several consecutive values of a statistic have exceeded the threshold. In computing the detection delays for the statistics in Table 5, a fault is indicated only when six consecutive statistic values have exceeded the threshold, and the detection delay is recorded as the first time instant in which the threshold was exceeded. Assuming independent observations and $\alpha = 0.01$, this corresponds to a false alarm rate of $0.01^6 = 1 \times 10^{-12}$.

The detection delays for all 21 faults are listed in Table 5. These results were obtained by applying the same thresholds as those used to determine the missed detection rates. For the multivariate statistics, an examination of Tables 4 and 5 reveals that the fault detection methods exhibiting small detection delays usually exhibit small missed detection rates and vice versa. The smallest detection delays were usually exhibited by either the CVA-based T_r^2 or Q statistics.

It is instructive to compare these results with those of earlier studies. Our static PCA results for the Tennessee Eastman process qualitatively agree with those of Georgakis et al. [30]. That the residual-based DPCA statistic was more sensitive than the score-

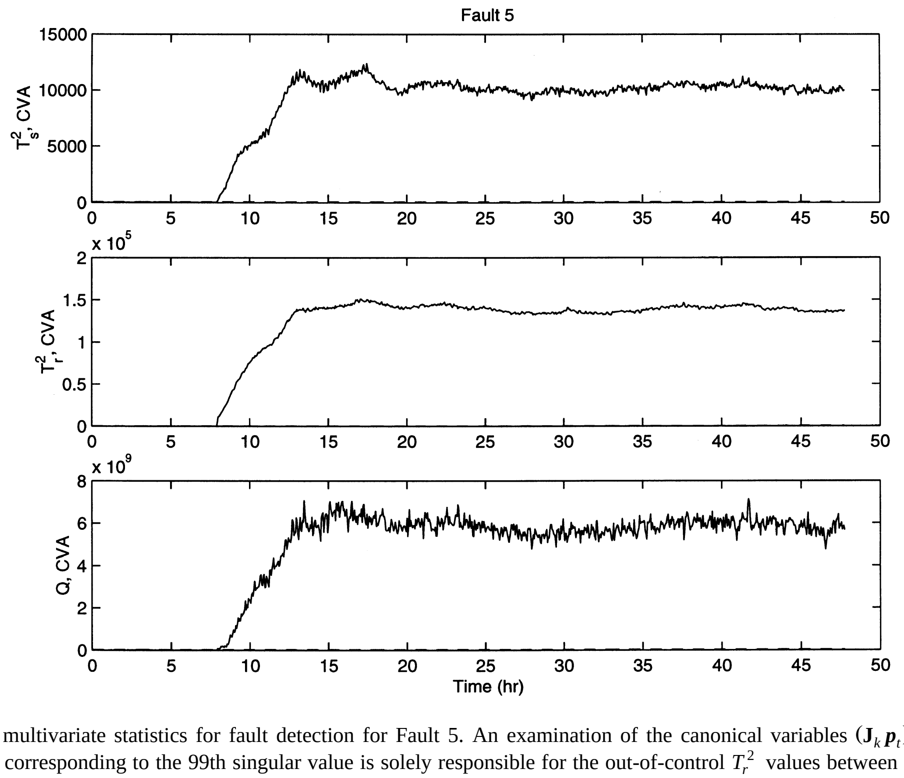

Fig. 5. The CVA multivariate statistics for fault detection for Fault 5. An examination of the canonical variables $(\mathbf{J}_k \boldsymbol{p}_t)$ reveals that the canonical variable corresponding to the 99th singular value is solely responsible for the out-of-control T_r^2 values between 10 and 40 h after the fault occurred.

Table 5
Detection delays (minutes) for the testing set

Fault	PCA T^2	PCA Q	DPCA T^2	DPCA Q	CVA T_s^2	CVA T_r^2	CVA Q
1	21	9	18	15	$\boxed{6}$	9	$\boxed{6}$
2	51	$\boxed{36}$	48	39	39	45	75
3	–	–	–	–	–	–	–
4	–	9	453	$\boxed{3}$	1386	$\boxed{3}$	–
5	48	3	6	6	3	3	$\boxed{0}$
6	30	3	33	3	3	3	$\boxed{0}$
7	3	3	3	3	3	3	$\boxed{0}$
8	69	$\boxed{60}$	69	63	$\boxed{60}$	$\boxed{60}$	63
9	–	–	–	–	–	–	–
10	288	147	303	150	75	$\boxed{69}$	132
11	912	33	585	$\boxed{21}$	876	33	81
12	66	24	9	24	6	6	$\boxed{0}$
13	147	$\boxed{111}$	135	120	126	117	129
14	12	$\boxed{3}$	18	$\boxed{3}$	6	$\boxed{3}$	$\boxed{3}$
15	–	2220	–	–	2031	–	–
16	936	591	597	588	42	$\boxed{27}$	33
17	87	75	84	72	81	$\boxed{60}$	69
18	279	252	279	252	249	$\boxed{237}$	252
19	–	–	–	246	–	$\boxed{33}$	–
20	261	261	267	252	246	$\boxed{198}$	216
21	1689	855	1566	858	$\boxed{819}$	1533	906

based DPCA statistic for the Tennessee Eastman process agrees with Ku et al. [6]. While Ku et al. found that DPCA outperformed PCA for the faults considered in their study, we found that DPCA gave similar performance as PCA for most faults (see Table 4). Chen and McAvoy found that the multiway PCA algorithm gave somewhat smaller detection delays than a DPCA algorithm for three faults [14]. In our study, at least one of the CVA statistics gives a smaller or equal detection delay than for the DPCA statistics for all but one of the original faults (see Table 5). Also, the CVA-based statistics gave much smaller detection delays for nearly all the faults than those reported by Chen and McAvoy [14] for multiway PCA and DPCA.

4. Conclusions

The Tennessee Eastman process simulator was used to compare PCA, DPCA, and CVA for detecting faults in terms of sensitivity, promptness, and robustness. This comparison included a proposed residual space CVA statistic (T_r^2) for fault detection. This appears to be the first application of this residual space CVA statistic for detecting faults.

Except for the CVA-based Q statistic, the statistics quantifying variations in the residual space (CVA T_r^2, PCA Q, and DPCA Q statistics) were more sensitive for most faults than the statistics quantifying the variations in the score or state space (CVA T_s^2, PCA T^2, and DPCA T^2 statistics). The statistics exhibiting a small missed detection rate usually exhibited a small detection delay and vice versa. The smallest detection delays were usually exhibited by either the CVA-based T_r^2 or Q statistics.

Based on the original thresholds, the PCA- and DPCA-based T^2 statistics gave lower false alarm rates than the DPCA-based Q and the CVA statistics. The lack of robustness of CVA was due to the CVA statistics being overly sensitive to a matrix inversion step. A strength of applying the proposed CVA-based T_r^2 statistic is that it is sensitive and prompt; however, its threshold must be adjusted to achieve robustness. With the thresholds defined so that the techniques had similar missed detection rates, DPCA had similar performance as to PCA for most faults.

Acknowledgements

This work was supported by International Paper.

References

[1] J.V. Kresta, T.E. Marlin, J.F. MacGregor, Can. J. Chem. Eng. 69 (1991) 35–47.
[2] M.J. Piovoso, K.A. Kosanovich, R.K. Pearson, in: Proc. of the American Control Conf., IEEE Press, Piscataway, NJ, 1992, pp. 2359–2363.
[3] A.C. Raich, A. Cinar, Chemom. Intell. Lab. Syst. 30 (1995) 37–48.
[4] T. Kourti, J.F. MacGregor, J. Qual. Technol. 28 (1996) 409–428.
[5] E.L. Russell, L.H. Chiang, R.D. Braatz, Data-driven Techniques for Fault Detection and Diagnosis in Chemical Processes, Springer, London, UK, 2000.
[6] W. Ku, R.H. Storer, C. Georgakis, Chemom. Intell. Lab. Syst. 30 (1995) 179–196.
[7] W.E. Larimore, Statistical Methods in Control and Signal Processing, Marcel Dekker, New York, 1997.
[8] D.M. Himes, R.H. Storer, C. Georgakis, in: Proc. of the

American Control Conf., IEEE Press, Piscataway, NJ, 1994, pp. 1279–1283.
[9] M.H. Kaspar, W.H. Ray, AIChE J. 38 (1992) 1593–1608.
[10] J.F. MacGregor, in: Proc. of the IFAC Conference on Advanced Control of Chemical Processes, Pergamon, New York, 1994, pp. 427–435.
[11] A.C. Raich, A. Cinar, in: Proc. of the IFAC Conf. on Advanced Control of Chemical Processes, Pergamon, New York, 1994, pp. 427–435.
[12] A.C. Raich, A. Cinar, AIChE J. 42 (1996) 995–1009.
[13] H. Tong, C.M. Crowe, AIChE J. 41 (7) (1995) 1712–1722.
[14] G. Chen, T.J. McAvoy, J. Process Control 8 (1997) 409–420.
[15] A. Negiz, A. Cinar, AIChE J. 43 (1997) 2002–2020.
[16] Y. Wang, D.E. Seborg, W.E. Larimore, in: IFAC ADCHEM, Alberta, Canada,1997, June.
[17] J.E. Jackson, G.S. Mudholkar, Technometrics 21 (1979) 341–349.
[18] J.E. Jackson, A User's Guide to Principal Components, Wiley, New York, 1991.
[19] B.M. Wise, N.B. Gallagher, J. Process Control 6 (1996) 329–348.
[20] W.E. Larimore, in: Proc. of the IEEE Conf. on Decision and Control, IEEE Press, Piscataway, NJ, 1990, pp. 635–639.
[21] H. Akaike, IEEE Trans. Autom. Control 19 (1974) 667–674.
[22] H. Akaike, SIAM J. Control 13 (1975) 162–173.
[23] H. Akaike, in: R.K. Mehra, D.G. Jainiotis (Eds.), System Identification: Advances and Case Studies, Academic Press, New York, 1976, pp. 27–96.
[24] W.E. Larimore, ADAPTx Automated System Identification Software Users Manual, Adaptics, McLean, VA, 1996.
[25] P. Van Overschee, B. De Moor, Subspace Identification for Linear Systems: Theory — Implementation — Applications, Kluwer Academic Publishers, Norwell, MA, 1996.
[26] W.E. Larimore, Signal Process. 52 (1996) 131–144.
[27] W.E. Larimore, in: IFAC ADCHEM, Alberta, Canada, 1997, June.
[28] J.J. Downs, E.F. Vogel, Comput. Chem. Eng. 17 (1993) 245–255.
[29] H.B. Aradhye, B.R. Bakshi, R. Strauss, in: AIChE Annual Meeting, 1999, Paper 224g.
[30] C. Georgakis, B. Steadman, V. Liotta, in: Proc. of the 13th IFAC World Congress, IEEE Press, Piscataway, NJ, 1996, pp. 97–101.
[31] J. Gertler, W. Li, Y. Huang, T. McAvoy, AIChE J. 452 (1999) 323–334.
[32] I. Hashimoto, M. Kano, K. Nagao, in: AIChE Annual Meeting, 1999, Paper 224a.
[33] W.E. Larimore, D.E. Seborg, Short Course: Process Monitoring and Identification of Dynamic Systems Using Statistical Techniques, Los Angeles, CA, 1997.
[34] A.C. Raich, A. Cinar, in: Proc. of the 13th IFAC World Congress, IEEE Press, Piscataway, NJ, 1996, pp. 283–288.
[35] L.H. Chiang, E.L. Russell, R.D. Braatz, Chemom. Intell. Syst. (2000) in press.
[36] P.R. Lyman, C. Georgakis, Comput. Chem. Eng. 19 (1995) 321–331.
[37] D.R. Vinson, C. Georgakis, J. Fossy, in: Proc. of the American Control Conf., IEEE Press, Piscataway, NJ, 1995, pp. 250–254.
[38] T.J. McAvoy, Y. Nan, G. Chan, in: Proc. of the American Control Conf., IEEE Press, Piscataway, NJ, 1995, pp. 240–244.
[39] Y. Nan, T.J. McAvoy, K.A. Kosanovich, M.J. Piovoso, in: Proc. of the American Control Conf., IEEE Press, Piscataway, NJ, 1993, pp. 1900–1904.
[40] T.J. McAvoy, Comput. Chem. Eng. 22 (1998) 1543–1552.