Enhanced plant fault diagnosis based on the characterization of transient stages

Isaac Monroy[a], Raul Benitez[b], Gerard Escudero[c], and Moisès Graells[a].*

[a]Chemical Engineering Department (DEQ-CEPIMA), [b]Automatic Control Department (ESAII), [c]Software Department (LSI).

EUETIB, Comte d'Urgell 187, 08036-Barcelona.

Universitat Politècnica de Catalunya (UPC).

ABSTRACT.

Data-based process fault diagnosis is enhanced by using better characterization of the transient stages of the faults. After monitoring Normal operating conditions (NOC) data and projecting the model onto Abnormal operating conditions (AOC), statistical indices (T^2 and Q) are used to characterize fault transient stages by assessing faults' delay and span. Hence, a new NOC PCA model is constructed using a data subset based on the transient window and projected onto the NOC and AOC transient data. Model construction is finally achieved by applying a classification technique –Artificial Neural Networks (ANN) or Support Vector Machines (SVM), which are compared– to the

[*] Corresponding author: moises.graells@upc.edu

training matrix. Validation is done using simulated on-line data sets and using the Tennessee Eastman process as case study. The use of different statistical indices and classification methods is exposed and discussed, and the improvement obtained in the diagnosis performance is presented and analysed.

KEYWORDS. Chemical Processes, On-line fault diagnosis, Transient stages, Tennessee-Eastman, PCA, ANN, SVM.

1. Introduction

Fault diagnosis is a challenging problem in industrial and engineering practice. Reliable monitoring and fault diagnosis approaches for industrial systems address challenges from false alarms, delayed response and incorrect fault identification. The correct diagnosis and prediction of incipient system anomalies reduces operational costs and enhances the safety.

Most of the learning-based fault diagnosis approaches face this problem by using data corresponding to the steady state of the faults after their occurrence [1,2,3,4], random observations [5,6,7] or considering the immediate sample to the disturbance generation but not the real detection time of the fault [8,9]. Nevertheless, scarce attention has been paid to the data used for producing the classification models (or the methods used to select these data). It is quite common that works reporting interesting results on this field hardly mention this point [10,11,12,13,14,15,16,17,18,19]. In addition, many of these methods, assume that the process is in a known, well characterized state (e.g., a nominal steady state) when diagnosis is started. This assumption may result untenable for agile processes that undergo transitions [20].

Some approaches to process modelling, alarm management, fault diagnosis and other automation systems may be ineffective during transitions because they are usually configured assuming a single state of operation. When the plant moves out of that state, these applications may lead to false alarms even when a desired change is occurring. Thus, some frameworks have been already developed for managing transitions and detecting faults [21,22].

In order for an automatic fault diagnosis system to become a practical support tool for decision-making during plant operation, on-line detection and identification of the early transient stages of the fault evolution is required. Therefore, on-line monitoring of transient states is important to detect abnormal events and enable timely recovery. Previous attempts to deal with this problem include the determination of detection delays using different statistical indexes such as the Hotelling's T^2 and Q statistic in Principal Component Analysis (PCA) and Correspondence Analysis [23,24,25,26].

This work focuses on data-driven fault diagnosis and addresses the improvement of on-line diagnosis performance by introducing a method that takes advantage of transient stages data to model faulty / abnormal plant behavior and produce enhanced classification models.

First, a monitoring and detection step is required. PCA has been selected for this first step because its effectiveness in detecting process abnormalities and has been applied to simulated data in order to obtain the scores (principal component variables that are the axes of a new coordinate system and represent the directions of maximum variability), the Hotelling's T squared and Q statistics, and thereby the delays in the fault advent and the transition period for their real occurrence.

In addition, off-line learning of the transient stages during fault evolution is implemented. Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are used in this work as classification algorithms and are accordingly compared. Then, classification models are applied to validation data sets, acquired by simulating the process under Nominal Operation Conditions (NOC) and Abnormal Operation Conditions (AOC). Diagnosis is measured considering simulation samples since the moment in which the fault is harassed until the end of the simulation as an off-line procedure.

This paper is organized as follows. Problem formulation is next presented in section 2. Materials and Methods are developed in section 3 including the proposed methodology and the case study in which is applied. Section 4 exposes the results. Section 5 exposes the main discussions and finally the conclusions of the work are presented in the last section.

2. Problem formulation

The on-line fault diagnosis problem is presented as follows. First, raw data sets $\mathbf{X_i}$ are obtained from plant process data corresponding to *nv* observed variables at *ns* sampling times. It is asummed that each data set $\mathbf{X_i}$ corresponds to a certain dynamical regime of the plant (or scenario) under normal or abnormal operating conditions (NOC, AOC). In particular, AOC regimes include *nf* faults, which results in a global data set $\mathbf{\Omega}$. Therefore, $\mathbf{\Omega} = \left\{ \mathbf{X_i} \middle| \mathbf{X_i} \in \mathbb{R}^{ns \times nv}; i = 0 \ldots nf \right\}$, where $\mathbf{X_0}$ represents process data under NOC.

$$\mathbf{X_i} = \begin{pmatrix} x^i_{11} & x^i_{12} & \cdots & x^i_{1nv} \\ x^i_{21} & x^i_{22} & \cdots & x^i_{2nv} \\ \cdots & \cdots & \cdots & \cdots \\ x^i_{ns1} & x^i_{ns2} & \cdots & x^i_{nsnv} \end{pmatrix} \qquad i = 0 \ldots nf \tag{1}$$

The aim of the present method is to develop a novel model-based fault diagnosis method using data from transient stages of the fault evolution. Specifically, a classification algorithm on a training set **TR** that takes into account fault transient data is used, where $\mathbf{TR} = \left\{ \mathbf{R_i} \middle| \mathbf{R_i} \in \mathbb{R}^{nt \times np}; i = 0 \ldots nf \right\}$, being *nt* the number of transient observations and *np* the number of extracted variables. The basic assumption is that the use of transient data will improve the data classification performance with respect to standard methods that consider data in steady regimes when the fault has already been fully developed. A detailed description of the construction of the training set is presented in the methodology section. Once the training set is obtained, we construct classification models so that they can be applied to on-line data and then detect and diagnose any kind of AOC in the process. Such generation of models can be represented by the following expression:

$$\mathbf{M_i} = g_i(\mathbf{R_i}, \mathbf{H_i}) \qquad i = 0...nf \tag{2}$$

where $\mathbf{R_i}$ represents the training data for each scenario **i** and $\mathbf{H_i}$ is an occurrence matrix that characterizes the faults occurring at each sample time by means of binary elements h_{si} indicating whether or not a fault is occurring at each sample *s* [5,27].

$\mathbf{M_i}$ represents the classification models per each scenario **i** and the function g_i represents the calculation of each model by learning algorithms such as ANN and SVM.

Classification models will be applied to on-line validation sets of a fixed number of samples until the end of the simulation in the academic case study. Validation sets are joined in the validation ($\mathbf{V_i}$) matrix per scenario for an easier representation. The global set of validation matrices is represented by **VL.**

Thus, the binary matrix $\mathbf{D_i}$, indicating the result of the diagnosis, is obtained from the application of all these models to the validation sets $\mathbf{V_i}$.

$$\mathbf{D_i} = y_i(\mathbf{M_i}, \mathbf{V_i}), \qquad i = 0...nf \tag{3}$$

Hence, the diagnosis performance is calculated by means of the *F1* index [27,28,29] and considering all samples from the fault occurrence. Such index was selected in order to have a single assessment taking into account two complementary concepts: precision (positive predictive value) and recall (sensitivity).

The problem to be solved is to maximize the matching of matrix $\mathbf{D_i}$ to matrix $\mathbf{H_i}$ expressed in terms of the *F1* index, and the paper proposes a methodology to produce models $\mathbf{M_i}$ based on the analysis of transient states that enhances such diagnosis performance.

3. Materials and Methods

3.1. Methodology

This work proposes an on-line fault diagnosis methodology for continuous processes based on the analysis and characterization of each fault transient stage. A general diagram of the construction of the training set **TR** is described in Fig. 1. The training set per class or scenario ($\mathbf{R_i}$) is constructed from the $\mathbf{X_i}$ matrices and represents a $nt\times np$ data matrix considering the different *nf* scenarios to diagnose and the NOC case.

The methodology takes into account from the process monitoring and fault detection up to the identification and diagnosis of the abnormalities in the process.

First, process data are acquired and organized in $\mathbf{X_i}$ sets, where $i=0...nf$. Centering and scaling are applied to the data $\mathbf{\Omega}$ as a preprocessing step, rendering as result the global set $\mathbf{\Omega}^*$, where $\mathbf{\Omega}^*=\left\{\mathbf{X_i^*}\middle|\mathbf{X_i^*}\in\mathbb{R}^{ns\times nv};i=0...nf\right\}$. Then, PCA is used in order to detect any abnormality in the process by means of statistical indices. Specifically, PCA is applied to $\mathbf{NOC}^*$ set, where $\mathbf{NOC}^*=\left\{\mathbf{X_0^*}\middle|\mathbf{X_0^*}\in\mathbb{R}^{ns\times nv}\right\}$ and once a detection model is obtained, $\mathbf{AOC}^*$ set $\left(\mathbf{AOC}^*=\left\{\mathbf{X_i^*}\middle|\mathbf{X_i^*}\in\mathbb{R}^{ns\times nv},i=1,2...nf\right\}\right)$ data are projected onto such model so that the T_{si}^2 and Q_{si} statistics are obtained. As far as the values of these statistical indexes remain within a certain range, the plant is considered to be under normal operating conditions. When the indices exceed their control limits, the plant is assumed to enter into an anormal dynamical regime.

Next stage of the methodology is the construction of classification models taking into account the transient stages. Besides, a feature extraction is done before the model construction.

Transient data is represented by the global set **ω**, composed of the transient sets $\mathbf{XX_i}$ per each scenario, where $\boldsymbol{\omega} = \left\{ \mathbf{XX_i} \middle| \mathbf{XX_i} \in \mathbb{R}^{nt \times nv}; i = 0,1,...nf \right\}$ and $\mathbf{XX_i} = \left\{ x_{isv} \middle| x_{isv} \in \mathbb{R}^{nt \times nv}; \quad s = t_i^0 + nt - 1; \quad v = 1,...nv; \quad i = 0...nf \right\}$. Thus, transient data is given by a transient state window characterized by the corresponding samples start (ti0) and a span (nt) that has been assumed common to all the faults. This assumption is discussed later on. Starting times and samples t_i^0 at which each fault i is detected are determined by the indexes T_{si}^2 and Q_{si} as they exceed the control limits:

$$t_i^0 = \min_s \left\{ s \mid T_{si}^2 > T_{\lim}^2 \wedge Q_{si} > Q_a \right\} \qquad \forall i \tag{4}$$

Such transient data subsets allow obtaining a new PCA model, and therefore the P loadings, based on transient NOC data (**TNOC**). These loadings will be projected onto **TNOC** and transient AOC data (**TAOC)** of the same sample window per fault. The results of this projection are the score matrices $\mathbf{R_i}$. Scores matrices are then joined in an only matrix **TR** and used as training set and input of the classification algorithms. Parameter tuning of such algorithms is also considered and carried out. Afterwards, the classification models $\mathbf{M_i}$ are validated using simulated on-line data sets and finally, the diagnosis performance is evaluated in terms of the *F1* index.

The next subsections will explain in more detail the two main steps of the methodology namely the transient stages identification and the transient-based models construction.

3.1.1. Process monitoring: characterization of the transient stage and transient-based training set construction.

A process monitoring technique (PCA) is applied to a set of NOC observations ($\mathbf{X_0}$). Both **NOC** and **AOC** data sets $\left(\mathbf{NOC}=\{\mathbf{X_0}\};\mathbf{AOC}=\{\mathbf{X_i}\,|\,i=1...nf\}\right)$, included in the global data set $\mathbf{\Omega}$ are centered and scaled. Such normalization step renders as result the global data set $\mathbf{\Omega}^*$ as previously mentioned, composed of the $\mathbf{NOC}^*$ and $\mathbf{AOC}^*$ sets. PCA model (P^* loadings) is constructed on a centered and scaled NOC data set ($\mathbf{NOC}^*=\mathbf{X_0}^*$). Such model is applied to centered and scaled AOC data sets ($\mathbf{X_i}^*$, $i=1...nf$) so that T_{si}^2 and Q_{si} indices are obtained. Such statistical indices are used for fault detection when the $\mathbf{AOC}^*$ data ($\mathbf{X_i}^*$) are projected with the constructed PCA model (P^*) and T_{si}^2 and Q_{si} overstep their control limits T_{lim}^2 and Q_a. Even more, these T_{si}^2 and Q_{si} indices allow determining fault detection delays and the right moment in which a process disturbance is developing and expressing itself in the process. In this way, the starting point of the transient stage (t_i^0) is also identified. T_{si}^2 and Q_{si} indices are unifyed respectively in the **T** and **Q** matrices as shown in Figure 1.

Regarding both monitoring and fault detection indices, Q is faster than T^2 in detecting the delay from the fault event to the moment in which there is statistical evidence of the fault in the plant data as it will be corroborated in the results section. This is due to the fact that AOC data are not included in the PCA model in terms of monitoring and in addition, Q statistic is used to detect new events that are not taken into account in the model subspace [34].

PCA is also used as a feature extraction technique that allows reducing data dimensionality. The target consists on taking the observations just in the point where the abnormality is detected by the Q statistic until a certain sample window (*nt*). The *nt* used in this work is assumed to be the same for all faults. Seemingly, the value adopted has been decided after some preliminary assays commented later on.

Transient data sets under NOC and AOC (**TNOC** and **TAOC** sets from now on) are gathered in a whole transient set ($\boldsymbol{\omega}$) for representation purposes.

A second PCA model is constructed again taking NOC data ($\mathbf{X_0}$), but this time using *nt* observations, giving as result the **TNOC** set $\left(\mathbf{TNOC} = \{\mathbf{XX_0}\}\right)$. Therefore, the dimensions of this matrix $\mathbf{XX_0}$ will be $nt \times nv$ (number of transient observations times number of variables). As in the previous step, these data are centered and scaled, representing the $\mathbf{TNOC}^*$ set ($\mathbf{TNOC}^* = \mathbf{XX_0}^*$). PCA loadings (P) are obtained from applying PCA to these normalized data $\mathbf{XX_0}^*$. A NOC scores matrix ($\mathbf{R_0}$) is obtained when the normalized NOC data set ($\mathbf{XX_0}^*$) is projected onto the PCA model by means of this equation $\mathbf{R}_0 = \mathbf{XX}_0^{*\prime}\ \mathbf{P}$.

In the same way, AOC data sets are constructed considering the transient window when taking the nt number of samples from the historic data sets $\mathbf{X_i}$ and starting with the observation where the anomaly is properly detected by the Q statistic. These transient data sets are represented by the **TAOC** set $\left(\mathbf{TAOC} = \{\mathbf{XX_i} \mid i = 1,\ldots nf\}\right)$. Such AOC transient data sets ($\mathbf{XX_i}$) are also centered and scaled and joined in the $\mathbf{TAOC}^*$ set, where $\left(\mathbf{TAOC}^* = \{\mathbf{XX_i^*} \mid i = 1,\ldots nf\}\right)$. Both $\mathbf{TNOC}^*$ and $\mathbf{TAOC}^*$ sets are also gathered in the ($\boldsymbol{\omega}^*$) set as is represented in Figure 1.

$\mathbf{XX_i}^*$ matrices are then projected onto the PCA model (P loadings) so as to obtain AOC score matrices ($\mathbf{R_i}$, *i=1…nf*) by means of the same expression used for the NOC score matrix $\left(\mathbf{R}_i = \mathbf{XX}_i^{*\prime}\ \mathbf{P}\right)$.

Unifying the scores matrices for both NOC and AOC cases:

$$R_i = \left\{ r_{isp} \middle| r_{isp} \in \mathbb{R}^{ns \times np}; \quad s = t_i^0 + nt - 1; \quad p = 1,\dots np; \quad i = 0\dots nf \right\}.$$

where $np < nv$ represents the reduced number of extracted variables or specifically the retained principal components.

Training set (**TR**) is then obtained by gathering all the previous score matrices $\mathbf{R_i}$ $\left(\mathbf{TR} = \left\{ \mathbf{R_i} \middle| \mathbf{R_i} \in \mathbb{R}^{nt \times np}, \quad i = 0,1,2\dots nf \right\}\right)$ and used as the input of the classification algorithms.

In the next paragraphs the calculation of the Q and Hotelling T^2 statistics is explained.

Suppose that s samples are available and that v is the number of measured variables in each sample. Let x and S be the sample mean vector and covariance matrix, respectively, of these observations. The Hotelling T^2 statistic is defined in the following way:

$$T^2 = \left(X - \overline{X}\right)' S^{-1} \left(X - \overline{X}\right) \qquad (5)$$

and the corresponding upper control limit (T_{lim}^2) for this statistic [30] is:

$$T_{lim}^2 = \frac{v(s+1)(s-1)}{s^2 - sv} F_{\alpha,v,s-v} \qquad (6)$$

On the other hand, the Q plot, which is calculated with the sum of the squared residuals, represents the squared distance of each observation to this plane [31]. Q statistic calculation is given by eq 7.

$$Q = \sum_{v=1}^{v} e_m(v)^2, \qquad e = X - RP' \qquad (7)$$

Where P is the component matrix (loading matrix or eigenvectors matrix) and R is the scores matrix obtained by the product of the data matrices X times P. In fact, the scores plot presents the projection of each observation onto the reduced plane defined by the principal components. The control limit of Q statistic (Q_α) is calculated according to the equation in [32].

$$Q_\alpha = \theta_1 \left[1 - \frac{\theta_2 h_0 (1-h_0)}{\theta_1^2} + \frac{z_\alpha (2\theta_2 h_0^2)^{1/2}}{\theta_1} \right]^{\frac{1}{h_0}} \tag{8}$$

$$\theta_1 = \sum \lambda_i, \qquad \theta_2 = \sum \lambda_i^2, \qquad \theta_3 = \sum \lambda_i^3, \qquad h_0 = 1 - \frac{2\theta_1 \theta_3}{3\theta_2^2}$$

Where λ represents the eigenvalues. Both T^2 and Q statistics are indicators of processes "normality" when their values are below the control limits ($Q \leq Q_a$ and $T^2 \leq T_{lim}^2$) and they are used to determine the transient stages of the process when occurring faults. The Q statistic specifically quantifies the lack of fit between the sample and the model and denotes the distance of the sample from the nominal operation surface [33].

3.1.2. Plant fault modeling

Classification models for each scenario ($\mathbf{M_i}$) are constructed by off-line learning of the fault transient stages once these stages have been located using the PCA indices. This step constitutes the data-driven modeling from the training data.

The models for the classification of faults, properly called fault diagnosis, are obtained using standard machine-learning algorithms (ANN, SVM) as classification techniques. The input of the algorithms is the training set **TR**, obtained by gathering the score matrices $\mathbf{R_i}$, as mentioned in the last subsection. The structure of the ANN and the kernel function in SVM must be fixed. Both techniques are applied with comparative purposes.

The parameters to decide when using ANN are the number of input nodes, the number of hidden nodes and the transfer function in the layer (typically tangent sigmoidal function). The number of inputs will depend on the number of PC to retain; on the other hand, the number of hidden nodes can be optimized. Regarding SVM, support vectors per class are obtained for each type of kernel so that the one with the best performance is selected.

Finally, the network and the support vector models are submitted to on-line validation data sets ($\mathbf{V_i}$) obtained by simulations establishing the number of samples in which the classifiers will be applied. In fact, the real validation sets ($\mathbf{TT_i}$) are constituted by the score matrices ($\mathbf{TT_i}$=$\mathbf{V_i}$´ P) obtained from the projection of these original validation data sets onto the NOC PCA model (P loadings). Validation score matrices $\mathbf{TT_i}$ are gathered into a global test set ($\boldsymbol{\tau}$).

This situation simulates the reality of continuous processes, for which monitoring, detection and diagnosis (model application) tasks have to be implemented with data that are being obtained on-line. An important decision to be made for the prompt abnormality detection is to choose the number of observations to use for validating the models or network. The diagnosis performance is calculated with the F1 index in an off-line manner as mentioned in section 2.

3.2. Case study

The proposed fault diagnosis method is tested using the Tennessee Eastman Process (TEP) [35] and applied to all the faults presented there. TEP consists of 52 process variables (*nv=52*) and 20 faults (*nf=20*) to be diagnosed. Only 11 of these process variables are manipulated (XMV as originally reported), while the rest are simply measured (41 XMEAS according to the same nomenclature). Process variables are given in Table 1. Table 2 provides the account of the 20 faults or disturbances including those not described and reported as unknown in the original paper (faults 16 to 20). Some faults are quite comparable and they only differ on the type of variation (step or random, which is the case of Fault 3 and 9).

Fifty-hour simulation runs having the fault occurred by the second hour have been run for obtaining source data sets ($\mathbf{X_i}$, where *i=0,1,...20*) at a sampling time of one minute ($\mathbf{X_i}$*Î R$^{3000´\ 52}$). The PCA model is obtained using these observations of the process under NOC, previously centered and scaled. Then, the model is projected for monitoring purposes not only to these NOC data but also to the AOC observations in order to obtain the T^2 and Q statistics for each state of the process (T_{si}^2 and Q_{si}), and thereby identify the fault delays and the transient stages.

Classification models are constructed with observations since the sample where faults are detected by the Q statistic and first considering a *nt*=200 samples per scenario. PCA is applied to the centered and scaled 200 observations corresponding to the process under NOC and the corresponding 200 observations per fault ($\mathbf{XX_i}^* \in R^{200 \times 52}$, i=0...*nf*) are projected onto the resulted PCA model so that the score matrices are obtained ($\mathbf{R_i}$). These matrices constitute the training set (**TR**), composed of 2200 observations, which is the input of the classification algorithms (ANN and SVM).

A comparison between the diagnosis results obtained using transient models (TRM, those obtained by means of the approach introduced in this paper) and steady-state models (SSM, those obtained with random observations, observations from the immediate time of the harrassed fault and observations from the steady state) is next addressed and presented in the results section. Such models are applied to on-line validation sets of ten samples during a simulation of 500 samples, corresponding to 500 minutes of faulty state ($\mathbf{V_i} \in R^{500 \times 52}$, i=0...nf).

Regarding execution, the PCA model used in this work was implemented in Matlab. ANN and SVM^{light} [36] were also applied using the standard Matlab toolboxes: Neural Network and MATLAB MEX-interface.

4. Results

4.1. Characterization of the transient regime

PCA model is constructed with the raw data sets under NOC. Two principal components are retained by the broken-stick rule (26% of variance in the original nv).

Such model is then projected into normalized data sets under NOC and AOC with the fault produced at second hour. Therefore, T^2 and Q statistics are calculated. Figure 2 shows the T^2 values for some faults and figure 3 shows the Q values for the same process situations including their respective control limits CL (T_{lim}^2=9.2 and Q_α=66.7).

Table 4 and figure 4 show the delays of the twenty-fault occurrence considering both statistics. Moreover, figure 5 shows clearly the delays for some faults given by the overstep in the Q statistic. These results corroborate that for most of the faults, Q statistic detects such disturbances at earlier times in comparison to T^2. For that reason, models will be constructed since the observation in which Q statistic is over the control limit.

In addition, some tests showed that the delay times do not change when varying the simulation time or the time at which the fault is generated.

4.2. Selection of the transient data window and retained components

Once the transient stages have been localized and we know where we will take data from, classification models can be constructed. ANN and SVM were used in this work as previously mentioned.

The network structure consisted of 31 input nodes because we wanted retaining the more percentage of the original variance (97%). 2200 observations in the training matrix TR were used as inputs because of cost computational restrictions when employing more than these observations for creating the models or networks. The number of hidden nodes was optimized, being six the one that rendered the best classification performance. A tangent sigmoidal transfer function was used with 20 output nodes in the network with which validation data are classified. Therefore, the network structure is 31-6-20.

Regarding SVM, the kernel function that rendered the highest diagnosis performance was the polynomial function of fourth degree. A study of the data window in the training set was executed with 200, 300, 400 and 500 samples and 31 components. Columns 2 to 5 of Table 4 show these results that clearly inform that the best number of observations for constructing the classification models is **400**. This represents a 87% reduction of the original data set (*ns*=3000). Further work regarding *nt* should obviously include the consideration of the tuning of such a value for each faulty scenario *i*.

In addition, the number of components was also studied for the established four hundred observations (from the score matrices $\mathbf{R}_i$) per class by retaining the 97% (31 components), 90% (26 components) and 80% (19 components) of variance in the original attributes or variables. These results are presented in the last columns of Table 4 reporting that the best number of components is 31, representing a 97% of retained variance.

Figure 6 shows the F1 diagnosis performance, broken down in Precision and Recall, for this model using 400 observations and 31 components or variables, which demonstrated to render the best diagnosis performance by using SVM and taking the transient regimes, localized with the Q statistic, as input data.

4.3. Comparison between transient models and steady state based features

As it was mentioned before, a comparison between classification models is done in this work.

Table 5 shows the diagnosis performance in terms of *F1* index for these listed models and applying ANN as classification algorithm:

- Using observations from the period of time where faults are completely developed
- Using observations from the starting period of the fault since the moment in which is generated.

- Using a set of randomly chosen observations
- Using transient observations starting when the Q statistic exceeds its control limit.
- Using transient observations starting when the T^2 exceeds its control limit.

All these models are constructed with 200 observations in the training matrix because of cost computational restrictions with ANN and for being able to reproduce the same results changing the classification method such as SVM. A previous test with Q statistic model employing 200 observations per class or scenario showed that retaining 31 components resulted in a higher performance than retaining 26. Models are validated on the five hundred total observations for calculating the diagnosis performance as an off-line calculation. ANN structure is 31-6-20 as reported in section 4.2

Table 6 reports the results for the same models applying SVM with fourth polynomial kernel for comparative purposes. From the results in Table 5 and 6, we can conclude that the best diagnosis performance is obtained with those models that take into account the transient stages by using Q or T^2 statistics. Figure 7 represents all these models in a ROC diagram, which is a graphical plot of the sensitivity or true positive rate (recall) against false positive rate.

The model that presented the highest performance is the one constructed with 400 observations and 31 components, as shown in Table 4.

5. Discussion

Results show that the models considering the transient stages of faults (specifically the right moment in which the fault is detected by statistic indices) render better diagnosis performance in comparison to those including observations in the steady state of the fault or random observations.

Q statistic is faster than T^2 detecting AOC observations. In consequence, time delays are shorter with Q statistic, providing a better method to identify transient stages of the fault evolution. On the other hand, diagnosis performance obtained applying models that take into account since the observation where faults are really detected by the *Q* statistic is a little bit lower than the diagnosis performance with models considering T^2 statistic instead of *Q*.

The T^2 statistic monitors systematic variations in the PC subspace, while the *Q* statistic represents variations not explained by the retained PCs. That is, faults in the process that violate the normal correlation of variables are detected in the PC subspace by the T^2 statistic, whereas faults that violate the PCA models are detected in the residual space by the *Q* statistic, which is therefore used to detect new events not taken into account in the model subspace [34].

The previous statement explains the fact that *Q* statistic detects faults more rapidly than T^2 when there are only two PC retained representing a 26% of variance. However, classification models are constructed using 31 components or features retaining the 97% of variance in the original process variables and this is why diagnosis performance is a bit higher with models taking into account T^2 delays.

In conclusion, for this case study, most of the faults are detected in the principal component subspace. As another evidence, a PCA model retaining the 97% of variance is generated an applied again to the raw data sets under NOC and AOC so that the delays in detection can be obtained and compared with the previous ones. Results in Table 7 show that the delays are diminished much more in the case of T^2. Even more, whether models are constructed based on these delays as transient stages, the diagnosis performance can slightly improve. Table 8 shows a comparison between two models based on the difference of delays and transient stages with T^2 and applying ANN. The table shows that retaining more principal components leads to better models in terms of diagnosis performance and that for this case study where most of faults are in the PC subspace, models taking into account the transient stages defined by the T^2 statistic are the best ones.

It is worth mentioning that there are no significative deviations in performance when SVM or ANN are applied as classification algorithms and their models are validated on the same test sets. The main difference consists on higher diagnosis performances for SVM and no computational restrictions by processing data in comparison to ANN. This reveals the potential of taking into account transient stages when constructing classification models and validates the proposed approach regardless the learning method to apply.

6. Conclusions

An on-line fault diagnosis method based on data-driven models is proposed. The milestone is the detection of transient stages during the evolution of process abnormalities as well as the use of such dynamical information for constructing enhanced classification models. PCA is not only used for obtaining delays in the fault detection and therefore the faults transient stages, but also as a feature extraction technique for dimensionality reduction, gathering the projections (score matrices) in the training set, which is the input of the classification or fault diagnosis techniques. Delays are the same no matter the fault is produced at different times in a process simulation.

ANN and SVM are both applied in this work and a parameter tuning step is done for determining the network structure and the kernel function that give the highest performance.

The method is tested with the Tennessee Eastman benchmark considering all the 20 faults reported in the case study and a comparison among different data-models is covered. These models consist of that using a) data from periods at which the fault is fully developed, b) data from the period at which the faults are generated but not necessarily detected and c) random observations. Hence, the advantage of a method considering and characterizing the faulty transient stages and providing sensible data for construction is demonstrated in terms of the enhanced diagnosis performance obtained. This has been also corroborated regardless the learning algorithm (ANN, SVM) employed.

The potential of the entire approach, however, may be considered no fully exploited since some tuning opportunities exist that have not been considered, as it has been discussed.

Acknowledgements

Financial support from Generalitat de Catalunya through the FI fellowship program is fully appreciated. Financial support received through the research projects ToleranT (DPI2006-05673) and EHMAN (DPI2009-09386) funded by the European Union (European Regional Development Fund 2007-13) and the Spanish Ministry of Science and Innovation is also fully appreciated. PCA Matlab code was developed by modelEAU team of the Université Laval, Québec to whom we also acknowledge.

References

1. Ruiz D., Nougués J.M., Puigjaner L. (2001). Fault diagnosis for complex chemicals plants. Computers and Chemical Engineering 25, 151-160.

2. Kulkarni A., Jayaraman V.K., Kulkarni B.D. (2005). Knowledge incorporated support vector machines to detect faults in Tennessee Eastman Process. Computers and Chemical Engineering, 29, 2128-2133.

3. Bin He X., Pu Yang Y., Hong Yang Y. (2008). Fault diagnosis based on variable-weighted kernel Fisher discriminant analysis. Chemometrics and Intelligent Laboratory Systems, 93, 27-33.

4. Bin He X., Wang W., Pu Yang Y., Hong Yang Y. (2009). Variable-weighted Fisher discriminant analysis for process fault diagnosis. Journal of Process Control.

5. Yélamos I., Escudero G., Graells M., Puigjaner L. (2009). Performance assessment of a novel fault diagnosis system based on support vector machines. Computers and chemical engineering 33, 244-255.

6. Rokach L. (2007). Genetic algorithm-based feature set partitioning for classification problems.

7. Su C-T., Yang C-H. (2008). Feature selection for the SVM: An application to hypertension diagnosis. Expert Systems with Applications, 34, 754-763.

8. Chiang L.H., Russell E.L., Braatz R.D. (2000). Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares and principal component analysis. Chemometrics and intellingent laboratory systems, 50, 243-252.

9. Brydon D.A., Cilliers J.J., Willis M.J. (1997). Classifying pilot-plant distillation column faults using Neural Networks. Control Engineering Practice, 5, 10, 1373-1384.

10. Yoon S., MacGregor J.F. (2001). Fault diagnosis with multivariate statistical models part I: using steady state fault signatures. Journal of process control, 11, 387-400.

11. Vora N., Tambe S.S., Kulkarni B.D. (1997). Counterpropagation neural Networks for fault detection and diagnosis. Computers and Chemical Engineering, 21, 2, 177-185.

12. Yong M., Zheng X., Zheng Y., Youxian S., Zheng W. (2007). Fault diagnosis based on Fuzzy Support Vector Machine with parameter tuning and Feature Selection. Chinese Journal of Chemical Engineering, 15, 2, 233-239.

13. Chiang L.H., Kotanchek M.E., Kordon A.K. (2004). Fault diagnosis based on Fisher discriminant analysis and support vector machines, 28, 1389-1401.

14. Verron S., Tiplica T., Kobi A. (2007). Procedure base don mutual information and bayesian Networks for the fault diagnosis of industrial Systems. Proceedings of the 2007 American Control Conference, NY, USA, 420-425.

15. Lee J.M., Joe Qin S., Lee I-B. (1997). Fault detection and diagnosis based on modified independent component analysis. AIChE Wiley Interscience.

16. Li J., Cui P. (2009). Improved kernel fisher discriminant analysis for fault diagnosis. Expert Systems with Applications, 36, 1423-1432.

17. Zhang Y. (2009). Enhanced statistical analysis of nonlinear processes using KPCA, KICA and SVM. Chemical Engineering Science, 64, 801-811.

18. Cui P., Li J., Wang G. (2008). Improved kernel principal component analysis for fault detection. Expert Systems with Applications, 34, 1210-1219.

19. Verron S., Tiplica T., Kobi A. (2007). Fault detection and identification with a new feature selection based on mutual information. Journal of Process Control.

20. Srinivasan R., Qian M.S. (2006). Online fault diagnosis and state identification during process transitions using dynamic locus analysis. Chemical Engineering Science, 61, 6109-6132.

21. Sundarraman A., Srinivasan R. (2003). Monitoring transitions in chemical plants using enhanced trend analysis. Computers and chemical engineering, 27, 1455-1472.

22. Srinivasan R., Viswanathan P., Vedam H., Nochur A. (2005). A framework for managing transitions in chemical plants. Computers and chemical engineering, 29, 305-322.

23. Russell E., Chiang L.H., Braatz R. (2000). Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. Chemometrics and Intelligent Laboratory Systems, 51, 81-93.

24. Detroja K.P., Gudi R.D., Patwardhan S.C. (2007). Plant-wide detection and diagnosis using correspondence analysis. Control Engineering Practice, 15, 12, 1468-1483.

25. Shao J.D., Rong G., Lee J.M (2009). Generalized orthogonal locality preserving projections for nonlinear fault detection and diagnosis. Chemometrics and Intellingent Laboratory Systems, 96, 75-83.

26. Seng Y., Srinivasan R. (2009). An adjoined multi-model approach for monitoring batch and transient operations. Computers and chemical engineering, 33, 887-902.

27. Monroy I., Benitez R., Escudero G., Graells M. (2010). A semi-supervised approach to fault diagnosis for chemical processes. Computers and Chemical Engineering, 34, 631-642.

28. Kent A., Berry M.M., Luehrs F.U., Perry J.W. (1955). Machine literature searching: VIII. Operational criteria for designing information retrieval systems. American Documentation, 6, 2, 93-101.

29. Boser B., Guyon I., Vapnik V. (1992). A training algorithm for optimal margin classifiers. In: Proceedings of the Workshop on Computational Learning Theory, COLT, Pittsurgh, PA.

30. Montgomery D.C. (2005). Introduction to Statistical Quality Control, 5th. Wiley International Edition, USA.

31. Nomikos P., MacGregor J. (1995). Multivariate SPC charts for monitoring batch processes. Technometrics, 37, 1, 41-59.

32. Jackson J.E., Mudholkar G.S. (1979). Control Procedures for Residuals Associated with Principal Component Analysis. Technometrics, 21, 341-349.

33. Lu B., Upadhyaya B.R. (2005). Monitoring and fault diagnosis of the steam generator system of a nuclear power plant using data-driven modeling and residual space analysis. Annals of Nuclear Energy, 32, 897-912.

34.Yoo C., Villez K., Lee I., Rosén C. (2007). Multi-model statistical process monitoring and diagnosis of a sequencing batch reactor. Biotechnology and Bioengineering, 96, 4, 687-701.

35. Downs JJ, Vogel EF. (1992). A plant wide industrial process control problem, Computers and Chemical Engineering. 17, 3, 245-255.

36. Joachims T. (1999). Making large-scale SVM learning practical. In B. Schölkopf , c. Burges & A. Smola (Eds.), Advances in Kernel methods-support vector learning. MIT Press.

Table 1. Process variables of the TE benchmark

ATTRIBUTE	VARIABLE NAME	VARIABLE TYPE
1	COMPONENT “A” FEED FLOW (STREAM 1)	MEASURED
2	COMPONENT “D” FEED FLOW (STREAM 2)	MEASURED
3	COMPONENT “E” FEED FLOW (STREAM 3)	MEASURED
4	COMPONENTS “A” AND “C” FEED FLOW (STREAM 4)	MEASURED
5	RECYCLE FLOW FROM SEPARATOR (STREAM 8)	MEASURED
6	REACTOR FEED RATE (STREAM 6)	MEASURED
7	REACTOR PRESSURE	MEASURED
8	REACTOR LEVEL	MEASURED
9	REACTOR TEMPERATURE	MEASURED
10	PURGE RATE (STREAM 9)	MEASURED
11	PRODUCT SEPARATOR TEMPERATURE	MEASURED
12	PRODUCT SEPARATOR LEVEL	MEASURED
13	PRODUCT SEPARATOR PRESSURE	MEASURED
14	PRODUCT SEPARATOR UNDERFLOW (STREAM 10)	MEASURED
15	STRIPPER LEVEL	MEASURED
16	STRIPPER PRESSURE	MEASURED
17	STRIPPER UNDERFLOW (STREAM 11)	MEASURED
18	STRIPPER TEMPERATURE	MEASURED
19	STRIPPER STEAM FLOW	MEASURED
20	COMPRESSOR WORK	MEASURED
21	REACTOR COOLING WATER OUTLET TEMPERATURE	MEASURED
22	CONDENSER COOLING WATER OUTLET TEMPERATURE	MEASURED
23	COMPOSITION OF “A” IN REACTOR FEED	MEASURED
24	COMPOSITION OF “B” IN REACTOR FEED	MEASURED
25	COMPOSITION OF “C” IN REACTOR FEED	MEASURED
26	COMPOSITION OF “D” IN REACTOR FEED	MEASURED
27	COMPOSITION OF “E” IN REACTOR FEED	MEASURED
28	COMPOSITION OF “F” IN REACTOR FEED	MEASURED
29	COMPOSITION OF “A” IN PURGE GAS FLOW	MEASURED
30	COMPOSITION OF “B” IN PURGE GAS FLOW	MEASURED
31	COMPOSITION OF “C” IN PURGE GAS FLOW	MEASURED
32	COMPOSITION OF “D” IN PURGE GAS FLOW	MEASURED
33	COMPOSITION OF “E” IN PURGE GAS FLOW	MEASURED
34	COMPOSITION OF “F” IN PURGE GAS FLOW	MEASURED
35	COMPOSITION OF “G” IN PURGE GAS FLOW	MEASURED
36	COMPOSITION OF “H” IN PURGE GAS FLOW	MEASURED
37	COMPOSITION OF “D” IN PRODUCT FLOW	MEASURED
38	COMPOSITION OF “E” IN PRODUCT FLOW	MEASURED
39	COMPOSITION OF “F” IN PRODUCT FLOW	MEASURED
40	COMPOSITION OF “G” IN PRODUCT FLOW	MEASURED
41	COMPOSITION OF “H” IN PRODUCT FLOW	MEASURED

42	D FEED FLOW	MANIPULATED
43	E FEED FLOW	MANIPULATED
44	A FEED FLOW	MANIPULATED
45	“A” AND “C” FEED FLOW	MANIPULATED
46	COMPRESSOR RECYCLE VALVE	MANIPULATED
47	PURGE VALVE	MANIPULATED
48	SEPARATOR POT LIQUID FLOW	MANIPULATED
49	STRIPPER LIQUID PRODUCT FLOW	MANIPULATED
50	STRIPPER STEAM VALVE	MANIPULATED
51	REACTOR COOLING WATER FLOW	MANIPULATED
52	CONDENSER COOLING WATER FLOW	MANIPULATED

Table 2. Process disturbances in the TE

FAULT	PROCESS VARIABLE	TYPE
1	A/C FEED RATIO, B COMPOSITION CONSTANT (STREAM 4)	STEP
2	B COMPOSITION, A/C RATIO CONSTANT (STREAM 4)	STEP
3	D FEED TEMPERATURE (STREAM 2)	STEP
4	REACTOR COOLING WATER INLET TEMPERATURE	STEP
5	CONDENSER COOLING WATER INLET TEMPERATURE	STEP
6	A FEED LOSS (STREAM 1)	STEP
7	C HEADER PRES. LOSS – REDUCED AVAILABILITY (STREAM 4)	STEP
8	A, B, C, FEED COMPOSITION (STREAM 4)	RANDOM VARIATION
9	D FEED TEMPERATURE (STREAM 2)	RANDOM VARIATION
10	C FEED TEMPERATURE (STREAM 4)	RANDOM VARIATION
11	REACTOR COOLING WATER INLET TEMPERATURE	RANDOM VARIATION
12	CONDENSER COOLING WATER INLET TEMPERATURE	RANDOM VARIATION
13	REACTION KINETICS	SLOW DRIFT
14	REACTOR COOLING WATER VALVE	STICKING
15	CONDENSER COOLING WATER VALVE	STICKING
16	UNKNOWN	UNKNOWN
17	UNKNOWN	UNKNOWN
18	UNKNOWN	UNKNOWN
19	UNKNOWN	UNKNOWN
20	UNKNOWN	UNKNOWN

Table 3. Fault detection delays in minutes

FAULT	Delays (min) According to T^2	Delays (min) According to Q
1	15	5
2	33	16
3	283	64
4	2	1
5	292	8
6	20	1
7	1	1
8	30	20
9	271	64
10	563	33
11	9	7
12	316	61
13	167	64
14	283	5
15	292	309
16	30	13
17	83	64
18	300	64
19	258	64
20	171	64

Table 4: Optimization of the data window and PC in the classification models using SVM and the model that takes into account the delays given by the Q statistic

	MODEL BASED ON					
FAULT	31 COMPONENTS				400 SAMPLES	
	200 samples	300 samples	**400 samples**	500 samples	26 comp	19 comp
1	89.2	89.2	**90.1**	89.8	88.9	51.1
2	94.5	95.7	**95.8**	96	95.8	95.7
3	34.2	47.2	**53.1**	53.9	51.9	51.1
4	66.6	76.8	**90.8**	91.9	83.5	83.5
5	2.2	2.7	**15.8**	18.9	16.9	20.1
6	99.9	99.9	**99.4**	99.9	99.1	78.3
7	99.9	99.9	**97.6**	99.9	99.9	99.9
8	42.8	57.5	**60**	64.5	56.8	53.9
9	32.3	44.8	**52.8**	0	49.6	49.2
10	62.4	72.4	**84**	76.4	83	80.2
11	42.9	53	**73.2**	81.7	72.8	65.7
12	36.1	59.2	**71.1**	17.7	0	16.4
13	33.7	12.8	**61.9**	12.7	68.3	13.9
14	39.8	41.3	**48.9**	58.9	36.5	21.7
15	16.7	19.9	**23.9**	24.4	23.6	24.4
16	71.6	74.4	**80**	90	81.9	82.8
17	68.5	83.1	**86.2**	86.1	84.9	85.4
18	0.5	50.8	**14.3**	11.4	21.5	14.5
19	38.1	59.8	**69.9**	70.8	67.3	51.5
20	39.2	48.7	**17.6**	47.1	13.7	32.6
0	14.7	5.3	**6.9**	0	0	0.4
Mean	50.5	59.4	**64.3**	59.6	59.8	53.6

Table 5: Comparison among models constructed by ANN in terms of F1 index

FAULT	Model based on				
	Developed faults	Starting period of the fault	Random	T^2 statisic	Q statistic
1	0	57.4	86	55.6	59.8
2	79.1	63	86	74	83.4
3	0	0	0	0	0
4	0	53.3	0	79.7	73
5	12.1	0	0	0	0
6	10.2	69.7	99.6	61.2	99.9
7	0	67.6	54.9	92.8	67.4
8	8	0	47.4	34.5	38
9	0	0	0	0	0
10	0	28.7	0.2	0	63.6
11	14.7	35.1	35.8	45.3	34.8
12	0	0	0	25.4	23.2
13	0	20.6	31.1	25.6	21.4
14	0	3.1	0	41	0
15	0	0	0	0	0
16	11.2	0	0	49	72.1
17	9	15.2	69.2	86.3	51.8
18	0	6.1	62.5	64.4	0
19	0	0	0	18.9	0
20	6.4	36.5	43.8	7.9	45.4
0	0	14.4	0	17.8	17.1
Mean	7.5	22.8	30.8	38.1	36.7

Table 6: Comparison among models constructed by SVM in terms of F1 index

FAULT	Model based on				
	Developed faults	Starting period of the fault	Random	T^2 statisic	Q statistic
1	0	82.4	0	83.2	89.2
2	0	69.6	10	93	94.5
3	15.2	26.4	22.6	31.9	34.2
4	66.5	67.1	57.6	72	66.6
5	17	4.2	23.5	14.2	2.2
6	0	99.6	0	99.1	99.9
7	0	77.7	0	88.2	99.9
8	9.1	30.6	16.4	46	42.8
9	4.3	20.6	7.7	32.6	32.3
10	13.9	41.2	29.9	19.9	62.4
11	30.3	12.2	47.9	51.2	42.9
12	28.4	16	17.7	21.4	36.1
13	9.1	5.3	9.6	56.3	33.7
14	36.2	31.8	32.3	47.9	39.8
15	15	17.3	17.4	25.2	16.7
16	38.5	74.2	10.7	83.1	71.6
17	9.1	16.7	41.6	84.1	68.5
18	9.1	3.8	33.1	53.3	0.5
19	0	30.4	29	50.7	38.2
20	10.4	10.5	12.8	45.7	39.2
0	0	0	0	18.8	14.7
Mean	15.6	36.9	21	55	50.6

Table 7: Detection delays with PCA models retaining 26% (broken-stick rule) and 97% of variance

	DELAYS IN MINUTES			
FAULTS	2 PC's (26% var)	35 PC's (97% var)	2 PC's (26% var)	35 PC's (97% var)
	T^2	T^2	Q	Q
1	15	5	5	4
2	33	16	16	15
3	283	440	64	228
4	2	1	1	1
5	292	8	8	2
6	20	1	1	1
7	1	1	1	1
8	30	20	20	19
9	271	440	64	115
10	563	61	33	29
11	9	7	7	7
12	316	61	61	23
13	167	90	64	162
14	283	5	5	2
15	292	90	309	458
16	30	13	13	17
17	83	80	64	70
18	300	90	64	286
19	258	69	64	27
20	171	90	64	132

Table 8: Diagnosis performance for models based on the T^2 statisic

FAULT	F1 index Retained variance with PCA model 26% var	 97% var
1	55.6	83.2
2	74.0	67.3
3	0	0
4	79.7	75.9
5	0	8
6	61.2	48.9
7	92.8	87.4
8	34.5	20.0
9	0	3.7
10	0	68.0
11	45.3	21.0
12	25.4	30.6
13	25.6	30.9
14	41.0	39.2
15	0	0
16	49.0	77.8
17	86.3	80.4
18	64.4	0
19	18.9	0
20	7.9	45.9
0	17.8	0
Mean	38.1	39.4

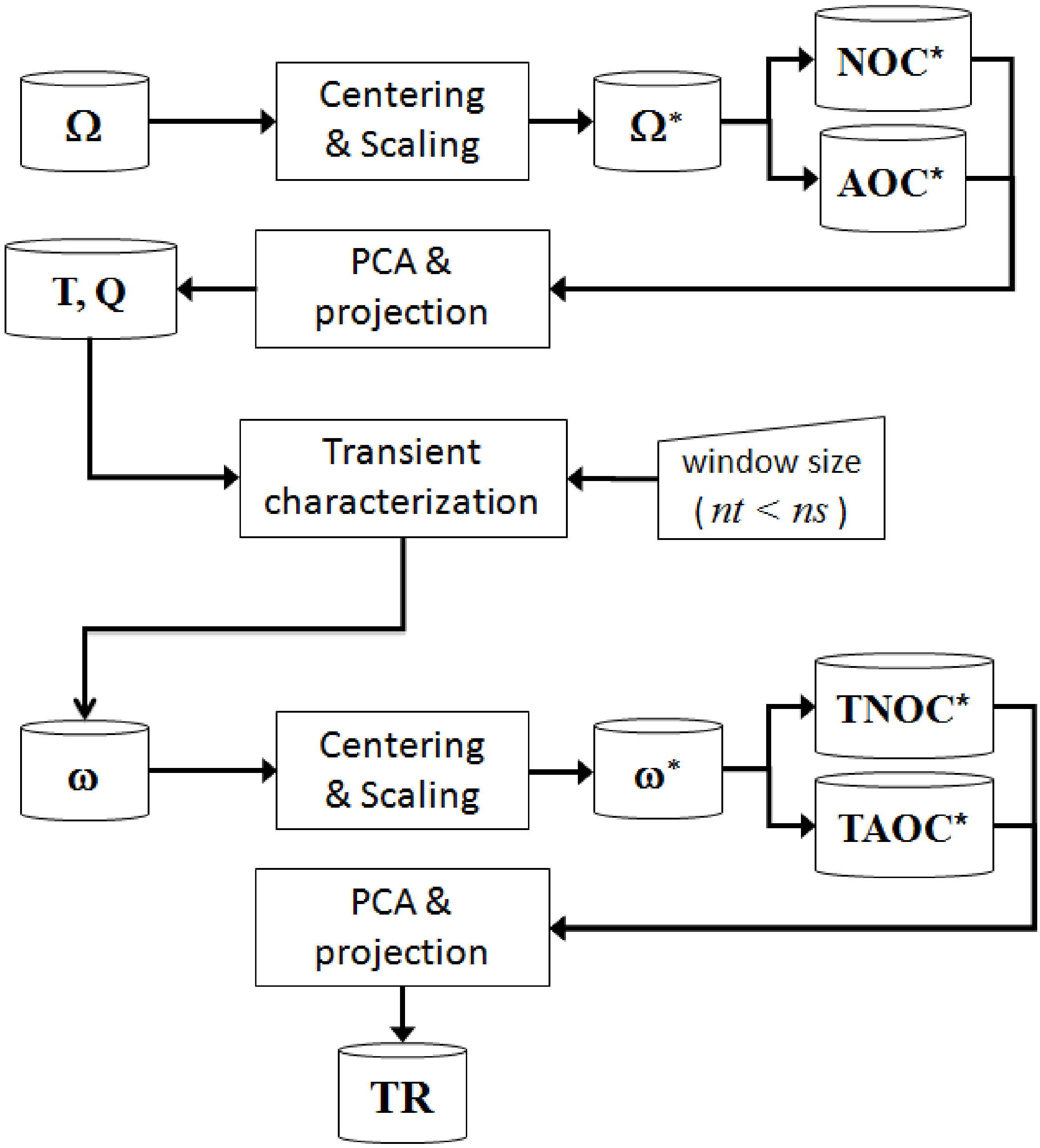

Figure 1. On-line fault diagnosis methodology based on transient stages models.

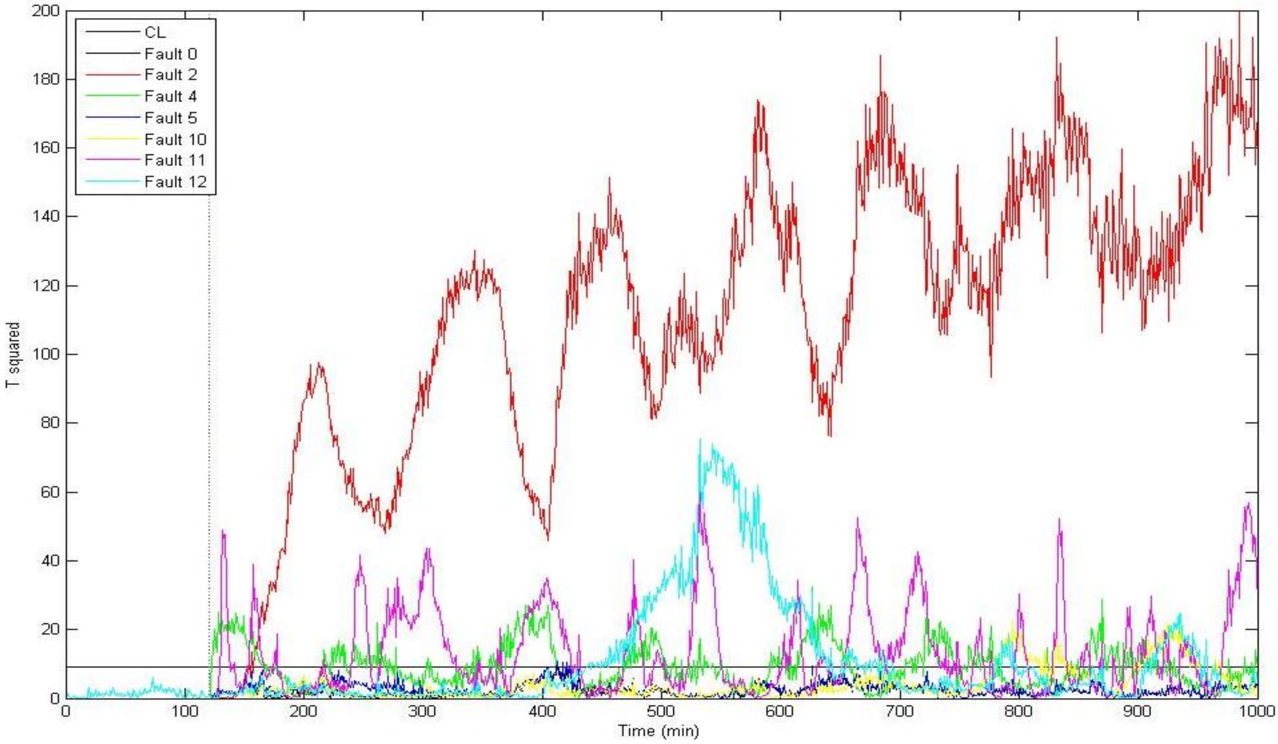

Figure 2. Hotelling T^2 for some faults of the TE process. The vertical line indicates the point where the fault is produced.

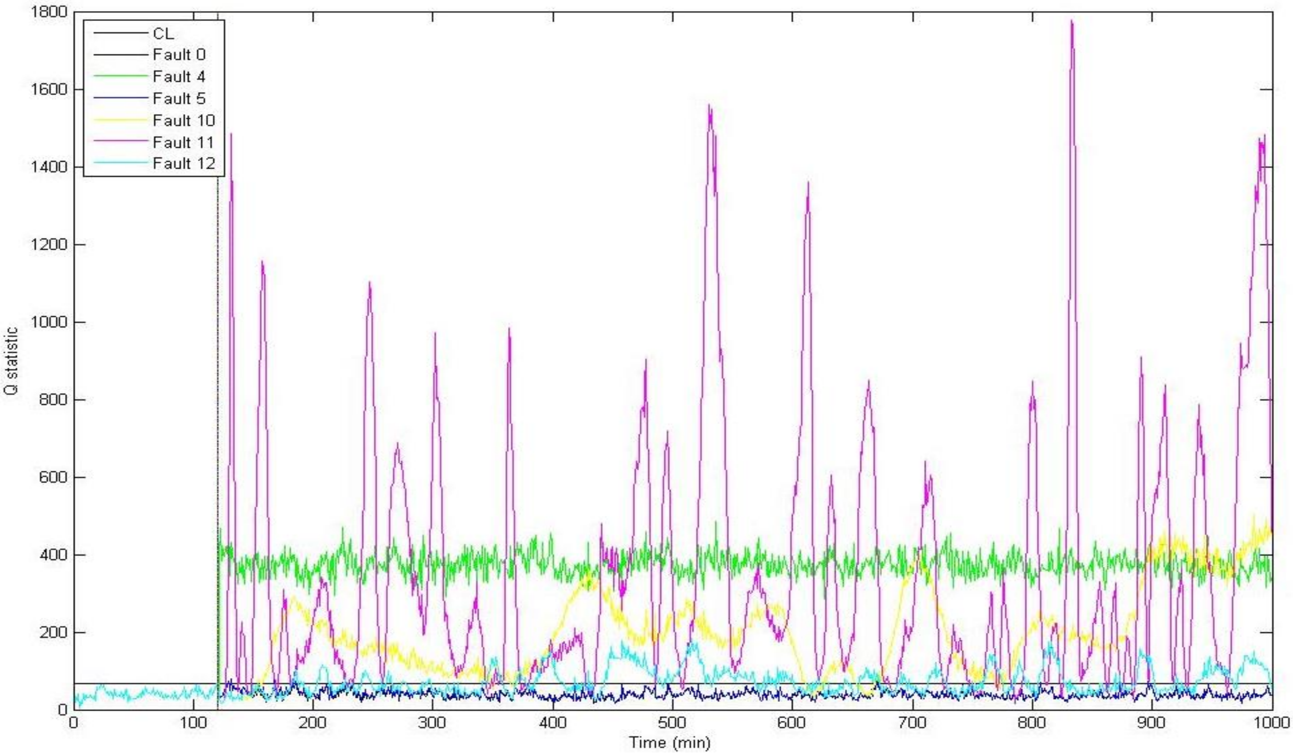

Figure 3. Q statistic for some faults of the TE process. The vertical line indicates the point where the fault is produced.

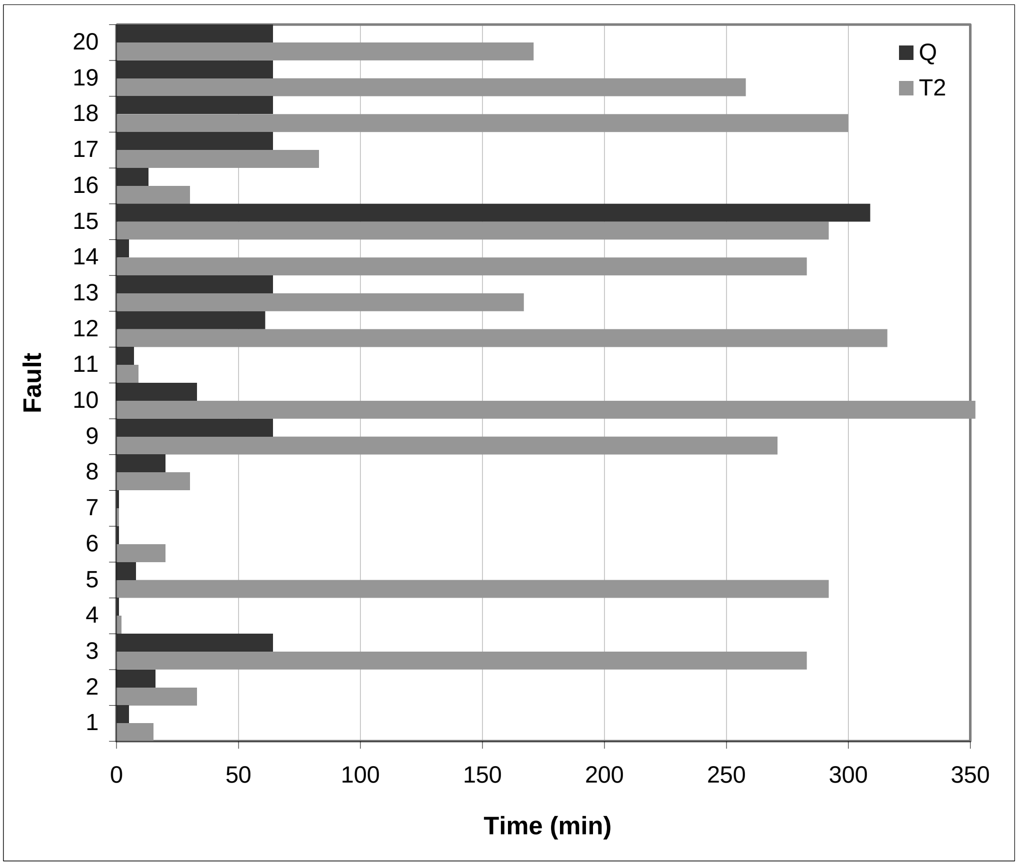

Figure 4. Faults delays in minutes using both T^2 and Q statistic indices.

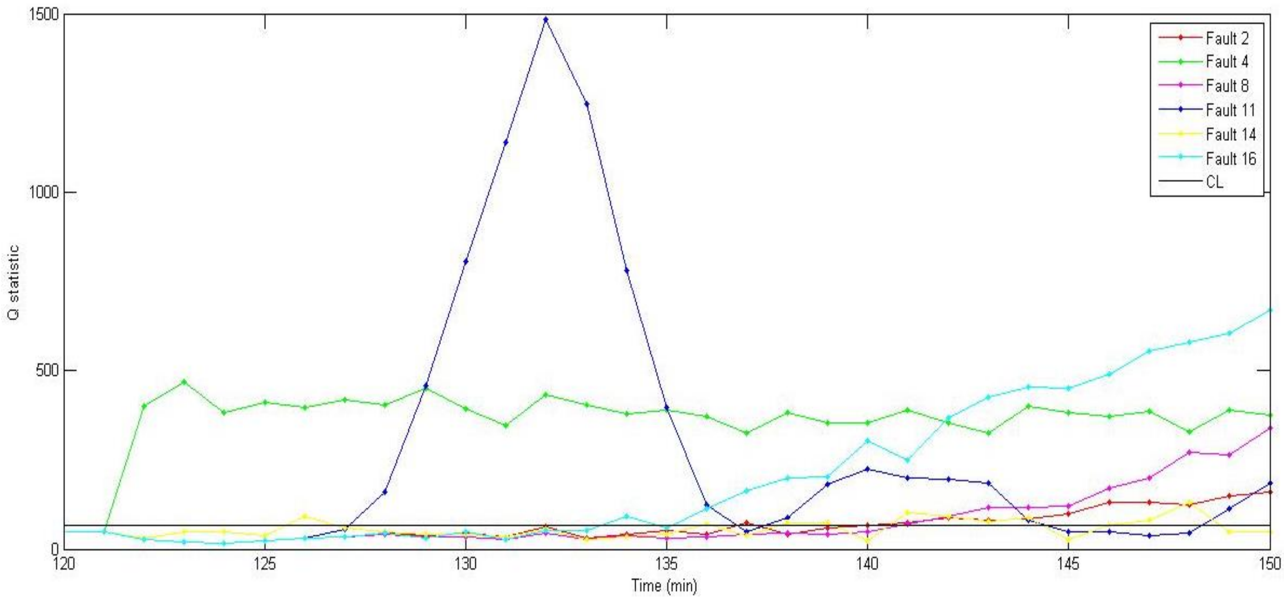

Figure 5. Q statistic for some faults showing their delays in minutes.

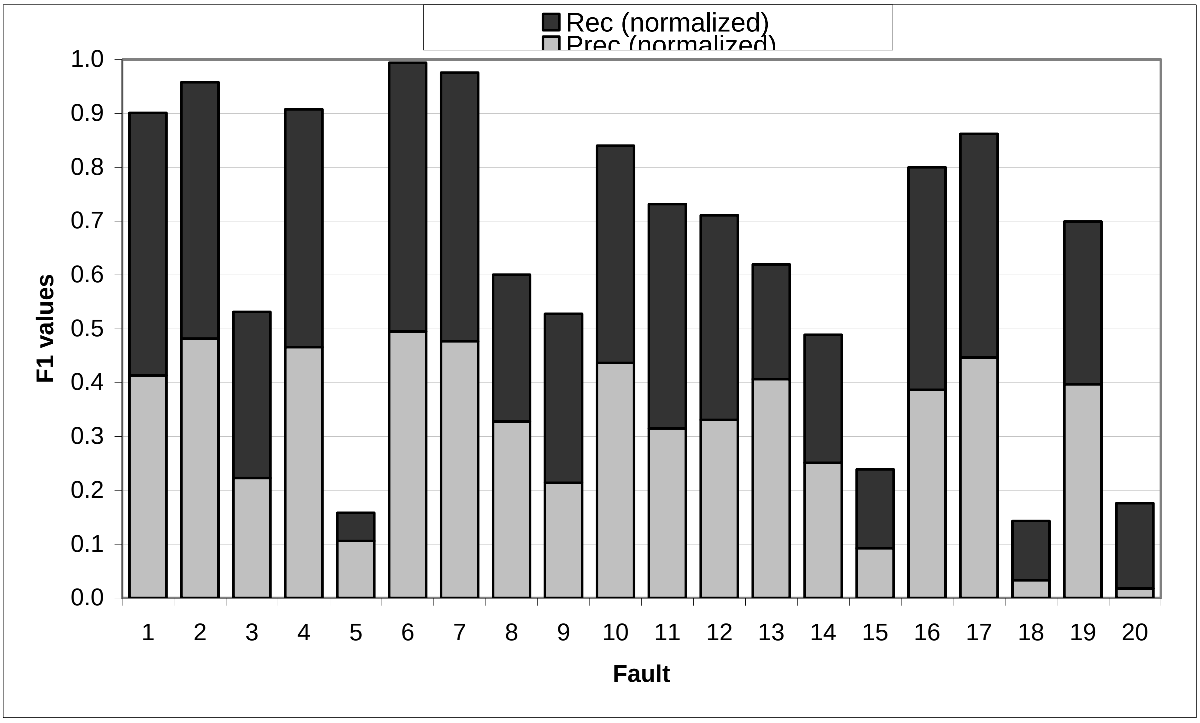

Figure 6. F1 index for the model created with 400 data in the transient regime and 31 variables by applying SVM, indicating Precision an Recall proportions in the F1.

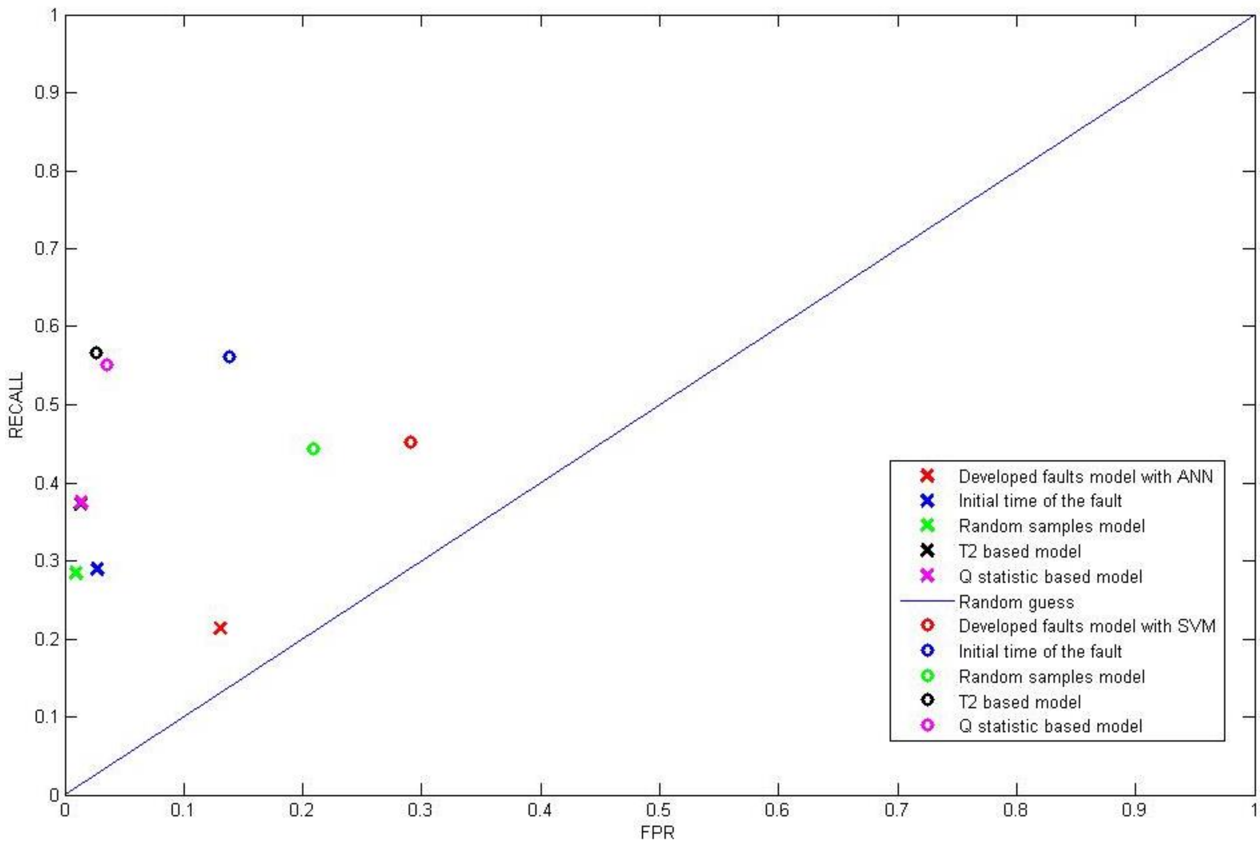

Figure 7. ROC diagram for some classification models using ANN and SVM.