ADVERTIMENT. La consulta d'aquesta tesi queda condicionada a l'acceptació de les següents condicions d'ús: La difusió d'aquesta tesi per mitjà del servei TDX (www.tesisenxarxa.net) ha estat autoritzada pels titulars dels drets de propietat intel·lectual únicament per a usos privats emmarcats en activitats d'investigació i docència. No s'autoritza la seva reproducció amb finalitats de lucre ni la seva difusió i posada a disposició des d'un lloc aliè al servei TDX. No s'autoritza la presentació del seu contingut en una finestra o marc aliè a TDX (framing). Aquesta reserva de drets afecta tant al resum de presentació de la tesi com als seus continguts. En la utilització o cita de parts de la tesi és obligat indicar el nom de la persona autora.

ADVERTENCIA. La consulta de esta tesis queda condicionada a la aceptación de las siguientes condiciones de uso: La difusión de esta tesis por medio del servicio TDR (www.tesisenred.net) ha sido autorizada por los titulares de los derechos de propiedad intelectual únicamente para usos privados enmarcados en actividades de investigación y docencia. No se autoriza su reproducción con finalidades de lucro ni su difusión y puesta a disposición desde un sitio ajeno al servicio TDR. No se autoriza la presentación de su contenido en una ventana o marco ajeno a TDR (framing). Esta reserva de derechos afecta tanto al resumen de presentación de la tesis como a sus contenidos. En la utilización o cita de partes de la tesis es obligado indicar el nombre de la persona autora.

WARNING. On having consulted this thesis you're accepting the following use conditions: Spreading this thesis by the TDX (www.tesisenxarxa.net) service has been authorized by the titular of the intellectual property rights only for private uses placed in investigation and teaching activities. Reproduction with lucrative aims is not authorized neither its spreading and availability from a site foreign to the TDX service. Introducing its content in a window or frame foreign to the TDX service is not authorized (framing). This rights affect to the presentation summary of the thesis as well as to its contents. In the using or citation of parts of the thesis it's obliged to indicate the name of the author

Departament d'Enginyeria Química

Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona

Universitat Politècnica de Catalunya

An investigation on automatic systems for fault diagnosis in chemical processes

Thesis

Presented for the degree of Doctor of Philosophy.
Directed by Dr. Moisès Graells Sobré and Dr. Gerard Escudero Bakx.
Barcelona, December 2011.

Isaac Monroy Chora

Biochemical Engineer

Copyright © by Isaac Monroy Chora. December 2011.
The copyright of this thesis rests with the author.
No quotations of it should be published without prior consent and
Information derived from it should be acknowledged.

ACKNOWLEDGEMENTS

First, I would like to thank my thesis directors Dr. Moisès Graells and Dr. Gerard Escudero, who proposed me to develop this thesis work. I want to acknowledge the comments and details regarding redaction to Moisès Graells. I am also deeply grateful to Gerard Escudero for all his support, guide and help during my PhD.

I want to express my gratitude towards Dr. Raul Benitez for his collaboration to my research. Without no doubt, his contributions motivated me and helped me. I am also grateful to professor Venkat Venkatasubramanian and Dr. Kris Villez for receiving me at Purdue University. Specially, I thank Kris Villez for his teaching and ideas during my internship at Purdue.

Moreover, I thank to Dr. Javier Cruz Gómez for accepting me during a short period of time at UNAM. There are many people in there, to whom I thank for their kindness and guidance. Certainly, I learned many things during my internship at UNAM.

Personally, I thank my mother for her unconditional support in all decisions I have made including this and for her love and comprehension; to you my good friend that has been there for me for good or bad, close or far; to my Mexican and Catalan family for all the sentimental support; and to my cousin and friends in Mexico and Barcelona for the good and bad experiences, for all your love and all I have learned with you.

Finally, I want to highlight the economical support during four years that has allow me finishing my PhD and developing this thesis. I acknowledge to the FI program from the Generalitat de Catalunya for this economical support and the opportunity of being hired.

AGRADECIMIENTOS

Primero, me gustaría agradecer a mis directores de tesis Dr. Moisès Graells y Dr. Gerard Escudero, quienes me propusieron desarrollar este tema de tesis para mi doctorado. Agradezco a Moisès Graells sus comentarios y detalles, especialmente en el aspecto de redacción. Agradezco también profundamente a Gerard Escudero for todo su apoyo, guía y ayuda durante todo mi doctorado.

Me gustaría expresarle mi gratitud al Dr. Raul Benitez por su colaboración a mi investigación. Sin duda, su aporte fue de gran ayuda y motivación. Agradezco también al profesor Venkat Venkatasubramanian y al Dr. Kris Villez por recibirme en la Universidad de Purdue. Especialmente, agradezco a Kris por sus enseñanzas e ideas durante mi estancia en Purdue.

Además, también le agradezco al Dr. Javier Cruz Gómez por aceptarme durante una corta estancia en la UNAM. Son varias las personas de dicha universidad a quienes les agradezco su amabilidad, apoyo y enseñanzas durante esa estancia y que prefiero no personalizar para evitar olvidarme de alguno.

Personalmente, le doy gracias a mi madre por su incondicional, constante y absoluto apoyo en todas las decisiones que he tomado, incluyendo ésta, y por todo su amor y comprensión; a ti mi buen amigo que has estado siempre en las buenas y en las malas conmigo, cerca o lejos; a mi familia mexicana y catalana por todo el apoyo sentimental; y finalmente, a mi prima y amigos de México y Barcelona por todas las buenas y malas experiencias, por todo su cariño y por todo lo aprendido con ustedes.

Finalmente, quiero enfatizar el apoyo económico recibido durante cuatro años para la realización de mi doctorado y redacción de mi tesis doctoral, así como la oportunidad de haber sido contratado. Dicho apoyo me fue otorgado por la Generalitat de Catalunya.

PART I
INTRODUCTION

Chapter 1
MOTIVATION AND SCOPE

Nowadays flexibility and adaptability are important traits of chemical processes either continuous or batch. Such characteristics come from the market demands and needs. In parallel, there is a tendency to increase the process, product and environmental quality together with an economical resource minimization. Beyond this, plant safety is the paramount concern of chemical industries and engineers (AIChE Code of Ethics).

However, the flexibility in chemical processes can increase the risk factor in productivity, safety and environmental quality. In this framework, fault detection and diagnosis, operation supervision and the preventive maintenance become the most important factors to take care in plants.

Fault detection has been widely addressed with multivariate statistical process monitoring (MSPM) techniques in both academics and industry. The range entails since the typical control charts until the PCA-based techniques and the corresponding statistical indices calculation. Statistical techniques are still used for monitoring and fault detection purposes.

On the other hand, fault diagnosis has been faced with several models, mainly grouped in three classes. The first class is the quantitative model based on the analytical equations and models of the process, however these models are normally too complex and non-reliable in real processes. The second class includes the qualitative models, which establish relationships between the input and outputs. As the first type of models, they can also be difficult to construct whether the process is big and complex. The third group includes the data-based models, which are related to the black-box models in which there is only information about the inputs and outputs. These models have demonstrated to deal with the fault diagnosis problem even if the process is big and complex. The only requirement is the existence of plant historical records in normal and abnormal process states.

Fault diagnosis started to be dealt with machine learning algorithms as classifiers or diagnosis systems. The problem was first formulated as a multi-class and mono-label classification. However, there can be multiple and simultaneous faults in a plant and the mono-label classification implied to treat this realistic situation as new classes or new single faults. This actually increased the number of classes and data-models and reduced the precision of the general system. Due to this unrealistic approach, the fault diagnosis

has been currently defined as a multi-class and multi-label classification problem in terms of machine learning.

Beyond this, it is important to develop a reliable data-based diagnosis system with the plant historical records including either single or multiple faults but with the guarantee that such system will learn to recognize and identify faulty situations during the plant operation. In general, although the fault diagnosis has been addressed with both analytic and data-based models, not much attention has been paid to the actual implementation of the reported approaches in the research field and still less efforts have been done for the development of a fault diagnosis system that allows the on-line diagnosis of chemical plants.

There have been some advances in the fault diagnosis area, however there are some important points to attack in order to have a robust automatic system for the fault diagnosis in chemical plants. The main points are next exposed. First, it is necessary the development of a general approach for the construction of data-based diagnosis models for continuous and batch processes no matter what techniques are used for developing the involved steps. Such models should improve the diagnosis performance by means of the data preprocessing and the parameter tuning of the algorithms applied.

Then, the application of the whole diagnosis system on the on-line process should be accomplished. This system should be quite robust in order to detect and diagnose novel faults included neither in the constructed models nor in the plant records. It is also desired that the on-line systems can further modify the diagnosis models based on changes in plant, new faults or including new information about the process.

Furthermore, the diagnosis system and the involved tools could be incorporated in a bigger robust system that includes preventive maintenance programs. This would generate a powerful plant safety and fault diagnosis system that ease the operators and process engineers' decision-making by taking not only anticipated corrective actions, but also preventive measures.

This thesis proposes a general approach to fault diagnosis based on data-driven models for both continuous and batch processes. Some different approaches are investigated on the research line but leading to the construction of such robust and automated system and its on-line implementation for the detection and diagnosis of faults (learned or novel). A preventive maintenance program is also analyzed to be further included in a whole plant safety system.

Chapter 2
STATE OF THE ART

Chemical industries have always been concerned about the different procedures and techniques for reducing the risk of accidents and dealing with process abnormalities and faults. Fault detection and diagnosis is an important process engineering field not only because of the impact of the abnormal situations on the safety and economy of the plant but also because of the obligation to keep the yield and quality of the process. Petrochemical industries for example have rated the Abnormal Event Management (AEM) as the first problem that needs to be solved.

Abnormal situations occur when processes deviate from their normal operating regime (Dash and Venkatasubramanian, 2000). Fault detection and diagnosis are gaining importance because they can avoid the progression of abnormal events, allow the selection of the proper corrective action and avoid loses and process productivity reduction. Such nature can be described by some characteristics such as:

- High complexity and size of modern process plants
- High automation level
- Dynamic and non-linear characteristics
- Usually driven by fewer essential variables which may not be measured
- High dimensional multivariate data
- Deviation from Gaussianity of the latent variables
- State variables may have uncontrollable disturbances and random noise.

It is worthy to mention regulatory control as the precedent of the fault detection and diagnosis techniques that exist nowadays. Regulatory control is a task that used to be performed by human operators and is now performed in an automated manner. Although the progress in this field and its benefits to all industrial segments have been quite important, there is a very important task in process plant management that is still remaining as a manual activity. Such task corresponds to responding to abnormal events in a process (Venkatasubramanian et al., 2003a).

There are four procedures associated with AEM and process monitoring: fault detection, fault identification, fault diagnosis and process recovery (Chiang et al., 2004).

- Fault detection determines when an abnormal process behavior occurs whether associated to equipment failures, process unit degradation, parameter drifts or extreme process faults.

- Fault identification consists of identifying the most relevant variables associated to the fault. The purpose is to focus the plant operator's attention on the most pertinent subsystems for the fault diagnosis so that the effect of the fault can be eliminated in a more efficient manner.
- Fault diagnosis lies on determining which fault occurred. In other words, on determining the root cause of the observed out-of-control status. This procedure is essential for the counteraction or elimination of the fault.
- Process recovery, also called intervention, is to remove the effect of the fault.

The main goal of Automatic Fault Detection and Diagnosis is to achieve the highest level of autonomy in real time dynamic systems with sophisticated control strategies. However, this task is difficult to achieve by intelligent and real-time systems due to several factors such as complexity of the process dynamics, lack of adequate models, diverse sources of knowledge, poor measurements (insufficient, incomplete and uncertain) and the great amount of effort and expertise required to develop and maintain such systems.

Due to the broad scope of the fault diagnosis problem and the difficulties in its real time solution, literature reports abundant approaches to face this problem. These approaches range from the analytical methods to statistical techniques and machine learning methods in the Artificial Intelligence area. The different approaches recently published will be presented and categorized in the next sections. In addition, machine learning techniques will be applied as classification or diagnosis methods in this thesis.

2.1 Fault detection methods

Real process operations are subjected to random disturbances and their future state can not be completely determined by the past and present states. When a process is under control, the variable observations have probability distributions corresponding to the normal mode of operation and when the process is out of control these distributions will change. Then, the factors of interest are the mean and the standard deviation, which may deviate from the nominal values in a faulty process. The term "fault" can then be defined as a departure from an acceptable range of a process variable (Himmelblau, 1978).

In general, there are three classes of failures or malfunctions according to Venkatasubramanian et al., 2003a):

- **Exogenous failures**: They arise when there is a disturbance in exogenous variables, which are not directly measured, due to the process environment. Change in the heat transfer coefficient due to fouling in a heat exchanger is an example of this kind of faults.
- **Structural failures**: They are changes in the process itself due to hard failures in the equipment. A stuck valve, failure of a controller, and a broken or leaking pipe are some examples of this kind of failures.
- **Instrument failures**: These gross errors are related to actuators and sensors failures. A failure in one of these instruments could cause the state variables to

deviate beyond acceptable limits unless the failure is detected promptly and corrective actions are accomplished on time.

The automation of the fault detection and diagnosis is the first and main step towards an efficient Abnormal Event Management (AEM). In particular, fault detection, as result of the process monitoring, has been faced with statistical techniques, first univariate and then multivariate. Traditional methods of monitoring are based on different statistical approaches, which include the use of distributions and different control charts. Control chart types include Shewhart, cumulative-sum and exponentially weighted moving-average (Montgomery, 1991).

Control charts, the first monitoring-based statistical technique, were introduced by Shewhart (1931). They are based in the fact that a process will remain in a state of statistical control where the process and/or the product variables will remain close to their desired values. Therefore, abnormal events can be detected by monitoring the performance of the process. However, these traditional methods have a drawback by not taking into account the possibility of change in operation points that can cause a change in performance.

The growing demand for product quality and process reliability led to an extensive use of Statistical Process Control (SPC), which has been used to monitor process signals and detect trends, outliers and other anomalies. However, this univariate approach only detects disturbances related to individual measurement sources (usually quality) and cannot handle correlation. Moreover, it revealed some limitations when using high dimensional multivariate data that is cross-correlated.

In conclusion, traditional Univariate Methods (Linear Regression, Univariate Statistical Control Charts) are often inadequate to evaluate masses of data and to detect abnormalities. It is not only the evolution of each variable by itself that gives useful information about faults, but also the evolution of each variable correlated to other variables. For this reason, Multivariate Statistical Process Monitoring (MSPM) was introduced and Multivariate Statistical Methods (MSM) were designed to model this correlation structure by compressing data and reducing the large number of correlated variables into a small number of fictitious uncorrelated variables so that essential and true information is extracted and easier to analyze (Kassidas et al., 1998).

In particular, Principal Component Analysis (PCA) and Projection to Latent Structures (PLS), previously known as Partial Least Squares, were the first MSPM methods which received a lot of attention and are still applied (Kresta et al., 1991; MacGregor et al., 1994). They have been successfully applied for fault detection and diagnosis in large chemical processes (Kourti and MacGregor, 1995). These methods can also handle two real problems in plant operation: noisy measurements and missing data. When applying these techniques, the evolution of the process can be observed in the reduced space (principal component space in case of PCA).

Overviews of using PCA and PLS in process analysis and control, fault detection and diagnosis were given by MacGregor et al. (1991, 1994) and MacGregor and Kourti (1995). In particular, Partial Least Squares (PLS) was originated from the pioneering work of Wold (1982) and was further developed by Wold et al. (1987). It is useful in reducing the

dimensions of both process variables and product quality variables to be analyzed. PLS models the relationship between two blocks of data while compressing them simultaneously and is used to extract latent variables that not only explain the variation in the process data but also the predictive variation of the quality data. In 2001, Yoon and MacGregor used PLS and PCA as process monitoring and fault detection methods.

Principal Component Analysis (PCA) method was initially proposed by Pearson (1901) and later developed by Hotelling (1947) and consists of extracting information about the major trends in the data using a small number of relevant factors or principal components. An on-line process monitoring with PCA is achieved by projecting new observations onto the plane defined by the PCA loading vectors, obtained using historical data during normal operation. Several works in which PCA is used for process monitoring and fault detection have been developed and reported. Some of them are found in Kano et al. (2001), Chen and Liao (2002), Chiang and Braatz (2003), Tamura and Tsujita (2007).

In addition, pioneering work in the specific area of Fault Detection and Identification (FDI) for batch processes was performed by Nomikos and MacGregor (1994, 1995a). Multi-way Principal Component Analysis (MPCA) has been successfully applied to batch processes as process monitoring and fault detection method (Nomikos and MacGregor, 1994; Tipping and Bishop, 1997; Chen and Liu, 1999).

Extensions and combinations of PCA with other techniques have been also developed. Dynamic Principal Component Analysis (DPCA) was proposed by Ku et al. (1995) and uses an augmenting matrix with time-lagged variables for taking into account correlations among observations. DPCA has shown to be valid in some practical applications (Russell et al., 2000; Li and Rong, 2006).

Russell et al. (2000) compared PCA, DPCA and CVA (Canonical Variate Analysis) for detecting faults. Lin et al. (2000) developed a Nonlinear Dynamic Principal Component Analysis (ND-PCA) approach for on-line monitoring and fault detection and diagnosis.

Kano et al. (2000, 2002) analyzed several MSPM methods such as conventional methods based on PCA, MPCA (moving PCA) and MS-PCA (multi-scale PCA). Chen and Liu (2002) presented the methods: Batch Dynamic Partial Least Squares (BDPLS) and Batch Dynamic Principal Component Analysis (BDPCA) as improved techniques for on-line batch monitoring.

Detroja et al. (2007) proposed a new approach to fault detection and diagnosis based on Correspondence Analysis (CA) and reported a superiority of CA in comparison to PCA and DPCA probably because of a better information representation. Ge et al. (2009) reported an improved KPCA (kernel PCA) based on the incorporation of a local approach.

A major limitation of PCA-based monitoring is that the PCA model is time invariant, while chemical processes are time-varying. Moreover, it does not possess fingerprinting properties for diagnosis which makes the faults isolation difficult. Independent Component Analysis (ICA) is developed as an alternative technique to PCA for extracting independent components and higher order statistics from multivariate observed data or mixed signals. Thus, ICA allows dealing with non-Gaussian distributed variables.

Kano et al. (2003) reported a statistical process control method based on ICA, evaluated its fault-detection performance and compared such performance with the one obtained with a conventional multivariate statistical process control method using PCA. They used a simple four-variable system and a continuous-stirred-tank-reactor as case study, showing in their simulated results, superiority of ICA over PCA.

Lee et al. (2006) proposed a novel MSPM method based on a modified ICA approach to extract some dominant independent components from normal operating process data and to combine them with statistical process monitoring techniques. Such monitoring method was applied to fault detection in a waste-water treatment process and in the Tennessee Eastman Process TEP (Downs and Vogel, 1993) and compared with conventional PCA.

Zhang (2009) analyzed some drawbacks of the original Kernel Independent Component Analysis (KICA) for the purpose of MSPM and pointed out that any single technique is not sufficient to extract the hidden information from the process variables. Zhang combined and applied both KICA and kernel PCA (KPCA) for fault detection.

The extension of DPCA to ICA led to a new method called Dynamic Independent Component Analysis (DICA). Lee et al. (2004c) proposed DICA to monitor processes with auto and cross-correlated variables and applied it to fault detection in both a simple multivariate dynamic process and the TEP reporting the effectiveness of the method with their simulation results. However, as well as other monitoring methods, DICA would not distinguish real process faults from normal changes in the process.

In conclusion, there are many techniques, mainly MSPM methods, that have been used to monitor chemical processes and detect faults. Most of them reduce the data dimensionality and select the appropriate features that might describe better the process than the original correlated process variables. These features are then used by classification algorithms to diagnose faults. Some of the fault diagnosis methods based on classification are mentioned in the next section.

2.2 Fault diagnosis systems

Fault diagnosis (FD) is seen as a classification problem and therefore, a diagnosis method is referred as a fault classifier. According to Artificial Intelligence (AI), there would be two different kind of diagnosis methods, those based on knowledge and those based on machine learning. On the other hand, in the chemical engineering field or Process Systems Engineering (PSE) from now on, there is a proposed classification of the fault diagnosis methods given by Venkatasubramanian et al. (2003a). This classification, which includes three categories: quantitative model-based methods, qualitative model-based methods and process history-based methods, is based on the manner in which these methods approach the problem of fault diagnosis.

In any of both classifications, a general diagnostic classifier has two characteristics: completeness and resolution, which allow comparing different classifiers.

In order a diagnostic classifier to be complete, the actual fault(s) are required to be in the fault set. In addition, the resolution in a classifier would require that the fault set does not contain too many faults. Then, there should be a trade-off between both concepts, which is reflected in the accuracy of the classifier predictions.

In this thesis both classifications of the fault diagnosis methods according to the AI and chemical engineering fields are unified so that a general scheme is devised, which is suitable to both areas. This classification is based on the kind of process information that is used to construct the diagnosis models. Figure 2.1 summarizes both previous classifications, which lead to the proposed taxonomy in this thesis.

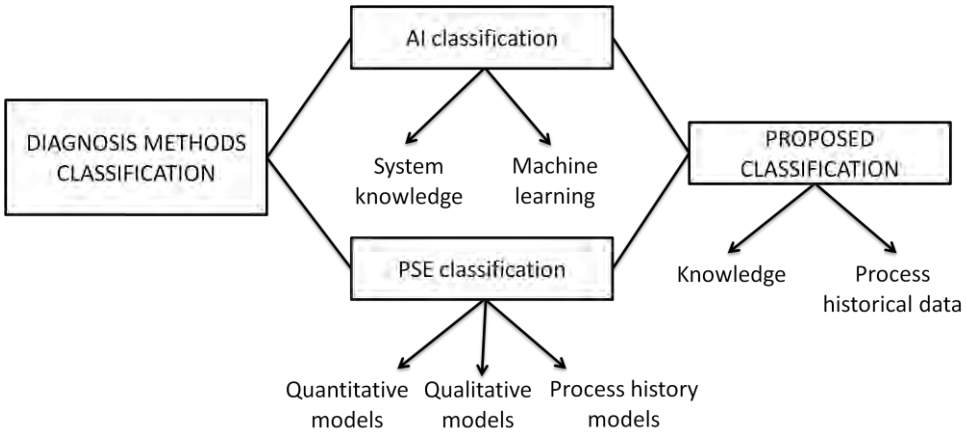

Figure 2.1 Fault diagnosis methods classifications

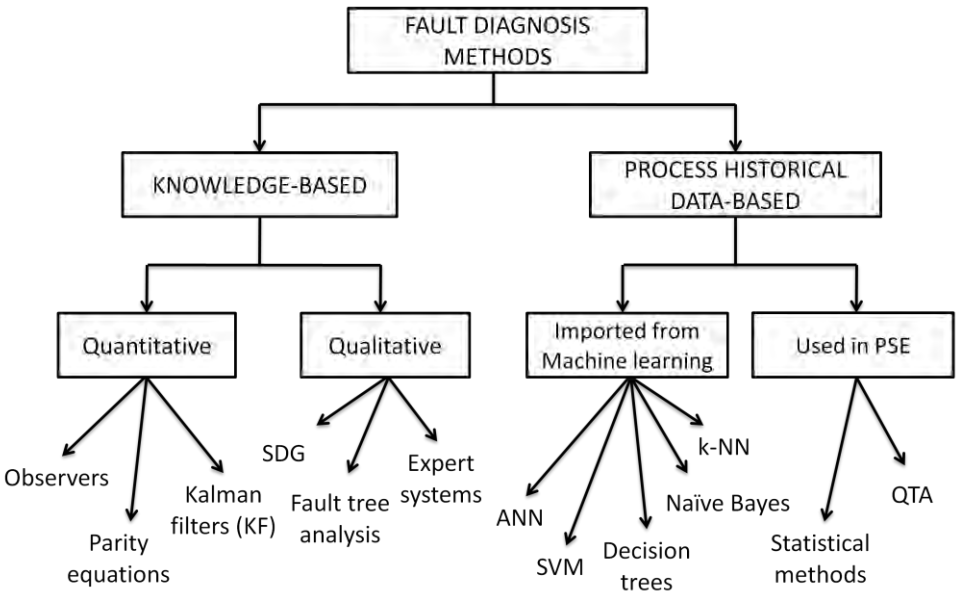

Figure 2.2 Proposed classification of the fault diagnosis methods

Figure 2.2 presents the proposed general scheme in detail. Thus, the process information can be divided in previous process knowledge or experience and process historical data. Regarding the process knowledge, this can be quantitative (knowledge obtained from existing or developed analytical models) or qualitative (knowledge obtained from qualitative rules and models or human experience and expertise). In contrast, the information given by process historical data can be processed with the methods that have been widely used in PSE or with methods imported from the machine learning area. Both class of methods comprise, of course, of qualitative or quantitative techniques. A good example of qualitative methods to process data are the Qualitative Trend Analysis (QTA) and Fault Tree Analysis.

Before breaking down the main categories of the proposed scheme and summarizing some of the several techniques that fit in, it is important to identify the transformations that process measurements go through before the final diagnosis decision is made.

Two important components in such transformations are the input process information and the search technique used. The basic information needed for fault diagnosis is the set of faults and the relationship between the observations (symptoms) and the faults. According to the proposed scheme, this information can be divided into process knowledge and process data.

In PSE, the classification is focused on the method instead of the input information. In this sense, quantitative and qualitative models are developed from a fundamental understanding of the physics of the process using first-principles knowledge. On the other hand, process history-based models are constructed with the past experience in the process, represented by large amount of historical process data.

Then, the series of transformations that process data goes through during FD are divided in four states or spaces:

1. Measurement space
2. Feature space
3. Decision space
4. Class space

The measurement space consists of the input measurements to the diagnosis system without any knowledge related to these measurements. The feature space is a space of points, called features, obtained as a function of the measurements by using the basic problem or process knowledge. The decision space is an space of decision variables, obtained in the feature space, that uses discriminant functions or learning algorithms to meet an objective function (such as minimizing the misclassification). The advantage in developing a feature space is the complexity reduction of the discriminant function. Finally, the class space is a set of numbers that are categorized into classes according to which fault or class the given measurement pattern belongs. This last transformation can be performed by threshold functions.

In general, the FD problem consists of two main stages: feature extraction and classification. A review of several techniques that have been proposed to solve this

challenging problem is provided in the next subsections. The methods are categorized according to the scheme proposed in this thesis. First, the different ways of representing and processing the information are introduced.

2.2.1 Information representation and processing

Despite the FD method used, the process information must be represented and processed in a certain way before it is used as input of the diagnosis method in order to facilitate the diagnosis. Data processing in the specific case of the process history-based methods can be developed in two ways: feature selection and feature extraction.

2.2.1.1 Feature selection and extraction

In feature selection, one selects a subset of relevant process variables or features from the original measurement space before building the diagnosis models. Huang and Wang (2006), Aleixander et al. (2004) and Yong et al. (2007) applied Genetic Algorithm (GA) as feature selection method before classification for improving performance.

Feature extraction is a procedure that facilitates the transformation of the measurement space into a space of fewer dimensions by identifying the relationships between observations of two or more process variables. In chemical engineering is widely applied and referred as quantitative feature extraction. In fact, quantitative feature extraction approaches are the most used before applying the diagnosis methods.

There are two main quantitative feature extraction methods in chemical engineering: statistical approaches and Neural Networks (NN) or most commonly Artificial Neural Networks (ANN), which are mostly used as classification algorithm or diagnosis method.

Statistical methods use data obtained from distributions of classes or process scenarios in order to perform classification. There are plenty of statistical techniques, which have been already brought up in the fault detection methods section because they are mainly used to monitor processes and detect disturbances. However, as these techniques also reduce data dimensionality and extract features, used later as inputs of the classification algorithms, will be then addressed in this subsection from this point of view.

Furthermore, statistical techniques can diagnose faults in some way by identifying the group of variables that are most relevant to the occurring fault. This is conventionally done in PCA with contribution plots, which can be also obtained for CA based on the Q statistic and for the rest of MSPM methods in the same way (Detroja et al., 2007). Some works in which statistical methods are used as feature extraction techniques are next reviewed.

Kassidas et al. (1998) used PCA as feature extraction step to reduce the dimension of the multivariate problem and enhanced the distance-based classification. They apply

Dynamic Time Warping (DTW) in order to classify faults independently of their magnitude, duration and plant production level.

Chiang et al. (2000) reported that Fisher Discriminant Analysis (FDA) and discriminant PLS are better dimensionality reduction techniques than PCA for fault diagnosis, adding that PCA is not well suited because does not take into account the information among classes when determining the lower dimensional representation.

Lee et al. (2006) applied ICA as MSPM method and developed variable contribution plots of the statistical indices for fault diagnosis, reporting better results in comparison to those obtained with PCA.

Verron et al. (2008) presented a supervised procedure for fault detection and diagnosis. Such procedure, after feature selection, develops classification of faulty observations with a Discriminant Analysis, quadratic or linear (QDA or LDA), reporting better classification results than other methods.

Finally Bin He et al. (2009) proposed a non-linear fault diagnosis approach with VW-KFDA (Variable Weighted Kernel FDA) by extracting discriminative features from overlapping fault data. They report a better fault diagnosis performance than the obtained with conventional FDA.

In chemical engineering the concept of qualitative feature extraction methods have been devised but they are not so common to use. Two important methods that employ qualitative feature extraction are expert systems and qualitative trend analysis (Venkatasubramanian et al., 2003).

2.2.2 Fault diagnosis methods

In this section, the different diagnosis methods will be properly addressed. According to Figure 2.2, these are divided in knowledge based methods and data-driven methods.

2.2.2.1 Knowledge-based methods

Knowledge-based fault diagnosis refers to all those models that are built by taking into account knowledge of the process. Thus, these models can be analytic or physical models of the process and relationships or rules among process variables or from some kind of experience given by operators. In this sense, such knowledge can be used qualitatively or quantitatively. Generally, quantitative knowledge can be associated to that given by quantitative models. However, qualitative knowledge can be used for developing either qualitative models or human expertise-based rules.

Quantitative knowledge based-methods

In the past, extensive research efforts were devoted to developing quantitative model-based methods, which were based on Analytical Redundancy (AR) to generate residuals. The residual signals are obtained from the difference between the measured output signal and the output value of a nominal system model, thus helping fault diagnosis. This kind of fault classifiers are called quantitative model-based methods in chemical engineering.

The main advantage of using quantitative model-based methods is the possibility of having some control over the behavior of the residuals. However, several factors such as the system complexity, uncertainty, high dimensionality, process non-linearity and/or lack of good data often make difficult and even impractical to develop accurate mathematical models for the processes (Venkatasubramanian et al., 2003a).

Quantitative methods rely on a set of mathematical relations that describe the interaction between process variables and consistent models. In general, quantitative methods make use of the next kind of equations (Dash and Venkatasubramanian, 2000):

$$\frac{dx}{dt} = Ax + Bu + Ed + Ff \qquad \textit{Eq 2.1}$$

$$y = Cx + Du \qquad \textit{Eq 2.2}$$

consisting of states (x), inputs (u), outputs (y), disturbance (d) and faults (f). There are mainly two kinds of quantitative methods: Residual-based methods and Assumption-based methods (Dash and Venkatasubramanian, 2000).

a) Residual-based methods

All model-based fault diagnosis methods generate inconsistencies between the actual and the expected behavior of a process. These inconsistencies are called residuals and are artificial signals that reflect the potential faults of the system. The residual-based methods consist of two basic steps: residual generation and residual evaluation to identify the cause.

The residuals generation consists of the subtraction between the real outputs from the different functions and the expected values of these functions under normal conditions. Analytical redundancy schemes for fault diagnosis are the most known residual-based methods and are basically signal processing techniques that use state estimation, parameter estimation, adaptive filtering and so on.

Residuals should be close to zero when no fault occurs and show significant values when the process leaves out its normal operation. The generation of the diagnostic residuals requires a mathematical model that can be either analytical, as stated, or a black-box model empirically obtained.

Many papers with different emphasis on model-based approaches have been published over the past three decades. The earliest is due to Willsky (1976) and covers the

design of specific fault-sensitive filters, state estimation methods (parity spaces and observers) and statistical tests on filter innovations (Kalman filters).

Some earlier works using diagnostic observers can be found in the works by Clark (1979) and Massoumnia (1986). Frank and Wünnenberg (1989) presented a disturbance decoupling method based on analytical redundancy. This method allowed differentiating faults and unknown inputs and developed a diagnostic observers approach. Apart from that, it is important to mention that there has not been too much work on observers applications to fault diagnosis in PSE.

On the other hand, Gertler and Luo (1989) presented a design procedure to generate isolate parity equation models chosen from a multitude of suitable models on the basis of sensitivity to the different failures and robustness. Both properties are related to the uncertainties in the selected parameters.

In 1990, Frank outlined the principles of parameter identification and state estimation. He reviewed some techniques with emphasis on the robustness regarding the modeling errors.

Chow and Willsky (1984) and then Gertler et al. (1990) developed techniques based on parity equations obtained by transforming the input-output models generated from process data. In addition, Gertler (1991) presented several residual generation methods including parity equations, diagnostic observers and Kalman filters. This work demonstrated that parity equations and observers lead to identical residual generators once the residual properties have been established.

Parameter estimation methods assume that faults in dynamic systems are reflected by changes in the parameters of the model. Isermann (1984) and Young (1981) analyzed different parameter estimation techniques such as least squares, instrumental variables and estimation with discrete-time models.

Residual evaluation trades off fastness and reliable properties in fault detection. Simple threshold functions, statistical classifiers and neural networks have been used for residual evaluation (Koppen-Seliger et al. 1995).

A general drawback of all the model-based approaches is that their disturbance matrices include only additive uncertainty. In practice, severe modelling uncertainties, due to the involved parameters, come in the form of multiplicative uncertainties.

b) Assumption-based methods

These methods associate residuals to the violation of certain assumptions on the normal behavior of the process. This is the principle of the Diagnostic Model Processor (DMP) method (Petti et al., 1990).

Qualitative knowledge-based methods

Qualitative models, as they are called in PSE, are usually developed based on some fundamental understanding of the process. There are two main kinds of qualitative models: qualitative causal models and abstraction hierarchies (Venkatasubramanian et al., 2003b).

Qualitative causal models apply a strategy of cause-effect about the process behavior. Among these methods, fault-trees and signed digraphs are found. Signed Digraphs (SDG) represent process variables as graph nodes and causal relations by directed arcs. SDG can be obtained from mathematical models of the process or from operational data and operator's experience.

Iri et al. (1979) were the first to use SDG for fault diagnosis. A Cause-Effect (CE) graph derives from SDG and consists of using only valid nodes (abnormal) and consistent arcs. Umeda et al. (1980) obtained SDG from differential algebraic equations of the process and Kokawa et al. (1983) used statistical information about equipment failures and digraphs in order to represent the failure propagation network and thus identify the fault location.

Kramer and Palowitch applied in 1987 a rule based method using SDG for fault diagnosis considering all the possible simulation trees consisting of direct paths from root or fault nodes to symptom nodes. Chang and Yu reported in 1990 some useful techniques for simplifying SDGs with applications to fault diagnosis and then, Vaidhanathan and Venkatasubramanian (1995) used digraph-based models for automated HAZOP analysis. In addition, an improvement of fault resolution in SDG models applying fuzzy set theory was obtained by Han et al. (1994).

Fault Tree Analysis (FTA) was developed at Bell Telephone Laboratories in 1961 and then used by Lapp and Powers in 1977. This technique applies backward chaining from the fault to the primary event that presents the possible root cause. A general FTA consists of four steps (Fussell, 1974):

1. System definition
2. Fault tree construction
3. Qualitative evaluation
4. Quantitative evaluation

Fault trees have been used in risk assessment and reliability analysis (Ulerich and Powers, 1988) and have been developed from digraphs (Lapp and Powers, 1977). The major disadvantage of qualitative models is the generation of large number of hypotheses and therefore poor and spurious solutions. Also, computational cost is high when on-line data is used. However, they possess the distinction of finding all the possible faulty candidates.

Expert Systems (ES) also fit in this category because they are based on some kind of qualitative knowledge of the process, given by an expert human. They are typically made up of a precedent part (series of events) and a consequence part that draws and

relates these events to a known fault. Process data enters to the system in the form of scenarios and consequences as if-then rules.

Knowledge-Based Systems (KBS) developed from expert rules are very system-specific, their representation power is quite limited, and they are difficult to update (Rich and Venkatasubramanian, 1987). However, the advantages are: the development facility, transparent reasoning, the ability to reason under uncertainty, and the ability to provide explanations for the given solutions.

There are many papers applying ES to fault diagnosis. Some of them are: Henley, 1984; Chester et al., 1984; Venkatasubramanian and Rich, 1988; Rich et al., 1989; Ramesh et al., 1992; Quantrille and Liu, 1991; Zhang and Roberts, 1991; Becraft and Lee, 1993; Wo et al., 2000; and Qian et al., 2003.

2.2.2.2 Process history-based methods (data-driven)

As increasingly large amounts of data are collected in most process industries, historical data based methods for fault detection and diagnosis look for extracting the maximum information from these records at a minimal or even null first-principle knowledge of the process.

Knowledge-based approaches depend heavily on the system model, which makes their application difficult for highly non-linear or uncertain systems. In contrast to these methods, data-driven methods do not need an analytical model and rely on data-driven techniques to estimate the system dynamics (Venkatasubramanian et al., 2003c). In many of these techniques, the normal and abnormal operation behaviors are treated as patterns.

According to the proposed taxonomy, the data-driven methods used in PSE include the statistical methods, already mentioned in the feature extraction section, and the Qualitative Trend Analysis (QTA). This method constructs trends from the process measurements in order to explain important events occurring in the process, diagnose and predict future states.

Process disturbances produce distinct trends that can be suitably used for identifying process faults. Trend identification should be robust to momentary signal variations (due to noise) and should capture only gross variations. Janusz and Venkatasubramanian (1991) developed a method for identifying a comprehensive set of initial trends that can be combined to represent any other trends. Such trends were mapped to faults in order to construct the knowledge used for the diagnosis. Later, Bakshi and Stephanopoulos (1994) used the trend representation for diagnosis and supervisory control.

In 2005, Maurya et al., proposed a PCA-QTA based approach consisting of applying QTA on the principal components rather than on the raw data so that the measurement redundancy could be drawn. Apart from these works, many more papers applying QTA can be found in the literature.

Methods imported from Machine learning

Machine learning is a branch of Artificial Intelligence that studies methods that can increase system performance through recorded experience. As their goal is to improve the behavior of a system, learning methods are related to the data representation. Therefore, there are several kinds of machine learning systems. ANN is the most popular machine learning algorithm in PSE.

The main differences between machine learning methods are in the kind of information produced. There are three main classes of learning algorithms: methods with supervised learning, methods with unsupervised learning and methods with reinforcement learning (Langley, 1996).

Supervised Learning is found when there is complete knowledge about the answer required in a determined situation. Its objective is to construct a knowledge-based system for reproducing the output as a function of the input. Classification and regression problems are found in this kind of learning. Nearest Neighbour, Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are some methods belonging to this kind of learning.

In Unsupervised Learning, there is only input information but not output information. The learning method has to extract useful knowledge from the available information. Clustering algorithms, which allow ordering the available information, belong to this kind of learning. In fact, clustering problems are the equivalent to the classification in the unsupervised learning. C-means, Fuzzy C-means, Learning Vector Quantization and Self-Organizing Maps (SOM) are some methods included in this kind of learning.

Regarding Reinforcement Learning, the knowledge about the system quality is only partial. There is no value that corresponds to the output for a determined set of input values but there is a penalization according to the result given by the system. Some algorithms of this kind of learning are: criterion of optimality, brute force approach, temporal difference methods, among others.

On the other hand, learning problems can be classified according to the kind of prediction that is required. According to this criterion, classification problems occur when the prediction is a discrete value and regression problems when the prediction is a continuous or real value.

Fault diagnosis can be treated as a classification problem using machine learning algorithms as classifiers, whose main goal is to classify data in predetermined classes obtained from the process information. A classification problem can be binary when the prediction only can have two values or multi-class when the prediction can have more than two values. In addition, multi-class classification problems can be mono-label (ml) when the prediction is only one value and consists of classifying a set of patterns into a univocal class or multi-label (ML) when the prediction can have more than one value and allows assigning each input data to more than just one class. An example of the ML approach is the simultaneous fault diagnosis.

Fault diagnosis was usually addressed with a ml approach. However, Yélamos et al. (2007) dealt with the problem with a ML approach because it allows representing the training information, obtained by historical data, in the most suitable way for each binary classifier and class.

Two main machine learning methods widely used in chemical engineering are: Artificial Neural Networks (ANN) and Support Vector Machines (SVM). SVM have been introduced more recently and have demonstrated to be a potential technique to be used as fault diagnosis method. These methods will be reviewed in the next subsections.

Apart from these methods, there are others that have not been yet applied in the FD area such as kNN (k-Nearest Neighbor) and others that start to being applied such as Decision Trees (Varga et al., 2008; Ma and Wang, 2009) and Naïve Bayes (Perezyk et al., 2005; Addin et al., 2007).

Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are a method imported from the machine learning and have been used as a fault diagnostic tool in chemical process engineering and cybernetics. In general ANN, used for fault diagnosis, can be classified along two dimensions (Venkatasubramanian et al., 2003c):

- The architecture of the network such as sigmoidal, radial basis and so on, and
- The learning strategy such as supervised or unsupervised learning.

The most popular supervised learning strategy in ANN has been the back-propagation algorithm. In chemical engineering, Venkatasubramanian (1985); Venkatasubramanian and Chan (1989); Watanabe et al. (1989); Ungar et al. (1990) and Hoskins et al. (1991) presented some of the first works in which the usefulness of ANN for the problem of fault diagnosis was demonstrated.

Zhou et al. (2003) proposed an approach to the fault diagnosis in batch processes using ANN. They implemented an input feature extraction process for the neural model and developed an additional radial basis function in the neural classifier, reporting satisfactory detection and isolation of faults.

Then Sharma et al. (2004) showed the potential of ANNs to detect and diagnose process faults in a packed tower. Connection strengths representing the correlation between inputs (sensor measurements) and outputs (faults) were learned by the network using the back propagation algorithm.

There are also a lot of papers on ANN applications in fault diagnosis and it is neither the goal nor the scope of this thesis to provide an exhaustive review of all the reported approaches. However, the literature presented here should provide a good entry point to the literature in this area.

Support Vector Machines (SVM)

Support Vector Machines (SVM) are a supervised learning algorithm and a classification method based on margin maximization and kernels. This algorithm is classified as a data-driven fault diagnosis method in PSE. SVM can also be considered as a hybrid method though, due to some peculiarities. A peculiarity is the use of kernels or functions in the classifiers that best fit the data.

SVM perform a classification task through a supervised learning procedure based on pre-classified data examples; however, they can be used for either supervised or unsupervised learning. Some characteristics of this algorithm are:

- Good performance when the diagnosis models are generalized
- Good tolerance to noise and outliers
- Good efficiency dealing with ML problems in a Multi-class approach.

SVM learn a linear hyperplane that separates a set of positive examples from a set of negative examples with the maximum margin. The margin is defined by the distance of the hyperplane to the nearest point in the positive and negative examples. Learning the maximal margin of the hyperplane is a convex quadratic optimization problem with a unique solution, which supposes an advantage over the conventional algorithms (Vapnik, 1998).

The strategy of the method consists on searching the support vectors. This searching mechanism uses a hypothetical space of linear functions in a highly dimensional feature space, learning with an algorithm of the optimization theory, derived from the Statistic Learning Theory (Cristianini and Shawe-Taylor, 2000).

When examples are not linearly separable or a perfect hyperplane is not needed, non-linear SVM using different types of kernels can be used. In this last case, SVM are converted into a dual form allowing the use of kernel functions to produce non-linear classifiers.

However, it may be preferable to allow errors in the training set so as to maintain a more simple hyperplane. This may be achieved by a variant of the optimization problem, referred as soft margin, in which the contribution of the margin maximization and the training errors can be balanced through the use of a parameter called C, which penalizes the points or samples wrongly classified. This parameter is widely used when applying linear classifiers in order to trade-off bias and variance. Small bias and big variance correspond to situations in which the model is overfitted to the data. In this case, the model has learned the outliers and does not have generalization capacity.

SVM can be applied to binary classification problems. Regarding the multi-class case as the fault diagnosis problem, some class binarization mechanisms have to be applied. One is called one versus all and consists of the training of a classifier per class, choosing as positive examples those from the class of interest (*f*) and as negative examples those from the rest of classes. The result from the classification are called weights or predictions (a real number per class).

In mono-label cases (fault detection) the chosen prediction is that for the class with the biggest weight. For the multi-label case (fault diagnosis) the chosen predictions are all those classes with positive weights.

As summary, SVM have been used for implementing Fault Diagnosis Systems (FDS) because of their efficiency solving ML problems, which has been demonstrated in different machine learning studies and papers (Boser et al., 1992; Manning and Schütze, 1999; Polat and Günes, 2007; Widodo and Yang, 2008; Chen and Hsiao, 2008; Sugumaran et al., 2008; Huang and Wang, 2006; Huang et al., 2008), as well as in PSE applications.

Chiang et al. (2004) investigated the proficiencies of Fisher Discriminant Analysis (FDA), SVM and proximal SVM (PSVM) for fault diagnosis. Then, Kulkarni et al. (2005) applied SVM with knowledge incorporation to detect faults in the TEP, a benchmark problem in chemical engineering.

Yong et al. (2007) employed a SVM-based method for fault diagnosis which incorporates efficient parameter tuning procedures. Such procedures are based on the minimization of the radius/margin bound for SVM's leave-one-out errors, into a multi-class classification strategy, and using the fuzzy decision factor.

Liang and Du (2007) presented a cost-effective FDD method for HVAC (Heating, Ventilation and Air Conditioning) systems by combining the model-based method and SVM. Also, Rocco and Zio (2007) presented a hierarchical structure for distinguishing among operation anomalies in nuclear systems on the basis of measured data. Their approach is based on different formulations of SVM, one-class SVM and multi-class SVM.

Yélamos et al. (2007) addressed the simultaneous fault diagnosis problem from the machine learning viewpoint by using a ML approach and SVM, obtaining good diagnosis performance even when dealing with four simultaneous faults. Later, Yélamos et al. (2009) applied SVM to fault diagnosis in chemical processes and proposed to measure and compare the FDS performance by referring each output to the corresponding input (in the raw data) supplied to the FDS.

Finally, Zhang (2009) uses both KPCA and KICA for fault detection and combine their advantages to develop a non-linear dynamic approach to detect on-line faults for the purpose of MSPM. They develop FD by applying SVM.

There is much more literature in which machine learning methods are applied to fault diagnosis in chemical processes. This field tends to gather techniques that improve the data arrangement, statistical monitoring, feature extraction and classification. In this sense, SVM showed proficiency at constructing models for every process scenario in a plant using historical data.

The main goal of FD methods is to process all the inputs obtained on-line in order to detect and then diagnose abnormal events in the plant in case there are. This thesis is therefore focused on proposing a general FDS and particular approaches to deal with this issue using machine learning methods as data-based fault diagnosis methods.

2.3 Fault diagnosis assessment

Fault diagnosis is as a module of the whole chemical process that can share information with other process operations or modules. Powerful knowledge representation schemes make capture the expertise that operators and control engineers have gained during years dealing with process plants.

There is a close relationship among diagnosis, process operations and design of chemical plants. The proper design of a chemical plant can reduce the burden on the diagnosis task. Moreover, the information given by the diagnosis module can be used for continuously improving the performance of process operations when it is incorporated to the data bases.

Some of the operation modules that can share information with the fault diagnosis module are the next (Venkatasubramanian et al., 2003c):

- Optimal sensor location: A good sensor location enhances fault observability, detectability and separability.
- Data reconciliation: Focuses on detecting sensor faults and sensor biases and on the reconciliation of measurement data. Consists of identifying the biased parameter, estimating the bias and rectifying the sensor measurement. It can be performed in both steady state and transient conditions by removing errors from sensor variables given a collection of data or under time evolutions of sensor variables.
- Supervisory control: Located between regulatory control and planning modules. This system would use the information available from the fault diagnosis to check and monitor the loops in the regulatory control system. If there were changes in the control loops, the supervisory control system would then look for different control configurations or set points that would improve the process operations.

Regarding the FD module, it is of paramount importance to be able to evaluate the diagnosis performance in terms of a reliable and comparable indicator. In this sense, some performance indices developed and used in the area are next reviewed.

2.3.1 Performance evaluation indices

Both fault detection and diagnosis performances need to be quantitatively evaluated. This subsection is focused on summarizing the several indices that have been devised and reported in order to measure such performances.

Most of the data-based fault diagnosis approaches that have been reported use monitoring or detection metrics such as: missing alarm rate (Shao et al., 2009), detection delay (Detroja et al., 2007; Shao et al., 2009), diagnosis success rate (Bin He et al., 2008 and 2009), fault recognition accuracy (Addin et al., 2007), misclassification percentage (Kulkarni et al., 2005; Chiang et al. 2000 and 2004), detection rates (Jakubek and Strasser, 2004) and monitoring indices such as T^2 and Q (Chiang et al., 2004; Jakubek and Strasser,

2004; Shao et al., 2009; Tamura and Tsujita, 2007; Kano et al., 2002; Lin et al., 2000; Ku et al., 1995; Ge et al., 2009). However, a strict fault diagnosis performance index, comparable among techniques, had not been yet addressed.

For this reason, Yélamos et al. (2007, 2009) and Monroy et al. (2010) adopted the F1 score, previously introduced in the machine learning field (Kent et al., 1955; Van Rijsbergen, 1979; Boser et al., 1992).

The F1 score allows having a single assessment by taking into account two complementary concepts (Kent et al., 1955): Precision (Positive Predictive Value) and Recall (also referred as sensitivity or True Positive Rate). In more detail, the fault diagnosis outcomes or responses can be arranged in a confusion matrix as shown in Table 2.1

Table 2.1 Confusion matrix

		Actual value	
		f	$\neg f$
Predicted value	F	a_f	b_f
	$\neg f$	c_f	d_f

Hence, for each fault f, a_f represents the true positive values, b_f the false positive values, c_f the false negative values and d_f the true negative values. Recall (Eq 2.3) accounts for the ratio between the number of positive outcomes that are correctly identified (true positives, a_f) with respect to the total number of actual positive cases (a_f+c_f), taking into account the omissions. Precision (Eq 2.4) measures the ratio between the same number of positive outcomes correctly identified (a_f) with respect to the total number of predicted positive outcomes (a_f+b_f), taking into account in this way the false alarms.

$$Rec_f = \frac{a_f}{a_f + c_f} \qquad \textit{Eq 2.3}$$

$$Prec_f = \frac{a_f}{a_f + b_f} \qquad \textit{Eq 2.4}$$

Therefore, for each fault, the F1 score (Van Rijsbergen, 1979) is given by the following expression:

$$F1_f = \frac{2 \times Prec_f \times Rec_f}{Prec_f + Rec_f}; \quad 0 \le F1_f \le 1 \qquad \textit{Eq 2.5}$$

Furthermore, F1 score is a normalized value that allows measuring the goodness and efficiency of the FDSs and allows comparing different FD approaches.

2.4 Risk analysis in process facilities

Chemical processes are monitored by many variables that demand to be extremely and perfectly controlled. The design of process facilities is based on normal operating conditions. However, but it is expected that such design allows facing any disturbance, although faults are desirable to occur seldom without causing material or human damages.

Risk Analysis (RA) is required in the design stage of process facilities and demands to consider every determining variable in the process. It addresses variations to such variables because of disturbances and consider the response capacity of the facility based on its characteristics and safety elements, which should guarantee an active response.

Hazard and Operability study (HAZOP) is a method designed by ICI (Imperial Chemical Industries, now ORICA) in 1963 for its application to pesticide plant design in order to detect unsafe situations. This method must be integrated to RA and other complementary methods, aimed at the preventive maintenance, in order to face the plant safety.

2.4.1 Typical accident causes in process facilities

Experience from accidents in process facilities allows classifying their causes in three groups (Bestratén Belloví, 1989):

- Component failures. Such as inappropriate design, failures in elements or equipments, failures in the control system, failures in the safety systems or in the pipelines and connections, etc.
- Deviations in normal operating conditions. Uncontrolled alterations of fundamental process variables, failures in services, generation of undesirable sub-products, residuals or impurities, failures in plant start-up and shut down.
- Human and organization errors. Operation errors, incorrect maintenance or reparation work, unofficial works, etc.

All these failures are considered by either risk analysis during the design phase or safety and maintenance programs. On the other hand, human errors, also possible, have to be carefully analyzed in probability terms so that they can be anticipated and controlled and their consequences prevented or mitigated.

2.4.2 Traditional risk analysis

Risk analysis should be quantitative, however because of its complexity, the quantitative analysis tend to be reserved to few situations that require of precise estimations. The adequate application of risk and safety analysis in process facilities should allow determining the limits of all the variables and parameters in which every

process stage is developed in normal and safe conditions. A risk analysis is comprised of five tasks (Faisal and Abbasi, 1998):

1. System definition
2. Hazard identification
3. Probability assessment
4. Consequence analysis
5. Risk results

Hazard identification is a critical step in a traditional risk analysis and has been widely examined in research field (Faisal and Abbasi, 1998). The major focus of a traditional risk analysis is to evaluate a variety of scenarios that may lead to undesirable outcomes. Both the likelihood and the magnitude of these outcomes (consequences) are estimated and displayed as results.

2.4.2.1 Hazard identification

This task is probably the most mature of various disciplines that are included in a risk analysis (API Publication 581, 2001). Potential hazard scenarios need to be identified. There are many techniques for doing so. Hazard and Operability Study (HAZOP) is one of the most known.

Hazard and Operability Analysis (HAZOP)

HAZOP is a rigorous, systematic and critical study of all the predictable faults or deviations in comparison to normal operating situations. It is applied to process facilities at design or functional stage. HAZOP is a qualitative method that allows detecting faults and their consequences so as to adopt preventive measures. It also estimates the danger potential of the deviations and their effects (API Publication 581, 2001).

At the end of a HAZOP, the deviations with more risk (in terms of both frequency and consequence) are prioritized and they must be mitigated as soon as possible. In a well-designed plant, the majority of the identified potential deviations are typically operability issues. However, potential safety concerns and environmental considerations are also identified.

HAZOP is typically performed by a team, familiar with the process, rather than individually, in order to brainstorm the potential hazards most effectively. Some techniques or additional tools that can help to the HAZOP accomplishment are the Failure Modes and Effects Analysis (FMEA), Checklists and Fault Tree Analysis (FTA). It is worthy to note that some of these tools have been also applied as fault diagnosis methods as it is the case of FTA.

There are many works in literature where some HAZOP applications are found and some of them will be next reviewed: Srinivasan and Venkatasubramanian (1998) proposed a framework for representing the HAZOP knowledge for batch chemical plants

and then presented a knowledge-based system, called Batch HAZOP Expert as an implementation of the proposed framework (Srinivasan and Venkatasubramanian, 1998b).

Galluzzo et al. (1999) presented a support system for the hazard and operability studies of batch processes using qualitative models for searching the causes and consequences of the process deviations. Venkatasubramanian et al. (2000) reviewed the progress of automating the process hazard analysis of chemical process plants. Later, Mushtaq and Chung (2000) presented a formalised approach to applying the HAZOP methodology to batch processes and specifically to pipe-less plants.

Mu and Venkatasubramanian (2003) proposed an approach that integrates MSPM and HAZOP for Abnormal Event Management (AEM) of batch processes. Their approach includes three main parts: process monitoring and fault detection based on MPCA, an automated on-line HAZOP analysis and a coordinator. In addition, Eizenberg et al. (2006) proposed and developed a quantitative variation of the HAZOP procedure using separate dynamic models.

Recently, Cui et al. (2010) claimed the HAZOP as an integral part of the process design. They integrated a commercial process design package with one of the HAZOP expert systems.

2.4.2.2 Risk calculation

Probability evaluation is conducted to estimate the probability of occurrence for the scenarios identified in the previous phase of the risk analysis. The most common measure of probability for a scenario is its frequency. On the other hand, consequences of a release from process equipment or pipes vary depending on factors such as material physical properties, material toxicity or flammability, weather conditions, release duration and mitigation actions. The effects may impact plant personnel or equipment, population in the nearby residences and the environment.

There is no single way to measure or present an estimate of the risk of operating a chemical process. In general, risk is calculated with the multiplication of the abnormal event frequency times the magnitude of a determined consequence of such event, as shown in eq 2.6

$$Risk_{Scenario} = F_{Scenario} \times C_{Scenario} \qquad Eq\ 2.6$$

The risk calculation allows prioritizing risks and over all, some equipments or locations in risk. In addition, this prioritization can be taken into account when implementing a preventive maintenance programs. Also for fault diagnosis, risk analysis can help to find out the causes of the occurring faults and the appropriate corrective actions.

2.5 Maintenance management

Despite a chemical plant is well designed, it will not remain safe if it is not maintained. In some industries, maintenance cost is now the second highest or even the highest element of operating costs (Arunraj and Maiti, 2007). Certain critical elements such as product quality, plant safety and the increase in maintenance costs can represent from 15 to 70 % of the total production costs (Bevilacqua and Braglia, 2000).

The main objective of the maintenance process is to make use of the knowledge of failures and accidents in order to achieve the maximum possible safety with the lowest possible cost. The biggest challenge of the maintenance engineer is to implement a maintenance strategy that maximizes availability and efficiency of the equipment, controls the rate of equipment deterioration, ensures the safe and environmental friendly operation and minimizes the total operation cost (Khan and Haddara, 2003).

The concept of Risk-Based Maintenance (RBM) was developed to inspect the high-risk components usually with greater frequency and to maintain them so as to achieve tolerable risk criteria (Arunraj and Maiti, 2007). The RBM methodology provides a tool for maintenance planning and decision-making in order to reduce the probability of failures in equipments and the consequence of such failures. Also, it is designed to study all the failure modes, determine the risk associated to those failure modes and develop a maintenance strategy that minimizes the occurrence of the high-risk failure modes.

2.5.1 Development of maintenance philosophies

Maintenance management techniques have gone through a major process of metamorphosis over recent years. The maintenance progress has been aroused by the increase of complexity in manufacturing processes and variety of products and the awareness growth of the maintenance impact on the environment, personnel safety, business profitability and product quality.

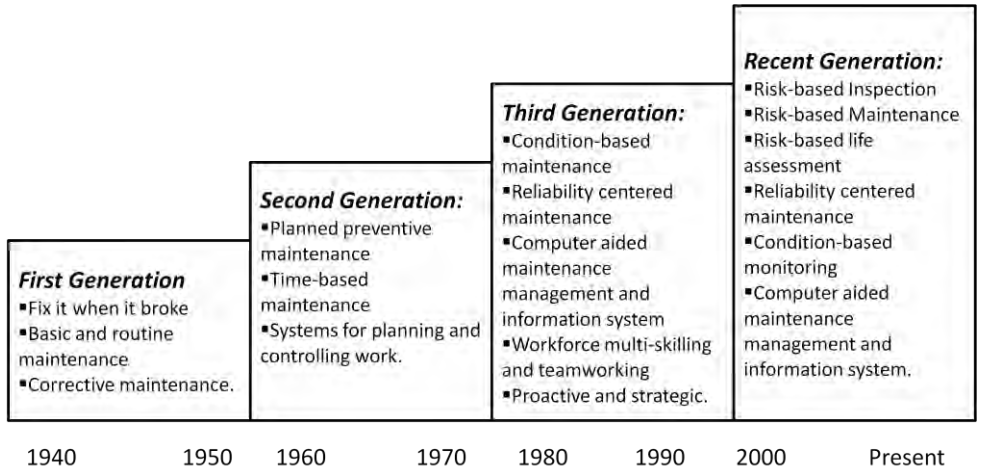

Figure 2.3 Development of maintenance philosophies (Arunraj and Maiti, 2007)

Figure 2.2 shows the development of maintenance philosophies. The maintenance policies have evolved over time and can be categorized in first, second, third and recent generations.

2.5.1.1 First generation

The first generation methodologies belong to the time before Word War II, when industries were not very highly mechanized. Equipment were simple and redesigned, which made them reliable and easy to repair. Machines were operated until they broke down and there was no way to predict failures (Arunraj and Maiti, 2007).

The typical maintenance practices at that time were: basic and routine maintenance, reactive breakdown service (fix it when it breaks) and corrective maintenance (Moubray 1994; Cooke 2003).

2.5.1.2 Second generation

The second generation belongs to the time period between Second World War and the late 1970s. At this time, industries became more complex with increasingly dependency on sophisticated machinery. Maintenance cost became higher than other related operating costs.

Maintenance policies adopted were: planned preventive maintenance, time based maintenance and system for planning and controlling work. However, these policies were criticized for imposing quite often unnecessary treatments, which disrupted normal operations and induced malfunctions, due to missed operations (Khan and Haddara, 2004).

2.5.1.3 Third generation

These maintenance strategies were developed within 1980 and 2000. This generation was typically characterized by:

- Continued growth in plant complexity
- Accelerating use of automation
- Just in time production systems
- Rising demand for standard of product and service quality and
- Tighter legislation on service quality.

Condition-based maintenance (CBM), reliability-centered maintenance (RCM) and computer aided maintenance management were adopted for maintenance during this period (Eti et al., 2006). More techniques included in this generation are listed in Figure 2.3 (Arunraj and Maiti, 2007).

2.5.1.4 Recent generation

Risk-based inspection (RBI) and maintenance methodologies started to emerge and gain popularity beyond 2000. Up till 2000, maintenance and safety were treated as separate and independent activities (Raouf, 2004). However, several authors suggested that an integrated approach incorporating maintenance and safety is the appropriate mean for optimizing plant capacity (Arunraj and Maiti, 2007).

The overall objective of the maintenance process is to increase the profitability of the operation and optimize the total life cycle cost without compromising safety or environmental issues. Inspection and maintenance planning based on risk analysis minimizes the probability of failure and its consequences. Chapter six will address the Risk-Based Maintenance methods (recent generation) in more detail, and more specifically Risk-Based Inspection (RBI), which is widely used in petrochemical industries.

2.6 Benchmark case studies for FDD research

There are some simulated processes that have been widely used not only in FDD area but also in many other research fields in Chemical Engineering. Some case studies are widely used and considered as benchmarks by academics and some of them will be used for testing the methods and approaches proposed in this thesis. The most used and known benchmark is the Tennessee Eastman Process (TEP), which allows simulating a continuous process with 52 process variables. Regarding batch processes, Penicillin Production Process (PPP) has been used by some authors to do research and is also developed and used in this thesis.

2.6.1 Tennessee Eastman Process (TEP)

TEP is an industrial chemical process consisting of a reactor/separator/recycle arrangement involving two simultaneous gas-liquid exothermic reactions and described by Downs and Vogel (1993). Figure 2.3 shows a TEP graph.

The process has 52 process variables and 20 faults or disturbances to be diagnosed. In regard to the process variables, 41 are measured (XMEAS in the original paper) and 11 are manipulated by valves (XMV in the original paper). Process variables are listed in Table 2.2. On the other hand, the 20 faults as different ways of operating the process are listed in Table 2.3.

Some faults in the TEP were not fully described by the authors in the original paper, as it is the case of the faults 16 to 20, reported as unknown. Other faults only differ on the type of disturbance (step or random variation) as it is the case of Faults 3 and 9, the first one due to a step fault and the second one due to a random variation. Faults 14 and

15 are the only faults generated due to a stuck valve. Fault 15, for instance, is due to a sticking in the condenser cooling water valve.

Table 2.2 Process variables of the TE benchmark

NUMBER	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

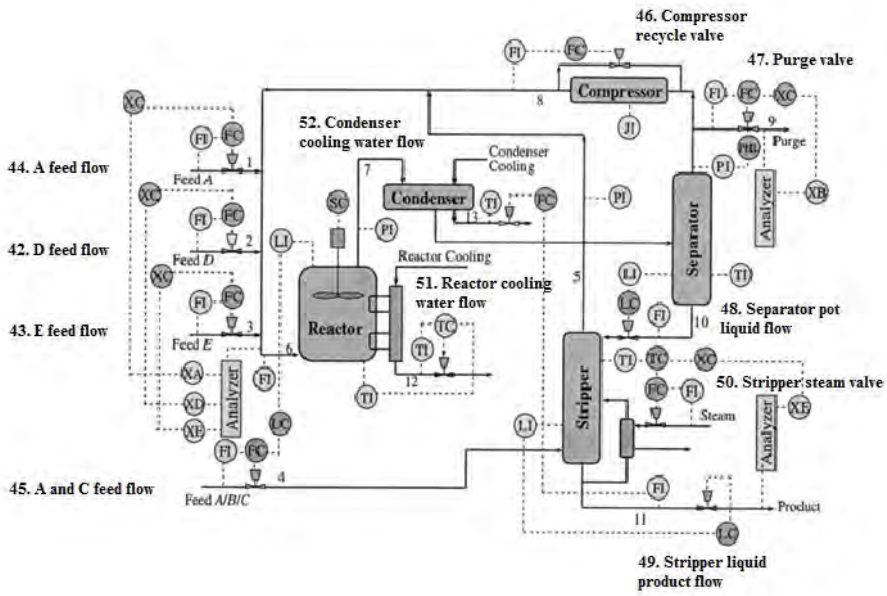

Figure 2.4 Tennessee Eastman Process

Table 2.3 Process disturbances in the TEP

NUMBER	PROCESS FAULT NAME	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

The set of faults of the TEP has been commonly addressed in a partial way. Many works consider only some faults for testing the performance of the proposed FDS (Addin et al., 2007; Kulkarni et al., 2005; Chiang et al., 2004; Detroja et al., 2007; Bin He et al., 2008, 2009; Jakubek and Strasser, 2004; Verron et al., 2008; Musulin et al., 2006).

The whole set of faults has been considered by few authors (Russell et al., 2000; Chiang and Braatz, 2003; Lee et al., 2004c; Yélamos et al., 2007, 2009; Zhang, 2009). Faults

3, 9 and 15 have shown difficulty to be diagnosed; however Musulin et al. (2006) reported to have identified faults 3 and 9 by using a fuzzy logic system, showing an enhancement in the fitness function. In addition, Yélamos et al. (2009) used randomly-selected data models, diagnosing faults 3 and 9 in terms of recall. Fault 15 is however reported as unobservable.

2.6.2 Birol's Penicillin Process (BPP)

The penicillin production process has been used as case study for addressing the problem of batch process monitoring and FDD by several authors (Birol et al., 2002; Yoo et al., 2004; Tian et al., 2009; Ündey et al., 2003, 2004). Penicillin was the first discovered antibiotic (Fleming 1929) and it is still one of the most important in terms of both therapeutic use and annual volume of production (Barber, 1996).

Today, the penicillin production process is economically important and has largely been studied and reported in literature (König et al., 1981; Bruggink et al., 1998; Elander, 2003). Typical modifications in the process are aimed at diminishing the number of downstream operations and get a better product yield or at obtaining a super-productive mold strain.

The effort on improving the efficiency of the penicillin production has resulted in an increase in titre from 2 U/ml in the Fleming's original isolate (one unit of penicillin is equivalent to approximately 0.6 μg) to 70000 U/ml and above in modern strains (Peñalva et al., 1998).

Birol's Penicillin Process (BPP) is one of the most complete mechanistic models developed. BPP has become a test bed for fed-batch fermentation processes, almost in the same way as TEP is for continuous processes.

There is an extensive literature on penicillin production modelling but many of the models developed and reported are too simplified or do not consider the effects of important operating variables such as temperature, pH, agitation power or substrate feed flow rate on the biomass growth and the penicillin production as considered in BPP (Birol et al., 2002).

Therefore, BPP considered as input variables: pH, temperature, aeration rate, agitation power and feed flow rate of substrate. It also introduces the CO_2 evolution term and uses experimental data to improve the simulation of the penicillin production by extending the existing mathematical models (Birol et al., 2002; Tian et al., 2009). The model developed and used in this thesis is an extension of the previous model. However, some minor model modifications are introduced and explained in chapter nine.

2.7 Conclusions

This chapter has presented a review of the state of the art of different methods that have been applied to FDD, within a general process integration and plant safety framework. There is a classification of fault diagnosis methods given in PSE area. However, many FDD techniques have been imported from the machine learning field and such techniques have their categorization based on the type of learning between the inputs and the outputs of the algorithms.

After analysing the different point of views given in both chemical engineering and machine learning areas, this chapter presented an effort to join both classifications in a single scheme, which categorizes the diagnosis methods based on the kind of information used to construct the diagnosis models. The information can be given by previous knowledge of the process or by historical or recorded data from the process. According to such scheme, this thesis applies specifically data-driven fault diagnosis methods imported from the machine learning area.

There are many classification methods that can be used in FDD and the decision to use one or other depends on issues such as: implementation simplicity, completeness and resolution. This thesis is focused on proposing a general FDD approach no matter the method applied. The proposed scheme should be used as guidelines for its implementation and automation in real practice.

An important issue to face is to standardize the different methods that could be used for the same purpose in terms of their final performance by an unique diagnosis index. The use of a single index would allow selecting one or another technique among the plethora existing and devised in the research field. Several indices have been used, such as the detection rate, time to detection, and others, but most of them are referred to the fault detection and not to the diagnosis performance. For that reason, the existence and application of a standard diagnosis index was needed in the field. In machine learning, the F1 score was introduced, and has been recently applied in the fault diagnosis area. This thesis continues using it in order to accomplish that needed standardization among methods and approaches.

Not only corrective actions but also preventive measures depend on FDD. In this sense, Risk Analysis, which is applied mainly in the process design step, would help to take the right corrective actions when a fault is occurring. RA techniques are also applied when there are plant modifications and some of them are used as diagnosis methods. A state of the art of RA has been included in this chapter. Moreover, there are some methodologies that are applied in real practice as part of the plant preventing maintenance so that the faults risk is diminished. A review of some techniques was also brought up in this chapter. The preventive maintenance task is important to consider in parallel to the automation of a FDS approach in order to develop a potential plant safety system. There are no works in FDD literature that propose corrective actions to the possible faults or their consequences, which is still an open research area.

This chapter also reviewed the general case studies that have been considered as benchmarks in the research area. TEP has been widely used as a continuous process benchmark and BPP starts to being considered as such in batch processes.

Concerning TEP, it does not offer complete information about the process and for reproducing the same simulations. Therefore, it is difficult somehow to compare methods. However, as it is widely used by many authors and allows simulating many disturbances, it is used in this thesis. In the case of the BPP, it has been easier to reproduce the mechanistic model in order to compare results.

Finally, a FDS is defined as a module of the chemical process integration. This module could take advantage of the information and knowledge provided from other modules. In the same way, the outputs from the FDS can be used as inputs of the rest of modules. This integration task is still a field to be exploited in future research.

PART II
METHODS

This second part of the thesis will cover all the methods that were used in the thesis work starting from the first step when process data is acquired up to the application of diagnosis or classification methods. All these steps are required to be followed in a Fault diagnosis system. Some techniques can be replaced for others in the cases where their purposes are the same. It is important to see the wide selection of techniques that could be used for the same purposes and it is more important to compare similar techniques in order to improve the performance of the final approach.

Chapter three will deal with the data acquisition, information representation and data pre-processing steps. This chapter includes then the decision-making regarding sampling, data unfolding when required, centering and scaling as data standardization and finally some alternatives to these data pre-processing or standardization ways.

Chapter four will cover the data or feature processing step, by including the main ways to do it such as Feature Selection and Feature Extraction. Recently a new kind of feature processing has been barely used, called Feature Extension. Feature Extraction is mostly done in a diagnosis approach by using statistical techniques. Some Multivariate Statistical Techniques will be then brought up in this section.

Regarding chapter five, this is dedicated to the fault diagnosis algorithms that were used in this PhD, which are methods imported from the Machine Learning and based on historical data. In the case of supervised learning (classification) methods, ANN and SVM were examined. On the contrary, GMM and BIC are investigated as unsupervised learning (clustering) methods.

Finally, chapter six addresses the most recent maintenance management techniques based on risk. Specifically, the Risk-Based Inspection (RBI) methodology, mostly applied in petrochemical industries for preventing maintenance, is widely explained. In general the Risk-Based Maintenance techniques take profit from the risk analysis for prioritizing the risks in a plant and the areas and equipment to inspect.

Chapter 3
DATA ACQUISITION, REPRESENTATION AND STANDARDIZATION

This chapter is focused on the data representation steps and the data standardization or pre-processing. Data acquisition can be considered somehow implicit to data representation. In general terms, the acquired data is represented in such a way that the information can be treated easily and standardized. Centering and scaling are commonly applied as data standardization methods.

Data acquisition is thus more referred to the decision-making concerning to the number of samples, the quality of such data, the data source and the time interval between sampled observations (sampling time). The final data chosen from the whole historical information is used to construct data-based diagnosis models. Data is easily available from the plant records. On the other hand, in case of lack of information, some simulations may be run for obtaining the necesary data.

Let's suppose then that historical data from a continuous process is available. The data would be registered in two dimensions according to Eq 3.1

$$X_f = \begin{pmatrix} x_{11}^f & x_{12}^f & \cdots & x_{1J}^f \\ x_{21}^f & x_{22}^f & \cdots & x_{2J}^f \\ \vdots & \vdots & \ddots & \vdots \\ x_{K1}^f & x_{K2}^f & \cdots & x_{KJ}^f \end{pmatrix}, \qquad X_f \in \Re^{K \times J}, \qquad f = 0..F \qquad \textit{Eq 3.1}$$

where *K* is the total number of samples or observations, *J* the number of process variables and *F* is the number of abnormal process scenarios or faults. In this context, a process scenario can be defined as any occurrence scenario in a chemical process either normal or abnormal. Simultaneous faults in a process can define a type of scenario or the combination from a normal process and the occurrence of a fault after a while can describe another type of scenario.

The concept of class is attributed to single scenarios either normal (class zero) or abnormal (single faults). Thus X_0 corresponds to data under nominal or normal conditions when there are no faults in the process as it should occur. In this thesis, only process scenarios with single faults are dealt and therefore the concepts of class and process scenario will be used without distinction.

Regarding batch processes, process data are first represented in three dimensions: *K*, *J* and *I*. The last one dimension corresponds to the number of batches. Therefore, Eq 3.1 would represent the data from one batch.

Fixing the sampling time is an important decision. It is obvious that in real plants there is experience on establishing this parameter in order to record significant samples of the process. Sometimes, it could happen that the sampling frequency is not the same than the frequency of samples to store. Furthermore, different process variables may have different sampling times. This is the case of quality variables. The first section of this chapter faces the sampling issue.

3.1 Sampling

Chemical processes develop slowly in comparison to other processes involving signals. Thus, sampling time is usually fixed in minutes. According to the state of the art, authors refer to the sampling time by setting it. Regarding the Tennessee Eastman Process, most of the authors sample every 3 minutes (Detroja et al., 2007; Bin He et al., 2008, 2009; Zhang 2009; Lee et al., 2004c; Chiang and Braatz, 2003) or every minute (Yélamos et al., 2007; Yélamos et al., 2009; Monroy et al., 2010). As a matter of fact, in other works the sampling interval is not even mentioned (Yong et al., 2007; Kulkarni et al., 2005; Chiang et al., 2004; Chiang et al., 2000).

Most authors have chosen a sampling interval of 0.5 h for batch processes (Lee, et al. 2004a; Lee, et al. 2004b) and up to one hour (Yoo et al., 2004) due to the existence of off-line measurements. There are also works on batch processes where the sampling time is not mentioned (Ündey, et al. 2004; Ündey, et al. 2003).

There is a way to determine the sampling time by transforming the process variable trajectories to signals by the Fourier transformation. Once data is represented in Fourier transformation, the cut frequency is determined. Then, the signals are filtered in order to remove noise, which can add high frequencies. In the filtered signals, the highest frequency (f_{max}) would be chosen, representing the shortest time. Finally, according to the Nyquist-Shannon theorem, the sampling time (t_s) would be twice f_{max}:

$$t_s = 2 \times f_{max} \qquad \text{Eq 3.2}$$

This process is out of the scope of the thesis, however it would be important to take it into account in the whole diagnosis system and as a future research line.

3.2 Unfolding in batch processes

Regarding data representation, batch processes are different to continuous processes. Historical data is collected and represented in a three-dimensional array, which must be then unfolded into a large two-dimensional matrix.

The original three-dimensional array, usually represented by $\bar{X}$ (*IxJxK*), is composed of the number of process variables (*J*) measured at (*K*) time intervals, where *K* represents the number of measured samples throughout a batch. These dara are reproduced on several batch runs (*I*).

Unfolding is referred as part of the Multi-way PCA (MPCA) technique, proposed by Wold et al. (1987) as an adaptation of the traditional PCA to batch processes. However, in this thesis unfolding is considered as part of the data representation step for batch processes no matter what statistical technique is used after that.

There are mainly two ways of unfolding, which started to being used for PCA-based monitoring of batch processes (Nomikos and MacGregor, 1994). These two unfolding ways are called: variable-wise unfolding and batch-wise unfolding.

3.2.1 Variable-wise unfolding

Variable-wise unfolding joins in one dimension the number of samples *K* plus the number of batches *I* and in this way each observation taken from every batch at a time index is considered as a single multivariate sample. The second dimension considers the number of process variables. Therefore, the final arrangement ($X_f \in \Re^{I \cdot K \times J}$) is showed in Eq 3.3

$$X_f = \begin{pmatrix} x_{111} & x_{112} & \cdots & x_{11J} \\ x_{211} & x_{212} & \cdots & x_{21J} \\ \vdots & \ddots & \ddots & \vdots \\ x_{K11} & x_{K12} & \cdots & x_{K1J} \\ x_{121} & x_{122} & \cdots & x_{12J} \\ \vdots & \ddots & \ddots & \vdots \\ x_{K21} & x_{K22} & \cdots & x_{K2J} \\ \vdots & \ddots & \ddots & \vdots \\ x_{KI1} & x_{KI2} & \cdots & x_{KIJ} \end{pmatrix} \qquad \textit{Eq 3.3}$$

This kind of unfolding is normally used for monitoring on-line batches due to the fact that a new observation sampled will have the same number of variables arranged in the second dimension. Such fact would allow taking directly the new observation without any additional arrangement.

3.2.2 Batch-wise unfolding

Batch-wise unfolding has been more used than variable-wise unfolding (Nomikos and MacGregor, 1994, 1995a, 1995b) because it allows inspecting whether a whole batch is good or bad (poor quality and/or occurrence of a fault). This inspection is the result of arranging a complete batch as a single observation by putting side by side all the measurements of each variable. The final arrangement ($X_f \in R^{I \times K \cdot J}$) is showed in Eq 3.4

$$X_f = \begin{pmatrix} x_{111} & x_{121} & \cdots & x_{1K1} & x_{112} & \cdots & x_{113} & \cdots & x_{1KJ} \\ x_{211} & x_{221} & \cdots & x_{2K1} & x_{212} & \cdots & x_{213} & \cdots & x_{2KJ} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{I11} & x_{I21} & \cdots & x_{IK1} & x_{I12} & \cdots & x_{I13} & \cdots & x_{IKJ} \end{pmatrix} \qquad \textit{Eq 3.4}$$

The main drawback of this unfolding is to assume that data for the complete batch is available. In the case of performing monitoring and diagnosis at time as the new batch (X_{new}) is evolving, such new matrix would not be complete until the end of the batch.

The most suitable approach to overcome this problem would be to build K different monitoring models when performing MPCA for example. However, this implies very large computational and storage requirements, which results unfeasible.

Three methods were suggested by Nomikos and MacGregor (1995a) to monitor the progress of a new batch as new observations become available. All of them are based on filling out the X_{new} matrix between the current time interval *k* and the end of the batch so that the number of observations in the second dimension is $K \cdot J$.

- The first approach assumes that the future observations are in accordance with their mean trajectories calculated from the historical batches. Thus the new batch will operate under normal conditions for the rest of its duration without deviations in its mean trajectories. In this way the X_{new} matrix has to be filled with the zeros in the unknown part. The main drawback is that when appling PCA, the scores would be reluctant to detect an abnormal operation, mainly at the start of the batch.
- The second approach assumes that the future observations in the rest of the batch will remain having the last value sampled. Under this assumption, it is expected somehow that the scores are capable of detecting an abnormality more quickly than with the first approach.
- The third approach consists of predicting the missing values of the X_{new} matrix by projecting the already known observations onto the reduced space and calculating the scores at each time interval.

3.3 Data representation in sets

Historical data is arranged in two dimensions for either continuous or batch processes via unfolding. Thus chemical process data is characterized by X_f matrices of dimensions $K \times J$, which stem from simulations or historical data and where f represents the class to be diagnosed ($f=0,1,...F$). X_0 represents the data under normal operating conditions (*NOC*) and X_F the data under abnormal or faulty operating conditions (*AOC*), where F represents the number of fault.

From the historical process data, three data sets are constructed: training ($\mathfrak{TR}$), validation ($\mathfrak{V}$) and test ($\mathfrak{TS}$) sets, constituted of subsets or matrices that are classified according to the belonging class (f). Thus $\mathfrak{TR}=\bigcup_{i=0}^{F} TR_i$, $\mathfrak{V}=\bigcup_{i=0}^{F} V_i$, and $\mathfrak{TS}=\bigcup_{i=0}^{F} TT_i$. Therefore, matrices TR_f, V_f and TT_f are constructed from the original data (X_f) and have rows smaller than the X_f ones and the same number of columns. TR_f matrices are used as the input of the classification algorithm so as to construct the diagnosis models, while V_f and TT_f matrices are used to evaluate such models. Normally, $\mathfrak{V}$ set is used for parameter tuning of the algorithms applied as well as for evaluating and improving the diagnosis performance. In contrast, $\mathfrak{TS}$ set is used for validating the final diagnosis models and generalizing results.

Either the training or the validation and test subsets go through the same consecutive steps for improving the information and having the information in the same space. The real application of the models to new on-line data is done once the models have been validated.

One important thing to add is that before the standardization data step, a certain number of observations needs to be fixed in the training subsets so that the corresponding diagnosis models are constructed. This is exactly the case for continuous processes. A first trial for choosing the number of observations for each class f is based on the behavior of the process variables. Also, statistical indices as result of process monitoring can help to observe when the process changes from one regime to another and then choose the most meaningful data. The most known and used indices are the Q statistic and Hotelling's T squared from PCA.

These indices are specially useful for making a first estimation of the number of observations to use for constructing the diagnosis models of faulty classes. This number can be determined based on the samples over the control limit (*CL*) of such indices. Then, it can be subject to consecutive improvements by analysing the diagnosis performance of the constructed models when evaluated on validation subsets. However, it is important to mention that the data used to construct diagnosis models has been mostly taken from steady-state or taking random data as it is pointed out in chapter eight.

In concern to batch processes, the whole information from historical batches is included in the data matrices and what has to be determined is the number of historical batches per class for constructing the corresponding diagnosis models. It has been previously reported that when the purpose is to monitor finished batches, usually more than fifty are required for obtaining a representative sample of sufficient size to correctly estimate statistical limits for the normal operating region (MacGregor and Kourti, 1995). In

this sense, more than fifty batches per class should be used to construct the training matrices TR_f.

3.4 Data standardization (centering and scaling)

As part of the information improvement, either training data or the validation and test data, has to be standardized by centering and scaling. Centering is applied by subtracting the mean of each column of the matrix *X* to the samples of the corresponding variable. Thus, the mean trajectory of each variable is subtracted. The centered values represent the variation at time of all the variables from their mean trajectories.

The observations in each column of *X* are also scaled by dividing the centered values of each column by their standard deviation so as to give equal weight to each variable at each time interval, as well as to face the differences in the measurements units between variables. This kind of scaling is typically named auto-scaling.

When centering and scaling are applied, zero mean and unit standard deviation in data are obtained. While centered data explains the variations around the mean trajectories, scaling places observations on an equal footing, relative to their variation. *X** matrix represents a centered and scaled data matrix, which is obtained by the application of Eq 3.5 to each element of *X*.

$$x^*_{k,j} = \frac{x_{k,j} - mean(j)}{std(j)} \qquad Eq\ 3.5$$

where *mean(j)* represents the mean of each process variable *j* and *std(j)* represents the corresponding standard deviation. Thus, Eq 3.6 represents the calculation of the means and Eq 3.7 the calculation of the standard deviations for each variable *j*.

$$mean(j) = \frac{1}{K} \cdot \sum_{k=1}^{K} x_{k,j} \qquad Eq\ 3.6$$

$$std(j) = \sqrt{\frac{1}{K} \cdot \sum_{k=1}^{K} \left(x_{k,j} - mean(j)\right)^2} \qquad Eq\ 3.7$$

The previous equations are referred to the case of two-dimensional data. Notice that *X* and X^* are referred without any subscript because the calculation is valid for both NOC and AOC data. Such equations are valid exactly in that way for continuous processes. In regard to batch processes, the indices would change according to the kind of unfolding.

Auto-scaling has been applied along this thesis and is the most known and applied scaling method. However, other ways of scaling have been proposed for batch processes. Nomikos and MacGregor (1995a) introduced another kind of scaling in batch-wise unfolding, which consists of scaling each variable at each time interval (troughout the batch duration) by its overall standard deviation. Later, Gurden, et al. (2001) named group scaling to the previous method and reported three methods of scaling: auto-scaling or

column scaling, group scaling or single-slab scaling and double-slab scaling, which combines the previous two methods into one.

3.5 Alternative data standardization

There is an alternative to the centering and scaling that has been developed and also applied in some approaches proposed in this thesis. The purpose of such data standardization is still the same, standardize somehow the different magnitude units of the process variables.

It consists of a normalization of the different process scenarios data by using the squared values of the standardized classes with class 0. The main difference with the centering and scaling consists on dealing with the variations of the observations under AOC regarding the observations under NOC, instead of the mean trajectories. The subtraction is then divided by the corresponding observation under NOC, instead of the standard deviation. This arrangement is next presented:

$$X_f^* = \left(\frac{X_f - X_0}{X_0} \right)^2, \quad f = 1,2,\ldots,F \qquad \textit{Eq 3.8}$$

As the resulting values of such standardization are smaller than those obtained with centering and scaling, the squared values would provide more representative information. Class 0 data used in the next steps are just the original process data, otherwise would be zero.

As general conclusions, this chapter has dealt with the data representation and pre-processing of both continuous and batch chemical processes when the final purpose is to construct diagnosis models. However, the same steps are also necesary when such models are validated and tested over different data.

The main difference between representing batch and continuous process data lies on unfolding the three-dimensional data in batch processes (number of observations, number of process variables and number of batches) into a two-dimensional representation.

Centering and scaling are two important steps for standardizing data, consisting of subtracting to all the observations of each process variable the corresponding variable mean and dividing the result by the corresponding standard deviation. An alternative to the centering and scaling is proposed and applied in the validation part.

Chapter 4
FEATURE PROCESSING

This chapter is dedicated to feature processing, the step after the representation of process historical data in the measurement space. Such data goes through the feature space, where normally the data dimensionality is reduced by selecting the best process variables, called attributes or features in this space, that describe better the process. Process variables can also be reduced by finding the features that explain correlations among the original variables. However, variables may also be extended in order to explain the process dynamics.

This chapter addresses three main ways of feature processing:

- Feature extension
- Feature selection
- Feature extraction

4.1 Feature extension

Feature extension consists of including more information regarding the dynamics of the process by finding statistical features that better describe the behavior of the process at each observation. The new features are not explicitly included in the process measurements and may enhance the characterization of the dynamic behavior of the process.

The data arrangement after feature extension results in expanded data matrices $X_{f,ext}$, which may provide valuable information to the learning algorithm, having as many rows as the original matrices X_f and more columns than those corresponding to the original J features.

In this thesis, standard deviations of the samples per variable are mainly considered as the extended features used to generate extended data subsets ($TR_{f,ext}$ and $V_{f,ext}$). Hence, the extended subsets will consist of the J original attributes plus the standard deviation values for each sample and variable ($2 \times J$), once given a time window heuristically determined (I).

This time window considers the actual observation plus some previous observations (*l-1*). Figure 4.1 shows an example of the calculation of the standard deviation for a *j* variable, considering a time window of *l=20* samples.

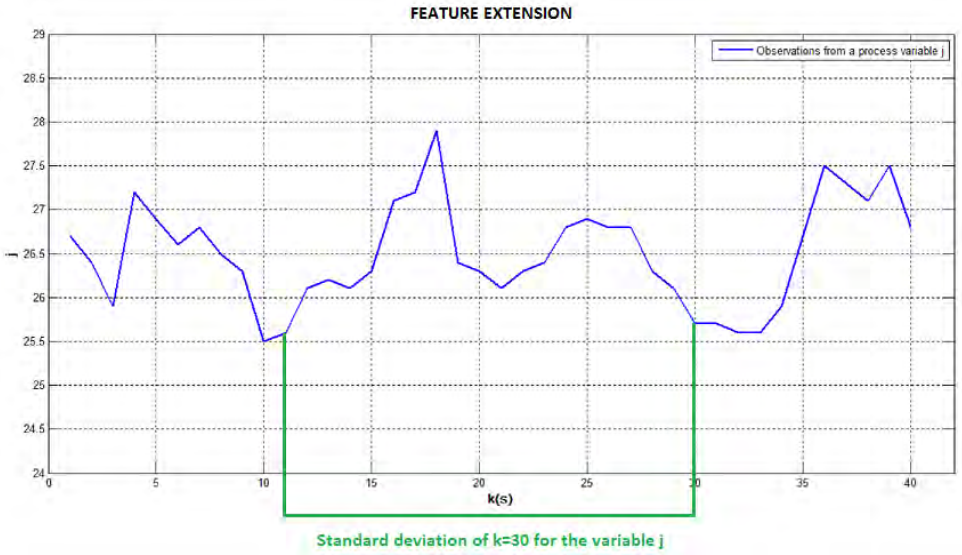

Figure 4.1 Standard deviation as extended feature considering *l=20*.

Feature extension has had few applications in the area because it is more frequent to reduce dimensionality than extending it. Moreover, similar procedures to the feature extension are entailed in the dynamic versions of feature extraction techniques (DPCA or DICA), which consider a time-lagged matrix with previous observations using a time window or time lag. In fact, feature extension could be considered as a previous step to the dimensionality reduction either by selecting or extracting features.

Feature extension has demonstrated to improve the diagnosis performance in comparison to using only the original variables in the data subsets (Yélamos et al., 2007; Monroy et al., 2010). On the other hand, the computational cost is increased in the classification models construction step. Finally, feature extraction has been much more used than feature extension in literature references and seems to perform better. Both methods are used in this thesis and validated in part IV.

4.2 Feature selection

Feature selection and/or reduction is an optimization problem consisting of choosing features from the original set based on some criterion in order to filter out redundant or irrelevant features. Such criterion or filter metrics for diagnosis or classification problems are: correlation and mutual information.

In machine learning, the best subset of features for each class ($J_{f,best}$) is determined on the training subsets TR_f. Therefore, the feature selection algorithms produce as result training data subsets TR_f^{best} with the corresponding $J_{f,best}$. Feature selection uses search algorithms in order to find the optimal or sub-optimal $J_{F,best}$, either by ranking the features with a metric or searching the possible features that compose the optimal subset. Feature selection might reduce the computational cost for the classification step, noise and redundant information.

Some search algorithms used for feature selection are: Simulated Annealing, Greedy Forward Selection, Greedy Backward Elimination and the most known and used, Genetic Algorithms. These algorithms are search heuristics that need a fitness function in order to evaluate the solution domain. Such fitness functions are based on the diagnosis method and the objective function is to maximize the diagnosis performance.

4.2.1 Genetic Algorithms for Feature Selection

Genetic Algorithm (GA) is a global searcher of one solution in an iterative process based on an analogy to the Darwinian natural selection and genetics that consists of creating a chromosomes population (possible solutions set), testing them with an evaluation function and selecting the better chromosomes. Thus, these chromosomes will be the seed of the new population that will be created in the next iteration step.

A fitness function is the evaluation function that assess the quality of a solution in the evaluation step. Thus, the fitness function depends on the diagnosis method applied. On the other hand, cross-over and mutation functions are the main operations that randomly impact the fitness value. Cross-over is the critical genetic operator that allows new solution regions in the search space to be explored consisting of a random mechanism for exchanging genes between two chromosomes, while in mutation, the genes may be occasionally altered.

According to Lo et al. (2004), the construction of a set of chromosomes, consists of four major modules: initialization and encoding method, evaluation, reproduction and generation selection. The parameters to fix in GA when applied as feature selection are:

- Initial population (IP_N) or populations composed of feature combinations, where *N* represents the number of populations
- Maximum number of iterations between each process and/or final condition (*NIT*)
- Number of populations to compute (*NP*)
- Impact function (*IF*) in each iteration step (*cross-over or mutation*)
- Evaluation function (*EF*).

The last population delivered by the algorithm will show the features to be selected. This process should be applied per class. GA do not find the better global solution by itself, but find sub-optimal or acceptable local solutions on a limited time. The chromosomes number or the population size can be chosen.

There should be a trade-off between the population size and the computational time. Whether the population size is small, one can find a local maximum that probably will not be the better solution. On the contrary, whether it is big, the computational cost is high. Thus, one can fix a limited iterations number so as to perform at least an exhaustive searching in the whole problem space. In general, the local optima of GA and other feature selection methods leaded to a wider application of feature extraction techniques.

4.3 Feature extraction

Feature extraction consists of transforming data into a new feature space with lower dimension so as to extract relevant information from the input data. In fact, the main difference between feature extraction and selection is this way to treat data.

There are many multivariate statistical techniques that have been used for feature extraction as it has been pointed out in chapter two. However, those techniques used in this thesis will be addressed in more detail in the next subsections.

The statistical techniques used for feature extraction can be applied either for process monitoring or fault diagnosis. When the only purpose is to monitor a process, a model with historical data under nominal operating conditions is constructed and some statistical indices and the corresponding control limits (*CL*) are calculated. New process data is then projected onto the constructed model. Thus the statistical indices for such data are calculated and compared with the CL for monitoring purposes.

When the purpose is fault diagnosis, models with normal and abnormal data (data under NOC and AOC) have to be constructed. With that aim, process data is transformed into a feature space by applying the statistical techniques for extracting features. The output of such techniques (i.e scores in PCA) is the input of the diagnosis algorithm. Next, the statistical methods used for feature extraction are exposed in detail.

4.3.1 Principal Component Analysis (PCA)

PCA is a multivariable statistical technique that extracts relevant information from a data set to explain process variability. PCA and similar methods are used to develop models that describe the expected variation under normal operating conditions (NOC). Such model under this NOC state in a chemical process can be constructed by using data collected from several periods of plant operation when the performance is good (Ündey and Çinar, 2002). Then, the NOC model is used to monitor the process and detect faults whether the statistical indices overstep their control limits (CL). However, operations under abnormal operating conditions (AOC) can also be modeled whether sufficient historical data is available to develop automated diagnosis of source causes of the abnormal behaviors (Raich and Çinar 1996).

The principal components (PC) are the new set of coordinates that are orthogonal to each other. Supposing that one has the mean centered and scaled data matrix X^* with *K* observations on *J* variables, the first principal component, which indicates the direction of largest variance in data, is defined as the linear combination $t_1 = Xp_1$ subject to $|p_1| = 1$. Then, the second principal component is obtained by the linear combination $t_2 = Xp_2$, which has the next greatest variance not accounted by the first principal component t_1 in a direction orthogonal to this. It is also subject to $|p_2| = 1$ and is uncorrelated to the first PC. Up to *R* principal components are similarly defined.

The loading vectors p_i are the eigenvectors of the covariance matrix of *X*. The corresponding eigenvalues give the variance of the principal components (i.e., $var(t_r)=\lambda_r$). One rarely needs to compute all the *R* eigenvectors since most of the predictable variability in the data is captured with the first principal components.

The number of principal components that provide an adequate description of the data can be evaluated using several methods (Jackson, 1991). One quick and highly-used criterion is the broken-stick rule (Jolliffe, 2003), which is based on the fact that if a line segment of unit length is randomly divided into z segments, the expected length of the r^{th} longest segment is:

$$G = 100\frac{1}{z}\sum_{i=r}^{z}\frac{1}{i} \qquad \textit{Eq 4.1}$$

As long as the percentage of variance explained by each principal component is larger than the corresponding *G*, then one can retain the corresponding principal component. The number of segments corresponds to the maximum possible rank of *X*, *z=min(K,J)*, and the rule should be applied to unit variance-scaled matrices. This criterion is still a quick method to judge if a principal component adds any structural information about the variance in the data or explains only noise.

By retaining the first *R* principal components when applying PCA, the matrix *X* is approximated to Eq 4.2

$$X = \sum_{r=1}^{R} t_r p_r^T + E \qquad \textit{Eq 4.2}$$

In other words, the data matrix *X* can be decomposed in terms of *R* linear principal components with $R \leq J$ as is shown from Eq 4.3 to Eq 4.6

$$X = t_1 p_1^T + t_2 p_2^T + \ldots + t_R p_R^T + E \qquad \textit{Eq 4.3}$$

$$X = TP^T + E \qquad \textit{Eq 4.4}$$

$$X^* = TP^T \qquad \textit{Eq 4.5}$$

$$E = X - X^* \qquad \textit{Eq 4.6}$$

where *T* is the score matrix of dimensions $K \times R$, *P* is the loading matrix of dimensions $J \times R$ and *E* is the residual matrix.

In conclusion, PCA involves the orthogonal decomposition of the data set *X* from a process along the directions that explain the maximum variation in the data. These

directions are the eigenvectors *P* or the principal component loadings. The eigenvalues λ define the corresponding amount of variance explained by each eigenvector. The projections of the data onto the eigenvectors define new points in the principal component space, which constitute the score matrix *T*.

Finally, regarding the algorithms used in PCA to decompose the data matrix, it can be decomposed by either the non-iterative partial least square (NIPALS) or the singular value decomposition (SVD) algorithm. NIPALS was first introduced to compute the principal components in a sequential manner when the number of variables is large (Wold, 1966).

4.3.1.1 Monitoring with PCA

Statistical indices are calculated and compared to the control limits of the reference distribution in order to monitor whether a process is under NOC or AOC. Typically, the *SPE* (or *Q* statistic) and the Hotelling's T^2 statistic are used to represent the variability in the Residual Space (RS) and the Principal Component Space (PCS) respectively (Qin, 2003).

Having established a PCA model based on historical data with common cause variation, future observations can be referenced against this "in-control" model. New data can be projected onto the plane defined by the PCA loading vectors in order to obtain its scores ($t_{k,new} = x_{new} \times p_k$), and the residuals $E_{new} = x_{new} - x^*_{new}$, where $x^*_{new} = t_{R,new} \times p_R^T$; *R* is the number of retained principal components in the model; $t_{R,new}$ is the $1 \times R$ vector of the scores from the model; and p_R is the *JxR* matrix of loadings.

The *Q* statistic is defined in the following way:

$$Q_k = \sum_{j=1}^{J} E(k,j)^2, \qquad E = X - TP^T \qquad \textit{Eq 4.7}$$

And it is calculated with the summation of the squared residuals of each observation and represents the deviations that are not captured in the retained PCs. The *Q* statistic is used to compare residuals of new observations to an upper control limit (Q_α) defined using a set of residuals obtained from data under NOC.

The control limit of *Q* statistic is calculated according to the equation in Jackson and Mudholkar (1979).

$$Q_\alpha = \theta_1 \left[1 - \frac{\theta_2 h_0 (1 - h_0)}{\theta_1^2} + \frac{z_\alpha (2\theta_2 h_0^2)^{1/2}}{\theta_1} \right]^{\frac{1}{h_0}} \qquad \textit{Eq 4.8}$$

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

where λ_k represents the set of eigenvalues.

Specifically, the *Q* statistic quantifies the lack of fit between the observations and the PCA model and denotes the distance of the current observation from the nominal operation surface (Yoo et al., 2007).

On the other hand, the T^2 statistic is the Mahalanobis distance between new data and the center of the NOC data in the space covered by the principal components. The T^2 for each k observation is calculated as follows:

$$T_k^2 = t_R^T S^{-1} t_R \approx \frac{R(K^2 - 1)}{K(K - R)} F_{R,K-R} \qquad \textit{Eq 4.9}$$

where *K* is the number of observations in the reference set; t_R is a vector of *R* scores; *S* is the (*RxR*) covariance matrix of the t-scores calculated during the model development, which is diagonal due to the orthogonality of the t-score values; *R* is the number of PCs retained in the model; and $F_{R,K-R}$ is the F-distribution value with *R* and *K-R* degrees of freedom.

For new samples (i.e., not part of the calibration data), the upper control limit (T_{lim}^2) for this statistic is as established by Montgomery (2005):

$$T_{lim}^2 = \frac{R(K+1)(K-1)}{K^2 - KR} F_{\alpha,R,K-R} \qquad \textit{Eq 4.10}$$

For a new sample x_{new}, if $T_{new}^2 < T_{lim}^2$ and $Q_{new} < Q_\alpha$, one considers the current observation to be in-control with *100(1-α)%* confidence. Otherwise, it is identified as out of control.

Therefore, the *Q* index helps to indicate whether the correlation structure normally present in data is valid or not. When it is not, this likely indicates an abnormal situation. In contrast, the Hotelling's T^2 statistical index measures the distance of the observation to the origin within the hyperplane described by the retained principal components. Large values of the Hotelling's T^2 statistic are indicators of a deviation from the normal operation condition, while the normal correlation is not necessarily broken.

Since the principal component sub-space typically contains normal process variations with large variance that represent signals and the residual sub-space contains mainly noise, the *UCL* of the T^2 defines a larger hypervolume than that from the *Q*. The hypervolume defined by Q_α includes residual components that are mainly noise. This makes Q_α sensitive to small faults that break the normal correlation structure. On the other hand, if a sample exceeds only the T_{lim}^2 but does not violate the Q_α, then this can be interpreted as a shift from the usual operating region without breaking this normal correlation structure. This can be due to faults, but also due to a desired change in the process operation.

Naturally, the T^2 statistic is thus used to detect faults associated with abnormal variations within the model subspace whereas the *Q* statistic is used to detect new events that are not taken into account in such subspace. Therefore, both T^2 and Q statistics are indicators of processes “normality” when their values are below the control limits ($Q \leq Q_\alpha$

and $T^2 \leq T_{lim}^2$). Furthermore, they can also be used to determine the transient stages of a continuous process when occurring faults as it is developed in detail in chapter eight.

4.3.2 Independent Component Analysis (ICA)

Independent component analysis (ICA) is a multivariate statistical method that searches for the components that are both statistically independent and non-Gaussian. ICA has been used as multivariate process monitoring technique and feature extraction method in previous works (Zhang, 2009; Lee et al., 2004c; Kano et al., 2003).

ICA has been developed as a statistical technique that extracts source signals or independent components in a complete statistical sense from mixed signals or multivariate observed data. By using higher order statistical properties of the data, ICA sometimes provides a decomposition into components that are related to the understanding of the process or easier to interpret in comparison to the decomposition obtained by standard PCA techniques (Li and Wang, 2002; Jiang and Wang, 2004; Lee et al., 2006).

A generic ICA model is represented in the next way:

$$\begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_J(k) \end{pmatrix} = A \begin{pmatrix} s_1(k) \\ s_2(k) \\ \vdots \\ s_R(k) \end{pmatrix} \qquad \textit{Eq 4.11}$$

where x_J is a vector of J observations at each time or sample index k, A is an unknown mixing matrix and S is the independent component data matrix. It can be assumed that they are generated as a linear mixture of R ($\leq J$) unknown independent components. When K samples are available, the preceding relationship can be written as:

$$X = AS + E \qquad \textit{being} \quad X \in \Re^{K \times J}, S \in \Re^{R \times K} \qquad \textit{Eq 4.12}$$

where R is the number of independent components, J the number of process variables, K the number of samples, and E the residual matrix.

ICA promises to reveal the driving forces that underlie a set of observed variables or signals and defines a generative model for multivariate observed data. The variables in the model are considered as linear combinations of unknown latent variables (assumed as non-Gaussian and independent), which are called independent components of the observed data.

As the problem of ICA is to estimate the original independent components S and the unknown matrix A from X, an alternative is to find a demixing matrix W so that the components of the reconstructed data vector $\hat{s}$ given in Eq 4.13 becomes as independent of each other as possible.

$$\hat{s}(k) = Wx(k) \qquad \textit{Eq 4.13}$$

The initial step when applying ICA is whitening, which eliminates most of the cross-correlation between random variables. This transformation can be also accomplished by classical PCA. The whitening transformation is represented next:

$$Z = QX = \Lambda^{-1/2} U^T X \qquad \textit{Eq 4.14}$$

where *Q* is the whitening matrix, *U* the orthogonal matrix of eigenvectors and *Λ* the diagonal matrix with the eigenvalues of the data covariance matrix. Once the whitening matrix is available, the following expression can be developed:

$$Z = QAS = BS \qquad \textit{Eq 4.15}$$

where *B* is an orthogonal matrix. By doing this, the problem of finding an arbitrary full-rank matrix *A* is reduced to the simpler problem of finding an orthogonal matrix *B*, which has fewer parameters to estimate. Then, the independent component vectors can be estimated as follows:

$$\hat{s}(k) = B^T z(k) = B^T Q x(k) \qquad \textit{Eq 4.16}$$

Therefore, the relationship between *W* and *B* can be expressed as:

$$W = B^T Q \qquad \textit{Eq 4.17}$$

Whereas ICA finds a set of independent source signals, PCA and FA find a set of signals that are uncorrelated with each other.

Independence implies a lack of correlation but a lack of correlation does not imply independence. Table 4.1 shows a general comparison between the properties of PCA and ICA in order to clarify the essence and the differences between both techniques.

Table 4.1 Comparison between PCA and ICA

PCA	ICA
• Second order statistics method. • Dimensionality reduction. • Represent observations of *J* variables with a smaller number created as linear combinations of the original ones. • Decorrelates variables not makes them independent. • Projects data into a lower dimensional space which maximizes the variance of the original or projected data. • Its components have to be orthogonal each other. • PC must allow foreseeing the original variables. • Not well-suited for FD because does not take into account information among fault classes when determining the lower dimensional representation.	• Involves higher order statistics. • Reveals hidden factors (IC) that underlie sets of variables, measurements or signals. • Searches for the components that are both statistically independent and non-Gaussian. • Reduces higher order statistical dependencies. • Data variables are assumed to be linear mixtures of some unknown latent variables (IC). • Has no orthogonality constraint. • Eliminates cross-correlation among random variables. • Complexity of source signal is less than that of any mixture containing the source signal.

4.3.2.1 Monitoring with ICA

Statistical indices of ICA for monitoring are similar to those indices of PCA. ICA model is based on historical data under NOC and new data under NOC and AOC is projected onto this model. The two statistical indices used to process monitoring are then the T^2 to monitor the systematic part of the process variations and the Q statistic to monitor the residual part of the process variations as in PCA. T^2 is also called D-statistic according to Lee et al. (2006), who apply both indices to process monitoring with ICA.

The independent component matrix S is equivalent to the scores T in PCA. The upper control limit for T^2 cannot be obtained using the F-distribution as in PCA because S is not Gaussian. Kernel density estimation can be used to define T_{lim}^2. On the other hand, Q statistic, also known as SPE statistic has been defined in Eq 4.7 and the upper control limit (Q_α) is calculated according to Eq 4.8. If the number of ICs is chosen such that the majority of non-Gaussianity is included in the ICs, then the residual subspace will contain mostly random noise.

There are other monitoring indices used when ICA is applied (Lee et al., 2004c). Instead of the T^2 index, two indices were proposed (I^2 and I_e^2). The I^2 is equivalent to the T^2, considering the retained independent components in the ICA model, while I_e^2 considers the excluded ICs and compensates for the error resulting from selecting an incorrect number of ICs for the dominant part.

The dominant and excluded parts are normally separated based on the magnitudes of the Euclidean norm. The dominant is then selected based on the assumption that the rows of W with the largest sum of square coefficients have the greatest effect on the variation of the independent component matrix S.

4.3.3 Dynamic Principal Component Analysis (DPCA)

In most cases, state variables are driven by uncontrollable disturbances and random noise that make them to present both auto and cross-correlation. Sometimes such substantial amount of auto-correlation is not only found in the process variables but also in the principal components.

PCA is a multivariate but static technique that gives a linear static model from a data matrix by assuming that there is no correlation over time even though the data might contain dynamic information. In order to deal with this problem, Dynamic PCA (DPCA) was proposed by Ku et al., (1995), which takes into account serial correlations in data.

This approach constitutes a process monitoring method that uses an augmenting matrix with time-lagged variables and has been shown to be valid in different practical applications (Chen and Liu, 2002; Lee et al., 2004c). DPCA performs PCA onto the data matrix in Eq 4.18, which is constructed from the original data matrix X by shifting each sample with "l" more samples. In other words, each sample is augmented as a vector of all the samples taken within the last "l" samples.

$$X_l = \begin{bmatrix} x(1)^T & x(2)^T & \cdots & x(l+1)^T \\ x(2)^T & x(3)^T & \cdots & x(l+2)^T \\ \vdots & \vdots & \ddots & \vdots \\ x(K-l)^T & x(K-l+1)^T & \cdots & x(K)^T \end{bmatrix} \qquad \textit{Eq 4.18}$$

where $x(k)^T = [x_1(k)x_2(k) \ldots x_J(k)]$ is the *J*-dimensional observation row vector at time *k*, *J* the total number of process variables, and l the time lag. By applying PCA to the extended matrix, the residuals obtained should be less correlated and would provide a better statistical basis. The dimensions of the X_l matrix are $(K-l) \times J(l+1)$.

Using DPCA, one can extract the static and dynamic relations from the data by determining the number of time lags and the principal components or the linear relations. A method for establishing the time lag l and the number of principal components is proposed by Ku et al. (1995), which is based on singular values and auto and cross-correlation plots of the scores.

The value of l is usually 1 or 2, which indicates the order of the dynamic system. Regarding to selecting the number of PCs, it is suggested to use parallel analysis and cross-correlation plots of the scores. The difficulty in selecting the correct number of principal components depends on the noise level in data. The desired condition is that the noise level is not high in comparison to the signal level.

Negiz and Cinar (1997) reported that process noise affects DPCA. If process variables are uncorrelated and a DPCA model is constructed, the score variables will be auto-correlated, which may harass an increase of false alarms. For that reason, some filters should be introduced to reduce or remove the autocorrelation. Also, a variation of the DPCA technique that is capable of producing consistent models in the presence of noise can be applied.

DPCA can be used for monitoring purposes as the original PCA by constructing models based on NOC data. Then *Q* and T^2 statistics are calculated in the same way than in PCA. The control limits for both statistical indices are also determined as in PCA. In this thesis DPCA is only applied to batch processes.

4.3.4 Dynamic Independent Component Analysis (DICA)

The extension of DPCA to ICA led to a method called Dynamic Independent Component Analysis (DICA), which applies ICA to the augmenting matrix with time-lagged variables (Lee et al., 2004c; Chen et al., 2005) for developing dynamic models and improving the monitoring performance. This method is able to extract the major dynamic features or source signals from the process and find statistically independent components from auto- and cross-correlated inputs.

ICA is extended to the modeling and monitoring of dynamic systems by augmenting each observation vector with the previous observations l, and stacks the data matrix in the following manner:

$$X_l = \begin{bmatrix} x(k)^T & x(k-1)^T & \cdots & x(k-l)^T \\ x(k+1)^T & x(k)^T & \cdots & x(k-l+1)^T \\ \vdots & \vdots & \ddots & \vdots \\ x(k+p)^T & x(k+p-1)^T & \cdots & x(k+p-l)^T \end{bmatrix}^T \qquad \textit{Eq 4.19}$$

where $x(k)^T$ is the *J*-dimensional observation row vector at time *k*, *(p+1)* or *K-l* is the number of samples or observations, and l is the number of lagged measurements. Thus, the matrix X_l dimension is *J(l+1)×(K-l).* The rows of these matrices represent the features. By performing ICA on the data matrix in Eq 4.19, a Dynamic ICA (DICA) model is extracted directly from the data.

Note that the data matrix arrangement for DICA is a little bit different from the DPCA one. DPCA considers the l observations previous to the current observation, while DICA is applied to the matrix in which each row includes first the current observation plus the l previous samples according to the literature. This should not represent a problem in terms of the extracted ICs but it should be studied in future to make it sure. By applying DICA, the matrices *A* and *S* are also obtained. The inverted independent component matrix (*S*) would correspond to the scores matrix (*T*) in PCA.

4.3.5 Multiway Principal Component Analysis (MPCA)

Multiway feature extraction methods (MPCA and MICA) have been used as an extension of SPC methods (PCA and ICA) to batch processes. These techniques project the information contained in the process-variable trajectories down into low-dimensional latent variable spaces. This fact allows summarizing the correlations across different variables and time instants (Nomikos and MacGregor, 1995a).

In a typical batch run, *j=1,2,...J* variables are measured at *k=1,2,...K* time intervals throughout the batch, and such data is reproduced with several batch runs *i=1,2,...I*. The whole data is arranged in a three-dimensional matrix $\bar{X}$ (*IxJxK*). Neither PCA nor ICA can be applied directly to such matrix. The so called unfolding procedure accommodates for this by re-arranging the data into a two-dimensional matrix. MPCA and MICA are equivalent to unfolding the three-dimensional array $\bar{X}$ into a large two dimensional matrix *X*, then performing centering and scaling and applying PCA or ICA.

MPCA was introduced by Wold et al. (1987) and was successfully applied to image analysis (Geladi et al., 1989) and to some cases in chemometrics (Smilde and Doornbos, 1991). Only later, were MacGregor and Nomikos (MacGregor and Nomikos, 1992; Nomikos and MacGregor, 1994) able to show that MPCA could also handle multi-way batch data for process monitoring.

MPCA is equivalent to unfolding the three dimensional array $\bar{X}$ slice by slice, rearranging the slices into a two dimensional matrix *X* and then performing a regular PCA. In addition, mean centering and scaling to unit variance are applied in the typical and conventional way giving as result the X^* matrix. PCA performed on the mean-centered

data is actually a study of the variation in the time trajectories of all the variables in all batches around their mean trajectories when the unfolding is batch-wise.

MPCA decomposes data *X* into a summation of *R* products of score vectors (t_r) and loading matrices (p_r) plus residuals (*E*), which are as small as possible in least square sense.

$$X = \sum_{r=1}^{R} t_r p_r^T + E = TP^T + E = TP^T + \tilde{T}\tilde{P}^T = \begin{bmatrix} T & \tilde{T} \end{bmatrix} \quad \begin{bmatrix} P & \tilde{P} \end{bmatrix}^T \equiv \bar{T}\bar{P}^T \qquad \textit{Eq 4.20}$$

where *T* represents the scores, *P* the loadings and, *R* the retained PC. Since the columns of $\bar{T}$ are orthogonal, the covariance matrix of the data is:

$$S \approx \frac{1}{I-1} X^T X = \bar{P}\bar{\Lambda}\bar{P}^T \qquad \textit{Eq 4.21}$$

where

$$\bar{\Lambda} = \frac{1}{I-1} T^T T = diag\{\lambda_1, \lambda_2, \ldots, \lambda_{KJ}\} \qquad \textit{Eq 4.22}$$

and

$$\lambda_i = \frac{1}{I-1} t_i^T t_i \approx var\{t_i\} \qquad \textit{Eq 4.23}$$

The score vector t_i is the i^{th} column of $\bar{T}$ and λ_i are the eigenvalues of the covariance matrix in descending order (Qin, 2003). Observe that the last equations are developed for the case of batch-wise ulfolding, which is the most meaningful way of unfolding matrices for monitoring batch processes (Nomikos and MacGregor 1994; Nomikos and MacGregor 1995a).

Usually, a few principal components can express most of the variable correlations and summarize their time trajectories when the variables are highly correlated as well as display out any similarities or differences among batches. Each element of the *t* vector corresponds to a single batch and depicts the overall variability of this batch with respect to the other batches in the data base throughout the whole batch duration.

The loading vectors (p_i) provide the directions of maximum variability giving a simpler description of the covariance structure of the data. Each one of them summarizes the time variation of the measurement variables around their average trajectories and therefore its elements are the weights applied to each variable at each time interval within a batch so as to calculate the *t* score for such batch.

MPCA is popular for batch process data analysis because: it is effective in modelling correlation between variables across the time length of batches (Harrington, 1975), and computational efforts are very low. Some authors have started using the term Unfold Principal Component Analysis (UPCA) instead of MPCA. This is because MPCA is not a multi-way method in the strict sense unlike the so called Tucker models and PARAFAC (Aguado et al., 2007; Gurden et al., 2001; Louwerse and Smilde, 2000; Villez et al., 2009).

4.3.5.1 Batch process monitoring with MPCA

Pioneering work in the area of Fault Detection and Isolation (FDI) for batch processes was performed by Nomikos and MacGregor (Nomikos and MacGregor, 1994; Nomikos and MacGregor, 1995a) and has been successfully applied to industrial processes on several occasions (Kosanovich et al., 1996; Neogi and Schlags, 1998; Tates et al., 1999). Here, the basis is to model the common-cause variation in collected data, obtained under NOC. The resulting model is subsequently used to determine whether a new batch corresponds to this historically recorded normal operating behavior or not. Therefore, the monitoring performance depends heavily upon this NOC data (Van Sprang et al., 2002).

Deviations in process variables during the progress of the batch can provide information about product properties and the likely quality of the final product well before the completion of the batch. Process monitoring and fault diagnosis have been very effective on achieving this goal of process supervision (Ündey et al., 2003).

Batch process monitoring can be divided into three phases: initial, training and application phases, as described by Ramaker et al. (2002). The initial phase consists of collecting a set of historical data from batch runs under NOC. This phase can be a bottleneck because of poorly designed structures of the database. In this sense, the measured data from all NOC batch runs are arranged in a three-way matrix as it has been stated. Moreover, in this phase previous knowledge can be applied to select good batches.

In the training phase, suspicious batches which are not considered to be under NOC are removed from the historical data and then, a representative set of NOC batches is grouped and used for constructing an empirical model. The most common model to use is a PCA model. Prior to performing PCA, data is mean centered and auto-scaled.

More specifically, Multi-way Principal Component Analysis (MPCA) has been successfully applied to batch processes in order to monitor the process and identify when the process shifts to a new operating condition. Such new operating regimes can be abnormal and therefore a fault is detected (Nomikos and MacGregor, 1994; Chen and Liu, 1999; Tipping and Bishop, 1997).

The broken-stick rule is normally applied to the selection of the number of retained principal components in the MPCA model as stated in the PCA section. It is also important to mention that Nijhuis et al. (2003) found out that MPCA method is quite robust to the selection of significant PCs in the sense, out-of-control situations are correctly identified for several numbers of retained components.

Finally, in the application phase, batches are monitored by means of statistical indices derived from the training phase. In the case of monitoring at the end of the batch, usually more than 50 batches are required for obtaining a representative sample of sufficient size to correctly estimate confidence the limits for the normal operating region (MacGregor and Kourti, 1995). After projecting NOC and AOC batches onto the NOC model, the same statistical indices as in PCA are calculated and compared to the corresponding control limits from the reference distribution.

Batches can be compared with a MPCA analysis by plotting their *T* scores and the sum of their squared errors, given by the *Q* statistic (eq 4.24). *Q* statistic represents the summation of the squared residuals for a specific batch and it is used to compare the residuals of new batches to an upper control limit (Q_α). This control limit is defined using a set of residuals obtained from batches that were run under NOC. The control limit for the *Q* statistic has been already defined in the section of monitoring with PCA.

$$Q_i = \sum_{c=1}^{KJ} E(i,c)^2 \qquad \textit{Eq 4.24}$$

In addition, the scores plot represents the projection of each batch onto the reduced plane defined by the principal components. At the end of the batch, the T^2 value for batch *i* is calculated as follows (Yoo et al., 2007):

$$T_i^2 = t_R^T S^{-1} t_R \approx \frac{R(I^2-1)}{I(I-R)} F_{R,I-R} \qquad \textit{Eq 4.25}$$

where *I* is the number of batches in the reference set, t_r is a vector of *R* scores, *S* is the covariance matrix of the t-scores, *R* is the number of PCs retained and $F_{R,I-R}$ is the F-distribution value with *R* and *I-R-1* degrees of freedom. The control limit for the T^2 statistic, used to evaluate new batches, has also been defined in the section monitoring with PCA. The only difference is that the *k* index is replaced by the *i* index, representing the current batch.

$$T_{lim}^2 = \frac{R(I+1)(I-1)}{I^2-IR} F_{\alpha,R,I-R} \qquad \textit{Eq 4.26}$$

The previous statistical indices have been referred for the case of off-line monitoring of already completed batches. However, some techniques to monitor each observation during the batch have been stated in the section 3.2.2 when historical batches are batch-wise unfolded.

It has been also mentioned that the best way to monitor a current batch is by unfolding the reference batches variable-wise. In that case, the calculations of the statistical indices are similar to those executed for continuous processes and also for the control limits. The difference is found during the training stage of monitoring (construction of the MPCA model with batches under NOC). In this stage, the rows of the data matrix are composed of *IxK* observations. Thus, the statistical indices for a new batch, will be referred to each observation *k* of such current batch.

4.3.6 Multi-way Independent Component Analysis (MICA)

For batch processes, ICA can be applied to the unfolded, mean-centered and scaled data matrix as for MPCA, thereby resulting in the MICA technique (Yoo et al., 2004a; Yoo and Vanrolleghem, 2004; Yoo et al., 2004b; Tian et al., 2009).

The principle of MICA is the same than that of ICA but applied to batch processes. When *K* observations are available and there is data from *I* batches, the original ICA equation can be rearranged as:

$$X = AS + E \quad being \quad X \in \Re^{JK \times I}, S \in \Re^{R \times I} \qquad Eq\ 4.27$$

where *R* is the number of independent components and *J* is the number of process variables. MICA can extract the independent scores *S* for NOC and faulty batches just like MPCA. Once again, this is the case of batch-wise unfolding. For variable-wise unfolding, the preceding equation would use the data matrix with dimensions *JxIK*.

In this thesis both MPCA and MICA have been applied as feature extraction techniques for batch processes with fault diagnosis purposes, as it will be detailed in the application section (chapter nine). However, MICA is not used for monitoring purposes, unlike MPCA, due to MPCA is easier to implement than MICA and it is available in some commercial software packages. For that reason, MPCA is practically the technique that is most used for monitoring. Actually, many researchers have focused on the use of PCA-based toolboxes or on the development of their own's.

4.3.7 Batch dynamic principal component analysis (BDPCA)

Batch Dynamic Principal Component Analysis (BDPCA) joins the concept of MPCA with Dynamic Principal Component Analysis (DPCA) (Chen and Liu, 2002). DPCA was developed earlier to implicitly account for dynamics in continuous processes and is based on restructuring the data to identify auto-regressive relationships between measurements across time (Ku et al., 1995).

The BDPCA method is more complicated than DPCA for two reasons. One is that the auto-regressive order needs to be identified as a meta-parameter, in addition to the number of principal components selected. The second is that for the author's knowledge, BDPCA is not commercially available. This means that using BDPCA may incur a severe development cost for commercial practice. Still, both techniques need to be compared to assess the extent of the performance improvement obtained using BDPCA.

In the BDPCA method, a matrix based on time lagged data for one batch is constructed. The method for constructing a BDPCA model consists of ordering the measurements from batch *i* in the matrix $X_l^i \in \Re^{(K-l)\times(l+1)J}$ according to Eq 4.27

$$X_l^i = \begin{bmatrix} x^i(1)^T & x^i(2)^T & \cdots & x^i(l+1)^T \\ x^i(2)^T & x^i(3)^T & \cdots & x(l+2)^T \\ \vdots & \vdots & \ddots & \vdots \\ x^i(K-l)^T & x^i(K-l+1)^T & \cdots & x^i(K)^T \end{bmatrix} \qquad Eq\ 4.28$$

Time lagged matrices for all *I* batches X_l^i, are arranged one after other in the matrix $X_l \in \Re^{I(K-l)\times J(l+1)}$ and thus the model can be determined applying PCA to X_l directly. In

more detail, batches are arranged in the dynamic structure and the resulting matrix is mean centered and scaled to unit variance over its columns.

The time lag l is determined using the method by Ku et al. (1995) as proposed in the original article (Chen and Liu, 2002). This time lag is then $l=2$. Regarding batch monitoring, control limits for BDPCA (T^2 and Q statistics) are calculated in a similar way as in variable-wise unfolded MPCA, which would allow comparing results between both techniques. In addition, as previously stated, variable-wise unfolding is the best way to monitor batches at each sample time.

Two ways of data representation can be used in BDPCA. The first one consists on unfolding the data in the dynamic structure as a dynamic variable-wise unfolding and then mean-centering and auto-scaling. The second one consists on unfolding in the way ($IxK{\cdot}J$), then mean-centering and auto-scaling and finally refolding to the dynamic structure [$I{\cdot}(K-d)xJ(d+1)$], which seems to be the best one (Chen and Liu, 2002). Both ways are addressed in chapter nine. Also, a comparison between MPCA and BDPCA is presented in there.

4.3.8 Multi-model MPCA

Multi-model MPCA was first developed to deal with the problem of batches on-line monitoring when unfolding batch-wise, which represents to have all the measurements during the current batch (Chen et al., 2002). By generating different K models at each time instance k constructed on the unfolded matrices X^{IxkJ} ($k=1,2…K$), it is avoided to fill in the test matrix in order to have the same dimensions than the MPCA model.

Some drawbacks of this method are the extensive number of models that can be originated, which depends on the batch duration, the computational cost that implies such operation, the storage capacity of the computer and the ability to manage such models when monitoring a new batch. An alternative to overcome such inconveniences is to fix a model window, which means to create less models, every certain number of samples.

Apart from that, this method brings up the idea of creating some different models to improve the monitoring performance of batch processes. For example, in fed-batch processes, there are two stages that might be explained by different kind of models during all the batch trajectory. Yoo et al. (2007) proposed to use multiple models for modeling different operating regions and identified such new operating conditions in a process with a new discrimination measure (DM). However, historical batches can be variable-wise unfolded instead of batch-wise. This type of unfolding is applied and discussed in chapter nine.

4.4 Conclusions

This chapter has summarized the diverse ways of processing data to a feature space before constructing diagnosis models. These ways of processing are: feature extension, feature selection and feature extraction.

Feature extension adds features that considers the dynamics of the process by fixing a time window. However, this option is not that usual because it increases data dimensionality and computational cost when creating the diagnosis models. However, it could be done for improving the information as a previous step to the dimensionality reduction.

Feature selection and feature extraction reduce the data dimensionality. The main difference between them consists on the data treatment. The original process variables are reduced according to selection algorithms in the first method while in the second one, the original data is transformed first in features before the extraction, which reduces or eliminates variable correlations.

Feature extraction is actually the most applied method for dimensionality reduction. There is a plethora of multivariate statistical techniques that have been used as feature extraction techniques. PCA-methods have been the most used in chemical engineering; however, there is an increasing research interest in new statistical methods and the comparison with the existing.

PCA and ICA and their dynamic versions (DPCA and DICA) have been summarized. They are used in this thesis for continuous processes. Regarding batch processes, MPCA, MICA, BDPCA and multi-model MPCA are also applied in this thesis. In fault diagnosis, such statistical techniques are applied as previous step to the construction of data-based diagnosis models. However, they have another important aim which consists of process monitoring by constructing projection models based on data under NOC. The projection of new data onto such models results in the generation of statistical indices that allow determining whether a process operates under a normal or an abnormal regime when compared with their respective control limits. Then, the process monitoring allows the fault detection, which is obviously done before diagnosis.

PCA-based methods are the most used for process monitoring and fault detection not only in academics but also in industry. Thus, this thesis uses them for such purposes and the use of the rest of statistical techniques is extended to the data-based diagnosis model construction.

Chapter 5
FAULT DIAGNOSIS ALGORITHMS

Previous chapters have dealt with the process historical data representation and the information improvement on the way to construct data-based models to diagnose faults. This chapter is focused on those algorithms that use the treated data as inputs in order to construct the diagnosis models.

The algorithms to be applied are imported from the machine learning area because they allow reproducing process behaviors and learning to recognize complex patterns as well as making intelligent decisions based on empirical data or historical data. Such algorithms have the capacity to see many possible relationships among the process variables in the input data. Besides, their generalization capacity allows them to produce an useful output in new scenarios that have not been trained, experienced or described with the input data.

As previously mentioned in chapter two, machine learning algorithms can be classified based on the desired output or the kind of learning of the algorithm. Two main categories can be highlighted: supervised and unsupervised learning.

In supervised learning, models are generated to be able to assign input data to desired or appropriate outputs. For creating such models, there is knowledge about the relations between input data and the corresponding output, which represents a label. Classification problems belong to this kind of learning, in which input data is assigned to classes. Therefore, fault diagnosis is associated to classification and fault diagnosis methods to classification algorithms.

In unsupervised learning, a set of input data are modeled without previous knowledge of the outputs. Clustering methods are the most applied to develop unsupervised learning. In fact, clustering methods can also be used in fault diagnosis when there is no knowledge in databases about plant faults. Then, input data must be categorized or clustered in classes. This thesis proposes the use of such methods as an alternative to diagnosing faults apart from the supervised methods that have been already imported to the chemical engineering area and successfully applied.

In this chapter, data-based fault diagnosis algorithms are classified in clustering and classification methods. Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are the classification methods applied in this thesis. SVM is specially exploited as a potential fault diagnosis algorithm in this thesis.

Regarding clustering methods, Gaussian Mixture Models (GMM) with Bayesian Information Criterion (BIC) have been imported and applied as a diagnosis method, which contributes to the use of clustering algorithms in this area.

5.1 Clustering

Clustering is a method of unsupervised learning which consists in dividing a set of observations into subsets, called clusters, in such a way that the observations of the same cluster are similar in some way.

Some clustering methods that have been used in many fields (machine learning, pattern recognition, image analysis, bioinformatics among others) are: k-means, Fuzzy k-means, quality threshold clustering (QT), locality-sensitive hashing, and hierarchical clustering, among others.

Clustering is imported in this thesis as a fault diagnosis algorithm in order to group input data from a chemical process into several classes or clusters as many faults can be found there. GMM are applied in this thesis as clustering algorithm.

5.1.1 Gaussian Mixture Models (GMM)

Mixture models are probabilistic models used to identify subsets within an overall data set and make statistical inferences about the properties of such subsets. This algorithm searches for the model that contains a number of clusters or categories with the highest likelihood. In this way, the number of new classes will be obtained depending on the algorithm output.

A standard GMM has been applied in this thesis as clustering method (Fraley and Raftery, 1998). Within this framework, Bayesian Information Criterion (BIC) has been used in order to determine the optimal number of resulting categories (Schwarz, 1978).

GMM-BIC can be defined as an unsupervised classification algorithm based on the probabilistic modelling of the feature space by means of a mixture of Gaussian distributions. In order to provide a final classification, this approach takes into account both the likelihood of the data regarding the model and the complexity of the model itself, thus avoiding overfitting. Therefore, the main benefit of this technique is that the number and form of the final clusters are obtained by a data-driven process and not provided by the user.

BIC is a criterion for model selection among several models with different numbers of parameters. It also provides a quantitative assessment of the trade-off between the likelihood of the model to fit data and the degrees of freedom of the model. Then, BIC resolves overfitting by adding a penalty term for the number of parameters in the model.

By using GMM-BIC, all the input data, even if one knows they belong to different classes, are classified in some groups which could define a new set of classes. Mathematically, GMM are next described in the context of signal detection according to Benitez and Zorac (2008). If a feature $x_k \in \Re^J$ is generated by the noise in data, a single-component Probability Density Function (PDF) model is proposed:

$$M_1 : f(x_k|\theta_1) = G(x_k, \theta_1) \qquad \textit{Eq 5.1}$$

Where *G* is a Gaussian PDF, $\theta_1 \triangleq \{\mu_1, S_1\}$, and μ_1 and S_1 are the mean and covariance matrices respectively. If x_k contains both the signal and noise, a convex combination of two components is proposed:

$$M_2 : f(x_k|V, \tau, \theta_2) = \tau_1 U(x_k, V) + \tau_2 G(x_k, \theta_2) \qquad \textit{Eq 5.2}$$

Where $U(x_k, V) = (1)/(V)$ is an uniform PDF, *V* is the hypervolume of the data, $\tau \triangleq \{\tau_1, \tau_2\}$, with $\tau_1+\tau_2=1$, and $\theta_2 \triangleq \{\mu_2, S_2\}$. In the GMM framework, an uniform component is often used as a model for outliers (Fraley and Raftery, 1998; Fraley and Raftery, 2002).

Now, given a sequence of features, $X = \{x_0, x_1, \ldots x_{K-1}\}$, the problem is to determine which model M_1 or M_2 fits best into the data *X,* and determine the parameters values of such optimal model *M**. Observe that the problem has been simplified by using observations from one process variable as features in order to define better the problem.

The Bayesian probability framework (Jaynes and Bretthorst, 2003) calculates the probability $p(M_n|X, I)$ of each model given the features *X* and some prior information I, which represents the knowledge about the problem before observing *X*. The integrated likelihood of the models is approximated by using BIC (Schwarz, 1978). BIC is then defined as:

$$BIC_n \cong \log L_n^*(X) - \frac{v_n}{2} \log K \approx \log f(X|M_n, I), \qquad n = 1,2 \qquad \textit{Eq 5.3}$$

where L_n^* and v_n are the maximum likelihood (ML) and the number of independent parameters of model M_n respectively, and *K* is the sample size or the number of observations. A straightforward application of the ML principle to model M_1 produces:

$$L_1^*(X) = \prod_{k=0}^{K-1} G(x_k, \theta_1^*) \qquad \textit{Eq 5.4}$$

where the optimal parameters θ_1^* are: the sample mean μ_1^* and the sample covariance S_1^*. In order to evaluate L_2^*, more ML schemes are needed and the Expectation Maximization (EM) algorithm has to be applied (Dempster et al., 1977).

Once L_1^* and L_2^* are known, the models can be compared based on their BICs. Choosing M1 as the optimal model indicates that no faults are present in data (only one class or cluster). For the case of more than 2 models, similar operations have to be applied for calculating the likelihood of the rest of clusters. The model with higher BIC indicates the number of clusters or classes in the data.

Other common unsupervised classification algorithms could be used to replace GMM, such as the Iterative Self-Organizing Data Analysis Technique (ISODATA). Similarly, BIC index could be replaced by the Akaike Information Criterion (AIC), an alternative measure of the fitting goodness to the probabilistic model.

5.2 Classification

Classification uses supervised learning, which means that the classification models are constructed based on the relation of the input data and the corresponding output in the way of classes.

Classification methods started to be used in the fault diagnosis area because they allow training multiple fault scenarios and cope with the computation burden given by large amounts of historical data. The resulting classification or diagnosis models can be applied to new process data in order to diagnose the occurring scenario, whether normal or abnormal, in the process.

ANN has been the most used machine learning method in the fault diagnosis area and for that reason it is applied in this thesis. Few years ago, SVM was imported to this area, offering better computational capacity and easiness. In fact, this technique is widely exploited as data-based fault diagnosis algorithm in this thesis, and compared with ANN in the validation part.

Some drawbacks could be found when using classification methods in fault diagnosis such as the fact that new faults neither learned nor trained by the algorithm could be classified as a learned class. One way to face this is to use a clustering algorithm besides to the classification. Another way is to apply one-class classification by using the same algorithms. This last case is actually named Anomaly Detection (AD) because it is focused on diagnosing “new” faults that are not modeled. Such technique is formulated in chapter seven when the general approaches to fault diagnosis are arisen.

The next subsections address in more detail the two previously mentioned machine learning algorithms that have been applied as fault diagnosis methods lately in literature and in this thesis: ANN and SVM.

5.2.1 Artificial Neural Networks (ANN) as fault diagnosis algorithm

ANN, used as fault diagnosis method, is suitable for processes for which accurate first-principle models is too difficult or too expensive to develop. ANN are among the most popular pattern recognition methods (Ruiz et al., 2000). ANN can face nonlinear and undetermined processes and learn the diagnosis by means of training data. In fact, they are trained on historical and simulated process data with the aim of detecting and diagnosing disturbances by separating various fault patterns from the normal operating condition according to the process output data.

ANN have the principle of non-linear, distributed, parallel and local processing and adaptation. They can be interpreted as networks of massively parallel distributed processors that can store experimental information and make it available for future use (Haykin, 1994). This information is acquired by learning the process and storing it in interneuron connection strengths, known as synaptic weights.

Originally, the conception of ANNs was an attempt to mimic the neural structure of the central nervous system by means of the arrangement of many simple processing units (neurons) in large connected networks. The nodes and information flows are arranged in such a way that the resulting network has signal inputs and outputs (Zhou et al., 2003).

In fault diagnosis area, ANN are used as a supervised learning algorithm because the correct classification of a set of historical observations (normal and faulty) is known. The Multilayer Perceptron (MLP) architecture of ANN is applied in this thesis, which is next defined. A single neuron in a MLP is represented mathematically by the following equations according to Leger et al., (1998):

$$v_n = \sum_{r=1}^{R} w_{nr} x_r \qquad \textit{Eq 5.5}$$

$$y_n = f(v_n) \qquad \textit{Eq 5.6}$$

where R is the total number of inputs to neuron n; w_{nr} represents the input weights to neuron n; x_r represents the output values from the previous layer, which is the hidden layer(s) with its hidden nodes h_n; v_n is the input of neuron n to the transfer function; f is the transfer function (typically a tangent sigmoidal function); and y_n is the output from the neuron n.

Typical transfer functions include linear functions, the Heaviside function, the logistic function, and hyperbolic tangent functions. The values of the synaptic weights are determined by training the network using the Back-Propagation Algorithm (BPA) which consists of two passes through the network layers.

A detailed description of the BPA is given by Haykin (1994). BPA is the most widely spread calibration algorithm in ANN. BPA belongs to the class of gradient search techniques (Rumelhart, et al 1986). In this thesis, Levenberg-Marquardt BPA is used.

When ANN are applied to FD, the input data consists of the extracted features obtained from any statistical technique mentioned in chapter four, while the output data consists of the predictions of each observation to the faulty classes. In ANN, predictions close to 1 indicate the probability of a fault occurrence.

5.2.2 Support Vector Machines (SVM) as fault diagnosis algorithm

As previously explained in chapter two, SVM are a supervised machine learning algorithm based on margin maximization and kernels. Their strategy consists of developing classifiers that choose the hyperplane with the maximum margin among all the possible

hyperplanes that separate the points of two classes. This hyperplane has the same distance to the samples of one class as to the samples of another. A hyperplane is a geometric figure that has one smaller dimension than the space in which is located and can be linear or nonlinear.

Kernels are projection functions among the spaces for the construction of models or classifiers allowing increasing their complexity and therefore their consistency and variance. Certainly, more complex models can correctly classify data that is not in the training set (known as generalization), but too complex models overfit data and produce a poor generalization. In this sense, the margin flexibility can be parametrized in SVM.

Geometrically, an example of a hyperplane in a two dimensional space ($\Re^2$) is shown in Figure 5.1. The hyperplane is defined by a weight vector (*w*) and a threshold (*b*), which corresponds to the distance from the hyperplane to the origin.

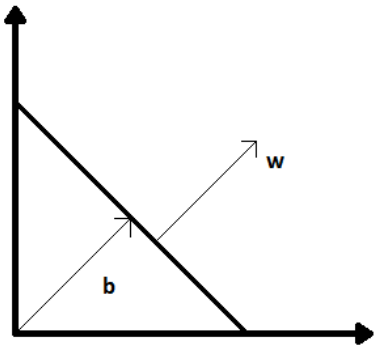

Figure 5.1 Hyperplane example.

A linear hyperplane to classify data from two classes is defined by the next equation:

$$h(x) = sign\left(b + \sum_{j=1}^{J} w_j \cdot x_j\right) \qquad Eq\ 5.7$$

where *J* is the number of features and classes (when there are data from different process scenarios). The weight vector $w = \{w_1, w_2, \ldots, w_J\}$ has a component for each feature. Both parameters *w* and *b* define the hyperplane or classifier, while $x = \{x_1, \ldots, x_J\}$ defines the example or observation to classify.

Figure 5.2 shows a linear hyperplane separating examples from two classes with the best margin, where ϒ corresponds to the margin and the bold points to the support vectors (those that touch the margin). A soft margin allows having some outliers in a class in exchange for augmenting the margin. Such thing is penalized with the term $-\xi/\|w\|$. In SVM , there is a parameter corresponding to the stiffness margin, which is called *C* and is inversely proportional to the flexibility.

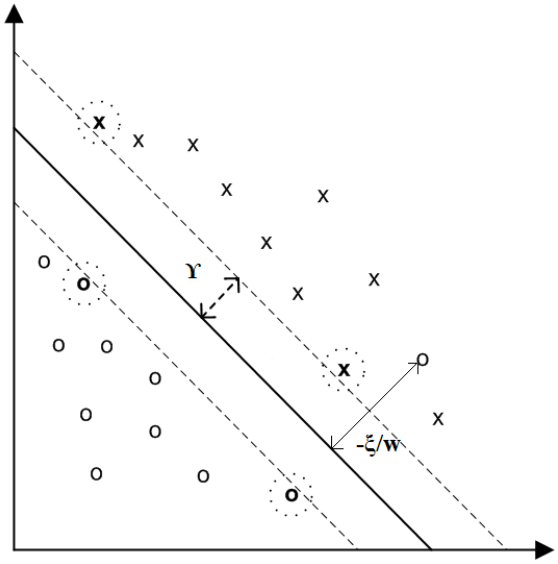

Figure 5.2 Linear hyperplane separating two classes with soft margin.

In a formal way, given a training set of observation-label pairs (x_k,y_k), *k=1,2,...K* where $x_k \in \Re^J$ and $y_k \in \{+1,-1\}$, such observations will be correctly classified by a linear classifier defined by:

$$h(x_k) = \langle x_k, w \rangle + b \qquad \textit{Eq 5.8}$$

where the first term corresponds to the scalar product between the support vectors and the process observation x_k. The classification rule $f(x_k)$ assigns *+1* to an observation x_k when $h(x_k) \geq 0$ and *-1* when is less than *0*.

Learning the hyperplane with maximum margin (w,b) as a convex quadratic optimization problem with an unique solution, holds the next formalization:

$$minimize_{w,b}\|w\|, \quad subject \quad to: \quad y_k\left(\langle w \cdot x_k \rangle + b\right) \geq 1, \quad \forall 1 \leq k \leq K \qquad \textit{Eq 5.9}$$

where *K* is the number of training observations.

The last statement can be transformed into its dual problem, in which the solution can be expressed as a linear combination of the training observations:

$$maximize: \sum_{k=1}^{K} \alpha_k - \frac{1}{2} \sum_{k,j=1}^{K} y_k y_j \alpha_k \alpha_j \langle x_k, x_j \rangle \quad subject \quad to: \sum_{k=1}^{K} y_k \alpha_k = 0, \alpha_k \geq 0, \forall 1 \leq k \leq K$$

Eq 5.10

From this equation, the orthogonal vector to the hyperplane is defined by:

$$h(x) = \langle w \cdot x \rangle + b = b + \sum_{k=1}^{K} y_k \alpha_k \langle x_k, x \rangle \qquad \text{Eq 5.11}$$

The dual form in SVM, in which training examples are inside the scalar products, allows using kernel functions in order to produce non-linear classifiers. If it is supposed to have a non-linear transformation (Φ) of $\Re^J$ to some Hilbert space (ϑ), it is possible to define a scalar product with a kernel function of the kind $K(x, y) = \langle \Phi(x), \Phi(y) \rangle$ and obtain the next dual formulation:

$$maximize: \sum_{k=1}^{K} \alpha_k - \frac{1}{2} \sum_{k,j=1}^{K} y_k y_j \alpha_k \alpha_j K(x_k, x_j), \quad subject \quad to: \sum_{k=1}^{K} y_k \alpha_k = 0, \alpha_k \geq 0, \forall 1 \leq k \leq K$$

Eq 5.12

And the orthogonal vector to the hyperplane:

$$h(x) = \langle w, \Phi(x) \rangle + b = b + \sum_{k=1}^{K} y_k \alpha_k K(x_k, x) \qquad \text{Eq 5.13}$$

Some kernel functions include polynomial, radial basis function (RBF) and sigmoid kernel, which are shown from Eq 5.14 to Eq 5.16 respectively. The kernel parameters in the function should be properly tuned in order to improve the classification accuracy.

$$K(x_k, x_j) = (1 + x_k \cdot x_j)^d \qquad \text{Eq 5.14}$$

$$K(x_k, x_j) = exp\left(-\gamma \left\| x_k - x_j \right\|^2\right) \qquad \text{Eq 5.15}$$

$$K(x_k, x_j) = tanh(K x_k \cdot x_j - \delta) \qquad \text{Eq 5.16}$$

Considering the flexible margin (C) allows trading-off the margin maximization and the outliers in the training set. Thus, the problem is then represented in the next way:

$$minimize: \frac{1}{2} \langle w, w \rangle + C \sum_{k=1}^{K} \xi_k \quad subject \quad to: y_k \left(\langle w, x_k \rangle + b \right) \geq 1 - \xi_k, \xi_k \geq 0, \forall 1 \leq k \leq K$$

Eq 5.17

Finally the result from the SVM algorithm, *h(x)*, represents the predictions of each observation in the training set to a class. It is evident that the algorithm learns the training observations (*k*) with their corresponding label, indicating the class they belong to. In fault diagnosis, such classes represent the number of faults trained with the algorithm. The number of faults registered in the historical data will indicate the number of classifiers or hyperplanes to construct and the number of process variables will indicate the number of features *J*.

The use of a feature extraction technique provides a representation of the data in a feature space, in which the new features are normally a linear function of the original process variables (in case of PCA). This step allows reducing the data dimensionality, as well as avoiding overfitting problems when considering a too large set of features.

Once the classifiers or diagnosis models are constructed, they must be validated on new data from plant, which previously have to go trough the same steps as the training

data (representation, centering and scaling, feature extraction). The predictions will indicate what is occurring in the process and should a fault occur, it will be indicated by positive values in the label corresponding to a specific fault from the set of learned classes.

5.3 Fault diagnosis performance

The performance of the fault diagnosis method has to be evaluated, no matter what algorithm is used. In section 2.3.1, the performance evaluation indices used in the state of the art were brought up and the *F1* score as diagnosis performance index was introduced. In the next paragraphs it will be developed in detail.

Process data is characterized by a given set of faults *F* that may be occurring (or not) and that may be being diagnosed (or not). Hence, a binary matrix *H* can be used to indicate which fault (*f*) is occurring at each sample time (*k*) and another binary matrix *D* can be used to indicate the diagnosis at each sample time. More specifically, the occurrence matrix is given by:

$$H = \begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1F} \\ h_{21} & h_{22} & \cdots & h_{2F} \\ \vdots & \vdots & \ddots & \vdots \\ h_{K1} & h_{K2} & \cdots & h_{KF} \end{pmatrix}, h_{kf} \in \{0,1\}; \forall k = 1,2,..K; \forall f = 1,2,..F \qquad \text{Eq 5.18}$$

And the diagnosis matrix by:

$$D = \begin{pmatrix} d_{11} & d_{12} & \cdots & d_{1F} \\ d_{21} & d_{22} & \cdots & d_{2F} \\ \vdots & \vdots & \ddots & \vdots \\ d_{K1} & d_{K2} & \cdots & d_{KF} \end{pmatrix}, d_{kf} \in \{0,1\}; \forall k = 1,2,..K; \forall f = 1,2,..F \qquad \text{Eq 5.19}$$

Moreover, $\sum_{f=1}^{F} h_{kf} = 0$ indicates that a sample *k* corresponds to normal operating conditions. On the other hand, $\sum_{f=1}^{F} h_{kf} = 1$ indicates that only a single fault is occurring at sample *k*. The unconstrained case $\sum_{f=1}^{F} h_{kf} \geq 1$ implies the consideration of multiple simultaneous faults; this situation can be addressed by the multilabel (ML) approach adopted in this thesis. Yet, the single fault situation ($\sum_{f=1}^{F} h_{kf} \leq 1$) is generally adopted for the validation of an approach. In any case, a good diagnosis is given when $d_{kf}=h_{kf}$ either a fault is happening ($h_{kf}=1$) or not ($h_{kf}=0$).

In order to quantify the diagnosis performance with a single score, the *F1* performance index is adopted, which is commonly used in the field of machine learning (Van Rijsbergen, 1979). As stated in chapter two, this index is a weighted measure of recall (Rec_f) and precision ($Prec_f$). More specifically and given the confusion matrix in Table 2.1, the *F1* score is given by the following expressions:

$$a_f = \sum_{f=1}^{F}\sum_{k=1}^{K}\left[1-\left(h_{kf}-d_{kf}\right)^2\right]\cdot h_{kf} \qquad \textit{Eq 5.20}$$

$$b_f = \sum_{f=1}^{F}\sum_{k=1}^{K}\left(h_{kf}-d_{kf}\right)^2\cdot h_{kf} \qquad \textit{Eq 5.21}$$

$$c_f = \sum_{f=1}^{F}\sum_{k=1}^{K}\left(h_{kf}-d_{kf}\right)^2\cdot d_{kf} \qquad \textit{Eq 5.22}$$

$$d_f = \sum_{f=1}^{F}\sum_{k=!}^{K}\left[1-\left(h_{kf}-d_{kf}\right)^2\right]\cdot d_{kf} \qquad \textit{Eq 5.23}$$

$$Prec_f = \frac{a_f}{a_f+b_f} = \frac{\sum_{f=1}^{F}\sum_{k=1}^{K}\left[1-\left(h_{kf}-d_{kf}\right)^2\right]\cdot h_{kf}}{\sum_{f=1}^{F}\sum_{k=1}^{K}h_{kf}} \qquad \textit{Eq 5.24}$$

$$Rec_f = \frac{a_f}{a_f+c_f} = \frac{\sum_{f=1}^{F}\sum_{k=1}^{K}\left[1-\left(h_{kf}-d_{kf}\right)^2\right]\cdot h_{kf}}{\sum_{f=1}^{F}\sum_{k=1}^{K}d_{kf}} \qquad \textit{Eq 5.25}$$

$$F1_f = \frac{2\cdot Prec_f\cdot Rec_f}{Prec_f+Rec_f} \rightarrow \overline{F1} = \frac{1}{F}\sum_{f=1}^{F}F1_f \qquad \left(0\le\overline{F1}\le 1\right) \qquad \textit{Eq 5.26}$$

It is worth to note that as recall takes into account the number of Type II errors (omissions, c_f) and precision considers the number of type I errors (false alarms, b_f), the *F1* score accounts for both Type I and II errors in the fault diagnosis problem.

From a general point of view, any fault diagnosis method can be described as a first estimation stage that computes the diagnosis matrix *D*:

$$D = g\left(\mathcal{TR}, \mathcal{V}\right) \qquad \textit{Eq 5.27}$$

where the function g represents the calculation with the classification algorithm (SVM or ANN for example) for constructing the matrix *D* from the training ($\mathcal{TR}$) and validation ($\mathcal{V}$) sets. Regarding SVM, the training set is used for constructing data models per classifier based on obtaining the support vectors per each traced hyperplane, which is calculated with kernel functions that separate samples of one class from the rest of classes. Each hyperplane per class is thereby defined by their support vectors. The weights calculation implies a vector calculation, determined by the type of kernel, between the support vectors and each observation from the test set. Such weights define the prediction of the class that is occurring in the test observations.

Once the diagnosis matrix *D* has been obtained, a further evaluation of the diagnosis performance is calculated by:

$$\overline{F1} = f\left(H, D\right) \qquad \textit{Eq 5.28}$$

where the function *f* represents the calculation of the $\overline{F1}$ score from the occurrence and diagnosis matrices (*H,D*).

5.4 Conclusions about fault diagnosis methods

This chapter has properly addressed the fault diagnosis step in a general approach with data-based methods. The methods are imported from the machine learning area, which are divided according to the way the algorithm learns the available information in order to create the diagnosis models.

Two main kind of learning can be summarized: supervised learning, in which there is information about the input data and the corresponding outputs given as labels or classes to which such inputs belong; and the unsupervised learning, where there is only information about the inputs and the algorithm has to extract models from such information.

Classification is the classical problem occurring in fault diagnosis by applying supervised learning. In classification, as many data-based models as the number of faults registered and labeled in the process are constructed. Such models are applied to new data and the predictions obtained reveal the diagnosis of the process.

Artificial Neural Networks (ANN) have been the most known and applied data-based method to fault diagnosis thus they are addressed and applied in this thesis. Recently, Support Vector Machines (SVM) method was imported from the Statistical Learning Theory to the fault diagnosis field. Although SVM have been formulated in previous works, it is highly exploited in this thesis. They allow classifying multiple faults at the same time and use kernel functions to develop the classifiers or diagnosis models.

The application of both algorithms as fault diagnosis methods in an independent step or module as part of a fault diagnosis approach that will be later developed, as well as the use of the same nomenclature is a positive contribution of this thesis, which will allow using different and new methods that could be implemented or improved for this field without losing the raw formulation of the step.

Regarding unsupervised learning, clustering methods develop this kind of learning. Clustering consists of creating groups or clusters with the input data. Clustering methods had not been yet applied to fault diagnosis and this is also a contribution of the thesis. The use of clustering would allow treating unknown data and classifying it into clusters, which would represent different process scenarios or faults. New data acquired from the process would be processed in the same way as the historical data and grouped in some of the modeled clusters or in a different one.

Gaussian Mixture Models (GMM) with Bayesian Information Criterion (BIC) for evaluating the likelihood of the number of gaussians or models is a clustering method that has been here addressed as fault diagnosis algorithm.

The diagnosis performance calculation is addressed by means of the *F1* score, which is properly developed and broken down in this chapter from the predictions, obtained by applying any classification algorithm to training data, up to their comparison with the values of the true occurrence matrix in the validation of the diagnosis models.

Such comparison allows constructing a confusion matrix, which is used to obtain the precision and recall values, needed for the final *F1* performance score.

Chapter 6
RISK-BASED MAINTENANCE MANAGEMENT TECHNIQUES

This chapter is focused on process maintenance and risk prevention. As it was outlined in the state of the art, the maintenance policies have evolved through the time and have been classified in generations. The recent generation is characterized by the inception of Risk-Based Inspection (RBI) and Risk-Based Maintenance (RBM) techniques.

Maintenance tasks during plant operation stage are executed based on policies and methods that can take advantage of the results from the previous Risk Analysis methodologies, such as HAZOP. The two most useful techniques for plant maintenance are exposed in this chapter. Specifically, RBI is used quite frequently in petrochemical industry to develop the maintenance and risk prevention tasks (Chang et al., 2005; Tien et al., 2007; Bertolini et al., 2009; Singh and Markeset, 2009.

6.1 Risk-Based Maintenance (RBM)

The main aim of this methodology is to reduce the overall risk of unexpected failures in operating facilities. The risk based maintenance framework is comprised of two main phases (Khan and Haddara, 2004):

1. *Risk assessment*: It may be quantitative or qualitative. Quantified risk is only appropriate where it is both reasonable and practicable, while the qualitative risk assessment is applicable when the risks are small and well known.
2. *Maintenance planning based on risk.*

Risk Analysis techniques can be categorized into deterministic, probabilistic and a combination of both. The deterministic methods take into account the product, the equipment and the quantification of consequences for various targets. The probabilistic methods are based on the probability or frequency of a hazardous situation. These methods are again cross classified into qualitative, quantitative and semi-quantitative (Arunraj and Maiti, 2007).

It is worthy to mention that the techniques classified in the deterministic and qualitative group are highly used for the hazard identification step in the risk assessment process. On the other hand, techniques categorized in the deterministic and quantitative group are mainly hazard indices, which are used to assess the risk easily.

Inspection and maintenance activities are prioritized based on the quantified risk caused due to failures in components, so that the total risk can be minimized using Risk-Based Maintenance (RBM). The high risk components are inspected and maintained with higher frequency and thoroughness so as to achieve tolerable risk criteria (Brown and May, 2003).

Regarding the maintenance planning, this is focused on decreasing risks under a given criterion and reducing the probability of failure (Khan and Haddara, 2003, 2004). So far Reverse Fault Tree Analysis is used for the calculation of the maintenance interval based on risk (Arunraj and Maiti, 2007). Further information about this technique can be found in Jovanovic, 2003; Khan et al., 2004; Schröder and Kauer, 2004; Arunraj and Maiti, 2007).

6.2 Risk-Based Inspection (RBI)

The American Petroleum Institute (API) RBI Project was initiated in may 1993 by a group sponsored by industries in order to develop practical methods for Risk Based Inspection. The sponsor group was organized and administered by the API and included the following members: Amoco, ARCO, Ashland, BP, Chevron, CITGO, Conoco, Dow Chemical, DNO Heather, DSM Services, Equistar Exxon, Fina, Koch, Marathon, Mobil, Petro-Canada, Philips, Saudi Aramco, Shell, Sun, Texaco and UNOCAL.

API 581 standard presents the Base Resource Document (BRD) with the proposal of the project and the guidelines for applying a RBI program. This document clearly states the next limitation: “to accurately portray the risk, a more rigorous analysis may be necessary, such as the traditional risk analysis” (API 581).

Risk-Based Inspection (RBI) is a method that uses risk as basis for prioritizing and managing the efforts of an inspection program. In an operating plant, a relatively large percentage of the risk is associated to a small percentage of the equipment items. RBI allows the shift of inspection and maintenance resources in order to provide a higher level of coverage on the high-risk items and an appropriate effort on the lower risk equipment. A potential benefit of a RBI program is to increase operating times while improving or at least maintaining the same level of risk.

RBI program is not a full risk analysis but a hybrid technique that combines two disciplines: Risk Analysis and Mechanical Integrity. Some of the steps and techniques in RBI are similar to some in traditional RA. However, the two methods are not interchangeable. The purposes of the RBI program are summarized as follow (Zhaoyang et al., 2011):

1. Evaluate operating units within a plant to identify areas of high risk.

2. Estimate a risk value associated with the operation of each equipment item in a refinery or chemical process plant based on a consistent methodology.
3. Prioritize the equipment based on the measured risk.
4. Design an appropriate inspection program.
5. Allow the management of the risk inspection in an integrated and cost-effective manner.
6. Reduce systematically the likelihood of failures by making better use of the inspection resources, and
7. Identify areas of high consequences that can be used for plant modifications to reduce risk (risk mitigation).

In the past, the risk assessment has been focused on issues related to on-site safety. However, there is an increased awareness of the need to assess risk resulting from:

- On-site risk to employees
- Off-site risk to the community
- Business interruption risks
- Risk of damage to the environment

When the risk associated to individual equipment items is determined and the relative effectiveness of different inspection techniques on reducing risk is quantified, the appropriate information for developing an optimization tool for planning and implementing a RBI is available.

Not all the inspection programs are equally effective for detecting in-service deterioration and for reducing risks. Some inspection techniques are usually available to detect any damage mechanism but each method will have a different cost and effectiveness. It is required then to determine the most appropriate inspection techniques for each high risk equipment.

Inspection reduces risk through a reduction in future failure frequencies by taking corrective and preventive measures in the identified problematic areas (You et al., 2006; Tien et al., 2007). Inspection does not alter fault consequences, but the RBI methodology can identify areas where the consequences of possible failure events can be reduced by changes in the process or mitigation procedures.

Risk cannot be reduced to zero solely by inspection efforts. Some unexpected factors that can cause loss of material in equipments include the following:

- Human error
- Natural disasters
- External events (e.g., collisions or falling objects)
- Secondary effects from nearby units
- Deliberate acts (e.g., sabotage)
- Fundamental limitations of the inspection method
- Design errors
- Previous unknown mechanisms of deterioration

The accuracy and utility of risk analysis could be improved whether process-specific failure data were available. A future phase of the API Project on RBI under consideration is intended to establish an equipment failure database so as to support with high quality data the methodology described in the BRD. RBI should incorporate such data when becomes available, either from industry groups or internally within a company.

6.2.1 Current inspection practices

In process plants, inspection programs are established to detect and evaluate deterioration and damage due to in-service operation. The most comprehensive inspection programs are designed for identifying the in-service deterioration modes and designing an inspection program for detecting specific faults. These programs are based on the understanding of all the potential damage mechanisms in each equipment item.

A risk-based ranking of all the equipment items allows inspecting sophistically and frequently high-risk areas, while low-risks areas are inspected in a commensurate manner with their low risk. Inspection practices, extracted from working documents such as API 510, API Std 653 and API 570, are deeply embedded in the equipment and items prioritization procedure. Some codes and standards from API, ASME and other organizations have been used whenever possible in the screening and evaluation procedures of operating units. In industries where definitive standards have not been yet established, industry experience and good practices provide the basis for the inspection and maintenance. On the other hand, API will issue the document Recommended Practice for Risk-Based Inspection (RP 580).

6.3 A Risk-Based Inspection system

An RBI analysis includes inspection activities, inspection data collection, updating and continuous improvement of the system. A fully integrated Risk-Based Inspection system should contain the steps shown in Figure 6.1 according to API. QPI represents the Quality Improvement Process in the whole program.

Any risk study can only reflect the situation at the time in which data was collected. Although there is lack of data in any system when first established, the RBI program can be established based on the available information, using conservative assumptions for unknown information. As far as knowledge is gained from inspection and testing programs and the database improves, the program uncertainty is reduced.

The knowledge gained from the inspection, engineering evaluation and maintenance is captured and used to update the plant database. The new data will affect the risk calculations and risk ranking for the future. Next subsections will detail the steps of a RBI program according to the API 581 standard.

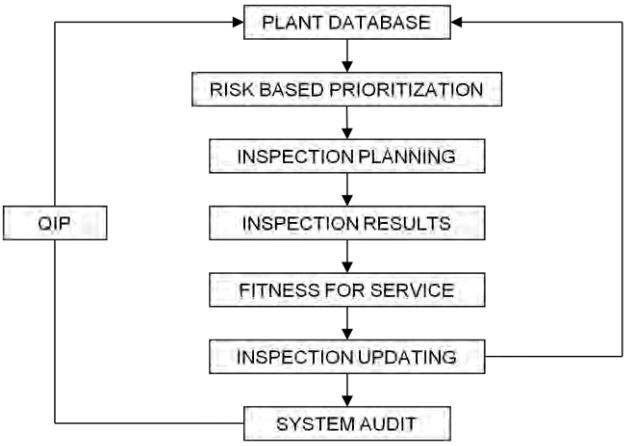

Figure 6.1 Risk-based inspection program for in-service equipment

6.3.1 Plant database structure

A quantitative RBI analysis requires a complete description of the design, fabrication, service conditions and inspection program for each item of equipment to be evaluated. In order to ensure that the analysis produces accurate, reproducible and consistent results from one study to the next, a clear definition of the data used in the analysis for each item has to be established.

Data sheets present information required for a RBI analysis. A completed data sheet is required for each equipment item and consists of the following six sections:

1. **Heading**: Description of the specific equipment item and a list of some of the primary data sources.
2. **Universal information**: Information that applies to all the equipment items in the study (project, plant condition, winter daily low temperature, seismic activity). This section needs only to be completed once.
3. **Mechanical information**: Data that define the design and fabrication of the item.
4. **Process information**: Information concerning to the process, the process fluids and the impact of process conditions on the equipment item.
5. **Inspection/Maintenance Information**: A summary of the significant item's inspection and maintenance history.
6. **Safety System Information**: Record of any detection and/or mitigation devices that serve to protect the equipment item.

6.3.2 Risk-based prioritization

In this step, the RBI analysis starts to be properly applied. The RBI procedure can be applied qualitatively, quantitatively or in a combination. The two approaches provide a

systematic way for identifying areas of potential concern and developing a prioritized list for more in-depth inspection.

The Inspection is an activity intended to limit the risk by reducing one or both of the risk components (likelihood and consequence). Specifically, a quantitative RBI analysis provides risk values for each equipment item and pipe segment. With this level of information, a comprehensive inspection plan can be developed for the unit.

The failure of pressure-containing equipment and subsequent release of hazardous materials can lead to many undesirable effects. The RBI program has condensed these effects into four basic risk categories:

1. Flammable events
2. Toxic releases, only addressed when affect personnel
3. Environmental risks
4. Business interruption.

Quantitative RBI prioritization starts with the extraction of process and equipment information from the RBI database. Various scenarios are then developed to show how leaks may occur and how they can progress into undesirable events. In the quantitative RBI calculation, one of the defining factors in a leak scenario is the size of the hole in the equipment. Since there is a one-to-one correspondence between hole sizes and scenarios, these terms are often used interchangeably. Risk calculation is performed for each scenario, for the four risk categories, if desired. The risk for each equipment item is then found by adding the risk calculation for each scenario (hole size).

6.3.2.1 Likelihood analysis

The likelihood analysis is developed in order to calculate the frequency of a faulty scenario for an specific item or equipment *j*. It is required for the calculation of risk in an item. It is executed by modifying the generic failure frequencies for each equipment FQ_j^g, found in databases, with two factors that reflect identifiable differences in the equipment. The Equipment Modification Factor (FE_j) examines the specific environment in which each equipment operates. The Management System Evaluation Factor (FM_j) is based on an evaluation of the facility's management practices that affect the mechanical integrity of the equipment. Likelihood is then calculated by:

$$FQ_{ad,j} = FQ_j^g \times FE_j \times FM_j \qquad \textit{Eq 6.1}$$

Both modification factors are always positive numbers. Regarding the database of generic failure frequencies, it is based on a compilation of available records of equipment failures. Generic failure frequencies are specific for each type of equipment and each diameter of piping. In addition, a comparison of some approaches for estimating pipe rupture frequencies is reported by Simola et al. (2004).

Equipment modification factor (FE)

The *FE* value allows identifying the specific conditions that can have a bigger influence on the failure frequency in the equipment item *j*. These conditions are categorized into four subfactors $s_{k,j}$:

1. Technical module: analyses materials of construction, the environment and the inspection program
2. Universal conditions that affect all equipment items at the facility
3. Mechanical considerations that vary from item to item
4. Process influences that can affect the equipment integrity.

The equipment modification factor is based on the addition of all the subfactors according to Eq 6.2. The result can be either positive or negative and it will normally range from -10 to +20. At the beginning of the program, the factor can be higher than these values when a piece of equipment has high damage rate and a relatively ineffective inspection history. The final numeric value is converted to the equipment factor (FE_j) as shown in Eq 6.3.

$$\sigma_j = \sum_{k=1}^{4} s_{k,j} \qquad \textit{Eq 6.2}$$

$$FE_j = \begin{Bmatrix} 1/\|\sigma_j\| & if & \sigma_j < -1 \\ 1 & if & -1 \le \sigma_j \le 1 \\ \sigma_j & if & \sigma_j = 1 \end{Bmatrix} \qquad \textit{Eq 6.3}$$

The resulting FE_j is unique for each equipment item and is based on the specific operating environment of the equipment.

Technical modules

The technical modules are the systematic methods used to assess the effect of specific failure mechanisms on the probability of failure. They have four functions:

1. Examine the damage mechanisms under normal and upset operation conditions
2. Establish a damage rate in the environment
3. Quantify the effectiveness of the inspection program
4. Calculate the modification factor to be applied to the generic failure frequency.

The inspection techniques required to detect and monitor one failure mechanism may be totally different from those needed for other mechanisms. These differences are dealt by creating a separate technical module for each damage mechanism. The analysis of the effect of in-service damage and inspection on the failure probability involves the following steps:

1. Examine damage mechanisms and establish an expected damage rate

2. Determine the confidence level in the damage rate
3. Determine the effectiveness of the inspection programs on confirming damage levels and damage rates
4. Calculate the effect of the inspection program on improving the confidence level of the damage rate
5. Calculate the probability that a given level of damage will exceed the damage tolerance of the equipment
6. Calculate the technical module subfactor
7. Calculate the composed technical module subfactor for all the damage mechanisms.

Management systems evaluation factor (FM)

The RBI procedure uses the management systems evaluation factor for an equipment (FM_j) to adjust the generic failure frequencies according to the differences in Process Safety Management (PSM) systems that affect risk in a facility.

At least two questions are required to be respond in order to convert the management systems evaluation score to a management systems evaluation factor:

1. What score would an average plant achieve on the management systems evaluation?
2. How much would the equipment risk be reduced if a plant with an average PSM system were to install a perfect system?

FM is applied equally to all the equipment items and therefore it does not change the ranking of items for inspection prioritization.

6.3.2.2 Consequence analysis

The consequences of releasing a hazardous material are estimated in five distinct steps:

1. Estimate the release rate or the total mass available for release.
2. Determine if the fluid is dispersed in a rapid manner (instantaneously) or slowly (continuous).
3. Determine if the fluid disperses in the atmosphere as a liquid or a gas.
4. Estimate the impacts of any mitigation system.
5. Estimate the consequences.

Flammable, toxic, environmental and business interruption effects are inspected in the RBI methodology. The environmental consequence takes its input directly from the release rate or mass information. Also, business interruption risks are derived directly from the results found for the flammable events.

The outcome of a release refers to the physical behavior of the hazardous material (safe dispersion, explosions, jet fires, flash fires, etc). Outcomes should not be

confused with consequences. Consequences are the adverse effects on people, equipment and environment as a result of the outcome. The effect models, also known as impact criteria, are used to estimate consequences from an outcome.

RBI uses two kinds of impact criteria to estimate consequences from a given outcome: the direct effect model and the probit. Direct effect models are used for flammable, environmental and business interruption consequences, while toxic consequences are estimated using the probit.

6.3.2.3 Calculation of risk

The likelihood and consequences are combined to produce an estimate of risk for each equipment. The items can then be ranked based on the risk calculation, but the likelihood, consequence and risk are all stated separately, identifying the highest contributor to risk.

Given the RBI definition of risk as the product of the consequence ($C_{s,j}$) and the likelihood of failure ($F_{s,j}$), the risk for a scenario *s* and an equipment *j* is:

$$Risk_{s,j} = C_{s,j} \times F_{s,j} \qquad Eq\ 6.4$$

For each equipment item, its risk is the addition of the risks of all the item's scenarios *S*. The units of risk depend on the consequence of interest. In the RBI approach, ft^2 per year are typically the units for flammable or toxic consequences and dollars per year for environmental or business interruption. The risk for an equipment item *j* is then:

$$Risk_j = \sum_{s=1}^{S} Risk_{s,j} \qquad \forall\, j \qquad Eq\ 6.5$$

where $Risk_s$ is the risk for a scenario (ft^2 or $ per year) and $Risk_{item}$ the risk per equipment item (ft^2 or $ per year). Therefore, the items can be prioritized with the risk calculation. However, some API standards also allow identifying areas of potential concern as next exposed.

6.3.3 Development of inspection programs to reduce risk (Inspection planning and results)

Once the risks have been calculated for each equipment or pipeline, the items with higher risk values are prioritized for developing inspection. Inspection programs have two purposes:

1. Development of inspection programs that deal with the types of damage the inspection should detect. The appropriate inspection techniques to detect such damages are selected.
2. Reducing risk through inspection: By applying RBI tools the risk is reduced and the inspection programs are optimized.

Inspection reduces the probability of failure. Many factors (design errors, fabrication flaws, malfunction of control devices) can lead to equipment failure. However, the in-service inspection is mainly concerned with the detection of progressive damage. The probability of failure due to such damage is a function of four factors:

1. Damage mechanism and resulting type of damage
2. Rate of damage progression
3. Probability of damage detection and future damages prediction with inspection technique(s)
4. Tolerance of the equipment to the type of damage.

The purpose of an inspection program is to define and perform the activities necessary to detect deterioration of in-service equipment before failures occur. An inspection program is developed by systematically identifying:

- What type of damage to look for
- Where to look for it
- How to look for the damage (what inspection technique)
- When (or how often) to look.

Some approaches to the RBI inspection planning are also found in literature (Straub and Havbro, 2005; Tien et al., 2007 Sing and Markeset, 2009).

6.3.3.1 What type of damage to look for

Damage types are the damage physical characteristics that can be detected by an inspection technique and the damage mechanisms are the mechanical actions that produce such damage. According to these concepts, damages can be classified into the next broad types, according to the petrochemical industry:

- Thinning
- Metallurgical changes
- Surface connected cracking
- Dimensional changes
- Subsurface cracking
- Blistering
- Micro fissuring/microvoid formation
- Material properties changes
- Positive Material Identification (PMI).

The existing API Inspection Standards (API 510-Pressure Vessel Inspection Code; API 570-Piping Inspection Code, and API 653-Tank Inspection, Repair, Alteration and Reconstruction) represent the body of accepted inspection practices for equipment with boundary-pressure. The RBI procedure profits from these API Standards and other industry practices to identify areas with potential problems and quantify the relative severity of such problems.

Damage may occur uniformly throughout a piece of equipment or it may occur locally depending on the mechanism at work. Damage occurring uniformly can be inspected and evaluated at any convenient location since the results can be expected to be representative of the overall condition. In contrast, damage occurring locally requires a more focused inspection effort. This may involve inspection of a larger area to ensure that the localized damage is detected. If the damage mechanism is sufficiently well understood to allow the prediction of the locations where damage will occur, the inspection can be focused on such areas.

6.3.3.2 How to look for damage (Inspection technique)

Inspection techniques are selected based on their ability to find the damage type; however, the mechanism that caused the damage can affect the inspection technique selection. Some inspection techniques are next listed:

- Visual examination
- Ultrasonic straight beam
- Ultrasonic shear wave
- Fluorescent magnetic particle
- Dye penetrant
- Acoustic emission
- Flux leakage
- Radiography
- Dimensional measurements
- Metallography

For almost all the damages, more than one inspection technique can be used, each one enhancing the effectiveness of the other. For example, ultrasonic thickness measurements are much more effective for locating internal corrosion if they are combined with an internal visual inspection. The following five factors have to be evaluated when quantifying the effectiveness of an inspection plan for a piece of equipment:

1. Damage density and variability
2. Inspection sample validity
3. Sample size
4. Detection capability of the inspection methods
5. Validity of future predictions based on past observations

The inspection techniques are also categorized according to their ability to detect and quantify the anticipated progressive damage according to five scales of effectiveness. Thus, important calculations as part of the inspection program are the damage rate (according to the type of damage detected), the remaining life of the equipment and finally the next inspection date, which are part of the fitness for service step.

6.3.3.3 When to look for the damage (How often look for it)

The inspection frequency is selected as some fraction of the equipment's remaining life. The remaining life is defined as:

$$RemainingLife_j(years) = \frac{DamageTolerance_j(units)}{DamageRate_j(units/year)} \qquad Eq\ 6.6$$

Equipment life is seen as an increasing probability of failure over time by considering uncertainties in the damage growth rate, the tolerance of the equipment to damage, the processed material and the operating conditions. The real probability of failure can be estimated based on whether the damage is progressing at a high or low rate. Decisions regarding an equipment to continue in service for a while or be replaced can be made based on this new information.

6.3.4 Fitness for service

Fitness for service (FFS) procedures can be used to set the inspection intervals for some damages mechanisms with limited guidance or information for setting their inspection frequencies. For instance in petrochemical industry, cracking or change of material properties are damage mechanisms with limited guidance in the API standards for setting their inspection frequencies. However, in these cases, API inspection standards have established some rules for setting minimum inspection frequencies in situations where the damage mechanism is loss of material. In the same way, long intervals are allowed if the material is non-corrosive.

In the FFS procedure, a conservative estimate of the deterioration rate is calculated. The actual rate of deterioration is a function of a complex interaction between material properties, process environment, operating conditions and state of stress. The amount of damage that the component can withstand is then calculated and the next inspection is scheduled well before the anticipated failure. In each future inspection, the actual deterioration rate becomes better defined and the inspection frequencies can be adjusted accordingly.

6.3.5 The interaction between RBI and other safety initiatives

The RBI methodology has been designed to interact with other safety initiatives whenever possible. The output from several of these initiatives provides an input to RBI methodology. In some instances, the RBI-based risk rankings can be used to improve other safety systems. Some examples are Process Hazard Analysis (PHA), Process Safety Management (PSM), Equipment Reliability and Traditional quantitative risk assessment.

Hazards identified in PHA can be specifically addressed in RBI analysis. Moreover, a strong PSM system, as described in API RP 750, can significantly reduce the risk in a process plant. Several of the features of a good PSM program provide an input to a RBI

study. In return, the RBI procedures can improve the PSM program. An effective PSM program must include a well-structured equipment inspection program.

Due to RBI has parentage from traditional Risk Analysis, Quantitative Risk Analysis (QRA) shares many of the data requirements with RBI. If QRA has been prepared for a process unit, the RBI program can borrow extensively information from this effort. Finally, information common to both QRA and RBI is next listed:

- Generic data
- Population information
- Ignition sources
- Meteorological data
- Dispersion distances
- Conditional probabilities for the fate of a vapor cloud.

6.4 Conclusions about risk-based maintenance

There is no unique way to perform Risk Analysis and Risk-Based Maintenance. Two main maintenance management techniques of recent generation have been exposed in this chapter. Both of them are based on risk: Risk-Based Maintenance (RBM) and Risk-Based Inspection (RBI). The application of these methodologies highly depends on the depth of the analysis, area of application and results quality. Also the experience of the analyst with one or other technique is an important factor to consider.

RBI is well presented as a method for reducing risk in the in-service equipment as well as developing the prevention and maintenance tasks because it is widely applied in the petrochemical industry. RBI technique also develops inspection programs in order to detect the damage type in plant control units and the damage mechanism. Thus, it establishes the appropriate inspection technique for detecting such damages and in this way, prevents serious consequences that could have economic, environmental and human impacts. RBI procedure is based on API 581 standard.

A RBI program is concerned with the detection of progressive damage and consists of these steps: Plant database, Risk based prioritization, Inspection planning, Inspection results, Fitness for service, Inspection updating and System Audit.

Quantitative and qualitative RBI can be used to prioritize the risks to analyze, inspect and control. Quantitative RBI procedure has some similarities to the Traditional Risk Analysis because both of them calculate the risk in terms of the probability and consequence of the risk, however RBI procedure is applied to operating plants and to the areas and equipments identified as potential concern. Such areas are identified with the help of some API standards and industry practices such as: API 510 (Pressure Vessel Inspection Code), API 570 (Piping Inspection Code) and API 653 (Tank Inspection, Repair, Alteration and Reconstruction). The calculation of the frequency of the scenarios (considered as rupture hole sizes) is different from a Traditional Risk Analysis as well as the

methodology to calculate it. Furthermore, risk prioritization can be also executed by following the previous standards and procedures.

A quantitative risk analysis can also be applied as part of the inspection planning when the purpose is to reduce risk through inspection and optimize the inspection program that is being used.

HAZOP is one of the more used methodologies that can be applied in the design stage of a plant to identify possible risks. The results from such study can be exploited and used to prioritize risks or the areas of potential concern besides the API standards. However, HAZOP can not replace a RBI procedure.

As part of the inspection planning, the main damage mechanisms and types of damage must be identified in order to select the appropriate inspection techniques to apply. Such identification requires of information such as: design and construction data, process data and equipments information. Finally, a brief application of the RBI procedure in PEMEX Refinement is addressed in Appendix 1 of this thesis.

PART III
FAULT DIAGNOSIS FRAMEWORK

Chapter 7
FAULT DIAGNOSIS SYSTEM

Once the methods used in this thesis for accomplishing the specific tasks involved in fault diagnosis have been provided, the third part and the current chapter formulate the general system proposed for Fault Diagnosis based on historical data models, which is the main purpose of the thesis. Some specific approaches of the general system for continuous and batch processes are also proposed and introduced as they were addressed along the research work.

The general methodology of the data-based Fault Diagnosis is divided in three main stages:

1. Construction of diagnosis models
2. Off-line models validation
3. On-line diagnosis of the process by applying the models on current process data.

Figure 7.1 presents the general Fault Diagnosis System (FDS) including the first two stages. Data acquisition, data representation and data standardization were addressed in chapter three, feature processing in chapter four and the diagnosis methods in chapter five. The techniques and algorithms used specifically in this thesis were presented in those previous chapters in order to deal with the corresponding task of the whole methodology.

The general FDS can be summarized as follows: Process data from the different scenarios occurred along the plant history is acquired from the plant records. In the case of lack of records about some faulty scenarios, simulations of the process can contribute with data. Simulations runs are the main source of process data in academic works. Either the way of obtaining process data, a decision has to be made before data acquisition, which is referred to the sampling time to be set in order to acquire the data that will be used for the construction of diagnosis models as commented in chapter three.

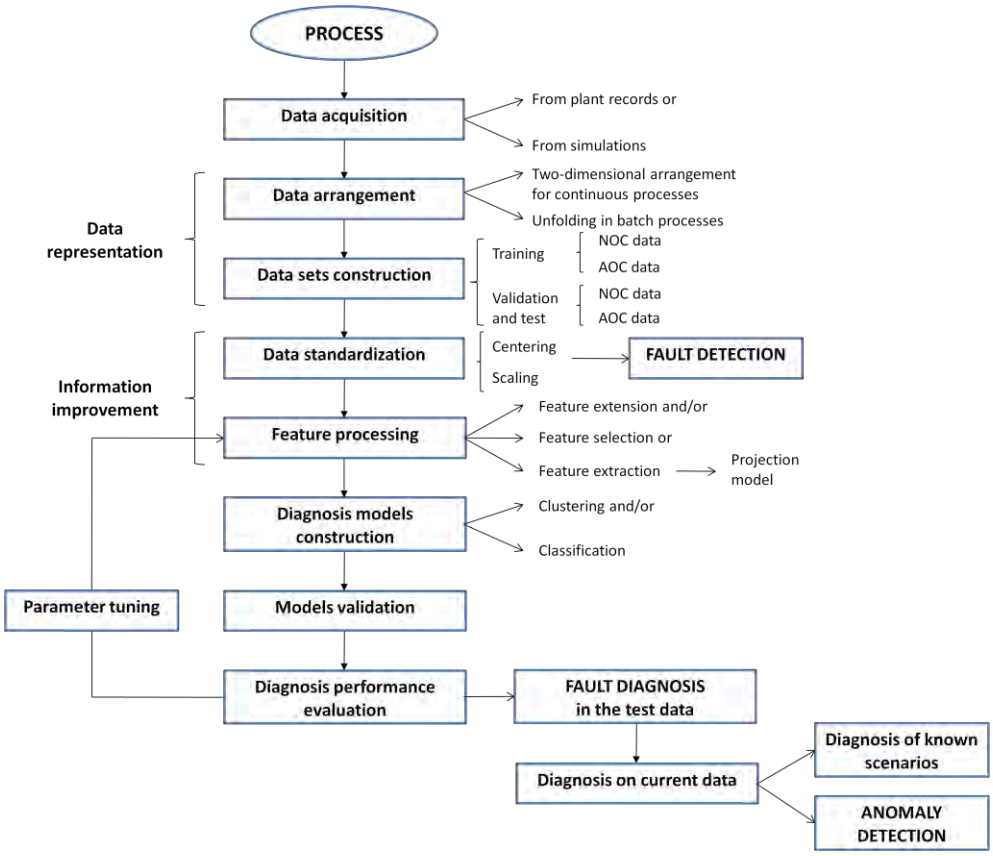

Figure 7.1 General system of data-based Fault Diagnosis

Process data is arranged in two dimensions as in a typical matrix, where one dimension represents the process observations at each time interval and the other one represents the process variables. In case of batch processes, there is another dimension represented by the number of historical batches. Thus, an unfolding step must be developed in order to have a two-dimensional matrix as in continuous processes.

A second data arrangement is required, which consists of dividing the process data in the three sets: training ($\mathfrak{TR}$), validation ($\mathfrak{V}$) and test ($\mathfrak{TS}$). The training data is used practically in the rest of steps for constructing the diagnosis models, while the test data is used to assess the constructed models. It is convenient to use a validation set for parameter tuning purposes and for evaluating and comparing the diagnosis performance. Each one of these data sets includes NOC and AOC data. In the AOC set, few or many faults can be included depending of the records.

A fault detection step is normally required before fault diagnosis. The data representation step is the same for both purposes. Sometimes less data is used for constructing diagnosis models than for detection. The reason is the computational restriction of the diagnosis algorithms (classification) when computing data.

Regarding data standardization, a typical procedure is centering and scaling, as explained in chapter three. Normally, the calculation of the mean and the standard deviation is performed over each column of the training matrix with NOC data (TR_0). After that, the columns of the TR_f, V_f and TT_f matrices (including NOC and AOC data) are subtracted by the corresponding mean and divided by the corresponding standard deviation. In this way, mean-centering and auto-scaling are applied.

When the aim is fault detection, a projection model is constructed based on the standardized NOC training data (TR_0^*). Typically PCA is applied for fault detection. Standardized training, validation and test data (TR_f^*, V_f^*, TT_f^*) are projected onto the model and some statistical indices are calculated. When applying PCA, the Q statistic and the Hotelling's T^2 are the statistical indices that allow determining whether a process is under NOC or not (a fault is occurring). In the validation part of this thesis, a extended application of fault detection is addressed.

Regarding fault diagnosis, data standardization is followed by a feature processing step. As stated in chapter four, feature extension, feature selection and/or feature extraction can be applied. Whether a feature extraction method is applied, a projection model is constructed using the standardized training data under NOC (TR_0^*) and both the NOC and AOC training matrices (TR_f^*) are projected onto such model in order to obtain the projections (T scores when using PCA), which are the real input of the diagnosis algorithm. It is typical to represent the scores with the T symbol when applying PCA techniques; however, the outputs from the feature processing step applying any projection technique will be represented with a second asterisk (TR_f^{**} for instance). In batch process, it results better in terms of performance to use both NOC and AOC training data for centering and scaling and thus, calculate the mean and standard deviation of each column of this unfolded set. This is addressed in the validation part.

The final step of this first stage is the application of diagnosis algorithms by using either a clustering or a classification algorithm. The algorithms used to develop this step were introduced and explained in chapter five. Using clustering algorithms is an important contribution to the area since the state of the art remarks that classification methods have been practically the only methods applied to data-based Fault Diagnosis.

The second stage is to validate the diagnosis models on validation data. NOC and AOC validation subsets (V_f) are standardized with the values calculated based on the NOC training data (NOC+AOC training data in batch processes). The resulting standardized subsets (V_f^*) are then projected onto the projection model, whose projections (V_f^{**}) are used for validating the diagnosis models. The results from this validation are the diagnosis matrices (D_f). In the case of SVM, positive predictions in D_f indicate the occurrence of the corresponding fault while negative predictions indicate a process under NOC. In the case of ANN, values in D_f over 0.5 would indicate a change in the process operation that would probably end up in a fault.

Once D_f are obtained, their comparison with the occurrence matrix values (H_f) allow calculating the diagnosis performance ($F1_f$). Such performance can be improved by adjusting some parameters of the feature processing and/or diagnosis method. After achieving a high diagnosis performance with the validation data, the diagnosis models are

applied on test data (in fact the projections TT_f^{**}) and a similar performance should be obtained, indicating a good generalization of the techniques.

The third stage consists of the on-line diagnosis by applying the models to on-line current data from the process, also represented as *TT*. The same preliminary steps as in the second stage must be followed with the new data before applying directly the diagnosis models. This means to center and scale each measured observation and calculate its projection (TT^{**}). Assuming that PCA has been used as feature extraction method, Figure 7.2 integrates the first and third stage of the general FDS.

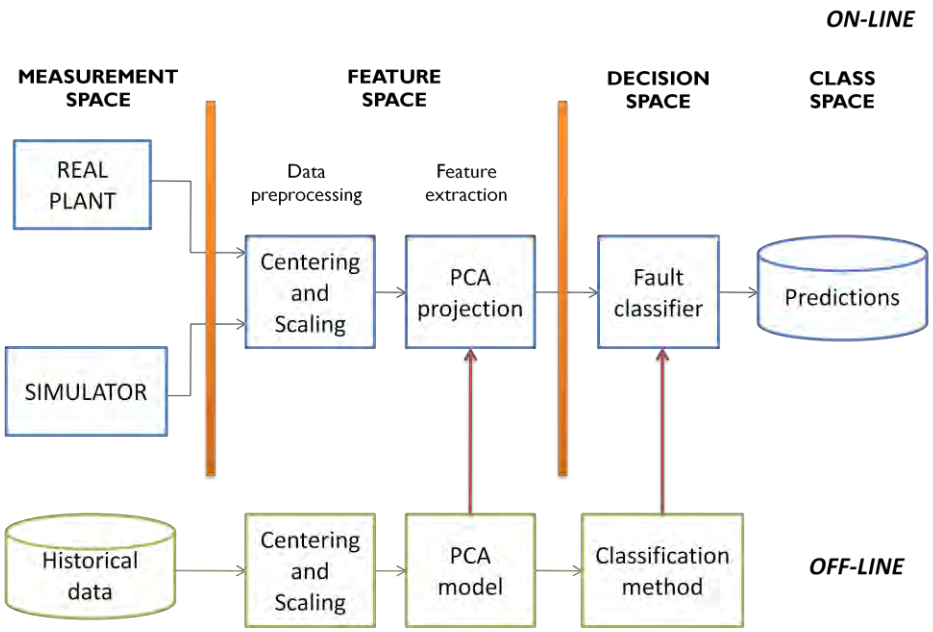

Figure 7.2 Integration of the first and third stage of the general FDS

There could be different approaches to FD by considering different steps, omitting or replacing others of the general system. However, the main line to follow when dealing with this issue has been presented with the proposed system.

Along the thesis work and on the way to develop the general system, some different methods were used, but more important several methodologies were brought up by considering data from different process regimes to construct the diagnosis models. The next two chapters address such specific approaches for continuous and batch processes.

A remaining issue is diagnosing faults that were not learned by the diagnosis algorithm because of the lack of data. This situation can be expected due to errors in the preventive maintenance as a general explanation, for example an error in the maintenance time period, a human error with the inspection techniques, an important deviation not prioritized or an equipment skipped during the preventive maintenance task.

Diagnosing “new” or novel faults with data-models that do not include information about them is called Anomaly Detection and is addressed in the last section of this chapter.

7.1 Fault diagnosis approaches

The approaches that have been addressed along this work are summarized in Figure 7.3. Some of the approaches are an improvement from the others as it is the case of the steady-state data approach, which uses exactly data in steady state, rather than the random data approach, which uses random data to construct the diagnosis models. Regarding the transient data approach, it was addressed as an improvement to the steady-state and latent approaches by using data during the transient stages of the faults.

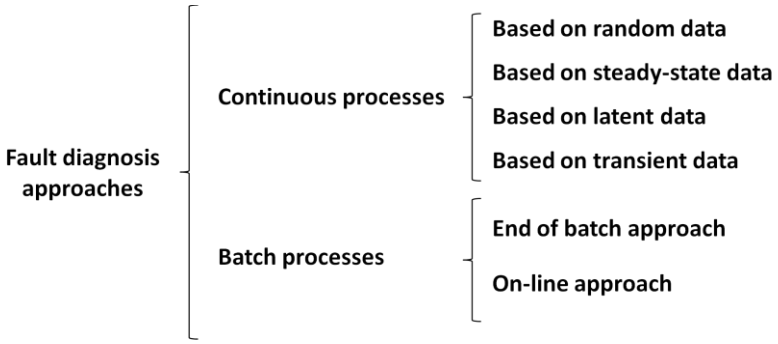

Figure 7.3 Fault diagnosis approaches

Certainly, this last approach is the best in continuous processes when diagnosing data under transient stages but not when diagnosing the developed faults in steady-state. Thus, it resulted that two models are required when diagnosing faults in continuous processes. This will be explained and justified in the validation part (IV).

Regarding batch processes, there are two main fault diagnosis approaches: by detecting and diagnosing faults at the end of the batch and the on-line approach. The first approach is widely used in pharmaceutical industries in order to inspect the quality of the batch in an easy, cost-effective and rapid manner instead of diagnosing each sample taken during the batch. This first approach requires to arrange the historical batches by means of the batch-wise unfolding in order to construct the diagnosis models.

The second approach has been recently addressed and can be faced in two ways depending of the type of unfolding: batch-wise or variable-wise. As explained in section 3.2.2, batch-wise unfolding implies to fill in the batch with all the rest of observations according to the three stated methods. On the other hand, variable-wise unfolding is the most appropriate data arrangement way for diagnosing on-line batches.

Chapter eight formulates and validates the fault diagnosis approaches for continuous processes on the TEP benchmark while chapter nine formulate and validates the two approaches regarding batch processes on the BPP as case study.

7.2 Anomaly detection (AD)

Anomaly detection (AD), also known as novelty detection, outlier detection, data description and concept learning is a classification problem of growing interest in Statistics and Machine Learning (Cabrera et al., 2008; Oliveira et al., 2008). The analogous problem in Chemical Engineering is detecting operation faults which have not been reported, thus not included in the diagnosis models.

AD had been addressed in chemical processes using ANN (Arranz et al., 2008) but then, soft margin SVM showed to discriminate normal and abnormal classes using acquired learning information (Rocco and Zio, 2007). SVM learn to perform the classification task through a supervised learning procedure based on classified data.

One-class SVM have been applied for detecting anomalies because of their unlabeled classification capability. This method identifies and classifies outliers among positive observations (one single class) and classifies them as negative, requiring neither existing knowledge nor labelling (Shon and Moon, 2007).

An AD method has been proposed and developed in this thesis work which combines both fault detection and diagnosis in two different stages and therefore uses both sort of classification, binary classification for the detection stage and multi-class classification for the diagnosis stage. Specifically, SVM have been used as the learning algorithm in the AD methodology.

The AD methodology extends the diagnosis capacity of the general FDS, by detecting the occurrence of novel faults. Given the information about f classes (normal and abnormal), the corresponding models for each faulty scenario have been constructed, validated and applied. The AD methodology is divided in two stages, showed in Figure 7.4 and Figure 7.5

In the first stage, in order to construct the detection models, the training set joins the faulty observations in a positive class and the data under NOC in a negative class. In Figure 7.4, the test set is referred to the on-line process observations ($\mathcal{JJ}$ set). Whether the predictions are positive values after applying the detection model, a process abnormal behavior or fault is identified.

DETECTION STAGE					
Training set ($\mathfrak{TR}$)			Test set ($\mathfrak{TS}$)		
Class	*Information*	*Label*	*Class*	*Information*	*Label*
Normal (positive)	X_0 (NOC)	-1	Novel	***TT*** (current process)	+1
Abnormal (negative)	X_1 (fault 1)	+1	Whether PREDICTIONS	+1 FAULT	
	X_2 (fault 2)	+1		-1 NOC PROCESS	
	X_F (fault F)	+1			

Figure 7.4 Binary classification stage in the AD methodology

DIAGNOSIS STAGE			
Training set ($\mathfrak{TR}$)		Test set ($\mathfrak{TS}$)	
Class information	*Label*	*Class information*	*Label*
X_0	-1	***TT*** (current process	-1 as NOC
X_1	+1 as fault 1	Whether PREDICTIONS	-1 NEW FAULT
X_2	+1 as fault 2		+1 DATA BASE FAULT
X_F	+1 as fault F		

Figure 7.5 Multi-class classification step in the AD methodology

The second stage (diagnosis) consists of applying a multi-class classification system as in the proposed general system. The training data to construct the models is the same than in first step but labeled according to the corresponding class; thus, there are as many diagnosis models as classes. Anomaly Detection results from the simultaneous detection of an abnormal situation (stage 1) and the lack of positive diagnosis of all the previously known faults (stage 2). Additionally, the AD methodology should be first validated on data from novel faults, which can be obtained by taking out faults from the diagnosis models and considering them as new. The whole AD approach is applied to the fault detection and diagnosis in a lab heat exchanger and a fully-monitored Photo-Fenton pilot plant, both operating batch-wise, and the results obtained are presented in chapter ten.

7.3 Conclusions of the general fault diagnosis approach

This chapter has addressed the main contribution of the thesis, which is a general fault diagnosis system based on historical data models. The proposed scheme is divided

into three main stages: construction of the diagnosis models, model validation and on-line application.

The diagnosis model construction is the bottleneck of the whole methodology and involves a sequence of steps: data acquisition; data representation (data arrangement and data sets construction), information improvement (data standardization and feature processing by feature extension, feature selection or feature extraction); and diagnosis method application. Several methods can be used for the application of every step. The methods used for each step were formulated in the methods section.

Several approaches were derived from the general FDS for continuous and batch processes. In the case of continuous processes, four approaches have been addressed based on the kind of process data: steady-state, random, latent, and transient. Regarding batch processes, two main approaches have been extracted based on the moment in which diagnosis is performed, either at the end of the batch or at each sampled time (on-line diagnosis).

Latent and transient data approaches in continuous processes are contributed to the fault diagnosis area. Also, few works had been developed before for dealing with on-line diagnosis of batch processes. Strengths and weaknesses of the addressed approaches will come up along the next section, as well as the final conclusions when dealing with the fault diagnosis problem in both kind of processes.

A question that comes up with machine learning methods is how they can classify (diagnose in PSE) data from classes (faults) that were not learned because of the lack of data. In order to deal with this issue, a new methodology for the Anomaly Detection (AD) has been also contributed, which is also considered part of the general FD scheme. The proposed methodology involves two stages: a binary classification (fault detection) and a multi-class classification (fault diagnosis). In this way, the problem of diagnosing non-learned faults is also covered. The application of these methodologies is next presented in the validation section.

PART IV
VALIDATION AND APPLICATIONS

Chapter 8
FAULT DIAGNOSIS ON CONTINUOUS PROCESSES

This chapter focuses on the application of the general fault diagnosis system on continuous processes. The approaches that have been proposed differ between each other in the data selection from the plant records and are next listed:

1. Random data approach
2. Steady-state data approach
3. Transient data approach

The first approach uses feature extension as an information improvement step, in the second approach a squared standardization of the faulty data with respect to the normal data is applied. In the third approach, centering and scaling are applied. Regarding the general feature processing stage, feature selection is used in the first approach, feature extension in the second one, and feature extraction in the third one. In addition, the second approach has been exploited to validate clustering methods as part of the diagnosis models construction and finally, a comparison between ANN and SVM as classification algorithms is presented with the third approach. However, in the three approaches, SVM in a multi-label framework are applied.

Despite the different methods used in each approach, a comparison among diagnosis models is presented by using random data, data from steady-state, latent and transient data and applying centering, scaling and PCA as feature extraction method. The next sections present the listed approaches and their application to the Tennessee Eastman Process (TEP).

Regarding the TEP, fifty-hour simulations runs of the f=21 classes with a sampling time of 1 s were carried out for obtaining the original data matrices X_f composed of K=180001 samples (considering the sample at zero time) and J=52 variables. In the simulations with fault, these are generated at the second hour. The random approach uses the X_f matrices as the original process data; however, the other two approaches use process data, sampled at each minute. Therefore $X_f \in \Re^{3001x52}$ for the last two approaches.

8.1 Random data approach

In this approach, the data used for constructing the diagnosis models are randomly selected from the original data matrices X_f. Feature extension is applied as data pre-processing instead of a data standardization and feature selection as feature processing step, using Genetic Algorithms (GA).

By applying feature selection, the performance optimization problem is defined for each binary classifier *F* and by determining for each fault the set of features ($J_{F,best}$) that maximizes $F1_F$ when diagnosing the validation subsets (V_F). Assuming a good generalization of the learning algorithm, this optimization would also lead to the maximization of the diagnosis performance in the test subsets (TT_F), as well as in new and on-line data.

8.1.1 Methodology

The methodology proposed for solving the problem consists basically of a feature extension step followed by a feature selection before the diagnosis step as shown in Figure 8.1. The learning algorithm used in this approach is SVM. The next subsections explain these steps in detail.

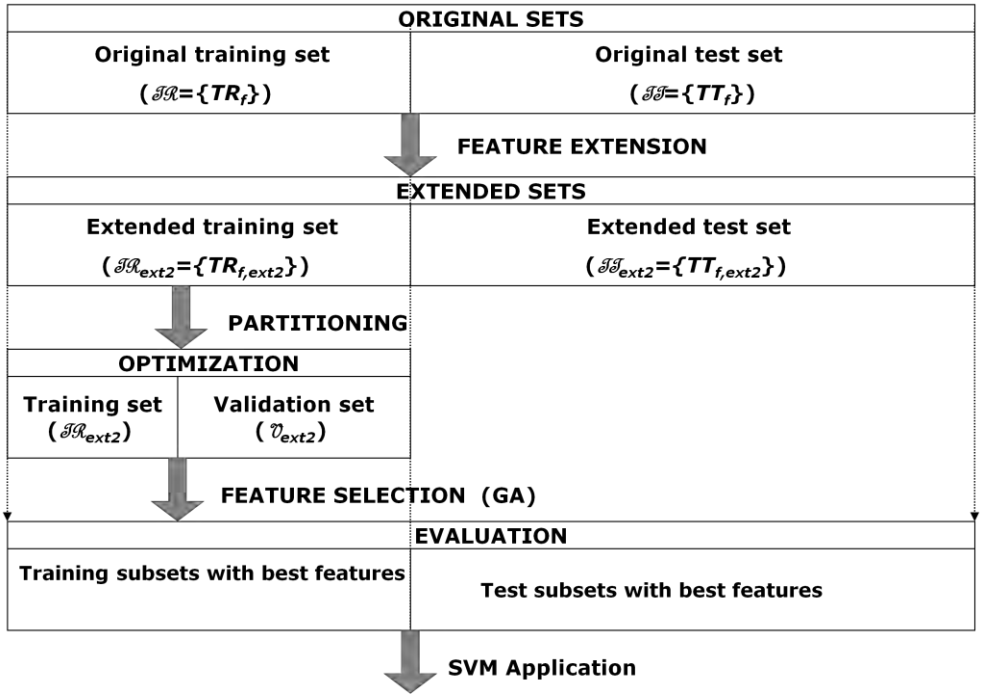

Figure 8.1 Methodology applied for the Fault Diagnosis random data approach

8.1.1.1 Feature extension and partitioning

Extended features are first added to the original process variables by two consecutive extensions (*ext1* and *ext2*). Standard deviations and the linear trend (slopes) are calculated for each given time window and original process variable. The addition of the new data to the training, validation and test sets produces expanded sets ($\mathcal{TR}_{ext1}$, $\mathcal{TR}_{ext2}$, $\mathcal{V}_{ext1}$, $\mathcal{V}_{ext2}$, $\mathcal{TE}_{ext1}$, $\mathcal{TE}_{ext2}$) that might provide valuable information to the diagnosis or learning algorithm. The time window fixed is *l=20* samples. Therefore, for both extensions, the calculation of the standard deviations and the linear trend for each sample *k* is applied to the corresponding sample plus the previous 19 (20 in total).

The partitioning of the extended training subsets produces extended validation subsets ($V_{f,ext2}$) and other subsets that keep the training term ($TR_{f,ext2}$), as shown in the second step of the Figure 8.1. These subsets are then used in the feature processing step for selecting the best features per fault *F*.

8.1.1.2 Feature selection

Feature selection is an optimization problem addressed with GA and it is chosen because of the possible computational cost reduction for the next steps, the reduction of noise and redundant information as well as the possibility to select the best features that govern the different faults *F*.

The inputs of the GA are the extended training and validation data subsets ($TR_{f,ext2}$ and $V_{f,ext2}$) per fault; the initial populations of features ($IP_{N,F}$), where *N* is the number of initial populations for each fault; the number of populations to compute (*NP=25*), the number of iterations (*NIT=50*) between each resulting population; the impact function in each iteration (*IT=cross-over*); and the evaluation function (*EF*), which is obviously the diagnosis algorithm (*SVM* in this approach). The outputs of the GA will be the selected features for each fault ($J_{F,best}$), representing the best chromosomes in the final population, and the performance obtained per *F* in terms of the $F1_F$ score. Figure 8.2 summarizes the general structure of the GA with inputs and outputs.

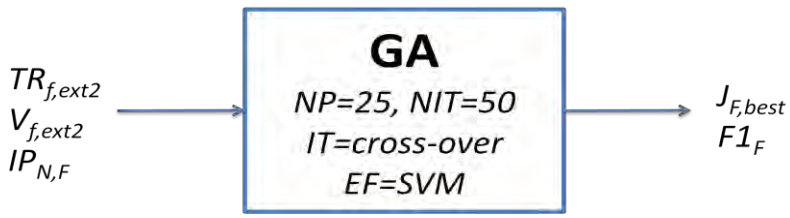

Figure 8.2 Genetic Algorithm general structure

8.1.1.3 Classification and model validation

The diagnosis models are next built based on the training data subsets before partitioning with the selected features per fault (TR_F^{best}) and applying SVM with linear kernel and a default soft margin value (c=0.1). Such models are then validated on the test subsets with the same selected features per fault (TT_F^{best}). Finally, the diagnosis performance is obtained ($F1_F$ and $\overline{F1}$).

Feature selection might produce "overfitting", which could cause the decreasing of the generalization behavior and the diagnosis performance when applying the diagnosis models on the TT_f^{best} subsets. Therefore, validation and test sets are required in order to compare performance.

8.1.2 Results

Training (TR_f) and test (TT_f) data subsets per class are obtained with samples from the original data matrices X_f, randomly selected. Thus $TR_f \in \Re^{372x52}$, $TT_f \in \Re^{1059x52}$, $\mathcal{TR} \in \Re^{7812x52}$ and $\mathcal{TT} \in \Re^{22239x52}$. The subsequent extended training, validation and test sets are obtained after the feature extension and partitioning steps. In this way, $TR_{f,ext1} \in \Re^{279x104}$, $TR_{f,ext2} \in \Re^{279x156}$, $V_{f,ext1} \in \Re^{93x104}$, $V_{f,ext2} \in \Re^{93x156}$, $TT_{f,ext1} \in \Re^{1059x104}$, $TT_{f,ext2} \in \Re^{1059x156}$.

The diagnosis performance in terms of the F1 score is calculated for the data subsets with:

- The original variables ($J=52$)
- The original variables plus their standard deviation values ($J_{ext1}=104$)
- The original attributes plus their standard deviation and their linear trend values ($J_{ext2}=156$)
- The selected features given by the GA application to the training $TR_{F,ext2}$ and validation $V_{F,ext2}$ subsets and
- The best set of features ($J_{F,best}$) found from the previous step.

Table 8.1 shows the diagnosis results for the partitioned training and validation data subsets as well as for the original training and test data subsets. The GA column reports the results obtained from applying such algorithm to the faults with poor diagnosis performance. In general, there are performance improvements when applying feature extension (104 and 156) and therefore with an improved information that considers the dynamics of the process. Table 8.1 also shows that the approach works fairly well for almost all the classes: the F1 score for 17 classes is over 80% and 12 of them is over 95%, which implies a good efficiency of the SVM algorithm but above all, the randomly selected data renders a good diagnosis performance.

In addition, Table 8.1 reports the generalization of the approach by comparing the diagnosis models constructed with the original and partitioned data sets. Similar results are obtained for both cases, thus the diagnosis improvement obtained by feature extension is corroborated. However, there are 3 faults that are poorly identified even after the feature extension steps (3, 9 and 15). In order to increase the performance of these 3

classes, GA were applied for feature selection as a further improvement. The maximum scores obtained with GA for classes 3, 9 and 15 were 11.3%, 10.4% and 9.7% respectively. The selected features were then used for such classes when constructing and validating the diagnosis models as shown in the $J_{F,best}$ columns. For the original data subsets TR_f and TT_f, the diagnosis performance $F1_f$ obtained with the process variables ($J=52$) and with the best features ($J_{F,best}$) is plotted in Figure 8.3.

Table 8.1 F1 score for the partitioned and original data sets used in the Fault Diagnosis random data approach

	F1 SCORE (%)								
Class	TR_f AND V_f DATA SUBSETS					TR_f AND TT_f DATA SUBSETS			
	$J=52$	$J_{ext1}=104$	$J_{ext2}=156$	GA	$J_{F,best}$	$J=52$	$J_{ext1}=104$	$J_{ext2}=156$	$J_{F,best}$
1	99.4	98.9	98.9	-	98.9	97.5	99.4	99.4	99.4
2	96.5	100.0	100.0	-	100.0	93.5	98.8	98.8	98.8
3	0	0	0	11.3	11.3	0	0	0	0
4	86.1	98.3	98.3	-	98.3	90.5	98.5	98.6	98.6
5	0	84.2	87.5	-	87.5	0	91.2	92.2	92.2
6	100.0	100.0	100.0	-	100.0	100.0	100.0	100.0	100.0
7	100.0	99.4	99.4	-	100.0	100.0	99.8	99.8	100.0
8	46.6	98.9	97.7	-	98.9	42.3	95.3	95.3	95.3
9	0	0	0	10.4	10.4	0	0.2	1.2	9.2
10	0	91.9	91.9	-	88.5	17.2	86.9	86.8	87.7
11	0	99.4	98.9	-	99.4	0	98.7	98.7	98.8
12	0	96.5	96.5	-	96.5	0	95.4	95.4	95.4
13	0	86.5	86.5	-	86.5	0	86.5	86.7	86.8
14	0	100.0	100.0	-	100.0	0	99.3	99.4	99.4
15	0	0	0	9.7	9.7	0	0	0	8.7
16	4.4	98.3	98.3	-	98.3	42.1	99.4	99.4	99.4
17	92.6	95.8	95.8	-	95.8	95.3	96.0	96.0	96.0
18	75.2	80.3	80.3	-	80.3	79.6	81.8	81.9	81.9
19	0	95.2	95.2	-	95.2	0	98.6	98.6	98.6
20	75.7	88.1	88.1	-	88.1	86.0	92.5	92.5	92.5
0	26.8	50.3	50.8	-	0	27.2	52.3	52.2	0
Mean	38.8	80.6	80.7		82.2	42.2	80.9	81.0	81.9

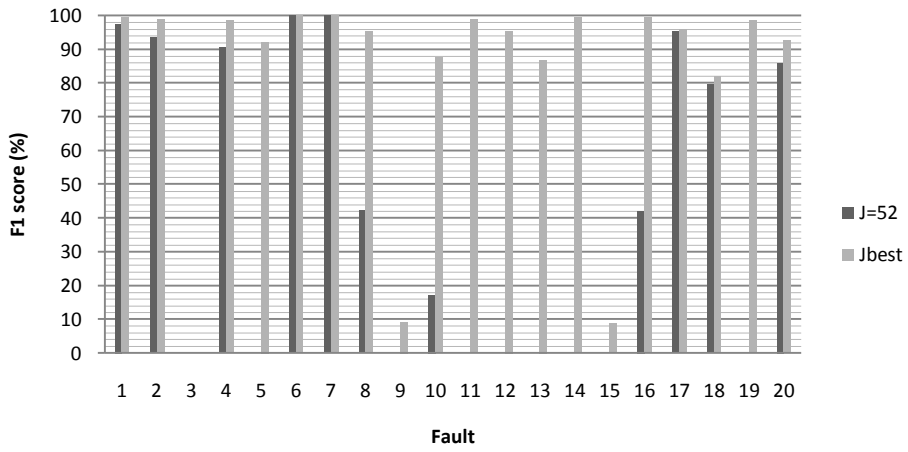

Figure 8.3 Diagnosis performance improvement with the random data approach

On the other hand, Table 8.2 reports the performance obtained after applying GA to each single fault and to the partitioned data subsets, as well as the corresponding initial populations ($IP_{N,F}$). The resulting selected features and the $F1_f$ score from the SVM application to TR_f^{best} and the model validation on TT_f^{best} are also reported. Class zero data has $J_{f,ext2}$ features.

Table 8.2 Initial populations and selected features for each fault of the TEP applying GA. Individual (from GA) and general performance (from SVM) in terms of the F1 score.

Class	$IP_{N,F}$	GA	Selected features	CPU	F1 score
1	IP_1=1-104, IP_2=1-156	100.0	4,8,9,11,13,14,16,17,20,27,28,31, 32,34,37,40,47-94,96-156	78:13:40	99.4
2	IP_1=1-104, IP_2=1-156	100.0	1-104,116	73:45:34	98.8
3	IP_1=2,4,7,10,11,17,30,40, 41,47,49,56,93; IP_2=2,4, 7,11,17,30,40,41,47,56, 92,93; IP_3=2,4,7,33,40, 41,47,49,87	11.3	2,4,7,33,40,41,47,49,87	104:51:26	0
4	IP_1=1-156, IP_2=1-104	98.3	1-156	66:42:42	98.6
5	IP_1=1-156, IP_2=1-104	96.6	2,5-8,13-15,17,18,20-24,26,30,32-35,39, 40,43, 51,52,54-56,60,66,69,71, 75,78,80,82-97,100,102104,106, 107,109,112,114,116,117,120, 122,124,125-127,129,132,137,142, 144-146,149,151,152,154-156	56:30:29	95.2
6	IP_1=1-152, $1P_2$=1-104	100.0	1-52	5:37:49	100
7	IP_1=1-152, IP_2=1-104	100.0	1-52	40:57:31	100
8	IP_1=1-104, IP_2=1-156	98.9	1-104	6:55:40	95.3
9	IP_1=2,4,7,11,17,30,40,41, 47,56,92; IP_2=2,4,7,11, 30,41,47,56,93; IP_3=2,3, 6,7-9,11,15,23,25,28,29, 31,47,49	10.4	3,7,11,35,44,47,56,93	93:20:58	10.4
10	IP_1=1-156, IP_2=53-104, IP_3=1-104	91.9	1-129,147	30:13:23	86.8
11	IP_1=1-156, IP_2=53-104, IP_3=1-104	99.4	13-33,41-104,120	51:14:29	98.7
12	IP_1=1-156, IP_2=53-104, IP_3=1-104	97.1	1-3,5-8,10,11,18-20,26,30-32,35, 38-40,42,45-48,51,5357,59,64,66, 68,71,72,74-76,81,83,87,88,91,93, 95,97,100,102,104,105,115,117, 119-121,123,124,126,130-132, 134-137,140,141,144,146-148, 150-152,154,156	8:46:18	95.7
13	IP_1=1-156, IP_2=1-104	86.5	1-156	8:52:04	86.7
14	IP_1=1-156, IP_2=53-104, IP_3=1-104	100.0	1-24,26-156	52:28:14	99.4
15	IP_1=2,4,7,11,17,30,40,41, 47,49; IP_2=2,4,7,11,40, 41,47,56,93; IP_3=2,4,7, 40,41,43,47,56,93	9.7	2,4,7,19,56,93	8:6:54	8.7
16	IP_1=1-156, IP_2=53-104, IP_3=1-104	98.3	1-104	5:32:58	99.4
17	IP_1=1-156, IP_2=1-104	95.8	1-104	24:25:12	96.0
18	IP_1=1-156, IP_2=1-104	80.3	1-156	37:27:37	81.9
19	IP_1=1-156, IP_2=53-104, IP_3=1-104	95.8	1-104	13:10:30	98.6
20	IP_1=1-156, IP_2=1-104	88.7	1-104	50:14:24	92.5
0	-	-	1-156	-	0
Mean					82.2

8.1.3 Discussions about the random data approach

Although two feature processing steps are applied in this approach, faults 3, 9 and 15 still render low performance. This issue has been experienced and addressed by other authors using different techniques such as Principal Component Analysis (PCA) and Correspondence Analysis (CA) for fault diagnosis (Detroja et al., 2007). Hence, it could be conjectured that there is not enough good and reliable information for the construction of a learning model for these classes.

Another point to highlight is the generalization property. As a particular case, for class 3 the use of GA increases the F1 score to 11.3% on the validation data set. However, when selecting the resulting features in the training and test sets, the F1 score returns to 0%. In this case, the system had over-learned the partitioned training set, which is known as overfitting. Thus, overfitting may occur when applying an optimization procedure (GA) for feature selection before a learning algorithm that includes an implicit optimization (SVM). For other schemes (ANN, kNN, the mono-label approach, etc.) feature selection with GA is reported to perform appropriately (Decadt et al., 2004; Aleixandre et al., 2004). Therefore, overfitting for the ML&SVM framework reveals the good tolerance of the approach to noise and redundant information.

The last issue to discuss is about the drop in the diagnosis performance of the normal class (0) to 0% when selecting the $J_{F,best}$ features per fault. The observations from the normal class are diagnosed as occurring some simultaneous faults, among these 3, 9 and 15. This is actually the main drawback found with the random approach. Moreover, despite the F1 scores for the rest of faults are quite good, similar results are not expected when applying the same approach to another random or any other data set from the same process or from other process. For this reason, a fault diagnosis approach considering steady-state data and therefore, the dynamics of the process by using sequential data is proposed.

8.2 Steady-state data approach

This approach is developed in order to improve the performance of the random approach by combining both supervised and unsupervised learning methods. Within this framework different techniques have been applied, such as Independent Component Analysis (ICA) as feature extraction method, Gaussian Mixture Models (GMM) with Bayesian Information Criterion (BIC) for unsupervised clustering and Support Vector Machines (SVM) with linear kernel and a default soft margin for the classification steps.

Since the supervised learning may fail to fully discriminate some individual faults, clustering algorithms allow the unsupervised grouping of some critical faults having a diagnosis performance below a threshold defined by the user. Besides, an additional classification step would provide practical information about diagnosis in terms of the quantitative confidence on the occurrence of one fault (or some) among the reduced fault set.

8.2.1 Problem formulation

The three operational data subsets (TR_f, V_f and TT_f) are created from the original X_f data matrices. These data subsets, whose rows k have a size smaller than the original X_f and the same number of columns J, are constructed from consecutive samples in such a way that their intersection is null (i.e., there are no samples belonging to two subsets).

Also, a feature extension step is performed to each data matrix X_f by adding a new feature for each process variable in order to include more information regarding the dynamics of the process and find statistical features that better describe the behavior of the process at each observation. The results from this feature processing step are the extended subsets $TR_{f,ext}$, $V_{f,ext}$ and $TT_{f,ext}$ with the same K observations than TR_f, V_f and TT_f, but with J_{ext} features.

In the next step, modified training, validation and test subsets ($TR_{f,ext}^*$, $V_{f,ext}^*$ and $TT_{f,ext}^*$) are obtained from a squared data standardization or second information improvement step, which is an alternative proposed to scaling the process variables and is required to homogenize the variable units. These steps are detailed in the methodology subsection.

The whole $TR_{f,ext}^*$ subsets are gathered into the $\mathcal{TR}_{ext}^*$ set, which is the input of the classification algorithm for constructing the diagnosis models. In the same way, the $V_{f,ext}^*$ and $TT_{f,ext}^*$ subsets are used for validating and assessing the models. As indicated in chapter five, the binary matrix H labels what fault is occurring at each sampled time per fault ($h_{k,f}$) and the binary matrix D indicates the diagnosis at each sampled time ($d_{k,f}$). From these two matrices, the *F1* score is calculated (section 5.3).

8.2.2 Methodology

The steady-state data methodology is further explained in this subsection. This methodology (Figure 8.4) is susceptible to an automated implementation except by one step that requires human supervision and gathers powerful algorithms of clustering and classification.

8.2.2.1 Data representation and information improvement

The second step of the methodology is applied for improving not only the information representation, but also the fault diagnosis performance. First, feature extension is applied. The standard deviations of the process variables are calculated for each sample k given a time window heuristically determined and the results considered as the extended features.

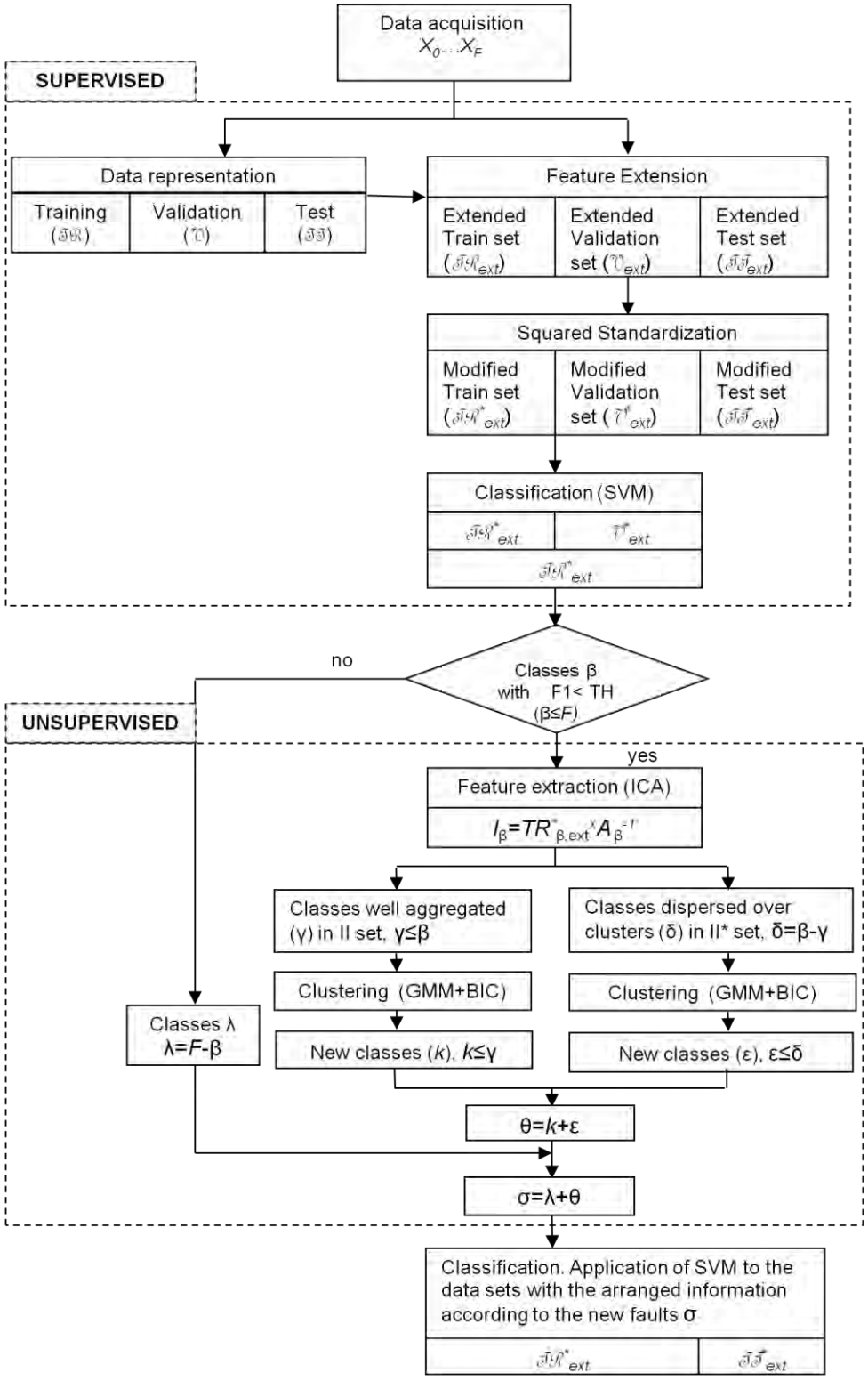

Figure 8.4 Steady-state data fault diagnosis approach

After feature extension, a second information improvement is applied through a data standardization step by using the squared values of the standardized classes with class 0. This calculation is presented in Eq 8.1 and is an alternative to centering and scaling.

$$U^*_{F,ext} = \left(\frac{U_{F,ext} - U_{0,ext}}{U_{0,ext}} \right)^2 \qquad \textit{Eq 8.1}$$

where $U_{F,ext}$=[$TR_{F,ext}$, $V_{F,ext}$, $TT_{F,ext}$] corresponding to the extended subsets and $U^*_{F,ext}$ = [$TR^*_{F,ext}$, $V^*_{F,ext}$, $TT^*_{F,ext}$]. Class 0 data used in the standardized subsets is just the extended data ($U_{0,ext}$), otherwise this would be zero. Therefore, the standardized sets obtained after the second information improvement are:

$$\mathfrak{TR}^*_{ext} = \left[TR_{0,ext}, X^*_{1,ext}, X^*_{2,ext}, \ldots, X^*_{F,ext} \right] \qquad \textit{Eq 8.2}$$

$$\mathfrak{V}^*_{ext} = \left[V_{0,ext}, V^*_{1,ext}, V^*_{2,ext}, \ldots V^*_{F,ext} \right] \qquad \textit{Eq 8.3}$$

$$\mathfrak{TT}^*_{ext} = \left[TT_{0,ext}, TT^*_{1,ext}, TT^*_{2,ext}, \ldots TT^*_{F,ext} \right] \qquad \textit{Eq 8.4}$$

8.2.2.2 Application of the classification algorithm to the standardized training and validation sets

Once the data subsets have been extended and standardized, SVM are applied to the standardized training subsets ($TR^*_{f,ext}$,) and the resulting models validated on $V^*_{f,ext}$ for adjusting parameters. After SVM application, the classes with a diagnosis performance over a certain threshold (TH) are considered as correctly classified and diagnosed. This threshold must consider a trade-off between the number of classes below such limit and a high diagnosis performance of the classes over this threshold.

Therefore, the initial classes *F* are divided into classes β below the TH performance (β≤*F*) and classes λ over TH (*λ*=*F*-β). Thus, classes β will be used for the next steps of the methodology, which are addressed for increasing their performance. Classes λ will be kept and used again in the last step of the approach.

8.2.2.3 Classes visualization and feature extraction

Classes β are selected for the subsequent clustering step. A previous step to clustering is the application of a latent technique as feature extraction method, such as PCA or ICA. These techniques allow reducing data dimensionality and obtaining the *R* principal or independent components of the process variables, considered as features. The number of these features is normally smaller than the number of process variables *J*.

On the other hand, the space representation of the three components allows visualizing the aggregations of the observations from each class in the three dimensional space, which will help to make important decisions in the clustering step. These components are the input of the clustering algorithm.

Independent component analysis (ICA) is used in this approach because it searches for the components that are both statistically independent and non-Gaussian. In addition, ICA has not been applied combined with a clustering algorithm. $TR^{*}_{\beta,ext}$ and $V^{*}_{\beta,ext}$ subsets are gathered and used as input of the ICA algorithm for constructing the projection models. The projections of the three independent components of data from classes β on the three dimensional space offer a visual clustering of such data.

The set *II* contains the three independent components of the classes γ (γ≤β), which are the resulting classes well aggregated in clusters. This set is used as the input of the clustering algorithm.

$$I_{\beta} = TR^{*}_{\beta,ext} \times A^{-1}, \quad \beta < F \qquad \textit{Eq 8.5}$$

$$II = \left[I_{\gamma}, I_{\gamma}\right], \quad \gamma \leq \beta \qquad \textit{Eq 8.6}$$

where β represents the classes with a diagnosis performance below the fixed TH and ϒ the classes well aggregated in clusters.

In contrast, the set II^{*}, containing the three independent components of the classes δ (δ=β-γ) that are dispersed over all the established clusters (γ), will be treated by the same clustering algorithm.

$$II^{*} = \left[I_{\delta}, I_{\delta}\right], \quad \delta = \beta - \gamma \qquad \textit{Eq 8.7}$$

where δ represents the classes dispersed over the formed clusters.

This is the step of the whole methodology where human supervision is required in order to separate the established clusters (γ) from the classes that are not capable of grouping (δ) because such dispersed classes would incorporate all the rest of classes as part of them and a poor diagnosis of the classes γ would be obtained.

8.2.2.4 Clustering in new classes

The input of the clustering algorithm is data from many classes. A clustering algorithm searches for the model containing a number of clusters with the highest likelihood. In this way, the number of new classes is obtained depending of the algorithm's output (clusters).

An improvement to the clustering method is implemented by using the three independent components obtained from ICA as input data. In order to perform this clustering step, the Gaussian Mixture Models (GMM) technique is applied. Within this framework, Bayesian Information Criterion (BIC) is used in order to determine the optimal number of resulting classes. BIC provides a quantitative assessment of the trade-off between the likelihood of the model of fitting the input data and the degrees of freedom of the model. In this way, all data is classified in clusters and therefore, the new set of classes defined.

The data set used as input of the GMM algorithm for establishing new classes k (k≤γ) is the *II*, obtained from the ICA projection and aggregation of the components in γ classes.

$$BIC_k \cong log\, L_k^*(II) - \frac{v_k}{2} log\, K \approx log\, f(II|\mu_k, \rho) \qquad Eq\ 8.8$$

where L_k^* is the maximum likelihood, v_k is the number of independent parameters of the model μ_k, ρ represents previous information, and *k* the new classes obtained.

In the same way, GMM algorithm is applied to the data set II^* containing the independent components of the rest of classes δ not graphically clustered from their projection with ICA. The results from this application are then the new clusters ε (ε≤δ).

$$BIC_\varepsilon \cong log\, L_\varepsilon^*(II^*) - \frac{v_\varepsilon}{2} log\, K \qquad Eq\ 8.9$$

where ε represents the new class(es) obtained as a result of dealing with the scattered classes δ. The new classes θ (θ=*k*+ε), obtained from the ICA&GMM combination are finally considered in the last step of the approach, as well as the classes λ, initially well-diagnosed in the third step.

8.2.2.5 Application of the SVM to the redefined classes

As last step of the methodology, SVM have to be applied again to the λ and θ classes in order to diagnose all the faults presented in the chemical plant. The new faults (σ=λ+θ) will be probably less than the original faults (*F*). $TR_{\sigma,ext}^*$ and $V_{\sigma,ext}^*$ subsets are gathered and considered as the training data set for this last step ($\mathfrak{IR}^*_{ext}$ in the figure 8.4). The new classifiers' models, obtained after applying SVM to this whole training set, are assessed on the $\mathfrak{IT}^*_{ext}$ set. Both sets contain information about the new σ classes.

The diagnosis matrix (*D*), whose number of columns is the same than the number of classes σ, is obtained when performing SVM onto the $\mathfrak{IR}^*_{ext}$ and $\mathfrak{IT}^*_{ext}$ sets. The comparison between *D* and *H* allows calculating the $F1_\sigma$ score and the diagnosis performance should be higher than for all the original faults ($F1_f$), although some of them are integrated in only one fault σ.

8.2.3 Validation and results

The data subsets TR_f, V_f and TT_f are constructed with data in sequential order from the matrices X_f with observations sampled at each minute. The training subsets TR_f are composed of 200 samples (from 2201 to 2400). The validation subsets V_f are composed of 100 samples (from 2101 to 2200) and the test subsets TT_f are composed of 600 samples (from 1501 to 2100). The results obtained from the methodology application are presented in the next subsections.

8.2.3.1 Results obtained from the application of SVM to the improved information

Feature extension is performed calculating the standard deviations of the samples with a time window of twenty and adding them as features to the corresponding data subsets. Thus, $TR_{f,ext}$, $V_{f,ext}$ and $TT_{f,ext}$ subsets have 104 features. The proposed data standardization consists of subtracting to each extended data subset $U_{F,ext}$ the extended data subset of class 0 ($U_{0,ext}$), dividing by the same $U_{0,ext}$ and calculating the square of this result as formulated in equation 8.1, where *U=TR, V* and *TT*. Therefore, the standardized sets are represented as in equations 8.2, 8.3 and 8.4

In order to compare the performance of the fault diagnosis among the original data sets ($\mathcal{TR}$, $\mathcal{V}$, $\mathcal{TT}$) and the improved sets ($\mathcal{TR}_{ext}$, $\mathcal{V}_{ext}$, $\mathcal{TT}_{ext}$, $\mathcal{TR}^{*}_{ext}$, $\mathcal{V}^{*}_{ext}$, $\mathcal{TT}^{*}_{ext}$), the F1 score was obtained for:

1. The original variables (*J*=52) using the $\mathcal{TR}$ and $\mathcal{V}$ sets (set 1-52 in Table 8.3),
2. The original variables plus their standard deviation data (J_{ext}=104) using the $\mathcal{TR}_{ext}$ and $\mathcal{V}_{ext}$ sets (set 1-104 in Table 8.3),
3. The original variables plus their standard deviation and the squared standardization with class 0 using the $\mathcal{TR}^{*}_{ext}$ and $\mathcal{V}^{*}_{ext}$ sets (squared standardization in Table 8.3),
4. The original variables (*J*=52) using the sum of the training and validation sets as training set ($\mathcal{TR}+\mathcal{V}$) and the $\mathcal{TT}$ set (set 1-52 in Table 8.4).
5. The original variables plus their standard deviation data (*J*=104) using the training and validation sets as training set ($\mathcal{TR}_{ext}+\mathcal{V}_{ext}$) and test set $\mathcal{TT}_{ext}$ (set 1-104 in Table 8.4),
6. The original variables plus their standard deviation and the squared standardization with class 0 using the $\mathcal{TR}^{*}_{ext}$ and $\mathcal{TT}^{*}_{ext}$ sets (squared standardization in Table 8.4).

The three tests in Table 8.3 were performed applying SVM to the training and validation sets as part of the methodology for parameter tuning and in order to find the classes with low diagnosis performance. Such classes are selected for the next step: clustering. SVM application to the training and test sets (Table 8.4) is also reported for comparative purposes. The results in Table 8.3 clearly show a better fault diagnosis performance when standardizing the extended information, therefore indicating the improvement obtained with this step.

A diagnosis performance threshold TH=80% is considered in the feature extraction and clustering steps. Faults over such limit are considered as well-diagnosed. This value has been heuristically selected by the trade-off between the reduction of complexity of the classes' models for the next steps and the good diagnosis performance of the classes over this limit.

The diagnosis performance of faults 1, 2, 4, 6, 7, 11, 14, 16 and 19 (then λ=9) is over 80%; therefore, they are considered as well-diagnosed and are not taken into account for the next steps. However, it is interesting to see how faults 4 and 16 are below the

diagnosis threshold when SVM are applied to $TR^*_{f,ext}$ and $TT^*_{f,ext}$ subsets (third column of Tables 8.3 and 8.4). This particular case will be discussed with the last results.

Table 8.3 F1 score obtained after applying SVM to the $\mathfrak{IR}$ and $\mathfrak{V}$ sets with the original variables and the features obtained from the two information improvements

F1 SCORE (%)			
Class	1-52	1-104	Squared standardization
1	100	100	**100**
2	100	100	**100**
3	8.6	5.5	0
4	10.5	13	**90.5**
5	6.6	5.7	0
6	100	100	**100**
7	100	100	**100**
8	12.2	7	21.7
9	11.1	9.6	0
10	10.1	7.8	58.9
11	9.3	18.6	**83.1**
12	6.6	12.1	41.8
13	14.8	14.1	78.3
14	9	5.6	**98.5**
15	8	8.7	0
16	4.8	9.3	**98.5**
17	100	99	70.7
18	19.8	83.9	35.4
19	5	4.1	**81**
20	85.7	88.5	6.5
0	0	0	21.5
Mean	36.1	39.6	58.2

Table 8.4 F1 score obtained after applying SVM to the ($\mathfrak{IR}+\mathfrak{V}$) and $\mathfrak{IJ}$ sets with the original variables and the features obtained from the two information improvements

F1 INDEX (%)			
Class	1-52	1-104	Squared standardization
1	95.6	100	**98.4**
2	100	100	**98.8**
3	7.3	13.2	0
4	8.4	12.6	55.3
5	8	7.9	53.1
6	100	100	**95.6**
7	100	100	**100**
8	10	6	9.5
9	9.5	7.6	16.5
10	9.3	9.3	27.8
11	8.4	14.1	**80.2**
12	9.8	10.1	65.3
13	13.5	10.5	63.1
14	5	6.6	**92.8**
15	6.3	6.7	0
16	10.4	7.7	0
17	99.4	98.1	8.8
18	17.4	50	42.7
19	7.7	8	**83.1**
20	41.2	65.8	6.2
0	18.9	0	29.1
Mean	33.4	36.7	49.9

8.2.3.2 ICA&BIC results

The diagnosis performance of the faults 3, 5, 8, 9, 10, 12, 13, 15, 17, 18 and 20 is low or nil when SVM are applied to the corresponding TR_F and V_F subsets, therefore β=11. No improvement is obtained with either the feature extension or the squared standardization with class 0.

ICA is applied to ($TR^*_{\beta,ext} + V^*_{\beta,ext}$) data of the 11 faults β and three independent components (I_β) are extracted so that these can be represented and visualized in a three dimensional space. The projection plot is showed in Figure 8.5.

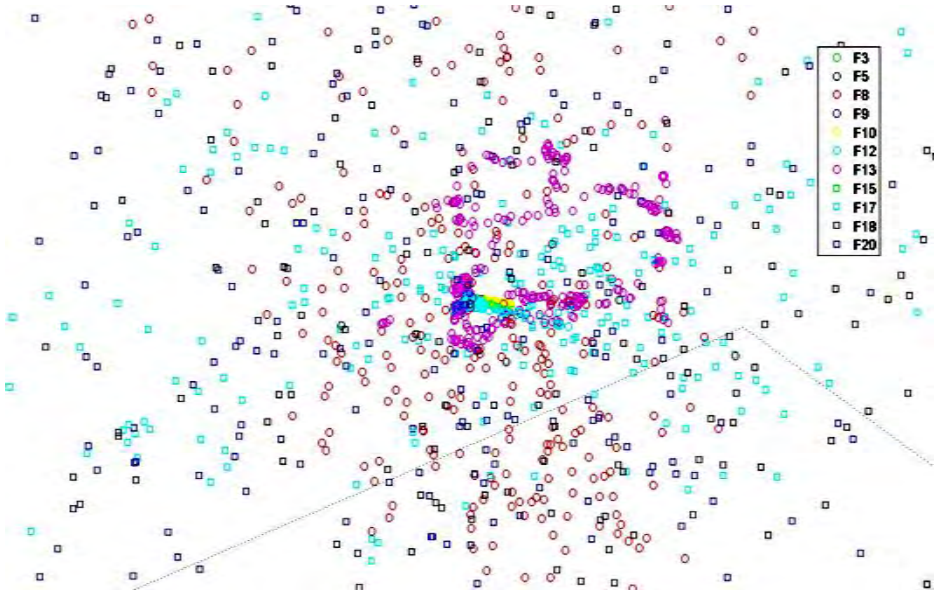

Figure 8.5 ICA projections of the ($TR^{*}_{\beta,ext} + V^{*}_{\beta,ext}$) data for the faults 3, 5, 8, 9, 10, 12, 13, 15, 17, 18 and 20 in the TEP.

Faults 3, 5, 9, 10, 12, 13 and 15 clearly constitute clusters (then γ=7); however, faults 8, 17, 18 and 20 (δ=4) data are scattered over all the formed clusters γ. Figure 8.5 shows how faults 8, 17, 18 and 20 are dispersed over all the space and the clusters of faults 10, 12 and 13 are observed. The clusters of the faults 3, 5, 9 and 15 are behind the fault 12 cluster. Figure 8.6 shows the clusters of these four last faults.

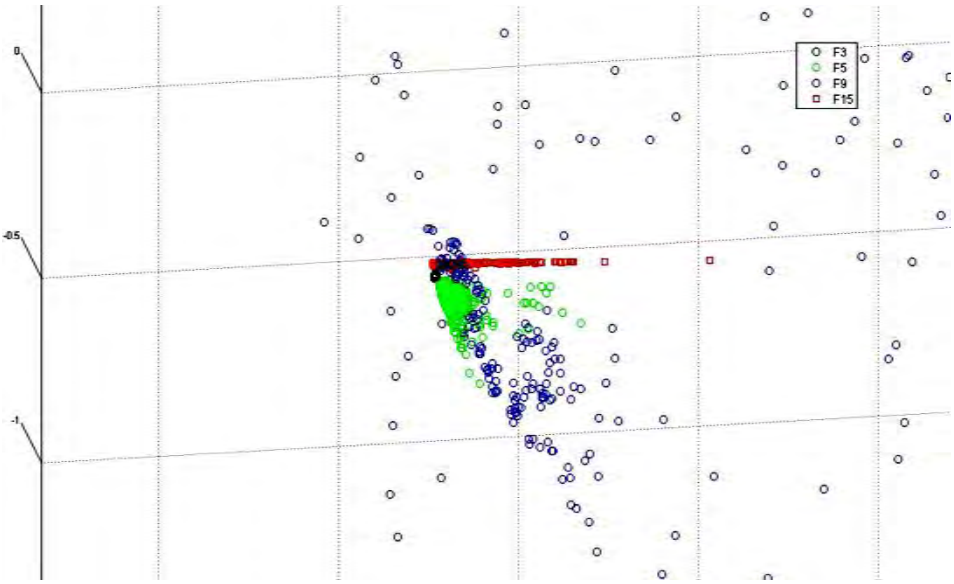

Figure 8.6 ICA projections of the ($TR^{*}_{\beta,ext} + V^{*}_{\beta,ext}$) data for the faults 3, 5, 9 and 15 in the TEP.

Therefore, once three extracted features I_β have been obtained from the application of ICA to data from classes β, GMM-BIC are applied to the three independent components I_γ of the faults 3, 5, 9, 10, 12, 13 and 15 (classes γ), which are capable of aggregating in clusters. This data is joined in the *II* set and is the input of the GMM-BIC algorithm. The results show that faults 3, 5 and 15 are assigned to only one class. This fact can be understood by analyzing the origin of these faults. Indeed, faults 5 and 15 are the same faults but one is due to a step change in the condenser cooling water temperature and the other due to a stuck valve of the condenser cooling water. Faults 9, 10 and 12 are grouped as another class and fault 13 is clustered as single class. Figure 8.7 shows the assignation of the observations from the classes ϒ to three new classes (*k*=3) after GMM&BIC application. In this plot, there are 300 observations per fault (2100) ordered from minor to major.

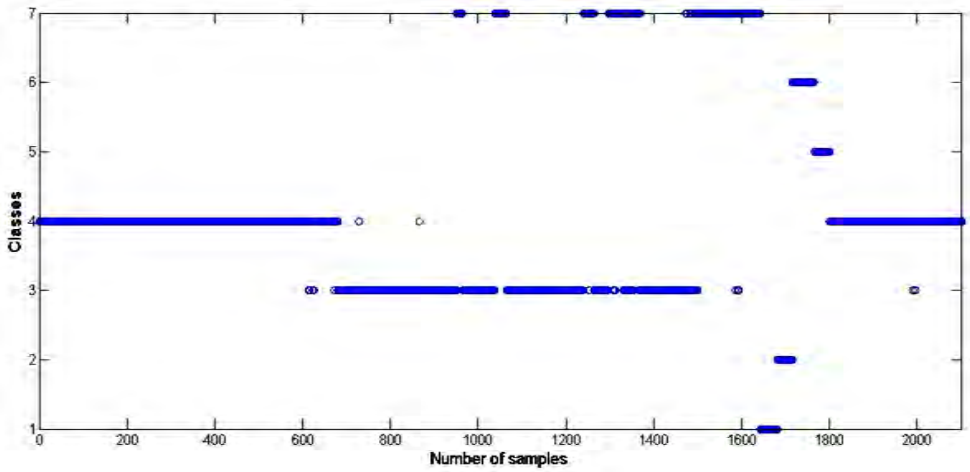

Figure 8.7 Assignation of data from the faults ϒ (3, 5, 9, 10, 12, 13 and 15) to clusters k after applying GMM-BIC

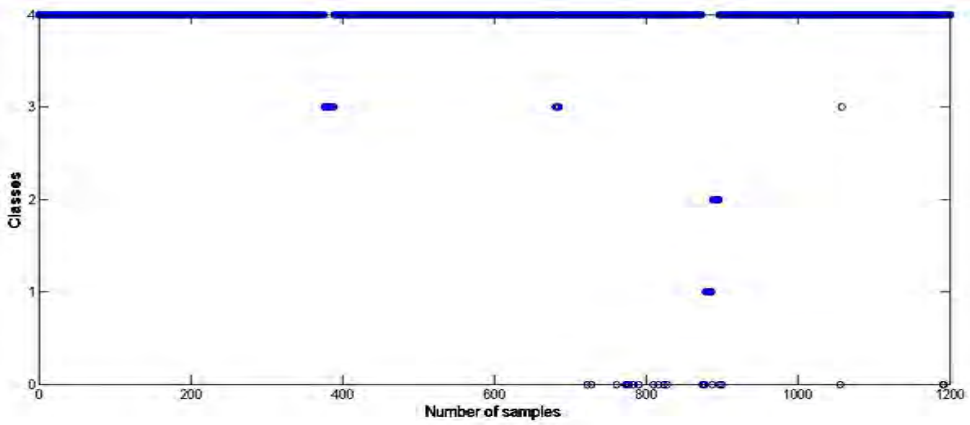

Figure 8.8 Assignation of data from the faults δ (8, 17, 18 and 20) to clusters ε after applying GMM-BIC

Consequently, GMM-BIC are applied to the three independent components I_δ of the faults δ (8, 17, 18 and 20) that did not show any capacity of aggregation. The data of these faults is gathered in the II^* set. All the observations are assigned to only one class (ε=1) by the unsupervised clustering algorithm. It is important to mention that all these δ faults excepting fault 8 (due to a random variation of the components A, B and C composition) are reported as unknown in the original paper. Figure 8.8 shows the assignation of faults δ data to only one fault ε. After applying ICA and BIC algorithms to the eleven faults β that were not diagnosed by SVM, four new clusters or faults (θ=4) were obtained.

8.2.3.3 Results obtained from applying SVM to the improved training and test sets with the new faults σ information

TEP faults have been reduced to 13 faults (σ=λ+θ=9+4) after applying the proposed steady-state data methodology. The assignment of the faults *F* to the new classes σ is showed in Table 8.5

Table 8.5 F1 score obtained after applying SVM to $\mathcal{IR}^*_{ext}$ and $\mathcal{IT}^*_{ext}$ sets with the new faults σ information

TEP FAULT (*F*)	NEW CLASSES (σ)	F1 SCORE
1	1	98.4
2	2	99.4
3, 5, 15	3	70.3
4	4	63.7
6	5	95.6
7	6	100
9, 10, 12	7	43.4
11	8	80.2
13	9	69
14	10	94
16	11	0
19	12	92
8, 17, 18, 20	13	92.1
0	0	50.3

SVM are applied to the standardized training set $\mathcal{IR}^*_{ext} = \{TR^*_{\sigma,ext}\} + \{V^*_{\sigma,ext}\}$) for obtaining the diagnosis models. These models are validated against the standardized test set ($\mathcal{IT}^*_{ext}$) in order to obtain the diagnosis matrix (*D*), which has 13 columns as the number of classes σ. Moreover, the contingency matrix is created from the comparison between matrices *D* and *H* in order to obtain the $F1_\sigma$ diagnosis performance.

The observations of faults *F* grouped in one class are arranged according to the new classes σ for the classification step with SVM. Table 8.5 shows the diagnosis performance obtained for the new faults σ generated after applying the proposed steady-state data fault diagnosis methodology.

With this approach, faults 3 and 15 are diagnosed with 70% performance; meanwhile fault 9 is diagnosed with 43% performance. Regarding fault 15, it can be observed its impossibility to be diagnosed due to its nature (condenser cooling water valve stuck); however it is diagnosed when combined with fault 5 (step change in condenser cooling water inlet temperature) due to their close relation.

Figure 8.9 shows a comparison between the third step of the methodology when SVM are applied for parameter tuning to the $\mathcal{IR}$ and $\mathcal{V}$ sets (original variables and the 2 information improvement steps) with the last step when new faults σ are obtained and SVM are applied to $\mathcal{IR}^*_{ext}$ and $\mathcal{IT}^*_{ext}$ sets (some of the faults σ group the original faults *F*).

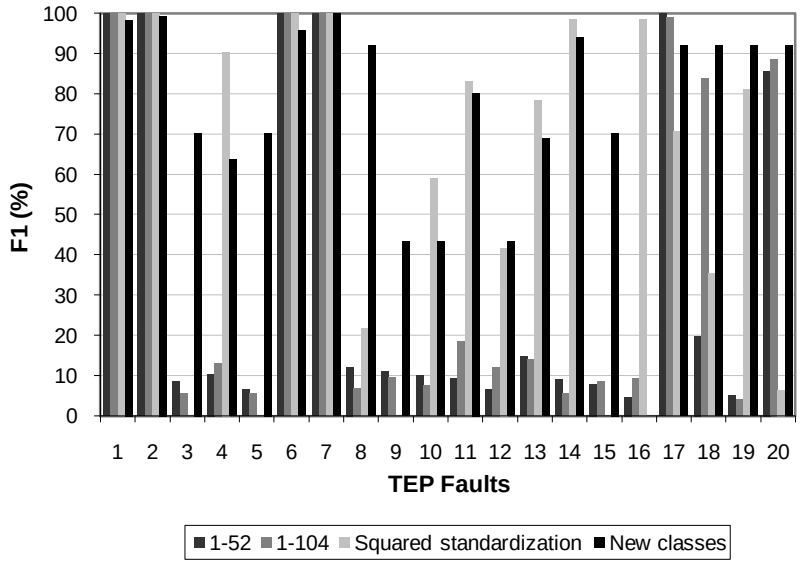

Figure 8.9 Comparison among the diagnosis performance obtained after applying SVM to the $\mathcal{IR}$ and $\mathcal{V}$ sets with the original variables, the two information improvement steps and the data arranged in the new classes σ.

Furthermore, Figure 8.10 offers the contribution in percentage of the four cases compared so that the best step can be distinguished for each fault in terms of $F1_F$. For instance, in case of faults 1 and 2 all the four steps give the same performance, but in case of Faults 3, 5, 8, 9, 15, the contribution of the whole methodology is clearly demonstrated.

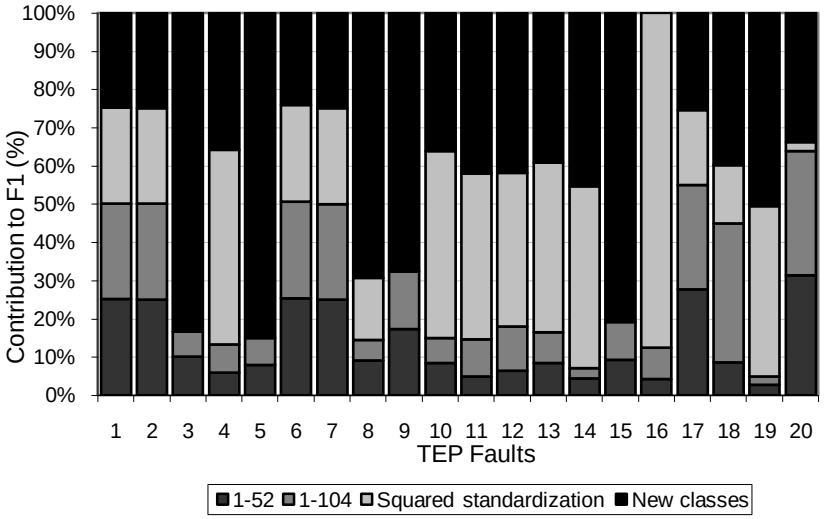

Figure 8.10 Percentage contribution of the four steps of the steady-state data FD approach to the F1 score

8.2.4 Discussions and conclusions about the steady-state data approach

As shown in the last subsection, although fault 16 had been diagnosed with a high performance when applying SVM to the $TR^*_{f,ext}$ and $V^*_{f,ext}$ subsets (98.5%), the performance drops to 0% when applying SVM to the ($TR^*_{f,ext} + V^*_{f,ext}$) and $TT^*_{f,ext}$ subsets. A detailed analysis of the predictions from fault 16 data indicates that this drop in the performance is due to the time distance between the training, validation and test subsets for this particular case. Indeed, the approach assumes that the data describes all the possible dynamical configurations of the process variables during the fault. This basic assumption however, is not fulfilled in fault 16 because its dynamical evolution ends up with a configuration that better resembles faults 3, 5 and 15 (fault σ 3) than its initial configuration. Figure 8.11 shows the behavior of some process variables for the classes 0, 1, 3 and 16 so that the similarity among the classes 0, 3 and 16 can be observed.

This limitation indicates that $\mathcal{TR}$, $\mathcal{V}$ and $\mathcal{TT}$ sets should be defined considering the particular dynamics of the fault under study. This is not taken into account in the steady-state data approach because the main aim was to evaluate the performance of the method when using the same selection of data for each fault and to improve the diagnosis performance from the random data approach. In a real situation, a previous analysis of the dynamics of each fault should be carried out before applying this methodology, yielding to the construction of the data subsets for each fault that would consequently improve the overall performance of the approach.

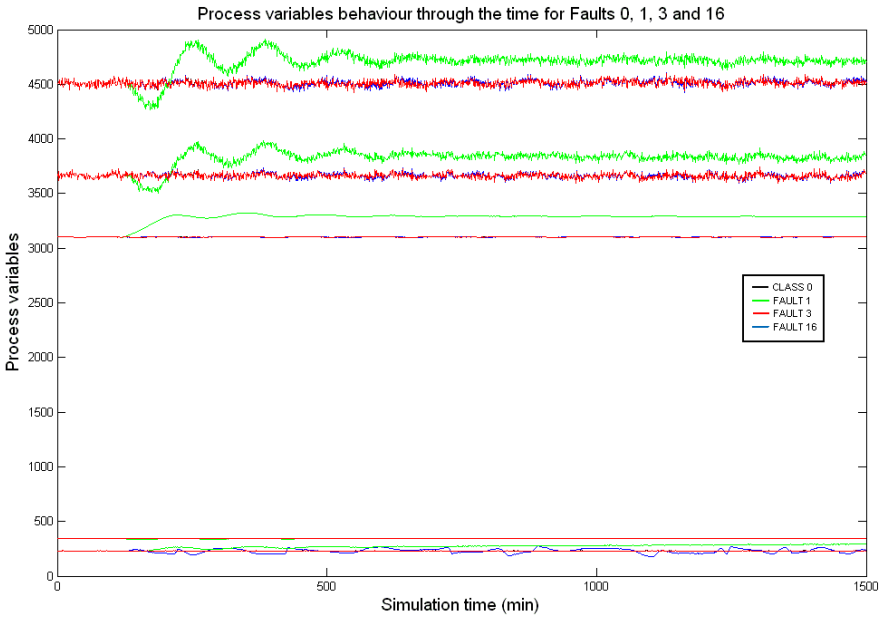

Figure 8.11 Behavior of some process variables of the TEP during a simulation of 25 h when a fault is originated at second hour for the faults 1, 3 and 16 and compared to the process in normal conditions

As summary, the main contribution of the steady-state data fault diagnosis approach is the development of a complete methodology for fault diagnosis combining feature extraction, clustering and classification techniques so that a more accurate diagnosis can be reached by building data models per class (supervised) and per subsets of classes (unsupervised). Therefore, the systematic approach presented in here explores the limits of different individual techniques and provides aggregated information of certain faults, otherwise unnoticed, in new classes, which may strongly benefit the operators' decision-making.

Finally, the last point to discuss is the fact that the diagnosis models are created from steady-state data. However, these models need to be able to diagnose any fault occurring in any process state, such as a transient stage. Therefore, a new FD approach considering the transient stage in a fault development is proposed in this thesis and exposed in the next section. Its capability for diagnosing faults at different stages of their evolution is also analyzed and reported.

8.3 Transient approach

A correct diagnosis of incipient faults reduces operational costs and enhances safety. The FDS to be implemented should be robust to common situations such as false alarms, delayed plant responses and incorrect fault identification. Most of the learning-based fault diagnosis approaches address these problems by using data from either the

steady state reached after fault occurrences (Ruiz et al., 2001; Kulkarni et al., 2005; Bin He et al., 2008, 2009), random observations (Yélamos et al., 2009; Rokach, 2008; Su and Yang, 2008) or by considering data starting immediately after disturbances or in latent stage (Chiang et al., 2000; Brydon et al., 1997).

In general, not much attention has been paid to the data used for producing the classification model, and even less to the importance of working with data starting from the actual detection time of the fault. Often, previous works report relevant results on this field, but only refer indirectly to this point (Yoon and MacGregor, 2001; Vora et al., 1997; Yong et al., 2007; Chiang et al., 2004; Lee et al., 2006; Li and Cui, 2009; Zhang, 2009; Cui et al., 2008; Verron et al., 2008). In addition, many of these reported methods assume that the diagnosis starts once the process has reached a known, well-characterized state. This assumption may result untenable for dynamical processes that undergo transition operations such as plant start-up and shutdowns (Srinivasan and Qian, 2006).

Some approaches to process modeling, alarm management, fault diagnosis and other automation systems may be ineffective during transient stages because they are usually configured assuming a single state of operation. When the plant moves out from that state, these applications may lead to false alarms even when a desired change is occurring. Thus, some frameworks have already been developed for managing transitions and detecting faults (Sundarraman and Srinivasan, 2003; Srinivasan et al., 2005).

In this context, in order for an automatic fault diagnosis system to become a practical support tool for decision-making during plant operation, on-line detection of the early transient stages during a fault evolution is required. More specifically, on-line monitoring of transient states is important to detect abnormal events and to enable the timely recovery of the process. Previous attempts to deal with this problem include the determination of detection delays using different statistical indices such as the Hotelling's T^2 and Q statistic in Principal Component Analysis (PCA) and Correspondence Analysis (CA) (Russell et al., 2000; Detroja et al., 2007; Shao et al., 2009; Seng and Srinivasan, 2009).

Previous sections have presented fault diagnosis approaches using random data and steady-state data as in the cited references. For this reason, the aim of this transient data approach is to improve the performance of the data-driven fault diagnosis system by including data from transient stages in order to model faulty/abnormal plant behaviors. The main assumption is that the incorporation of transient data enhances the overall performance of the classification models used for fault diagnosis.

Figure 8.12 illustrates the different dynamical regimes of a process variable undergoing a fault. Initially, the process is in a steady regime under NOC. At a certain time, a perturbation occurs and the plant changes to an AOC regime. The period between the fault occurrence and the detection of the fault defines a latency period. In Figure 8.12, the fault is detected when the Q-statistic index exceeds its control limit. The process then evolves to a transient regime that will eventually lead to a new steady state. It is worth noting the difficulty of distinguishing the latent, transient and steady stages just from the dynamical evolution of the process variable.

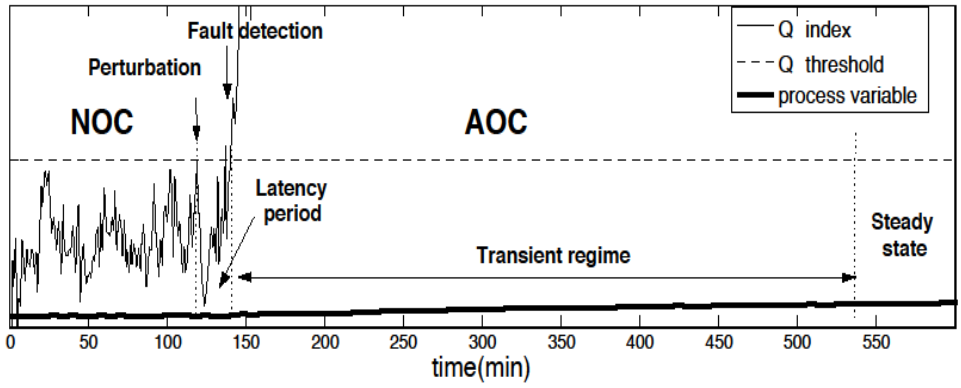

Figure 8.12 Different dynamical regimes of a process variable undergoing a fault

Considering this framework, the approach proposed in this section consists of two main stages: fault detection and transient characterization and diagnosis models construction. First, a PCA model is obtained from simulated steady-state data under NOC. PCA is chosen for the initial monitoring and detection step (Chiang et al., 2000; Yoon and MacGregor, 2001). In order to characterize the transient period, data under AOC is projected onto the PCA model and the principal components and the Hotelling's T^2 and Q-statistic indices are produced. These indices are both used to detect the fault and to estimate the duration and delay of its transient evolution.

A second NOC PCA model is used for dimensionality reduction before constructing the fault diagnosis models with standard classification methods such as ANN or SVM, allowing comparison between both techniques. The proposed methodology implements an off-line learning of the transient stages during the fault evolution. The classification models are then applied to validation data subsets, acquired by simulating the process under NOC and AOC. The diagnosis performance is measured using an off-line procedure that considers the observations from the occurrence of the fault until the end of the simulation. The results obtained from this approach are discussed and compared with the previous approaches and the reported methodologies in literature.

8.3.1 Problem formulation

This approach focuses on transient data corresponding to the development of a fault. Local fluctuations around a NOC regime are not considered as transient dynamics under the assumption that their average stochastic properties are maintained in a steady state by the process control. The approach is also robust to normal transient stages not associated to the development of a fault but related to external changes in the plant configuration such start-up and shutdown regimes or set point changes. An example of the robustness of the approach to the occurrence of normal transients is described in the discussion subsection.

Specifically, a classification algorithm is applied on a training set $\mathfrak{IR}$ that takes into account fault transient data, where $\mathfrak{IR} = \{TR_f^{**}\}, TR_f^{**} \epsilon \Re^{nt \times R}$, where *nt* is the

number of transient observations, *R* the number of reduced features (*R<J)*, and TR_f^{**} represents the real input data to the diagnosis algorithm. The basic assumption is that the use of transient data will improve the classification performance in comparison 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 subsection. Once the training set is obtained, classification models are constructed for each fault as in the previous approaches. These models are then applied to validation and test data in order to detect and diagnose the presence of faults.

For each fault, the corresponding model can be mathematically represented as:

$$M_f = g_f\left(TR_f^{**}, H_f\right) \qquad \textit{Eq 8.10}$$

where TR_f^{**} represents the training data for each class *f***,** and H_f the occurrence matrix that characterizes the scenario *f* occurring at each sample time by means of binary elements h_{kf} indicating whether or not a fault is occurring at sample *k* (Kent et al., 1955; Yélamos et al., 2009). M_f represents the classification model for each scenario *f* and g_f represents the output function from a learning algorithm such as ANN or SVM.

Classification models are applied to validation subsets**.** The binary matrices D_f indicate the result of the diagnosis and are obtained from the application of all the classification models to the validation subsets.

$$D_f = \psi\left(M_f, V_f^{**}\right), \qquad f = 0,1,\ldots,F \qquad \textit{Eq 8.11}$$

Again, the problem to be solved is to maximize the number of matches between matrices D_f and matrices H_f expressed in terms of the *F1* score. However, it is useful to complement this performance index with the false alarm rate or False Positive Rate (FPR), since it considers the incorrectly predicted observations to a fault in terms of the total observations in which such fault does not occur, either diagnosed or not.

$$FPR_f = \frac{b_f}{b_f + d_f} \qquad \textit{Eq 8.12}$$

The present approach proposes a methodology that uses transient data to construct models M_f so that the overall fault diagnosis performance is improved. Next section describes this methodology.

8.3.2 Methodology

A general diagram of the construction of the training set $\mathfrak{IR}^{**}$ is described in Figure 8.13. The methodology takes into account not only process monitoring and fault detection, but also the diagnosis of the faults in the process and it is divided in two main stages: the process monitoring (identification of faulty transient stages plus the construction of training data subsets) and the construction of transient-based classification models. These two steps are next detailed.

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

NOC data is obtained from steady state simulations of the plant. Mean centering and scaling of NOC and AOC data are applied using the mean and standard deviation values of NOC data. Then, PCA is applied in order to detect any abnormality in the process by means of statistical indices.

Specifically, a PCA model (from now on referred as the first PCA model) is constructed on the centered and scaled NOC data subset X_0^*. Centered and scaled AOC data subsets X_F^* are projected onto the first PCA model so that T_{kF}^2 and Q_{kF} indices are obtained. In fact, either X_0^* or X_F^* subsets are projected onto the first PCA model and as far as the values of these statistical indices remain below their control limits (T_{lim}^2 and Q_a) the plant is considered to be under NOC (this is of course the case of the whole X_0^* subset).

When the indices exceed their control limits, the plant is assumed to enter into an abnormal dynamical regime and a fault is detected. Even more, these T_{kF}^2 and Q_{kF} indices allow determining fault detection delays and the right moment in which the fault is developing and fault symptoms are becoming apparent in the generated data. In this way, the starting point of the transient stage (t_F^0, eq. 8.13) is also identified. T_{kf}^2 and Q_{kf} indices (including NOC indices) are gathered respectively in the ***T*** and ***Q*** matrices as shown in Figure 8.13.

$$t_F^0 = \min_k k \left| T_{kF}^2 \rangle T_{lim}^2 \wedge Q_{kF} \rangle Q_\alpha \qquad \forall F \qquad \textit{Eq 8.13}$$

Regarding both monitoring and fault detection indices, *Q* could be even slightly faster than T^2 at detecting the moment in which there is statistical evidence of the fault in the plant data (shorter delay) as it will be explored in the results section. This can be explained due to the fact that AOC data is not included in the first PCA model in terms of monitoring; in addition, *Q* statistic is used to detect new events for which data is not captured well by the existing PCA model (Montgomery, 2005).

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

NOC subset and the transient data subsets under AOC (TR_0 and TR_F) are gathered into the transient set $\mathfrak{TR}$. $TR_f \in \Re^{nt \times J}; f = 0,1,\dots,F$, *nt* is the number of transient observations, and *J* the number of variables. Thus, transient data is given by a transient state window characterized by the corresponding start (t_F^0) and duration (*nt*), which are determined by the indexes T_{kF}^2 and Q_{kF} when exceeding the control limits.

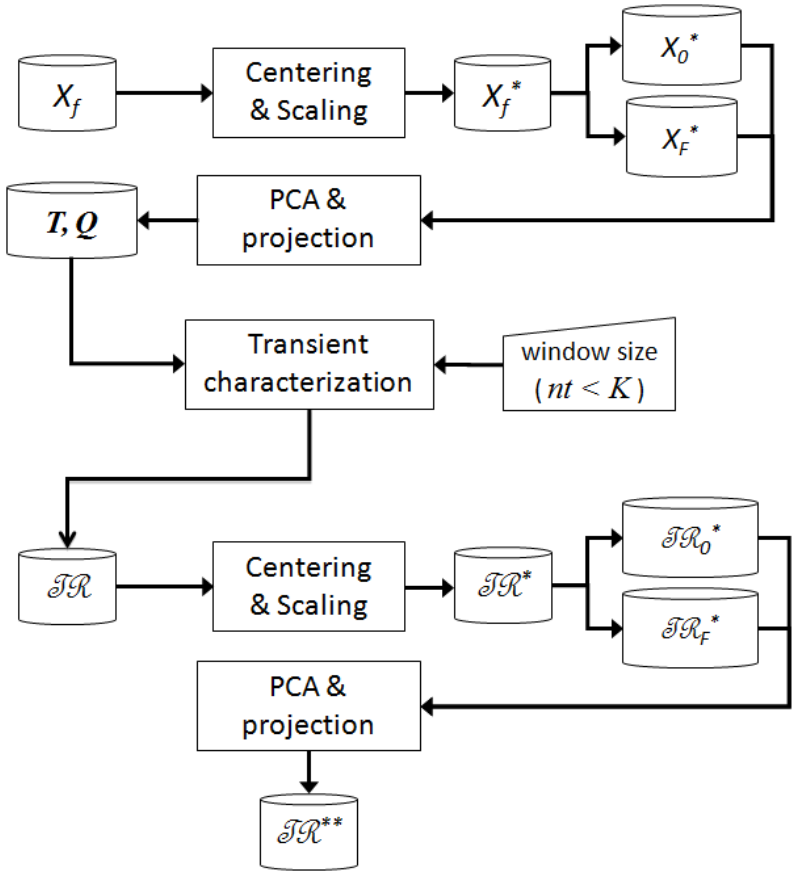

Figure 8.13 On-line fault diagnosis methodology based on transient data models

A second PCA model is constructed again with steady-state NOC data (TR_0), but this time using *nt* observations. As in the previous step, data is mean centered and scaled. PCA loadings (*P*) are obtained from applying PCA to the standardized NOC subset (TR_0^*). The NOC scores matrix TR_0^{**} is obtained when TR_0^* is projected onto the second PCA model ($TR_0^{**} = TR_0^* \times P$). Note that the scores have a different simbology from the typical *T* symbol in PCA, as referred in chapter four.

In the same way, AOC data subsets TR_F are constructed considering the transient window when taking the *nt* number of samples from the historical data matrices X_F and starting with the first observation in which the fault is properly detected by the *Q* statistic (t_F^0). TR_F subsets are also centered and scaled. Thus, TR^*_0 and TR^*_F subsets are gathered in the $\mathfrak{TR}^*$ set as represented in Figure 8.13.

The TR_F^* subsets are then projected onto the second PCA model (*P* loadings) so as to obtain the AOC score matrices TR_F^{**} by means of the same expression used for NOC score matrix ($TR_F^{**} = TR_F^* \times P$). Next, the score matrices for both NOC and AOC cases are joined in the $\mathfrak{TR}^{**}$ set, which is used as input of the classification algorithms. $TR_f^{**} \in \Re^{nt \times R}$, where *R*<*J* represents the number of extracted features or specifically, the retained principal components in the second PCA model.

8.3.2.2 Plant fault modeling

Classification models for each scenario (M_f) are constructed by off-line learning of the faulty transient data once the transient stages have been determined using the PCA indices. This step constitutes the data-driven modeling of the training data.

The diagnosis models are obtained using standard machine-learning algorithms (ANN, SVM) as classification techniques. The input of such algorithms is the training set $\mathfrak{IR}^{**}$. The structure of the ANN and the kernel function in SVM must be fixed. Both techniques are applied in this approach for comparative purposes.

The parameter values to be decided 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 depends on the number of PC to retain; in addition, the number of hidden nodes can be optimized.

On the other hand, SVM are also applied to the training data set $\mathfrak{IR}^{**}$. Then, support vectors per class are obtained for different types of kernel so that the one that produces the best performance is selected. The learning bias in SVM has shown to present good properties in terms of generalization bounds for the classifiers. In fact, Yélamos et al. (2007) reported that classifiers from single faults give good predictions for data that undergo the occurrence of up to four simultaneous faults in the TEP.

Finally, the ANN and SVM models are applied to the validation subsets (V_f). In fact, the real validation subsets are constituted by the score matrices $\left(V_f^{**} = V_f^* \times P\right)$ obtained from the projection of these original data subsets onto the second PCA model under NOC (P loadings). The diagnosis performance is calculated with the F1 score in an off-line way as previously stated. An important decision to make for the prompt on-line fault diagnosis is to choose the number of observations that would be used for evaluating the models.

8.3.3 Validation and results

The first PCA model is constructed with X_0^* data. The classification models are constructed with the observations subsequent to t_F^0 and considering 200 samples per scenario (nt). This nt value is selected because the behavior in the Q statistic for some faults showed that approximately after 200 observations over the control limit (Q_a), there are some faults that show returning to a normal state.

The second PCA model is obtained with the centered and scaled 200 observations under NOC. The corresponding 200 observations per scenario under NOC and AOC ($X_f^* \in \Re^{200\times 52}$) are projected onto this model so that the score matrices are obtained TR_f^{**}. These matrices constitute the $\mathfrak{IR}^{**}$ set composed of 2200 observations, which is the input of the classification algorithms (ANN and SVM).

A comparison between the diagnosis results obtained using transient data models (TRM) and different steady-state data models (SSM) is addressed and discussed in the next

subsections. Such models are applied to validation subsets of 500 observations, corresponding to 500 minutes of faulty state in case of faulty simulations ($\mathcal{V}= \{V_f\}, V_f \in \Re^{500\times52}$).

8.3.3.1 Characterization of the transient stage

The first PCA model is constructed with the X_0 data. Two principal components are retained with the broken-stick rule (26% of variance in the original J variables). The preliminary PCA analysis using the broken-stick rule is conducted to discriminate between normal and abnormal situations, as also reported by other authors (Nomikos and MacGregor, 1995a). This criterion judges whether a principal component adds any structural information to the data variance or just describes noise.

In this specific case, 26% of variance was assumed acceptable considering the total number of variables and that the rest of variability is mainly due to noise and random variation in NOC data. This issue should be considered when detecting faults that violate the PCA models and that are not taken into account in the model subspace (Montgomery, 2005). This also explains why models should be constructed considering observations starting from t_F^0.

The X_f^* matrices are projected onto the first PCA model. Therefore, T^2 and Q statistics are calculated. Figure 8.14 shows the T^2 values for some faults and Figure 8.15 shows the Q values for the same faults including their respective control limits CL (T_{lim}^2=9.2 and Q_α=66.7). In comparison, a PCA model is constructed retaining the 97% of variance and the delays are also calculated with such model.

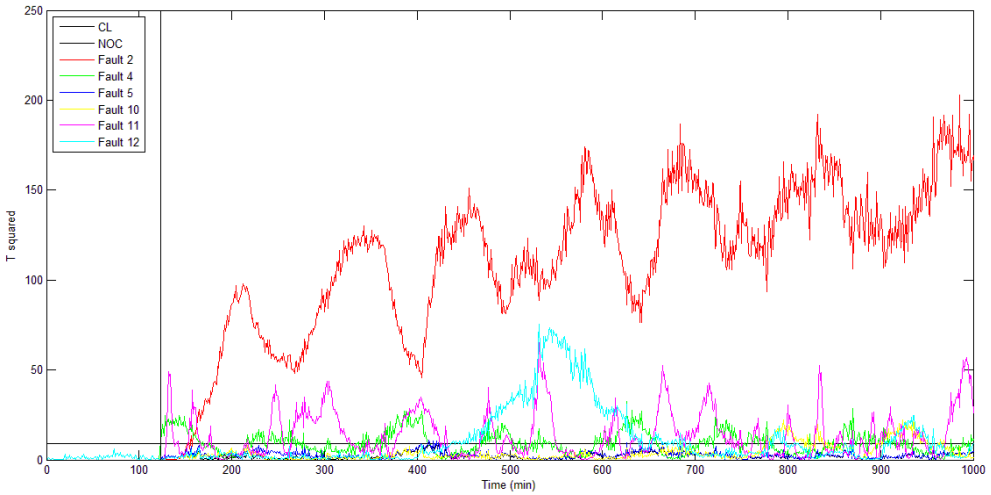

Figure 8.14 Hotelling T² for some faults of the TEP. The vertical line indicates the point where the fault is originated.

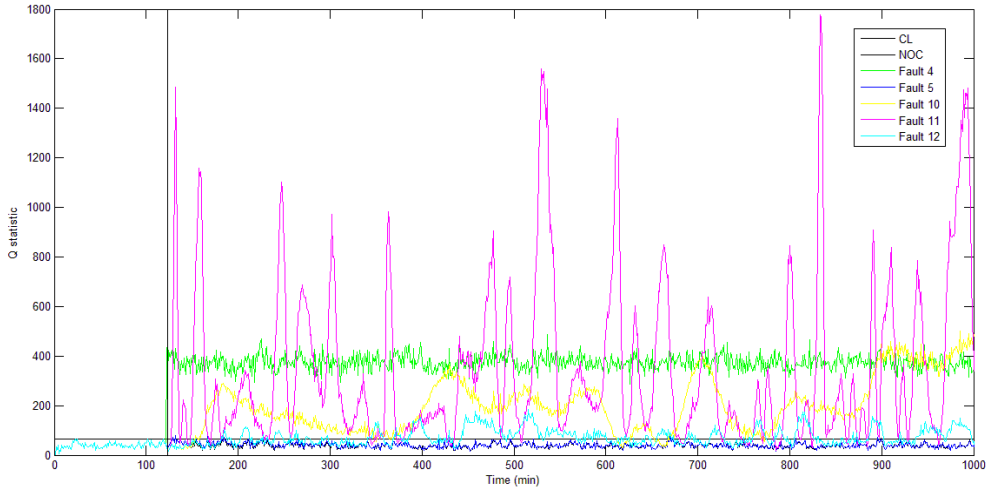

Figure 8.15 Q statistic for some faults of the TEP. The vertical line indicates the point where the fault is originated

Table 8.6 Detection delays with PCA models retaining the variance according to the broken-stick rule (26%) and in a 97%.

FAULT	DELAYS IN MINUTES			
	2 PC's (26% var) T^2	35 PC's (97% var) T^2	2 PC's (26% var) Q	35 PC's (97% var) Q
1	15	5	5	4
2	33	20	16	15
3	283	440	64	2273
4	2	1	1	1
5	292	8	46	2
6	20	1	1	1
7	1	1	1	1
8	30	20	20	19
9	271	440	115	115
10	563	61	33	29
11	9	7	7	7
12	316	322	61	59
13	167	165	104	162
14	283	15	5	14
15	292	90	309	458
16	30	15	13	17
17	83	82	80	70
18	300	319	64	291
19	258	1732	64	1310
20	171	170	106	141

Table 8.6 shows for all the twenty faults the detection delays produced by both statistics and both variance values. The results corroborate first, that for most of the faults *Q* statistic detects such disturbances earlier than T^2, and second, that the delays are smaller with a PCA model retaining the main variance given by the broken-stick rule. Figure 8.16 compares the detection delays (only for the broken-stick rule) between both indices and Figure 8.17 details the delays for some faults (only given by the *Q* statistic). It is worth noting the drop of the *Q* statistic value for fault 11.

The PCA model with 97% of variance shows for some of the faults that once the *Q* value exceeds Q_α, it does not fall down again. This drop tends to occur for the same faults with the less-retained variance model. 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.

Another important aspect to be added is that the method is tolerant to the presence of local fluctuations and noise around NOC. Some isolated observations in the NOC regime, for which the *Q* or the T^2 statistics violate their respective confidence limit, represent local fluctuations. By means of this approach based on a NOC model, transient regimes in the process due to faulty situations can be monitored, and therefore, such faults detected.

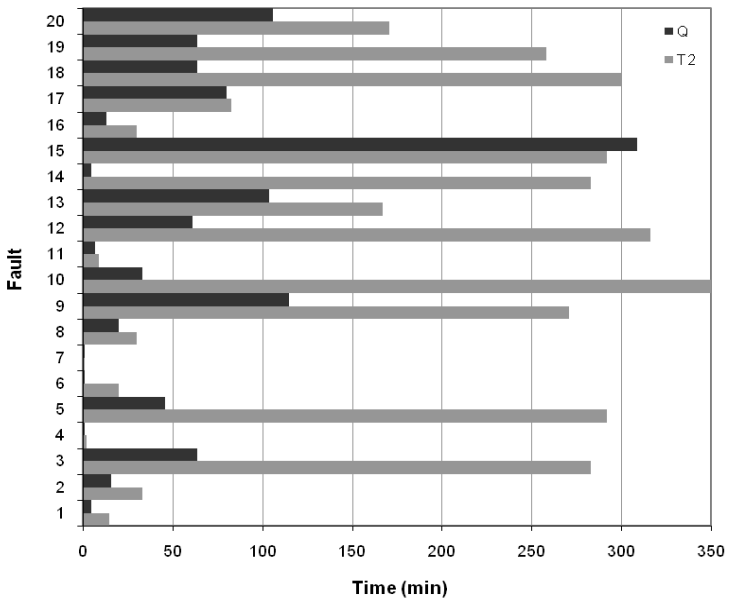

Figure 8.16 Fault delays in minutes using both T^2 and Q statistic indices and a PCA model with the broken-stick rule

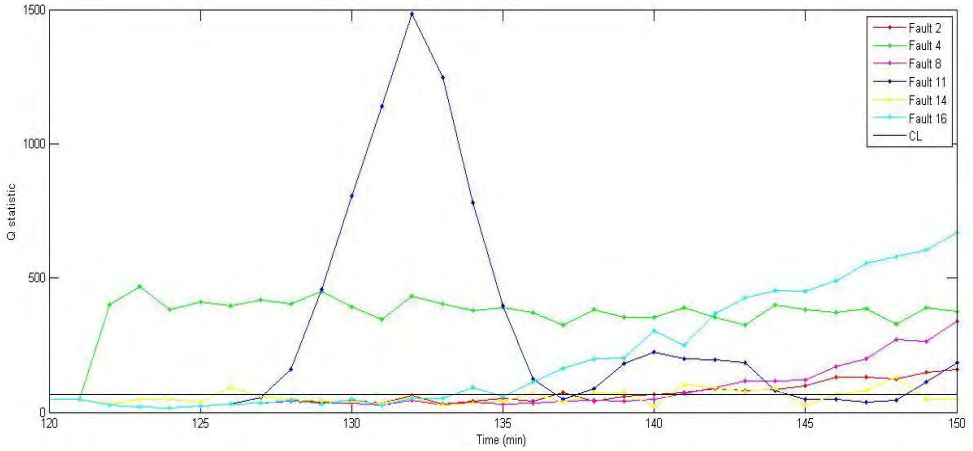

Figure 8.17 Q statistic for some faults showing their delays in minutes

8.3.3.2 Model construction from different data sets

Once the transient stages have been characterized and t_F^0 is determined, the classification models can be constructed. ANN and SVM are used for this purpose as previously mentioned.

The inputs of the classification models are the scores calculated after projecting the transient data onto the second PCA model, which retains the 97% of the variance (31 principal components). Dimensionality reduction is achieved by a high variance with slightly more than half of the variables number in the retained components.

Regarding the ANN structure, this consisted of 31 input nodes. $\mathfrak{IR}^{**}$ set contains 2200 observations, 200 per class, because of the mentioned behavior of the *Q* statistic in some faults and the computational cost caused by including more data. Six hidden nodes were used as they yielded the best classification performance. Finally, a tangent sigmoidal transfer function was chosen with as many output nodes as the number of faults, thus resulting in a 31-6-20 classification network. Regarding SVM, the kernel function that provided the highest diagnosis performance over the validation set $\mathfrak{V}^{**}$ was the fourth degree polynomial.

Both classification methods are next compared on the basis of the following data sets used to construct the corresponding classification models:

- observations corresponding to the completely developed faults (steady-state)
- observations subsequent to the actual fault occurrence (latency period).
- a set of random observations
- transient observations starting with that in which the *Q* statistic exceeds its control limit per fault ($Q_{kF}>Q_a$).
- transient observations starting with that in which the T^2 statistic exceeds its control limit per fault ($T_{kF}^2>T_{lim}^2$).

Table 8.7 Comparison among models constructed with 200 observations and 31 features per class and using ANN. Mean fault diagnosis performance in percentages (F1 score)

FAULT	MODEL BASED ON				
	Steady state data	Starting period of the fault	Random data	Delays with T^2 statistic	Delays with Q statistic
1	0	57.4	86.0	55.6	59.8
2	79.1	63.0	86.0	74.0	83.4
3	0	0	0	0	0
4	0	53.3	0	79.7	73.0
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	38.0
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	0
15	0	0	0	0	0
16	11.2	0	0	49.0	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 8.7 shows the diagnosis performance in terms of F1 percentages considering all the observations in the validation subsets when using ANN as classification algorithm, while Table 8.8 reports the results when applying SVM. The results in these tables reveal that the best diagnosis performance is obtained with those models that take into account the transient stages characterized by using either Q or T^2 statistical indices.

Table 8.7 and Table 8.8 show the F1 scores for the diagnosis of individual faults and for the class 0 (absence of fault). F1 is a score that measures the general performance of the system but excluding the results for the normal predictions. Therefore, the F1 score for the NOC state is a measure of the expected improvement that can still be achieved. The diagnosis of normal conditions is already solved as fault detection and explained by the use of the class 0. The system diagnoses class 0 when none of the fault classifiers diagnose a fault.

Table 8.8 Comparison among models constructed with 200 observations and 31 features per class and using SVM. Mean fault diagnosis performance in percentages (F1 score)

FAULT	MODEL BASED ON				
	Steady state data	Starting period of the fault	Random data	Delays with T^2 statistic	Delays with Q statistic
1	0	82.4	0	83.2	89.2
2	0	69.6	10.0	93.0	94.5
3	15.2	26.4	22.6	31.9	34.2
4	66.5	67.1	57.6	72.0	66.6
5	17.0	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	46.0	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	16.0	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	15.0	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	29.0	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	21.0	55.0	50.6

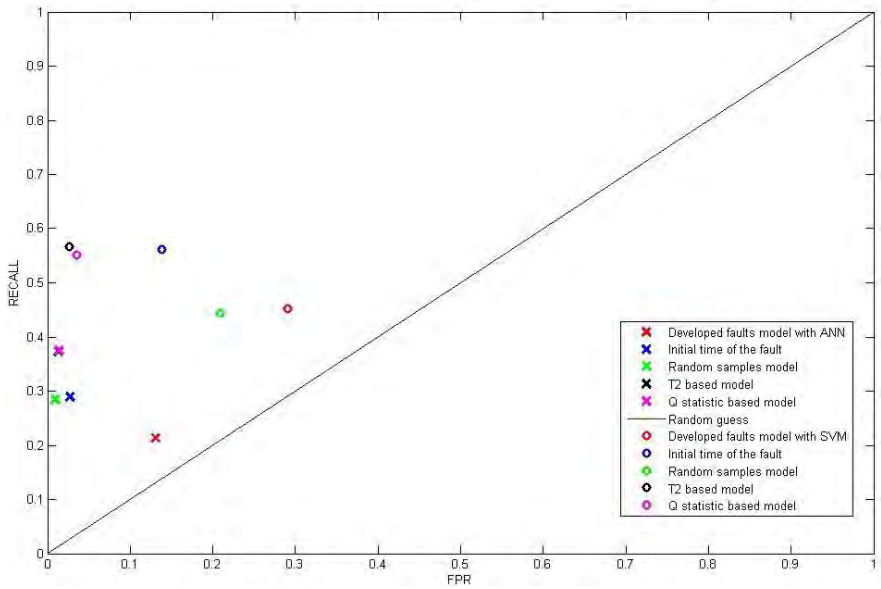

Figure 8.18 ROC diagram for some classification models using ANN and SVM

Figure 8.18 represents a ROC diagram describing the performance of the five previous classification models. The ROC diagram plots the sensitivity or True Positive Rate (recall) against the False Positive Rate (FPR) and allows comparing the classification performance of the models constructed from different data process stages. This diagram corroborates that the best diagnosis models are those constructed with data based on the T^2 and Q statistics and using SVM. The diagram shows the highest recall and the smallest FPR for these models in comparison to the rest.

8.3.3.3 Data window and feature selection

The size of the data window (*nt*) used in the training subsets was studied by considering 200, 300, 400 and 500 observations and 31 components. The results in columns 2 to 5 of Table 8.9 clearly show that the best number of NOC observations for constructing the classification models is **400**. This represents an 87% reduction in size of the original data matrices (K=3000). Further work regarding *nt* should obviously include the consideration of tuning such value for each class *f*, which is addressed in the discussion subsection.

Table 8.9 Study of the data window and PCs variation in the classification model using SVM (taking into account the delays given by the Q statistic). Results in terms of the mean percentages of the diagnosis performance based on the F1 score

FAULT	MODEL BASED ON					
	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**	96.0	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	**60.0**	64.5	56.8	53.9
9	32.3	44.8	**52.8**	0	49.6	49.2
10	62.4	72.4	**84.0**	76.4	83.0	80.2
11	42.9	53.0	**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	**80.0**	90.0	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

The effect of the number of components is also studied. For the 400-sample case (from the score matrices TR_f^{**}) different performances were obtained by retaining 97%, 90%, and 80% of variance in the original variables (31, 26 and 19 components, respectively). The last columns of Table 8.9 present these results, which show that despite significant particular differences, the best average performance is achieved with 31 components (97% of variance). However, retaining more than 97% of variance with the principal components is addressed in the discussions section.

In addition, the performance results are complemented with the corresponding FPR in percentages, given in Table 8.10. As expected, these values are low for most of the faults, except for faults 18 and 20. Note that some faultless observations are diagnosed as occurring faults 18 and 20, which produces an increase of their FPR. In addition, faults 18 and 20 are reported as unknown in the TEP original paper and when they are clustered in a single class (Monroy et al., 2010), the FPR of the new aggregated class decreases. Finally, Figure 8.19 shows the F1 diagnosis performance for the best choice of the data window size and the number of features (400 observations and 31 PCs), split in Precision and Recall.

Table 8.10 Diagnosis performance (F1) and false alarm rate (FPR) in percentages obtained by applying the best classification models to the $\mathcal{U}$ set

Fault	Diagnosis performance (%)	False alarm rate (%)
1	90.1	0.99
2	95.8	0.18
3	53.1	3.76
4	90.8	0.32
5	15.8	1.87
6	99.4	0.05
7	97.6	0.24
8	60.0	1.41
9	52.8	4.09
10	84.0	0.58
11	73.2	2.37
12	71.1	1.93
13	61.9	0.26
14	48.9	2.36
15	23.9	6.36
16	80.0	1.21
17	86.2	0.49
18	14.3	15.14
19	69.9	0.73
20	17.6	40.70
0	6.9	0
Mean	64.3	4.25

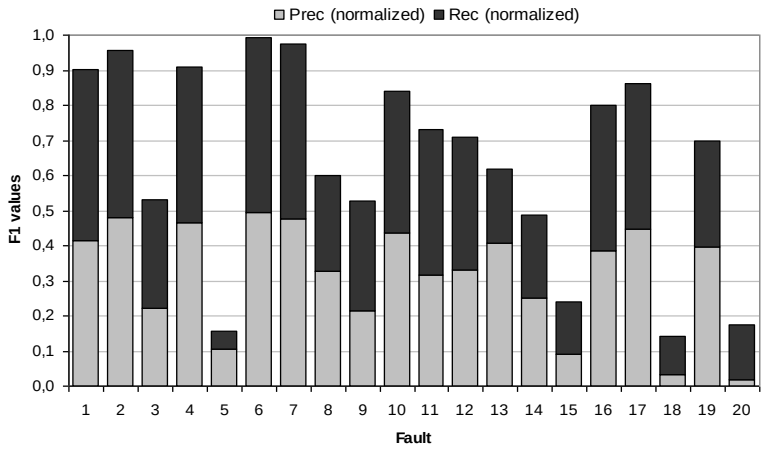

Figure 8.19 F1 score for the model created with 400 observations in the transient regime and 31 features by applying SVM and indicating the Precision and Recall proportions

8.3.3.4 Dynamical response during fault evolution

As expected, each fault in a process evolves in a different way and the dynamic response of each one occur in different time scales for all the measured variables. Thus, the contribution plots are next used to discern the group of variables that most affect to the model or the residuals, and therefore the occurrence of an specific fault. In order to know which process variables most contribute to a fault, it is necessary to determine how each variable contributes to the *Q* statistic (Westerhuis et al., 2000).

$$c_{k,j} = \left(e_{new,j}\right) = \left(x_{new,j} - \hat{x}_{new,j}\right) \qquad \textit{Eq 8.14}$$

In order to evaluate the behavior of the variables during a fault evolution, the residuals of each sample are computed using Eq 8.14. Then, the sum of the contributions of the variables to the *Q* statistic is calculated for periods of 15 minutes (in a similar way to batch processes). The second PCA model of 400 NOC observations is used for calculating the contributions and the control limit.

Figure 8.20 shows the contribution plot for the time period from 16' to 30', subsequent to the actual occurrence of fault 2. Figure 8.21 and 8.22 show the contribution plots for the time periods from 31' to 45' and from 46' to 60', respectively. The contribution plot corresponding to the first 15' is omitted because all the contributions remain far below the control limit. These figures illustrate the different process dynamics and show the variables that most clearly reveal the occurrence of an specific fault. These figures also show that some variables only contribute to the evolution of the fault during a short period of time, while other variables start contributing later.

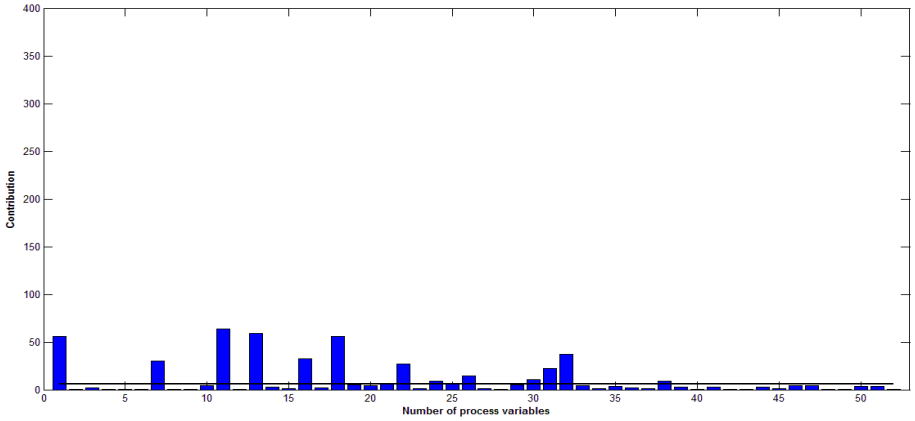

Figure 8.20 Contribution plot for the time period 16' to 30' after the occurrence of Fault 2

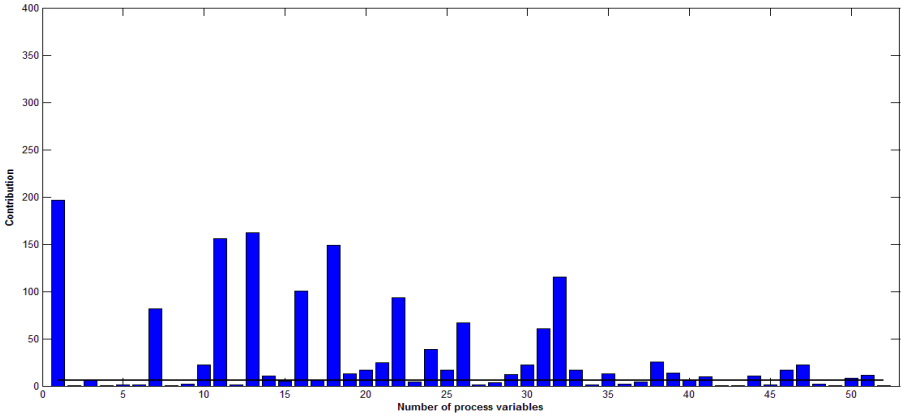

Figure 8.21 Contribution plot for the time period 31' to 45' after the occurrence of Fault 2

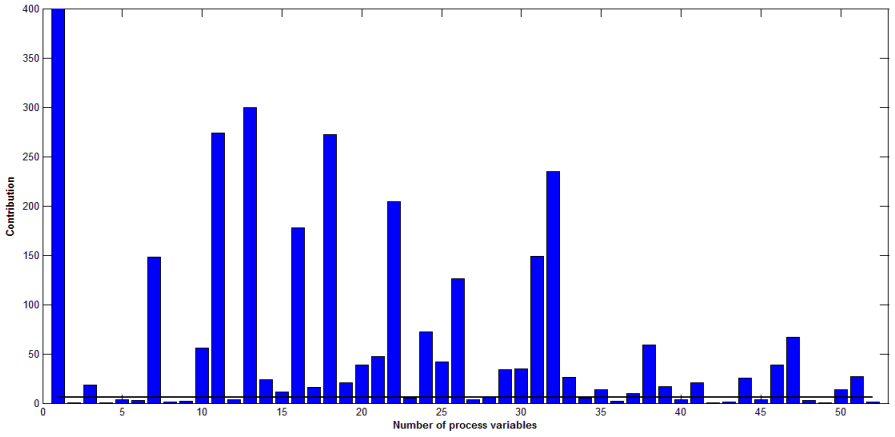

Figure 8.22 Contribution plot for the time period 46' to 60' after the occurrence of Fault 2

Once the process variables that caused the process disturbance have been detected, it is advisable to compare the trajectory of such variables to their normal trajectory in the fault-less process. Figure 8.23 shows the evolution of four process variables (30-33) that were identified as those most contributing to the occurrence of fault 2.

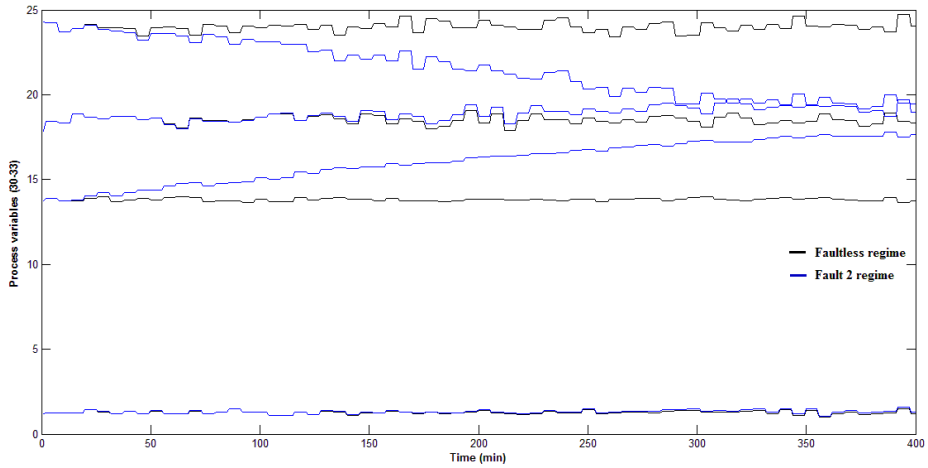

Figure 8.23 Dynamic behavior of four process variables (30-33) for a fault-less regime and the Fault 2 regime

The previous considerations allow not only obtaining potential information about the identification of a fault, but also improving the diagnosis models by including just the variables that most contribute to the occurrence of a specific fault.

8.3.4 Discussions and conclusions about the transient approach

Previous results show that the models considering the transient stages of faults produce better diagnosis performance than those models obtained only from steady-state observations or randomly-chosen observations.

Additionally, the Q statistic has also shown to be faster than T^2 at detecting AOC observations, thus producing shorter time delays and constituting a better means for identifying the transient stages in the fault evolution. On the other hand, the diagnosis performance obtained by the Q-based models is slightly lower than the obtained by the T^2-based models. The fact that the Q-index deals with residual data may explain its faster detection response than the T^2 one.

In addition, a better diagnosis performance is obtained by using observations from transient stages starting when the Q statistic exceeds its control limit in a PCA model that retains the principal components with the broken-stick rule (compare results in Table 8.11, columns 2 and 3). Furthermore, keeping as many principal components as the

number of variables decreases the diagnosis performance in comparison to retaining 97% of the variance (Table 8.11, columns 2, 4 and 5). This is consistent with the fact that retaining less variance effectively removes noise from the data, and explains why the models with 31 components provide better results.

Table 8.11 Comparative results of the PCA models, retaining features by the broken-stick rule or fixing a variance over the 90% and using 200 observations per class. Diagnosis results in percentages given by the F1 score

FAULT	F1 SCORE			
	PCA models retaining 97% of variance (31 PC)		Observations considering the delay with the broken-stick rule	
	Observations considering the delay with the broken-stick rule	Observations considering the delay with the 97% of variance	PCA model retaining the 74% of variance by the broken-stick rule (15 PC)	PCA model retaining 100% of variance (52 PC)
1	89.2	87.7	84.8	90.2
2	94.5	95.2	94.8	81.7
3	34.2	3.9	24.4	35.8
4	66.6	63.9	52.4	83.3
5	2.2	1.2	6.1	1.5
6	99.9	99.9	95.8	70.0
7	99.9	82.1	99.9	99.9
8	42.8	34.5	35.8	56.1
9	32.3	25.0	26.4	33.3
10	62.4	47.4	43.2	47.5
11	42.9	40.8	29.1	55.3
12	36.1	41.4	32.4	33.9
13	33.7	51.9	10.3	18.3
14	39.8	29.6	11.1	33.9
15	16.7	3.1	15.1	14.5
16	71.6	70.2	84.8	86.2
17	68.5	50.6	76.0	72.9
18	0.5	47.2	0.01	3.0
19	38.2	8.6	22.3	35.1
20	39.2	45.4	37.4	34.1
0	14.7	21.3	7.9	7.7
Mean	50.6	46.5	44.2	49.3

The effect of the size of the transient window per fault is also considered. A reasonable assumption is that adapting the size of the window to the characteristic time scale of each fault may result into a reduction of computational cost and an improvement in the overall performance. In order to address this problem, a simulation is performed with a different number of observations per fault considering the widow size showing the best diagnosis performance according to the results in Table 8.9.

Table 8.12 Diagnosis performance obtained with a classification model constructed with the best transient-observations window per class in terms of the F1 score

Fault	Number of observations	Diagnosis performance (%)
1	400	77.9
2	300	95.8
3	500	52.3
4	400	91.0
5	500	16.3
6	200	99.9
7	200	99.9
8	500	61.5
9	400	52.5
10	400	83.6
11	500	65.6
12	400	70.8
13	400	13.2
14	500	54.9
15	500	21.1
16	500	75.7
17	400	86.3
18	300	12.0
19	500	67.7
20	300	49.1
0	400	6.5
Mean		62.4

Table 8.12 reports the corresponding results and shows that despite the effort to find the number of observations that best describe the transient stage of each fault, the best diagnosis performance is obtained with a constant number of observations for every scenario (400 observations).

In order to estimate the capacity of the proposed transient approach to handle the AD issue, new classification models are constructed with data subsets occurring faults 2 to 20 and explicitly excluding data related to fault 1. This model is then tested against data from fault 1 and the results show that the situation is detected as faulty, while it is jointly diagnosed with faults 6, 7, 10, 11, 12, 15 and 18. Although the method cannot be expected to diagnose unprecedented faults, it is capable of detecting faults that have not been included in the diagnosis models. However, its performance and robustness require to be further investigated.

The comparative study of classification algorithms shows no significant performance discrepancy between the use of SVM or ANN. The main difference is that SVM attains slightly higher diagnosis performances than ANN. The fact that the performance improvement occurs regardless of the learning method also supports the convenience of using transient stages for constructing classification models.

8.3.4.1 Normal Vs abnormal transitions

A relevant issue to discuss is how to deal with normal, non-faulty transitions. In order to address this problem, two simulations have been performed by modifying the base values of two manipulated variables: the A feed flow (variable 44, from 0.251 m^3/h to

0.271 m^3/h) and the stripper liquid product flow (variable 49, from 22.848 m^3/h to 23.83 m^3/h). The first simulation is faultless while the second includes the occurrence of fault 2. Accordingly, two 500-sample test sets were prepared and used to evaluate the best diagnosis model. The performance results obtained for the faulty and faultless situations are given and compared in Table 8.13.

The results show that the model is able to distinguish between faulty and normal transitions. It is worthy to note that the FPR for faults 18 and 20 is higher in the faultless case, in accordance to the high values also found previously in the subsection 8.3.3.3. Regarding the test subset with fault 2, the diagnosis performance is almost 100% and the FPR for the rest of faults is negligible.

The use of data-based diagnosis models with supervised-learning algorithms produces particular models for specific abnormal situations. In turn, this loss of generalization means that patterns different to those used to train the algorithm will never be diagnosed as such. However, distinguishing between normal transitions and true faults is a real and practical concern to solve. An example of this has been addressed and the results reported indicate the potential of the transient data approach and the algorithms applied to cope with this issue.

Table 8.13 Diagnosis performance (F1 score) and FPR of test data sets in fault-less fault 2 regimes, applying the best classification model and SVM

FAULT	F0 DATA		F2 DATA	
	F1 (%)	FPR (%)	F1 (%)	FPR (%)
1	-	0	-	0.2
2	-	0	97.8	0
3	-	3.6	-	1.0
4	-	0	-	0
5	-	0	-	0
6	-	0	-	0
7	-	0	-	0
8	-	0	-	0.2
9	-	4.2	-	1.0
10	-	0	-	0.6
11	-	0	-	1.8
12	-	0.8	-	1.6
13	-	0	-	0.2
14	-	0.2	-	1.8
15	-	10.8	-	4.2
16	-	0	-	0
17	-	0	-	0
18	-	48.8	-	0.4
19	-	0.4	-	0.4
20	-	93.0	-	4.2
0	100.0	0	-	0
Mean	100.0	8.1	97.8	0.9

8.3.4.2 Validation of transient and steady-state models on data from different process stages

The last and relevant issue to discuss with the transient approach is the validation of the constructed diagnosis models on data sets from different stages of the process, such as latency and steady-state and the comparison with the validation of a model constructed with steady-state data on latency and transient data.

In order to be able to compare models both models, the same methodology, techniques and test data set have to be used. Thus, the steady-state data model developed in section 8.2 is reproduced by using the methodology addressed in the transient data approach, which means the application of PCA and SVM techniques. As summary, the TR_f subsets were composed of 200 samples (from 2201 to 2400), V_f composed of 100 samples (from 2101 to 2200) and TT_f composed of 600 samples (from 1501 to 2100).

Table 8.14 shows the mean diagnosis performance in percentage considering the 20 faults of the TEP and some scenarios indicated in the first four columns according to the data sets used, number of retained principal components and kernel in SVM. The first trial is the use of a linear kernel as in the steady-state data approach.

Table 8.14 Application of the transient data methodology to steady-state data

Data sets		Number of	Kernel	Mean diagnosis perf.
Training	Test	PCs		(F1 score in %)
Training	Validation	31	Linear	44.64
Training	Validation	43	Linear	46.01
Training	Validation	52	Linear	52.20
Training	Validation	**52**	**Polynomial 2**	**61.98**
Training	Validation	52	Polynomial 3	42.79
Training	Validation	52	Polynomial 4	19.17
Training+Validation	Test	**52**	**Polynomial 2**	**61.95**
Training	Test	52	Polynomial 2	56.83

According to the results in Table 8.14, it can be extracted that by applying the methodology used in the transient data approach, a slight improvement in the mean diagnosis performance is obtained by comparing the obtained 61.98% against the 58.2% obtained with the steady-state data methodology as reported in Table 8.3. However, the main non-reproducible aspect between both methodologies is the application of a polynomial kernel of second degree with the transient data methodology in comparison to a linear kernel, which could have a significant influence on the results.

It is also important to mention that some tests with 31 components (representing the 97% of variance as in the transient approach) and 43 components (representing the 100% of variance) were executed. The results allow suspecting that a better diagnosis performance can be obtained by retaining the 100% of the variance in the principal components when applying PCA. Also, a better performance is obtained with the methodology from the transient data approach using $\mathfrak{IR}$+$\mathfrak{V}$ as training set (61.95%, as extracted from Table 8.14), in comparison to the 49.9% (from Table 8.4) with the steady-

state data methodology. The second column from Table 8.15 breaks down this performance for all the faults.

Then, this model constructed with steady-state data is applied and validated on the $\mathcal{V}$ set used in the transient data approach, which includes exactly data from the process in this stage. The third column from Table 8.15 shows in detail the corresponding results and demonstrates that, the steady-state data model performs inefficiently when diagnosing faults during their transient stage. This is not of course, an unexpected behavior.

Table 8.15 Comparison of the steady-state and transient data models validated on steady and transient data. Diagnosis performance based on F1 score (%)

Fault	Steady-state data model validated on		Transient data model validated on	
	Steady-state data	Transient data	Transient data	Steady-state data
1	100.0	0	90.5	69.7
2	100.0	20.5	96.3	90.0
3	65.7	27.0	55.5	42.6
4	96.7	89.9	93.0	20.3
5	59.7	51.2	50.1	15.6
6	100.0	0	99.9	100.0
7	100.0	0	99.9	66.7
8	32.2	7.3	62.2	20.6
9	24.4	10.2	51.7	21.3
10	70.7	36.9	87.8	36.1
11	48.6	39.3	72.2	23.1
12	43.9	35.7	76.5	47.9
13	71.6	24.5	57.6	77.6
14	34.1	33.4	55.8	44.6
15	21.3	16.8	16.3	26.9
16	49.2	77.0	89.3	32.9
17	76.3	1.6	91.7	78.1
18	56.1	10.7	58.1	18.1
19	50.0	0.2	66.4	18.8
20	38.5	0.8	41.0	26.6
0	23.5	15.2	35.0	9.1
Mean	**62.0**	24.1	**70.6**	43.9

In contrast, the transient data model constructed in this subsection is also validated on steady-state data ($\mathfrak{IJ}^*_{ext}$ set used before in the steady-state data approach). It is observed from Table 8.14 that retaining the 100% of variance renders better diagnosis performance. Consequently, the transient data model is constructed with 400 observations per scenario, as previously; however, the principal components that keep the 100% of variance (44 components) are retained, as well as the total 52 components.

Table 8.16 shows that, 44 components retaining the 100% of variance offer better diagnosis performance when still applying SVM with a polynomial kernel of fourth degree. Additionally, other kernels were evaluated and it was found that an improvement in the diagnosis performance is obtained with a second degree polynomial. These results show that the kernel function in SVM applied to model data depends on the number of observations used to construct the model; however, a deeper investigation would be required to conclude this observation. In this specific case, for 200 observations of the TEP, the best kernel was polynomial of fourth degree but for 400 observations, a

polynomial function of second degree. Moreover, it is observed that retaining the 100% of the variance of the variables with the principal components enhances the diagnosis performance when using a feature extraction technique (in this case PCA) before the classification method.

Table 8.16 Effect of the PCs number and kernel on the performance of the transient data model

Number of PCs	Kernel function	Mean diagnosis performance (F1 score in %)
31	Polynomial 4	64.30
52	Polynomial 4	67.16
44	**Polynomial 4**	**67.33**
44	Polynomial 3	37.53
44	**Polynomial 2**	**70.59**
44	Linear	43.0

Finally, the two last columns from Table 8.15 show in detail the diagnosis performance for the 20 faults of the TEP applying the new transient data model, which is constructed by retaining 44 PCs and using a polynomial kernel of second degree. Furthermore, when this model is validated on steady-state data, the diagnosis performance decreases; however, this performance is better than the obtained from applying a steady-state data model on transient data.

Therefore, it is recommended to construct and apply two models when diagnosing a continuous process, a transient data and a steady-state data model. Both models should be applied at the same time when diagnosing on-line data, relying on the one with higher performance. Consequently, the diagnosis would be more robust and less delayed.

8.4 Comparison of the diagnosis performance from the proposed approaches with reported performances of the TEP

Most of the papers that present an approach to fault diagnosis and are applied to the TEP deal with only some faults of the twenty in total. According to the research on this process, three specific faults are found to cause problems with their detection and diagnosis (3, 9 and 15). Moreover, some of the faults have a similar pattern, which makes them to overlap when data from them is projected on a feature space. Because of that, almost all the papers have just reported the diagnosis performance of their approaches based on some faults.

The results of some papers (from data models on steady-state), which corroborate the previous statement, are next listed. Also, a comparison with the results obtained with both approaches proposed in this thesis (steady-state and transient) is discussed.

Chiang et al. (2004) applied Fisher Discriminant Analysis (FDA), Support Vector Machines (SVM) and Proximal SVM (PSVM) in their FD approach. It was validated on only three faults of the TEP (4,9,11) and the diagnosis performance obtained (misclassification percent) is shown in Table 8.17.

Table 8.17 Classification results for faults 4,9 and 11 using SVM, PSVM and FDA, as reported by Chiang et al. (2004)

	SVM	SVM	SVM	PSVM	FDA
Kernel (parameter)	RBF, σ=(1,1,1)	Polynomial (1,2,2)	Polynomial (1,2,2)	Polynomial (1,2,2)	-
Regularization parameter (C)	920000	1.2 E11	1	2	-
FDA rank	-	-	-	-	2
Misclassification percent (training data)	1.3	0	2.9	5.7	22
Misclassification percent (test data)	44	45	44	35	38

In the transient approach proposed in this thesis, the mean of the diagnosis performance of the faults 4, 9 and 11 (72.3%), from Table 8.15, is higher than the 65% reported in this work. Also, the steady-state data approach that clusters some faults of the TEP renders a mean of 62.4% for these three faults (from Table 8.5).

Jakubek and Strasser (2004) apply Artificial Neural Networks (ANN) as fault detection algorithm and use four faults in the TEP (1,4,5,11) and report the performance of their approach with a classification rate as shown in Table 8.18.

Table 8.18 Detection results for the TEP as reported by Jakubek and Strasser (2004)

Fault#	1	4	5	11
Detection rate (%)	99.0	96.0	60.0	74.0
Over-detection rate (%)	0.6	8.1	9.3	6.8
Proper classification rate (%)	**99.0**	**95.0**	**65.0**	**77.0**

The mean classification rate of these four faults is 84%. The steady-state approach proposed in this thesis, gathering some faults, renders a mean diagnosis of 78.2% (from Table 8.5). However, it is important to keep in mind that the approaches presented in this thesis consider the twenty faults simultaneously, which allows considering the FPR of the faults.

Li and Cui (2009) reported an approach based on Kernel Fisher Discriminant Analysis (KFDA) with Feature vector selection (FVS). They also applied their approach on four faults in the TEP (1,5,7 and 14) and reported the diagnosis performance as a mean of the four scenarios with an index called maximum recognition rate in percentages.

Table 8.19 Methods comparison as reported by Li and Cui (2009)

Method	Training sample number per class	Maximum recognition rate (%)
Without FVS	80	**83.0**
FVS-60	60	82.2
FVS-30	30	80.6

The steady-state approach that clusters some of the TEP faults performs a mean diagnosis of 90.7% for the four faults considered in the work reported by Li and Cui, which is better than the 83% as observed in Table 8.19.

Bin He et al. (2009) applied FDA and Variable-Weighted FDA (VW-FDA) for diagnosing two groups of faults and measure the performance with the diagnosis success rate in percent as shown in Table 8.20.

Table 8.20 Comparison among diagnosis success rates using different approaches in Bin He et al (2009)

		Diagnosis success rate (%)	
		VW-FDA	FDA
Case 1	Fault 9	90.0	78.0
	Fault 4	90.0	90.0
	Fault 11	41.0	30.0
	Total	73.7	66.0
Case 2	Fault 2	100.0	100.0
	Fault 5	100.0	100.0
	Fault 8	99.0	98.0
	Fault 12	70.0	66.0
	Fault 13	72.0	44.0
	Fault 14	90.0	95.0
	Total	88.5	83.8

Although TEP is a widely used benchmark, the results are hardly comparable in a quantitative basis due to different faults and indices addressed by authors.

In general, the approaches proposed in this thesis, the steady state data (applying feature extension, squared standardization and SVM) and the transient data (applying centering, scaling, PCA as feature extraction technique and SVM), yield better diagnosis performance in comparison to the results found in literature.

Finally, the approaches addressed in this thesis deal with the 20 faults from the TEP, which decreases the general diagnosis performance because some of these faults report a significant false positive rate as in the case of the faults 18 and 20. Even more, some fault-less observations are diagnosed as occurring these last faults and the problematic faults 3, 9 and 15. This situation has been investigated and faced in this chapter.

8.5 Conclusions about the fault diagnosis on continuous processes

This chapter has presented the Fault Diagnosis approaches based on data-driven models for continuous processes and specifically on the process period from where data is taken. In this sense, such approaches use a) data from periods at which the fault is fully developed (steady-state), b) data from the period starting with the observation where the faults are generated but not necessarily detected (latency stage), c) random observations and d) data from the period at which the fault is detected (transient period).

The methodologies proposed to address each approach differ slightly in the application of different techniques and some additional steps. However, the transient data methodology is applied to data from the four stated stages and the results are compared. Data from steady and transient stages show to perform the best diagnosis performances when used for constructing the diagnosis models. All the methodologies are tested with the TEP benchmark considering all the 20 faults reported.

The transient data approach renders the best diagnosis performance when validated on transient data. The keystone of the approach 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. The performance of this approach does not depend on the particular classification technique used. The results are slightly affected by the details in the size of the transient time window, as reported by using 200 and 400 observations per class. Moreover, the system is tolerant to the presence of unknown faults, which are detected as an abnormal plant behavior. On the other hand, the early detection of the fault is shown to be a crucial aspect in order to provide a proper determination of the transient state.

However, the transient models do not work with the same good performance obtained from their validation on transient data when validated on data from the process in steady-state. In contrast, this last performance is higher than the obtained from the validation of steady-state data models on transient data. Therefore, these results lead to the conclusion of using two diagnosis models (a transient and a steady-state) when dealing with on-line process data in order to obtain a reliable diagnosis of the process. Also, sometimes it could be required to group some original faults registered with the historical data into new classes in order to improve the fault diagnosis performance as presented in the TEP, demonstrated with the steady-state data approach.

Finally, the diagnosis performance values obtained from the proposed approaches are compared with the values found in some references and published articles, demonstrating the potential of the proposed approaches in this chapter. Although TEP is a benchmark widely used in academics and research, it has been revealed that it is difficult to compare diagnosis results among the approaches that have been investigated and reported. This is due to the use of different data sets, different performance indices and the use of only some of the whole faults that can be addressed with the original process. Therefore, an important contribution of this thesis is the proposal of several FD approaches that are applied to the complete set of 20 faults of TEP. Furthermore, the

diagnosis performance is evaluated under a single index, the F1 index, which is encouraged to still be used for further investigations.

Finally, the most important thing when dealing with Fault Diagnosis is the methodology to be used and the precise process information, no matter what techniques or algorithms are used, as investigated, contributed to the area of continuous processes and demonstrated in this chapter.

Chapter 9
FAULT DIAGNOSIS ON BATCH PROCESSES

Batch and semi-batch processes play an important role in the chemical industry, particularly for producing high value-added products in relatively small volumes. Many chemical, pharmaceutical and food products are produced in batch processes, which are characterized by finite duration, unsteady-state behavior, high conversions and, most importantly, a recipe-driven operation. Monitoring the behavior of the batches is important for process optimization and safety, meeting product specifications and quality improvement, waste reduction, and enhancing the process knowledge.

Batch processes suffer a lack of reproducibility and variations from batch to batch due to disturbances. These variations may be difficult for an operator to discern, but could have an adverse effect on the final product quality. Additionally, on-line quality measurements may be difficult, expensive, and sometimes unavailable. Often, a disturbance or an operational problem can be undetected and the poor product quality may remain in such process state until significant expense has been incurred.

Several techniques have been developed for process monitoring and diagnosis of batch chemical processes. These techniques were broadly classified into three categories: model-based techniques, expert systems and data-driven methods (Himmelblau, 1978; Pau, 1981). The first two categories have been developed earlier in history. However, the advent of an increasing computational capacity and the need for operating ill-understood processes have expanded the attention to the third category, which establishes models on the basis of historical data with minimal input of knowledge. For instance, the complexity of biochemical processes such as the production of antibiotics makes difficult to create a detailed and practical model. Instead, empirical models based on process historical data to supplement simple mechanistic models are available (Kourti 2002).

As a result, a growing interest in the use of Multivariate (MV) techniques in batch process modelling and Fault Detection and Diagnosis (FDD) has been noticed in literature (Dahl et al., 1999; Westerhuis et al., 1999; Wise et al., 1999; Cinar et al., 2003). A large part of the available literature is focused on the development and application of the so called latent variable methods such as PCA and PLS. These methods can handle highly correlated data sets and allow analysis and visualization which aids in the understanding of process data and possibly of the process itself. Furthermore, these MV techniques are well-suited for on-line Statistical Process Monitoring (SPM) and FDD of batch production and biotechnological processes (Nucci et al., 2010).

Pioneering work in the area of batch processes monitoring was performed by Nomikos and MacGregor (Nomikos and MacGregor, 1994; Nomikos and MacGregor, 1995a) and has been successfully applied to industrial processes on several occasions (Kosanovich et al., 1996; Neogi and Schlags, 1998; Tates et al., 1999). Here, the basis is to model the common-cause variation present in collected data obtained under NOC. This model is subsequently used to determine whether a new batch corresponds to this historically recorded normal operating behavior or not. Therefore, the monitoring performance depends heavily upon this NOC data (Van Sprang et al., 2002).

Deviations in process variables during the progress of a batch can provide information about product properties and an estimation of the quality of the final product well before the completion of the batch. Process monitoring and FD have been very effective in achieving this goal of process supervision (Ündey et al., 2003). More specifically, MPCA has been successfully applied to batch processes in order to monitor the process and identify when the process shifts to a new operating condition, detect and diagnose abnormalities (Nomikos and MacGregor, 1994; Chen and Liu, 1999).

As in the last two decades a plethora of techniques for fault detection and diagnosis have risen (Venkatasubramanian et al., 2003a, 2003b; 2003c), it has become difficult for practitioners in research and industry to choose a method. Moreover, for most commercial processes suitable techniques exist for monitoring and diagnosis. As such, a collaborative effort has been set up in view of establishing guidelines for the choice between some techniques, rather than extend the existing that have been reported to perform successfully in bioprocesses (Pokkinen et al., 1992; Román et al., 2011).

This chapter tackles the application of the general FDS proposed in this thesis on batch processes. In chapter seven, two main approaches were brought up according to the moment in which diagnosis is performed either at the end of the batch or at each time interval (on-line). The next sections will address both approaches and their application to a specific biotechnological case study: the Birol Penicillin Process (BPP, by Birol et al., 2002).

9.1 Diagnosis at the end of batch

As batch processes have different time trajectories in their variables, it is more reliable to construct monitoring or diagnosis models based on the whole batch information in contrast to construct several models for different parts of the process. This is why the approach at the end of batch has been widely used not only in academic works, but also in industry.

The end of batch approach has been addressed in this thesis by using a combination of four techniques: Multiway Principal Component Analysis (MPCA), Multiway Independent Component Analysis (MICA), Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The purpose is to find out whether the choice for MPCA or MICA as feature extraction method and the choice for ANN or SVM as classification technique is important for finding a suitable FD strategy in batch processes.

The comparison among the four combined approaches that results from selecting one technique from the feature extraction methods and one from the classification is applied to several process scenarios with different levels of noise in the data so as to assess how noise affects the diagnostic performance of the individual approaches as well as generalize the result of the best approach.

9.1.1 Case study

The BPP model has been implemented in Matlab. The state variables of the process are: culture volume (V), reactor temperature (T), reaction heat rate (Q_{rxn}), substrate concentration (S), biomass concentration (X), penicillin concentration (P), dissolved oxygen concentration (C_L), carbon dioxide concentration (C_{CO2}), and pH (-log C_H, where C_H is the hydrogen ion concentration). Equations 9.1 to 9.13 show the differential equations for all the state variables included in the model, where μ is the specific growth rate, μ_{pp} is the specific penicillin production rate, F_{loss} accounts for the evaporated loss during fermentation, T_0 and T_v are the freezing and boiling temperatures of the culture medium, which is assumed to have the same properties as water, k_{la} is the overall mass transfer coefficient and B is a term that adds the contribution of acid and base volume in the culture to the pH.

The original nomenclature is given in Table 9.1, which also provides the data for the case study. The extension of the original model consists of the inclusion of a more practical PI controller for temperature and pH, as well as the related variables: acid flow rate (F_a), the base flow rate (F_b) and the heating/cooling water flow rate (F_c).

$$\frac{dV}{dt} = F + F_a + F_b - F_{loss} \qquad \textit{Eq 9.1}$$

$$F_{loss} = V\,\lambda\left(e^{\eta\left(\frac{T-T_o}{T_v-T_o}\right)} - 1\right) \qquad \textit{Eq 9.2}$$

$$\frac{dT}{dt} = \frac{F}{s_f}(T_f - T) + \frac{1}{V\,\rho\,C_p}\left(Q_{rxn} - \frac{a\,F_c^{b+1}}{F_c + a\,F_c^{b}/2\,\rho_c c_{pc}}\right) \qquad \textit{Eq 9.3}$$

$$\frac{dQ_{rxn}}{dt} = r_{q1}\frac{dX}{dt}V + r_{q2}XV \qquad \textit{Eq 9.4}$$

$$\frac{dX}{dt} = \mu X - \frac{X}{V}\frac{dV}{dt} \qquad \textit{Eq 9.5}$$

$$\mu = \mu_x \;\frac{1}{1 + \frac{K_1}{C_H} + \frac{C_H}{K_2}}\;\frac{S}{K_x X + S}\;\frac{C_L}{K_{ox} + C_L}\left(k_g e^{\frac{E_g}{RT}} - k_d e^{\frac{E_d}{RT}}\right) \qquad \textit{Eq 9.6}$$

$$\frac{dS}{dt} = -\frac{\mu}{Y_{x/s}} X - \frac{\mu_{pp}}{Y_{p/s}} X - m_x X + \frac{Fs_f}{V} - \frac{S}{V}\frac{dV}{dt} \quad \textit{Eq 9.7}$$

$$\mu_{pp} = \mu_p \; \frac{S}{K_p + S + \frac{S^2}{K_I}} \; \frac{C_L^p}{K_{op} X + C_L^p} \quad \textit{Eq 9.8}$$

$$\frac{dP}{dt} = \mu_{pp} X - K\, P - \frac{P}{V}\frac{dV}{dt} \quad \textit{Eq 9.9}$$

$$\frac{dC_L}{dt} = -\frac{\mu}{Y_{x/o}} X - \frac{\mu_{pp}}{Y_{p/o}} X - m_o\, X + K_{la}\left(C_L^* - C_L\right) - \frac{C_L}{V}\frac{dV}{dt} \quad \textit{Eq 9.10}$$

$$K_{la} = \alpha \sqrt{f_g \left(\frac{P_w}{V}\right)^{\beta}} \quad \textit{Eq 9.11}$$

$$\frac{dC_{CO2}}{dt} = \alpha_1 \frac{dX}{dt} + \alpha_2\, X + \alpha_3 \quad \textit{Eq 9.12}$$

$$\frac{dC_H}{dt} = \gamma\left(\mu\, X - \frac{F\, X}{V}\right) + \left(\frac{-B + \sqrt{B^2 + 4 K_w}}{2}\right)\frac{1}{dt} \quad \textit{Eq 9.13}$$

$$B = \frac{\left(\frac{K_w}{C_H} - C_H\right)V - F_a C_a dt + F_b C_b dt}{V + \left(F_a + F_b\right)dt} \quad \textit{Eq 9.14}$$

The static form of the PI control algorithm is expressed with the next equations. Parameter values are given in Table 9.1. Specifically, the control action U^n_k is calculated as in Eq 9.15, which is a function of the error E^n_k and the integral and proportional signals (I^n_k and P^n_k):

$$U_k^n = I_k^n + P_k^n \quad \textit{Eq 9.15}$$

$$I_k^n = I_{k-1}^n + \left(\frac{K_c^n}{\tau_I^n} \times E_k^n\right)dt \quad \textit{Eq 9.16}$$

$$P_k^n = K_c^n \times E_k^n \quad \textit{Eq 9.17}$$

$$E_k^n = Y_k^n - Y_{SP}^n \quad \textit{Eq 9.18}$$

Set points (Y^n_{SP}) are established at pH=5 and T=298 K, and the integration step *dt* is set to match the sampling interval (0.02h). Furthermore, the substrate feed rate (*F*) is controlled by an on/off controller, which switches operation to fed-batch when the glucose concentration reaches the 0.3 g/l threshold.

The initial values of some of the input variables (feed substrate concentration, substrate feed temperature, power density, and air flow rate) and the set points of the controlled variables (temperature, pH, and substrate feed rate in the fed-batch stage) were randomly sampled from independent normal distributions with the mean values reported in Table 9.1 and standard deviations equal to 1% of their corresponding mean.

Table 9.1 Initial conditions, kinetic and controller parameters for normal operation

Parameter symbol	Parameter description	Value	Unit
Initial conditions			
C_{CO2}	Carbon dioxide concentration	0.5	mmol/l
C_H	Hydrogen ion concentration	$10^{-5.1}$	mol/l
C_L	Dissolved oxygen concentration (=C_L^* at saturation)	1.16	mg/l
P	Penicillin concentration	0	g/l
Q_{rxn}	Heat generation	0	Cal
S	Substrate concentration	15	g/l
T	Temperature	297	K
V	Culture volume	100	L
X	Biomass concentration	0.1	g/l
Kinetic parameters and variables			
A	Heat transfer coefficient of cooling/heating liquid	1000	Cal/h °C
B	Constant	0.60	-
C_a, C_b	Acid and Base molarity	3	M
E_d	Activation energy for cell death	50000	Cal/mol
E_g	Activation energy for growth	5100	Cal/mol
F	Feed flow rate of substrate	0.042	l/h
F_a	Acid flow rate	-	l/h
F_b	Base flow rate	-	l/h
f_g	Air flow rate	20	m^3/h
K	Penicillin hydrolysis rate constant	0.04	h^{-1}
k_d	Arrhenius constant for cell death	10^{33}	-
k_g	Arrhenius constant for growth	$7 . 10^3$	-
K_I	Inhibition constant for product formation	0.10	g/l
K_{ox}, K_{op}	Oxygen limitation constant (no limitation)	0	-
K_p	Inhibition constant	0.0002	g/l
K_w	Ionic product of water	$1 . 10^{-14}$	[-]
K_x	Contois saturation constant	0.15	g/l
K_1	Constant	10^{-10}	mol/l
K_2	Constant	$7. 10^{-5}$	mol/l
m_x	Maintenance coefficient on substrate	0.014	gsubstrate/gbiomassh
m_o	Maintenance coefficient on oxygen	0.467	gO_2/gbiomass h
P	Constant	3	-
P_w	Power density	600	W
r_{q1}	Yield of heat generation	60	Cal/g biomass
r_{q2}	Constant in heat generation	$1.6783 . 10^{-4}$	Cal/g biomass h
s_f	Feed substrate concentration	600	g/l
T_f	Feed temperature of substrate	298	K
T_v	Boiling temperature of the culture medium	373	K
T_0	Freezing temperature of culture medium	273	K
$Y_{x/o}$	Yield constant of biomass on oxygen	0.04	g biomass/g oxygen
$Y_{x/s}$	Yield constant of biomass on substrate	0.45	g biomass/g glucose
$Y_{p/o}$	Yield constant of product on oxygen	0.20	g penicillin/g oxygen
$Y_{p/s}$	Yield constant of product on substrate	0.90	g penicillin/g glucose
α, β	Constants in K_{la}	70, 0.4	-
α_1	Constant relating CO_2 to growth	0.143	mmol CO_2/g biomass
α_2	Constant relating CO_2 to maintenance energy	$4 . 10^{-7}$	mmol CO_2/g biomass h
α_3	Constant relating CO_2 to penicillin production	10^{-4}	mmol CO_2/l h
Υ	Proportionality constant	10^{-5}	mol $[H^+]$/g biomass
λ	Evaporation rate	$2.5 . 10^{-4}$	l/h
μ_p	Specific rate of penicillin production	0.005	gpenicillin/gbiomass h
μ_x	Maximum specific growth rate	0.092	h^{-1}
ρ C_p	Density x heat capacity of medium	1500	Cal/ºC
ρ_c C_{pc}	Density x heat capacity of cooling liquid	2000	Cal/ºC
Controller parameters			
K^1_c	Proportional part of Acid in pH control	$1 . 10^{-4}$	-
τ^1_I	Integral proportion of Acid in pH control	8.4	h
K^2_c	Proportional part of Base in pH control	$8 . 10^{-4}$	-
τ^2_I	Integral proportion of Base in pH control	4.2	h
K^3_c	Proportional part of cooling in Temperature control	70	-

τ^3_I	Integral proportion of cooling in Temperature control	0.5	h
K^4_c	Proportional part of heating in Temperature control	5	-
τ^4_I	Integral proportion of heating in Temperature control	0.8	h

Noise was also added to the input and output variables at four different levels (0%, 1%, 5% and 10%) in the model. Different nominal values, from which the noise is calculated, are set for each input, controlled and output variable.

In order to simulate AOC in the process, three different faults are introduced:

1. Decrease in the agitation power from 600 W to values between 30 and 200 W,
2. Increase of the saturation constant (K_x) from 0.15 g/l (nominal value) to values ranging from 0.3 to 0.9 g/l, and
3. Decrease of the substrate feed rate in the fed-batch stage from 0.042 l/h to values ranging from 0.001 to 0.01 l/h.

It is worthy to notice that both faults 1 and 3 are likely caused by faults in the equipment, while fault 2 would likely be generated by human error (culture contamination or the addition of an impure substrate), which affects the saturation constant value. The chosen sampling time was set to 0.02 h by default, as in the original work (Birol et al., 2002). In the process, a fed-batch operation follows the batch operation when the carbon source (glucose) almost depletes (0.3g/l). All the simulated batches lasted four-hundred hours. For the monitoring and diagnosis steps, one sample per hour is taken, giving as result a matrix per batch of dimensions 401x9.

9.1.2 Integration of feature extraction and classification methods

In order to monitor and diagnose batches at the end, it is required to unfold the process information batch-wise and then to have an $I \times KJ$ arrangement. As this data ($\mathfrak{IR}$) is centered and scaled ($\mathfrak{IR}^*$) before applying a latent or feature extraction method, the variability among time observations and process variables during a batch is taken into account with the retained components. Also the distribution of classes is considered by including batches of all the reported faults.

In this way, each row of the training matrix ($\mathfrak{IR}^{**}$) containing the scores obtained from the feature extraction method represent a whole batch instead of a single sample. The multi-class and multi-label machine learning algorithms are therefore applied off-line as in continuous processes for constructing the diagnosis models per each fault. Finally, the validation batches go through the same data arrangement and improvement process before applying the models. The result is thus the diagnosis of each validation batch. In the same way, new test batches are diagnosed once finished.

Specifically, fault detection is performed in this approach using MPCA. Next, four different options for diagnosis are analyzed, which result from combining two feature extraction methods (MPCA and MICA) and two classification methods (ANN and SVM). The

resulting two-step procedure is summarized in Figure 9.1, and the two steps are further explained in the next subsections.

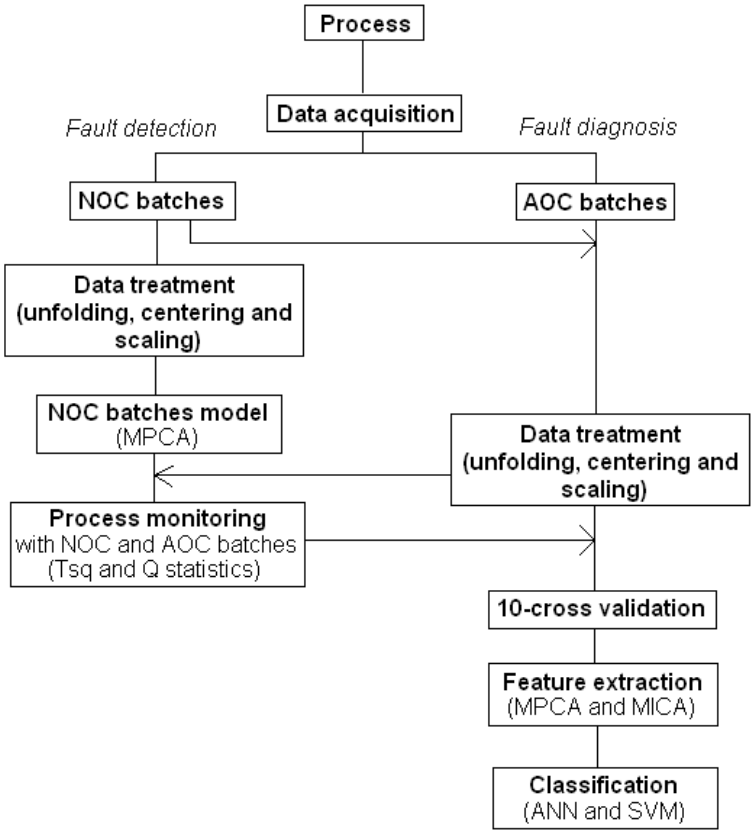

Figure 9.1 Methodology scheme at the end of batch

9.1.2.1 Monitoring step

One way of monitoring either continuous or batch processes is to use Multivariate Statistical Process Control (MSPC). Specifically, the monitoring step of finished batches is applied as explained in section 4.3.5.1. As fault diagnosis is addressed at the end of batch, this would not allow improving the past batch in practice. However, it is noted that several authors have addressed this problem in the past (Yoo et al., 2004a; Nomikos and MacGregor, 1995a; Van Sprang et al., 2002; Yoo et al., 2004b; Chen and Liu, 2002; Camacho and Picó, 2006; Camacho et al., 2009).

9.1.2.2 Fault diagnosis procedure

Once validation batches are monitored by their projection onto the previous NOC model, the next stage is their diagnosis. In order to create generalized classification models, NOC and AOC batches are gathered in the same data set $\mathcal{IR}$ and then, centering and scaling operations are applied to this set, giving as result the $\mathcal{IR}^{*}$ set. Both MPCA and MICA are used for dimensionality reduction before applying the classification algorithms to the $\mathcal{IR}^{**}$ scores matrix.

Furthermore, cross validation is applied in order to minimize the performance variance among the different combinations. A 10-cross validation is applied so that ten models considering both nominal and faulty batches can be tested with their respective validation sets $\mathcal{V}^{**}$. The number of principal components *R* in the models is retained by the broken-stick rule and the independent components are determined by a graphical technique similar to the PCA SCREE test (Jackson, 1991).

Regarding ANN, the number of inputs is automatically set to the number of retained components. The remaining architectural parameter to optimize in ANN is the number of hidden nodes, which is optimized based on the obtained cross-validated classification performance. The number of outputs is the same as the number of faults *F* to classify.

In contrast, by using SVM as classification algorithm, the tunning for each scenario involves the type of kernel function and its parameters (e.g. order for polynomial kernel, width for Gaussian kernel). The *F1* score is used again to assess the diagnosis performance for each class *f* separately.

9.1.3 Results

Figures 9.2 to 9.4 show sets of $I_f=50$ simulation runs for three different scenarios: batches under NOC, batches with fault 2 (increase in the saturation constant), and batches with fault 3 (decrease in the substrate feed rate in the fed-batch stage), all of them with 1% of noise level in input and output data.

Figure 9.2 shows the trajectories of the biomass concentration and illustrates how the decrease of the substrate feed rate (fault 3) affects the biomass production: those flow rates below the initial value for normal operation are insufficient for the biomass growth and probably just for cell maintenance. On the other hand, the culture contamination reflected in an increase of the saturation constant (fault 2) results in a slower production of penicillin and more conversion of substrate to biomass.

Figure 9.3 shows the trajectories of the penicillin production. The decrease in the feed flow rate is shown again to affect the penicillin production. Furthermore, it also shows that an increase in the saturation constant delays the penicillin production. Moreover, Figure 9.4 shows the trajectories of the dissolved oxygen concentration observing that the decrease in the substrate feed flow rate causes higher concentrations

of dissolved oxygen in the culture medium because the amount of biomass is smaller and because of that, the oxygen requirements.

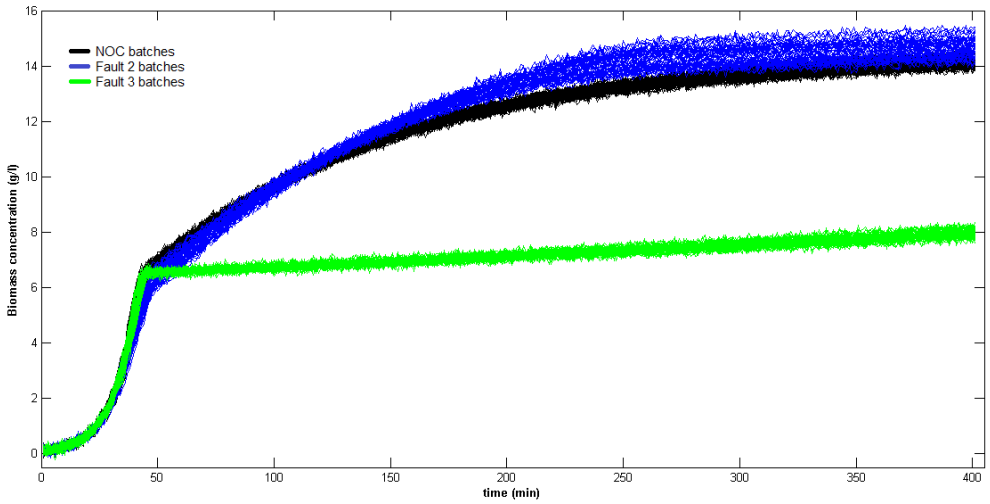

Figure 9.2 Biomass concentration tendency through the process time for some scenarios

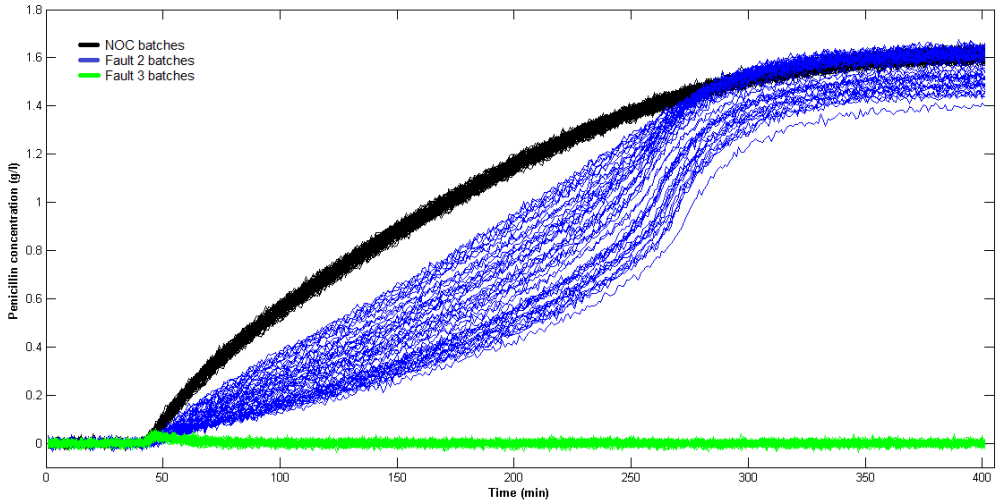

Figure 9.3 Penicillin concentration tendency through the process time for some scenarios

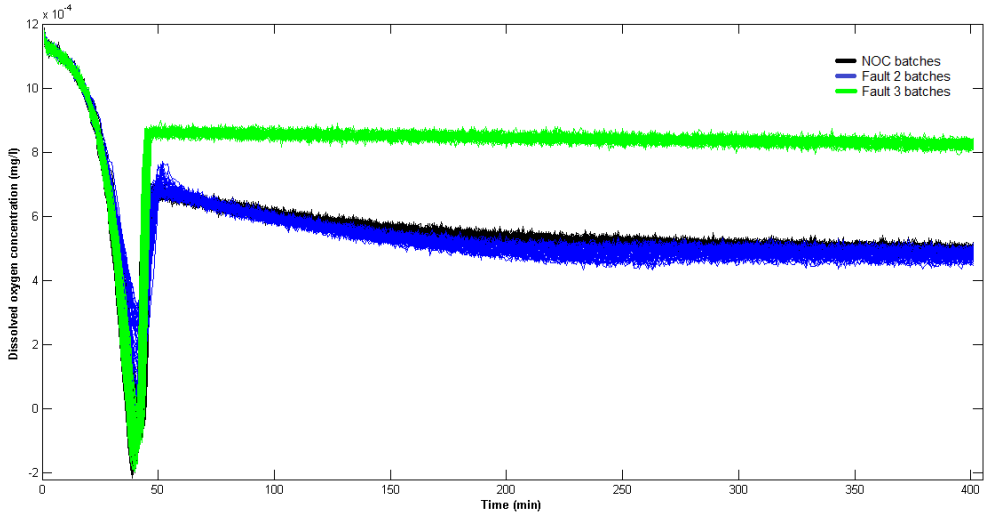

Figure 9.4 Dissolved oxygen concentration tendency through the process time for some scenarios

As previously indicated (MacGregor and Kourti, 1995), more than 50 batches are required for batch process monitoring and for correctly estimating the confidence limits in the normal operating region. Accordingly, 50 AOC batches are taken to construct representative diagnosis models for each fault. The NOC model was just built upon as many NOC batches as faulty batches used in the diagnosis step (100). The high ratio of faulty to normal batches is unrealistic from the point of view of industrial practice, but it does not affect the comparative purposes of the approach. Thus, UPCA or MPCA was performed on hundred batches under NOC (I_0=*100*), simulated with 1% of noise level. Figure 9.5 plots the eigenvalues corresponding to this analysis in logarithmic scale. According to this plot, two components are selected for the monitoring and detection model, which jointly capture 59.5% of the total variance.

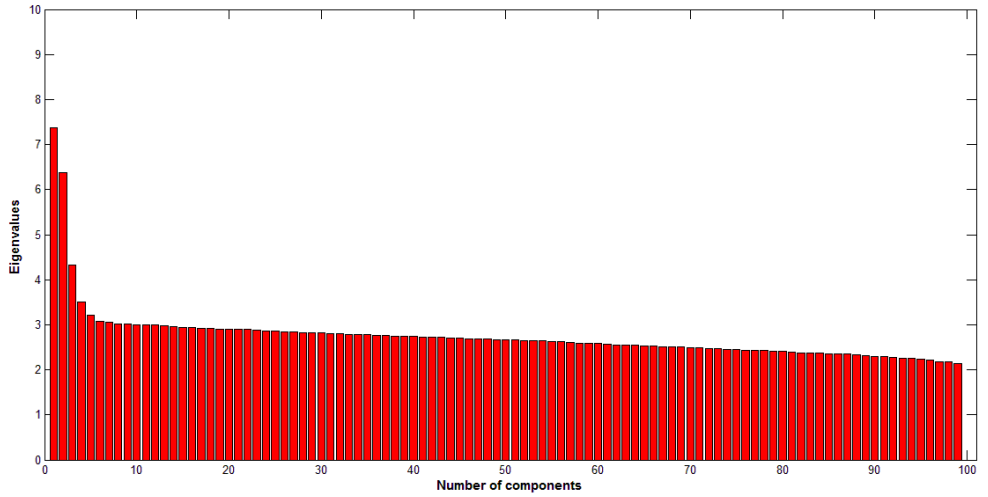

Figure 9.5 Eigenvalues of hundred NOC batches

Figure 9.6 and Figure 9.7 report the *Q* statistic and T^2 values for the NOC batches with 1% of noise level in data and illustrate that both statistics remain below their control limits for all the batches. Similar results are obtained from the analysis of NOC batches in the rest of process scenarios (different noise levels).

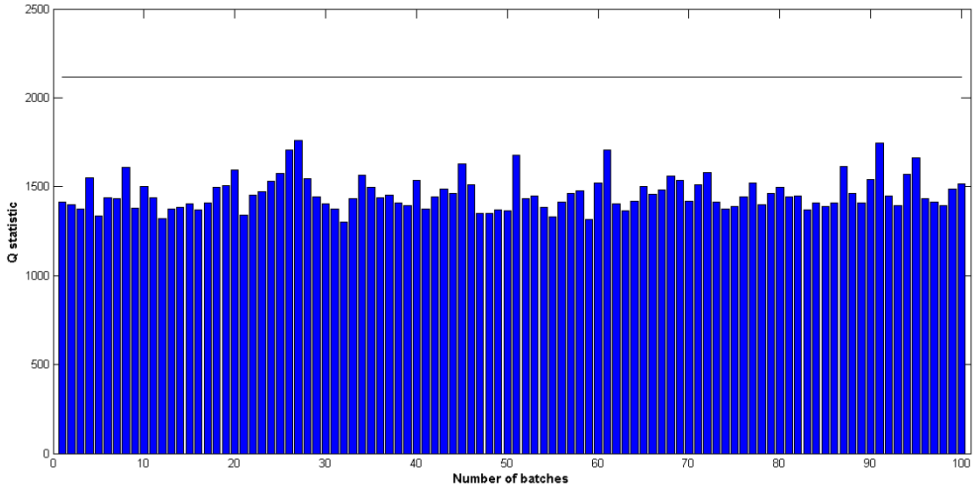

Figure 9.6 Q statistic of penicillin production batches under nominal conditions (1% noise)

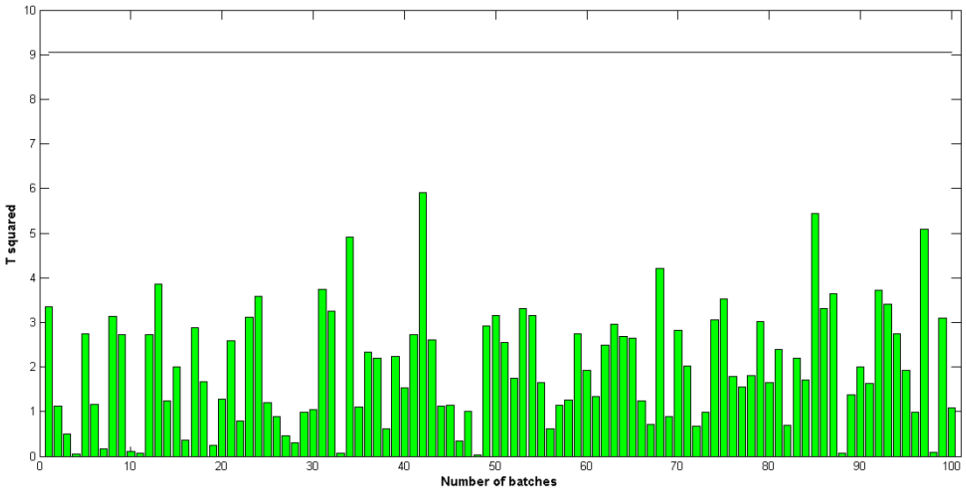

Figure 9.7 T squared statistic of penicillin production batches under nominal conditions (1% noise)

Regarding AOC batches, 50 batches per fault (I_F=*50*) are simulated and monitored. Figure 9.8 shows the plot of the two first scores obtained from the projection of the NOC batches onto the monitoring model, as well as the scores corresponding to the AOC

batches projected onto the same model. This bi-plot shows that batches corresponding to fault 2 and 3 are well separated from the NOC batches and from each other, in contrast to the batches corresponding to fault 1 (decrease in the agitation power) that are found in the same region as the NOC batches.

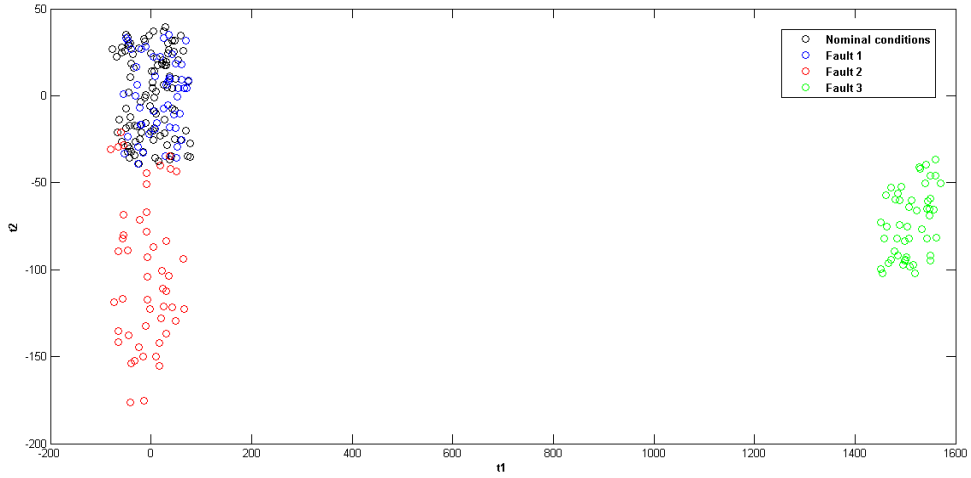

Figure 9.8 Scores plot of penicillin production batches under nominal and faulty conditions

This plot only considers the two first scores out of 100, obtained by the broken-stick rule and representing the 59.5% of variability. Therefore, the other PCs could be expected to allow discerning between NOC and Fault 1 batches. However, an evaluation of bi-plots for some left PC's (not shown) indicated that this is not the case. Fault 1 batches cannot be discriminated from the NOC set because due to two reasons. It cannot be beforehand granted that a faulty situation always results in a distinguishable symptom. Besides, the UPCA model is calibrated only on NOC batches and therefore, not oriented at capturing differences between NOC and AOC batches.

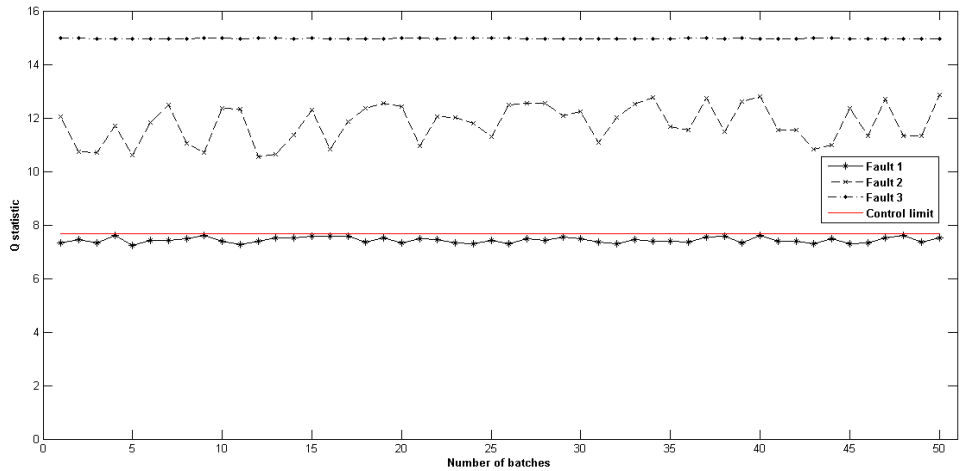

Figure 9.9 Q statistic of the AOC batches

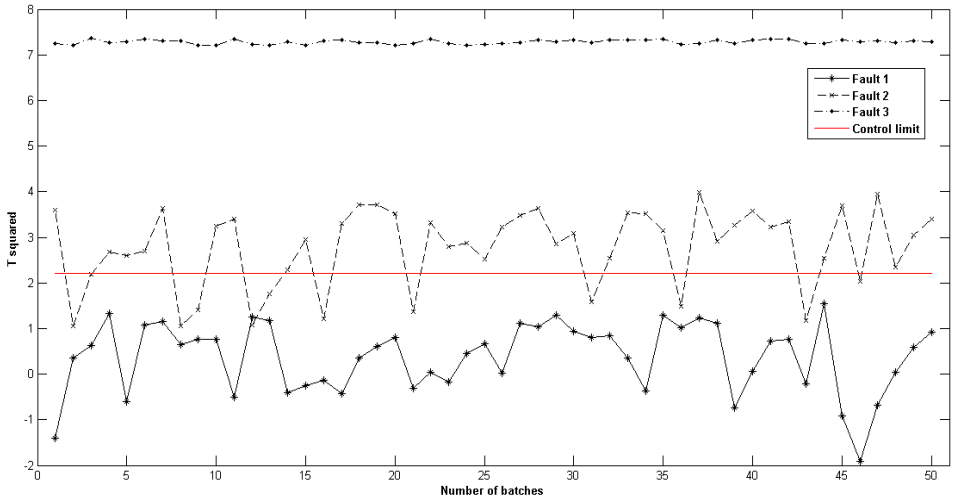

Figure 9.10 Hotelling's T^2 statistic of the AOC batches

Figures 9.9 and 9.10 show the *Q* statistic and T^2 values in logarithmic scale for the batches with the three simulated faults. The confidence level applied for the calculation of the UCL is 99%. As observed, the *Q* and T^2 statistics for the batches with fault 3 are over the UCL. In the case of the batches with fault 2, the *Q* statistic value is over the UCL for all of them but in the case of the T^2 statistic, 12 batches are below the UCL and 38 batches over. In this sense, as AOC batches are not taken into account in the model subspace, the Q statistic detects all the faulty batches as new events. Regarding batches with fault 1, both statistics are below the UCL for all of them, which means that this fault is not detected and all these faulty batches are considered as batches under NOC. A closer inspection of the state variables indicated that their patterns were hardly affected by this fault. As such, fault 1 was ignored and excluded for the diagnosis step that follows. The results showed in Figure 9.9 and Figure 9.10 for the *Q* and T^2 statistics are quite similar to those obtained with the other process scenarios.

The same assumptions for monitoring hold for calibration and validation. Thus, a minimum set of batches is also considered for constructing diagnosis models: 100 AOC batches (50 fault 2 and 50 per fault 3) and 100 NOC batches. The 10-fold venetian blind cross-validation results in selecting 10 normal batches and 5 batches of each faulty condition (total = 20) in a single validation set ($\mathcal{V}^{**} \in \Re^{20\times 401\cdot 9}$).

The feature extraction techniques retain four components (broken-stick rule) for the ten projection models, using both MPCA and MICA and a different level of noise in the input and output variables (0%, 1%, 5% and 10%). Table 9.2 shows the percentage of variance in the retained components after applying both feature extraction techniques and according to the corresponding noise level in data.

Table 9.2 Variability percentage with the retained components by broken-stick rule for both MPCA and MICA techniques and all the noise levels in data

Retained variance (%)				
Feature extraction technique	Noise level (%)			
	0	1	5	10
MPCA	98	86	81	41
MICA	98	86	81	41

The diagnosis step consists of applying properly ANN and SVM as classification algorithms using the scores matrix ($\mathcal{TR}^{**} \in \Re^{180\times 401\cdot 9}$), obtained from MPCA and MICA, as inputs. Regarding ANN, the analysis has been restricted to a single hidden layer network since this is capable of mapping all the data (Huang and Huang, 1991). The number of tangent sigmoid nodes which performed the least mean squared normalized error (MSE) and the best classification performance is reported in Table 9.3 according to the feature extraction technique used and the noise scenario. The networks have two logistic output nodes as the number of faults to classify.

Table 9.3 Number of hidden nodes optimized using both MPCA and MICA with ANN for all the noise scenarios

Number of hidden nodes				
Feature extraction technique	Noise level (%)			
	0	1	5	10
MPCA	1	3	2	1
MICA	14	9	9	7

Regarding SVM as classification technique, Table 9.4 reports the kernel functions that offered the best classification performance after applying MPCA or MICA and for the different scenarios.

Table 9.4 Kernel function optimized using both MPCA and MICA with SVM for all the noise scenarios

Kernel function				
Feature extraction technique	Noise level (%)			
	0	1	5	10
MPCA	Linear	Poly 2	Poly 2	Poly 2
MICA	Poly 3	Linear	Poly 3	Poly 3

The diagnosis results for each combination (MPCA&ANN, MPCA&SVM, MICA&ANN and MICA&SVM) and for each noise scenario (0, 1, 5 and 10%) are summarized in Figures 9.11 to 9.14. The performance was evaluated according to the F1 score as a mean from every time observation per test batch and from the whole batches included in the ten validation sets.

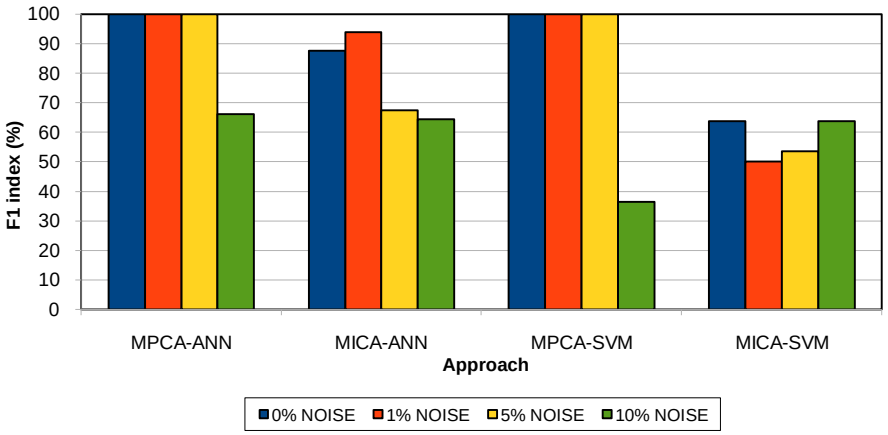

Figure 9.11 Diagnosis performance of the different combined approaches. Mean of all the process scenarios under different noise levels

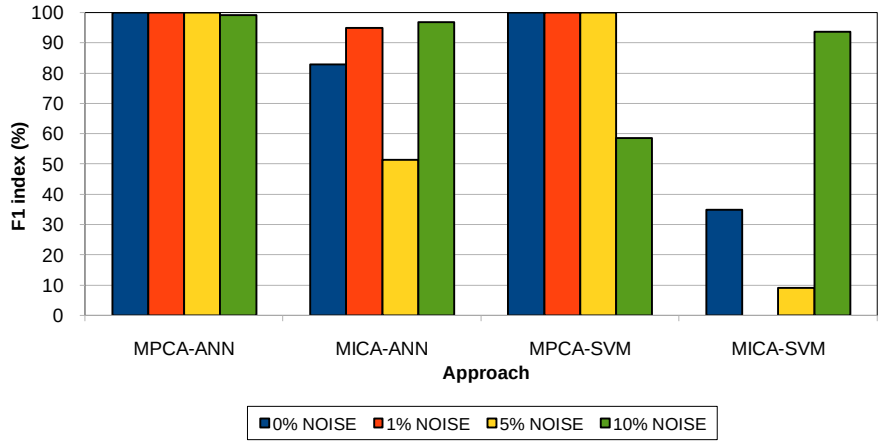

Figure 9.12 Diagnosis performance of the different combined approaches. Faultless scenario under different noise levels

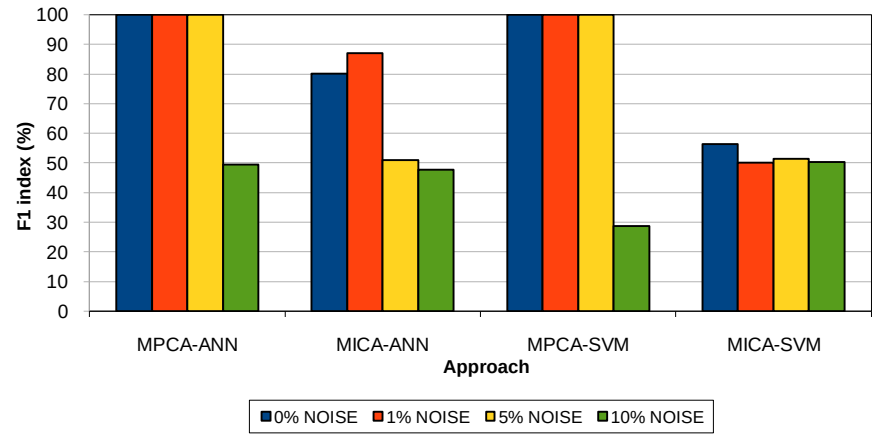

Figure 9.13 Diagnosis performance of the different combined approaches. Fault 2 under different noise levels

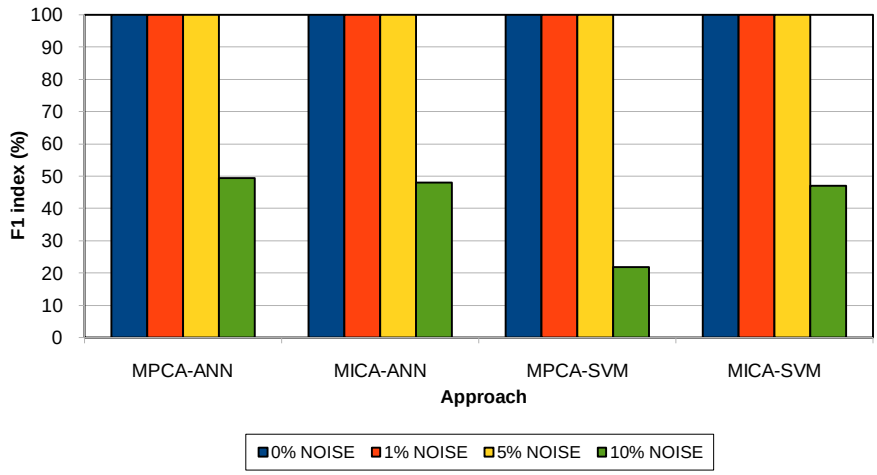

Figure 9.14 Diagnosis performance of the different combined approaches. Fault 3 under different noise levels

The mean diagnosis performance shown in Figure 9.11 indicates that the combination of either ANN or SVM with MPCA performs significantly better than the combinations with MICA. Therefore, the choice and optimization of the feature extraction method is more important than the selection of the classification technique, as illustrated by the four different process scenarios. In the case of the 10% noise scenario, the performance is low in comparison to the rest of scenarios, however it is important to consider that the retained components with the broken-stick rule only explain the 41% of the variance of the process variables, which could be the main reason of such low performance. Although quite reasonable, this is an assumption that should be corroborated by further investigations.

Breaking down the results, Figure 9.12 shows the diagnosis performance for the nominal class (batches under NOC) for the 16 situations and Figures 9.13 and 9.14 do the same for the faults 2 and 3. These results reveal that:

- In general, for all the process scenarios and a given classification method, MPCA leads to better results than MICA. The only exception is found for the 10% noise scenario, for which the faults are poorly classified by all the combined approaches.
- In general, for the four combined approaches, higher noise levels lead to worse diagnosis performance, which it is not surprising. There is an exception however in the normal class, which shows apparently a good performance in the highest noise scenario when using MICA.
- Fault 3 is well diagnosed no matter the combination of feature extraction and classification techniques used. Again, the exception is the 10% noise scenario, probably because of the high noise in data and low variance in the extracted components.
- MICA performs better when it is combined with ANN rather than with SVM.

- MPCA plus SVM combination seems to be more affected by either high noise levels or low variance percentages explained by the retained principal components.

Both the expected and unexpected results are next discussed.

9.1.4 Discussions about the FD approach at the end of batch

Results support the benefits of the step-wise procedure (feature extraction plus fault diagnosis) in terms of the final diagnosis performance. The application of a feature extraction technique allows dimensionality reduction and obtaining features that summarize the information in the data. Such features retain a given percentage of the variance of the original variables. Moreover, as these feature extraction techniques are multivariate statistical techniques, they allow monitoring new batches in order to know whether they are successful or faulty previously to their diagnosis. Once the diagnosis models have been constructed with historical batches and the appropriate techniques, the diagnosis of new and current batches becomes an easier task.

The comparison between feature extraction and classification techniques has been performed with different noise scenarios (0%, 1%, 5% and 10%). The combinations of both classification techniques used with MICA allow concluding that the best approaches are those in which MPCA is applied as feature extraction no matter the classification technique used for diagnosis. Also the results indicate that more effort should be devoted to the feature extraction technique rather than to the diagnosis algorithm.

The MICA&ANN combination works well without noise or with low noise in data (1%), but performance decreases when noise increases (5%). In fact, batches with 1% of noise show the best diagnosis performance, which can be explained by the 86% of the retained variance with the PCs in comparison to the 81% with 5% of noise. However, retaining less variance below a certain upper threshold may sometimes improve the diagnosis, rather than retaining a higher variance, which can explain why the diagnosis performance is better in the case with slight noise in the data (1%) than without noise.

In faultless batches, MICA&SVM combination seems to be more affected by noise, although the 10% scenario shows an unexpected behavior. The predictions obtained from this combined approach show that the reason for a high diagnosis performance is that faulty batches are simultaneously diagnosed as occurring both faults. Therefore, the incapacity of this approach to distinguish faults is revealed.

The sensitivity part in the F1 score accounts for the right diagnosed batches divided by the total number of batches, and this is therefore calculated taking into account half of a batch when it is double-classified. One third of a batch would be counted in the case of three simultaneous classes and so on. In the same way, specificity counts the right diagnosed batches divided only by the number of batches diagnosed in each class. This is the reason why the F1 score reflects such modifications in the final performance.

According to these results, there are no significant differences between ANN and SVM when either of these are combined with MPCA, which indicates that the decision-making should be concentrated on the feature extraction technique rather than on the classification technique. In general, these results can be useful for diagnosing other processes.

For other highly non-linear processes it would be interesting to simulate some different faults as well as to increase the number of NOC historical batches for constructing the diagnosis models. It would be also worthy to investigate other types of scaling such as group scaling before selecting and applying the feature extraction technique and to consider a specific percentage of variance in the retained components as in the continuous process final approach.

9.2 On-line diagnosis

This approach develops the diagnosis of individual batches at each sampled time *k*, which represents an on-line diagnosis. In order to do so, the historical batches used to construct the diagnosis models as well as the current batches to diagnose are variable-wise unfolded (*IK*×*J*). It is worthy to remember that one drawback of the batch-wise unfolding is that the method assumes that all the observations for the complete batch are available and the future observations must be estimated for on-line monitoring.

Due to this, variable-wise unfolding ($X^{IK\times J}$) was proposed for overcoming the issue of having to get the samples for the full batch in the batch-wise unfolding (Wold et al., 1998). In addition, Westerhuis et al. (1999) reported that one may include as many principal components as original process variables in some cases when applying variable-wise unfolding.

However, as the variable-wise unfolded batches are mean-centered and auto-scaled, there will be a high variance in each observation per variable (column of the training set) due to all the the time trajectory counted in the same column. In addition, when applying a feature extraction method, the retained components will explain just the variability and correlation among process variables.

This new issue brought up the use of multi-model latent methods such as multi-model PCA. However, due to the computational cost of creating, uploading and applying several models for on-line diagnosing a current batch, an alternative data treatment was proposed. Such treatment implies a batch-wise centering and scaling and then a data rearrangement to the variable-wise unfolding before applying the feature extraction and the classification methods, which should perform better than directly centering and scaling to the variable-wise unfolded information. On the other hand, the machine learning algorithms are applied in the same way as in continuous processes and one can have the choice of validating a number of samples after a time window or one to one at each sampling time.

9.2.1 Case study and methods

The penicillin production process formulated in the previous approach is again used as case study. Three different faults are simulated so that they can be detected and diagnosed:

1. Decrease in the agitation power to values between 0.1 and 1 W
2. Increase of the saturation constant (K_x) to values ranging from 0.3 to 0.9 g/l and
3. Decrease of the substrate feed rate in the fed-batch stage to values from 0.001 to 0.1 l/h.

The only difference with the faults simulated and considered in the last approach is regarding to the decreasing of the agitation power to smaller values (fault 1). Noise was also added to the input and output variables in the model at 1% level in the same way as in the previous approach. Finally, the resulting matrix per batch run has also the same dimensions 401×9.

9.2.1.1 MPCA, BDPCA and SVM

As it has been stated and reported in this thesis, there are plenty of techniques to address the process monitoring. However, most of the times the existing methods are neither critically evaluated in terms of the guidelines development and their different applications nor compared under the same conditions. Therefore, considering the previous results obtained from the applied monitoring methods, Multi-way Principal Component Analysis (MPCA) and Batch Dynamic Principal Component Analysis (BDPCA) are applied and compared in this approach.

The main reasons to choose these methods is that MPCA was the first extension of PCA adapted to batch processes and is relatively simple and available in some commercial software packages. On the other hand, BDPCA brought up the concept of applying a dynamical and improved version of MPCA instead of developing multi-model PCA. Both MPCA and BDPCA methods were widely introduced in chapter four.

Regarding to the monitoring with variable-wise MPCA, the well-known statistical indices (T^2 and Q) are calculated as in PCA for continuous processes as well as their control limits (CL) so as to detect any disturbance during the batch. The Q statistic and its control limit is calculated as in Eq 4.7 and 4.8. The main difference with continuous processes is that the Q statistic has a different control limit (Q_α) at each sampled observation k. The T^2 index is calculated as in Eq 4.9., and its corresponding control limit is calculated as follows:

$$T_{lim}^2 = \frac{R(IK+1)(IK-1)}{IK^2 - IKR} F_{\alpha,R,IK-R} \qquad \textit{Eq 9.19}$$

BDPCA is developed as exactly explained in section 4.3.7. Before applying the BDPCA method, the centering and scaling is applied in the next two ways for comparative purposes:

- Variable-wise unfolding and rearrangement to the dynamic structure plus mean-centering and auto-scaling.
- Centering and scaling are applied to the batch-wise unfolded matrix and then the resulting matrix is rearranged to the variable-wise dynamic structure [I·(K-d)×J(d+1)].

Also, the both previous standardization ways are developed before applying MPCA. On the other hand, the fault diagnosis step is covered using SVM as classification method. At this point, the application of this method is equivalent to its application to continuous processes either in an off-line or in an on-line way.

9.2.2 Validation of the approach and results

The validation of the approach will be divided in two sections. The first one corresponds to the batches monitoring and fault detection (if it is the case) and the second one corresponds to the diagnosis of the faulty batches.

9.2.2.1 Monitoring and detection

Fifty batches under NOC ($I_0=50$) are used to construct the monitoring model using either variable-wise MPCA or BDPCA. The two types of standardization (batch-wise and variable-wise centering and scaling) are used for both methods. The monitoring success is evaluated by projecting test batches under NOC and AOC onto the PCA model. Five batches per class (faulty and fault-less), twenty in total, are monitored with the projection model. Regarding BDPCA, the validation batches are also arranged in the dynamic way. T^2 and Q statistical indices are calculated and inspected whether or not overstep their corresponding CL.

Figure 9.15 and Figure 9.16 show the T^2 of one validation batch per class with the two centering and scaling arrangements and applying BDPCA. In the same way, Figure 9.17 and Figure 9.18 show the T^2 for the same batches with both centering and scaling arrangements and applying MPCA.

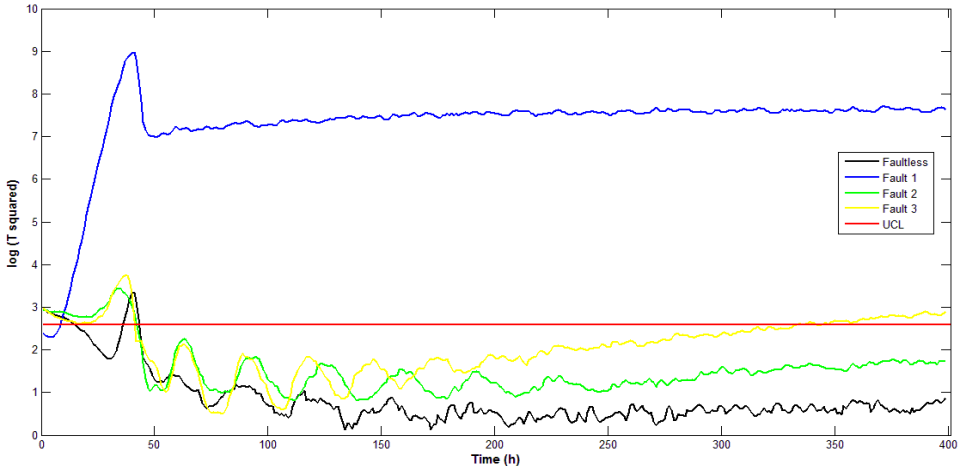

Figure 9.15 Hotelling T^2 applying BDPCA and variable-wise centering and scaling

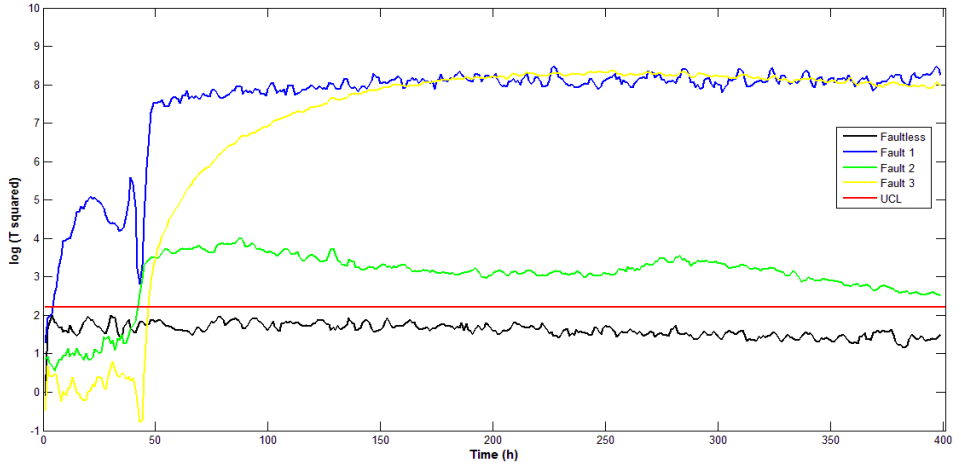

Figure 9.16 Hotelling T^2 applying BDPCA and batch-wise centering and scaling

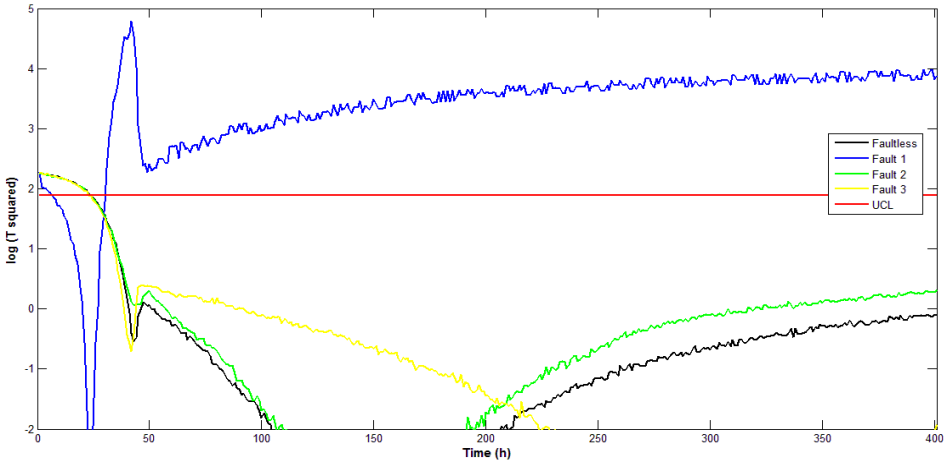

Figure 9.17 Hotelling T^2 applying MPCA and variable-wise centering and scaling

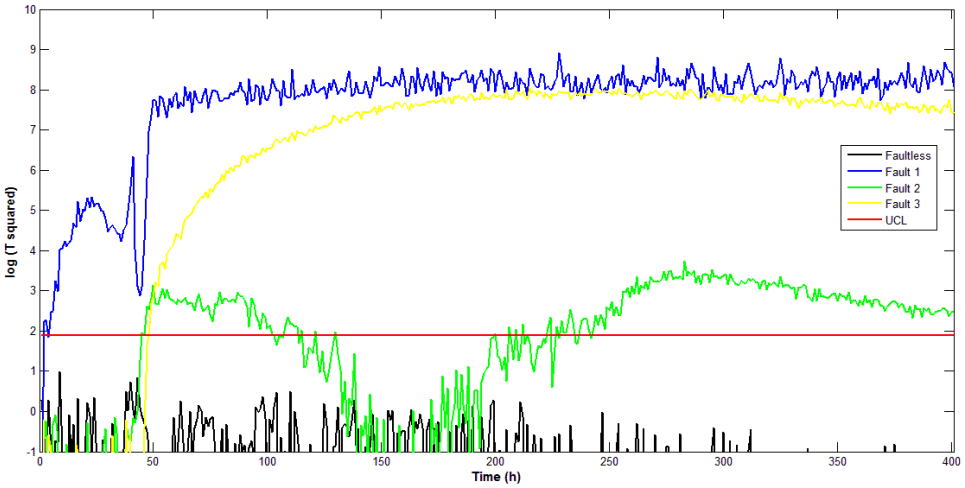

Figure 9.18 Hotelling T² applying MPCA and batch-wise centering and scaling

As it is observed in the T^2 statistic schemes, the centering and scaling to the batch-wise arrangement contributes to the right monitoring of the batches and to the right fault detection applying either MPCA or BDPCA to the variable-wise rearranged data matrix. Also a right monitoring of the validations batches is accomplished by using the Q statistic, whose plots are saved due to the similar behavior to the T^2 statistic plots.

Therefore, it is recommended to use batch-wise centering and scaling for batch process monitoring and apply either MPCA or BDPCA to the variable-wise standardized data. At this point, both techniques are equivalent in terms of monitoring and fault detection.

One last important thing to point out is that in variable-wise centering and scaling plots (with both statistical indices and both methods), a peak is evidenced approximately in the fortieth hour, which corresponds to the depletion of glucose and the process change to the fed-batch stage. This behavior can bring up the idea of constructing two separate monitoring models, one for the batch stage (up to 44 h) and another for the fed-batch stage (from 45 to 400 h).

Figure 9.23 and Figure 9.24 show the T^2 index as result of applying variable-wise centering and scaling and BDPCA to the batch and fed-batch stages separately. Analyzing these plots, one can conclude that the batch-wise centering and scaling renders a better performance than the construction of two models with variable-wise standardization, which in fact would correspond to the application of multi-model MPCA.

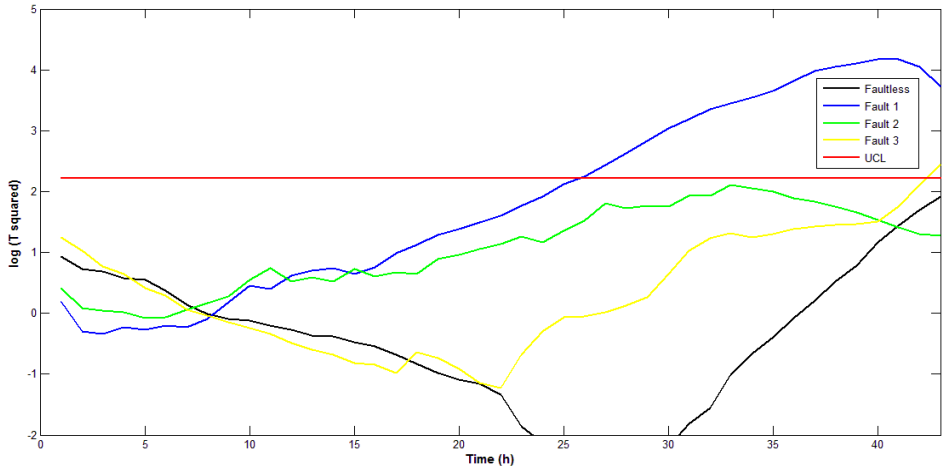

Figure 9.19 T^2 using BDPCA and variable-wise centering and scaling to the batch stage

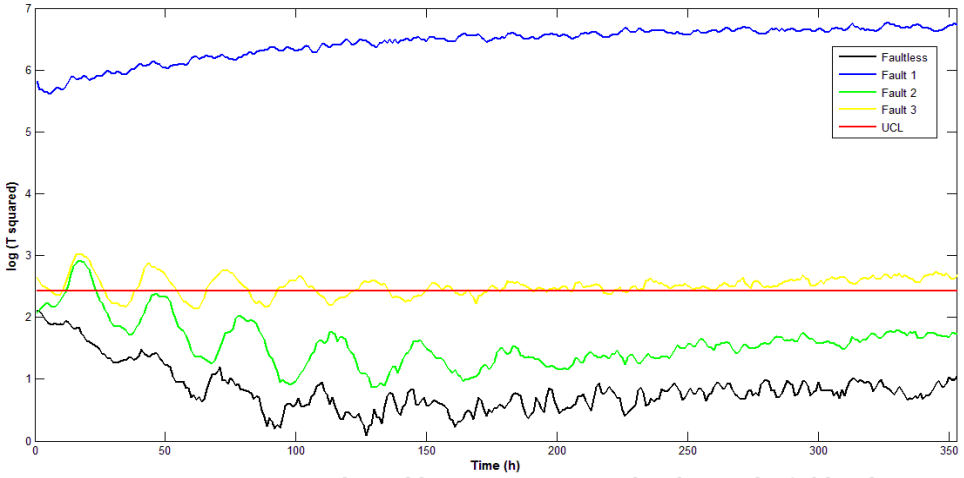

Figure 9.20 T^2 using BDPCA and variable-wise centering and scaling to the fed-batch stage

9.2.2.2 Diagnosis

Fifty batches per class under NOC and AOC (I_f=50), 200 batches in total, are used to construct the diagnosis models as training data ($\mathfrak{TR} \in \Re^{200 \cdot 401 \times 9}$), using the two ways of centering and scaling, rearranging the data to the variable-wise unfolding and applying both MPCA and BDPCA (four scenarios). The score matrices (TR_f^{**}), obtained by projecting the historical batches onto the projection models are used as input of the diagnosis algorithm. SVM are again applied for this step. Ten batches per class (40 in total) represent the validation set ($\mathcal{V}$), and are used exactly to validate such models and evaluate the diagnosis performance in terms of the *F1* score.

Table 9.5 shows the diagnosis performance as a mean of the forty batches and all the observations during their duration. Also, the kernel type that rendered the best

diagnosis performance for each scenario is reported. In this Table 9.5, the forty validation batches are joined in only one validation set $\mathcal{V}$ (as an off-line diagnosis), which allows evaluating the diagnosis models and obtaining a general performance that considers the wrongly-diagnosed batches in a different class and lets tune the parameters of the diagnosis algorithm and obtain the best type of kernel.

Table 9.5 Fault diagnosis performance in percentages with the F1 score using both ways of pre-processing and either MPCA or BDPCA

CLASS	Variable-wise centering and scaling		Batch-wise centering and scaling	
	MPCA	BDPCA	MPCA	BDPCA
Fault-less	67.7	80.5	52.3	80.5
Fault 1	39.6	100	0	100
Fault 2	39.2	50.5	0	64.6
Fault 3	85.5	78.1	97.9	98.6
Mean	58.0	77.3	37.5	85.9
Kernel function	Poly 3^{rd}	Poly 4^{th}	Poly 2^{nd}	Linear

The information in the previous table allows to conclude first that BDPCA is better than MPCA for fault diagnosis in batch processes and second, that the batch-wise centering and scaling is the best way of data pre-processing as in the off-line approach that unfolds batch-wise. This is coherent because in this way, the variance between each sampled observation and process variable is kept and explained by the retained principal components. In this sense, it is worthy to mention that despite variable-wise centering and scaling with MPCA performs better performance than the batch-wise standardization, it is found that the observations of the faulty batches are doubled-diagnosed with two faults, which makes increasing the mean performance. Also with BDPCA and the variable-wise standardization, few samples are simultaneously diagnosed too, which confirms the previous statement.

Therefore, a batch-wise data standardization previous to applying BDPCA renders a good performance for the on-line fault diagnosis in batch processes. In this sense, Table 9.6 presents the results for this scenario when it is applied to each validation batch per separate in an on-line way and in real practice. The given percentage in the mean column per class is not only an average of the 10 batches, but also of all the performances obtained sample by sample during one batch. Thus, a better fault diagnosis is obtained in this way (92.4% in average), which is closer to the 100% obtained with the approach at the end of the batch and applying batch-wise unfolding, MPCA-SVM and 1% of noise level in the input and output data. This comparison suggests to use MPCA for off-line diagnosis and BDPCA for on-line diagnosis of batch processes.

Table 9.6 Diagnosis performance obtained by evaluating each validation batch per separate in an on-line way.

Class	Validation batches										**Mean**
Fault-less	100	100	100	100	100	100	100	100	100	100	**100**
Fault1	100	100	100	100	100	99.9	100	100	100	100	**100**
Fault2	80.5	61.2	53.0	75.5	55.4	79.1	83.8	69.8	64.1	84.0	**70.7**
Fault3	98.9	99.3	98.9	99.4	98.7	98.0	99.4	99.0	99.2	98.9	**99.0**

9.3 Comparison of the diagnosis performance from the proposed approaches with reported performances of the penicillin production process

Most of the papers found in literature deal with the process monitoring and fault detection of batch processes rather than with fault diagnosis and many of them uses the penicillin production process as case study. Lee et al. (2004b), Ündey et al. (2004) and Yoo et al. (2004a) simulate the decrease of the substrate feed flow as fault, either as step fault or ramp fault and use MSPM techniques such as MPCA, MICA or MKPCA for its detection. They report the detection of the fault by using the already known statistical indices.

In addition, Lee et al. (2004a) monitor four type of faults. Besides the step and ramp decrease of the substrate feed rate, they simulate a pH controller failure and also the 15% step decrease in agitation power by using on-line variable-wise MPCA and obtaining better results than with the batch-wise MPCA and predicting the future observations, required for its on-line application.

In contrast, Li and Cui (2009) apply KFDA to the fault diagnosis in the penicillin production process. They use 60 fault batches in total, from which 50 are faulty, for validating their approach and apply a five cross-validation for generalizing results. However, they do not report the type of faults simulated. Their results are reported in terms of the recognition rate as shown in Table 9.7

Table 9.7 Recognition rate of different cases reported by Li and Cui (2009)

Kernel parameters	Kernel function	Cases (%)					Average (%)
		1	2	3	4	5	
First order	Polynomial	66.7	91.7	83.3	75.0	83.3	80.0
	Cosine	66.7	75.0	100	100	100	83.3
Second order	Polynomial	66.7	91.7	75.0	83.3	83.3	80.0
	Cosine	58.3	91.7	100	91.7	100	83.3
Third order	Polynomial	66.7	91.7	83.3	83.3	83.3	81.7
	Cosine	58.3	91.7	100	100	100	90.0

Apart from this work on fault diagnosis in batch processes, few efforts in research have been dedicated to deal with this issue as it has been developed in this chapter.

9.4 Conclusions of the fault diagnosis on batch processes

This chapter has presented two fault diagnosis approaches for batch processes according to the moment in which the batch is monitored and diagnosed, either at the end in order to discard it from the rest of batches or during its run in an on-line manner. Both approaches are developed and applied on a benchmark fermentation process. Regarding the approach at the end of the batch, different available techniques were applied with the aim to provide guidelines for the general problem of selecting data-based methods for

modelling and diagnosing batch processes. Then, a comparative study is developed, which focuses on the selection of a practical combination of feature extraction and classification techniques that can be of general use for fault diagnosis of highly non-linear batch processes.

Two feature extraction techniques, MPCA and MICA, and the two well-known non-linear classification algorithms used in this thesis, ANN and SVM, were used for this purpose. As such, four combinations were evaluated, respectively tagged as MPCA-ANN, MPCA-SVM, MICA-ANN and MICA-SVM. The previous data representation and pre-processing steps consists of unfolding the data batch-wise, centering and scaling, which are typically applied when monitoring batches. The feature extraction techniques (MPCA, MICA) were effective at reducing the dimensionality of the inputs to the classification algorithms (ANN, SVM). The ANN number of hidden nodes and the SVM kernel function were optimized to improve directly the classification performance.

The four resulting approaches were tested and assessed for several scenarios, based on the noise level introduced in the data, in order to generalize the study regarding to the combination between the proposed techniques. The results obtained allow concluding that regarding fault diagnosis, the selection of the classification method is not as decisive as the choice of the feature extraction technique, which is the issue to be stressed in the design of a fault diagnosis approach with the available techniques. This is demonstrated in all the scenarios tested in the current study.

The conclusion from the off-line approach regarding to stress the choice of the feature extraction method rather than the classifier is taking into account for the on-line approach. In this approach, MPCA and BDPCA are applied and compared as monitoring and feature extraction techniques and SVM used as classification algorithm. The data is variable-wise unfolded in both cases so that a diagnosis per sampled time can be developed. Besides, two types of centering and scaling are tested: batch-wise and variable-wise, no matter that the resulting data is then re-arranged in the variable-wise unfolding before applying the techniques.

Therefore, it was confirmed that a batch-wise centering and scaling should be applied in on-line batch fault detection and diagnosis. For monitoring purposes, either MPCA or BDPCA can be used because both of them render equivalent results. However, for diagnosis purposes, BDPCA showed a significantly better performance than MPCA, which allows concluding to use the dynamic version of latent methods rather than the static and original versions in on-line batch process diagnosis.

Chapter 10
ANOMALY DETECTION ON BATCH PROCESSES

Anomaly Detection was introduced in chapter seven as a complementary stage of the general FDS because it faces the main limitation of the data-based diagnosis models constructed with supervised learning algorithms. As already known and discussed, such algorithms train the different patterns of data in a process, named as classes, and develop diagnosis models which are not prepared to identify and diagnose faults not included in there. The AD methodology proposed in this thesis allows diagnosing this kind of novel faults, by combining simultaneously a detection stage with a diagnosis stage applying the original classification algorithms used for diagnosis.

As the AD methodology involves both binary and multi-class classification, it requires previous data representation and improvement before applying properly the classification algorithm. The methodology is applied on two case studies: a lab heat exchange system and a Photo-Fenton pilot plant. Both case studies operate batchwise and were designed to test the methodology using experimental data from physical measurements. The corresponding applications are developed and discussed in the next two sections, as well as the steps before constructing the classification models. In both applications, SVM algorithm is selected as the binary and multi-class classifier.

10.1 Application of the AD method to a lab batch-wise heat exchanger

In the first AD application, a simple pilot plant case study is addressed (Figure 10.1). A heat exchange system melting a batch of ice illustrates a recipe for producing a certain amount of product. The nominal case is the heat exchange between a water stream at room temperature and a bath with water and ice. Hence, the bath temperature above 1ºC is easily established as the recipe final condition. The process variables (stream flow rates and temperatures -both inlet and outlet) were measured at one second intervals (t_s=*1s*). Figure 10.2 shows the variables behavior through the time in the nominal case. Four different faults were run with the system:

- External heating of the cold water (*F1*),
- Failure of the cold stream inlet temperature monitoring (*F2*),
- Failure of the cold stream flow monitoring (*F3*) and

- Simultaneous heating of the bath and failure of the cold stream inlet temperature monitoring (*F4*).

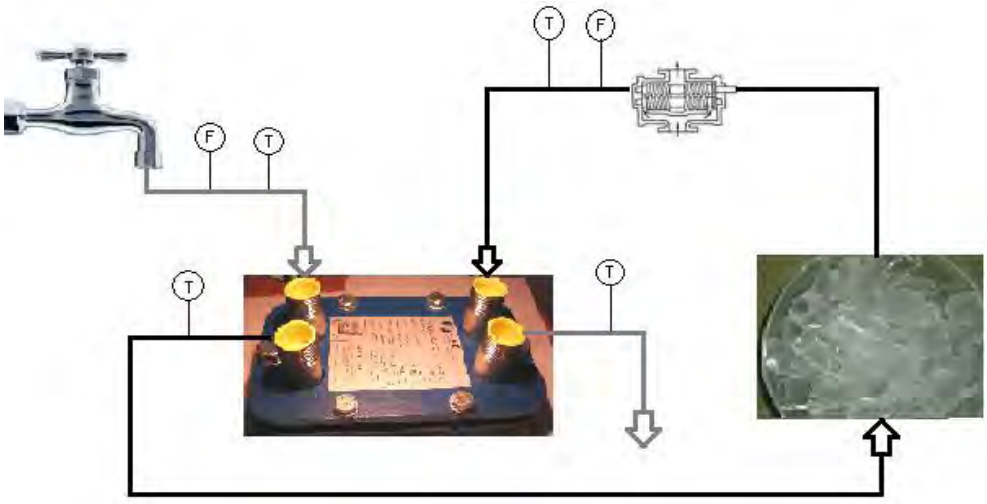

Figure 10.1 Batch wise heat exchange system as case study

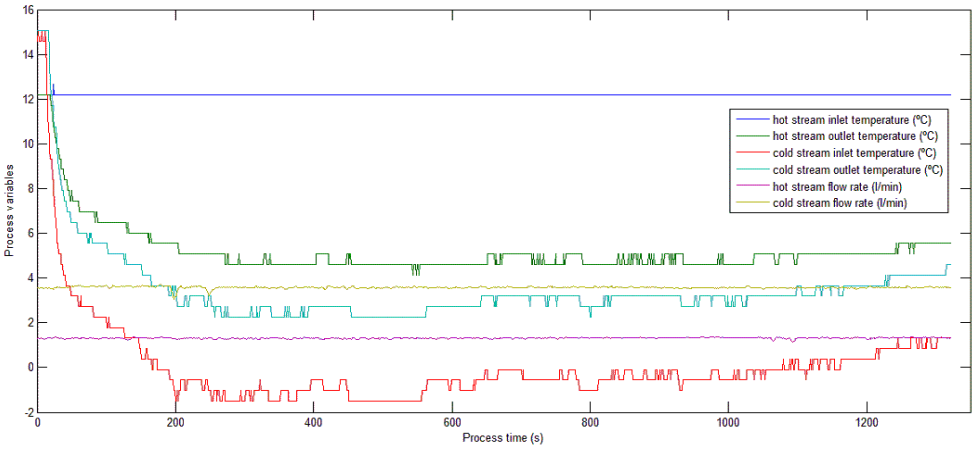

Figure 10.2 Process variables behavior through the time in the batch heat exchange system

Five batches under NOC and three batches per fault (*I=22*) were run and variable-wise unfolded. The random data approach proposed for continuous processes was applied. Therefore two data sets, training ($\mathfrak{IR}$) and test ($\mathfrak{IS}$), were created from randomly selected data with the same magnitude of rows in both sets.

A feature extension step is applied as feature processing. The extended features are the standard deviations and linear trends (slopes) of the data within a time window of 20s. The AD method is developed using only 4 classes for building the binary and multi-class classification models (3 faults and the class 0) because the fourth fault was taken as

the novel fault not registered before. The results obtained are presented and discussed in the next subsection.

10.1.1 Validation and results

Before applying the AD methodology, the FD approach with finished batches and randomly selected observations is applied. SVM with linear kernel are applied to data from the 5 classes. Both $\mathfrak{IR}$ and $\mathfrak{II}$ sets are composed by 2275 samples (455 per class). Table 10.1 shows the diagnosis results obtained.

Table 10.1 F1 score of the FDS using linear kernel for the original and extended features

Features	F1 score (%)					
	Class0	Class1	Class2	Class3	Class4	Mean
Original	**99.8**	**100**	64.4	**99.7**	90.1	88.6
Orig+Std. dev.	99.4	**100**	81.5	**99.7**	98.5	94.9
Orig+Slopes	81.0	**100**	37.6	**99.7**	**98.8**	84.0
Orig-Std dev-Slopes	99.4	**100**	**96.0**	**99.7**	93.6	**97.3**

The four faults are diagnosed with good performance, especially when using two feature extension steps as it is observed in Table 10.1. Then, new training sets composed of 2264 samples and 4 classes (566 samples per class) are used for assessing the performance of the AD methodology. These classes correspond to the class 0 and 3 faults. The test sets change according to the fault to detecting and diagnosing. These sets are thus:

- Test set containing samples of class 0 (2264 samples)
- Test set containing samples of fault 1 (565 samples)
- Test set containing samples of fault 2 (1428 samples)
- Test set containing samples of fault 3 (568 samples)
- Test set containing samples of the novel fault not included in the diagnosis models (fault 4, 910 samples).

Table 10.2 and 10.3 show the results for all these cases. The novel fault to detect is fault 4 and its detection and classification is reported in the last row of both tables.

Table 10.2 F1 score for the AD method using a training set with class 0-3 information and test sets with information of only one class. Binary classification stage.

Test set data	F1 index (%)		
	Class 0	Fault	Mean
Class 0 data	100	0	100
Class 1 data	0	99.8	99.8
Class 2 data	0	99.9	99.9
Class 3 data	0	100	100
Fault 4 data	0	100	100

Table 10.3 F1 score for the AD method using a training set with class 0-3 information and test sets with information of only one class. Multi-class classification stage

Test set data	F1 score (%)				
	Class 0	Class 1	Class 2	Class 3	Mean
Class 0 data	**100**	0	0	0	100
Class 1 data	0	**100**	0	0	100
Class 2 data	0	0	**99.4**	0	100
Class 3 data	99.7	0	0	**100**	100
Class 0-3 data	**99.4**	**100**	**99.4**	**100**	**99.8**
Fault 4 data	**100**	0	0	0	100

The results of the 4 first classes (0-3) confirm the good performance of the Fault Detection and Diagnosis System (FDDS) and the last row shows the success of the AD method. As it is observed, fault 4 is simultaneously detected as fault and diagnosed as none of the modeled faults, but a novel one.

The proposed AD methodology has been applied to a lab batch wise case study and it has been demonstrated the method efficiency at 100%. Although this case study is a simple case study, the results motivate to extend the AD application to a bigger case study. Therefore, it will be applied to a pilot plant in order to scale the application and devise whether or not promising results can be expected from the method.

10.2 Application of the AD method to a Photo-Fenton process

Photo-Fenton process is an Advanced Oxidation Process (AOP) used for remediating wastewaters which contain recalcitrant contaminants. It consists of the in-situ generation of strongly oxidant hydroxyl radicals ($OH^{\cdot}$) from the reaction between H_2O_2 with an iron salt in presence of UV radiation. The final remediation achieved extremely depends on the process variables, especially those concerning to the dosing reagents and light intensity and wavelength (Pignatello et al., 2006).

In the Photo-Fenton reaction, Fe^{3+} is excited when it is irradiated between 300 and 650 nm wavelengths, which increases the reaction velocity of the Fenton process and produces more hydroxyl radicals as it is shown in the next equation.

$$Fe^{3+} + H_2O \xrightarrow{UV} Fe^{2+} + H^+ + \cdot OH \qquad Eq\ 10.1$$

The reactions between Fe^{2+} and Fe^{3+} with H_2O_2 allow the mineralization of the organic matter into CO_2 and H_2O (Tokomura et al. 2008). It is favorable and profitable that iron salts are the catalyst of the reaction due to its abundance in the environment. Besides, it is well known that a pH range of the Fenton reaction must be between 2 and 4 in order to produce the highest quantity of hydroxyl radicals, allowing the oxidation of the organic matter. Furthermore, it has been reported that the ratio between Fe^{+2}/H_2O_2 affects the rate reaction (Gulkaya et al. 2006).

10.2.1 Description of the pilot plant

The pilot plant, shown in Figure 10.3, is composed of two reactors and a pump system of 0.5 HP between them that allows the flow recirculation through the reactors. Both reactors can work in series or independently. The main reactor is a 12 L jacketed glass reactor including in the middle of it a medium pressure Hg lamp of 700 W, capable of irradiating between 390 and 410 nm. In contrast, the second reactor is a 2L tubular reactor with a 55 W low pressure Hg lamp irradiating at 254 nm in a quartz lampholder. After the tubular reactor, there is a motorized valve that deviates a small part of the flow to the measurement line.

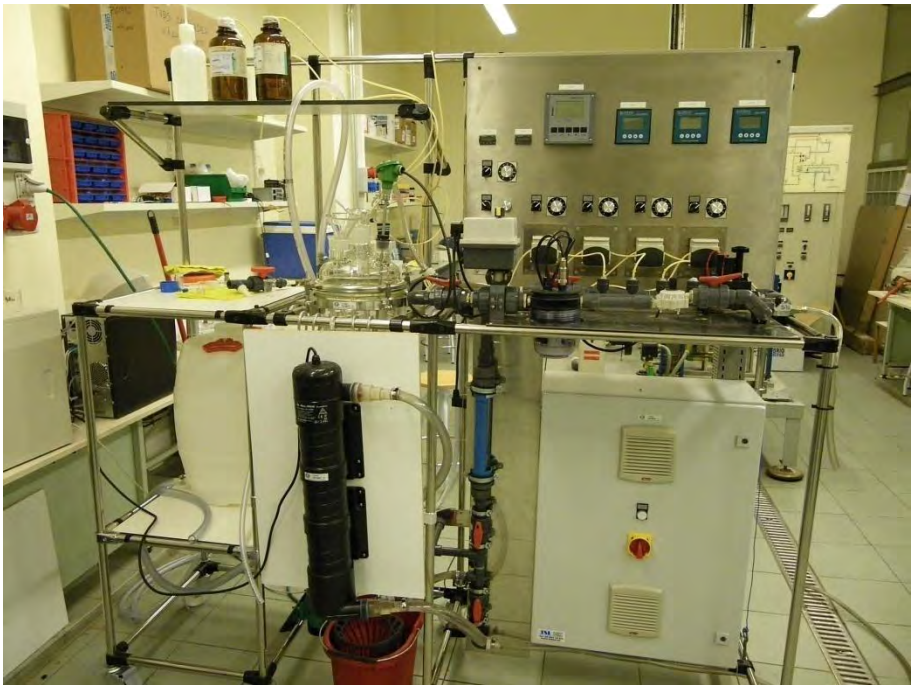

Figure 10.3 Photo-Fenton pilot plant at EUETIB-UPC

Thus, the plant is monitored with an automatic control and data acquisition system, in which eight process variables can be on-line registered: dissolved oxygen (*DO*), redox potential (*ORP*), pH, conductivity, main reactor temperature (T_1), in-line temperature (T_2), recirculation flow (Q_1) and measurement line flow (Q_2). The sampling time at which the measurements are registered and stored in the data acquisition system is 1s. The recirculation flow is controlled by the pump and the measurement line flow is controlled by the motorized valve with PID control. Also, pH is controlled with PID control.

The flexible pilot plant could then be configured in several ways affecting volume and irradiation source as indicated in Table 10.4. Moreover different contaminants are also possible to treat with Photo-Fenton. Currently, the treated contaminants in the photo-Fenton plant at the EUETIB campus from UPC are coffee, tetracycline (TC) and

sulfamethazine (SMT). Therefore, the situations that can be possibly presented in such plant are categorized according to the Table 10.4

Table 10.4 Plant configurations and discrete process decisions considered

Class of irradiation	Configuration medium/low P lamp	**Class of contaminant**	Contaminant
A	OFF/OFF	CT0	None
B	ON/OFF	CT1	Coffee
C	OFF/ON	CT2	TC
D	ON/ON	CT3	SMT

Furthermore, total organic carbon (TOC) and H_2O_2 concentrations are off-line measured at different time scales (each 10 or 15 minutes) in the Photo-Fenton reactions that take place in the pilot plant.

10.2.2 Description of the nominal and faulty conditions in the plant for setting up the AD method

In order to apply the AD method to the Photo-Fenton pilot plant, the nominal operating conditions of the process are fixed for the involved variables and parameters, according to Table 10.5. As it can be observed in this table, the Photo-Fenton treatments of coffee are considered as part of the NOC. Also, it was required a 2^2 factor analysis (factors governing the H_2O_2 dosage: t_{ini} and P_0) for establishing the batches under NOC. The central design considers that t_{ini} =15 min and P_0=20%. The off-line process variables were not included in the construction of the data models since they cannot be integrated in an automatic FDS.

Table 10.5 Variables and parameters values in the Photo-Fenton process at NOC

Variable/Parameter	Value/Arrangement	Units
Contaminant	CT1	-
Irradiation	C	-
Contaminant initial concentration ($CT1_o$)	300.0	mgL^{-1}
Fe(II) concentration (C_{FeII})	10.0	mgL^{-1}
Equivalent initial hydrogen peroxide concentration $[H_2O_2]_{eq}$	500.0	mgL^{-1}
H_2O_2 initial dosage time (t_{ini})	0 – 36.2	min
H_2O_2 initial percentage of the total dosage (P_0)	5.9– 34.1	%
Operation volume (V_{op})	8.0	L
Ph	3.0	-
Dosage time (t_{add})	60.0	min
Recirculation flow (Q_1)	11.3	$Lmin^{-1}$
Pump work (PW)	75.0	%

Records from three different situations out of the experimental design and thus of the NOC were available and considered as faults. These three faults are:

1. Higher concentrations of $[H_2O_2]_{eq}$ than 500 mgL^{-1} (fault 1)
2. Change of configuration of the irradiation source from C to B (fault 2)
3. Different H_2O_2 dosage protocol out of the experimental design (fault 3)

These three faults were used for constructing the detection and diagnosis models required in both stages (binary and multi-class) of the AD method. Moreover, two different batch conditions were considered as novel faults so that the AD method could be validated. These novel faults are higher Fe(II) concentration (C_{FeII}=40ppm) and configuration A in the irradiation source (lack of irradiation).

10.2.3 Validation and results

A set of twenty experimental batches under NOC ($I_0=20$) and 10 batches per fault ($I_F=10$) were used for constructing the models due to the lack of more available batches and the time consumed for their generation. The final sampling time considered for the model construction was t_s=3 min and the duration of the batch for the NOC batches was established in 2 h, giving forty-one total observations per batch ($K=41$). In addition, the time was included as one more process variable, thus $J=9$. All the batches ($I=40$) were variable-wise unfolded, centered and scaled and merged in the training set ($\mathcal{IR}^{*}$). The reason for selecting such standardization method is that in the previous chapter the variable-wise centering and scaling showed to render better performance than batch-wise when applying MPCA to batch processes.

A feature extraction step was done by applying MPCA to the centered and scaled NOC batches (TR_0^{*}), keeping the same number of components than variables (9) and giving as result the projection model. The projection of both NOC and AOC batches onto such model produces the scores ($\mathcal{IR}^{**} \epsilon \mathcal{R}^{50 \cdot 41X9}$), used as input of both binary and multi-class classifiers, applying SVM as classification algorithm.

The training sets ($\mathcal{IR}^{**}$) were used as validation sets ($\mathcal{V}^{**}$) in both classifiers for adjusting parameters. The polynomial kernel of third order showed the best performance for the binary classifier and a polynomial kernel of fourth order for the multi-class classifier. On the other hand, a test set ($\mathcal{IT}^{**}$) containing one batch per class is constructed and used for evaluating the performance of both classifiers. Table 10.6 and 10.7 show the corresponding results.

Table 10.6 Performance of the binary classification stage

Class	F1 score (%)	
	Validation set	Test set
Nominal	100	96.0
Faulty	100	98.2

Table 10.7 Performance of the multi-class classification stage

Class	F1 score (%)	
	Validation set	Test set
Nominal	100	96.4
Fault 1	100	93.1
Fault 2	100	99.5
Fault 3	100	94.6
Mean	100	95.7

The results in Tables 10.6 and 10.7 show the good performance of both detection and diagnosis stages, which integrate the AD method. In order to assess this method, two batches with two novel faults are tested. These novel faults are:

- Higher concentration of Fe^{2+} (40 ppm) than the nominal (10 ppm) as fault 4
- Lack of irradiation source (configuration A) as fault 5.

The results after applying simultaneously both detection and diagnosis stages and therefore the AD methodology are reported in Table 10.8. As it can be observed, the whole batch with fault 4 is diagnosed as a novel fault, indicated by its detection as fault by the binary classifier and the lack of positive diagnosis of all the known faults (faults 1 to 3). On the contrary, fault 5 is detected as fault but just few observations are diagnosed as novel fault and most of them as diagnosed as fault 3. This points out that the behavior of the on-line variables when there is no irradiation source is similar to their behavior when the H_2O_2 dosage protocol is different to the protocol included in the experimental design.

Table 10.8 AD method validation on two novel faults. Performance in terms of F1(%)

Novel fault	Detection stage		Diagnosis stage			
	NOC	Fault	NOC	Fault 1	Fault 2	Fault 3
Fault 4	-	100	100	-	-	-
Fault 5	-	98.8	7.3	-	-	92.7

In general, successful results have been achieved by the proposed AD methodology. These results encourage to go on with further research in a next future. A first step may be using batches with different contaminants as novel faults. It would be probably require to include the off-line variables in the training data so as to perform potential models that take into account the degradation behavior of the different contaminants. Such models might be then capable of interfering composition changes in wastewater treatments, which would constitute a powerful tool.

10.3 Conclusions about the AD application on batch processes

This chapter reports the application of the anomaly detection (AD) method proposed in chapter seven on batch processes. AD allows detecting novel faults not

registered previously or not included in the diagnosis models. The proposed methodology combines detection (binary classification) with diagnosis (multi-class classification). For both cases machine learning algorithms, such as SVM, are used for classification.

The AD method is applied to a lab-scale heat exchange system and a Photo-Fenton pilot plant. In both applications, real data was used for constructing the models of the two classifiers, as well as for their validation. Moreover, variable-wise unfolding was done in both cases so as to apply the method in an on-line manner and obtain results at each sampling time.

In the lab case study, the models are constructed with random and extended data and a novel fault is detected successfully as such. Regarding the pilot plant, the on-line approach was applied using MPCA and SVM. Three faults are integrated in the models and two novel faults are used to validate the AD method.

In general, successful results are obtained from applying the AD method to the Photo-Fenton pilot process. Then, this motivates to applying the method in water research for detecting novel contaminants in wastewater treatment. Besides, the implementation of a real data-based FDDS together with an AD method in a real pilot plant encourages to apply them to a bigger scale.

PART V
CONCLUSIONS AND CONTRIBUTIONS

Chapter 11
THESIS CONTRIBUTIONS AND FINAL CONCLUSIONS

This final chapter is dedicated to summarizing the main conclusions of this thesis and to list its contributions to the PSE area. In addition, publications and congress contributions resulting from the research and the development of the thesis are stated in here. Finally, future work is envisaged.

11.1 General conclusions

The main data-driven FD methods used in PSE are imported from the machine learning area. These methods have been applied as classifiers for diagnosing faults in both continuous and batch processes along the research work.

A general FDS has been developed in this thesis based on several stages from data acquisition to the construction of diagnosis models. The methods applied in this thesis for developing each of the steps of the proposed methodology are provided in the chapters corresponding to the methods section.

Regarding the data pre-processing/standardization step, centering and scaling have been applied. In addition, the three existing ways of processing data to a feature space (feature extension, feature selection and feature extraction) have been developed in this thesis. Specifically, feature extraction is nowadays widely applied for dimensionality reduction. With this aim, many MSPM techniques have been used in PSE. In this thesis, PCA, ICA, DPCA, and DICA have been implemented for dealing with continuous processes, while MPCA, MICA, and BDPCA with batch processes. Furthermore, PCA, MPCA and BDPCA have been used for monitoring purposes.

The construction of the FD models is the bottleneck in the development of a FDS. Thus, different approaches have been proposed, evaluated and compared in continuous and batch processes.

Classification methods can be used as FD methods. In this sense, ANN and SVM have been used in this thesis as classification methods under a multi-class and multi-label

scenario. Besides, clustering methods are introduced in this thesis as diagnosis methods. Specifically, GMM with BIC have been used as FD method.

Next, diagnosis models need to be validated and their application needs to be assessed. Due to the lack of a single universal diagnosis performance index, the F1 score, which was first introduced in machine learning, has been used so as to compare different methods and approaches at the same level.

In the case of continuous processes the FD approaches proposed are based on the process period from which data is taken and each methodology differs in the application of different techniques. All these approaches have been tested on the TEP benchmark.

- It has been revealed that it is difficult to compare the TEP diagnosis results among the approaches that have been investigated with the reported in literature.
- The main conclusion is the advantage of combining two diagnosis models: a transient-data based model and a steady-state-data based model. When a single diagnosis model is needed, transient-data models yield better diagnosis performance than the rest of models.
- Therefore, the methodology for the treatment of the process information has been revealed of greater significance than the specific classification algorithms.
- Regarding the specific application to TEP, it can be also extracted that similar faults can be grouped in one class in order to obtain more robust models that can diagnose the reduced number of faults with better performance.

In the case of batch processes, the FD approaches proposed are based on the moment at which the batch is diagnosed, whether at the end or sample by sample (on-line). These approaches have been applied on the BPP, also considered as a benchmark for batch processes as the TEP is for continuous. All the approaches for continuous and batch processes are compared with the results obtained with some other approaches, reported by their corresponding authors and found in the literature.

Regarding the implementation of the FDS (Appendix 2), this is included as module in a whole communication architecture for the reading of the diagnosis results through a Web application. TEP is connected with the FDS in a Simulink-Matlab framework and on-line results are obtained.

One of the main weaknesses of data-based FD methods is their difficulty for detecting novel faults not learned with the models. This issue is called Anomaly Detection (AD) and has been dealt with a proposed methodology. Finally, the FDS and the AD methodology have been applied to real cases. A lab and pilot plant operating batchwise were the real case studies and successful and promising results were obtained for further applications.

The diagnosis task needs to be complemented with the management of preventive actions. In this sense, although Risk Analysis is applied in the process design stage, it can share information with the FDS in order to take suitable corrective actions. In contrast, Risk-Based Maintenance techniques of recent generation have been addressed for developing the preventive maintenance tasks. Therefore, preventive maintenance

programs together with the powerful FDS would constitute a complete plant safety program to implement.

11.2 Contributions to the area

The contributions of this thesis are next summarized. First, a general classification of the FD methods is given by joining the classification from the PSE area with the given by the AI area. The proposed classification is based on the kind of information used to construct the diagnosis models (knowledge or historical data).

The main contribution of this thesis is the proposal of a general FDS based on data-driven models on the way to its implementation and automation in order to provide a decision-support tool for plant operators. The methodology involved in the FDS can be used as guidelines for dealing with FD in several processes. Furthermore, the development and use of a single formulation and nomenclature in the FD algorithms as the proposed in here would allow replacing easily the applied methods by others in future.

Regarding the specific methodology steps, an alternative to centering and scaling as data pre-processing method is proposed. Its validation on continuous processes has shown enhanced diagnosis performance. A comparison between batch-wise and variable-wise centering and scaling has been performed when applying a PCA-based method in batch processes. Batch-wise standardizations have shown to produce better performance when using BDPCA, while variable-wise standardization is better when applying MPCA.

Feature extension has been proposed and used as information improvement step before reducing dimensionality by selection or extraction, obtaining enhanced performances. Regarding diagnosis algorithms, SVM are highly exploited in this thesis and they are applied successfully on both continuous and batch processes, which is corroborated by the diagnosis performances obtained. In the same context, clustering methods have been applied for the first time for developing the diagnosis step by grouping the process data into classes or clusters. Specifically, GMM technique is applied for this purpose.

Different general FDS approaches have been proposed for continuous and batch processes. Regarding continuous processes, the corresponding approaches are based on the process state (steady-state, random, latent, transient). Specifically, latent and transient data approaches are contributed in this thesis. Also, the combined use of two diagnosis models (transient and steady-state) has been proposed. The benefits of this contribution are supported by the comparison between models and their evaluation with different validation data sets and therefore, by the resulting diagnosis performances.

Regarding batch processes, two general FD approaches have been developed and evaluated, which are based on the moment in which the diagnosis is developed, either at the end of batch (off-line) or sample by sample (on-line). A comparative study between feature extraction and classification techniques in batch processes with several levels of noise in data is developed, revealing that the choice of the feature extraction method is

more decisive than the choice of the classification method. A second comparative study between batch-wise and variable-wise centering and scaling is addressed for both MPCA and BDPCA techniques in the on-line diagnosis approach. It is thus reported that BDPCA performs better when diagnosing the process on-line and a batch-wise centering and scaling should be developed. However, when MPCA is applied, the variable-wise centering and scaling is better in terms of diagnosis performance.

On the other hand, an AD methodology has been contributed for detecting novel faults neither reported before nor included in the diagnosis models. This method merges detection and diagnosis stages by using binary and multi-class classification methods and deals with the main drawback of data-based diagnosis models. The AD method has been successfully applied to a lab heat exchanger and a Photo-Fenton pilot plant by detecting novel faults with the detection and diagnosis models constructed.

Finally, these thesis contributions have been validated not only to simulated processes, but also against real cases, a lab case study and a pilot plant. The success of such implementation validates once again the methodology and techniques developed and supports their industrial application.

Furthermore, RBI is proposed to be applied to different PSE fields for reducing the risk in the in-service equipment and for the detection of progressive damage in there, as part of a preventive maintenance program.

The thesis has also proposed to develop and implement a whole plant safety program in future by joining the FDS with an appropriate preventive maintenance program. The implementation of the general FDS proposed and an application example of a preventive maintenance technique (RBI) are given in the appendixes section.

11.3 Future work

Some areas that are still open to research and future work that can be undertaken is next exposed. Regarding specific investigation that may provide deeper insight and lead to the enhancement of the methods developed, the following issues should be considered:

- A study of the retained variance and the number of components to retain with a feature extraction method should be done for batch processes in order to investigate the effects of both parameters on the diagnosis performance. It is probable to obtain similar results than in continuous processes. Higher variance retained, better diagnosis performance.
- An optimization work could be also done for making decisions regarding to the different key steps in the FDS. For instance, the parameters to optimize in a batch process would be then the number of batches to include in the models, the number of inputs to the diagnosis algorithm (number of retained components) and the kernel function (in SVM) or number of hidden nodes (in ANN).

Regarding open research lines, the following challenges should be considered:

- A more complete work to develop with the FDS is to include corrective actions to the possible faults and the consequences of such faults. This would generate a more complete and realistic research.
- A field to be exploited is to profit the information from the different modules that integrate a chemical process. In the same way, the output from the diagnosis module can be integrated and used for others. In this sense, the proposed FDS could incorporate a preventive maintenance program in a whole robust plant safety program.
- Finally, the AD method can be extended for further investigations such as the detection of contaminants in wastewater treatments. In addition, the methodology should be also tested in continuous processes as it has been tested on batch processes.

11.4 Work done and publications

The next publications have resulted from the research work for the development of the current thesis:

1. **A semisupervised approach to fault diagnosis for chemical processes.** Monroy I., Benítez R., Escudero G., Graells M. Computers and Chemical engineering, 34 (2010), 631-642.
2. **Fault diagnosis of a benchmark fermentation process. A comparative study of feature extraction and classification techniques.** Monroy I., Villez K., Graells M., Venkatasubramanian V. Bioprocess and biosystems engineering. Accepted.
3. **Enhanced plant fault diagnosis based on the characterization of transient stages**. Monroy I., Benítez R., Escudero G., Graells M. Computers and Chemical engineering. Accepted.

In addition, the works submitted and presented in congresses are next listed:

1. Improving the Representation of Process Information in Multi-Label Fault Diagnosis Systems. **Isaac Monroy**, Gerard Escudero and Moisès Graells. 18th European Symposium on Computer Aided Process Engineering (ESCAPE 18).
2. Using extended information as features for enhancing the performance of Fault Diagnosis Systems for chemical processes. **Isaac Monroy**, Gerard Escudero and Moisès Graells. 11th Mediterranean Congress of Chemical Engineering.
3. Anomaly detection in batch chemical processes. **Isaac Monroy**, Gerard Escudero and Moisès Graells. 19th European Symposium on Computer Aided Process Engineering (ESCAPE 19).
4. DICA enhanced SVM classification approach to fault diagnosis for chemical processes. **Isaac Monroy**, Raul Benítez, Gerard Escudero and Moisès Graells. ESCAPE 19.
5. Semisupervised methodology for fault diagnosis in chemical plants. **Isaac Monroy**, Raul Benitez, Gerard Escudero and Moisès Graells. AIChE meeting 2009.

6. On-line fault diagnosis based on the identification of transient stages. **Isaac Monroy**, Raul Benitez, Gerard Escudero and Moisès Graells. 20th European Symposium on Computer Aided Process Engineering (ESCAPE 20).
7. Dynamic process monitoring and fault detection in a batch fermentation process: comparative performance assessment between MPCA and BDPCA. **Isaac Monroy**, Kris Villez, Moisès Graells and Venkat Venkatasubramanian. 21th European Symposium on Computer Aided Process Engineering (ESCAPE 21).
8. Anomaly detection in a Photo-Fenton pilot plant. **Isaac Monroy**, Evelyn Yamal, Gerard Escudero, Moisès Graells and Montserrat Pérez-Moya. 12th Mediterranean Congress of Chemical Engineering.
9. An open architecture for the fault diagnosis of chemical processes. Javier Silvente, **Isaac Monroy**, Gerard Escudero, Antonio Espuña and Moisès Graells. 12th Mediterranean Congress of Chemical Engineering.
10. A novelty detection approach for detecting faulty batches in a Photo-Fenton pilot plant. **Isaac Monroy**, Evelyn Yamal, Gerard Escudero, Moisès Graells and Montserrat Pérez-Moya. 22th European Symposium on Computer Aided Process Engineering (ESCAPE 22). *To be presented*.
11. A promising OPC-based computer system applied to fault diagnosis. Javier Silvente, **Isaac Monroy**, Gerard Escudero, Antonio Espuña and Moisès Graells. ESCAPE 22. *To be presented.*

Participation in research projects

- Estudio centrado en la obtención de la Homologación de la "Papier Cheque Societé Generale" y de "La Poste" para Vilaseca S.A, 2007.
- TOLERANT T. Advanced Support Systems for Flexible Production Processes in Chemical and Petrochemical Industry, supported by the European Union (European Regional Development Fund 2007-13) and the Spanish Ministry of Science and Innovation. (DPI2006-05673), 2007-2009.
- EHMAN. Expanding the Horizons of Manufacturing: Solving the Integration Paradox, supported by the European Union (European Regional Development Fund 2007-13) and the Spanish Ministry of Science and Innovation. (DPI2009-09386), 2009-2011.
- Implementación del programa SIMECELE en PEMEX Refinación para la inspección y control de espesores en tuberías y equipos, 2010.

APPENDIX 1
RBI APPLICATION

This appendix presents an application of RBI to PEMEX REFINEMENT. PEMEX prioritizes thinning of equipment and pipes, dimensional changes and deterioration of screws and nipples as progressive damages to deal with and prevent. Such damages are one of the main causes of ruptures, leakages and releases of equipments and pipes.

The mechanical calibrations file has to be updated with all the required elements and including the analysis of thickness measurements, damage rates, next measurement dates, estimated useful life of equipments and pipes and replacement dates. Nowadays, PEMEX REFINEMENT manages such information by a program created by CEASPA group (Study center for the management of polymeric and petrochemical processes and environmentalprotection) at the UNAM. This system includes an application capable of capturing data directly from industry and identifying anomalies in the moment of the measurement.

The inspection planning is thus focused on the management of the mechanical calibration files and the establishment of some calculations as part of the preventive maintenance and inspection. Visual examination, thickness measurements and in some specific cases (welded arrangements) radiography and ultrasonic devices are the inspection techniques used to deal with the progressive damages. Next, the parameters calculated when developing the inspection program are presented:

Replacement limit (L_r): It has to be calculated when it is not given by the manufacturer for all the diameters and materials that integrate the control unit according to the code ANSI B 31.3. The replacement limit for pipes with flat ends according to PEMEX criterion is calculated as in A1.1:

$$L_r = \frac{P \times D}{2S} \qquad \text{A1. 1}$$

where P is the design pressure (lb/in^2), D the nominal diameter (in) and S the maximum effort permissible to the design temperature (lb/in^2). On the other hand, the replacement limit for pipes with screw ends is calculated as in the next equation:

$$L_{rr} = L_r + R \qquad \text{A1 2}$$

where R is the additional thickness due to the screw depth (in).

Specific damage rate (*d*): It is the damage that suffers the material between two consecutive date.

$$d = \frac{e_i - e_f}{f_f - f_i} \qquad \text{A1. 3}$$

where e_i is the thickness (milch) obtained in date f_i and e_f is the thickness obtained in date f_f

Damage rates are filtered according to these criteria:

- Damage rates in which any of the two thickness measurements are bigger than the nominal thickness plus the 20% of swelling are not considered in calculations.
- Negative damage rates are also discarded, which indicate a thickness increase.
- Moreover, damage rates higher than 20 milch per year are considered as critical values.

It is also important to mention that the minimum quantity of thickness values in a control unit must be 32 so that the statistical analysis is reliable.

Next calculations are developed differentiating two groups of values characterized according to the specific damage rates values and called typical and atypical. The original criterion used by PEMEX Refinement is next presented. Also a comparison of this method with PCA was performed.

Quartile method

Quartilees are three values that divide an ordered set of data in four parts porcentual equal. Given a set of values, the first quartilee is defined as the median of the first half of values, in other words, it's the value below which there is a quarter (25%) of all the values of the ordered series. The second quartilee is the median of the total set of values, while the third quartilee is the value below which there are three quarters of the data (75%). In other words, third quartilee is the median of the second half of values.

The values over the upper threshold (UCL) are considered as atypical.

$$UCL = Q_2 + 1.5 \times IQR \qquad \text{A1. 4}$$

where $IQR = Q_3 - Q_1$.

Statistically adjusted damage rate (D_{max}): It is a statistical adjustment in order to obtain the maximum damage of a control unit with a 90% of reliability. PEMEX Refinement has their own expressions but they were slightly modified by CEASPA group. These equations are next presented (A1.5 for typical values and A1.6 for atypical values).

$$D_{max,t} = D_{m,t} + t_{\alpha,t} \frac{S_t}{\sqrt{n_t}} \qquad \text{A1. 5}$$

$$D_{max,a} = D_{m,a} + t_{\alpha,a} \frac{S_a}{\sqrt{n_a}} \qquad \text{A1. 6}$$

where *S* is the standard deviation and D_m is the mean of the damage rate values and *a* or *t* represent whether the values are typical or atypical.

Estimated useful life (*VUE*): It is the supposed time before the control unit reaches its replacement limit and it is calculated with the next equation:

$$VUE = \frac{e_k - L_r}{D_{max}} \qquad \text{A1. 7}$$

where e_k is the minimum thickness found during the inspection in the control unit and L_k is the replacement limit. *VUE* is also calculated for typical and atypical values separately.

Next thichness measurement rate (*FPME*): It is the date when the next measurement must be taken. It is calculated with the next equation:

$$FPME = f_k + \frac{VUE}{3} \qquad \text{A1. 8}$$

where f_k is the date of the current measurement. There is one date for the typical values of the control unit and another one for the atypical values.

Probable replacement date (*FRP*): It is the final calculation in an inspection program and it is the date when the control unit is estimated to be replaced because its useful life has been reached. This value is calculated with the next equation:

$$FRP = f_k + VUE \qquad \text{A1. 9}$$

There are also two dates corresponding to the typical and atypical values in the control unit. In general, by dividing thickness calibration in two groups (typical and atypical), two next inspection dates are obtained, which are fairly representative of the damage level existing. An economical benefit is also obtained by avoiding calibrating levels not necessary to measure. The calculations after inspection must be evaluated according to the norm DG-SASIPA-IT-0204 so that they can be valid and reliable. Although the calculations are not complex, the main problems can be originated due to a wrong way of calibrating thicknesses.

Deterioration rates are better defined with future inspections and therefore the inspection frequencies become established in a more accurate way. This is actually part of a good inspection updating and it becomes the basis of a successful system audit. Table A.1 shows the calculations for a control unit with data from two inspections. The pipe line has a diameter of 8 in, a nominal thichness of 322 mil and a replacement limit of 150 mil.

Table A1.1 Inspection results from a control unit

Parameters calculated	Different damage rates (d)				
	d	d without negatives	d without critical	d typical	d atypical
Thickness number	185	130	129	122	8
Minimum value	-44.73	0.36	0.36	0.36	10.18
Maximum value	25.82	25.82	15.64	8.73	25.82
Mean	1.12	4.41	4.24	3.82	13.45
Std	7.93	3.37	2.8	2.17	5.32
1^{st} quartile	0.73	1.91	1.82	1.82	10.18
2^{nd} quartile	2.55	4.0	4.0	4.0	11.45
3^{rd} quartile	4.73	5.82	5.82	5.45	13.45
4^{th} quartile	25.82	25.82	15.64	8.72	25.82
UCL	10.73	9.86	10.0	9.45	16.36
LCL	5.64	1.86	-2	-1.45	6.54
IQR	5.45	3.91	4.0	3.64	3.27
Outliers	5	8	7	0	1
Negative outliers	1	0	0	0	0
Limit 95% std	16.99	11.16	9.85	8.17	24.10
Limit 5% std	14.75	2.34	-1.37	-0.53	2.81
t_α	1.65	1.66	1.66	1.66	2.37
D_{max} (mil/year)	2.09	4.9	4.65	4.14	17.91
VUE (years)	45.5	19.4	20.4	22.9	**7.3**
FPME (years)	5.0	5.0	5.0	5.0	**2.42**

As it is observed, the atypical regions of the control unit require a next inspection in shorter time to five years in comparison to the rest of measurement levels.

APPENDIX 2
FDS IMPLEMENTATION

This appendix presents an implementation of the FDS proposed in this thesis to continuous processes. The diagnosis system is connected with the process in order to produce real-time results. The process is TEP simulator in Simulink/Matlab and the FDS is implemented incorporated as an S-function to the process. Next, the FDS is explained in detail.

Fault Diagnosis System

The diagnosis models are the transient models with 44 features (retained PCs) and obtained with a polynomial function of second degree by SVM algorithm. Such models demonstrated to produce a good diagnosis performance when diagnosing not only transient regimes, but also steady-state data (with a less performance though). A further improvement of the system would be to incorporate two diagnosis models working at the same time, one from transient regime and another from steady-state regime.

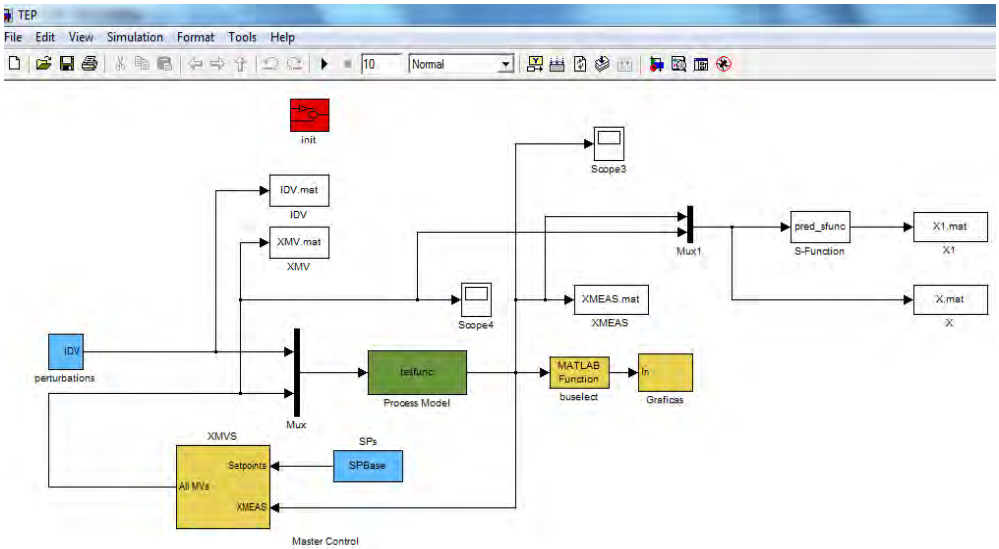

Figure A2.1 Scheme of the FDS implementation in Simulink

Therefore, the diagnosis models are loaded in Simulink by means of an S-function, which is incorporated to the TEP process simulator. Such models are then applied to the real-time measurements generated by the process. As result, diagnosis values or predictions to the corresponding scenarios are also obtained on-line. In more detail, the whole diagnosis system receives the on-line data by the simulator and generates the diagnosis, interpreted from the predictions to each faulty scenario. If the diagnosis value of one fault changes from a negative to a positive value, such fault is occurring and being diagnosed, otherwise the process remains under NOC. Figure A.2 shows the scheme of the FDS incorporated to the TEP process in Simulink.

An OPC-based open architecture

As further work, an OPC-based architecture is being designed with an application to fault diagnosis. This architecture would allow performing diagnosis on-line by connecting a simulator that can be later replaced for a real plant with the FDS within the architecture.

This architecture consists of the next components: a data generator, that can be either a real plant or a simulator, a diagnosis system as decision-making support tool, a standard specification based on OPC, an information repository and a customer interface to the supervisory system. Figure A2.2a shows the general scheme of this architecture and Figure A2.2b shows the particular implementation.

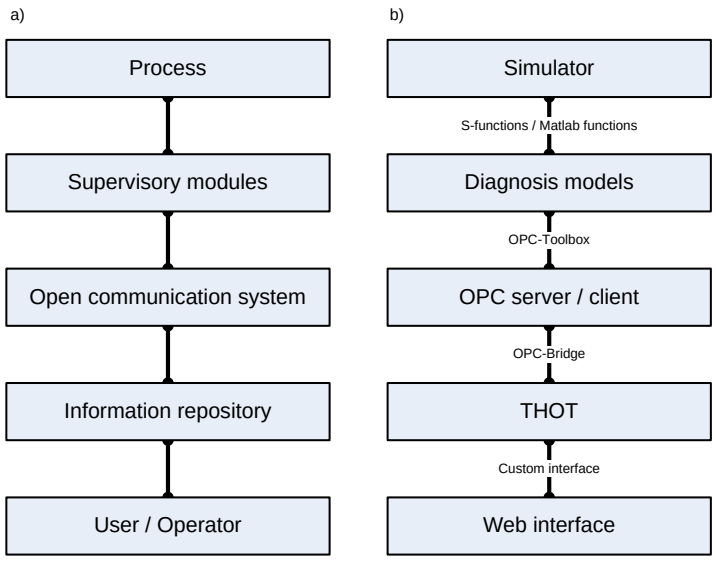

Figure A2.2 a) General structure of the proposed architecture, and b) particular implementation

Specifically, the data generator consists of the process itself. A process simulator can perfectly work as data generator for validating the information architecture. Then, process samples from all the measured and controlled variables are the input of the

diagnosis module or FDS. This module is developed in Matlab and incorporated in Simulink with an S-function in order to generate the diagnosis results on-line at real time.

An OPC server is incorporated into the architecture to set up the communication of the real-time process data from the process or from the FDS (as in this case) to the user. Indeed, supervision engineers can play this role in real industrial applications. An intelligent platform for real-time management of decisions at different levels of production, called THOT, is an important element in the architecture. This platform was designed by CEPIMA research group, and works as a repository of all the stored and available information obtained from different processes and data-bases in a transparent XML format. It plays an important role in the computer architecture by receiving the diagnosis information from OPC through an OPC-bridge and sending it to the operator through a custom interface. The platform is designed to be as open and flexible as possible. Thus, a flexible platform would allow sending information in the inverse sense to the plant and receiving and sending information from and to other decision-making support tools.

Finally, the custom interface in the computer architecture has been implemented as a Web application that uses an applet to render information coming from the process and the diagnosis module and passing through the platform. The real-time variable values of each variable are presented with visual aids allowing their reading and comprehension. Furthermore, the corresponding diagnosis values to each modeled fault will be provided in the same manner, allowing the diagnosis of the process as an application of the computer architecture. Beyond this, real-time diagnosis would allow taking appropriate corrective actions on time. This promising architecture is in the construction phase and represents a real implementation of the proposed FDS in this thesis.

BIBLIOGRAPHY

(AIChE),American Institute of Chemical Engineers. http://www.aiche.org/about/code.aspx.

Addin, O, S.M Sapuan, E Mahdi, and M Othman. "A Naïve-Bayes classifier for damage detection in engineering materials, 28, 8." *Materials and Design*, 2007: 2379-2386.

Aguado, D, J Ferrer, and A Seco. "Multivariate SPC of a sequencing batch reactor for wastewater treatment." *Chemometrics and Intelligent Laboratory Systems, 85*, 2007: 82-93.

Aleixandre, M, et al. "Analysis of neural networks and analysis of feature selection with Genetic Algorithm to discriminate among pollutant gas." *Sensors and Actuators B: Chemical, 103, 1-2*, 2004: 122-128.

API Publication 581. *Risk based inspection. Base Resource Document.* Information Handling Services, 2001.

Arranz, A, A Cruz, M.A Sanz-Bobi, P Ruiz, and J Coutiño. "DADICC: Intelligent systems for anomaly detection in a combined cycle gas turbine plant." *Expert Systems with Applications, 34*, 2008: 2267-2277.

Arunraj, N.S, and J Maiti. "Risk-based maintenance-Techniques and applications." *Journal of Hazardous Materials, 142*, 2007: 653-661.

Bajpai, R, and M Reuss. "A mechanistic model for penicillin production." *Journal of Chemical Technology and Biotechnology*, 1980: 330-344.

Bakshi, B.R, and G Stephanopoulos. "Representation of process trends –IV: Induction of real-time patterns from operating data for diagnosis and supervisory control." *Computers and Chemical Engineering, 18*, 1994: 303-330.

Barber, M. *The Penicillins Business.* Michael Barber and Associates, 1996.

Becraft, W, and P Lee. "An integrated neural network/expert system approach for fault diagnosis ." *Computers and Chemical Engineering, 17, 10*, 1993: 1001-1014.

Benitez, R, and N Zorac. "Robust Unsupervised Detection of Actions Potentials with probabilistic methods." *IEEE Transactions on Biomedical Engineering, 55, 4*, 2008: 1344-1354.

Bertolini, M, M Bevilacqua, F.E Ciarapica, and G Giaccheta. "Development of Risk-Based Inspection and Maintenance procedures for an oil refinery." *Journal of Loss Prevention in the Process Industries, 22*, 2009: 244-253.

Bestratén Belloví, Manuel. *Los análisis de peligros y de operabilidad en instalaciones de proceso.* Notas técnicas de prevención NTP-238. Instituto Nacional de seguridad e higiene en el trabajo., Madrid, Esp.: Centro Nacional de condiciones de trabajo, 1989.

Bin He, X, W Wang, Y Pu Yang, and Y Hong Yang. "Variable-weighted Fisher discriminant analysis for process fault diagnosis." *Journal of Process Control, 19, 6*, 2009: 923-931.

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

Birol, G, C Ündey, and A Çinar. "A modular simulation package for fed-batch fermentation: penicillin production." *Computers and Chemical Engineering, 26*, 2002: 1553-1565.

Boser, B, I Guyon, and V Vapnik. "A training algorithm for optimal margin classifiers." *Proceedings of the Workshop on Computational Learning Theory, COLT.* Pittsburgh, PA, 1992.

—. "A training algorithm for optimal margin classifiers." *Proceedings of the Workshop on Computational Learning Theory, COLT.* Pittsurgh, PA, 1992.

Brown, S.J, and I.L May. "Risk-based hazardous protection and prevention by inspection and maintenance." *Trans. ASME J. Press. Ves. Technol. 122*, 2003: 362-367.

Bruggink, A, E Roos, and E De Vroom. "Penicillin Acylase in the industrial production of β-Lactam antibiotics." *Organic Process Research and Development, 2, 2*, 1998: 128-133.

Brydon, D.A, J.J Cilliers, and M.J Willis. "Classifying pilot plant distillation column faults using Neural Networks." *Control Engineering Practice, 5*, 1997: 1373-1384.

Cabrera, J.B.D, C Gutiérrez, and R.K Mehra. "Ensemble methods for anomaly detection and distributed intrusion detection in Mobile Ad-Hoc Networks." *Information fusion, 9*, 2008: 96-119.

Camacho, J, and J Picó. "On-line monitoring of batch processes using multi-phase principal component analysis." *Journal of process control, 16*, 2006: 1021-1035.

Camacho, J, J Picó, and A Ferrer. "The best approaches in the on-line monitoring of batch processes based on PCA: Does the modelling structure matter?" *Analytica Chimica Acta 642*, 2009: 59-68.

Chang, C.C, and C.C Yu. "On-line fault diagnosis using the signed directed graph." *Industrial and Engineering Chemistry Research, 29, 7*, 1990: 1290-1299.

Chang, M-K, R-R Chang, C-M Shu, and K-N Lin. "Application of risk based inspection in refinery and processing piping." *Journal of Loss Prevention in the Process Industries, 18*, 2005: 397-402.

Chen, A, Z Song, and P Li. "Soft Sensor Modeling based on DICA-SVR." *Lectures notes in computer science Vol 3644.* 2005.

Chen, J, and J Liu. "Mixture principal component analysis models for process monitoring." *Ind Eng Chem Res, 38*, 1999: 1478-1488.

Chen, J, and K-C Liu. "On-line batch process monitoring using dynamic PCA and dynamic PLS models." *Chemical Engineering Science, 57*, 2002: 63-75.

Chen, J.H, and C.M Liao. "Dynamic process fault monitoring based on neural network and PCA." *Journal of Process Control, 12, 2*, 2002: 277-289.

Chen, L-H, and H-D Hsiao. "Feature selection to diagnose a business crisis by using a real GA-based support vector machine: An empirical study." *Expert Systems with Applications, 35, 3*, 2008: 1145-1155.

Chen, Y, J Liang, and H Lu. "Multi-model MPCA with Applications in the Monitoring and Fault Diagnosis of Multivariable Control System." *Journal of Jiangnan College*, 2002.

Chester, D, D Lamb, and P Dhurjati. "Rule-based computer alarm analysis in chemical process plants." *Proceedings of 7th Micro-Delcon.* 1984. 22-29.

Chiang, L, E Russell, and R Braatz. "Fault diagnosis in chemical process using Fisher discriminant analysis, discriminant partial least squares and principal component analysis." *Chemometrics and Intelligent Laboratory Systems, 50*, 2000: 243-252.

Chiang, L.H, and R.D Braatz. "Process monitoring using causal map and multivariate statistics: fault detection and identification." *Chemometrics ans Intelligent Laboratory Systems, 65, 2*, 2003: 159-178.

Chiang, L-H, M Kotanchek, and A Kordon. "Fault diagnosis based on Fisher discriminant analysis and support vector machines." *Computers and Chemical Engineering, 28*, 2004: 1389-1401.

Chow, E.Y, and A.S Willsky. "Analytical redundancy and the design of robust failure detection systems." *IEEE Transactions on Automatic control, 29, 7*, 1984: 603-614.

Cinar, A, S Parulekar, C Undey, and G Birol. *Batch fermentation: Modeling, Monitoring and Control.* New York: Marcel Dekker, 2003.

Clark, R.N. "The dedicated observer approach to instrument fault detection." *Proceedings of the 15th IEEE-CDC.* Ford Lauderdale, FL, 1979. 237-241.

Cristianini, N, and J Shawe-Taylor. *An introduction to Support Vecctor Machines and other kernel-based learning methods.* Cambridge: Cambridge University Press, 2006.

Cui, L, J Zhao, and R Zhang. "The integration of HAZOP expert system and piping and instrumentation diagrams." *Process Safety and Environmental Protection, 88, 5*, 2010: 327-334.

Cui, P, J Li, and G Wang. "Improved kernel principal component analysis for fault detection." *Expert Systems with Applications, 34*, 2008: 1210-1219.

Dahl, S, M Piovoso, and K Kosanovich. "Translating third-order data analysis methods to chemical batch processes." *Chemometrics and Intelligent Systems, 46*, 1999: 161-180.

Dash, S, and V Venkatasubramanian. "Challenges in the industrial applications of fault diagnostic systems." *Computers and chemical engineering, 24*, 2000: 785-791.

Decadt, B, V Hoste, W Daelemans, and A Van den Bosch. "GAMBL, Genetic Algorithm Optimization of Memory-based WSD." *Proceedings of the International Workshop on evaluating word sense disambiguation systems.* Barcelona: Senseval-3, 2004.

Dempster, A.P, N.M Laird, and D.B Rubin. "Maximum likelihood from incomplete data via the EM algorithm." *J. Roy. Stat. Soc. B Met, 39, 1*, 1977: 1-38.

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

Downs, J.J, and E.F Vogel. "A plant-wide industrial process control problem." *Computers and Chemical Engineering, 17, 3*, 1993: 245-255.

Eizenberg, S, M Shacham, and N Brauner. "Combining HAZOP with dynamic process model development for safety analysis." *Computer Aided Chemical Engineering, 21*, 2006: 389-394.

Elander, R. "Industrial production of β-lactam antibiotics." *Applied Microbiology and Biotechnology, 61*, 2003: 385-392.

Fleming, A. *British Journal of Experimental Pathology, 10*, 1929: 226-236.

Fraley, C, and A.E Raftery. "How many clusters? Which Clustering Method? Answers via Model-based Cluster Analysis." *The Computer Journal, 41, 8*, 1998: 578-588.

Fraley, C, and A.E Raftery. "Model-based clustering discriminant analysis, and density estimation." *J. Amer. Stat. Assoc, 97, 458*, 2002: 611-631.

Frank, P.M. "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy –a survey and some new results." *Automatica, 26*, 1990: 459-474.

Frank, P.M, and J Wünnenberg. "Robust fault diagnosis using unknown input observer schemes." In *Fault diagnosis in dynamic systems: theory and applications*, by Frank P.M., Clark R.N Patton R.J. NY: Prentice Hal, 1989.

Fussell, J.B. "Fault tree analysis –state of the art." *IEEE Transactions on Reliability, 23, 1*, 1974: 51-53.

Galluzzo, M, V Bartolozzi, and C Rinaudo. "Automating HAZOP analysis of batch processes." *Computers and Chemical Engineering, 23, 1*, 1999: S661-S664.

Ge, Z, C Yang, and Z Song. "Improved kernel PCA-based monitoring approach for nonlinear processes." *Chemical Engineering Science, 64*, 2009: 2245-2255.

Geladi, P, H Isaksson, L Lindqvist, S Wold, and K Esbensen. "Principal Component Analysis of Multivariate Images." *Chemometrics and Intelligent Laboratory Systems, 5*, 1989: 209-220.

Gertler, J. "Analytical redundancy methods in fault detection and isolation." *Proceedings of IFAC/IAMCS symposium on safe process.* Baden-Baden, 1991. 91.

Gertler, J, and Q Luo. "Robust isolable models for failure diagnosis." *AIChE Journal, 31, 11*, 1989: 1856-1868.

Gertler, J, X Fang, and Q Luo. "Detection and diagnosis of plant failures: the orthogonal parity equation approach." *Control and dynamic systems, 37*, 1990: 159-216.

Gulkaya, I, A Gulerman, and F Dilek. "Importance of H2O2/Fe+2 ratio in Fenton's treatment of a carpet dying wastewater." *Journal of Hazardous Materials, B136*, 2006: 763-769.

Gurden, S, J Westerhuis, R Bro, and A Smilde. "A comparison of multiway regression and scaling methods." *Chem. Intell. Lab. Syst., 59*, 2001: 121-136.

Han, C, R Shih, and L Lee. "Quantifying signed directed grapas with the fuzzy set for fault diagnosis resolution improvement." *Industrial and Engineering Chemistry Research, 33, 8*, 1994: 1943-1954.

Harrington, J. *Clustering algorithms.* New York: Wiley, 1975.

Haykin, S. *Neural Network, A comprehensive foundation.* MacMillan College Publishing, 1994.

Henley, E.J. "Application of expert systems to fault diagnosis." *AIChE annual meeting.* San Francisco, CA, 1984.

Himmelblau, D.M. *Fault detection and diagnosis in chemical and petrochemical processes.* Amsterdam: Elsevier Press, 1978.

Hoskins, J.C, K.M Kaliyur, and D.M Himmelblau. "Fault diagnosis in complex chemical plants using artificial neural networks." *American Institute of Chemical Engineers Journal, 37, 1*, 1991: 137-141.

Hotelling, H. "Multivariate quality control illustrated by the testing of sample bombsights." In *Selected techniques of statistical analysis*, by Eisenhart O., 113-184. New York: McGraw-Hill, 1947.

Huang, C, H-C Liao, and M-C Chen. "Prediction model building and feature selection with support vector machines in breast cancer diagnosis." *Expert Systems with Applications, 34*, 2008: 578-587.

Huang, C-L, and C-J Wang. "A GA-based feature selection and parameters optimization for support vector machines." *Expert Systems with Applications, 31*, 2006: 231-240.

Huang, S-C, and Y-F Huang. "Bounds on the number of hidden neurons in multilayer perceptrons ." *IEEE Trans On Neural Networks, 2*, 1991: 47-55.

Iri, M, K Aoki, E O'Shima, and H Matsuyama. "An algorithm for diagnosis of system failures in chemical processes." *Computers and Chemical Engineering, 3*, 1979: 489-493.

Isermann, R. "Process fault detection based on modelling and estimation methods –a survey." *Automatica, 20, 4*, 1984: 387-404.

Jackson, J.E. *A User's Guide to Principal Components.* New York: Wiley, 1991.

Jackson, J.E, and G.S Mudholkar. "Control Procedures for Residuals Associated with Principal Component Analysis." *Technometrics, 21*, 1979: 341-349.

Jakubek, S.M, and T.I Strasser. "Artificial neural networks for fault detection in large-scale data acquisition systems ." *Engineering Applications of Artificial Intelligence, 17*, 2004: 233-248.

Janusz, M, and V Venkatasubramanian. "Automatic generation of qualitative description of process trenes for fault detection and diagnosis." *Engineering Applications of Artificial Intelligence, 4, 5*, 1991: 329-339.

Jaynes, E, and G.L Bretthorst. *Probability Theory: The Logic of Science.* New York: Cambridge Univ Press, 2003.

Jiang, L, and S Wang. "Fault diagnosis based on Independent Component Analysis and Fisher Discriminant Analysis." *Proceedings of the third Conference on Machine learning and cybernetics.* Shanghai, 2004.

Jolliffe, I.T. *Principal Component Analysis.* New York: Springer-Verlag, 2003.

Jovanovic, A. "Risk-based inspection and maintenance in power and process plants in Europe." *Nuclear Engineering and Design, 226*, 2003: 165-182.

Kano, M, et al. "Comparison of multivariate statistical process monitoring methods with applications to the Eastman challenge problem." *Computers and Chemical Engineering, 26*, 2002: 161-174.

Kano, M, et al. "Comparison of statistical process monitoring methods: application to the Eastman challenge problem." *Computers and Chemical Engineering, 24*, 2000: 175-181.

Kano, M, S Hasebe, I Hashimoto, and H Ohno. "A new multivariate statistical process monitoring method using principal component analysis." *Computers and Chemical Engineering, 25, 7-8*, 2001: 175-181.

Kano, M, S Tanaka, S Hasebe, I Hashimoto, and H Ohno. "Monitoring Independent Components for Fault Detection." *AIChE Journal*, 2003: 969-976.

Kassidas, A, P Taylor, and J.F MacGregor. "Off-line diagnosis of deterministic faults in continuous dynamic multivariable processes using speech recognition methods." *Journal of Process Control*, 1998: 381-393.

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

Khan, F.I, and M Haddara. "Risk-based Maintenance (RBM): a new approach for process plant inspection and maintenance." *Process Saf. Prog. 23, 4*, 2004: 252-265.

Khan, F.I, and M Haddara. "Risk-based Maintenance (RBM): a quantitative approach for maintenance/inspection scheduling and planning." *J. Loss Prevent. Process Ind. 16*, 2003: 561-573.

Khan, F.I, R Sadiq, and M.M Haddara. "Risk-Based Inspection and Maintenance (RBIM). Multi-attribute decision-making with Aggregative Risk Analysis." *Process Safety and Environmental Protection, 82*, 2004: 398-411.

Kokawa, M, M Satoshi, and S Shigai. "Fault location using diagraph and inverse direction search with application." *Automatica, 19, 6*, 1983: 729-735.

König, B, C Seewald, and K Schügerl. "Process engineering investigations of penicillin production." *Applied Microbiology and Biotechnology, 12, 4*, 1981: 205-211.

Koourti, T. "Process analysis and abnormal situation detection: From theory to practice." *IEEE Control systems magazine, 22*, 2002: 10-25.

Koppen-Seliger, B, P.M Frank, and A Wolff. "Residual evaluation for fault detection and isolation with RCE neural networks." *Proceedings of the American control conference.* Seattle, Washington, 1995. 3264-3268.

Kosanovich, K.A, K.S Dahl, and M.J Pisovo. "Improved process understanding using multiway principal component analysis." *Ind. Eng. Chem. Res, 35*, 1996: 138-146.

Kourti, T, and J.F MacGregor. "Process analysis, monitoring and diagnosis using multivariate projection methods - A tutorial." *Chemometrics and Intelligent Laboratory Systems, 28*, 1995: 3-21.

Kramer, M.A, and B.L Palowitch. "A rule based approach to fault diagnosis using the signed directed graph." *American Institute of Chemical Engineers Journal, 33, 7*, 1987: 1067-1078.

Kresta, J.V, J.F MacGregor, and T.E Marlin. "Multivariate statistical monitoring of processes." *Canadian Journal of Chemical Engineering, 69, 1*, 1991.

Ku, W, R Storer, and C Georgakis. "Disturbance detection and isolation by dynamic principal component analysis." *Chemometrics and Intelligent Laboratory Systems, 30*, 1995: 179-196.

Kulkarni, A, V.K Jayaraman, and B.D Kulkarni. "Knowledge incorporated support vector machines to detect faults in Tennessee Eastman Process." *Computers and Chemical Engineering, 29, 10*, 2005: 2128-2133.

Langley, P. *Elements of machine learning.* New York: Morgan Kaufmann Publishers, 1996.

Lapp, S.A, and G.A Powers. "Computer-aided synthesis of fault trees." *IEEE Transactions on Reliability, 26, 1*, 1977: 2-13.

Lee, J-M, C Yoo, and I-B Lee. "Statistical monitoring of dynamic processes based on dynamic independent component analysis." *Chemical Engineering Science, 59*, 2004c: 2995-3006.

Lee, J-M, C.K Yoo, and I-B Lee. "Enhanced process monitoring of fed-batch penicillin cultivation using time-varying and multivariate statistical analysis." *Journal of Biotechnology, 110*, 2004a: 119-136.

Lee, J-M, CK Yoo, and I-B Lee. "Fault detection of batch processes using multiway kernel principal component analysis." *Computers and chemical engineering*, 2004b: 1837-1847.

Lee, J-M, S Qin, and I-B Lee. "Fault Detection and Diagnosis based on Modified Independent Component Analysis." *AIChE Journal, 52, 10*, 2006: 3501-3514.

Leger, R, W Garland, and W Poehlman. "Fault detection and diagnosis using Statistical Control Charts and Artificial Neural Networks." *Artificial Intelligence in Engineering, 12*, 1998: 35-47.

Li, J, and P Cui. "Improved kernel fisher discriminant analysis for fault diagnosis." *Expert Systems with Applications, 36*, 2009: 1423-1432.

Li, R, and G Rong. "Fault isolation by Patial Dynamic Principal Component Analysis in Dynamic Process." *Chinese Journal of Chemical Engineering, 14, 4*, 2006: 486-493.

Li, R.F, and X.Z Wang. "Dimension reduction of process dynamic trends using independent component analysis." *Computers and Chemical Engineering, 26*, 2002: 467-473.

Liang, J, and R. Du. "Model-based fault detection and diagnosis of HVAC systems using Support vector machine method." *International Journal of Refrigeration, 30*, 2007: 1104-1114.

Lin, W, Y Qian, and X Li. "Nonlinear dynamic principal component analysis for on-line process monitoring and diagnosis." *Computers and Chemical Engineering, 24*, 2000: 423-429.

Lo, C.H, Y.K Wong, and A.B Rad. "Model-based fault diagnosis in continuous dynamic systems." *ISA Transactions, 43*, 2004: 459-475.

Louwerse, D.J, and A.K Smilde. "Multivariate statistical process control of batch processes based on three-way models." *Chemical Engineering Science, 55*, 2000: 1225-1235.

Ma, C.Y, and X.Z Wang. "Inductive data mining based on genetic programming: Automatic generation of decision trees from data for process historical data analysis." *Computers and Chemical Engineering, 33*, 2009: 1602-1616.

MacGregor, J, and P Nomikos. *Monitoring Batch Processes in Batch Processing Systems Engineering: Current status and future directions (NATO ASI Series F).* Heidelberg: Springer-Verlag, 1992.

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

MacGregor, J.F, C Jacckle, C Kiparissides, and M Koutondi. "Process monitoring and diagnosis by multiblock PLS methods." *American Institute of Chemical Engineers Journal, 40, 5*, 1994: 826-838.

MacGregor, J.F, T.E Marlin, J Kresta, and B Skagerberg. "Multivariate statistical methods in process analysis and control." *Chemical process control –CPCIV. CACHE-AIChE*, 1991: 79-100.

Manning, C, and H Schütze. *Foundations of Statistical Natural Language Processing.* The MIT Press, 1999.

Massart, D.L, B.G.M Vandeginste, and L.M.C Buydens. *Handbook of Chemometrics and Qualimetrics. Part A.* Netherlands: Elsevier, 1997.

Massoumnia, M.A. "A geometric apporoach to the synthesis of failure detection filters." *IEEE Transactions on Automatic Control, AC-31.* 1986. 839-846.

Maurya, M.R, R Rengaswamy, and V Venkatasubramanian. "Fault diagnosis by qualitative trend analysis of the principal components." *Chemical Engineering Research and Design, 83, A9*, 2005: 1122-1132.

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

Montgomery, D.C. *Introduction to Statistical Quality Control, 2nd edn.* New York: John Wiley, 1991.

Mu, F, and V Venkatasubramanian. "Online HAZOP analysis for abnormal event management of batch process." *Computer Aided Chemical Engineering, 14*, 2003: 803-808.

Mushtaq, F, and P.W.H Chung. "A systematic HAZOP procedure for batch processes, and its application to pipeless plants." *Journal of Loss Prevention in the Process Industries, 13, 1*, 2000: 41-48.

Musulin, E, I Yélamos, and L Puigjaner. "Integration of Principal Component Analysis and Fuzzy Logic Systems for Comprehensive Process Fault Detection and Diagnosis." *Ind. Eng. Chem. Res., 45*, 2006: 1739-1750.

Negiz, A, and A Cinar. "Statistical Monitoring of Multivariate Dynamic Processes with State-Space Models." *AIChE Journal, 43:8*, 1997: 2002-2020.

Neogi, D, and C.E Schlags. "Multivariate statistical analysis of an emulsion batch process." *Ind. Eng. Chem. Res., 37*, 1998: 3971-4979.

Nijhuis, A, S De Jong, and B Van Deginste. "Multivariate statistical process control in chromatography." *Chemometrics and Intelligent Laboratory Systems, 38*, 2003: 51-62.

Nomikos, P, and J MacGregor. "Monitoring of batch processes using Multiway Principal Component Analysis." *AIChE Journal, 40*, 1994: 1361-1375.

Nomikos, P, and J MacGregor. "Multivariate SPC charts for monitoring batch processes." *Technometrics, 37, 1*, 1995a: 41-59.

Nomikos, P, and J.F MacGregor. "Multi-way partial least squares in monitoring batch processes." *Chemometrics and intelligent laboratory systems, 30*, 1995b: 97-108.

Nucci, E.R, A.J.G Cruz, and R.C. Giordano. "Monitoring bioreactors using principal component analysis: production of penicillin G acylase as a case study." *Bioprocess Biosyst Eng, 33*, 2010: 557-564.

Oliveira, A.L.I, F.R.G Costa, and C.D.S Filho. "Novelty detection with constructive probabilistic neural networks." *Neurocomputing, 71*, 2008: 1046-1053.

Pau, L. *Failure diagnosis and performance monitoring.* New York: Marcel Dekker, 1981.

Pearson, K. "On lines and planes of closest fit to systems of points in space." *Philisophical Magazine Series B 2*, 1901: 559-572.

Peñalva, M, R Rowlands, and G Turner. "The optimization of penicillin biosynthesis in fungi." *Tibtech, 16*, 1998: 483-489.

Perezyk, M, R Biernacki, and A Kochanski. "Modeling of manufacturing processes by learning systems: The naïve Bayesian classifier versus artificial neural networks." *Journal of Materials Processing Technology,164-165*, 2005: 1430-1435.

Petti, T.F, J Klein, and P.S Dhurjati. "Diagnostic model processor: using deep knowledge for process fault diagnosis." *AIChE Journal, 36, 4*, 1990: 565-575.

Pignatello, J, E Oliveros, and A.A MacKay. "Advanced oxidation processes for organic contaminant destruction based on the Fenton reaction and related chemistry." *Critical Reviews in Environmental Science and Technology, 36*, 2006: 1-84.

Pokkinen, M, Z Flores, H Asama, I Endo, R Aarts, and P Linko. "A knowledge based system for diagnosing microbial activities during a fermentation process." *Bioprocess and Biosystems Engineering, 7*, 1992: 331-334.

Polat, K, and S Günes. "A novel approach to estimation of E. coli promoter gene sequences: Combining feature selection and least square support vector machine (FS_LSSVM)." *Applied Mathematics and Computation, 190*, 2007: 1574-1582.

Qian, Y, X Li, Y Jiang, and Y Wen. "An expert system for real-time fault diagnosis of complex chemical processes." *Expert Systems with Applications, 24, 4*, 2003: 425-432.

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

Quantrille, T.E, and Y.A Liu. "Artificial intelligence in chemical engineering." *Acaddemic Press.* San Diego, LA, 1991.

Raich, A, and A Çinar. "Statistical process monitoring and disturbance diagnosis in multivariable continuous processes." *AIChE J, 42, 4*, 1996: 995-1009.

Ramesh, T.S, J.F Davis, and G.M Schwenzer. "Knowledge-based diagnostic systems for continuous process operations based upon the task framework." *Computers and Chemical Engineering, 16, 2*, 1992: 109-127.

Rich, S.H, and V Venkatasubramanian. "Model-based reasoning in diagnostic expert systems for chemical process plants." *Computers and Chemical Engineering, 11*, 1987: 111-122.

Rich, S.H, V Venkatasubramanian, M Nasrallah, and C Matteo. "Development of a diagnostic expert system for a whipped toppings process." *Journal of Loss Prevention in the Process Industries, 2, 3*, 1989: 145-154.

Rocco, C, and E Zio. "A support vector machine integrated system for the classification of operation anomalies in nuclear components and systems." *Reliability Engineering and System Safety, 92*, 2007: 593-600.

Rokach, L. "Genetic algorithm-based feature set partitioning for classification problems, 41." *Journal pattern recognition*, 2008.

Román, R.C, O.G Hernández, and U.A Urtubia. "Prediction of problematic wine fermentations using artificial neural networks." *Bioprocess Biosyst Eng, 34*, 2011.

Ruiz, D, J.M Nougués, and L Puigjaner. "Fault diagnosis for complex chemical plants." *Computers and chemical engineering, 25*, 2001: 151-160.

Ruiz, D, J.M Nougués, Z Calderón, A Espuña, and L Puigjaner. "Neural network based framework for fault diagnosis in batch chemical plants." *Computers and Chemical Engineering, 24*, 2000: 777-784.

Rumelhart, D.E, G.E Hinton, and R.J Williams. "Learning representations by back-propagating errors." *Nature, 323*, 1986: 533-536.

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

Schröder, H-C, and R Kauer. "Regulatory requirements related to risk-based inspection and maintenance." *International Journal of Pressure Vessels and Piping, 81*, 2004: 847-854.

Schwarz, G. "Estimating the dimension of a model." *The Annals of Statistics, 6,2* , 1978: 461-464.

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

Shao, J-D, G Rong, and J Lee. "Generalized orthogonal locality preserving projections for nonlinear fault detection and diagnosis." *Chemometrics and Intellingent Laboratory Systems, 96*, 2009: 75-89.

Sharma, R, K Singh, D Singhal, and R Ghosh. "Neural network applications for detecting process faults in packed towers." *Chemical Engineering and Processing, 43*, 2004: 841-847.

Shewhart, W.A. *Economic control of quality of manufactured product.* Princeton, NJ: Van Nostrand, 1931.

Shoon, T, and J Moon. "A hybrid machine learning approach to network anomaly detection." *Information Sciences, 177*, 2007: 3799-3821.

Simola, K, U Pulkkinen, H Talja, P Karjalainen-Roikonen, and A Saarenheimo. "Comparison of approaches for estimating pipe rupture frequencies for risk-informed in-service inspections." *Reliability Engineering and System Safety, 84*, 2004: 65-74.

Singh, M, and T Markeset. "A methodology for risk-based inspection planning of oil and gas pipes based on fuzzy logic framework." *Engineering Failure Analysis*, 2009: 2098-2113.

Smilde, A, and D Doornbos. "Three-way methods for the Calibration of Chromatographic Systems: Comparing PARAFAC and Three-way PLS." *Journal of Chemometrics, 5*, 1991: 345-360.

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

Srinivasan, R, and V. Venkatasubramanian. "Automating HAZOP analysis of batch chemical plants: Part I. The knowledge representation framework." *Computers and Chemical Engineering, 22, 9*, 1998: 1345-1355.

Srinivasan, R, and V. Venkatasubramanian. "Automating HAZOP analysis of batch chemical plants: Part II. Algorithms and application." *Computers and Chemical Engineering, 22, 9*, 1998: 1357-1370.

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

Straub, D, and M Havro Faber. "Risk based inspection planning for structural systems." *Structural Safety, 27*, 2005: 335-355.

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

Sugumaran, V, G.R Sabareesh, and K.I Ramachandran. "Fault diagnosis of roller bearing using kernel based neighborhood score multi-class support vector machine." *Expert Systems with Applications, 34, 4*, 2008: 3090-3098.

Sundarraman, A, and R Srinivasan. "Monitoring transitions in chemical plants using enhanced trend analysis." *Computers and chemical engineering, 27*, 2003: 1455-1472.

Tamura, M, and S Tsujita. "A study on the number of principal components and sensitivity of fault detection using PCA." *Computers and Chemical Engineering, 31*, 2007: 1035-1046.

Tates, A.A, D.J Louwerse, A.K Smilde, G.L.M Koot, and H Berndt. "Monitoring a PVC batch process with multivariate statistical process control charts ." *Ind. Eng. Chem. Res., 38*, 1999: 4769-4776.

Tian, X, X Zhang, Deng X, and S Chen. "Multiway kernel independent component analysis based on feature samples for batch process monitoring." *Neurocomputing, 72, 7-9*, 2009: 1584-1596.

Tien, S-W, W-T Hwang, and C-H Tsai. "Study of a risk-based piping inspection guideline system." *ISA Transactions, 46*, 2007: 119-126.

Tipping, M.E, and C.M Bishop. *Mixtures of probabilistic principal component analysers.* Technical report, Aston University, UK: Neural Computing Research Group, 1997.

Tokomura, M, H Tawfeek, and Y Kawase. "Decolorization of dark brown colored coffee effluent by solar photo-Fenton reaction: Effect of solar light dose on decolorization kinetics." *Water Research, 42*, 2008: 1665-1673.

Ulerich, N.H, and G.A Powers. "Online hazard aversion and fault diagnosis in chemical processes: the diagraph + fault tree method." *IEEE Transactions on Reliability, 37, 2*, 1988: 171-177.

Umeda, T, T Kuriyama, E O'Shima, and H Matsuyama. "A graphical approach to cause and effect análisis of chemical processing systems." *Chemical Engineering Science, 35, 12*, 1980: 2379-2388.

Ündey, C, and A Çinar. "Statistical Monitoring of Multistage, Multiphase Batch Processes." *IEEE Control Systems Magazine*, 2002: 40-52.

Ündey, C, E Tatara, and A Çinar. "Intelligent real-time performance monitoring and quality prediction for batch/fed-batch cultivations." *Journal of Biotechnology, 108*, 2004: 61-77.

Ündey, C, E Tatara, and A Çinar. "Real-time batch process supervision by integrated knowledge-based systems and multivariate statistical methods." *Engineering applications of artificial intelligence, 16*, 2003: 555-566.

Ündey, C, S Ertunç, and A Çinar. "Online batch /fed-batch process performance monitoring, quality prediction and variable-contribution analysis for diagnosis." *Ind.Eng. Chem. Res., 42*, 2003: 4645-4658.

Ündey, Cenk, Eric Tatara, and A Cinar. "Intelligent real-time performance monitoring and quality prediction for batch/fed-batch cultivations." *Journal of Biotechnology, 108*, 2004: 61-77.

Ungar, L.H, B.A Powell, and S.N Kamens. "Adaptive networks for fault diagnosis and process control." *Computers and Chemical Engineering, 14, 4-5*, 1990: 561-572.

Vaidhanathan, R, and V Venkatasubramanian. "Diagraph-based models for automated HAZOP análisis." *Reliability Engineering and Systems Safety, 50, 1*, 1995: 33-49.

Van Rijsbergen, C.J. *Information retrieval (2nd ed.).* London: Butterworths, 1979.

Van Sprang, E, H Ramaker, J Westerhuis, S Gurden, and A Smilde. "Critical evaluation of approaches for on-line batch process monitoring." *Chemical Engineering Science, 57*, 2002: 3979-3991.

Vapnik, V.N. *Statistical Learning Theory.* New York: Wiley, 1998.

Varga, T, F Szeifert, J Réti, and J Abonyi. "Decision tree based qualitative analysis of operating regimes in industrial production processes." *Computer Aided Chemical Engineering, 25* (Elsevier), 2008: 1039-1044.

Venkatasubramanian, V. "Inexact reasoning in expert systems: a stochastic parallel network approach." *Proceedings of the second conference on artificial intelligence applications.* Miami, FL, 1985. 191-195.

Venkatasubramanian, V, and K Chan. "A neural network methodology for process fault diagnosis." *American Institute of Chemical Engineers Journal, 35, 12*, 1989: 1993-2002.

Venkatasubramanian, V, and S.H Rich. "An object-oriented two-tier architecture for integrating compiled and deep-level knowledge for process diagnosis." *Computers and Chemical Engineering, 12, 9-10*, 1988: 903-921.

Venkatasubramanian, V, J Zhao, and S Viswanathan. "Intelligent systems for HAZOP analysis of complex process plants." *Computers and Chemical Engineering, 24, 9-10*, 2000: 2291-2301.

Venkatasubramanian, V, R Rengaswamy, and S Kavuri. "A review of process fault detection and diagnosis Part II: Qualitative models and search strategies." *Computers and chemical engineering, 27*, 2003b: 313-326.

Venkatasubramanian, V, R Rengaswamy, K Yin, and S Kavuri. "A review of process fault detection and diagnosis Part I: Quantitative model-based methods." *Computers and chemical engineering*, 2003a: 293-311.

Venkatasubramanian, V, R Rengaswamy, S Kavuri, and K Yin. "A review of process fault detection and diagnosis Part III: Process history based methods." *Computers and Chemical Engineering, 27*, 2003c: 327-346.

Verron, S, T Tiplica, and A Kobi. "Fault detection and identification with a new feature selection based on mutual information." *Journal of Process Control, 18, 5*, 2008: 479-490.

Villez, K, K Steppe, and D.J.W De Pauw. "Use of Unfold PCA for on-line plant stress monitoring and a sensor failure detection." *Biosystems Engineering, 103*, 2009: 23-34.

Vora, N, S.S Tambe, and B.D Kulkarni. "Counterpropagation neural networks for fault detection and diagnosis." *Computers and Chemical Engineering, 21*, 1997: 177-185.

Watanabe, K, I Matsura, M Abe, M Kubota, and D.M Himmelblau. "Incipient fault diagnosis of chemical processes via artificial neural networks ." *American Institute of Chemical Engineers Journal, 35, 11*, 1989: 1803-1812.

Westerhuis, J, T Kourti, and J MacGregor. "Comparing alternative approaches for multivariate statistical analysis of batch process data." *Journal of Chemometrics,13*, 1999: 379-396.

Westerhuis, J.A, S.P Gurden, and A.K Smilde. "Generalized contribution plots in multivariate statistical process monitoring." *Chemometrics and Intelligent Laboratory Systems, 51*, 2000: 95-114.

Widodo, A, and A Yang. "Wavelet support vector machine for induction machine fault diagnosis based on transient current signal." *Expert Systems with Applications, 35, 1-2*, 2008: 307-316.

Willsky, A.S. "A survey of design methods for failure detection in dynamic systems." *Automatica, 12*, 1976: 601-611.

Wise, B, N Gellagher, S Butler, D White Jr, and G Barna. "A comparison of principal component analysis, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process." *Journal of Chemometrics, 13*, 1999: 379-396.

Wo, M, W Gui, D Shen, and Y Wang. "Export fault diagnosis using role models with certainty factors for for the leaching process." *Proceedings of the third world congress on intelligent control and automation, volume 1.* Hefei, China, 2000. 238-241.

Wold, H. "Non-linear estimation by iterative least squares procedures." In *Research papers in Statistics*, by F David. New York: Wiley, 1966.

Wold, H. "Soft modeling, the basic design and some extensions." In *System under indirect observations*, by Wold H. Joreskog K. Amsterdam: North Holland, 1982.

Wold, S, N Kettaneh, H Fridén, and A Holmberg. "Modelling and diagnostics of batch processes and analogous kinetic experiments." *Chem. Intell. Lab. Syst, 44*, 1998: 331-340.

Wold, S, P Geladi, K Esbensen, and J Öhman. "Multi-way principal components and PLS-analysis." *J. Chemometrics, 1*, 1987: 47-56.

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

Yélamos, I, M Graells, L Puigjaner, and G Escudero. "Simultaneous fault diagnosis in chemical plants using a MultiLabel approach." *AIChE Journal, 53, 11*, 2007: 2871-2884.

Yong, M, X Zheng, Y Zheng, S Youxian, and W Zheng. "Fault diagnosis based on Fuzzy Support Vector Machine with Parameter Tuning and Feature Selection." *Chinese Journal of Chemical Engineering, 15, 2*, 2007: 233-239.

Yoo, C, and P.A Vanrolleghem. "Multivariate analysis and monitoring of sequencing batch reactor using multiway independent component analysis." *Computer Aided Chemical Engineering, 18*, 2004: 859-864.

Yoo, C, D Lee, and P.A Vanrolleghem. "Application of multiway ICA for on-line process monitoring of a sequencing batch reactor." *Water Research, 38, 7*, 2004b: 1715-1732.

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

Yoo, C, P Lee, P.A Vanrolleghem, and I Lee. "On-line monitoring of batch processes using multiway independent component analysis." *Chemometrics and Intelligent Laboratory Systems, 71, 2*, 2004a: 151-163.

Yoon, S, and J.F MacGregor. "Fault diagnosis with multivariate statistical models part I: using steady state fault signatures." *Journal of Process Control,11*, 2001: 387-400.

You, J-S, H-T Kuo, and W-F Wu. "Case studies of risk-informed inservice inspection of nuclear piping systems." *Nuclear Engineering and Design, 236*, 2006: 35-46.

Young, P. "Parameter estimation for continuous time models –a survey." *Automatica, 17, 1*, 1981: 23-39.

Zhang, J, and P Roberts. "Process fault diagnosis with diagnostic rules based on structural decomposition." *Journal of Process Control, 1, 5*, 1991: 259-269.

Zhang, Y. "Enhanced statistical analysis of nonlinear processes using KPCA, KICA and SVM." *Chemical Engineering Science, 64*, 2009: 801-811.

Zhaoyang, T, L Jianfeng, W Zongzhi, Z Jianhu, and H Weifeng. "An evaluation of maintenance strategy using risk based inspection." *Safety Science, 49*, 2011: 852-860.

Zhou, Y, J Hahn, and M Sam Mannan. "Fault detection and classification in chemical processes based on neural networks with feature extraction." *ISA Transactions, 42.* 2003. 651-664.