INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY

MONTERREY CAMPUS

GRADUATE PROGRAM IN MECHATRONICS AND INFORMATION TECHNOLOGIES

FAULT DETECTION AND DIAGNOSIS IN A HEAT EXCHANGER USING DYNAMIC PRINCIPAL COMPONENT ANALYSIS AND DIAGNOSTIC OBSERVERS

THESIS

PRESENTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

MASTER OF SCIENCE IN AUTOMATION

BY

JUAN CARLOS TUDÓN MARTÍNEZ

MONTERREY, NUEVO LEÓN, MÉXICO , DECEMBER 2008

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

DIVISION OF MECHATRONICS AND INFORMATION TECHNOLOGIES

GRADUATE PROGRAM IN MECHATRONICS AND INFORMATION TECHNOLOGIES

THE MEMBERS OF THE THESIS COMMITTEE HEREBY APPROVE THE THESIS OF
JUAN CARLOS TUDÓN MARTÍNEZ, BSc.
AS A PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF SCIENCE IN

AUTOMATION

COMMITTEE MEMBERS

RUBÉN MORALES MENÉNDEZ, PH.D.
THESIS ADVISOR
ITESM, CAMPUS MONTERREY

FEDERICO GUEDEA ELIZALDE, PH.D.
SYNODAL
ITESM, CAMPUS MONTERREY

RICARDO A. RAMÍREZ MENDOZA, PH.D
SYNODAL
ITESM, CAMPUS MONTERREY

JOAQUÍN ACEVEDO MASCARÚA, PH.D.
DIRECTOR OF RESEARCH AND GRADUATE PROGRAMS
ITESM, CAMPUS MONTERREY

MONTERREY, NUEVO LEÓN, MÉXICO, DECEMBER 2008

FAULT DETECTION AND DIAGNOSIS IN A HEAT EXCHANGER USING DYNAMIC PRINCIPAL COMPONENT ANALYSIS AND DIAGNOSTIC OBSERVERS

BY

JUAN CARLOS TUDÓN MARTÍNEZ

THESIS

PRESENTED TO THE GRADUATE PROGRAM IN MECHATRONICS
AND INFORMATION TECHNOLOGIES

THIS THESIS IS A PARTIAL REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN

AUTOMATION

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

MONTERREY, NUEVO LEÓN, MÉXICO , DECEMBER 2008

Acknowledgements

I thank to the Department of Mechatronics and Automation (Tecnológico de Monterrey, campus Monterrey) for giving to me a scholarship in order to study a Master degree.

I thank to my thesis advisor, Dr. Rubén Morales-Menéndez, for his knowledge, patience and direction. Thanks for sharing your experiences.

I thank to my synodals, Dr. Ricardo Ramírez and Dr. Federico Guedea, for their suggestions and contributions to this thesis work.

I thank to my classmates, called *Banda MAT*, for their support and friendship. Really, we shared great moments in and out of the school.

I thank to Amparo Herrera and Juan Pineda, for their support during this time in which I worked as teacher assistant.

Dedication

To my parents:

Josefina and Malaquias

All my successes will be your successes.
Thanks for your unconditional love. I am proud of you.

To Erika Rangel, my great love.

Beside to you, everything is possible. You are my inspiration. Thanks for being wonderful.

To all my family.

Thanks to my sister and brothers, sisters-in-law, brother-in-law, nephews and beautiful niece for your comprehension, affection and love. Thanks to my aunt and uncle: Socorro and Roberto, you have been my second family since I arrived to Monterrey.

Abstract

Quick detection and correct isolation of soft faults in a control system allow to improve the product quality, particularly in chemical processes, for example: an industrial heat exchanger. According to Venkatasubramanian, the fault methods can be classified as: model-based methods and historical data-based methods. In this thesis, two Fault Detection and Isolation (FDI) systems are designed and validated in the same process, i.e. in a shell and tube industrial heat exchanger. One of them is based on the Dynamic Principal Component Analysis (DPCA) method and the another one on a set of diagnostic observers. The former method requires historical data of the process, whereas, the diagnostic observers use quantitative models. Before testing both methods, they are trained in the same normal operating point. Four kinds of faults are introduced under the same process conditions in order to compare the performance of both diagnostic methods. All these fault cases are considered as soft faults in sensors or actuators; the faults are implemented with abrupt or gradual behavior. Similar metrics are defined in both FDI methods in order to analyze the desirable characteristics of any fault diagnostic system: robustness, quick detection, isolability capacity, explanation facility, false alarm rates and multiple faults identifiability. Experimental results show the principal advantages and disadvantages of both methods and allows to present a comparative table with the achieved performance of each method.

This work allows to design and development both methods in parallel. The *Recursive Least Squares (RLS)* method is used to identify the process through a *Random Binary Signal* (RBS) test. The reliable model of each fault allows to design a set of diagnostic observers. On the other hand, a statistical analysis based on historical data is designed to know the operating status. The DPCA method projects the data into two new spaces in order to detect any abnormal event using a smaller number of process variables. In this manner, two methods, based on different approaches, are tested under the same experimental data. This work shows that a set of diagnostic observers can detect a soft fault in a sensor or actuator at shorter time than the DPCA method. The diagnostic observers present a lower false alarm rate than the DPCA method, when soft faults in actuators are implemented. Furthermore, diagnostic observers can identify multiple faults, whereas the DPCA method can not associate correctly the errors to the occurred faults. However, the training and testing stages of the diagnostic observers require greater computational resources than the stages of the DPCA method.

Contents

List of Tables

List of Figures

Table 1: Description of variables.

Variable	Description	Dimensions	Domain (values)
$\mathbb{X}$	Data matrix	matrix of m $\times$ n	$\mathbb{R}^+$
$\mathbb{X}_\mathbb{D}$	Augmented data matrix	matrix of m $\times$ (n$\times$w)	$\mathbb{R}^+$
m	Observations	scalar	integer belonging to $\mathbb{R}^+$
n	Process variables	scalar	integer belonging to $\mathbb{R}^+$
r	New set of variables for reducing the dimensions of data matrix X	scalar	integer belonging to $\mathbb{R}^+$
$\bar{\mathbb{X}}$	Normalized data matrix	matrix of m $\times$ n	-1 - 1
$\bar{x}$	Normalized data vector	vector of m	-1 - 1
$\mathbb{D}$	Diagonal matrix containing standard deviation of each variable	matrix of n $\times$ n	$\mathbb{R}^+$
$\mathbf{1}$	Vector of elements equal to 1	vector of m	1
μ_{x_i}	Mean of process variable *i*	scalar	$\mathbb{R}^+$
σ_{x_i}	Standard deviation of process variable *i*	scalar	$\mathbb{R}^+$
$\mathbb{T}$	Scores matrix	matrix of m $\times$ r	$\mathbb{R}$
$\mathbb{P}$	Matrix of loading vectors	matrix of n $\times$ r	$\mathbb{R}$
$\mathbb{I}$	Identity matrix	matrix of r $\times$ r	$0 - 1$
p_j	Loading vector with j direction	vector of n	$\mathbb{R}$
$\mathbb{A}$	Correlation Matrix	matrix of n $\times$ n	$\mathbb{R}^+$
λ_j	Eigenvalue j	scalar	$\mathbb{R}^+$
$\mathbb{X}^*$	Back-transformed data matrix	matrix of m $\times$ n	-1 - 1
T^2	Hotelling's T^2 statistic	scalar	$\mathbb{R}^+$
T^2_{alpha}	T^2 threshold	scalar	$\mathbb{R}^+$
x	New data vector (normalized)	vector of n	-1 - 1
Λ	Diagonal matrix which contains the first r eigenvalues	matrix of r $\times$ r	$\mathbb{R}^+$
$F_\alpha(r,m-r)$	F-distribution with r and m-r degrees of freedom	scalar	$\mathbb{R}^+$
Q	Q statistic	scalar	$\mathbb{R}^+$
Q_α	Q threshold	scalar	$\mathbb{R}^+$
θ_i	Square sum of the higher eigenvalues than the eigenvalue i	scalar	$\mathbb{R}^+$
h_o	Coefficient for calculating Q_α	scalar	$\mathbb{R}$

continuing

Variable	Description	Dimensions	Domain (values)
c_α	Normal deviate corresponding to $(1-\alpha)$ percentile	scalar	$\mathbb{R}^+$
Con_i	Contribution of the variable i to total error	scalar	0 - 1
R_i	Residual vector of the variable i	vector of m	$\mathbb{R}$
w	Delays into the data matrix X	scalar	integer belonging to $\mathbb{R}^+$
a	Temperature measurement in the temperature transmitter	scalar	0 - 100
b	Current value in mA of the temperature transmitter	scalar	4 - 20
η	Number of eigenvalues of the matrix G	scalar	$\mathbb{R}^+$
β	Number of inputs of the state space model	scalar	$\mathbb{R}^+$
δ	Number of outputs of the state space model	scalar	$\mathbb{R}^+$
$x_p(k)$	Process states	vector of η	$\mathbb{R}^+$
$u(k)$	Inputs vector	vector of β	$\mathbb{R}^+$
$y(k)$	Outputs vector	vector of δ	$\mathbb{R}^+$
$v(k)$	Noise vector in $x_p(k)$	vector of β	$\mathbb{R}^+$
$z(k)$	Noise vector in $y(k)$	vector of δ	$\mathbb{R}^+$
$\mathbb{G}$	State matrix	matrix of $\eta \times \eta$	$\mathbb{R}$
$\mathbb{H}$	Input matrix	matrix of $\eta \times \beta$	$\mathbb{R}$
$\mathbb{C}$	Output matrix	matrix of $\delta \times \eta$	$\mathbb{R}$
$\mathbb{V}$	Noise matrix in $x_p(k)$	matrix of $\eta \times \eta$	$\mathbb{R}$
$\mathbb{Z}$	Noise matrix in $y(k)$	matrix of $\delta \times \eta$	$\mathbb{R}$
a_i	Output parameters in the ARX model	scalar	$\mathbb{R}$
b_j	Input parameters in the ARX model	scalar	$\mathbb{R}$
d	Number of delays to include in the input variable **u**	scalar	integer belonging to $\mathbb{R}^+$
$\tilde{x}_o(k)$	Observer states	vector of η	$\mathbb{R}^+$
$\mathbb{K}_e$	Observer feedback matrix	matrix of $\eta \times \delta$	$\mathbb{R}$

Chapter 1

Introduction

Early detection and diagnosis of abnormal events in industrial processes can represent economic, social and environmental profits. Jämsä-Jounela [1] describes the importance that nowadays have the process monitoring and fault diagnosis for optimizing the production chain and improving the product quality, particularly in chemical processes. Jämsä-Jounela presents the estimated annual losses (US$ 20 billion) that has the petrochemical industry in the USA by developing an inadequate management under abnormal process situations.

The process monitoring systems have been evolved continuously for getting processes with greater economic, social and environmental profits. In the area of automatic control, fault detection systems are based on an explicit model of the monitored plant. Presence of faults generates inconsistencies between the actual and expected behavior (model). In order to determine this inconsistency it is necessary some form of redundancy.

The first step for checking the inconsistencies could be to implement additional sensors which are related to expected process faults. However, the use of additional sensors or transmitters does not only increase the costs, but at the same time, the probability of faults increases because now there are more elements. Thus, the implementation of additional sensors must be clearly justified by the process behavior; even though more sensors be implemented it is possible that can not be detected the fault due to the inherent process characteristics.

On the other hand, the inconsistencies can be evaluated using some supervisory system. The two supervisory tasks: detection and diagnosis may be performed in parallel or sequentially. In some diagnostic systems, it is crucial to detect a fault and specify its location as soon as possible. In other systems, the detection task is running permanently while the diagnostic task is triggered only upon the detection of the presence of a fault.

When a diagnostic system is implemented in an online process, it is important to consider that the supervisory system must be capable to distinguish faults from disturbances. Although both signals are unknown inputs to the process, the FDI (Fault Detection and Isolation) system must be capable of detecting and isolating faults whereas disturbances are nuisances which must be ignored. Another important consideration is that it is rather difficult to detect and diagnose faults in closed loop since the control law makes a effort for regulating all deviations of the process variables without considering its source [2].

Future trends of the advanced methods of supervision must be developed for realizing the next goals:

- Early detection of small faults with abrupt or incipient behavior.
- Online supervision of engineering processes.
- Fault detection in closed loops.

Recently, Venkatasubramanian *et al.* [3], [4] and [5] present a broadly classification of the advanced supervision techniques for detecting and isolating faults: model-based methods and historical data based methods. Diagnostic observers (named for their application in detection and isolation of faults using a bank of observers) and Dynamic Principal Component Analysis (DPCA) are two monitoring methods which belong to model-based methods and historical data based methods respectively, both approaches are applied in the field of chemical processes. Kazantzis and Kravaris [6], Rajaraman *et al.* [7] and Simmani *et al.* [8] use chemical-engineering processes for validating approaches based on diagnostic observers, whereas Gudi *et al.* [9] and Verde *et al.* [10] validate their DPCA approaches in this same kind of processes, due to their nonlinear dynamic.

Diagnostic observers are based on quantitative process models and require a priori knowledge when the faults occur for developing a set of observers. A bank of observers allows to isolate multiple faults [11]. Many other approaches based on diagnostic observers propose an alternative for detecting and isolating faults in nonlinear systems [12]. For testing the design algorithm of a diagnostic observer, abrupt and gradual faults in available measurements are activated in real or simulated cases [13], [14].

DPCA method is an improvement of PCA. PCA method is a multivariate statistical analysis which offers an alternative way for detecting and isolating faults without considering a modelling effort. The basic idea of PCA is to reduce the dimensionality of a data set considering the interrelation between the variables. The main limitation of PCA is that it assumes normality and independence of the samples. *Moving* PCA [15], MultiScale PCA (MSPCA) [16], PCA based on wavelet [17] and multi-step PCA [18] are proposed approaches for overcoming the limitations of PCA.

1.1 Motivation

The principal motivation for this investigation resides in the next statements:

- The management of abnormal events in a FDI system establishes an early detection and diagnosis of faults in industrial processes. Many of these faults are originated by the operators. Bailey [19] shows that the 70% of the industrial accidents are caused by human mistakes.

- When the process has a great quantity of sensors or actuators (i.e. chemical processes), the Fault Detection and Diagnosis (FDD) task results to be very difficult.

- Generally, control systems have been developed under the assumption that the measuring elements and actuating elements associated with the control loop are of high reliability and will not fail [20]. In practice, however, sensors and actuators fail; these malfunctions are called soft faults.

- Abnormal events can have a serious impact on the plant economy, product quality and safety, productivity, and pollution level. Jämsä-Jounela [1] describes the importance that nowadays have the process monitoring and fault diagnosis for optimizing the production chain and improving the product quality, particularly in chemical processes.

1.2 Problem Description

To determine a *Fault Detection and Isolation (FDI)* system in a shell and tube industrial heat exchanger based on diagnostic observers (quantitative model) or DPCA (process historical data). The proposed FDI system must detect and isolate soft faults quickly with the highest possible robustness.

1.3 Objectives

To design, in a shell and tube heat exchanger, two different FDI systems. One of them is based on the DPCA method and the another one on a set of diagnostic observers. Both methods are designed to detect and isolate online soft faults.

To compare the performance between the DPCA method and the diagnostic observers into an industrial heat exchanger, both methods based on their FDD task. Soft faults in sensors and actuators are activated in order to analyze the desirable characteristics of the FDI systems: robustness, quick detection, isolability capacity, explanation facility, false alarm rates and multiple faults identifiability.

1.4 Related Research

Several researches are focused in heat exchangers for testing their proposed FDI methods. However, in our best knowledge, almost no one has used the DPCA method and/or the state observers for detecting faults in this kind of industrial equipment.

Recently, Gudi *et al.* [9] use DPCA for detecting faults in the Tennessee Eastman process and compare the results between DPCA and CA method. In their investigation, the dynamic counterpart of PCA is recommended for overcoming the limitations of conventional PCA specially for chemical processes which have many correlated process variables. Gudi *et al.* use the same tools to detect and isolate faults that those which are used in this thesis work. CA, comparing with DPCA, shows a greater efficiency of fault detection in terms of the shorter detection delay and lower false alarm rates. However, CA has a greater computational requirement.

Verde and Mina [21] proposed a DPCA improvement and used a simulation of a flow control valve for showing the effectiveness of the proposed methodology. The proposed method allows to extend the DPCA method for no stationary time series, thus, it is possible to reduce the false alarms during the detection stage. Verde and Mina used

the same tool for detecting faults that the used tool in this thesis work: T^2 statistic.

Similarly, Verde and Mina [10] proposed a DPCA method based on adaptive standardization for MIMO systems. This method is tested in a simulated case of three interconnected tanks system. Results allows to detect faults using Hotelling T^2 statistic and omits normal variations due to changes in process signals.

On the other hand, many other approaches based on the diagnostic observers propose an alternative for detecting and isolating faults in nonlinear systems. Rajaraman *et al.* [7], Kazantzis *et al.* [6] and Simmani *et al.* [8] test their proposed nonlinear observer in different chemical processes.

For testing the design algorithm of a diagnostic observer, abrupt and gradual faults in available measurements are activated in real or simulated cases [13], [14].

Verde [11] presents the application of a bank of unknown input-observers for detecting and locating multiple-leaks in pipelines. Similarly, in this thesis work, a set of diagnostic observers is used for isolating multiple faults in the heat exchanger.

1.5 Contributions

The contributions of this work are classified on three areas: 1) scientific, 2) implementation and 3) functionality.

- Contribution in the scientific area.

  The major contribution of this work is the design and development of two FDI systems; one of them is based on process quantitative models (a set of diagnostic observers) and the another one on process experimental data (DPCA).

  In order to design a diagnostic observer is required to have a reliable model. Using the *Recursive Least Squares (RLS)* method, an online identification system has been designed in order to obtain a state space model of the process. The RLS method allows to obtain a model of multiple-inputs and single-output through a *Random Binary Signal (RBS)* test. The state space model is used to generate an online diagnostic observer based on a predictive action. The observer feedback matrix is computed through the pole placement method. Particularly, each fault model generates its observer and all observers can be computed at the same time.

  On the other hand, a statistical analysis based on experimental data is designed to know the operating status. According to the process dynamics, it is possible to introduce delays into the data matrix for eliminating the data correlations in the time. Two new data spaces can be used for detecting any abnormal event, once the correlations are eliminated. These new spaces are called *space of the principal components* and *space of the residuals.* Finding the relation between the cumulative explained variance and the eigenvalues, it is possible to delimit both spaces. A new measurement is projected online into the new spaces. Since the control limits of both spaces are statistically known, the data projections are compared with the control limits in any sample time.

- Contribution in the implementation area.

  Both FDI systems have been implemented in order to detect and isolate soft faults into a heat exchanger. Furthermore, both methods are designed using a *Human Machine Interface (HMI)* in LabView, and they can be tested in parallel. The HMI interface is designed to display the operating conditions and to highlight any abnormal event. Process monitoring can be saved in a data file for its future use. The used sample time can be equal or different in both methods.

  All fault cases are implemented online under same conditions in both diagnostic methods. Particularly, four fault cases have been designed in order to test the FDI methods; these kinds of soft faults are: abrupt faults in sensors, gradual faults in sensors, abrupt faults in actuators and multiple faults in sensors. Thus, it is possible to compare both approaches under the same metrics: robustness, quick detection, isolability capacity, explanation facility, false alarm rates and multiple faults identifiability.

- Contribution in the functionality area.

  The experimental results show that a set of diagnostic observers can detect a soft fault in a sensor or actuator at shorter time than the DPCA method. Both methods can isolate individual faults, however, the contribution plots indicate which variables are more associated hypothetically to the fault since there exist more fault possible cases. Diagnostic observers show a lower false alarm rate than the DPCA method when the faults generate a change in the process operating conditions. Both diagnostic methods allow to explain the fault propagation, i.e. the methods can define if the occurred fault is abrupt or gradual. Furthermore, diagnostic observers can identify multiple faults, whereas the DPCA method can not associate correctly the errors caused by faults. However, the training and testing stages of the diagnostic observers require greater computational resources than the stages of the DPCA method.

1.6 Contents

This work is organized in 6 chapters and 6 appendixes.
The motivation to this work, the objectives, contributions and related works to this investigation are presented in the CHAPTER 1.
The CHAPTER 2 describes a brief summary about principal supervision techniques, particularly the major advantages and disadvantages between the diagnostic observers (quantitative model-based method) and the DPCA (process history based method). Furthermore, a review of the state-of-the-art is presented.
The CHAPTER 3 presents the methodology for establishing a PCA/DPCA algorithm and describes the steps to design a set of diagnostic observers. The metrics for comparison of these approaches are defined in this chapter.
The CHAPTER 4 describes the used experimental system. Furthermore, the design of experiments is described for training both methods firstly and then for their validation under different tests.
The CHAPTER 5 presents the obtained results of both methods based on their FDD task. Finally, a comparison among the FDI methods is presented in this chapter.
The CHAPTER 6 presents the final conclusions of this work. Furthermore, the chapter includes recommendations for future investigation works.

Chapter 2

Theoretical Framework

In this chapter, firstly an introduction to fault detection and diagnosis is presented, next a brief summary about these supervision techniques is described as well as the major differences between these methods are highlighted. The chapter specifies, in the section 2.2.4, the major advantages and disadvantages between the model-based methods and the process history based methods, particularly the characteristics which has the design of a set of observers and the Dynamic Principal Component Analysis (*DPCA*) method, both based on their fault detection and diagnosis task. Finally, the chapter presents a review of the state-of-the-art in different approaches of DPCA and observers using simulated and real cases.

2.1 Introduction to Fault Detection and Diagnosis

Due to constant improvement about the development of automatic control systems which nowadays demand efficient processes with a greater product quality, the supervisory systems have played an important role in chemical processes engineering. In this manner, the supervisory system is designed for ensuring the plant safety and maintaining the quality of its products through the following functions: monitoring, automatic protection and supervision.

There are a wide variety of supervision techniques, for detecting and isolating faults, which have been classified into model-based methods and historical data based methods. Venkatasubramanian *et al.* [3], [4] and [5] show a review of monitoring methods, especially those applied in the field of chemical processes.

For getting the main objective without considering the supervision technique used, the advanced supervisory system must quickly detect and diagnose the occurrence of an abnormal event while the plant is still operating. Early detection and diagnosis allow to avoid abnormal event progression and to reduce productivity loss.

When the system recognizes that an abnormal event has happened, the fault detection occurs. On the other hand, when the supervisory system finds the cause of the fault and specifies its location and magnitude, the fault diagnosis occurs. In practice, the systems that contain the fault detection and diagnosis stages are referred to as FDI (Fault Detection and Isolation) systems.

As the treated field from faults is used over many different technological areas, there are many efforts for standardizing the terminology of this concept in FDI systems. The IFAC-Technical Committee SAFEPROCESS has made an effort to accept the following fault definition [2]: *"A fault is an unpermitted deviation of at least one characteristic property of the system from the acceptable, usual, standard condition."*

Gertler [22] classifies the basic models of faults in chemical engineering processes like *multiplicative* and *additive* process faults which can be presented like *sensor* or *actuator* faults. The description of these basic models of faults can be reviewed in the Appendix A.

2.2 Approaches to Fault Detection and Diagnosis

The advanced methods of fault detection and diagnosis may be classified into two major groups: those which do not utilize the model of the plant and those which do. From a modelling perspective, there are advanced methods that require accurate process models, these methods may be classified into two subgroups: **quantitative and qualitative model-based methods**. On the other hand, the **process history based methods** correspond to fault detection and diagnosis methods which do not assume any form of model information and rely only on historic process data. Venkatasubramanian *et al.* [3],[4] and [5] provide a systematic and comparative study of various diagnostic methods under different perspectives, which the most important distinguishing feature is a priori knowledge of the process. The classification of diagnostic systems, provided by Venkatasubramanian [3], is shown in Figure 2.1.

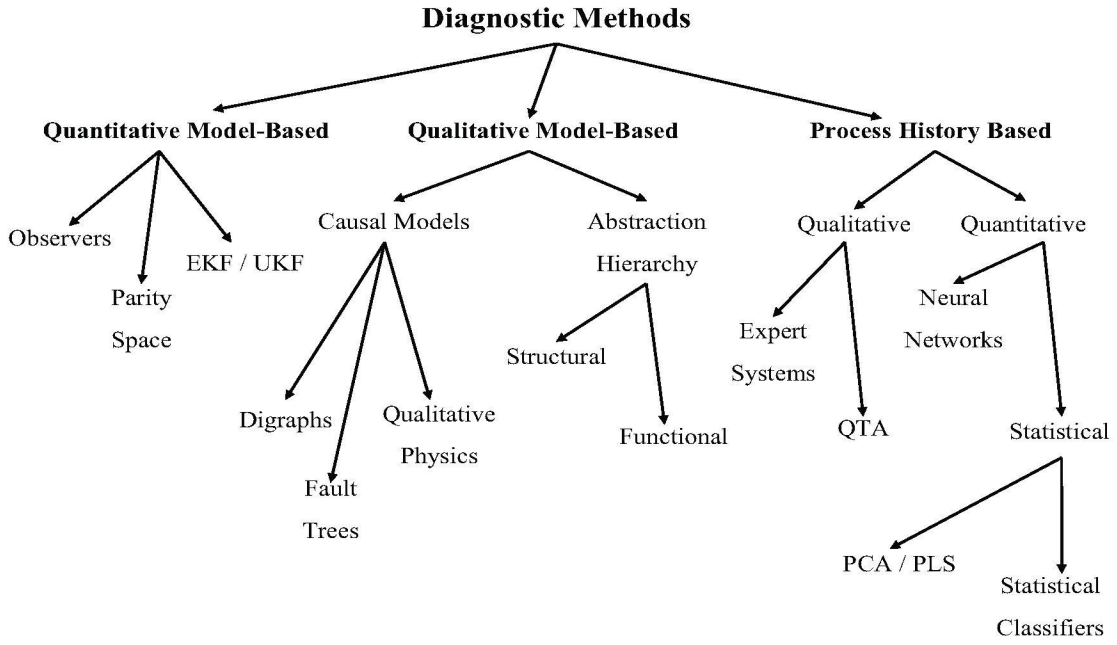

Figure 2.1: Taxonomy of diagnostic systems.

Diagnostic methods classification according to the form of the process knowledge proposed by Venkatasubramanian [3].

2.2.1 Quantitative Model-based Methods

The main objective of the quantitative model based approaches is to generate residuals that can be used for isolating process failures. These residuals are artificial signals which reflect the faults of the process through a model based on a FDI system. The FDI can detect and isolate a fault using any inconsistency expressed as residuals. This inconsistency requires one redundancy, which can be hardware or analytical redundancy. Hardware redundancy involves an extra cost and requires additional space for its implementation. On the other hand, analytical redundancy specifies mathematical relationships among different sensor measurements which can use for computing the value of a sensor measurement using measurements of other sensors. In this manner, when the computed value of the sensor is compared with the measurement sensor and exists a discrepancy, it indicates that have occurred a fault in the sensor.

For implementing an efficient system in fault detection and diagnosis, based on quantitative models, the residuals must be generated from an excellent process model. Thus, when the quantitative model correctly represents to the process, the fault isolation will be possible, otherwise the FDI will detect the fault without identifying its cause. The residuals will have "significant" values when a fault occurs and should be close to zero when the plant is in its normal operation point.

The generation of the residuals must be based in an explicit mathematical model of the process. The principal methods for generating residuals are parity relations, Kalman filters and diagnostic observers. These methods are based on analytical redundancy and generate residuals which are insensitive to uncertainties but sensitive to faults.

Parity relations allow to find a consistency between the plant models and known process inputs including the measurements.This rearranged model looks for getting the best fault isolation using the analytical redundancy which generates orthogonal residuals for different faults. Gertler and Singer [23] use parity equations for generalizing the isolability criteria and defining marginal sizes for fault alarms. In recent investigations, Chan *et al.* [24] use fully decouple parity equations for generating residuals which are sensitive to specific actuator or sensor faults; these faults, which are estimated using the recursive least-squares method, are simulated in a DC motor; whereas, Fagarasan *et al.* [25] present a FDI system based on parity equations leading to detect and isolate faults of a MIMO process which models the behavior of a heat exchanger.

Kalman filter is an optimal state estimator which allows to solve the fault diagnosis problem when the process has random fluctuations in its variables. The major applications of Kalman filters can be found in chemical processes. It is important to consider that Kalman filters are applicable only in linear processes whereas Extended Kalman Filters (EKF) can model nonlinear systems achieving first-order (Taylor series expansion) accuracy [26]; in contrast with Unscented Kalman Filters (UKF) which have a third-order accuracy (Taylor series expansion) for any nonlinearity in the process [27]. Similarly to diagnostic observers, if exist a priori knowledge of all possible failures or changes in the process, it can compute a bank of Kalman filters. Each Kalman filter will be sensitive to one fault and insensitive to remaining faults. Patwardhan *et al.* [28] and [29] use a Kalman filter for integrating a model predictive control (MPC) with a FDI system, the new scheme called fault-tolerant model predictive control was probed in a *Continuous Stirred Tank Reactor* (CSTR) simulated process for detecting and eliminating soft faults.

The diagnostic observers require a priori knowledge when the faults occur for developing a set of observers, each one of them is sensitive to a fault while insensitive to the remaining faults. Verde [11] presents the application of

a bank of unknown input-observers for detecting and locating multiple-leaks in pipelines. Recently, investigations reviews the application of diagnostic observers in nonlinear systems, Caccavale and Villani [12] propose an adaptive observer for fault diagnosis of nonlinear discrete-time systems under presence of actuator faults, the case study was developed for an industrial mechanical manipulator. Similarly, Demetriou and Armaou [13] propose an adaptive diagnostic observer for a nonlinear distributed parameter system; both abrupt and gradual fault profiles were simulated in a numerical case for evaluating the proposed fault monitoring scheme which can reduce the detection time via the use of a time-varying threshold.

In order to detect and isolate faults in Multiple-Input-Multiple-Output (MIMO) systems, reduced-order observers can be used. Zhang *et al.* [30] propose a reduced-order observer with design in the $\gamma-$stability for the system decomposition and evaluate it using a numerical example. On the other hand, Xiong and Saif [14] show that Sliding Mode Observer (SMO) can detect abrupt faults in a more effective way than Unknown Input Fault Diagnosis Observer (UIFDO), the case study was a three-phase current motor.

Recent investigations about the use of diagnostic observers applied to chemical engineering processes are presented in the section 2.3.

2.2.2 Qualitative Model-based Methods

Qualitative models are usually developed based on some fundamental understanding of the physics and chemistry of the process. This kind of models require a search strategy for recognizing the operation pattern under different process behaviors, Venkatasubramanian *et al.* [4] broadly classify them as topographic and symptomatic search techniques. Topographic searches perform malfunction analysis using a template of normal operation, whereas, symptomatic searches look for symptoms to direct the search to the fault location.

Based on the forms of qualitative knowledge, qualitative models can be classified into two major groups: causal models and abstraction hierarchies. Qualitative causal models use a reasoning tool for modelling a system and thus to capture the causal structure of the system. The three kinds of reasoning are *abductive*, *inductive* and *default*. Abduction is the generation of one or more hypothetical explanations for one observation; inductive learning allows to classify experiences into categories or concepts, whereas, default reasoning allows to generate an answer in situations where all the information is not available at a time. Diagraphs, fault trees and qualitative physics are kinds of causal models used for detecting and diagnosing faults.

Diagraphs needs a reasoning about the cause and effect relationships in the process. Cause-effect relations are represented in a graph with directed arcs, these arcs have special directions between the nodes for expressing the relation from the cause to the effect. In this manner, each node in the signed diagraph corresponds to the deviation from the steady state of a variable or event. Signed Diagraphs have been most widely used form of causal knowledge for process fault diagnosis. Recently, the literature has combined fuzzy logic and diagraphs for improving the representational scope of qualitative models. For example, Li and Wang [31] present how fuzzy diagraphs can be used for qualitative and quantitative simulation of temporal behavior of process systems.

Fault trees are used for analyzing the system reliability and safety. A fault tree is a logic tree that propagates primary events or faults to the top level event through different logic operations like AND and OR. For constructing

the fault tree, the logic operators are used for answering previous questions which generate other events in a top level. After its construction, the fault tree must be evaluated with the assessment of probability of occurrence of top level events. The biggest problem with fault trees is that its construction is prone to mistakes at different stages. The accuracy in fault trees depends on the developer capacity for representing the system in a concise manner. Recently, Amari *et al.* [32] propose a new methodology for capturing the dynamic behavior of system failure mechanisms through introducing dynamic gates into fault trees. On the other hand, Zampino presents a recent application of this method, he uses the fault tree analysis to attack a non-repeatability problem in the parameters of the NASA GRC Icing Research Tunnel (IRT); the results were critical in restoring the repeatability of the icing cloud parameters even though no single root cause of the problem could be identified [33].

Qualitative physics, called common sense reasoning about physical systems, uses the process differential equations for deriving qualitative equations or the qualitative behavior. Qualitative physics has been an area of major interest to artificial intelligence community. Recently, Zhou and Ting [34] use qualitative physics for modelling movable objects in a virtual environment with lower computational cost if the exact physics laws are used to model the behavior of moving these objects (chairs, boxes, etc.).

Abstraction hierarchies, another form of qualitative model, is based on decomposition. This decomposition allows to make inferences about the functionality of the overall system in terms of the input-output relationships solely from the laws governing the behavior of its subsystems or units. The decomposition of process system employs a topographic search which can be structural or functional. The former decomposition specifies the connectivity information of a unit, whereas, the latter specifies the output of a unit as a function of its inputs. In a recent work, Petersen [35] uses a combination of a specific model (Three Layer Model) and abstraction hierarchies for defining a set of generic inference steps underlaying the data integration and interpretation process. Both of them play a important role in supervisory control.

2.2.3 Process History based Methods

Venkatasubramanian *et al.* [5] present a review of process history based methods. These kind of methods only require a large amount of historical process data. In terms of data-extraction process, these methods are classified into qualitative or quantitative approaches. Two of the important methods that employ qualitative feature extraction are the expert systems and trend modelling approaches. An expert system first classifies the reasons for the observed problem and then offers prescriptive remedies to restore the process to normal operation. Wo *et al.* [36] have presented an expert fault diagnostic system that uses rules with certainty factors; Leung and Romagnoli [37] have presented a probabilistic model-based expert system for fault diagnosis; whereas, an expert system approach for fault diagnosis in batch processes is discussed in Scenna [38].

The Qualitative Trend Analysis (QTA) leads a suitable classification employing some kind of filtering. The result of these distinct trends helps to detect the fault earlier and lead to quick control. Recently, Rengaswamy *et al.* [39] have discussed the utility of trend modelling in control loop performance assessment.

On the other hand, the quantitative approaches can be broadly classified as non-statistical or statistical methods. Respect to the non-statistical methods, there is a considerable interest in the literature about the application of neural

networks for fault diagnosis. Neural networks used as event classifier can be reviewed in [18] and [40].

Statistical methods use a priori knowledge of class distributions to perform classification. *Partial Least Squares* (PLS) and PCA form a major component of statistical feature extraction methods. PLS method is used to extract latent variables that explain the variation in the process data and use the process variables to predict and to detect changes in the product quality variables.

Principal Component Analysis (PCA) method is a multivariate statistical analysis which offers an alternative way for detecting and isolating faults without considering a modelling effort. This method is more attractive where the available process measurements are highly correlated but only a small number of faults produce unusual patterns [41]. The basic idea of PCA is to reduce the dimensionality of a data set considering the interrelation between the process variables. The new reduced space through a new set of variables must retain as much as possible the variation presented in the data set. This new set of variables are called *principal components*. Hotelling $\mathbf{T}^2$ and $\mathbf{Q}$ statistics are usually used to monitor the variation in the principal component and residual analysis respectively. When a fault occurs in the process, both statistics be greater than their thresholds.

Conventional PCA is ideally suited for monitoring steady state processes based on the assumption that the measurements are time independent and normally distributed. Applying conventional PCA directly to dynamic systems can raise false alarms; thus, Tien *et al.* [15] present a comparative study of other PCA approaches for process monitoring and fault detection applied on data obtained from a petrochemical plant. The result showed that *moving* PCA, which uses the mean and standard deviation of a moving window for scaling purpose, appeared to outperform the other methods (*conventional*, *exponential weighted* and *adaptive* PCA) in monitoring processes with presence of faults. Zhang *et al.* [16] propose the MultiScale PCA (MSPCA) method for extending the suitability of PCA to statistically monitor processes based on autocorrelated measurements; this approach builds models for wavelet coefficients individually at different scales, the PCA models at each scale are combined to monitor the process and are expected to be more sensitive to abnormal events than conventional PCA. Similarly, a wavelet-based time-frequency approach is developed by Lu *et al.* [17] to improve PCA-based methods by extending the time-domain process features into time-frequency information; the approach was tested in two simulated cases: Tennessee Eastman process and a three-tank process.

Many other PCA improvements can be reviewed in recent investigations. Wang *et al.* [42] propose an improved PCA by introducing two new statistics of PVR (*Principal Variable Residual*) and CVR (*Common Variable Residual*) to take place the Q statistic in the conventional PCA. A simulation of CSTR process was used for demonstrating that the improved PCA with optimized sensor locations is more sensitive to small changes than the conventional one. Yang *et al.* [40] present a monitoring strategy using multiple PCA models in processes with more than one operation stage, the number of stages corresponds to number of required PCA models; a soft classifier implemented by neural network is designed for obtaining membership grades which will determine the contribution to the process. Chuanqiang *et al.* [18] propose an improvement algorithm of PCA called multi-step PCA; the basic idea of this approach is to apply the PCA method using neural networks step-by-step where a linear neural networks with k neurons can obtain k principal components (PC); after selecting the PC, the data transformed will be the new input data for the PCA method.

Dynamic Principal Component Analysis (DPCA) is an PCA improvement used particularly in chemical pro-

cesses due to their dynamic nature which results in serial and cross-correlation among the process variables [43]. The Dynamic PCA requires to augment each observation vector with a few past observations resulting a bigger matrix; this extension allows to consider the possible correlations. Similar to conventional PCA, Hotelling $\mathbf{T}^2$ and $\mathbf{Q}$ statistics are used to monitor the variation in the principal component and residual analysis respectively. Recent investigations about the use of DPCA applied to chemical engineering processes are presented in the section 2.3.

2.2.4 General Comparison Among the Fault Detection and Diagnosis Approaches

A comparison of all reviewed methods (quantitative model-based, qualitative model-based and process history based) in terms of the desirable characteristics of diagnostic systems is described by Venkatasubramanian *et al.* [5]. The principal observations are:

1. From industrial application viewpoint, the maximum number of fault diagnostic applications in process industries are based on process history based approaches, its scope is restricted to sensor failures.
2. Statistical approaches like PCA and its dynamic counterpart (DPCA), which do very well on fast detection of abnormal situations, are easier to implement in industrial applications.
3. Due to chemical process are inherently nonlinear, quantitative model-based approaches are more difficult for fault detection and diagnosis task.
4. Process history based methods can not identify multiple faults whereas qualitative and quantitative model-based methods have the multiple-fault identifiability property.
5. Particularly, diagnostic observers can isolate multiple faults through the design of a bank of observers, however each observer require a model with the possible greatest reliability.
6. Parameter drifts, which come in the form of multiplicative uncertainties, can not be identified by model-based approaches.
7. Qualitative model-based methods can provide an explanation of the path of propagation of a fault.

2.3 State-of-the-Art

2.3.1 Fault Detection and Diagnosis using DPCA

In order to detect and isolate faults, recent investigations show a trend of improving PCA methods, i.e. DPCA method. A comparative analysis between DPCA and Correspondence Analysis (CA) is presented by Gudi *et al.* [9]. Both approaches are tested in the Tennessee Eastman process. Hotelling $\mathbf{T}^2$ and $\mathbf{Q}$ statistics are used to monitor the process. Contribution plots are used for helping to fault isolation. The results show better data discrimination using CA and obviously a lower dimensional representation, greater information retained using the same *principal components* of PCA. CA compared with DPCA results to have greater efficiency of fault detection in terms of the shorter detection delay and lower false alarm rates. However, CA has a greater computational requirement. Similar results from comparative analysis between DPCA and CA can be reviewed in a experimental validation on a quadruple-tank setup presented by Gudi *et al.* [44].

Verde and Mina [21] propose a DPCA improvement. The basic idea of this investigation is to improve the DPCA performance introducing on-line average estimations of the input and output of the dynamic systems. The new parameters of the data allow extending the DPCA method for no stationary time series reducing the false alarms during the detection stage using Hotelling $\mathbf{T}^2$ statistic. A simulation of a flow control valve was used for showing the effectiveness of the proposed methodology. The results show that conventional DPCA generates false alarms even with the valve operating under normal condition whereas proposed DPCA has a correct fault detection. Similarly, Verde and Mina [10] propose a DPCA method for detecting faults. This latter approach is based on adaptive standardization for MIMO systems. The proposed fault detection was tested in a simulated case of a three interconnected tanks system. Results allows to detect faults using Hotelling $\mathbf{T}^2$ statistic and omits normal variations due to changes in process signals.

2.3.2 Fault Detection and Diagnosis using Diagnostic Observers

Many researches are focused in process monitoring and in fault detection and diagnosis task using state observers. Recently, **the diagnostic observers are designed for nonlinear systems**, Caccavale and Villani [12] propose an adaptive observer for fault diagnosis of nonlinear discrete-time systems under presence of actuator faults, the case study is developed for an industrial mechanical manipulator. Kazantzis and Kravaris [6] propose a systematic nonlinear observer for process monitoring, the performance of the proposed observer is evaluated in two chemical-engineering simulation studies: (1) estimation of the catalyst activity in a catalytic batch reactor and (2) monitoring of the heat released by the reaction occurred in an exothermic CSTR reactor. Also in the context of chemical engineering (i.e. numerical example of an exothermic CSTR), Rajaraman *et al.* [7] present a fault detection, isolation and identification algorithm for nonlinear systems based on a nonlinear observer. The proposed observer includes a parameter estimator and a fault isolation and identification filter; the approach is evaluated on a simulated reactor. The faults are simulated in temperature sensors and can be detected and isolated correctly.

Simmani *et al.* [8] suggest to exploit identified linear models in connection with dynamic observer for diagnostic purposes, although the system considered is nonlinear. The FDI strategy was applied to the simulation data of a single-shaft industrial gas turbine plant in the presence of measurement and modelling errors. On the other hand, Chen and Saif [45] propose two nonlinear observer design strategies with sliding-mode observer concepts; both the proposed schemes are capable of estimating the shape of the faults which were simulated in a numerical example.

In order to detect multiple faults in a process, a set of observers can be used, each one of them is sensitive to a fault while insensitive to the remaining faults. Verde [11] presents the application of a bank of unknown input-observers for detecting and locating multiple-leaks in pipelines.

2.3.3 FDI approaches applied in a Heat Exchanger

Many researches are focused in heat exchangers for testing their proposed FDI methods. However, in our best knowledge, almost no one has used the DPCA method and/or state observers for detecting faults in this kind of industrial equipment. Quantitative model-based methods are more used for detecting and isolating faults in real or simulated heat exchangers. Ballé *et al.* [46] present a model-based adaptive control and reconfiguration based on faults detection/diagnosis applied to a heat exchanger plant, using fuzzy-model of the process. The FDI system uses

residuals for detecting faults; these residuals are identified as fuzzy models. When a residual is equal to zero the process is free of fault, whereas, a residual deviation indicates that a fault has occurred. Using a set of fuzzy-rules contained in a special fuzzy classification tree, the system can isolate faults. Seven abrupt faults were tested whit different sizes ($3\% - 15\%$), each fault had a characteristic and unique residual incidence. Pappa *et al.* [47] propose a method for diagnosing faults in a heat exchanger. The FDI approach, based on quantitative models, uses a residual generator for creating fault signatures. The diagnostic system was tested in closed loop in real time. Due that each fault has a unique signature, the fault signatures are used for isolating faults. In order to not make the fault isolation procedure very conservative, a threshold is considered to have been exceeded if the residual is greater than 85% of time the diagnosis is performed. Experimental results show a residual deviation when sensor and actuator faults are activated (step changes); however the controller output compensates this deviation.

Thomson *et al.* [48] propose a fault diagnosis method that provides early detection of fouling of a combined heat and power (CHP) unit. Static-quantitative model based techniques have been used to monitor the operation of the heat exchangers. For detecting faults, residuals of a logarithmic mean temperature difference model are used for creating a chart based on Statistical Process Control (SPC). The coefficient UA (overall heat transfer coefficient and the transversal area) forms the basis of the chart providing information about the independent condition of the heat transfer characteristics of a heat exchanger under different operating conditions. If fouling increases, UA will reduce; if fouling reduces, UA will increase. Simulation results demonstrate the effectiveness of the method for detecting faults and distinguishing between heat exchanger fouling (faults) and pump degradation (disturbances).

Peng *et al.* [49] present a design procedure for residual generation using parity relation design and residual evaluation via a GLR (*Generalized Likelihood Ratio*) test. The proposed procedure provides a correct detection and isolation task of sensor and actuator faults classified as bias change (step) and drift (ramp). The performance in multiple fault detection is assured under the hypothesis that the number of fault modes is smaller than the number of measurements using structured residuals; however a GLR test in structured residuals requires more investigation and it is not presented in this paper. A$\ddot{i}$touche *et al.* [50] propose a fault detection system based on the GLR method, this method allows to estimate the fault magnitude. A set of interconnected process heat exchangers was used for testing faults in multiple sensors. The detection task can be achieved through a residual generation whereas the fault isolation is determined by maximizing likelihood ratio (GLR)of the fault magnitude. Experimental results show the correct detection and isolation of three additive faults. These faults are step changes with different magnitudes ($20\% - 30\%$) and they have been biased simultaneously in two different flow sensors and one temperature sensor.

Non conventional methods have been used too for process monitoring and fault detection. Morales-Menendez *et al.* [51] propose an algorithm for predicting the probability distribution of different heat exchanger states (faults), the result is used for adjusting the process control system. Weyer *et al.* [52] propose a grey box model-based method for fault diagnosis. The method is based on a first principle model of a heat exchanger and on a grey-box model of the fault. The fault analyzed in this paper corresponds to the deterioration of the heat exchanger surface by aging (settling of $CaCO_3$). With age, $CaCO_3$ "stones" can cause abrupt positive jumps in the heat exchanger coefficients. Using a cumulative sum (CUSUM) test, Weyer *et al.* can detect the settled material breakage fault when the heat exchanger coefficients rise sharply; these coefficients are tracking through a recursive least-squares estimator with forgetting factor. The simulated results shows a correct detection of this kind of faults and jump detector can find the fault location when all measurements of the three-celled model of the heat exchanger are available.

Himmelblau *et al.* [53] propose a detection system of leaks in a heat exchanger using passive acoustic noise. The measure of the acoustic noise is converted in an autoregressive (AR) model for preserving useful information and eliminating redundant information. A set of AR models are extracted from different leak conditions. The results of three classification strategies applied to the data are presented. The artificial neural network achieved better performance than the other classifiers (quadratic discrimination and nearest neighbors).

Table 2.1 presents a brief comparative summary of related investigations and the cases where were tested these approaches.

2.4 Summary

Firstly an introduction to fault detection and diagnosis is presented, then a brief review about principal supervision techniques is described. Furthermore, the major differences between these methods are highlighted, particularly the major advantages and disadvantages between the diagnostic observers (quantitative model-based method) and the DPCA (process history based method). Finally, the chapter presents a review of the state-of-the-art in different approaches of DPCA and diagnostic observers.

The reviewed investigations propose FDI systems, based on DPCA or diagnostic observers, capable of detecting correctly abnormal events in simulated and/or real cases. However, these works are tested under different faults and nonuniform process conditions; thus, it is impossible to establish a comparative analysis among these investigations. This work proposes the comparison between the DPCA method and a set of diagnostic observers (i.e. two methods based on different approaches) under the same experimental data. This comparative analysis will help to know the performance of both methods in parallel when sensors and/or actuators fail.

Table 2.1: Contributions to the State-of-the-Art.
Recent investigations related to DPCA method and diagnostic observers for their FDD task are presented in this table, as well as, FDD approaches which have been tested in heat exchangers. The symbol ✓indicates the kind of used implementation for validating the approach.

Author	**Fault Detection and Diagnosis Method**	**Application**	**Sim**	**Real**	**Year**
F. Caccavale *et al.*	Adaptive Observer	Industrial mechanical manipulator	-	✓	2004
N. Kazantzis *et al.*	Systematic nonlinear observer	Batch reactor and exothermic CSTR reactor	✓	-	2000
S. Rajaraman *et al.*	Luenberger observer	CSTR reactor	✓	-	2004
S. Simmani *et al.*	Dynamic observer	Single-shaft industrial gas turbine plant	✓	-	2008
W. Chen *et al.*	Sliding mode observer	Numerical case	✓	-	2007
C. Verde	A bank of unknown input-observers	Dynamics of pipelines	✓	✓	2001
K. P. Detroja *et al.*	Comparison between DPCA and CA	Tennessee Eastman process	✓	-	2005
K. P. Detroja *et al.*	Comparison between DPCA and CA	Quadruple-tank setup	✓	✓	2006
C. Verde *et al.*	DPCA by Average Estimation	Dynamics of a flow control valve	✓	-	2005
C. Verde *et al.*	DPCA based on adaptive standardization	Three interconnected tanks system	✓	-	2007
P. Ballé *et al.*	Fault Detection using Fuzzy- models	Heat exchanger plant	-	✓	1997
R. Krishnan *et al.*	Quantitative model	Heat exchanger	-	✓	2005
M. Thomson *et al.*	Static-quantitative model based	Heat exchanger	✓	-	2000
Y. Peng *et al.*	Parity equations	Heat exchanger	✓	-	1997
A. Aïtouche *et al.*	FDD system based on GLR method	Heat exchanger	-	✓	1998
R. Morales-Menendez *et al.*	Particle Filtering	Heat exchanger	-	✓	2003
This work	Comparison between DPCA and diagnostic observers	Heat exchanger	-	✓	2008

Chapter 3

Approaches

In this chapter, firstly the methodology for establishing a PCA/DPCA algorithm for detection and diagnosis of faults is described. Statistical Analysis of PCA/DPCA for achieving the fault detection is presented and the way for diagnosing a fault is explained. On the other hand, the methodology for designing a set of state observers based on their fault detection and diagnosis task is presented. Finally, the metrics for comparison of these approaches are defined.

3.1 Principal Component Analysis

Principal Component Analysis (PCA) method is a multivariate statistical analysis which offers an alternative way for detecting and isolating faults without considering a modelling effort. This method is more attractive where the available process measurements are highly correlated but only a small number of faults produce unusual patterns [41].

The basic idea of PCA is to reduce the dimensionality of a data set considering the interrelation between the variables. The new reduced space through a new set of variables must retain as much as possible the variation presented in the data set. This new set of variables, called *principal components*, are ordered in decreasing form where the first components retain most of the variation present in all of the original variables.

For explaining the PCA methodology, it is assumed that $\mathbb{X}$ is a matrix of m observations and n variables which are collected from an operating process. This data set represents the process normal behavior. Before applying PCA, data set must be normalized by subtracting each variable by its sample mean and then dividing each variable by its standard deviation, see equation 3.2. The notation of these variables can be reviewed in detail in the Table 1.

$$\bar{x} = (\frac{1}{m})\mathbb{X}^T\mathbf{1} \tag{3.1}$$

$$\bar{\mathbb{X}} = (\mathbb{X} - \mathbf{1}\bar{x}^T)\mathbb{D}^{-1} \tag{3.2}$$

where, $\bar{\mathbb{X}}$ is the scaled data matrix and $\bar{x}$ is a vector containing mean (μ) of each variable. $\mathbb{D}$ is a diagonal matrix containing standard deviation (σ) of each variable and $\mathbf{1}$ is a vector of appropriate dimension having elements equal to 1. The matrix arrangement of each term can be represented as:

$$\mathbb{X}_{[m\times n]} = \begin{bmatrix} x_1(0) & x_2(0) & \dots & x_n(0) \\ x_1(1) & x_2(1) & \dots & x_n(1) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(m) & x_2(m) & \dots & x_n(m) \end{bmatrix}$$

$$\mathbf{1}_{[m\times 1]} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$\bar{\mathbf{x}}_{[n\times 1]} = \begin{bmatrix} \mu_{x_1} \\ \mu_{x_2} \\ \vdots \\ \mu_{x_n} \end{bmatrix}$$

$$\mathbb{D}_{[n\times n]} = \begin{bmatrix} \sigma_{x_1} & 0 & \dots & 0 \\ 0 & \sigma_{x_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_{x_n} \end{bmatrix}$$

Thus, the scaled matrix will be,

$$\bar{\mathbb{X}}_{[m\times n]} = \begin{bmatrix} \frac{x_1(0)-\mu_{x_1}}{\sigma_{x_1}} & \frac{x_2(0)-\mu_{x_2}}{\sigma_{x_2}} & \dots & \frac{x_n(0)-\mu_{x_n}}{\sigma_{x_n}} \\ \frac{x_1(1)-\mu_{x_1}}{\sigma_{x_1}} & \frac{x_2(1)-\mu_{x_2}}{\sigma_{x_2}} & \dots & \frac{x_n(1)-\mu_{x_n}}{\sigma_{x_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{x_1(m)-\mu_{x_1}}{\sigma_{x_1}} & \frac{x_2(m)-\mu_{x_2}}{\sigma_{x_2}} & \dots & \frac{x_n(m)-\mu_{x_n}}{\sigma_{x_n}} \end{bmatrix}$$

With the scaled data matrix, a set of a smaller number ($r < n$) of variables is searched through decomposing the variance in the data. r must preserve most of the information given in these variances and covariances. The dimensionality reduction is obtained by a set of orthogonal vectors, called *loading vectors*. Loading vectors (p) are obtained by solving an optimization problem involving maximization of variance explained in the data matrix by each direction (**j**). Using the new orthogonal coordinate system, the data matrix $\bar{\mathbb{X}}$ can be transformed into a new and smaller data matrix $\mathbb{T}$, called *scores matrix*.

Isermann [54] proposed the next procedure for determination of the *loading matrix* $\mathbb{P}$ and the *scores matrix* $\mathbb{T}$:

1. Calculation of the correlation matrix $\mathbb{A}$, see equation 3.3.

$$\mathbb{A} = \bar{\mathbb{X}}^T\bar{\mathbb{X}} \tag{3.3}$$

$$\mathbb{A}_{[n\times n]} = \begin{bmatrix} \bar{x}_1(0)^2 + \ldots + \bar{x}_1(m)^2 & \bar{x}_1(0)\bar{x}_2(0) + \cdots + \bar{x}_1(m)\bar{x}_2(m) & \dots & \bar{x}_1(0)\bar{x}_n(0) + \ldots + \bar{x}_1(m)\bar{x}_n(m) \\ \bar{x}_2(0)\bar{x}_1(0) + \ldots + \bar{x}_2(m)\bar{x}_1(m) & \bar{x}_2(0)^2 + \ldots + \bar{x}_2(m)^2 & \dots & \bar{x}_2(0)\bar{x}_n(0) + \ldots + \bar{x}_2(m)\bar{x}_n(m) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{x}_n(0)\bar{x}_1(0) + \ldots + \bar{x}_n(m)\bar{x}_1(m) & \bar{x}_n(0)\bar{x}_2(0) + \ldots + \bar{x}_n(m)\bar{x}_2(m) & \dots & \bar{x}_n(0)^2 + \ldots + \bar{x}_n(m)^2 \end{bmatrix}$$

2. Calculation of the eigenvalues and eigenvectors. For getting the eigenvectors (*loading vectors*), it is necessary to solve an optimization problem involving maximization of the explained variance in the data t_j with,

$$t_j = \bar{\mathbb{X}} p_j \tag{3.4}$$

and the maximal variance of data t_j must be computed from the next constraint:

$$max(t_j^T t_j) = max(p_j^T \bar{\mathbb{X}}^T \bar{\mathbb{X}} p_j) = max(p_j^T \mathbb{A} p_j) \tag{3.5}$$

Such that $p_j^T p_j = 1$.

Solving the optimization problem, the eigenvalues λ_j of the matrix $\mathbb{A}$ can be calculated through the equation (3.6) which relates the eigenvalues with their eigenvectors. Table 1 shows the notation and dimensions of all used variables.

$$(\mathbb{A} - \lambda_j \mathbb{I}) p_j = 0 \text{ for j=1,...,n} \tag{3.6}$$

3. Selection of the largest (most significant) eigenvalues λ_j and corresponding eigenvectors p_j from j= 1 to j= r.

4. Determination of the transformation matrix $\mathbb{P}$, see equation (3.7).

$$\mathbb{P} = [p_1, p_2 ... p_r] \tag{3.7}$$

$$\mathbb{P}_{[n \times r]} = \begin{bmatrix} p_1(1) & p_2(1) & \dots & p_r(1) \\ p_1(2) & p_2(2) & \dots & p_r(2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(n) & p_2(n) & \dots & p_r(n) \end{bmatrix}$$

5. Calculation of the new data matrix through the equation (3.8).

$$\mathbb{T}_{[m \times r]} = \bar{\mathbb{X}}_{[m \times n]} \mathbb{P}_{[n \times r]} \tag{3.8}$$

$$\mathbb{T} = [t_1, t_2 ... t_r] \tag{3.9}$$

The matrix arrangement of $\mathbb{T}$ is represented as:

$$\mathbb{T}_{[m \times r]} = \begin{bmatrix} \bar{x}_1(0)p_1(1) + \ldots + \bar{x}_n(0)p_1(n) & \bar{x}_1(0)p_2(1) + \cdots + \bar{x}_n(0)p_2(n) & \dots & \bar{x}_1(0)p_r(1) + \cdots + \bar{x}_n(0)p_r(n) \\ \bar{x}_1(1)p_1(1) + \ldots + \bar{x}_n(1)p_1(n) & \bar{x}_1(1)p_2(1) + \cdots + \bar{x}_n(1)p_2(n) & \dots & \bar{x}_1(1)p_r(1) + \cdots + \bar{x}_n(1)p_r(n) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{x}_1(m)p_1(1) + \ldots + \bar{x}_n(m)p_1(n) & \bar{x}_1(m)p_2(1) + \cdots + \bar{x}_n(m)p_2(n) & \dots & \bar{x}_1(m)p_r(1) + \cdots + \bar{x}_n(m)p_r(n) \end{bmatrix}$$

Due $\mathbb{P}$ contains loading vectors of the principal components, $\mathbb{T}$ will be a smaller matrix than $\mathbb{X}$. As this transformation is a rotation matrix, it holds

$$\mathbb{P}^T\mathbb{P} = \mathbb{I} \tag{3.10}$$

Therefore also

$$\bar{\mathbb{X}} = \mathbb{T}\mathbb{P}^T \tag{3.11}$$

is valid. Thus, PCA decomposes the matrix $\bar{\mathbb{X}}$ as,

$$\bar{\mathbb{X}} = t_1 p_1^T + t_2 p_2^T + ... + t_r p_r^T \tag{3.12}$$

This new data matrix $\mathbb{T}$ contains all original data but with a reduced number of coordinates ($r < n$).

6. Back-transformation in the original data coordination system yields

$$\mathbb{X}^*_{[m\times n]} = \mathbb{T}_{[m\times r]}\mathbb{P}^T_{[r\times n]} = \bar{\mathbb{X}}_{[m\times n]}\mathbb{P}_{[n\times r]}\mathbb{P}^T_{[r\times n]} \tag{3.13}$$

$$\mathbb{X}^*_{[m\times n]} = \begin{bmatrix} t_1(0)p_1(1) + \ldots + t_r(0)p_r(1) & t_1(0)p_1(2) + \ldots + t_r(0)p_r(2) & \ldots & t_1(0)p_1(n) + \ldots + t_r(0)p_r(n) \\ t_1(1)p_1(1) + \ldots + t_r(1)p_r(1) & t_1(1)p_1(2) + \ldots + t_r(1)p_r(2) & \ldots & t_1(1)p_1(n) + \ldots + t_r(1)p_r(n) \\ \vdots & \vdots & \ddots & \vdots \\ t_1(m)p_1(1) + \ldots + t_r(m)p_r(1) & t_1(m)p_1(2) + \ldots + t_r(m)p_r(2) & \ldots & t_1(m)p_1(n) + \ldots + t_r(m)p_r(n) \end{bmatrix}$$

This last operation allows to get the data transformation in its original data coordination system, involving only significant variances. Figure 3.1 shows the block diagram for getting the characterization of the process normal behavior using PCA algorithm.

3.1.1 Fault Detection using PCA

The statistical model developed by PCA, from the historical data of process normal behavior, can be used for the purpose of online monitoring and fault detection. When PCA is used in real time, new scores are generated by projecting the new measurements onto the loading vectors, using the equation (3.8).

However for fault detection using PCA, Gudi *et al.* [9] show that normal operation of the process can be characterized by Hotelling's T^2 statistic. Equation (3.14) allows to generate online the T^2 statistic based on the first r loading vectors (*principal components*) retained [55]. If the value of T^2 statistic stays within its control limit then the status of the process is considered normal.

$$T^2 = x^T_{[1\times n]}\mathbb{P}_{[n\times r]}\Lambda^{-1}_{[r\times r]}\mathbb{P}^T_{[r\times n]}x_{[n\times 1]} \tag{3.14}$$

where, x is the new measurement vector taken online and Λ is a diagonal matrix which contains first r eigenvalues of

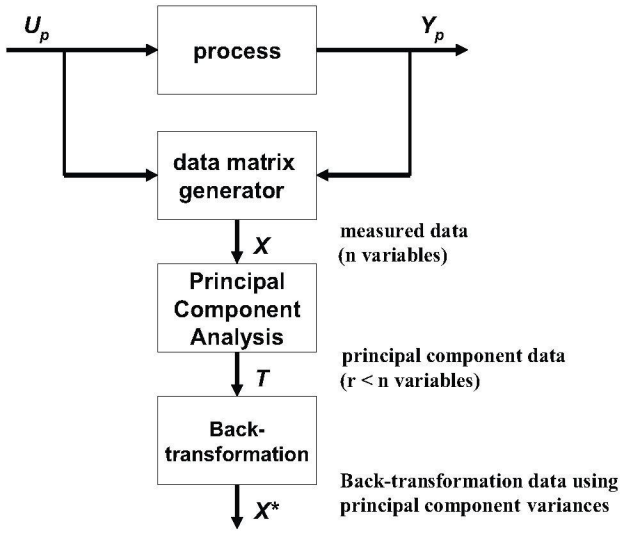

Figure 3.1: Characterization of the process normal behavior using PCA algorithm.
Generation of a set of reduced number of variables for generating $\mathbb{T}$ with only significant variables out of original $\mathbb{X}^T = [U_p^T \; Y_p^T]$ variables by PCA. Back-transformation allows to get the data into original coordinates ($\mathbb{X}^*$).

the correlation matrix ($\mathbb{A}$). T^2 is a scalar, whereas the matrix arrangement of the remainder terms can be represented as:

$$x_{[n\times 1]} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$\Lambda_{[r\times r]} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_r \end{bmatrix}$$

For establishing the threshold of T^2 statistic, Ku *et al.* [43] propose to calculate the control limit according to equation (3.15). Fault will occur when a value of T^2 statistic be greater than its control limit (T_α^2).

$$T_\alpha^2 = \frac{(m-1)r}{(m-r)} F_\alpha(r, m-r) \tag{3.15}$$

where, $F_\alpha(r, m-r)$ is the upper $100\alpha\%$ critical point of F-distribution with r and $m-r$ degrees of freedom. Since m corresponds to number of observations, F changes when the FDD system is applied online. The appendix G shows the tables of the F-distribution with 99% of confidence limit.

Due T^2 statistic only detects variation in the direction of the first r principal components, Jackson *et al.* [56] propose to monitor the variation in the residual space (components associated with the smallest singular values) using Q statistic for helping to fault detection. Both statistics must detect a process fault, however they have not the same resolution in the deviation when the fault occurs. Similarly to T^2 statistic, when a value of Q statistic be greater than its threshold (Q_α) a fault will have occurred. The values of Q statistic and its control limit can be calculated through the equations (3.16-3.17) respectively.

$$Q = [(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x]^T[(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x] \tag{3.16}$$

$$Q_\alpha = \theta_1[\frac{h_0 c_\alpha \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0(h_0 - 1)}{\theta_1^2}]^{\frac{1}{h_0}} \tag{3.17}$$

where, $\theta_i = \sum_{j=r+1}^{n}(\lambda_j)^{2i}$, $\mathrm{h}_0 = 1 - \frac{2\theta_1\theta_3}{3\theta_2^2}$ and c_α is the normal deviate corresponding to $(1-\alpha)$ percentile. All parameters from equation (3.17) do not change when the FDD system is applied online, because they depend of the characterization of process normal operation.

The notation of all used variables for computing the statistics and their control limits can be reviewed in detail in the Table 1.

3.1.2 Fault Diagnosis using PCA

Once a fault is detected by T^2 and Q statistic, the next step is to find the cause of the fault. This task can be difficult if the number of variables is large. For determining which of these variables is the most relevant to cause the fault, Miller *et al.* [57] propose to use contribution plots for helping to fault diagnosis. Contribution plots quantify the error of each process variable when the process is on abnormal status. The process variable which shows the highest contribution to the error is isolated as the most relevant to the fault which has occurred.

Contribution of each variable to residual vector, based on Q statistics, can be defined as:

$$Con_i = \frac{R_i^2}{\sum_{j=1}^{r} R_j^2} \tag{3.18}$$

where, R represents the residue generated in the residuals space [54]. The range of each error contribution oscillates between 0 and 1 where close to 1 represents a higher contribution to the total error whereas close to 0 specifies a minimal contribution to error. Do not exist a specific difference between the error contributions for associating the fault, thus the process variable with the greatest error contribution will be isolated and associated to the occurred fault. When two or more error contributions have the same value once only one fault has occurred, it will be impossible to isolate this fault using contribution plots.

The residue can be calculated by subtracting the back-transformation data ($\mathbb{X}^*$) to scaled data matrix ($\bar{X}$), as defines the equation (3.19) where $\mathbb{P}$ contains in this case the loading vectors corresponding to components with the

smallest singular values.

$$\mathbb{R}_{[m\times n]} = \bar{\mathbb{X}}_{[m\times n]} - \mathbb{X}^*{}_{[m\times n]} = \bar{\mathbb{X}}_{[m\times n]} - \bar{\mathbb{X}}_{[m\times n]}\mathbb{P}_{[n\times r]}\mathbb{P}^T_{[r\times n]} \tag{3.19}$$

The matrix arrangement of $\mathbb{R}$ can be represented as:

$$\mathbb{R}_{[m\times n]} = \begin{bmatrix} \bar{x}_1(0) - x_1^*(0) & \bar{x}_2(0) - x_2^*(0) & \dots & \bar{x}_n(0) - x_n^*(0) \\ \bar{x}_1(1) - x_1^*(1) & \bar{x}_2(1) - x_2^*(1) & \dots & \bar{x}_n(1) - x_n^*(1) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{x}_1(m) - x_1^*(m) & \bar{x}_2(m) - x_2^*(m) & \dots & \bar{x}_n(m) - x_n^*(m) \end{bmatrix}$$

3.2 Dynamic Principal Component Analysis

Vijaysai *et al.* [58] found that the main limitation of PCA is that it assumes normality and independence of the samples. Due to dynamics of the process, particularly chemical plants, the process data present a serial and cross-correlation among the variables. This violates the assumption in traditional PCA about the statistical independence of the samples. To overcome these limitations of PCA, its dynamic counterpart (DPCA) has been proposed. However, the Dynamic PCA requires to augment each observation vector with a few past observations resulting a bigger matrix. In this manner, for including the data correlation, the column space of the data matrix X must be augmented for generating a static context of dynamic relations. In the equation 3.20, w represents the quantity of delays to include.

$$\mathbb{X}_{\mathbb{D}}(t)_{[mx(nxw)]} = [X_1(t)X_1(t-1), ..., X_1(t-w), ..., X_n(t)X_n(t-1), ..., X_n(t-w)] \tag{3.20}$$

For example: taking $w = 1$, the matrix $\mathbb{X}_{\mathbb{D}}$ can be represented as:

$$\mathbb{X}_{\mathbb{D}}(t)_{[mx(nxw)]} = \begin{bmatrix} x_1(0) & - & x_2(0) & - & \dots & x_n(0) & - \\ x_1(1) & x_1(0) & x_2(1) & x_2(0) & \dots & x_n(1) & x_n(0) \\ x_1(2) & x_1(1) & x_2(2) & x_2(1) & \dots & x_n(2) & x_n(1) \\ \vdots & \vdots & \ddots & \vdots & & & \\ x_1(m) & x_1(m-1) & x_2(m) & x_2(m-1) & \dots & x_n(m) & x_n(m-1) \end{bmatrix}$$

By performing PCA on the augmented data matrix, a multivariate auto regressive (AR) model is extracted directly from the data [43]. For fault detection, the T^2 and Q statistics and their control limits proposed for PCA can be generalized directly to DPCA, using the equations (3.14-3.15-3.16-3.17). These equations can be extended for applying in online monitoring (see equations 3.21-3.22-3.23-3.24), where $\mathbb{P}, \Lambda$ and Q_α depend on the training stage.

$$T^2(t) = x^T(t)\mathbb{P}\Lambda^{-1}\mathbb{P}^T x(t) \tag{3.21}$$

$$T^2_\alpha(t) = \frac{(m-1)r}{(m-r)}F_\alpha(r, m-r)(t) \tag{3.22}$$

$$Q(t) = [(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x(t)]^T[(\mathbb{I} - \mathbb{P}\mathbb{P}^T)x(t)] \tag{3.23}$$

$$Q_\alpha = \theta_1[\frac{h_0 c_\alpha\sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0(h_0-1)}{\theta_1^2}]^{\frac{1}{h_0}} \tag{3.24}$$

Figure 3.2 shows the flow diagram for applying DPCA in the online fault detection task. Similarly, contribution plots proposed for PCA can be used for helping the fault diagnosis in DPCA, using the equation (3.18). Figure 3.3 shows a flow diagram for achieving the fault diagnosis using contribution plots.

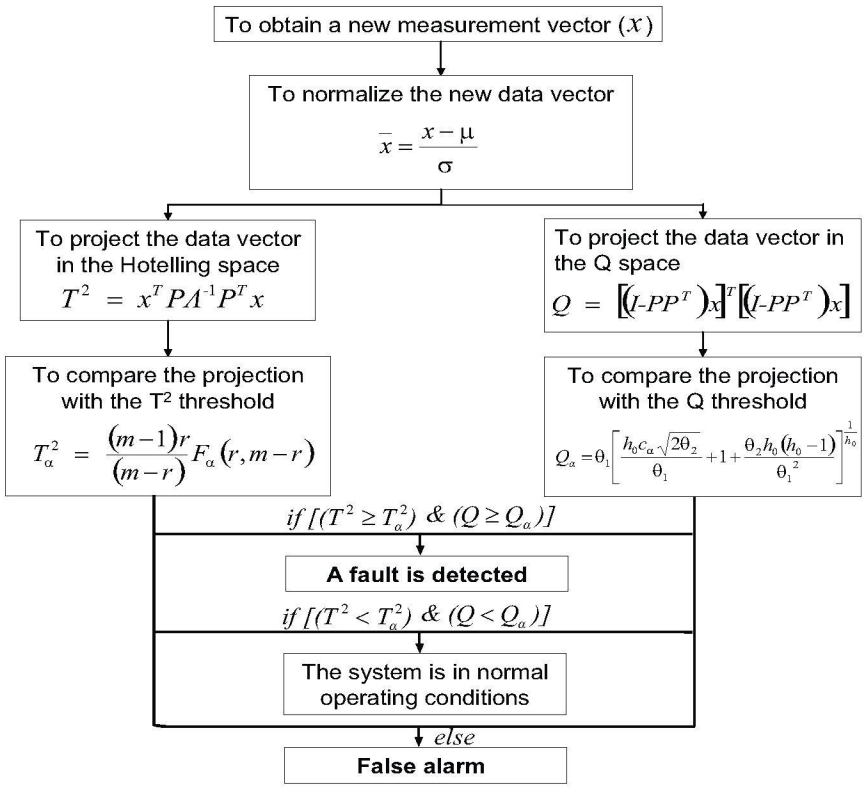

Figure 3.2: Flow diagram for detecting faults using DPCA method.
Similarly to PCA, DPCA uses the T^2 and Q statistics and their control limits for detecting faults.

3.3 Design of a Set of State Observers

As state observers use an error between the process states and adjustable model states, they are further alternative for model-based fault detection. State observers adjust the state variables according to initial conditions and to the time course of the measured input and output signals.

First step for constructing a state observer, is to get an accurate state space model of the process. The discrete state space model which can describe the process dynamic like a linear-time invariant process has the next structure,

$$x_p(k+1) = \mathbb{G}x_p(k) + \mathbb{H}u(k) + \mathbb{V}v(k) \tag{3.25}$$

$$y(k) = \mathbb{C}x_p(k) + \mathbb{Z}z(k) \tag{3.26}$$

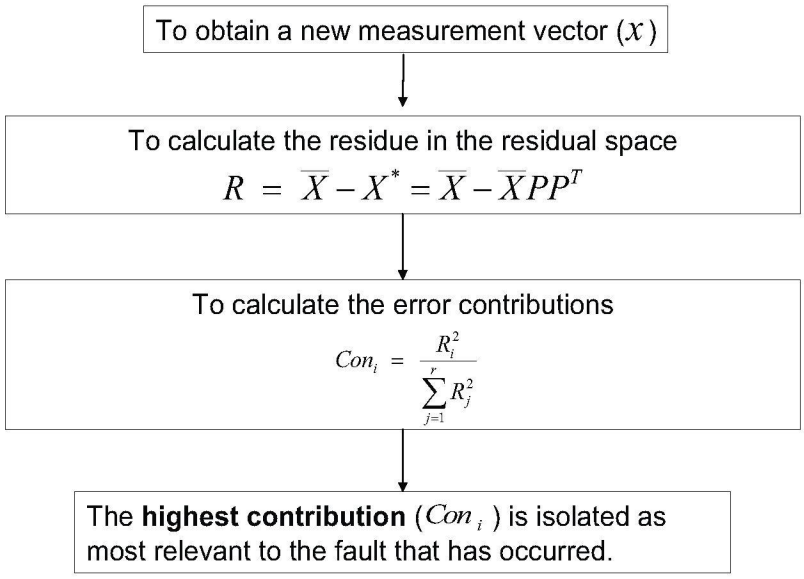

Figure 3.3: Flow diagram for diagnosing faults using Contribution Plots.
Similarly to PCA, DPCA uses contribution plots for helping the fault diagnosis

here, equations (3.25-3.26) represent the discrete states and output variables of the process respectively. Dimensions of the matrixes $\mathbb{G}$, $\mathbb{H}$ and $\mathbb{C}$ depend of the number of the inputs and outputs that has the process. $v(k)$ and $z(k)$ represent the inherent noise which affects to the process and the output, respectively. In this case $\mathbb{V}$ and $\mathbb{Z}$ are considered zero.

A Random Binary Signal (RBS) or a Pseudo-Random Binary Signal (PRBS) can be used for identifying to the process. In this case, an Autoregressive model with exogenous inputs (ARX) is extracted directly from the identification process using recursive least squares method. The structure of the ARX model is described by

$$y(k) + a_1 y(k-1) + a_2 y(k-2) + ... + a_i y(k-i) = b_1 u(k-1-d) + b_2 u(k-2-d) + ... + b_i u(k-i-d) \quad (3.27)$$

where, y represents the output variable, u the input signal and d is the number of delays to include in u. **a** and **b** correspond to parameters of the model obtained from the identification process (some of them can be zero); the number of these parameters into a model specifies the required number of past observations of the output and input signal for expressing the current value of the output variable. In MISO (Multiple Inputs - Single Output) processes, the equation (3.27) would have more input parameters (c_j, d_j, etc. for j= 0 to j=i), depending of the inputs number.

For example: considering 5 parameters in the 5 inputs and 1 output of the MISO process, the matrix arrangement of $\mathbb{G}$, $\mathbb{H}$ and $\mathbb{C}$ can be represented in its *companion form* as:

$$\mathbb{G}_{[5\times5]} = \begin{bmatrix} -a_1 & 1 & 0 & 0 & 0 \\ -a_2 & 0 & 1 & 0 & 0 \\ -a_3 & 0 & 0 & 1 & 0 \\ -a_4 & 0 & 1 & 0 & 1 \\ -a_5 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbb{H}_{[5\times5]} = \begin{bmatrix} b_1 & c_1 & \dots & f_1 \\ b_2 & c_2 & \dots & f_2 \\ \vdots & \vdots & \ddots & \vdots \\ b_5 & c_5 & \dots & f_5 \end{bmatrix}$$

$$\mathbb{C}_{[1\times5]} = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}$$

Knowing the ARX model, a great variety of forms exist for translating a parametric model to a discrete state space model [59]. With the assumption that the structure and the parameters of the discrete state space model are known, a state observer is used to reconstruct the unmeasurable state variable based on measured inputs and outputs. Thus, the state observer can be represented as

$$\tilde{x}_o(k+1) = \mathbb{G}\tilde{x}_o(k) + \mathbb{H}u(k) + \mathbb{K}_e[y(k) - \mathbb{C}\tilde{x}(k)] \tag{3.28}$$

$$\hat{y} = \mathbb{C}\tilde{x}_o(k) \tag{3.29}$$

where, $\mathbb{K}_e$ is the gain matrix of the predictive observer for reducing the differences between the dynamic model and the process. Furthermore, the observer feedback matrix $\mathbb{K}_e$ must be designed for ensuring the observer stability, Ogata [59] describes 4 different ways for calculating the observer feedback matrix (i.e. pole placement, Ackermann formula, etc.). Figure 3.4 shows the block diagram of a state observer.

Equation (3.28) can be written as

$$\tilde{x}_o(k+1) = (\mathbb{G} - \mathbb{K}_e\mathbb{C})\tilde{x}_o(k) + \mathbb{H}u(k) + \mathbb{K}_e y(k) \tag{3.30}$$

where the *eigenvalues* of $\mathbb{G}$-$\mathbb{K}_e\mathbb{C}$ are called *observer poles.*

Before designing a state observer, it is necessary to check the rank of the observability matrix (see equation 3.31). If the rank of the observability matrix is equal to the rank (η) of $\mathbb{G}$, then the system is completely observable and it is possible to generate a state observer of the process.

$$[\mathbb{C}^T \vdots \mathbb{G}^T\mathbb{C}^T \vdots \dots \vdots (\mathbb{G}^T)^{\eta-1}\mathbb{C}^T] \tag{3.31}$$

For ensuring the observer stability, the *observer poles* must be stable, which can be reached by proper design of the observer feedback matrix $\mathbb{K}_e$, e.g. by pole placement. The design of the observer feedback matrix using pole placement can be reviewed in detail in the appendix E. The matrix arrangement of $\mathbb{K}_e$ can be represented as:

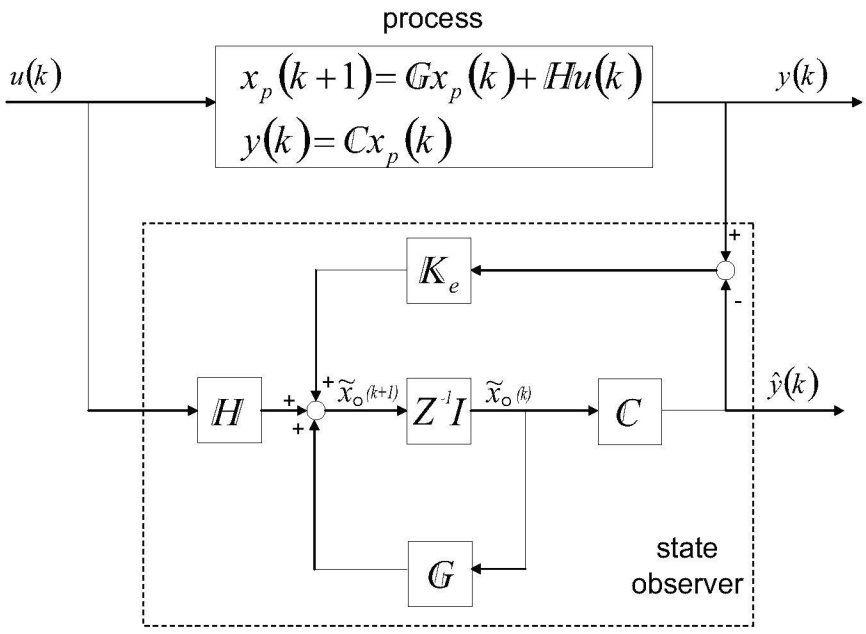

Figure 3.4: Process and state observer.
$\mathbb{K}_e$ is the feedback matrix of the state observer.
Z^{-1} represents the function variable in the domain of discrete time.

$$\mathbb{K}_{e[\eta\times\delta]} = \begin{bmatrix} k_{e1}(1) & k_{e2}(1) & \dots & k_{e\delta}(1) \\ k_{e1}(2) & k_{e2}(2) & \dots & k_{e\delta}(2) \\ \vdots & \vdots & \ddots & \vdots \\ k_{e1}(\eta) & k_{e2}(\eta) & \dots & k_{e\delta}(\eta) \end{bmatrix}$$

All used variables for the observer design can be reviewed in the Table 1, including their dimensions.

3.3.1 Fault Detection using State observers

When the observer has the same output that the process output, the observer and process states will be identical (i.e. $\tilde{x}_o = x_p$). Thus, the error equation of the observer can be computed by subtracting the equation 3.28 from the equation 3.25:

$$x_p(k+1) - \tilde{x}_o(k+1) = (\mathbb{G} - \mathbb{K}_e\mathbb{C})[x_p - \tilde{x}_o] \tag{3.32}$$

Defining $x_p - \tilde{x}_o$ as the error vector $e(k)$,

$$e(k) = x_p(k) - \tilde{x}_o(k) \tag{3.33}$$

where, the predicted error can be calculated as

$$e(k+1) = (\mathbb{G} - \mathbb{K}_e\mathbb{C})e(k) \tag{3.34}$$

In this manner, the dynamic behavior of the error signal e(k) is determined by the eigenvalues of $\mathbb{G}$-$\mathbb{K}_e\mathbb{C}$. If the matrix $\mathbb{G}$-$\mathbb{K}_e\mathbb{C}$ is a stable matrix, the error vector will converge to zero for any initial error e(0).

When an unknown input (fault) changes the process normal operation, and this one is not considered by the observer, the error signal, called residual, should be different to zero. It is considered that if the residual is close to zero (i.e. noise with $\mu = 0$ and $\sigma = 1$) the process variable is into its normal operating condition, called nominal behavior. However, a residual different to zero means that this one changes from its nominal behavior due to 2 possible reasons: error in the modelling and presence of a fault. The former reason involves residual deviations from zero value in all time instants, whereas the latter reason presents a residual deviation from its nominal behavior once the fault has occurred.

However, if the process changes its operating conditions, the residual will be deviated. This deviation can be associated to the change in the operating condition since the control valves are monitored too.

3.3.2 Fault Detection and Diagnosis using Bank of Observers

When the process can be affected by two or more faults, one possibility to distinguish between different faults, is to use special observers with different parameters. All observers are designed from different fault models and they are sensible to any fault except the used fault for their design. In this case, the fault detection and its isolation can be achieved in parallel, e.g. when a fault occurs, which is considered in one model and the remainder models does not involve it, the error in the observer which was designed from the model of the occurred fault will be zero since it considers the occurred fault, whereas, all other residuals will be deviated. It is important to consider that these faults must be independent, and they only affect to the process output independently.

Therefore, the error or residual which does not change its value will be associated with the fault that has occurred, that is, it will be isolated as most relevant to the fault that has occurred.

The fault detection time can be computed from the time instant when all residuals changed except the residual which is associated to the occurred fault.

For identifying multiple faults which occur in different time instants, it is necessary to analyze all residues. The diagnosis of the first fault will be associated by the residue which does not change its nominal value (the residual value continuous being close to zero). However, the detection and diagnosis of new faults will be determined by the residual which does not change its past behavior once the fault has occurred (in this case, the residual will have a different value to zero).

3.4 Metrics for Comparison

Venkatasubramanian *et al.* [3] present the desirable characteristics of a fault diagnostic system. Based on these characteristics, the metrics for comparison the capability of detecting and diagnosing failures, in online systems are the following:

3.4.1 Quick Detection

The diagnostic system should inform quickly in detecting process malfunctions. Without considering the fault cause, the FDI system must be sensitive to faults and insensitive to noise which can generate false alarms during normal operation.

For implemented faults, the detection time can be computed from the time interval between the time when the fault occurred and the time instant when an indicator overshoots its control limit or is deviated of its normal condition.

3.4.2 Isolability

Isolability is the ability of the diagnostic system to distinguish between different faults. FDI systems use a diagnostic classifiers for generating outputs which can recognize the cause of fault. The ability to design isolable classifiers depends to a great extent on the process characteristics.

When a fault occurs under ideal conditions free of noise and modelling uncertainties, the FDI system must identify correctly the fault origin.

3.4.3 Robustness

The FDI robustness can be analyzed when this system can be used for detecting and isolating various kinds of failures or uncertainties under different magnitudes. It is desired that the FDI performance allows to detect different faults correctly instead of failing.

Using a not-conservative threshold, the FDI system must be robust and sensitive to faults and insensitive to noise. Furthermore, the FDI robustness can be analyzed when the system is tested under different operating conditions to the training conditions.

3.4.4 Explanation Facility

Besides the ability to identify the source of malfunction, the diagnostic system should also provide explanations on how the fault originated and propagated to current situation.

Under presence of different faults in the process, the diagnostic system must be capable of explaining which was the occurred fault behavior.

3.4.5 False Alarm Rate

False alarms are considered when: (1) the FDI system does not detect an occurred fault or (2) the FDI system detects a fault which did not happen. The false alarm rate represents the number of false events which occurs in the experiment. The lowest false alarm rate in the FDI system is required.

3.4.6 Multiple fault Identifiability

The ability to identify multiple faults is an important but a difficult requirement. It is difficult problem due to the interactive nature of faults or process variables which involves these possible faults.

For identifying multiple faults, the FDI system must be able to detect the different times when occur all faults, besides that it must isolate each one of them.

3.5 Summary

Firstly the methodology for establishing a PCA/DPCA algorithm for detection and diagnosis of faults was described. Statistical analysis for detecting faults and contribution plots for isolating them are explained in detail. Then, the methodology for designing a set of state observers based on their fault detection and diagnosis task was presented. Finally, the metrics for comparison of these approaches were defined.

Chapter 4

Experimentation

4.1 Experimental Set Up

4.1.1 Industrial Heat Exchanger

The experimental system which is used for designing and testing the FDI system based on DPCA and diagnostic observers, consists in a shell and tube industrial heat exchanger. A heat exchanger, regardless of physical design, is a device built for efficient heat transfer from one fluid to another. A shell and tube heat exchanger is the most common type of heat exchanger in oil refineries and other large chemical processes. As its name implies, this type of heat exchanger consists of a shell (a large pressure vessel) with a bundle of tubes inside it. Figure 4.1 shows the industrial heat exchanger used for testing both FDI methods: DPCA and diagnostic observers.

In this process, the fluid which runs through the tubes is water (to room temperature), and low-pressure steam flows over the tubes (inside the shell). Heat is transferred from steam to the water through the tube walls. This latter results by the heat release of the steam during its condensation process in form of latent heat. In terms of process control, the heat exchanger has a complex dynamics. The non-linearity of the process and its slow transient response are the most relevant features of the system.

For FDI design, the constant monitoring of all process variables is necessary. For this task, the heat exchanger has 2 temperature sensor/transmitters, 2 flow sensor/transmitters and control valves which are connected to a data acquisition (DAQ) system. The DAQ system (NI USB-6215) is communicated with a computer, in bidirectional form, through a data-transmission protocol. The Figure 4.2 shows the experimental system and the data acquisition system.

4.1.2 Process Instrumentation

The instrumentation, which is installed in the heat exchanger, can be seen in the Figure 4.3. The elements which compose the process instrumentation are described in the Table 4.1.

All instruments use 4-20 mA signal except solenoid valves. The solenoid valves use 110 volts in alternate when

Figure 4.1: Industrial heat exchanger.

This shell and tube heat exchanger is used for testing the FDD task of both FDI methods: DPCA and diagnostic observers.

Figure 4.2: Experimental System.

Data acquisition system (NI USB-6215) allows to have a bidirectional communication between the industrial process (heat exchanger) and the computer (Labview interface).

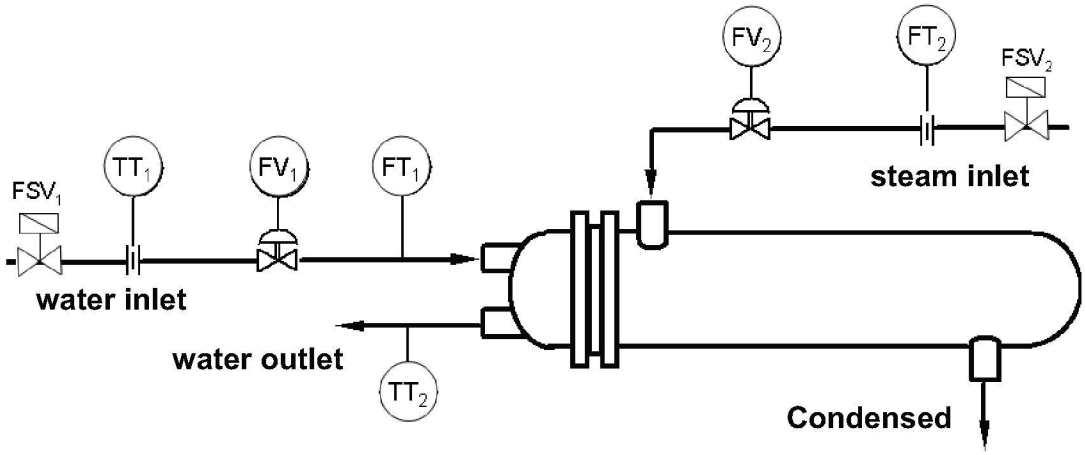

Figure 4.3: Process Instrumentation.

Instrumentation diagram of the heat exchanger shows that the water runs through the tubes whereas the steam flows inside the shell.

Table 4.1: Description of the sensor/transmitters installed in the industrial heat exchanger.

Tag name	**Description**	**Units**	**Range**
FT_1	Flow transmitter of the inlet water	mA	$4-20$
FT_2	Flow transmitter of the inlet steam	mA	$4-20$
TT_1	Temperature transmitter of the inlet water	mA	$4-20$
TT_2	Temperature transmitter of the outlet water	mA	$4-20$
FV_1	Pneumatic control valve in the feed water	mA	$4-20$
FV_2	Pneumatic control valve in the feed steam	mA	$4-20$
FSV_1	Solenoid valve in the inlet water	volts (AC)	$0-110$
FSV_2	Solenoid valve in the inlet steam	volts (AC)	$0-110$

are activated and 0 volts for their deactivation.

A conditioning of the signals is required for establishing a correct communication. Since the temperature sensors are analog, a calibration into a rank from 0°C to 100° is necessary. On the other hand, both actuators (pneumatic control valves of the steam and water) must be linearized for establishing an uniform trend between the opening percentage of the valve and the released flow.

The process instrumentation and the data acquisition system with the conditioning of the signals can be reviewed in detail in the appendix B.

4.1.3 Interface

The interface, which allows to monitor all process variables, is the last element of the experimental system. The signals supervision through a computer allows to establish a direct communication network between the heat exchanger and the user. The steam flow and water flow are analytically interpreted by percentage and the water temperature are represented in Celsius degrees.

Furthermore, the interface is capable of detecting and isolating faults when have occurred in the process through two kinds of FDI systems which were implemented in this thesis work.

The used interface is divided in three sections, the Figure 4.4 describes the screen of each section. The first section, called *Monitoring interface*, allows to monitor all process variables and to select the control mode. 1 second is used as sample time for this task.

The second section is called *FDD interface using DPCA* and the latter section, *FDD interface using diagnostic observers*. Both interfaces can have equal or different sample time. Before testing the FDD methods from interface, the methods must be trained. Each interface has a button which allows to select the training stage and another second button which allows to use the method for its FDD task.

The description of each interface section can be consulted in detail in the appendix B.5.

4.2 Design of Experiments

Both methods are designed in order to be used in open loop. The process is considered invariant in the time. Faults in sensors and actuators, called *soft faults*, have been implemented in both methods in additive form. In order to distinguish disturbances and faults in the process, it is necessary to consider the disturbances into the process model.

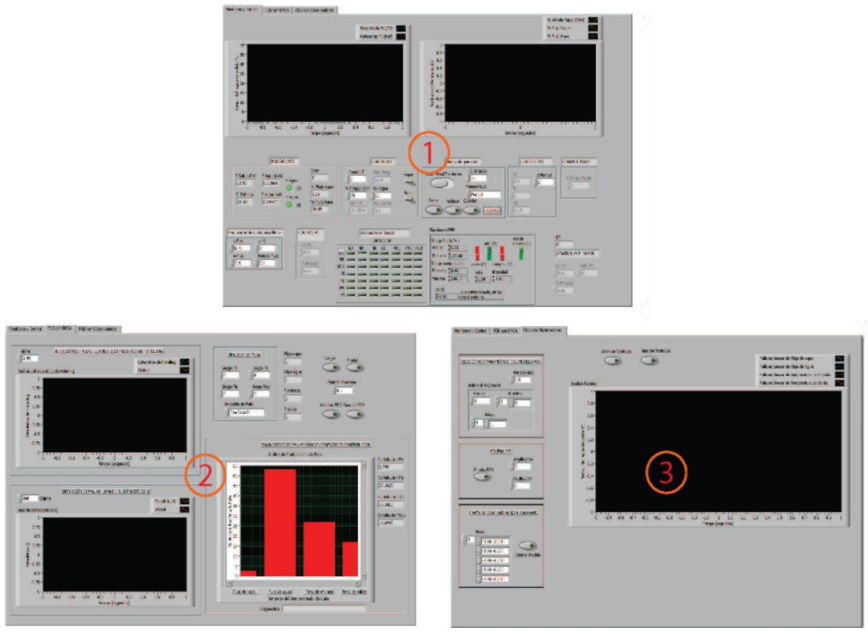

Figure 4.4: Description of the used interface.
(1) Monitoring interface. (2) FDD interface using DPCA. (3) FDD interface using diagnostic observers.

4.2.1 Training of FDD Systems

DPCA uses 1 second of sample time for this step. Due to the process dynamic which considers a high time constant, the data acquisition does not require a smaller sample time. However, the sample time must be increased if the DPCA method can not decompose the variance correctly, this occurs when the process dead time is long. In order to train the FDD system using DPCA, was taken 1900 measurement data of each sensor/transmitter of the heat exchanger. A flow diagram for training the FDD system based on DPCA can be reviewed in detail in the appendix D.1. Once the system is training, new measurements were analyzed for testing this FDD system.

In case of diagnostic observers, 5 seconds of sample time are used for obtaining the ARX models, one under each fault condition. The process identification is based on the Recursive Least Squares method. For ensuring the reliability of each ARX model, all models include a high level parametrization, particularly 5 parameters of **a** and **b** with 2 delays. Once the model is known and validated, it is possible to construct a state observer from each model. Since all models are obtained online, the diagnostic observers are reconfigured online. The observer feedback matrix in each observer is designed via pole placement with closed loop poles close to origin in the discrete space.

4.2.2 Testing of FDD Systems

In order to compare the performance of PCA/DPCA and state observers based on their fault detection and diagnosis task, additive soft faults under same conditions will be implemented into the industrial heat exchanger. Detection and diagnosis results of both methods will be tested on the metrics which were established in the chapter 3 of this thesis.

Firstly, DPCA and diagnostic observers will be tested with an abrupt fault simulating a transmitter bias. The behavior of an abrupt fault can be seen in the Figure 4.5. All sensor/transmitters signals of the industrial heat exchanger were tested with this kind of soft fault under different bias magnitudes.

Abrupt fault magnitude depends of the signal performance; for example if the sensor signal oscillates, the fault magnitude will be two times the oscillation rank. The magnitude of abrupt fault simulated in each sensor is shown in the Table 4.2.

Detection time can be directly measured of the online residual analysis whereas isolability will depend of the contribution plots for PCA/DPCA and of the observers capability for reaction under different faults. On the other hand, for testing and validating the robustness of each method, gradual faults will be simulated in all sensors signals of the industrial heat exchanger.

In this manner, an abrupt or gradual fault simulate a transmitter bias. The faults have been implemented in independent form. The behavior of both faults is shown in Figure 4.5.

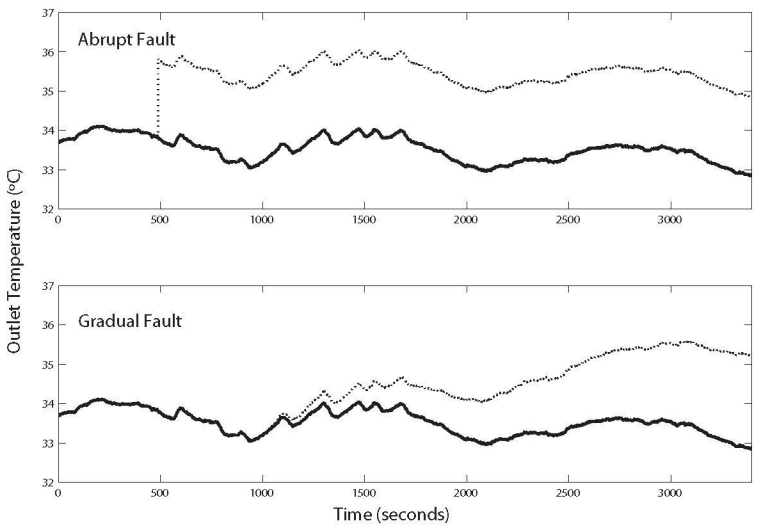

Figure 4.5: Kinds of implemented faults in all sensors and actuators signals of the heat exchanger.

In this case, an abrupt fault is simulated in the outlet temperature sensor/transmitter at time 500 (up) whereas a gradual fault is simulated at time 1000 (down).

Although is not specified a typical sensor bias in the literature, Patwardhan *et al.* [20] test a FDI system with different magnitudes of the sensor bias: 1, 3 and 5 times the standard deviation of the random error in corresponding measurement. They show that the FDI method correctly identifies the fault immediately when this one has a greater magnitude (5σ). In this case, the magnitudes for the different kinds of fault simulated in the sensors are described in the Table 4.2.

Table 4.2: Magnitudes of the implemented faults in all sensors of the industrial heat exchanger.

Sensor/transmitter	**Magnitude for abrupt fault**	**Magnitude for gradual fault (slope)**
Flow sensor of the inlet water	6% (5σ)	$0.1\%/sec$
Flow sensor of the inlet steam	8% (5σ)	$0.1\%/sec$
Temperature sensor of the inlet water	2C (8σ)	$0.1C/sec$
Temperature sensor of the outlet water	2C (8σ)	$0.1C/sec$

On the other hand, both FDD methods will be tested with soft faults in the actuators: steam and water control valves. This kind of fault is considered as a positive or negative change in the pressure of either pneumatic control valve. These faults do not be simulated, i.e. are emulated as small changes in the normal operating conditions for analyzing the method robustness. In order to implement this kind of soft fault, five different cases of actuator faults have been designed. The case 0 is considered as the normal operation point. The remainder of the cases can be reviewed in the Table 4.3. An abrupt small change in the opening percentage of a control valve is simulated as an abnormal pressure change in this valve.

Table 4.3: Design of faults in actuators.

Case	**Status of the steam control valve**	**Status of the water control valve**
0	normal	normal
1	low pressure	normal
2	high pressure	normal
3	normal	low pressure
4	normal	high pressure

Finally, the diagnostic systems will be tested with multiple faults introduced in sequence according their validation for detecting multiple faults and identifying the cause of all simulated faults. The former task must describe how was originated the soft fault (abrupt or gradual form).

4.3 Summary

In this chapter was described the experimental system. The instrumentation which is installed in the heat exchanger and the functions of the process interface were described. Finally, the design of experiments is described for training both methods firstly and then for their validation under different tests. Different tests are presented in the section 4.2.2 in order to analyze the desirable characteristics of a fault diagnostic system: robustness, quick detection, isolability capacity, explanation facility, false alarm rates and multiple faults identifiability.

Chapter 5

Analysis of Experimental Results

This chapter shows the results for the 2 approaches (DPCA and Diagnostic observers) based on their FDD (Fault Detection and Diagnosis) task. For validating the advantages and disadvantages of each method was used a shell and tube heat exchanger. Abrupt and gradual faults were simulated in all process sensors: inlet temperature sensor, outlet temperature sensor, water flow sensor and steam flow sensor. Furthermore, abrupt faults in actuators have been implemented.

The section 5.1 presents the results of detection and diagnosis of different kinds of implemented faults using the DPCA algorithm. The section 5.2 presents the fault detection and diagnosis results when is used a set of observers for this task. Finally, a comparison among the methods of fault detection and diagnosis is presented in the section 5.3; this comparative analysis is based on the next metrics: quick detection, isolability, robustness, explanation facility and multiple fault identifiability.

5.1 Fault Detection and Diagnosis using DPCA

The FDD task, using the DPCA method, requires firstly a training stage. Section 5.1.1 describes the applied training stage before testing this method. Once the FDI system are trained, it is possible to test its performance. The diagnostic results which are obtained under different fault conditions can be reviewed in the section 5.1.2.

5.1.1 Training Stage

The first step in the FDD task using DPCA is the characterization of the normal operating point. The behavior of the sensors measurements used for characterizing the normal operation point can be seen in the Figure 5.1. Based on the normal operation condition the system was represented in a new space. This space involves the correlations among the variables. It can reduce the data order through the elimination of these possible correlations. For explaining the variation completely, DPCA method augments each variable vector with past observations and stacking the data into a bigger matrix.

In this work, with one past observation of each measurement involved in the data matrix was possible to explain a higher quantity of variance including the possible auto and cross correlations (99.946% taking 5PC in DPCA

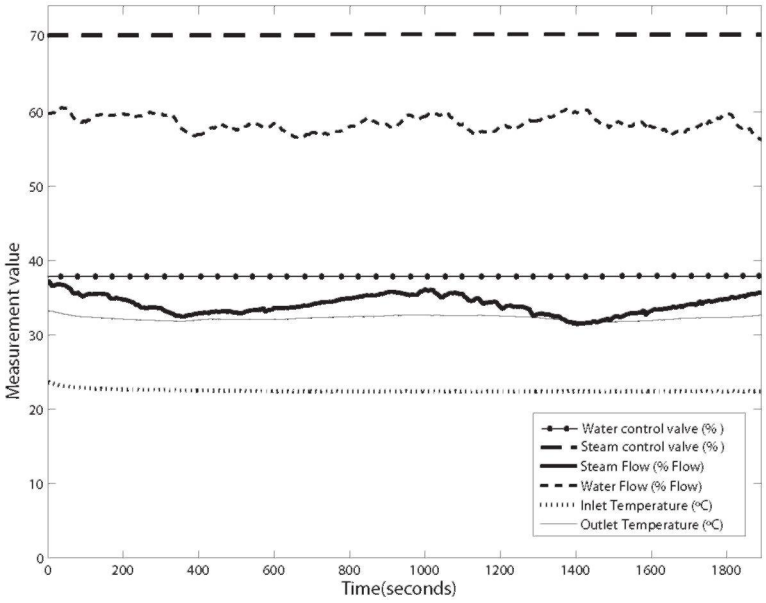

Figure 5.1: Normal operating conditions of Heat exchanger.

There are four variables at normal operating conditions: flows are in percentage and temperatures are in Celsius degrees.

versus 91.29% with 3PC in PCA). However, the number of the retained past observations in the data matrix are directly related with the used sample time. The explained variance is associated with the PC value based on the total number of principal components; when is obtained a high PC value, the percentage of explained variance will be greater than the obtained percentage by a small PC value (i.e. in the Table 5.1 the first PC explains a greater quantity of variance than the PC remainder, considering 8 PC). In this manner, the new matrix for reconstructing the original data contains eight *eigenvalues* and its respective *eigenvectors.* Thus, the data matrix has dimensions 8 *x N*.

The characterization of the process normal behavior using the DPCA algorithm (equation 3.6) is shown in the Table 5.1. The training process used for getting the characterization of the normal operation point can be reviewed in the appendix D.1. In this case, DPCA uses 1 second of sample time for this step.

Actually, there is no general fixed criterion proposed to determine how many Principal Components *(PC)* should be retained. Gudi *et al.* [9] propose as good choice, a plot of variance explained versus the number of PC. In this case, the number of retained PC will be, when the addition of one more does not improve the quality of representation significantly. In general, the major variance can be explained by retaining only first *r* principal components corresponding to the highest singular values (see Table 5.1). Applying this criterion, Figure 5.2 shows how much variance can be explained for each eigenvalue using DPCA.

In this case, with five PC's can be explained the 99.95% of total variance and one more PC does not improve the quality of representation significatively (i.e. the difference between the percentage of explained variance with 5 and 6 PC is lower than 1%). It is important to specify that 4 PC retained can not explain correctly the data variation

Table 5.1: Characterization of the normal operating point (using DPCA).
Each eigenvalue with its respective percentage of explained variance can be analyzed. With 5PC, the variance can be explained very closely to 100%.

Principal Component	Eigenvalue	Percentage of explained variance	Percentage of cumulative explained variance
1	3.1361	39.201%	39.201%
2	2.0643	25.804%	65.005%
3	1.3861	17.326%	82.331%
4	0.9925	12.406%	94.737%
5	0.4167	5.209%	99.946%
6	0.0013	0.016%	99.962%
7	0.0003	0.004%	99.966%
8	0.0002	0.003%	99.969%

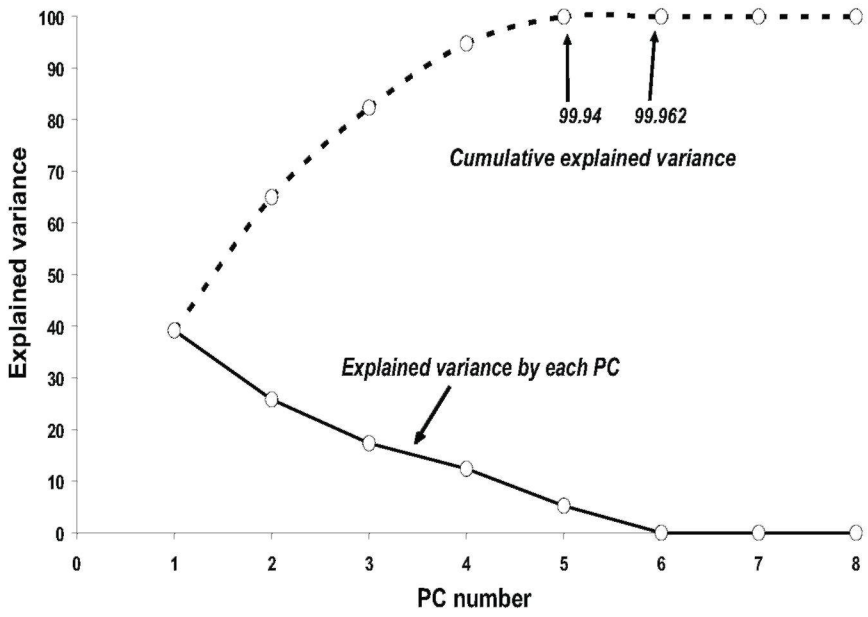

Figure 5.2: Explained variance by each PC using DPCA.
With five PC can be explained the 99.946% of total variance.

caused by a simulated fault. Appendix D.2 shows that a simulated fault in the outlet temperature sensor is impossible to isolate using 4 PC, whereas with 5 PC, contribution plot allows to isolate this fault correctly.

Using the characterization of the process normal operation (i.e. the first r eigenvalues and eigenvectors p are known), it can be tested a new measurement vector through its projection in the Q and T^2 spaces. The data projection must be compared with their respective control limits which are obtained from the training stage; the equations (3.15-3.17) are used for computing these thresholds.

5.1.2 Testing Stage

For validating the DPCA algorithm based on its FDD task, in all sensors are simulated abrupt and gradual faults from the Labview® interface; the behavior of these kinds of fault is shown in Figure 4.5 and the magnitude and slope for simulating abrupt and gradual faults in each measurement can be seen in Table 4.2. The faults are introduced in independent form at real time.

On the other hand, the DPCA method will be tested with soft faults in the actuators: steam and water control valves. These faults do not be simulated, i.e. are emulated as small changes in the normal operating conditions for analyzing the method robustness. Finally, multiple faults will be introduced in order to test the multiple fault identifiability.

Detection Step

When a fault occurs that results as a change in covariance structure of the normal operating data, the Q and T^2 statistics value will be increased; both statistics can be computed through equations (3.23-3.21) respectively, whereas their thresholds depend on the training stage (see equations 3.24-3.22). The latter asseverations are demonstrated by Jackson *et al.* [56] and Ku *et al.* [43] who proposed the Q and T^2 statistics respectively for detecting faults using the DPCA algorithm.

Figure 3.2 shows a flow diagram for detecting online faults using both statistics.

Diagnosis Step

Due that it is impossible to isolate faults only using PCA, contribution plots are used to help fault isolation [57]. Contribution plots involve the individual error of each variable, this error to be generated by the difference between the real data and the reconstructed data implies the analysis in the principal components space. The process variable which shows the highest contribution to the residuals is isolated as most relevant to the fault that has occurred. Contribution of each variable can be computed through the equation (3.18).

Do not exist a specific difference between the error contributions for associating the fault, thus the process variable with the greatest error contribution will be isolated and associated to the occurred fault. When two or more error contributions have the same value once only one fault has occurred, it will be impossible to isolate this fault

using contribution plots.

When one fault changes the normal operating conditions in the process, the error contribution can be projected in more of one process variable. This event can be associated to a specific fault, if exist redundant measurements which discard the option of a normal change in the process.

Detection and Diagnosis of Abrupt Faults in Sensors

According to design of experiments (see section 4.2) for comparing the Diagnostic Observers and DPCA method based on their FDD task, firstly the DPCA algorithm was tested with abrupt faults. An abrupt fault simulates a transmitter bias, its behavior can be seen in Figure 4.5. All sensor/transmitters signals of the industrial heat exchanger were tested with this kind of soft fault under different bias magnitudes. Table 4.2 shows the bias magnitude for each sensor.

When a fault is simulated in the outlet temperature sensor, the DPCA algorithm can explain more quantity of data variance and allows to isolate this kind of fault, unlike PCA which can not isolate this fault correctly. Figure 5.3(a) allows to show how Q and T^2 statistics overshoot their control limits at time 105 when an abrupt fault is activated in the outlet temperature sensor and these do not return to the normal condition. The abrupt fault has a simulated bias of 2°C and its behavior can be seen in Figure 4.5(up).

On the other hand, contribution plot helps correctly with the fault isolation. Figure 5.3(b) shows that 78% of total error corresponds to outlet temperature signal, whereas the remainder of error corresponds to the other sensors.

It is important to specify that this kind of fault corresponds to a sensor bias on the outlet temperature signal. Since this fault does not change the normal behavior of the other variables, the deviation is associated to a transmitter malfunction. For faults in the actuators, the normal operating conditions change in more of two sensors and the diagnosis task can be complicated.

FDD system, using DPCA, generates the same result when an abrupt fault (bias of 5s) is simulated in the steam flow sensor. Online monitoring of Q and T^2 statistics overshoot their control limits at time which the fault was introduced. Figure 5.4(a) shows that both indicators shoot up at time 8 and do not return to the normal condition.

On the other hand, contribution plot can isolate this kind of fault correctly, see figure 5.4(b). The 86% of total error corresponds to steam flow signal, whereas the other sensors does not have an error contribution greater to the 10%.

Satisfactory results are obtained too after simulating faults in the inlet temperature sensor and water flow sensor. Diagnostic results of these simulated faults can be reviewed in the appendix D.3.

Detection and Diagnosis of Gradual Faults in Sensors

For validating the robustness of detection and diagnosis of faults with the DPCA algorithm, the FDD system was tested with gradual faults in all heat exchanger sensors. A gradual fault corresponds to a soft and constant increase

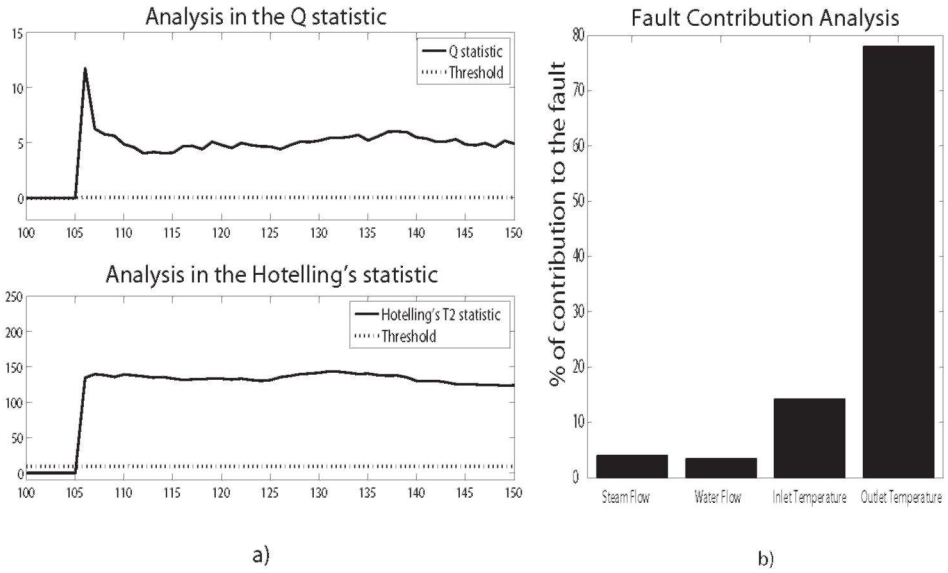

Figure 5.3: FDD analysis for an abrupt fault in the outlet temperature sensor using the DPCA method.

Q and T^2 statistics overshoot their control limits at time 105 when an abrupt fault is activated in the outlet temperature sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 78% of total error corresponds to outlet temperature signal, thus the fault was originated in this sensor/transmitter (right).

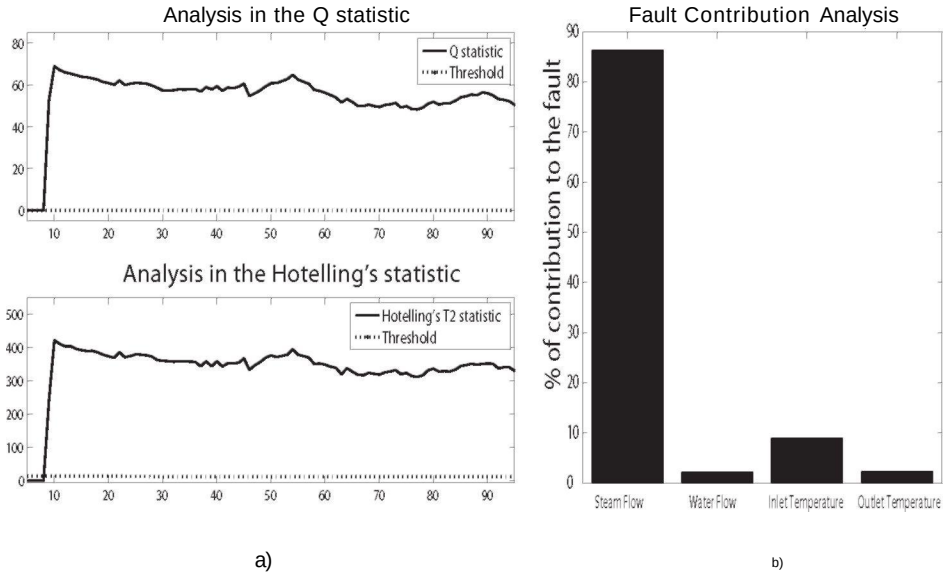

Figure 5.4: FDD analysis for an abrupt fault in the steam flow sensor using the DPCA method.

Q and T^2 statistics overshoot their control limits at time 8 when an abrupt fault is activated in the steam flow sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 86% of total error corresponds to steam flow signal, thus the fault was originated in this sensor/transmitter (right).

with specific slope which will generate a measurement deviation. The standard behavior of the gradual faults can be seen in Figure 4.5. For each measurement was established a slope of 0.1 units per second.

A satisfactory result is obtained when a gradual fault is simulated in the outlet temperature sensor at time 200. The Q and T^2 statistics overshoot their control limits and indicate the fault detection after 14 and 10 seconds respectively once the fault has happened, see figure 5.5(a). Due to established slope, when the fault achieves 1.4°C can be detected by the Q statistic whereas this fault with 1°C of bias already can be detected by the T^2 statistic. As it is evident, the behavior of both statistics corresponds to the fault behavior which was introduced.

For isolating the fault, contribution plot based on the residuals for DPCA is used. Figure 5.5(b) shows that 64% of total error corresponds to outlet temperature signal, whereas the remainder of error corresponds to the other sensors. None has an error contribution greater than the 18%. For error contributions with equal values once only one fault has occurred, it will be impossible to isolate this fault using contribution plots.

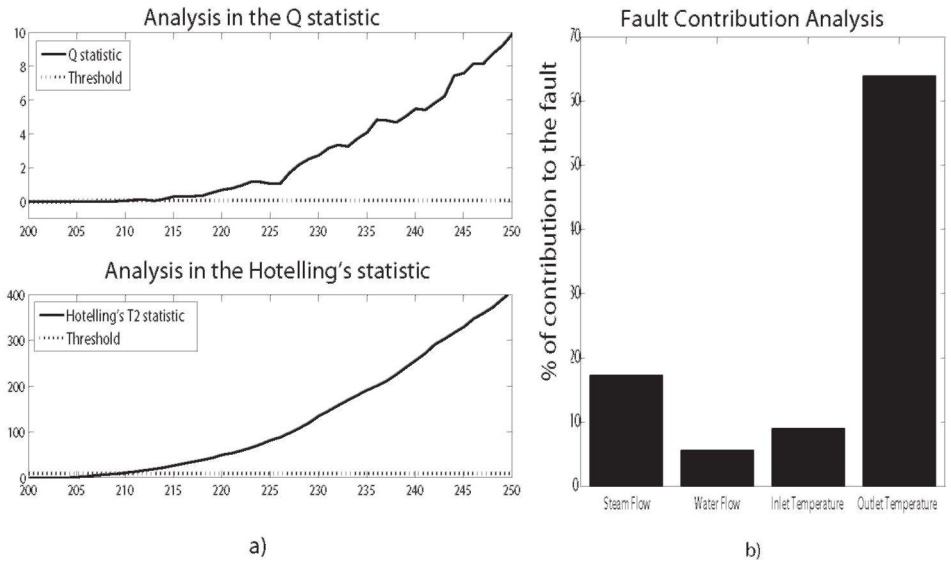

Figure 5.5: FDD analysis for a gradual fault in the outlet temperature sensor using the DPCA method.

Q and T^2 statistics overshoot their control limits after 14 and 10 seconds respectively when was introduced a gradual fault in the outlet temperature sensor at time 200 (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 64% of total error corresponds to outlet temperature signal, thus the fault was originated in this sensor/transmitter (right).

Similarly, FDD system detects and isolates a gradual fault simulated in the steam flow sensor. Online monitoring of Q and T^2 statistics overshoot their control limits after 93 and 59 seconds respectively once the fault has occurred at time 10. In this case too, the slope of the gradual fault is 0.1% per second and after 93 and 59 seconds, the total bias will be between 9.3 and 5.9 percent; value which is close to normal oscillation rank (8%). Figure 5.6(a) shows the behavior of both indicators whereas figure 5.6(b) shows the correctly fault diagnosis.

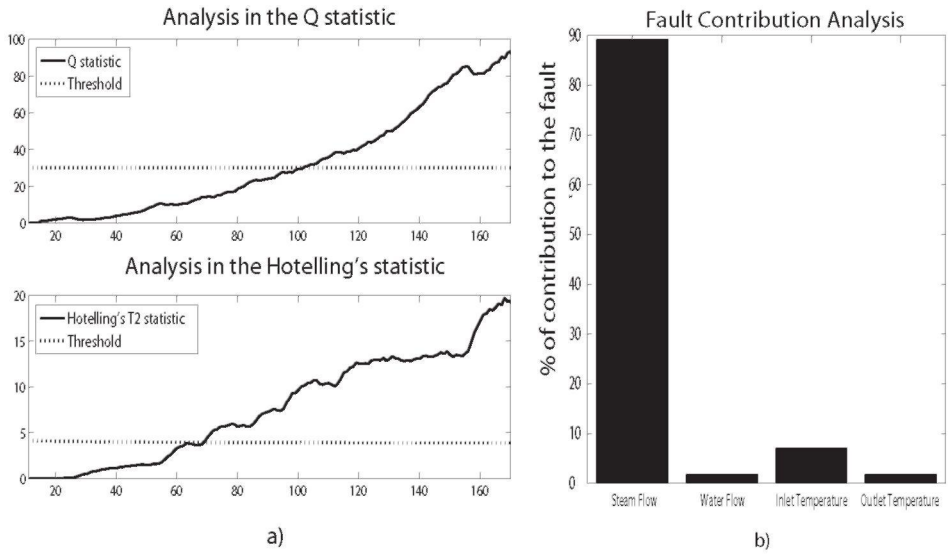

Figure 5.6: FDD analysis for a gradual fault in the steam flow sensor using the DPCA method.

Q and T^2 statistics overshoot their control limits after 93 and 59 seconds respectively when was introduced a gradual fault in the steam flow sensor at time 10 (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that close to 90% of total error corresponds to steam flow signal, thus the fault was originated in this sensor/transmitter (right).

Gradual faults in the inlet temperature sensor and water flow sensor are detected and isolated correctly. These results can be reviewed in the appendix D.4.

Detection and Diagnosis of Abrupt Faults in Actuators

For testing the robustness of DPCA method, abrupt faults in the actuators (i.e. steam and water control valves) were activated. This kind of fault is considered as a positive or negative change in the pressure of either pneumatic control valve. The faults in these actuators are implemented as small changes in the normal operating conditions.

The process is considered as normal when the steam and water control valves have 70% and 38% of opening respectively, and the sensors measurements have the behavior shown in Figure 5.1. Pressure changes in the pneumatic control valves will affect the outlet temperature signal and the respective flow signal. A change in opening percentage of the control valve is simulated as a pressure change.

In this work, a positive or negative change in the pressure is emulated when the opening percentage of the control valves increases or decreases 10% respectively. Five cases have been designed for testing the robustness of DPCA method; the case 0 is considered as the normal operation point. The cases can be reviewed in the Table 5.2.

Figure 5.7 shows the process behavior under different operating points (i.e. cases [1 - 4]).

Table 5.2: Different fault cases emulated in the actuators.

Case	Opening percentage in the steam control valve	Opening percentage in the water control valve
0	70%	38%
1	60%	38%
2	80%	38%
3	70%	28%
4	70%	48%

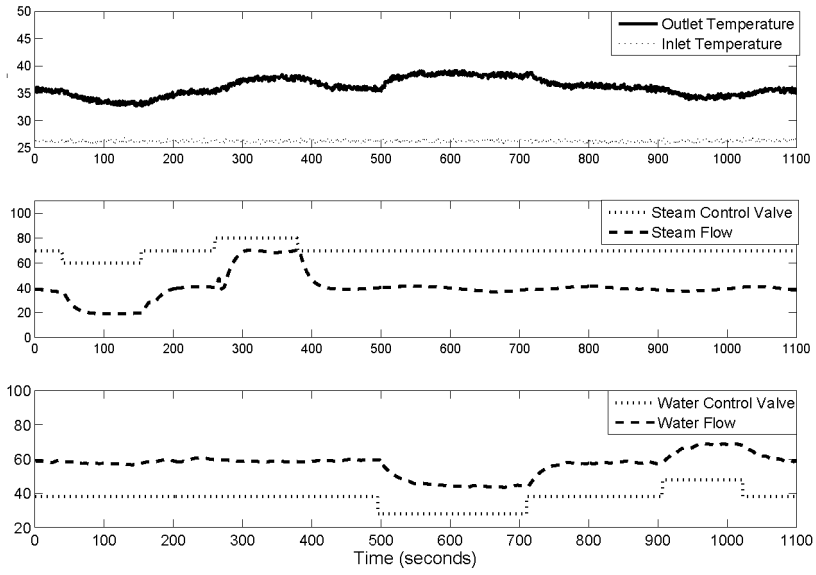

Figure 5.7: Implemented abrupt faults in actuators fortesting the DPCA method.

The case 1, 2, 3 and 4 are activated at time 40, 260, 496 and 910 respectively. Both pressure changes are emulated with an amplitude of 10 units. The used sample time is 1 second.

Applying DPCA, the FDI system shows that both statistics (Hotelling and Q) are below of their control limit (i.e. there is not fault) when the process is on the case 0. On the other hand, when it occurs any fault case [1 - 4], independently if the bias is positive or negative in the valves position, there is a reaction in both statistics. Hotelling and *Q* statistics overshoot their control limits; thus, a fault is detected. Figure 5.8 shows the behavior of both statistics under different fault cases [1 - 4].

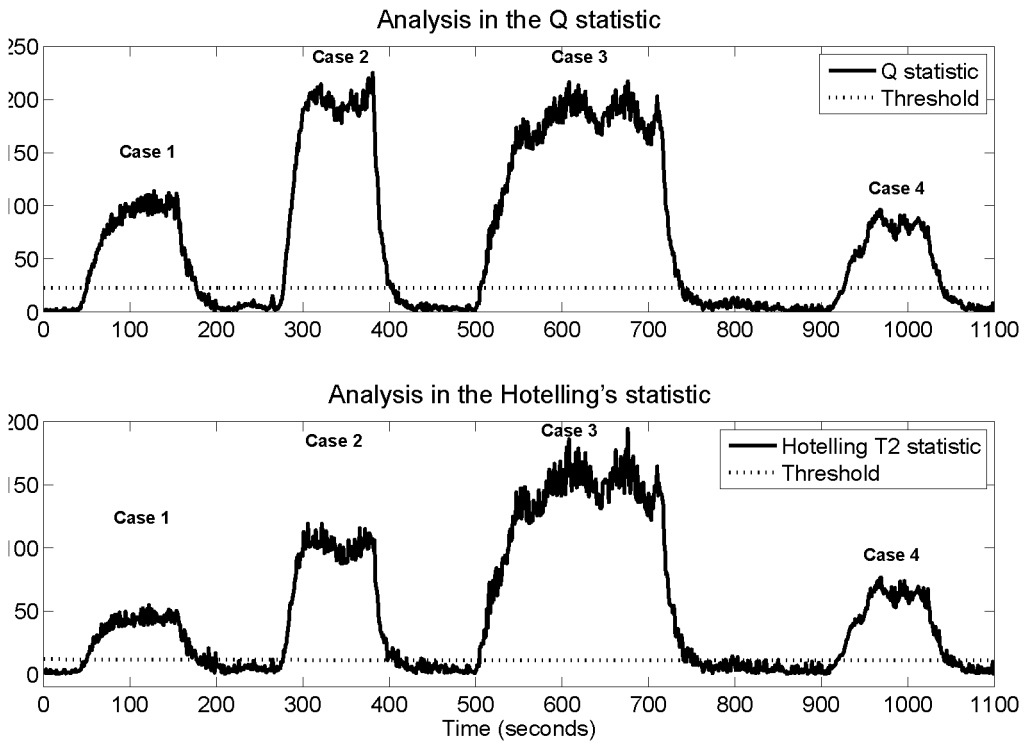

Figure 5.8: Fault detection analysis for emulated faults in actuators using the DPCA method. *Q* and T^2 statistics overshoot their control limits in all fault cases [1 - 4].

Since this kind of actuator faults represent changes in the normal operating conditions, the detection time is not instantaneous even when these faults are abrupt. Detection time is defined as the period between the fault activation time and the time when the *Q* and T^2 statistics overshoot their control limits once the fault has occurred. Table 5.3 shows the detection time which is obtained in each fault case.

Contribution plots have been used for isolating these faults. For the cases 1 and 2, the steam flow signal has the greatest error contribution followed by the outlet temperature signal. This result is considered correct because the faults are associated to changes in the pressure of steam valve (negative and positive respectively). Figure 5.9(a) and 5.9(b) show the diagnostic result for the fault cases [1-2] respectively.

Similarly, the water flow signal has the greatest contribution to the error when the water valve is affected by

Table 5.3: Detection time for faults in the actuators using the DPCA method.

Fault case	Activation time (s)	Reaction Time of Q statistic (s)	Reaction Time of T^2 statistic(s)	Detection Time (s)
1	40	51	51	11
2	260	278	278	18
3	496	505	505	9
4	910	928	921	18 using Q 11 using T^2

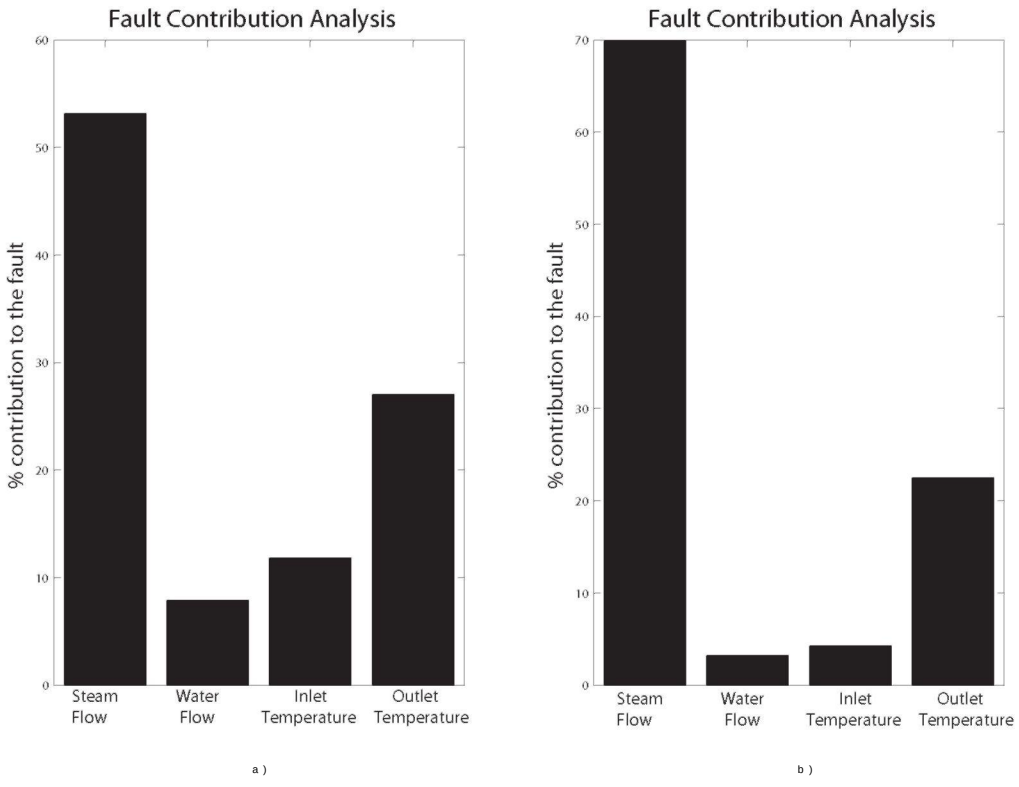

Figure 5.9: Diagnostic result for emulated faults in the actuators (cases [1 - 2]).

When are activated the cases 1 and 2, contribution plots show that close to 53% (left) and close to 70% (right) of total error corresponds to steam flow signal, respectively.

a pressure change (it does not matter if is positive or negative). The diagnostic result for the fault cases [3-4] can be reviewed in the Figure 5.10(a) and 5.10(b) respectively. It is important to note that in all fault cases [1-4], the outlet temperature signal has the second greater error contribution. This occurs because any change in the normal operating conditions will cause a deviation in this signal. It will be possible to discard the option of a fault in the outlet temperature signal, when occurs a fault in one actuator, if exist redundant measurements in the process. These latter signals must be capable of associating the change in the outlet temperature signal to a flow change.

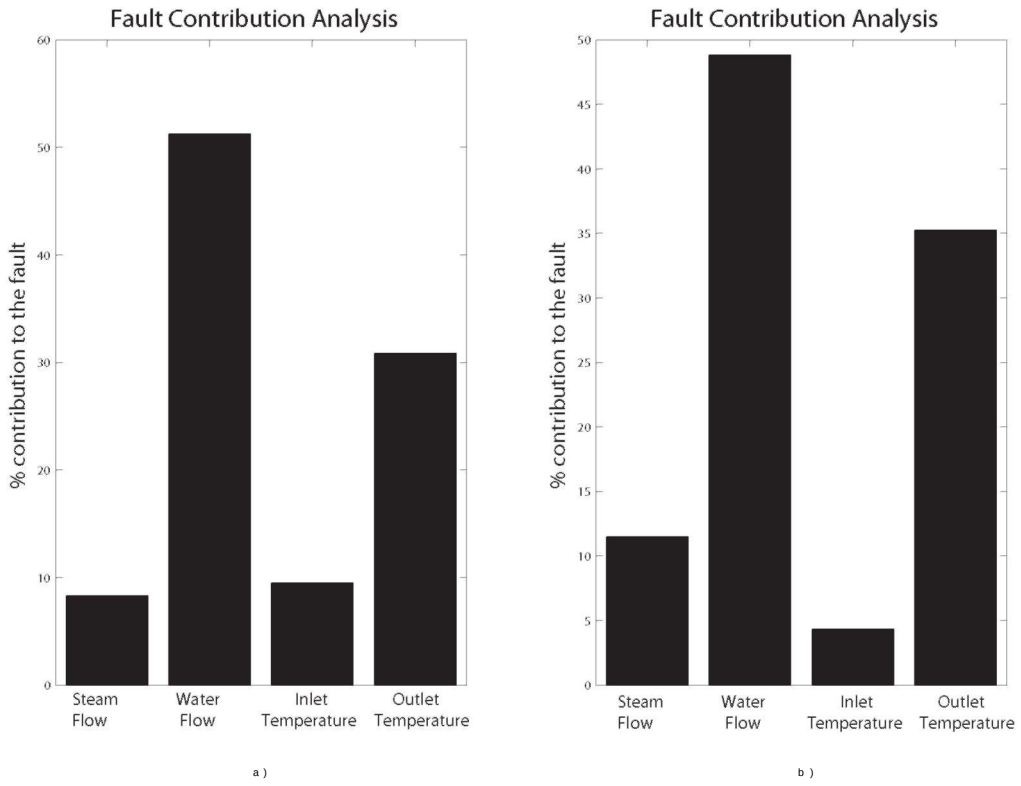

Figure 5.10: Diagnostic result for emulated faults in the actuators (cases [3 - 4]).

When are activated the cases 3 and 4, contribution plots show that close to 51% (left) and close to 49% (right) of total error corresponds to water flow signal, respectively.

Detection and Diagnosis of Multiple Faults in Sensors

Multiple faults have been activated at different time instants, i.e. once one or more faults have occurred another fault is activated in a different sensor. Table 5.4 shows the sequence in which the faults have been introduced. All these deviations are considered as abrupt faults.

Figure 5.11 shows the fault detection result using the DPCA method. Both statistics overshoot their control limits when the fault 1 has occurred at time 10. When the remainder of the faults are introduced, the statistics belong still upon their control limits, however, they move more away from their thresholds. None statistic comes back to

Table 5.4: Sequence of simulated faults in multiple form.

Fault	Failed sensor	Activation time (s)
1	Water flow sensor	10
2	Inlet temperature sensor	48
3	Steam flow sensor	147
4	Outlet temperature sensor	250

its normal status since none fault is deactivated.

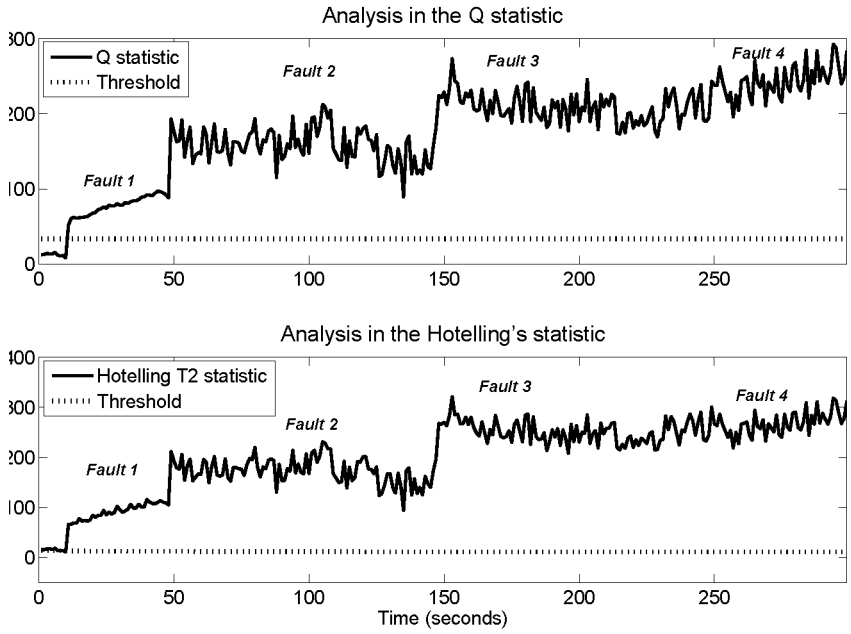

Figure 5.11: Fault detection result using the DPCA method under multiple faults.

When is activated the fault 1, Q and T^2 statistics overshoot their control limits at time 10. When are introduced abrupt faults in the remainder of sensors, both statistics move more away from their thresholds.

Diagnostic result using contribution plots can be reviewed in the Figure 5.12.

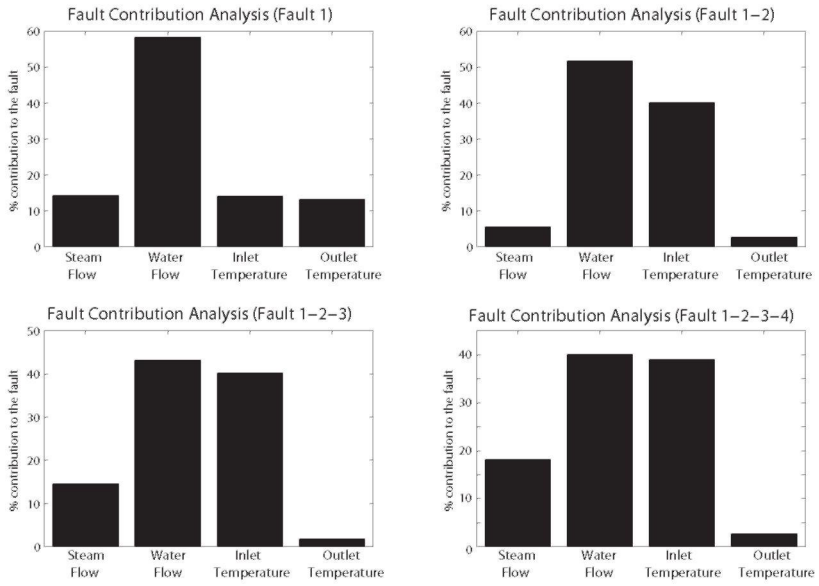

Figure 5.12: Diagnostic result for implemented multiple faults in all sensors.

When is activated the fault 1, the highest error corresponds to water flow signal. Then, when are implemented the remainder of the faults, the contribution plots can not associate the error to a specific fault.

5.2 Fault Detection and Diagnosis using Observers

5.2.1 Training Stage

An accurate state space model of the process is required for constructing a diagnostic observer. Thus, the first step for designing a diagnostic observer consists of getting a reliable state space model which must be observable. In this case, a Random Binary Signal (RBS) was used for identifying to the process at different fault conditions. The RBS test was executed firstly on the steam control valve and then on the water control valve. Figure 5.13 shows the RBS test applied on the system at normal operating conditions for identifying an Autoregressive model with exogenous inputs (ARX). Steam and water flow, inlet temperature and the control valves are considered inputs of the system. The outlet temperature is the system output.

Using recursive least squares method, four different ARX models were identified, one for each fault condition: fault in the steam flow, water flow, inlet temperature and outlet temperature. All ARX models include the same number of parameters in the inputs (5) and process output (a= 5) and the same number of delays (d=2), considering 5 seconds of sample time. Figure 5.14 presents the model reliability at normal operating conditions; the ARX model follows correctly to real process value (outlet temperature).

Knowing the reliability and accuracy of the ARX model, a discrete state space model was constructed from the parametric model. Each discrete state space model is a MISO (Multiple Inputs - Single Output) model with 5 inputs. The state matrix (G) and input matrix (H) have dimensions of [5 *x* 5], whereas the output matrix is defined as a

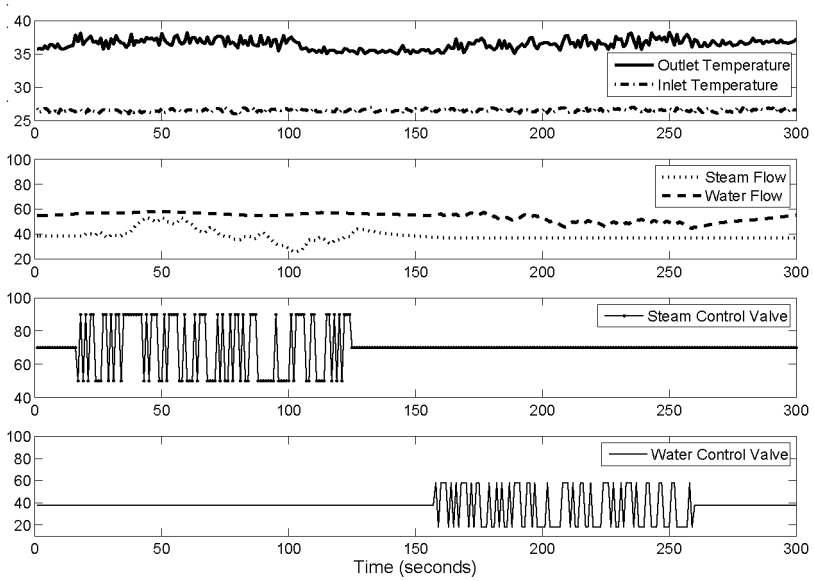

Figure 5.13: RBS test.

The identification test was applied at time 15 in the steam control valve, whereas at time 160 the RBS test was executed in the water control valve. Both RBS tests have an amplitude of 20 units.

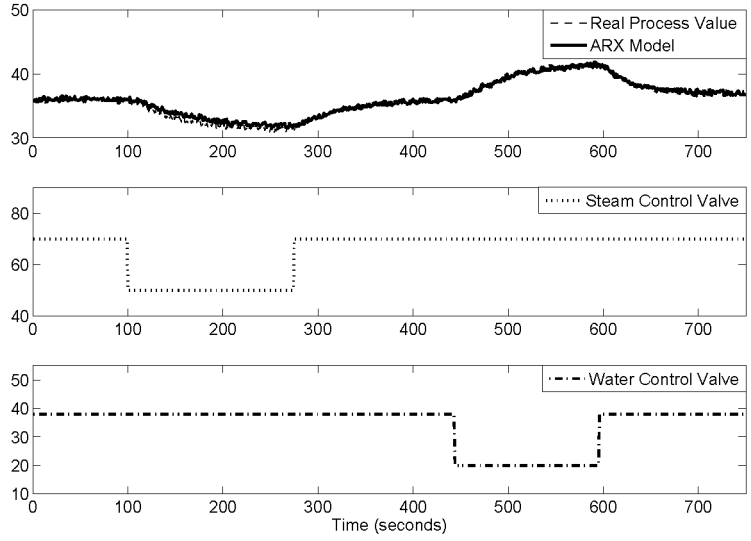

Figure 5.14: Model reliability.

One negative step into the steam control valve is applied at time 100, whereas its positive counterpart is introduced at time 280. Similarly, a negative and positive step is applied into the water control valve at time 440 and 695 respectively.

vector of length 5. With the assumption that the structure and the parameters of the discrete state space model are known, a state observer is used to reconstruct the unmeasurable state variables based on measured inputs and outputs.

Particularly, a bank of four diagnostic observers was designed for distinguishing different fault conditions. The stability of all diagnostic observers is reached by proper design of the observer feedback matrix K_e, in this case pole placement was used. The design of the observer feedback matrix using pole placement can be reviewed in detail in the appendix E. The proposed dynamics of all observers includes poles very close to the origin (lower of 0.1 in discrete analysis).

5.2.2 Testing Stage

Detection and Isolation Process

All observers are designed from different fault models and they are sensible to any fault except the used fault for their design. In this case, the fault detection and its isolation can be achieved in parallel, i.e. when a fault occurs, which is considered in one model and the remainder models does not involve it, the residue in the observer which was designed from the model of the occurred fault wont change from its nominal value since it considers the occurred fault, whereas, all other residuals will be deviated.

Therefore, for identifying multiple faults which occur in different time instants, it is necessary to analyze all residues. The diagnosis of the first fault will be associated by the residue which does not change its nominal value (the residual value continuous being close to zero). However, the detection and diagnosis of new faults will be determined by the residual which does not change its past behavior once the fault has occurred (in this case, the residue will have a different value to zero because the first fault changed it). Verde [11] presents the same residuals pattern when analyzes multiple faults in a process.

The fault detection time can be computed from the time instant when all residuals changed except the residual which is associated to the occurred fault.

Detection and Diagnosis of Abrupt Faults

Firstly, the set of diagnostic observers is tested with the same abrupt faults which were simulated in the DPCA method. The bias magnitudes of all sensor/transmitters can be reviewed in detail in the table 4.2.

When an abrupt fault is simulated in the outlet temperature sensor, the outlet temperature residue is the unique signal which does not change its nominal behavior whereas the remainder of the residues are deviated negatively 1.5 units. Thus, it is possible to assign the fault to outlet temperature sensor. Figure 5.15 allows to show how the outlet temperature residue does not change its behavior whereas the other residues are deviated negatively at time 10when an abrupt fault is activated in the outlet temperature sensor; these latter residues do not return to the normal condition. The abrupt fault has a simulated bias of 2°C and its behavior can be seen in Figure 4.5(up).

Similarly, using a set of diagnostic observers, the detection and isolation are achieved correctly when an abrupt

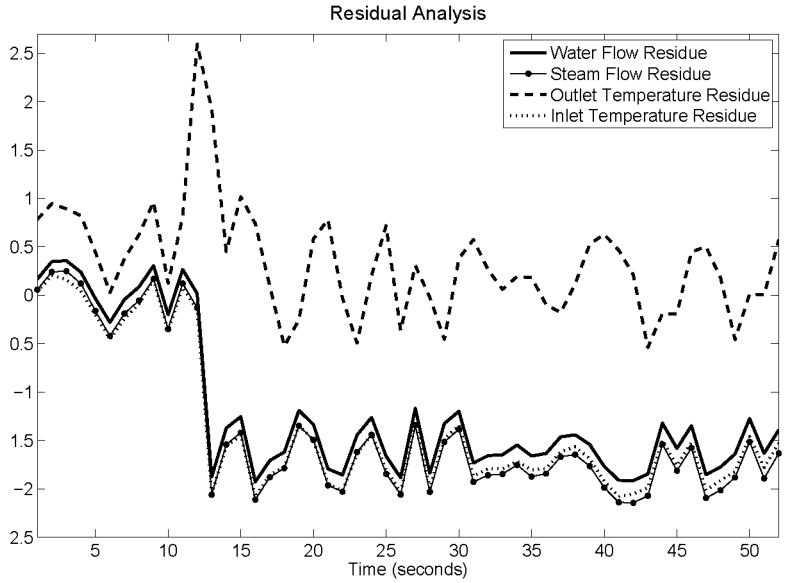

Figure 5.15: FDD analysis for an abrupt fault in the outlet temperature using diagnostic observers.

All residues except the outlet temperature residue, are deviated from their nominal behavior at time 10 when an abrupt fault is activated in the outlet temperature signal, thus the fault was originated in this sensor/transmitter.

fault (bias of 5s) is simulated in the steam flow sensor. Online monitoring shows that all residues except the steam flow residue, are deviated from their nominal behavior at time when the fault was introduced. Figure 5.16 shows that outlet and inlet temperature residue and water flow residue change their behavior at time 8 and do not return to the normal condition. Therefore, residual analysis allows to detect and isolate a possible fault in parallel.

Satisfactory results are obtained too after simulating faults in the inlet temperature sensor and water flow sensor. Diagnostic results of these simulated faults can be reviewed in the appendix F.1.

Detection and Diagnosis of Gradual Faults

The set of Diagnostic observers was tested with gradual faults in all heat exchanger sensors for validating the robustness of this approach. Each gradual fault was simulated similarly that in the DPCA algorithm. The standard behavior of the gradual faults can be seen in Figure 4.5. For each measurement was established a slope of 0.1 units per second.

In order to detect this kind of fault, called *integrator fault,* the diagnostic observer needs to consider an integrator in its design. For simplicity, in this work, the diagnostic observer does not involve an integrator since the implemented gradual-fault can be detected and isolated correctly.

A satisfactory diagnostic result is gotten when a gradual fault is simulated in the outlet temperature sensor at time 10. Due that all residues are deviated except the outlet temperature residue, it can be detected and isolated

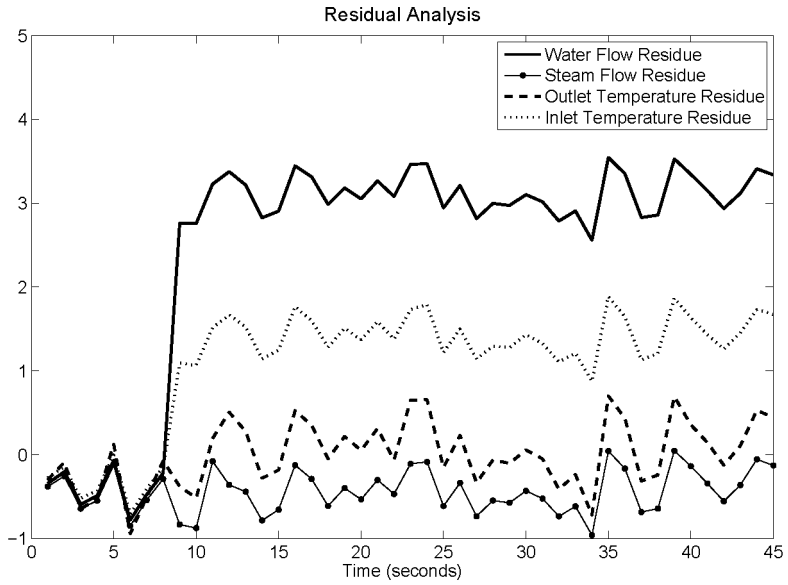

Figure 5.16: FDD analysis for an abrupt fault in the steam flow sensor using diagnostic observers.

All residues except the steam flow residue, are deviated from their nominal behavior at time 8 when an abrupt fault is activated in the steam flow signal, thus the fault was originated in this sensor/transmitter.

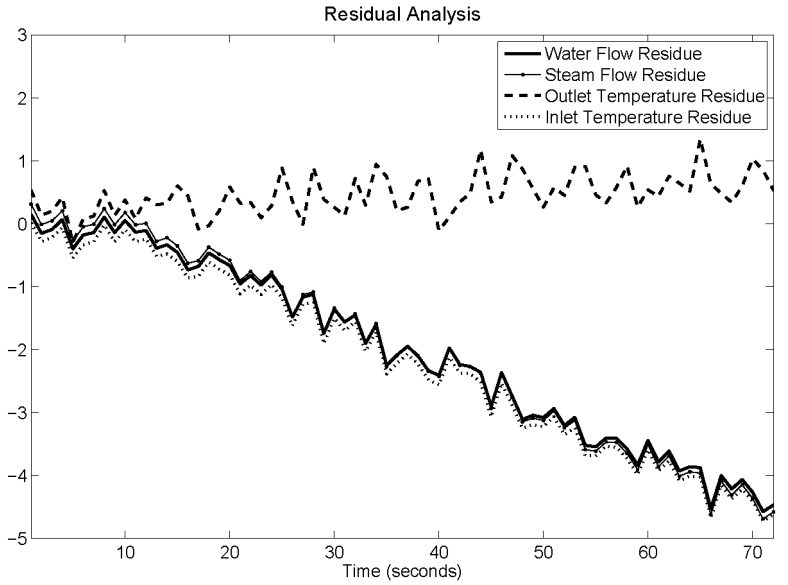

Figure 5.17: FDD analysis for a gradual fault in the outlet temperature using diagnostic observers.

All residues except the outlet temperature residue, are deviated negatively from their nominal behavior after 5 seconds when was introduced a gradual fault in the outlet temperature sensor at time 10, thus the fault was originated in this sensor/transmitter.

a fault in the outlet temperature sensor. Figure 5.17 shows the fault detection after 5 seconds once the fault has happened; this is the time when the residues are deviated. As it is evident, the behavior of all residuals corresponds to the fault behavior which was introduced. Due to established slope, when the fault achieves 0.5°C can be detected through the residual analysis.

Similarly, FDD system based on diagnostic observers detects and isolates a gradual fault simulated in the steam flow sensor. Online monitoring of residues allows to show the fault since all residues except the steam flow residue change its nominal behavior. After 10 seconds all residues except the steam flow residue, begin to be deviated once the gradual fault is simulated at time 10, see Figure 5.18. In this case too, the slope of the gradual fault is 0.1% per second and after 10 seconds, the total bias will be 1 percent; value which is close to normal oscillation rank (8%).

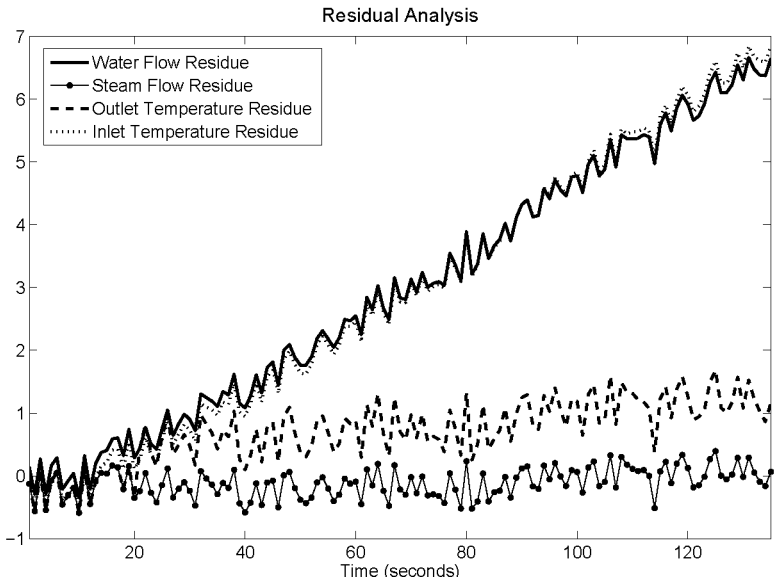

Figure 5.18: FDD analysis for a gradual fault in the steam flow sensor using diagnostic observers.

All residues except the steam flow residue, are deviated positively from their nominal behavior after 10 seconds when was introduced a gradual fault in the steam flow sensor at time 10, thus the fault was originated in this sensor/transmitter.

A satisfactory performance of the FDD system based on a set of diagnostic observers is obtained after simulating gradual faults in the inlet temperature sensor and water flow sensor. Diagnostic results of these simulated faults can be reviewed in detail in the appendix F.2.

Detection and Diagnosis of Abrupt Faults in Actuators

Abrupt faults in actuators are implemented for testing the robustness of this FDI system. Five cases have been designed as positive or negative changes in the pressure of either pneumatic control valve. The case 0 is considered as the normal operating point, the behavior of all measurements under the normal operating conditions can be seen

in the Figure 5.1. The fault cases can be reviewed in the Table 5.2. A change in opening percentage of the control valve (10%) is simulated as a pressure change.

Figure 5.19 shows the process behavior under different operating points (i.e. cases [1 - 4]).

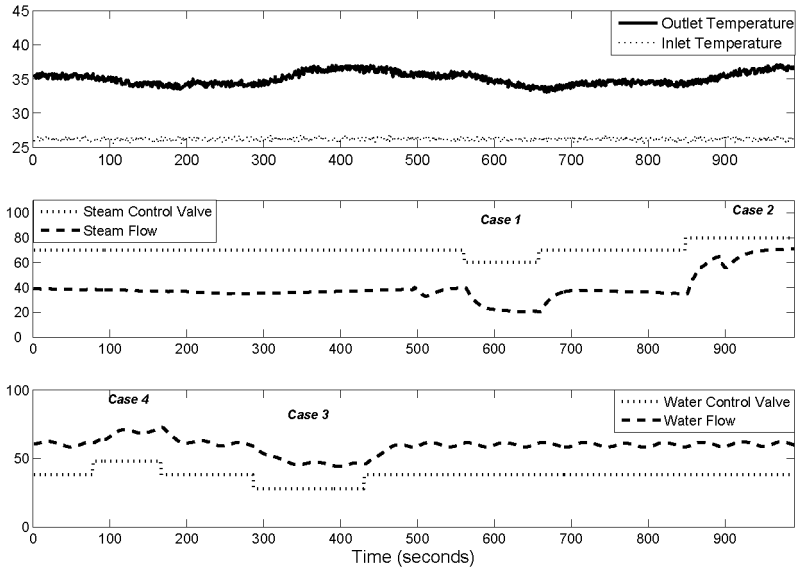

Figure 5.19: Implemented abrupt faults in actuators for testing the diagnostic observers.

The case 4, 3, 1 and 2 are activated at time 78, 286, 560 and 849 respectively. Both pressure changes are emulated with an amplitude of 10 units. The used sample time is 1 second.

When is implemented a fault in the water control valve, independently if the bias is positive or negative in the valve position, the water flow residue does not change its behavior from its nominal value; whereas, the remainder of the residues are deviated. Figure 5.20 shows the behavior of all residues under the fault cases [4 - 3]. On the other hand, when is implemented a fault in the steam control valve, the steam flow residue does not change its behavior; Figure 5.21 shows the behavior of all residues under the fault cases [1 - 2].

Table 5.5 shows the detection time which is obtained in each fault case. For this FDI system, the detection time is defined as the period between the fault activation time and the time when the residues are deviated from their nominal behavior. In all fault cases, the detection and diagnosis task occur at the same time.

Detection and Diagnosis of Multiple Faults in Sensors

Multiple faults in all sensors are implemented in the process for testing the robustness of the diagnostic observers. The sequence in which the faults have been introduced can be reviewed in the Table 5.4. Figure 5.22 shows the FDD result using a set of diagnostic observers.

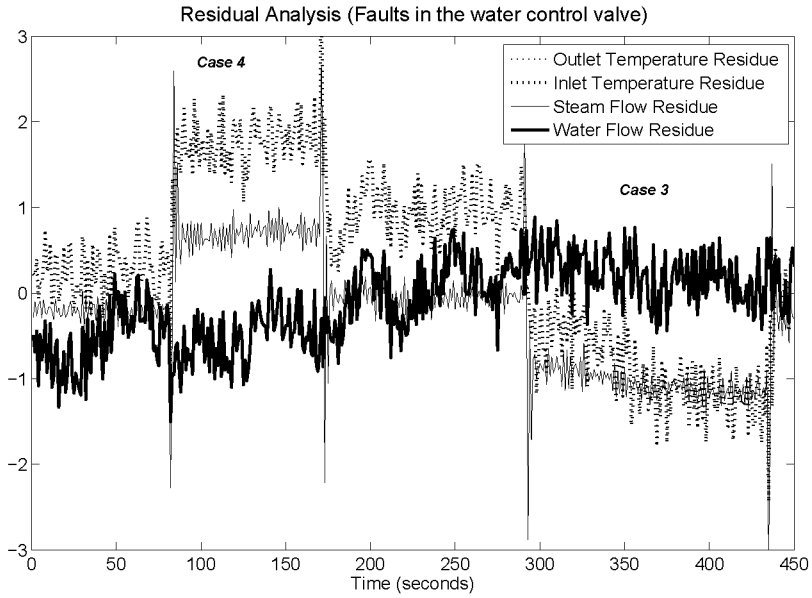

Figure 5.20: Fault detection analysis for faults in the water control valve using the diagnostic observers.

All residues except the water flow residue, are deviated from their nominal behavior when is implemented an abrupt fault in the water control valve (fault cases [4 - 3]).

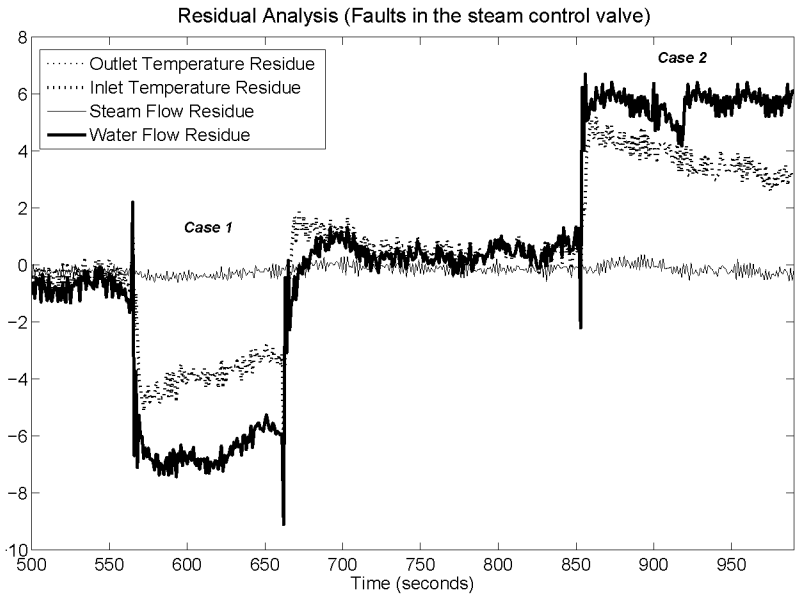

Figure 5.21: Fault detection analysis for faults in the steam control valve using the diagnostic observers.

All residues except the steam flow residue, are deviated from their nominal behavior when is implemented an abrupt fault in the steam control valve (fault cases [1 - 2]).

Table 5.5: Detection time for faults in the actuators using a set of diagnostic observers.

Fault case	**Activation time (s)**	**Reaction Time in the residuals (s)**	**Detection Time (s)**
1	560	566	6
2	849	854	5
3	286	294	8
4	78	83	5

It is important to note that only one signal is not deviated from its behavior when is introduced any abrupt sensor fault. The residual signal which does not change its behavior is associated to the occurred fault. None residual comes back to zero since none fault comes back to its nominal behavior (i.e. normal operating condition).

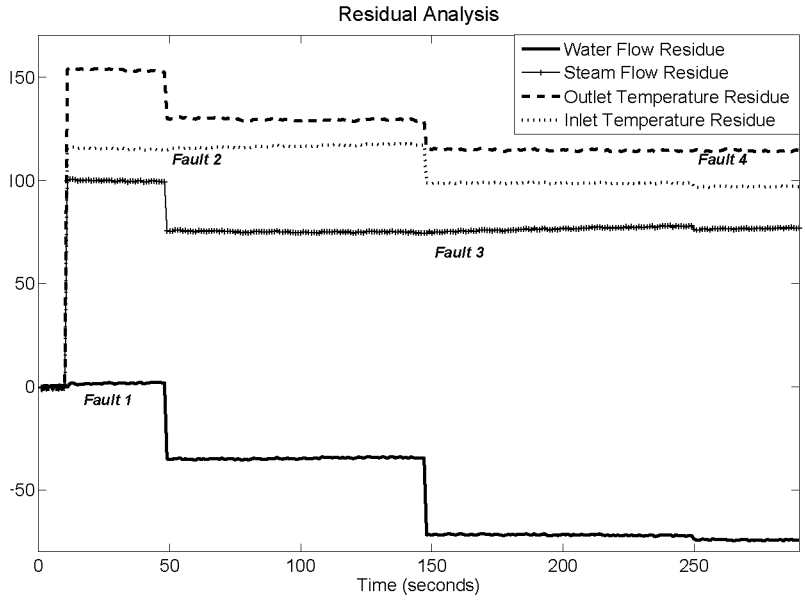

Figure 5.22: FDD result using diagnostic observers under multiple faults.

When are activated the faults 1, 2, 3 and 4 in sequence, the residuals which do not change their behavior are: water flow, inlet temperature, steam flow and outlet temperature respectively.

5.3 Comparison of the methods

Table 5.6 shows the obtained results of both FDI methods. The FDD results are based on the desirable characteristics which must have the fault diagnostic systems according to Venkatasubramanian *et al.* [3].

It is important to specify that in all cases, the implemented faults in both methods have the same conditions, i.e. the same kind of fault with the same abrupt deviation or gradual scope. Thus, the comparison between DPCA and diagnostic observers can be specified under same metrics.

Table 5.6: Comparative analysis between the DPCA method and diagnostic observers.

FDI method	Kind of fault	Location	Detection time (s)	Isolability	Explanation facility	False alarm rate (%)
DPCA	Abrupt	Any sensor	Instantaneous	✓	✓	0
	Gradual	Outlet temperature sensor	$10-14$	✓	✓	0
	Gradual	Steam flow sensor	$59-93$	✓	✓	0
	Abrupt	Actuators	$9-18$	✓	✓	14.13
	Abrupt	Multiple sensors	Instantaneous		✓	0
Diagnostic observers	Abrupt	Any sensor	Instantaneous	✓	✓	0
	Gradual	Outlet temperature sensor	5	✓	✓	0
	Gradual	Steam flow sensor	10	✓	✓	0
	Abrupt	Actuators	$5-8$	✓	✓	9.94
	Abrupt	Multiple sensors	Instantaneous	✓	✓	0

5.4 Discussion of Results

For highlighting the benefits of DPCA above its conventional counterpart (PCA), abrupt faults are implemented in the conventional method. Appendix C shows how the PCA method can detect and isolate a simulated fault in all sensors **except in the outlet temperature sensor.** The major reasons that do not allow to isolate simulated faults in the outlet temperature sensor using the PCA algorithm are:

- Physically, the outlet temperature of water depends of its inlet temperature, water feed and steam feed. Thus, the value of the outlet temperature has a direct correlation with all used measurements. Vijaysai *et al.* [58] and Luo *et al.* [60] define that the main limitation of PCA is that it assumes normality and independence of the samples. Ku *et al.* [43] propose the dynamic PCA algorithm for overcoming the shortcomings of conventional PCA.

- The number of the principal components retained can not explain correctly the data variation caused by a simulated fault in this sensor/transmitter.

In this work, the dynamic counterpart of PCA is recommended for overcoming the limitations of conventional PCA.

Technically, a fault and process disturbance are not differed in this thesis work; in all cases, the experimental tests have been implemented free of process disturbances.

On the other hand, abrupt and gradual faults are implemented in all sensors for validating both FDI systems: DPCA and diagnostic observers. Since these faults are simulated, the process do not change its normal operating conditions (i.e. only one variable is deviated when a fault has occurred). Thus, the fault isolation is easily achieved in both methods. Under these fault conditions, the obtained false alarm rate is close to zero.

However, when is implemented a gradual fault in a sensor signal, the detection time is not instantaneous. According to the Table 5.6, the diagnostic observers show a quicker detection than the DPCA method when is simulated this kind of faults.

It is important to note that the behavior of a gradual fault is the response of an integrator. Thus, the diagnostic observer must have an integrator in order to detect this kind of fault. In this work, the gradual faults are added to a signal and only the deviations about the normal operating point are analyzed as residuals. In this manner, the diagnostic observers have been designed easily without considering the integrator. The detection results are correct based on the established metrics.

In all fault cases, it is easy to explain the fault propagation using both FDI methods, i.e. the explanation facility metric is achieved.

It is important to consider that the contribution plots indicate which variables are hypothetically more associated to the fault since it is possible that more fault cases are involved. On the other hand, a set of diagnostic observers can correctly isolate a fault if all fault models and the model of the normal operating condition are known with high reliability.

For faults in actuators, the normal operating conditions change in more of two sensors and the diagnosis task can be complicated. When is implemented this kind of actuator faults, the detection time is not instantaneous even when these faults are abrupt. In this work, diagnostic observers show a quicker detection (i.e. almost the half of detection time) than the DPCA method when faults in both actuators (steam and water control valve) are implemented at different time instants. For these faults, the diagnostic observers present a lower false alarm rate than the DPCA method (see Table 5.6).

Furthermore, both FDI methods can detect and isolate faults which are implemented as positive and/or negative pressure changes in actuators. In all these faults, the outlet temperature signal changes its normal operating condition. It will be possible to discard the option of a fault in the outlet temperature signal, when occurs a fault in one actuator, if exist redundant measurements in the process. These latter signals must be capable of associating the change in the outlet temperature signal to a flow change.

On the other hand, both FDI methods can detect multiple faults which are implemented in all process sensors. However, the DPCA method can not isolate correctly when two or more faults have been implemented, i.e. the contribution plots can not help to this task since this tool only associates the highest error to the occurred fault.

It is important to note that all implemented experiments consider the same sample time, in this case, 1 second for the DPCA method and 5 seconds for the diagnostic observers. The sample time must be selected according to the FDI requirements. The DPCA method does not require a greater sample time since with one past observation and this value of sample time it is possible to detect any abnormal event. If the DPCA method can not detect the data variation when a fault has occurred, the sample time can be increased retaining the same number of past observations into the data matrix. For the design of diagnostic observers, 2 delays were introduced in the ARX models in order to calculate the output with the past inputs of 10 seconds ago. However, the sample time can not be greater than the dead time of the process (15 seconds).

According to computational requirements, the design of diagnostic observers needs greater computational resources. The training stage of this method is more complicated than the DPCA training; the diagnostic observers require firstly a reliable ARX model which must be translated to a state space model. Furthermore, each fault case must be modelled in a particular state space model. Once the fault model is known with high reliability, is designed a state observer; particularly in this work all models (fault cases and normal operating) are obtained in parallel. On the other hand, the DPCA training is quickly executed once historical data of the normal operating point are known.

In this work, the false alarm rate is the index of false events which occurs when: (1) the FDI system does not detect an occurred fault or (2) the FDI system detects a fault which did not happen. Gudi *et al.* [9], who compare the performance of two FDI methods, compute both false events separately in: false alarm rates and missed detection rates. When the FDI system can change its threshold, it is possible to generate a *Receiver Operating Characteristic (ROC)* curve which allows to find a relation between the positive detections and the false detections according to a specific threshold.

5.5 Summary

This chapter shows the obtained results by the 2 approaches based on their FDD task. The section 5.1 presents the results of detection and diagnosis of different fault cases using the DPCA algorithm; whereas, the section 5.2 presents the fault detection and diagnosis results when is used a set of observers for this task. Three fault conditions have been implemented: faults in sensors, faults in actuators and multiple faults in sensors.

Finally, a comparison among the FDI methods is presented in the section 5.3; this comparative analysis is based on the next metrics: quick detection, isolability, robustness, explanation facility and multiple fault identifiability.

Chapter 6

Conclusions

6.1 Final Conclusions

According to the training stage, in this chemical process (i.e. an industrial heat exchanger) it is more difficult to implement a FDI method based on a set of diagnostic observers since an accurate state space model of the process is required for constructing a diagnostic observer. Furthermore, each fault case must be modelled in a particular state space model. If the model has a low reliability, the diagnostic observer can not detect a fault correctly. Thus, the process model must be known with high precision. Particularly, in this work all models (fault cases and normal operating) are obtained in parallel.

On the other hand, statistical approaches like PCA and its dynamic counterpart (DPCA), which do very well on fast detection of abnormal situations, are easier to implement in applications of the chemical industry (i.e. the DPCA training is quickly executed). In this case, a process model is not required; however, a broad acquisition of the historical process measurements is necessary.

The limitations of conventional PCA are highlighted in this experimental system: the appendix C shows how the PCA method can detect and isolate a simulated fault in all sensors **except in the outlet temperature sensor.** This latter asseveration is based on the direct correlations which there are between the outlet temperature signal with the remainder of the signals (i.e. physically, the outlet temperature of water depends of its inlet temperature, water feed and steam feed).

According to experimental results, the fault detection is achieved instantaneously in both methods when are implemented abrupt and gradual faults in all sensors. Similarly, the fault isolation is easily achieved in both methods. These fault cases have been implemented for validating both FDI systems.

Abrupt faults in actuators and multiple faults in sensors have been implemented in the process at different time instants for testing the robustness of both methods: DPCA and diagnostic observers. For faults in actuators, the detection time is not instantaneous even when these faults are abrupt. Both methods can detect and isolate faults which are implemented as positive and/or negative pressure changes. In this work, diagnostic observers present a quicker detection (i.e. almost the half of detection time) than the DPCA method. Furthermore, the diagnostic

observers present a lower false alarm rate than the DPCA method (see Table 5.6).

False alarms are considered when: (1) the FDI system does not detect an occurred fault or (2) the FDI system detects a fault which did not happen. The former event is called *Positive False (FP)* whereas the latter event is called *Negative False (FN)*; when the FDI system detects a fault which really has occurred or the FDI system does not detect faults which did not happen, the events are considered true (i.e. *Positive True (PT)* and *Negative True (NT)* respectively). In this work, the false alarm rate is the index of false events (FP+FN) respect to the total events in a experiment. Gudi *et al.* [9], who compare the performance between CA and DPCA method, compute both false events separately in: false alarm rates (FP) and missed detection rates (FN). When the FDI system can change its threshold, it is possible to generate a *Receiver Operating Characteristic (ROC)* curve which allows to find a relation between the positive detections (PT) and the false detections (FP) according to a specific threshold. In this work, the thresholds has not been changed.

On the other hand, the DPCA method can not identify multiple faults whereas the diagnostic observers have the multiple-fault identifiability property. Using the DPCA method, it is not possible to isolate correctly when two or more faults have been implemented since the contribution plots only associates the highest error to the occurred fault. Particularly, diagnostic observers can isolate multiple faults through the design of a bank of observers, however each observer require a model with the possible greatest reliability.

In all fault cases, it is easy to explain the fault propagation using both FDI methods, i.e. the explanation facility metric is achieved.

According to the computational requirements, the training and testing stages of the FDI system, based on a set of diagnostic observers, require greater computational resources than the stages of the DPCA method, using the LabView programming language. The training stage of diagnostic observers requires a RBS test in order to identify an ARX model which must be translated to a state space model. Multiple models must be generated, one for each fault case. The DPCA training only needs the knowledge of the normal operating conditions; any deviation into a process variable is highlighted as a fault. However, the correct design of a set of diagnostic observers does not require an additional tool for the fault isolation, whereas, the DPCA method needs to use the contribution plots for helping to this task.

The selection of the sample time must satisfy the requirements of the FDI system. In this case, the used sample time is 1 second for training and testing the DPCA method. This method does not require a greater sample time since with one past observation and this value of sample time it is possible to detect any abnormal event. If the DPCA method can not detect the data variation when a fault has occurred, the sample time can be increased retaining the same number of past observations into the data matrix. On the other hand, 5 seconds of sample time are used in order to design an test a set of diagnostic observers. Since the dead time of the process is approximately 15 seconds, the used sample time must be lower than 7.5 seconds (considering 2 delays in the ARX model).

Although the DPCA method and diagnostic observers have a correct performance, the design of the FDI hybrid systems based on a supervisory control is the best option in the industrial implementation (e.g. a supervisory system based on expert systems, fuzzy models, etc.).

6.2 Future Investigation Works

Continuing with this investigation line, it is possible to analyze different schemes based on their FDD task:

- To analyze the effect of the sample time on the diagnostic results of both methods in order to obtain its optimal value.
- To test different numbers of delays in the DPCA method in order to obtain the best diagnostic results.
- To test the DPCA method and the diagnostic observers in the closed loop system.
- To take corrective actions without causing significant degradation in the closed loop performance due to the occurred faults, this kind of control is called *Fault-Tolerant Control (FTC).*
- To implement a complementary fault detection system in order to estimate the fault magnitude (for example Generalized Likelihood Ratio (GLR), Artificial Neural Network (ANN), etc.).
- To investigate how to consider process disturbances in the FDI training, as well as, how to distinguish between a possible perturbation to a possible fault.

Bibliography

[1] S. L. Jämsä-Jounela. Future Trends in Process Automation . *Annual Reviews in Control*, 31:211–220, 2007.

[2] R. Isermann and P. Ballé. Trends in the Application of Model-Based Fault Detection and Diagnosis in Technical Processes . *Control Engineering Practice*, 5(5):638–652, 1997.

[3] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin. A Review of Process Fault Detection and Diagnosis Part I Quantitative Model-Based Methods . *Computers and Chemical Eng.*, 27:293–311, 2003.

[4] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin. A Review of Process Fault Detection and Diagnosis Part II Qualitative Model-Based Methods . *Computers and Chemical Eng.*, 27:313–326, 2003.

[5] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin. A Review of Process Fault Detection and Diagnosis Part III Process History Based Methods . *Computers and Chemical Eng.*, 27:327–346, 2003.

[6] N. Kazantzis and C. Kravaris. Nonlinear Observer Design for Process Monitoring . *Ind. Eng. Chem. Res.*, (39):408–419, 2000.

[7] S. Rajaraman, J. Hahn, and M. S. Mannan. A Methodology for Fault Detection, Isolation, and Identification for Nonlinear Processes with Parametric Uncertainties . *Ind. Eng. Chem. Res*, 43:6774–6786, 2004.

[8] S. Simmani and R. J. Patton. Fault Diagnosis of an Industrial Gas Turbine prototype Using a System Identification Approach . *Control Engineering Practice*, (16):769–786, 2008.

[9] K. P. Detroja, R. D. Gudi, and S. C. Patwardhan. Plant-wide Detection and Diagnosis using Correspondece Analysis. *Publicated in Control Engineering Practice*, 2005.

[10] J. Mina and C. Verde. Fault Detection for MIMO Systems Integrating Multivariate Statistical Analysis and Identification Methods . In *American Control Conference*, pages 3234–3239, July 2007.

[11] C. Verde. Multi-leak Detection and Isolation in Fluid Pipelines . *Control Eng. Practice*, 9:673–682, 2001.

[12] F. Caccavale and L. Villani. An Adaptive Observer for Fault Diagnosis Nonlinear Discrete-Time Systems . In *American Control Conference*, pages 2463–2468, July 2004.

[13] M. A. Demetriou and A. Armaou. Robust Detection and Accommodation of Incipient Component Faults in Nonlinear Distributed Processes . In *American Control Conference*, pages 2054–2059, July 2007.

[14] Y. Xiong and M. Saif. Robust and Nonlinear Fault Diagnosis Using Sliding Mode Observers . In *IEEE Conf. on Decision and Control*, pages 567–572, December 2001.

[15] D. X. Tien, K.W.Lee, and L. Jun. Comparative Study of PCA Approaches in Process Monitoring and Fault Detection . In *30th Annual Conference of the IEEE Industrial Electronics Society*, pages 2594–2599, November 2004.

[16] H. Zhang, A.K.Tangirala, and S. L. Shah. Dynamic Process Monitoring using Multiscale PCA . In *IEEE Conference on Electrical and Computer Engineering*, pages 1579–1584, May 1999.

[17] N. Lu, F. Wang, and F. Gao. Combination Method of Principal Component and Wavelet Analysis for Multivariate Process Monitoring and Fault Diagnosis . *Ind. Eng. Chem. Res.*, (42):4198–4207, 2003.

[18] Y. Chuanqiang, G. Xiaosong, and Zhang An. An Improvement Algorithm of Principal Component Analysis . In *IEEE The Eight International Conference on Electronic Measurement and Instruments*, pages 529–534, 2007.

[19] S. J. Bailey. From Desktop to Plant Floor, a CRT is the Control Operators Window on the Process . *Control Engineering*, 31(6):86–90, 1984.

[20] S. C. Patwardhan, J. Prakash, and S. Narasimhan. A Supervisory Approach to Fault-Tolerant Control of Linear Multivariable Systems . *Ind. Eng. Chem. Res.*, 41:2270–2281, 2002.

[21] J. Mina and C. Verde. Fault Detection Using Dynamic Principal Component Analysis by Average Estimation . In *2nd Int. Conf. on Electrical and Electronis Engineering*, pages 374–377, September 2005.

[22] J. Gertler. *Fault Detection and Diagnosis in Engineering Systems* . CRS Press, 1st edition, 1998.

[23] J. Gertler and D. Singer. A New Structural Framework for Parity Equation-based Failure Detections and Isolation. *Automatica*, 26:381–388, 1990.

[24] C. W. Chan, S. Hua, and Z. Hong-Yue. Application of Fully Decoupled Parity Equation in Fault Detection and Identification of DC Motors . *IEEE Transactions on Industrial Electronics*, 53(4):1277–1284, 2006.

[25] I. Fagarasan and S. Illiescu. Parity Equations for Fault Detection and Isolation . In *IEEE Int. Conf. on Automation, Quality and Testing, Robotics*, pages 99–103, May 2008.

[26] R. E. Kalman. A New Approach to Linear Filterig and Prediction Problems . *Transaction of the ASME - Journal of Basic Engineering*, pages 35–45, 1960.

[27] S. J. Julier and J. K. Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems . *Proc. AeroSense: 11th Int. Symp. Aerospace/Defense Sensing, Simulation and Controls*, pages 182–193, 1997.

[28] S. C. Patwardhan, J. Prakash, and S. Narasimhan. Integrating Model Based Fault Diagnosis with Model Predictive Control . *Ind. Eng. Chem. Res.*, 44:4344–4360, 2005.

[29] S. C. Patwardhan, S. Manuja, S. Narasimhan, and S. L. Shah. From Data to Diagnosis and Control Using Generalized Orthonormal Basis Filters. Part II: Model Predictive and Fault Tolerant Control . *Journal of Process Control*, 16:157–175, 2006.

[30] Y. Zhang, J. Sheng, S. J. Qin, and T. Hesketh. Dynamic Process Monitoring using Multiscale PCA . In *IEEE Conference on Electrical and Computer Engineering*, pages 1579–1584, May 1999.

[31] R. Li and X. Wang. Qualitative/Quantitative Simulation of Process Temporal Behavior using Clustered Fuzzy Diagraphs . *American Institute of Chemical Engineers Journal*, 4(47):906–919, 2001.

[32] S. Amari, G. Dill, and E. Howald. A New Approach to Solve Dynamic Fault Trees . In *Proceedings Annual Reliability and Maintainability Symposium*, pages 374–379, January 2003.

[33] E. J. Zampino. Application of Fault-Tree Analysis to Troubleshooting the NASA GRC Icing Research Tunnel . In *Proceedings Annual Reliability and Maintainability Symposium*, pages 16–22, January 2001.

[34] S. Zhou and S.P. Ting. Qualitative Physics for Movable Objects in MOUT . In *Proceedings of the 39th Annual Simulation Symposium*, April 2006.

[35] J. Petersen. Model-Based Integration and Interpretation of Data . In *IEEE International Conference on Systems, Man and Cybernetics*, pages 815–820, 2004.

[36] M. Wo, W. Gui, D. Shen, and Y. Wang. Expert Fault Diagnosis Using Role Models with Certainty Factors for the Leaching Process . In *Third World Congress on Intelligent Control and Automation*, pages 238–241, June 2000.

[37] D. Leung and J. Romagnoli. Dynamic Probabilistic Model-Based Expert System for Fault Diagnosis . *Computers and Chemical Engineering*, 11(24):2473–2492, 2000.

[38] N. J. Scenna. Some Aspects of Fault Diagnosis in Batch Processes . *Reliability Engineering and System Safety*, 1(70):95–110, 2000.

[39] R. Rengaswamy, T. Hagglund, and V. Venkatasubramanian. A Qualitative Shape Analysis Formalism for Monitoring Control Loop Performance . *Engineering Applications of Artificial Intelligence*, 1(14):23–33, 2001.

[40] Y. Yang, N. Lu, L. Ma F. W. Wang, and Y. Chang. Statistical Process Monitoring using Multiple PCA Models . In *Proceedings of the American Control Conference*, pages 5072–5073, May 2002.

[41] D. Seborg. A Perspective on Advanced Strategies for Process Control . *Advances in Control. Highlights of ECC'99*, pages 104–134, 1999.

[42] H. Wang, Z. Song, and P. Li. Improved PCA with Optimized Sensor Locations for Process Monitoring and Fault Diagnosis . In *IEEE Conference on Decision and Control*, pages 4353–4358, December 2000.

[43] W. Ku, R. Storer, and C. Georgakis. Disturbance Detection and Isolation by Dinamic Principal Components Analysis . *Chemometrics and Intelligent Laboratory Systems*, 30:179–196, 1995.

[44] K. P. Detroja, R. D. Gudi, S. C. Patwardhan, and K. Roy. Fault Detection and Isolation using Correspondence Analysis. *Ind. Eng. Chem. Res.*, (45):223–235, 2006.

[45] W. Chen and M. Saif. Observer-based Strategies for Actuator Fault Detection, Isolation and Estimation for Certain Class of Uncertain Nonlinear Systems . *IET Control Theory Appl.*, 1(6):1672–1680, 2007.

[46] P. Ballé, M. Fischer, D. Füssel, and R. Isermann. Integrated Control, Diagnosis and Reconfiguration of a Heat Exchanger . In *American Control Conference*, pages 922–926, June 1997.

[47] R. Krishnan and N. Pappa. Real Time Fault Diagnosis for a Heat Exchanger A Model Based Approach . In *IEEE Indicon Conference*, pages 78–82, December 2005.

[48] M. Thomson, P.M. Twigg, B. A. Majeed, and N. Ruck. Statistical Process Control Based Fault Detection of CHP Units . *Control Engineering Practice*, 8:13 –20, 2000.

[49] Y. Peng, A. Youssouf, P. Arte, and M. Kinnaert. A Complete Procedure for Residual Generation and Evaluation with Application to a Heat Exchanger . *IEEE Transactions on Control Systems Technology*, 5(6):542–555, 1997.

[50] A. Aïtouche, D. Maquin, and F. Busson. Multiple Sensor Fault Detection in Heat Exchanger Systems . In *Proceedings of the Int. Conf. on Control Application*, pages 741–745, September 1998.

[51] R. Morales-Menendez, N. De Freitas, and D. Poole. State Estimation and Control of Industrial Processes using Particles Filters . In *American Control Conference*, pages 579–584, June 2003.

[52] E. Weyer, G. Szederkényi, and K. Hangos. Grey Box Fault Detection of Heat Exchangers . *Control Eng. Practice*, 8:121–131, 2000.

[53] J. T. Hsiung and D. M. Himmelblau. Detection of Leaks in a Liquid-Liquid Heat Exchanger using Passive Acoustic Noise. *Computers Chemical Engineering*, 20(9):1101 – 1111, 1995.

[54] R. Isermann. *Fault-Diagnosis Systems* . Springer, Germany, 1^{st} edition, 2006.

[55] H. Hotelling. Analysis of a Complex of Statistical Variables into Principal Components. *J. Educ. Psychol.*, 24, 1993.

[56] J. E. Jackson and G. S. Mudholkar. Control Procedures for Residuals Associated with Principal Component Analysis . *Technometrics*, 21:341–349, 1979.

[57] P. Miller, R. Swanson, and C. Heckler. Contribution Plots: A Missing Link in Multivariate Quality Control. . *Appl. Math. and Comp. Sci.*, 4(8):775–792, 1998.

[58] P. Vijaysai, R. D. Gudi, and S. Lakshminarayanan. Errors-in-Variables-Based Modeling Using Augmented Principal Components . *Ind. Eng. Chem. Res*, 2(44):368–380, 2005.

[59] K. Ogata. *Sistemas de Control en Tiempo Discreto* . Prentice Hall, Mexico, 2^{nd} edition, 2008.

[60] R. Luo, M. Misra, and D. M. Himmelblau. Sensor Fault Detection via Multiscale Analysis and Dynamic PCA. *Ind. Eng. Chem. Res.*, (38):1489–1495, 1999.

[61] C. S. Ramírez. Diseo e Implementación de un Controlador Difuso con Ganancia Auto-justable para un Intercambiador de Calor . Master's thesis, Tecnológico de Monterrey, Campus Monterrey, 2007.

Appendix A

Basic Models of Faults

The faults which interest in engineering processes belong to one of the following categories [22]:

- *Additive process faults.* These faults normally are zero, when occur they cause a change in the plant outputs, independently of the known inputs.
- *Multiplicative process faults.* These faults are changes (abrupt or gradual) in some plant parameters, these changes depend on the magnitude of the known inputs.
- *Sensor faults.* These are discrepancies between the measured and actual values of individual plant variables. In the majority of the cases the sensor faults are additive.
- *Actuator faults.* These correspond to the discrepancies between the input command of an actuator and its actual control output. In the majority of the cases these faults are additive too.

According to Figure A.1(a) the variable Y can be influenced by an addition of the fault f or by the product of another variable X with f, see Figure A.1(b).

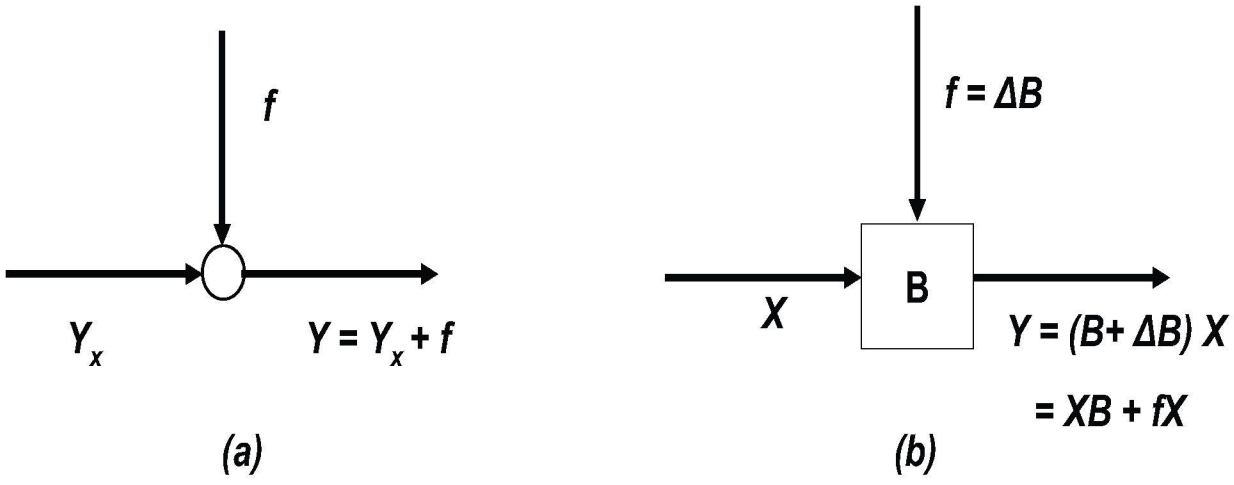

Figure A.1: Basic models of faults: (a)additive faults; (b)multiplicative faults

Appendix B

Experimental System

B.1 Description of Heat Exchanger

The shell and tube industrial heat exchanger MOD BEU-660, which is used for testing the experiments in this thesis, has a horizontal configuration with 11 tubes of Cu-Ni alloy (90-10). The tubes have $\frac{3}{4}$ inches of outside diameter BWG-20 in triangular arrangement. The shell is made of carbon steel with nominal measures of 6 inches of diameter and 70 inches of length. The design and configuration of this heat exchanger was calculated for heating 10 gpm of water from 25C to 70C with steam of low pressure ($5\frac{Kg}{cm^2}$).

B.2 Heat Exchanger Instrumentation

Figure B.1 shows the process instrumentation of the heat exchanger. Two solenoid valves allow to feed water and steam to the system. According to the arrows directions, after of the solenoid valves are located the flow sensor/transmitters, one of them for measuring the inlet steam and the another one for measuring the inlet water. In the same process current are located two pneumatic control valves which regulate the opening percentage and thus control the flows of both fluids. Furthermore, the experimental system contains two temperature sensor/transmitters, one of them quantifies the water temperature at the inlet and the another one at the outlet.

Next each instrument is described in detail:

1. *Sensor/transmitter of water flow (FT_1)*: This sensor consists of a orifice plate which measures small flows in tubes of $\frac{1}{2}$ inches of diameter; has a transmitter of pressure drop ST3000; requires 42 VCD; analog output 4-20mA.

2. *Sensor/transmitter of steam flow (FT_2)*: This sensor consists of a orifice plate which measures small flows; has a transmitter of pressure drop maximum of 3626psi; requires $10.5 - 55$ VCD; analog output 4-20mA.

3. *Pneumatic control valve of water (FV_1)*: This valve has an actuator type V1400UE+P25; electro-position E69P-B110-RS; and a pressure regulator B0123HE.

4. *Pneumatic control valve of steam (FV_2)*: This valve has an actuator type V4A+P25; electro-position E69P-BI1Q-RS; and a pressure regulator.

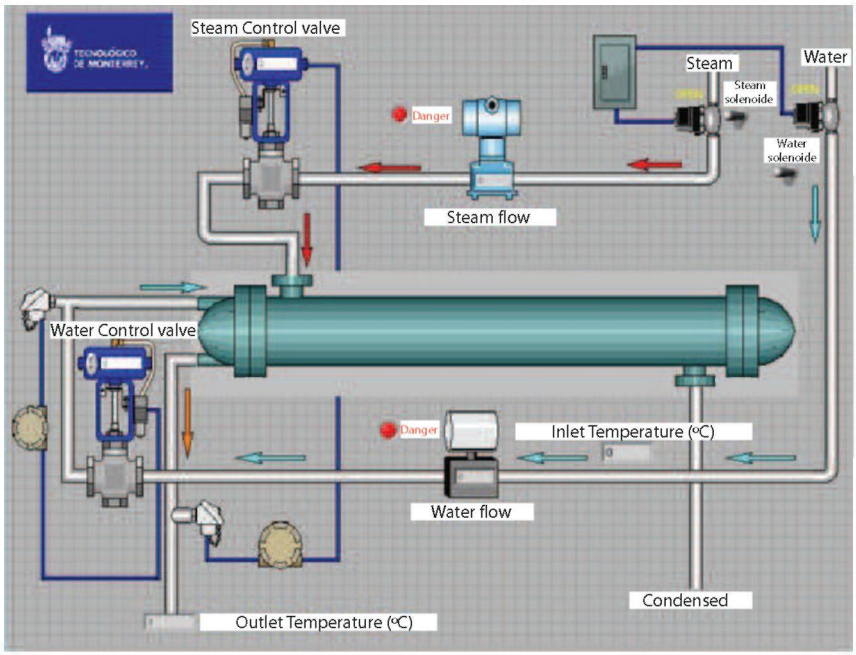

Figure B.1: Detailed process instrumentation.

5. *Solenoid valve of water (FSV_1)*: Valve for tubes of $\frac{1}{2}$ inches of diameter; designed for water (72218BN4UV00); requires 120 VCA, 60 Hz.

6. *Solenoid valve of steam (FSV_2)*: Valve for tubes of $\frac{1}{2}$ inches of diameter; designed for water (7221GBN4VESO); requires 120 VCA, 60 Hz.

7. *Temperature sensor/transmitter of the inlet water (TT_1)*: Consists of a RTD element (resistance temperature detection).

8. *Temperature sensor/transmitter of the outlet water (TT_2)*: Consists of a RTD element (resistance temperature detection).

B.2.1 Additional Instrumentation

1. *System of steam trap.*

   - Ball float steam trap FT-14-10Dn15, trademark: Spirax Sarco.
   - Filter ITDn15, trademark: Spirax Sarco.
   - Sensorial camera Spirta-Tec ST171 of $\frac{1}{2}$ inches, trademark: Spirax Sarco.
   - Check valve of bronze for tubes of $\frac{1}{2}$ inches of diameter, trademark: Spirax Sarco.
   - Thermostatic steam trap TA-125 of $\frac{1}{2}$ inches, trademark: Spirax Sarco.

2. *System of humidity separation and vacuum breaker for steam line*

- Humidity separator SIDn15, trademark Spirax Sarco.
- Thermodynamic trap TD-52EN of $\frac{3}{8}$ inches, trademark: Spirax Sarco.
- Filter AT of $\frac{3}{8}$ inches, trademark: Spirax Sarco.
- Vacuum breaker VB-14 of $\frac{1}{2}$ inches, trademark: Spirax Sarco.

3. *National Instruments equipment for the communication* FieldPoint FP-1000 network module connected to an industrial network by RS-232 to I/O Fieldpoint modules. This device uses the optomux protocol.

B.3 Data Acquisition System

B.3.1 Device Specifications

The National Instruments USB-6215 is a bus-powered isolated USB M Series multifunction data acquisition (DAQ) module optimized for superior accuracy at fast sampling rates. It offers 16 analog inputs; 250 kS/s single-channel sampling rate; two analog outputs; four digital input lines; four digital output lines; four programmable input ranges (0.2 to 10 V) per channel; digital triggering; and two counter/timers. USB-6215 also feature isolation for improved measurement accuracy and safety.

This device does not require external power and gives a bidirectional high-speed streaming of data across USB. USB-6215 is compatible with LabVIEW®, LabWindows®/CVI, and Measurement Studio for Visual Studio.

B.3.2 Physical Connections

National Instruments (NI) USB-6215 device is a data acquisition system designed for measurement and industrial control. NI USB-6215 includes networking capabilities for connecting process inputs and outputs in real-time. The input modules receive the transmitters measurements and the output modules activate the process actuators. In this particular case, the heat exchanger instrumentation is connected to a USB-6215 device through the module of analog inputs, module of analog outputs and module of digital I/O lines. The used signals in these modules are described in the table (B.1-B.2-B.3) respectively.

Analog Input Module

NI USB-6215 offers an analog input module whit 16 channels. The input range per channel is 1.6 to 6 V. Due to analog signals received from transmitters are current signals in a range of milliamperes (4-20 mA); it is necessary to use a electric circuit for translating current to voltage. The used signals in this input module are described in the Table B.1.

Analog Output Module

NI USB-6215 has an analog output module of two channels which activate voltage loops from 2 to 10 V. The analog signals must be converted to current signal for activating the pneumatic control valves (4-20 mA). The used signals in this analog output module are described in the Table B.2.

Table B.1: Inputs allocation in the module of analog inputs

Input Channel	Input Variable	Instrument	Units	Range
1	Steam flow	FT_2	% of Flow	$0 - 100$
2	Water flow	FT_1	% of Flow	$0 - 100$
3	Temperature of the inlet water	TT_1	$^{\circ}C$	$0 - 100$
4	Temperature of the outlet water	TT_2	$^{\circ}C$	$0 - 100$

Table B.2: Outputs allocation in the module of analog outputs

Output Channel	Output Variable	Instrument	Units	Range
0	Pneumatic valve control in the feed steam	FV_2	% opening	$0 - 100$
1	Pneumatic valve control in the feed water	FV_1	% opening	$0 - 100$

Digital Output Lines

NI USB-6215 offers four digital output lines. The zero logic emits 0 V whereas the one logic sends 5 V. Two digital output lines were used for activating the solenoid valves (see Table B.3).

Table B.3: Outputs allocation in the module of digital I/O lines

Line	Output Variable	Instrument	Units	Range
0	Solenoid valve in the inlet steam	FSV_2	% opening	ON-OFF
1	Solenoid valve in the inlet water	FSV_1	% opening	ON-OFF

Solenoid valves require a relay circuit for their activation. In this latter circuit, the solenoid of the valve is energized with the device output, then the circuit closes a contact which it will allow to activate the valve. In this manner, the USB-6215 output activates the solenoid valve through a relay circuit. Since the solenoid valve is ON/OFF, when the device output is in 5 volts the relay coil is energized and the solenoid valve is activated whereas when the digital output is in his lower limit the relay coil turns off and the solenoid valve is deactivated.

B.4 Manipulating of the Signals

B.4.1 Sensors Calibration

The used temperature sensors for monitoring the water temperature in the process (inlet-outlet) are RTDs sensors. The RTDs (Resistance Temperature Detectors) are temperature sensors that contain a resistor which changes its resistance value depending on a temperature change. They have been used for many years to measure temperature in laboratory and industrial processes due to its accuracy, stability and linearity. For converting the current signals

to temperature units, a calibration was done [61].

The straight line which represents the relation between the current signal and the water temperature is described by the equation B.1. The b is the current value in milliamperes and a represents the temperature measurement. The sensors was calibrated into a rank from $0°$C to $100°$C. The figure B.2 shows the linear relation between the current signal and the water temperature.

$$a = -31.77 + 6571b \tag{B.1}$$

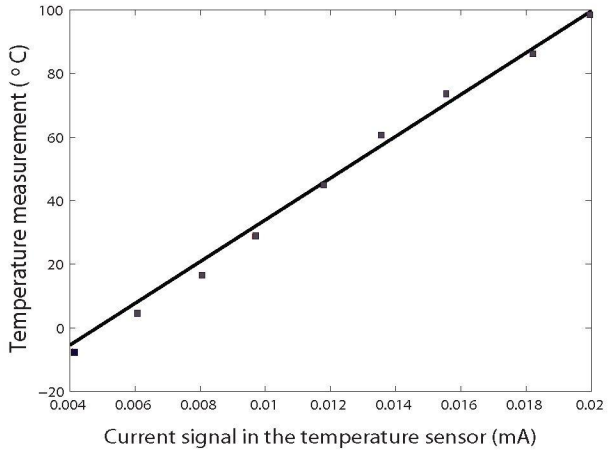

Figure B.2: Temperature sensors calibration.
Straight line shows the direct relation between the mA signal and the temperature measurement.

B.4.2 Actuators Linearization

Figure B.3 shows the normal performance of the pneumatic valve that regulates the steam inlet at the heat exchanger. When the valve operates below 40% the steam flow is zero, on the other hand when the pneumatic valve operates over 45% the flow transmitter begins to read the steam flow percentage in the process inlet. The linearization of this control valve can be observed in Figure B.4.

The opening percentage of the steam control valve was regulated between 0% and 100%, these values are showed to user through the Labview interface. Figure B.4 also shows one straight line between nearby values, in other words, a mobile average trend was calculated. In this manner, the intermediate values in the valve opening percentage are in function of each straight line calculated.

Similarly, the water control valve was linearized. Figure B.5 shows the normal operation of the control valve (a) as well as its calibration (b).

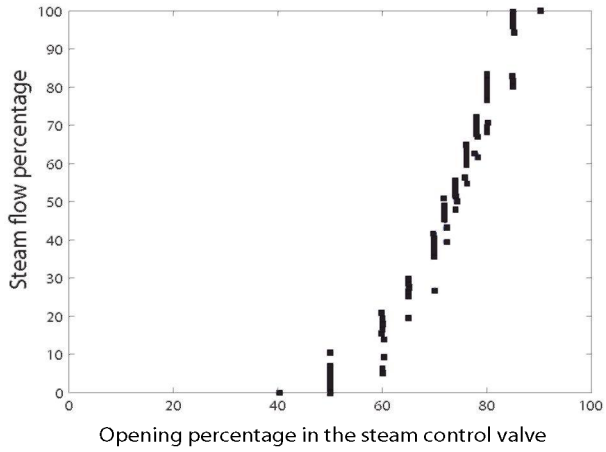

Figure B.3: Nonlinear behavior in the steam flow signal.

Behavior of the steam flow before being linearized its control valve. After 45% opening, the steam begins to flow into the shell.

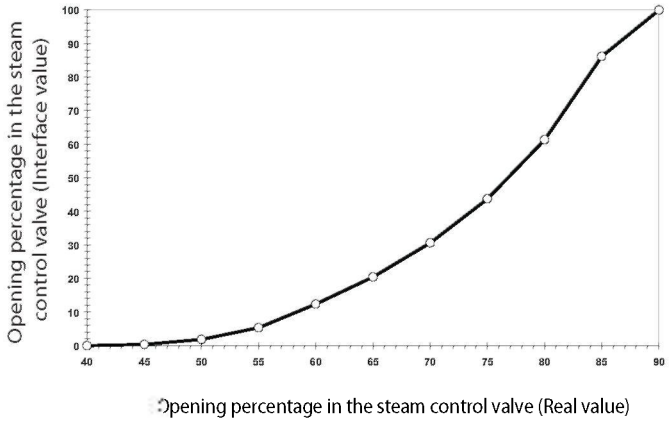

Figure B.4: Comparison of steam valve and steam flow.

There is a linear (direct) relationship between the real operating range of the steam valve and the displayed value of the interface.

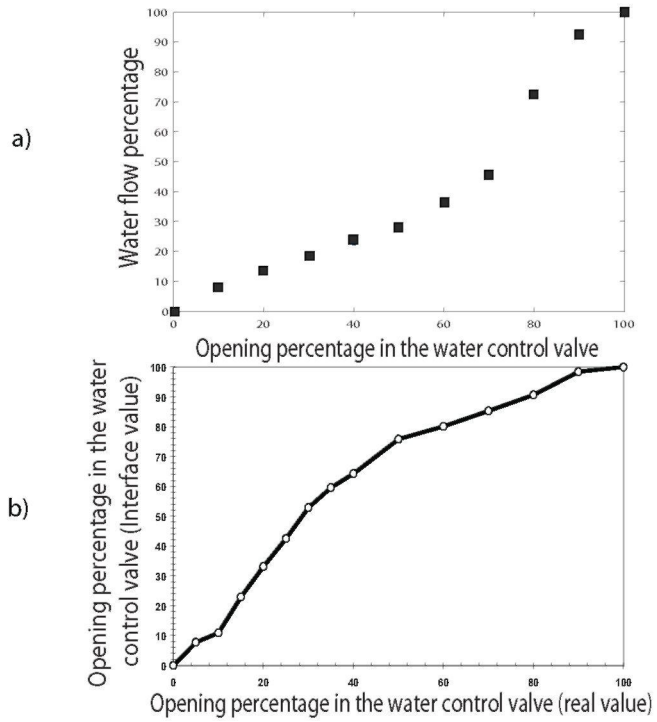

Figure B.5: Comparison of water valve and water flow.

Behavior of the water flow before being linearized its control valve (top plot). Direct relation between performance of the water control valve and interface value to be displayed to final user (bottom plot).

B.5 Interface Description

The interface allows to establish a bidirectional communication between the process and its operator. In this thesis work, three interfaces have been programmed:

- Monitoring interface.
- FDD interface using DPCA.
- FDD interface using diagnostic observers.

Figure B.6 describes the monitoring interface. This interface allows to monitor all process variables and to select the control mode. The steam flow and water flow are analytically interpreted by percentage and the water temperature are represented in Celsius degrees. In this thesis work, both FDD methods are tested only on opened loop (manual mode).

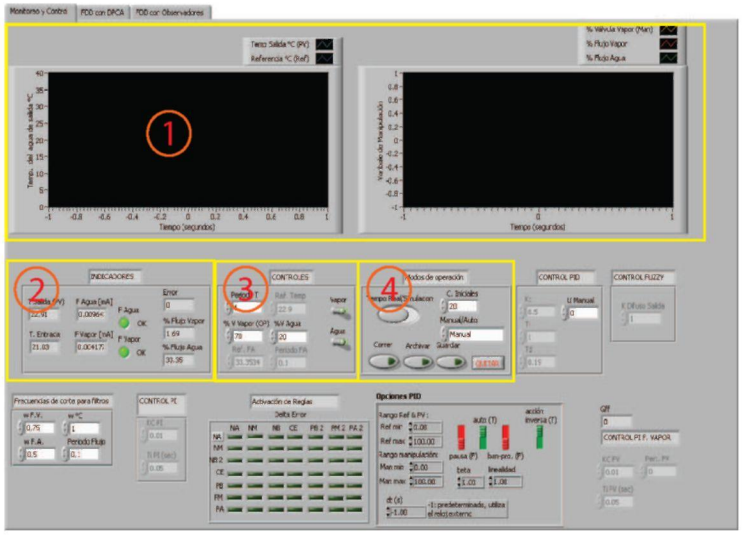

Figure B.6: Description of the monitoring interface.

(1) Plotting area of all process variables. (2) It indicates numerically the sensors measurements. (3) It controls the process through valves (pneumatic and solenoid). (4) It selects the control mode.

Figure B.7 describes the used interface for detecting faults using the DPCA method, whereas, Figure B.8 allows to detect and isolate faults using a set of diagnostic observers. Both interfaces can have equal or different sample time. Each interface has a button which allows to select the training stage and another second button which allows to use the method for its FDD task.

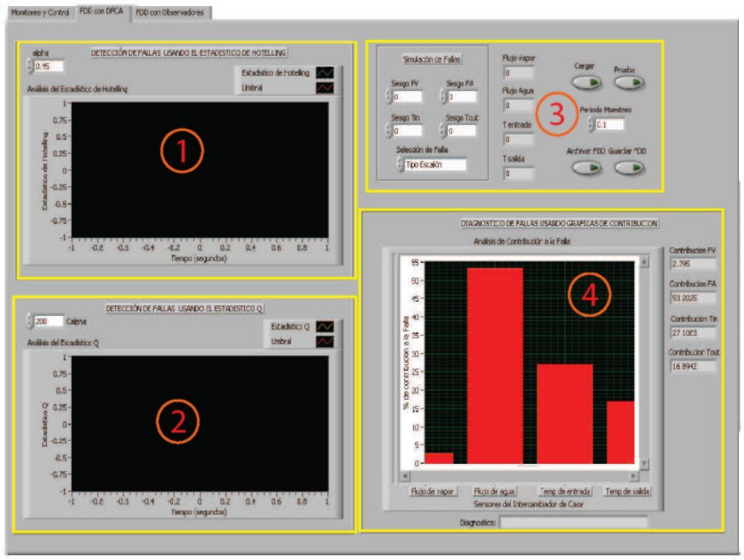

Figure B.7: Description of the DPCA interface.

(1) Plotting area for the Hotelling statistic and its threshold. (2) Plotting area for the Q statistic and its threshold. (3) It selects the training or testing stage; it allows to simulate abrupt or gradual faults in all sensors. (4) It isolates a fault using contribution plots.

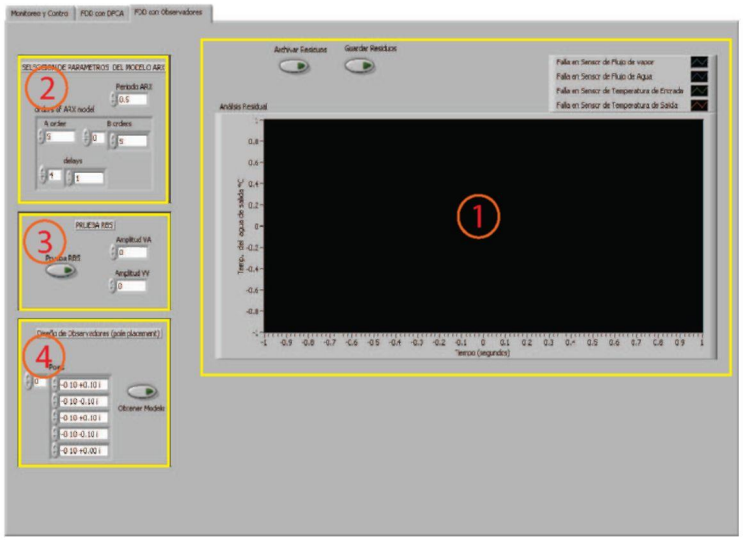

Figure B.8: Description of the diagnostic observers interface.

(1) Plotting area for the residuals. (2) It selects the number of parameters in the ARX model. (3) It executes the RBS test. (4) It selects the desired poles for designing the observer feedback matrix K_e.

Appendix C

Fault Detection and Diagnosis using PCA

PCA has been trained and tested under the same process operating conditions than the DPCA method. In this appendix Thus, it is possible to compare the performance of both approaches (PCA and DPCA) based on the FDD task. The same deviations (i.e. abrupt faults) in all sensors have been introduced. Limitations of PCA are highlighted in this appendix.

C.1 Training Stage

The training stage allows to represent the system in a new space which involves the correlations among the variables and can reduce the data order through the elimination of these possible correlations. The heat exchanger has four sensors, in this manner the new space for reconstructing the original data contains four *eigenvalues* and its respective *eigenvectors*. The behavior of the sensors measurements used for characterizing the normal operation point can be seen in the figure 5.1.

The training process leads to get the next characterization of the normal operation point.

Table C.1: Characterization of the normal operating point (using PCA).
Each eigenvalue with its respective percentage of explained variance can be analyzed. With 3PC's, the variance can be explained up 90%.

Principal Component	**Eigenvalue**	**Percentage of explained variance**	**Percentage of cumulative explained variance**
1	2.1331	53.33%	53.33%
2	1.0473	26.18%	79.51%
3	0.4713	11.78%	91.29%
4	0.3439	8.60%	99.89%

Using the proposal of Gudi *et al.* [9] for selecting how many principal components should be retained, was plotted the variance explained versus the number of PC. Figure C.1 shows how much variance can be explained for

each eigenvalue using PCA.

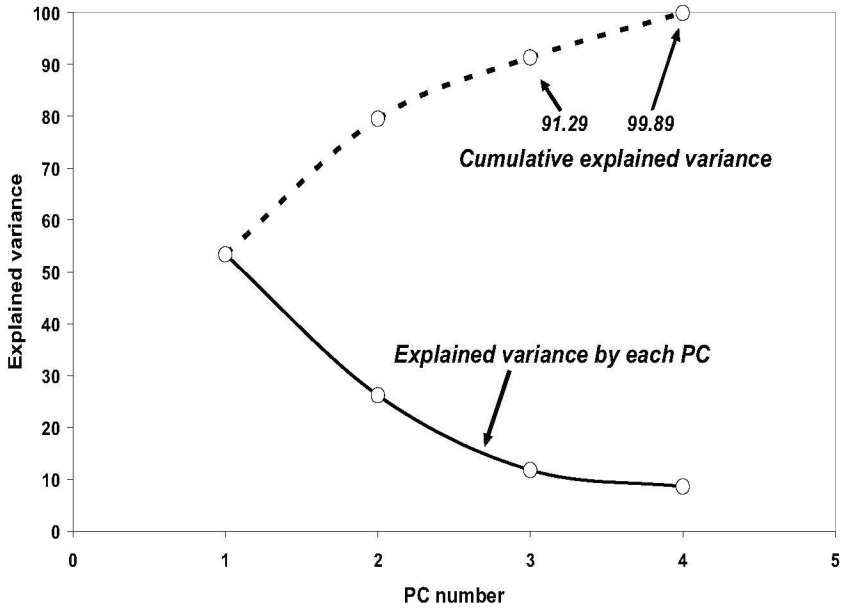

Figure C.1: Explained variance by each PC.
With three PC can be explained the 91.29% of variance, very closely to explained variance by all components.

The reconstruction of the real data can be generated based on the characterization of the normal operation condition shown in Table C.1. Using 2 principal components, the 79.51% of precision was achieving whereas with 3 PC's the PCA algorithm can explain the process real data with a precision of 91.29%, see Figure C.1. The transformed data will be more similar to real data if the PCA algorithm uses a major quantity of PC for explaining the sensors measurements. The number of PC to be retained using PCA will be three.

C.2 Testing Stage

Knowing the normal behavior of the process, the FDD system must be capable of detecting abnormal events. The used tools for detecting and isolating faults can be reviewed in the sections 3.1.1 and 3.1.2.

When an abrupt fault is simulated in the inlet temperature sensor with the behavior shown in Figure 4.5(up) and magnitude presented in the Table 4.2, three principal components are enough for detecting this failure using the PCA algorithm. Figure C.2 shows that the T^2 and Q statistics overshoot their control limits when the fault appears; the statistics and their thresholds can be calculated through the equations 3.14, 3.16, 3.15 and 3.17.

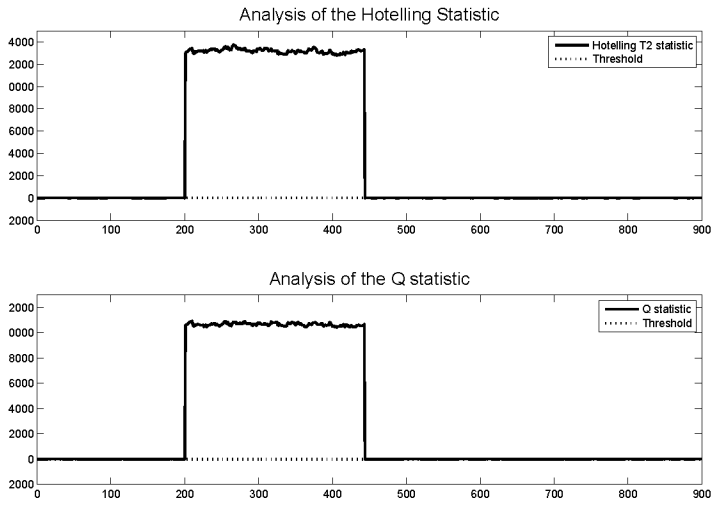

Figure C.2: Detection analysis for a fault simulated in the inlet temperature sensor using PCA.

Q and T^2 statistics overshoot their control limits at time 200 when an abrupt fault is activated, and they come back to zero when the fault is deactivated at time 440.

Although the statistics reflect the time when the fault occurs, it is necessary to create the error contribution plot for isolating the fault (equation 3.18). Contribution plots indicate which variables are more associated hypothetically to the fault since there exist more fault cases. The error in each sensor, which is generated when the fault occurs, can be shown in the figure C.3. All signals are deviated although the fault is originated only in the inlet temperature transmitter.

It is important to consider that the inlet temperature shows the greatest deviation since has the minor standard deviation and a fault in this signal will produce a major bias due to data normalization.

Due that all signals are deviated because the reconstructed data involve the total data variation and explain it in the principal components space, it is impossible to diagnosis the fault analyzing only using PCA. For isolating the fault it is necessary to calculate the error contribution. Thus, the figure C.2 shows that the inlet temperature signal has the greater percentage of the total error.

Detection and Diagnosis results are similar when abrupt faults are simulated in the steam and water flow sensor. Behavior of these faults is shown in Figure 4.5 whereas their respective magnitudes are presented in Table 4.2. Figure C.5(a) shows that the T^2 and Q statistics overshoot their control limits when a fault is activated in the steam flow sensor whereas Figure C.5(b) leads to diagnosis of this fault. Similarly, an abrupt fault in the water flow sensor is detected through the T^2 and Q statistics using equations 3.14, 3.16, 3.15 and 3.17 (see Figure C.6(a)) whereas the equation 3.18 allows to calculate the individual error for isolating this fault. Figure C.6(b) shows that the highest error is contributed by water flow sensor.

When an abrupt fault is simulated in the outlet temperature sensor (bias of $2^{\circ}C$), T^2 and $\mathbf{Q}$ statistics can detect

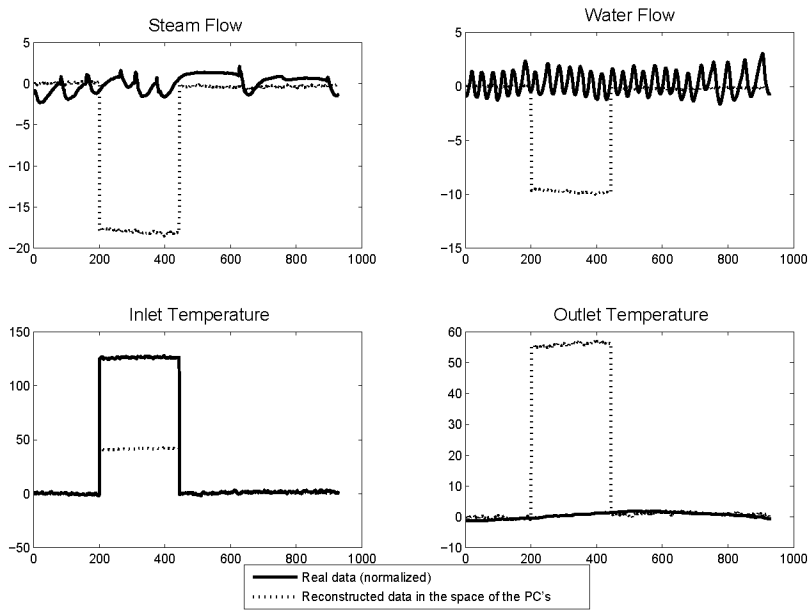

Figure C.3: Reconstructed data using PCA.

When occurs a fault in the inlet temperature sensor, all signals are deviated because the reconstructed data involve the total data variation and explain it in the principal components space. This is the major reason because PCA by itself can not isolate faults.

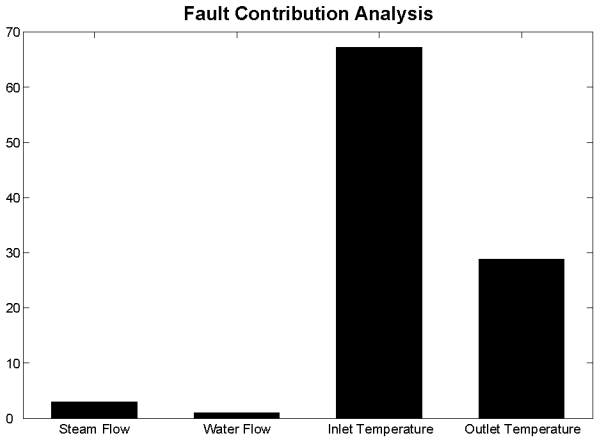

Figure C.4: Contribution Plot for diagnosing a fault in the system based on PCA residuals.
Due that 68% of total error corresponds to inlet temperature signal, the fault was originated in this sensor/transmitter.

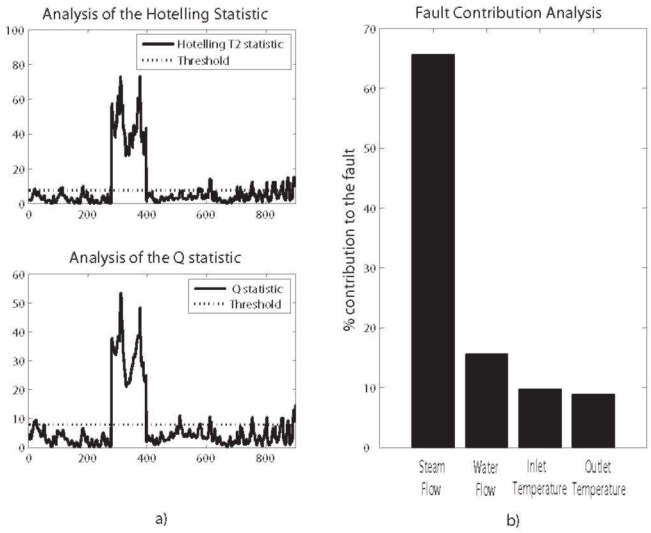

Figure C.5: Diagnostic and detection analysis for a fault simulated in the steam flow sensor using PCA.

Q and T^2 statistics overshoot their control limits at time 280 when an abrupt fault is activated in the steam flow sensor, and they come back close to zero when the fault is deactivated at time 395 (left). Contribution Plot for diagnosing a fault in the system based on PCA residuals shows that 66% of total error corresponds to steam flow signal, the fault was originated in this sensor/transmitter (right).

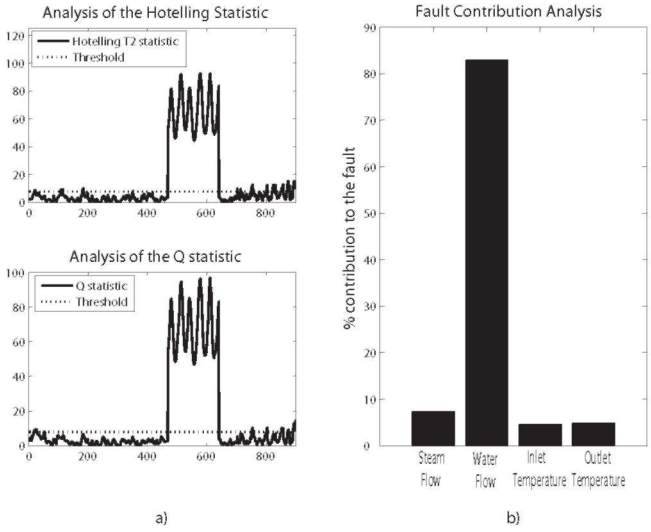

Figure C.6: Diagnostic and detection analysis for a fault simulated in the water flow sensor using PCA.

Q and T^2 statistics overshoot their control limits at time 465 when an abrupt fault is activated in the water flow sensor, and they come back close to zero when the fault is deactivated at time 640 (left). Contribution Plot for diagnosing a fault in the system based on PCA residuals shows that 83% of total error corresponds to steam flow signal, the fault was originated in this sensor/transmitter (right).

it, however **its isolation is not possible**. Figure C.7 shows the detection and diagnosis results when is simulated a fault in this sensor with similar behavior to an abrupt fault shown in Figure 4.5.

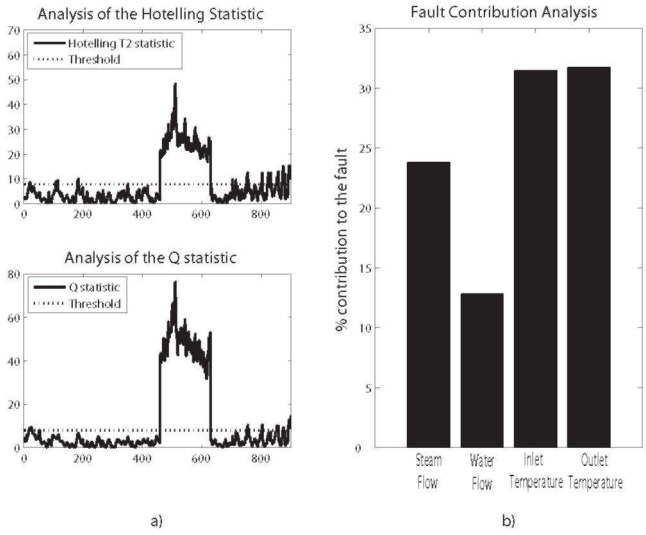

Figure C.7: Diagnostic and detection analysis for a fault simulated in the outlet temperature sensor using PCA.

Q and T^2 statistics overshoot their control limits at time 460 when an abrupt fault is activated in the outlet temperature sensor, and they come back close to zero when the fault is deactivated at time 630 (left). Contribution Plot based on PCA residuals **can not isolate this fault because all signals have similar contribution to total error** (right).

Appendix D

Complementary results using the DPCA Method

D.1 Training Process using the DPCA Method

For training the FDD system based on the DPCA method, it is necessary to follow the steps shown in the figure D.1.

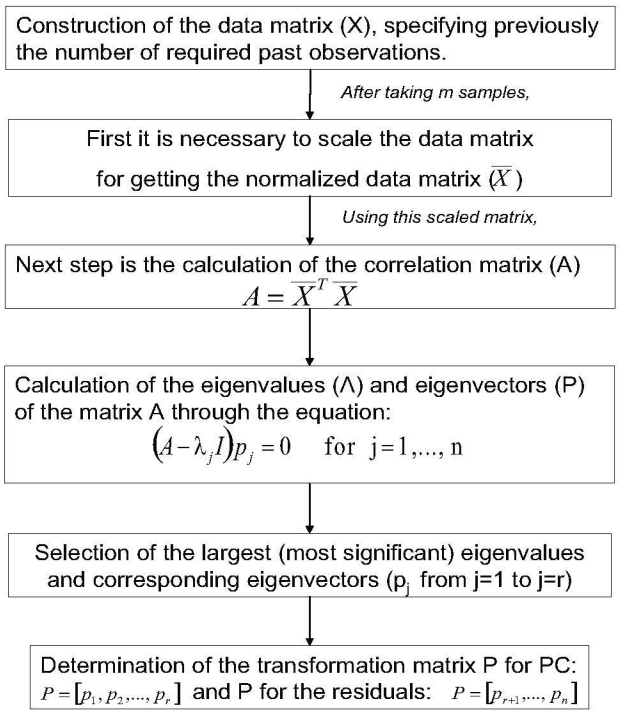

Figure D.1: Training process of FDD system using the DPCA method.
Steps for getting the eigenvalues and eigenvectors which characterize the normal process conditions.

Figure D.2 presents a numerical example with intermediate results of the DPCA training stage.

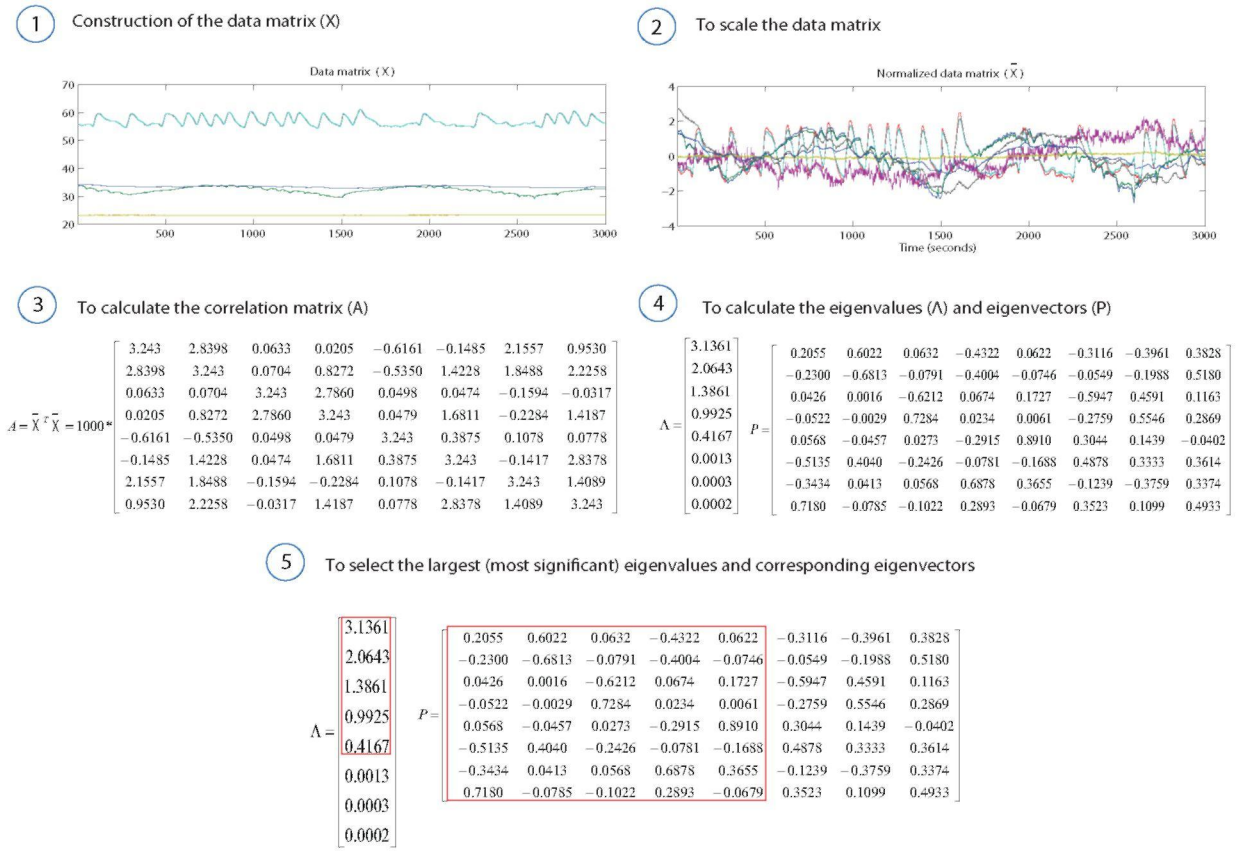

Figure D.2: Numerical example of the DPCA training stage.
5 steps in order to characterize the normal operating conditions.

Knowing the eigenvalues and determination of P for PC and for the residuals, can be used DPCA for detecting faults and Contribution Plots for isolating them. All equations for training the FDD system based on the DPCA method can be reviewed in detail in the Methodology chapter.

Next, the data logger of Matlab is presented for this numeric example, furthermore, the calculus in order to test the method is described:

% To obtain the data matrix for the training stage

load datanormaltraining.txt;

Xtrain=datanormaltraining;

% To obtain the data matrix for the testing stage

```
load datanormaltesting.txt;

Xtest=datanormaltesting;

% TRAINING STAGE: 1) To calculate the correlation matrix

B=Xtrain'*Xtrain;
M=size(Xtrain);
A=B/M(:,1);

% 2) To calculate the eigenvalues and eigenvectors

[Eigenvec,Eigenval]=eig(A);

% 3) To adapt the eigenvalues and eigenvectors

PC1 = sum(Eigenval(:,8));
PC2 = sum(Eigenval(:,7));
PC3=sum(Eigenval(:,6));
PC4=sum(Eigenval(:,5));
PC5 = sum(Eigenval(:,4));
PC6 = sum(Eigenval(:,3));
PC7=sum(Eigenval(:,2));
PC8=sum(Eigenval(:,1));

EigVect1=Eigenvec(:,8);
EigVect2=Eigenvec(:,7);
EigVect3=Eigenvec(:,6);
EigVect4=Eigenvec(:,5);
EigVect5=Eigenvec(:,4);
EigVect6=Eigenvec(:,3);
EigVect7=Eigenvec(:,2);
EigVect8=Eigenvec(:,1);

% 4) To select the number of principal components

%For 5 PC

Qp=[EigVect1, EigVect2, EigVect3, EigVect4, EigVect5];
Qr=[EigVect6, EigVect7, EigVect8];
Teta1=((PC6)^2+(PC7)^2+(PC8)^2);
Teta2=((PC6)^4+(PC7)^4+(PC8)^4);
```

```
Teta3=((PC6)^6+(PC7)^6+(PC8)^6);
k=5;
Lambda=[PC1,0,0,0,0;0,PC2,0,0,0;0,0,PC3,0,0;0,0,0,PC4,0;0,0,0,0,PC5];

% TESTING STAGE: 1) To obtain the Q statistic

N=size(Xtest);
m=N(:,1);

Q=0;
for i=1:m
Q(i) =((eye(8)-(Qr*Qr'))*Xtest(i,:)')'*((eye(8)-(Qr*Qr'))*Xtest(i,:)');
end

% 2) To obtain the Q threshold

alpha=.99;
hcero = 1-((2*Teta1*Teta3)/(3*(Teta2)^2));
Calpha = 1-alpha;
Qalpha = Teta1*(((hcero*Calpha*sqrt(2*Teta2))/Teta1)+1+((Teta2*hcero*(hcero-1))/(Teta1)^2))^(1/hcero);

Umbral=0;
for i=1:m
Umbral(i)=Qalpha;
end

% 3) To obtain the T2 statistic

T2=0;
for i=1:m
T2(i) = Xtest(i,:)*Qp*inv(Lambda)*Qp'*Xtest(i,:)';
end

% 4) To obtain the T2 threshold

T2alpha=(((m-1)*k)/(m-k))*Finv(alpha,k,(m-k));

Umbral2=0;
for i=1:m
Umbral2(i)=T2alpha;
end

% 5) To calculate the contribution plots
```

```
Xtransformed=Xtest*Qr*Qr';

for i=1:N
Residue(i,:) = Xtest(i,:) - Xtransformed(i,:);
SEC(i) = Residue(i,:)*Residue(i,:)';
end

ResidueFV=Residue(:,1);
ResidueFV1=Residue(:,2);
ResidueFA=Residue(:,3);
ResidueFA1=Residue(:,4);
ResidueTin=Residue(:,5);
ResidueTin1=Residue(:,6);
ResidueTout=Residue(:,7);
ResidueTout1=Residue(:,8);

SumResidue = ResidueFV'*ResidueFV + ResidueFV1'*ResidueFV1 + ResidueFA'*ResidueFA + ResidueFA1'*ResidueFA1
+ ResidueTin'*ResidueTin + ResidueTin1'*ResidueTin1 + ResidueTout'*ResidueTout + ResidueTout1'*ResidueTout1;

ContFV=(ResidueFV'*ResidueFV)/SumResidue;
ContFV1=(ResidueFV1'*ResidueFV1)/SumResidue;
Contsteam=ContFV+ContFV1;
ContFA=(ResidueFA'*ResidueFA)/SumResidue;
ContFA1=(ResidueFA1'*ResidueFA1)/SumResidue;
Contwater=ContFA+ContFA1;
ContTin=(ResidueTin'*ResidueTin)/SumResidue;
ContTin1=(ResidueTin1'*ResidueTin1)/SumResidue;
ContTinput=ContTin+ContTin1;
ContTout=(ResidueTout'*ResidueTout)/SumResidue;
ContTout1=(ResidueTout1'*ResidueTout1)/SumResidue;
ContToutput=ContTout+ContTout1;

Contributions=[Contsteam;Contwater;ContTinput;ContToutput];

% Plots

figure
subplot(2,1,1)
plot(T2) title('Analysis in the T2 statistic')
hold on
plot(Umbral2,'r')
hold off
```

```
subplot(2,1,2)
plot(Q) title('Analysis in the Q statistic')
hold on
plot(Umbral,'r')
hold off

figure bar(Contributions) title('Fault Contribution Analysis')
ylabel('Percentage of contribution to the fault')
xlabel(set(gca,'XTickLabel','Steam Flow';'Water Flow';'Inlet Temperature';'Outlet Temperature'))
```

D.2 Comparative Results using 4 PC versus 5 PC

From the characterization of the process normal behavior using the DPCA algorithm, 4 PC can explain the 94.737% of total variance, whereas, 5 PC explains the 99.946%.

For comparing the diagnostic results using 4 PC and 5 PC in the DPCA method, a fault in the outlet temperature was simulated in both cases. Figure (D.3) and figure (D.4) shows the diagnostic results using 4 PC and 5 PC respectively.

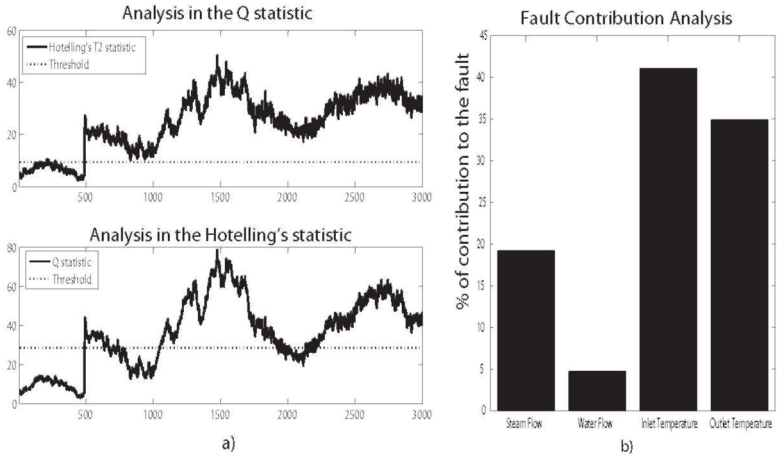

Figure D.3: FDD analysis for a simulated abrupt fault in the outlet temperature sensor using 4 PC.

Q and T^2 statistics overshoot their control limits at time 500 when an abrupt fault is activated in the outlet temperature sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals can not isolate this fault correctly using 4 PC (right).

Using 4 PC, is not possible to isolate a fault in the outlet temperature sensor correctly. Furthermore, the Q statistic drops below its threshold when the fault is still activated. On the other hand, figure (D.4) shows an accurate detection and correct diagnosis of the same fault, using 5 PC.

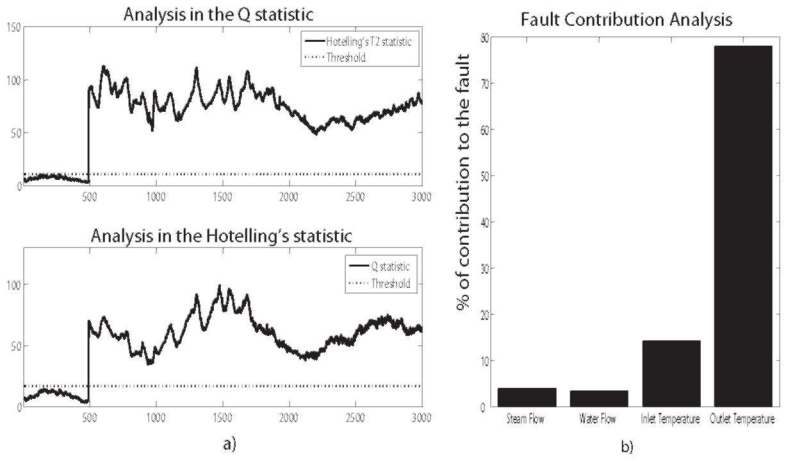

Figure D.4: FDD analysis for a simulated abrupt fault in the outlet temperature sensor using 5 PC.

Q and T^2 statistics overshoot their control limits at time 500 when an abrupt fault is activated in the outlet temperature sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 78% of total error corresponds to outlet temperature signal, thus the fault was originated in this sensor/transmitter (right).

D.3 Detection and Diagnosis of Abrupt Faults in TT1 and FT1 using DPCA method

D.3.1 Abrupt Fault in the inlet temperature sensor (TT1)

Online monitoring of Q and T^2 statistics describes a fault detection in the process when was introduced a fault (bias of $2°$C) in the inlet temperature sensor at time 361, see figure D.5(a). The figure D.5(b) shows that fault diagnosis was achieved too.

D.3.2 Abrupt Fault in the water flow sensor (FT1)

At time 135 was simulated a fault in the water flow sensor (bias of 6%) with similar behavior to an abrupt fault, which is shown in Figure 4.5. Fault simulated can be detected correctly for Q and T^2 indicators, see figure D.6(a). The diagnosis task is clearly highlighted by the contribution plot shown in figure D.6(b).

D.4 Detection and Diagnosis of Gradual Faults in TT1 and FT1 using DPCA method

D.4.1 Gradual Fault in the inlet temperature sensor (TT1)

After 3 and 10 seconds respectively, the Q and T^2 statistics describe a fault detection in the process when was introduced a gradual fault in the inlet temperature sensor at time 175, see figure D.7(a). As it is evident, the behavior of

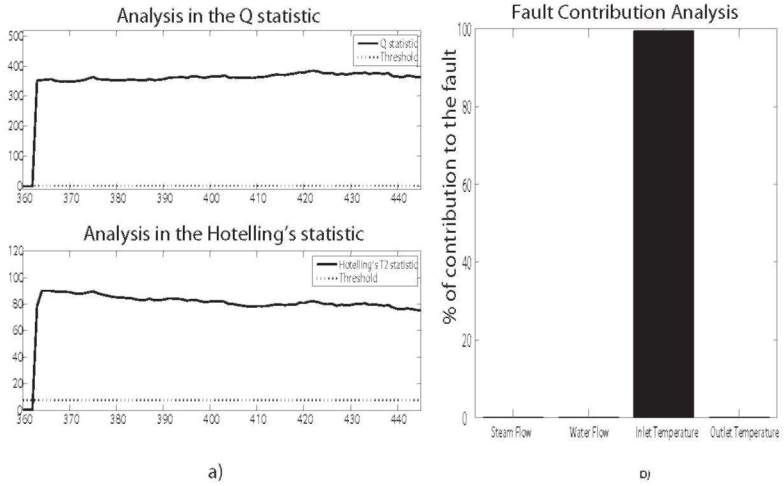

Figure D.5: FDD analysis for a simulated abrupt fault in the inlet temperature sensor using DPCA method.

Q and T^2 statistics overshoot their control limits at time 361 when an abrupt fault is activated in the inlet temperature sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that total error (100%) corresponds to inlet temperature signal, thus the fault was originated in this sensor/transmitter (right).

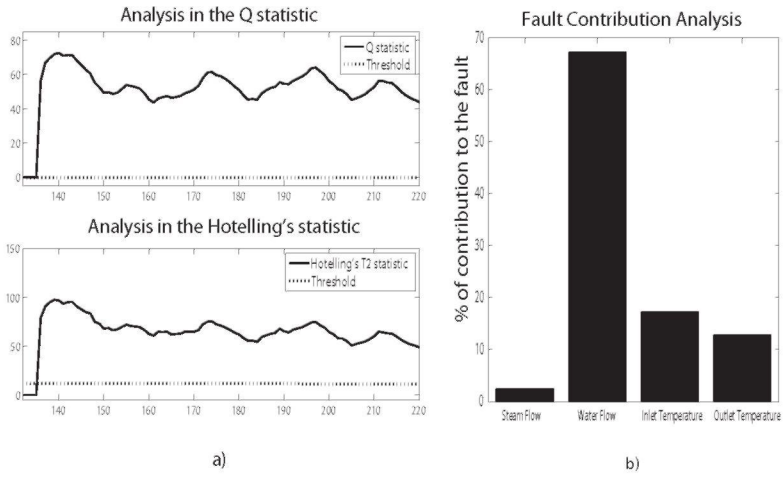

Figure D.6: FDD analysis for a simulated abrupt fault in the water flow sensor using DPCA method.

Q and T^2 statistics overshoot their control limits at time 135 when an abrupt fault is activated in the water flow sensor (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 67% of total error corresponds to water flow signal, thus the fault was originated in this sensor/transmitter (right).

both statistics corresponds to the fault behavior which was introduced. The figure D.7(b) shows that fault diagnosis was achieved too.

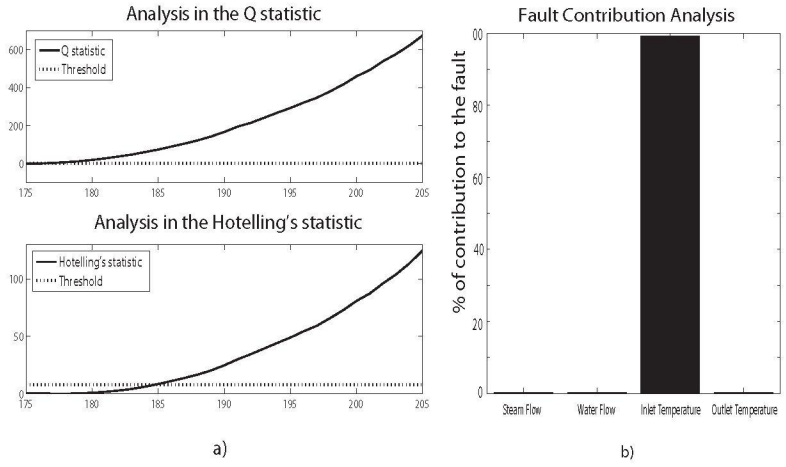

Figure D.7: FDD analysis for a simulated gradual fault in the inlet temperature sensor using DPCA method.

Q and T^2 statistics overshoot their control limits after 3 and 10 seconds respectively when was introduced a gradual fault in the inlet temperature sensor at time 175 (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that total error (100%) corresponds to inlet temperature signal, thus the fault was originated in this sensor/transmitter (right).

D.4.2 Gradual Fault in the water flow sensor (FT1)

It can be seen in figure D.8(a) that Q and T^2 indicators overshoot their control limits and detect a gradual fault after 60 and 70 seconds respectively once the fault has been simulated in the water flow sensor, at time 0. It is important to specify that the slope of the gradual fault is 0.1% per second, thus after 60 and 70 seconds the total bias will be between 6 and 7 percent; value which corresponds to normal oscillation rank (6%). The diagnosis task is clearly highlighted by the contribution plot shown in figure D.8(b).

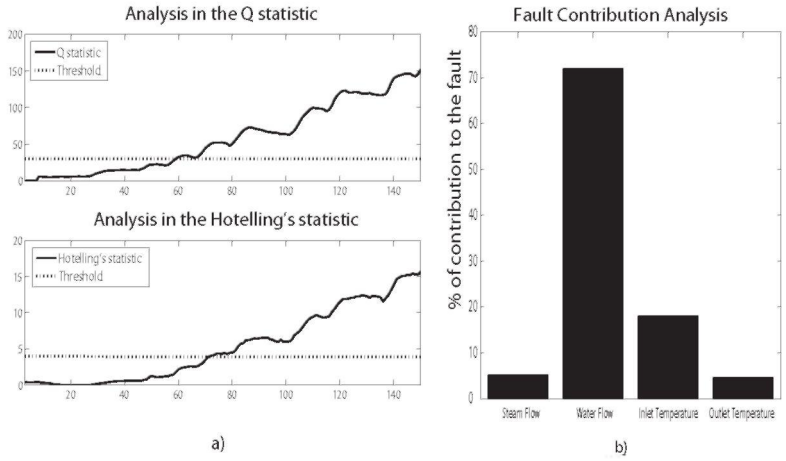

Figure D.8: FDD analysis for a simulated gradual fault in the water flow sensor using DPCA method.

Q and T^2 statistics overshoot their control limits after 60 and 70 seconds respectively when was introduced a gradual fault in the water flow sensor at time 0 (left). Contribution Plot for diagnosing a fault in the system based on DPCA residuals shows that 73% of total error corresponds to water flow signal, thus the fault was originated in this sensor/transmitter (right).

Appendix E

Design of the Observer Feedback Matrix Based on Pole Placement

Ogata [59] presents a detailed description in the design of the observer feedback matrix based on pole placement.

Since it is wanted that the dynamics of the observer be much faster than the system itself, it is necessary to place the observer poles at least five times closer to the origin than the dominant poles of the discrete system.

The characteristic polynomial which describes the dynamics of the observer is given as

$$(z-\gamma_1)(z-\gamma_2)\ldots(z-\gamma_\eta) = z^\eta + \alpha_1 z^{\eta-1} + \alpha_2 z^{\eta-2} + \ldots + \alpha_{\eta-1} z + \alpha_\eta \tag{E.1}$$

where, γ_i are the observer poles which determine the convergence velocity of the observer state to the process state.

Isermann [54] specifies that there is some freedom in the design of the observer feedback matrix $\mathbb{K}_e$. Ogata [59] describes 4 different ways for calculating the observer feedback matrix; the four methods are based on pole placement:

- METHOD 1. The observer feedback matrix can be computed through the next relation:

$$\mathbb{K}_e = \mathbb{M}\begin{bmatrix} \gamma_\eta - \alpha_\eta \\ \gamma_{\eta-1} - \alpha_{\eta-1} \\ \vdots \\ \gamma_1 - \alpha_1 \end{bmatrix} = (\mathbb{WN})^{-1}\begin{bmatrix} \gamma_\eta - \alpha_\eta \\ \gamma_{\eta-1} - \alpha_{\eta-1} \\ \vdots \\ \gamma_1 - \alpha_1 \end{bmatrix}$$

  where, the matrix $\mathbb{W}$ contains the coefficients α_i and $\mathbb{N}$ is the inverse of the observability matrix. Coefficients α_i determine the original characteristic polynomial; whereas the coefficients γ_i represent to the desired dynamic.

- METHOD 2. Ackermann formula:

$$\mathbb{K}_e = \phi(\mathbb{G}) \begin{bmatrix} \mathbb{C} \\ \mathbb{C}\mathbb{G} \\ \vdots \\ \mathbb{C}\mathbb{G}^{\eta-1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

where, $\phi(\mathbb{G})$ is defined as:

$$\phi(\mathbb{G}) = \mathbb{G}^{\eta} + \gamma_1 \mathbb{G}^{\eta-1} + \ldots + \gamma_{\eta-1}\mathbb{G} + \gamma_{\eta}\mathbb{I} \tag{E.2}$$

- METHOD 3. Using the *eigenvectors* (ν_i).

$$\mathbb{K}_e = \begin{bmatrix} \nu_1 \\ \nu_2 \\ \vdots \\ \nu_{\eta} \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

where, ν_i is defined as:

$$\nu_i = \mathbb{C}(\mathbb{G} - \gamma_i \mathbb{I})^{-1} \tag{E.3}$$

- METHOD 4. Method of direct substitution

The unknown elements of the matrix $\mathbb{K}_e$ can be computed through the next equality:

$$det|z\mathbb{I} - \mathbb{G} + \mathbb{K}_e\mathbb{C}| = (z - \gamma_1)(z - \gamma_2)\ldots(z - \gamma_{\eta}) \tag{E.4}$$

where, the coefficients γ_i represent the desired dynamic.

In this case, the observer feedback matrix is computed through the the method of direct substitution (METHOD 4). Figure E.1 presents a numerical example with intermediate results of the design of the observer feedback matrix.

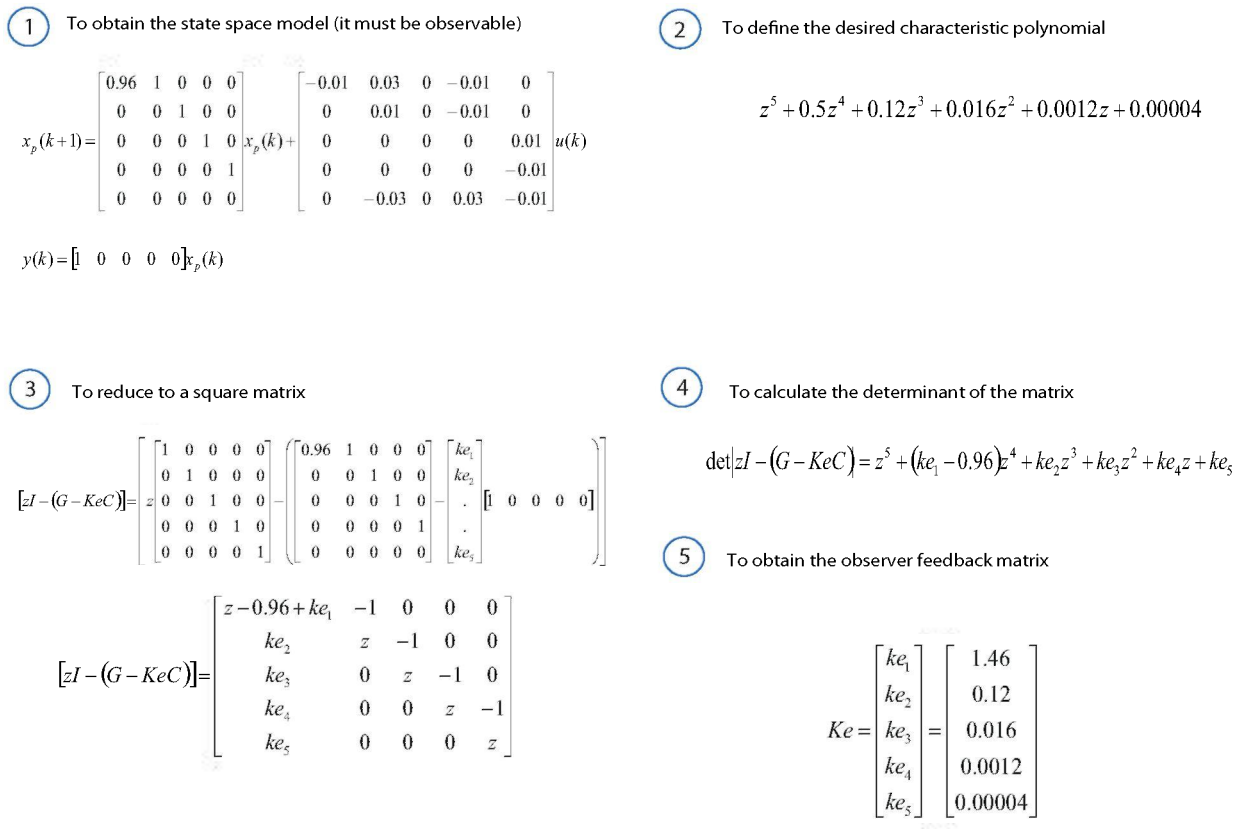

Figure E.1: Numerical example of the design of the observer feedback matrix.
5 steps in order to design the matrix $\mathbb{K}_e$ using the method of direct substitution.

Appendix F

Complementary results using Diagnostic Observers

F.1 Detection and Diagnosis of Abrupt Faults in TT1 and FT1 using Diagnostic Observers

F.1.1 Abrupt Fault in the inlet temperature sensor (TT1)

Online monitoring of all residual signals describes a fault detection in the process when was introduced a fault (bias of 2°C) in the inlet temperature sensor at time 10, see figure F.1.

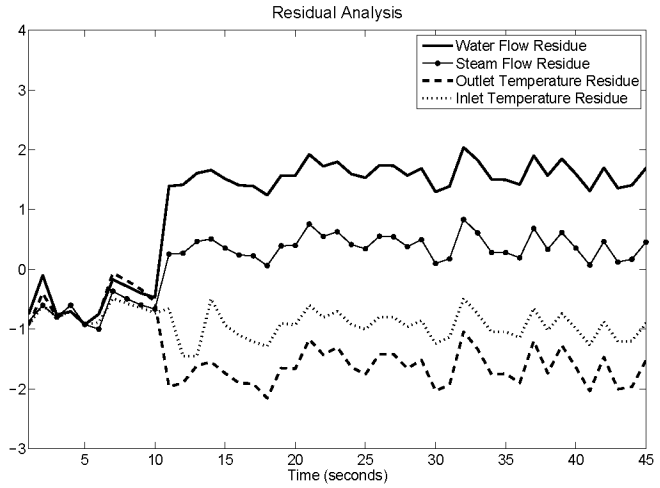

Figure F.1: FDD analysis for a simulated abrupt fault in the inlet temperature using diagnostic observers.

All residues except the inlet temperature residue, are deviated from their nominal behavior at time 10 when an abrupt fault is activated in the inlet temperature signal, thus the fault was originated in this sensor/transmitter.

F.1.2 Abrupt Fault in the water flow sensor (FT1)

At time 9 was simulated a fault in the water flow sensor (bias of 6%) with similar behavior to an abrupt fault, which is shown in Figure 4.5. Fault simulated can be detected and isolated in parallel for the residual analysis, see figure F.2.

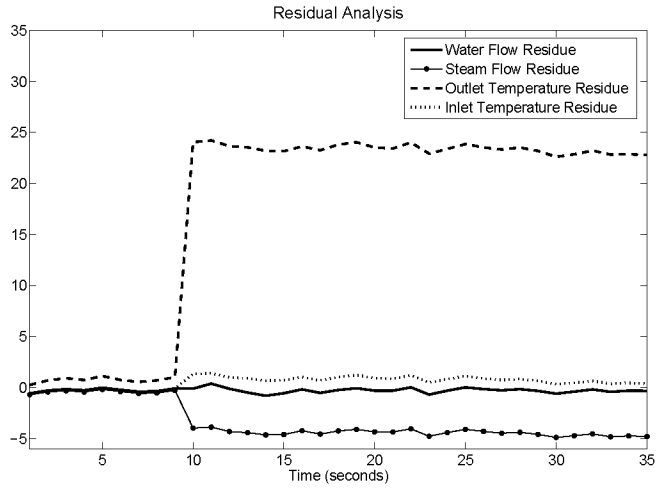

Figure F.2: FDD analysis for a simulated abrupt fault in the water flow sensor using diagnostic observers.

All residues except the water flow residue, are deviated from their nominal behavior at time 9 when an abrupt fault is activated in the water flow signal, thus the fault was originated in this sensor/transmitter.

F.2 Detection and Diagnosis of Gradual Faults in TT1 and FT1 using Diagnostic Observers

F.2.1 Gradual Fault in the inlet temperature sensor (TT1)

After 10 seconds, the water and steam flow residues are deviated positively when was introduced a gradual fault in the inlet temperature sensor at time 10, see figure F.3. However, the outlet temperature residue is deviated after 110 seconds once the fault was activated; this period corresponds to 11% of the signal deviation provided by the fault.

F.2.2 Gradual Fault in the water flow sensor (FT1)

It can be seen in figure F.4 that all residues except the water flow residue, are deviated from its normal behavior. After 3 seconds the inlet temperature residue and steam flow residue are affected once the fault has been simulated in the water flow sensor, at time 0, whereas the outlet temperature residue is deviated after 9 seconds.

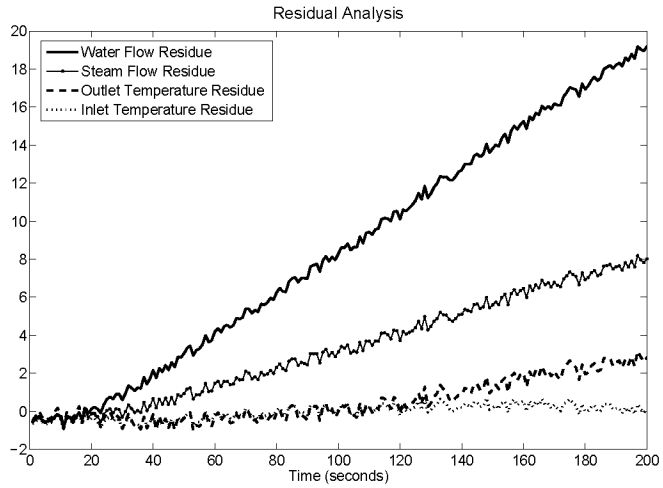

Figure F.3: FDD analysis for a simulated gradual fault in the inlet temperature using diagnostic observers.

All residues except the inlet temperature residue, are deviated positively from their nominal behavior when was introduced a gradual fault in the inlet temperature sensor at time 10, thus the fault was originated in this sensor/transmitter.

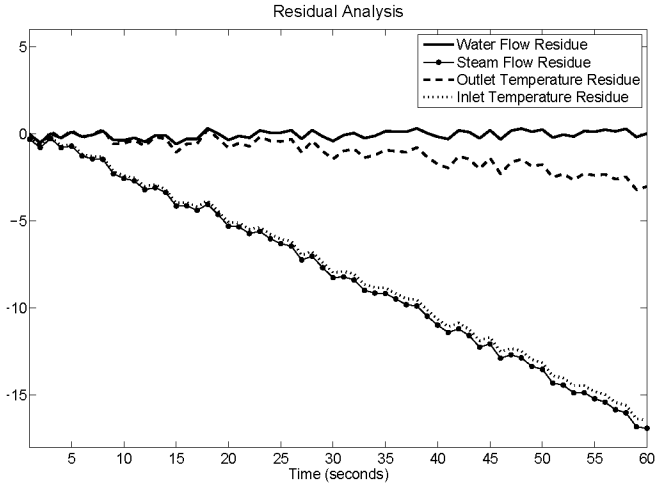

Figure F.4: FDD analysis for a simulated gradual fault in the water flow sensor using diagnostic observers.

All residues except the water flow residue, are deviated negatively from their nominal behavior when was introduced a gradual fault in the water flow sensor at time 0, thus the fault was originated in this sensor/transmitter.

Appendix G

Tables of the Fisher's F-distribution

A Fisher Distribution is commonly used for modelling the distribution of 3-dimensional orientation vectors, such as the distribution of joint orientations (pole vectors) on a sphere. A Fisher Distribution describes the angular distribution of orientations about a mean orientation vector, and is symmetric about the mean.

The Figures G.1-G.2 represent the tables of the Fisher's F-distribution, taking account 99% of confidence. In this case, the freedom degrees depend on the number of retained PC and the number of sampled observations.

Fisher Distribution

$\alpha = 99\%$

$F - distribution : F(\alpha, \nu_1, \nu_2)$

ν_1 = freedom degrees of the numerator

ν_2 = freedom degrees of the denominator

ν_2 \ ν_1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	4052.185	4999.34	5403.534	5624.257	5763.955	5858.95	5928.334	5980.954	6022.397	6055.925	6083.399	6106.682	6125.774	6143.004	6156.974	6170.012	6181.188	6191.432	6200.746	6208.662
2	98.502	99	99.164	99.251	99.302	99.331	99.357	99.375	99.39	99.397	99.408	99.419	99.422	99.426	99.433	99.437	99.441	99.444	99.448	99.448
3	34.116	30.816	29.457	28.71	28.237	27.911	27.671	27.489	27.345	27.228	27.132	27.052	26.983	26.924	26.872	26.826	26.786	26.751	26.719	26.69
4	21.198	18	16.694	15.977	15.522	15.207	14.976	14.799	14.659	14.546	14.452	14.374	14.308	14.249	14.198	14.154	14.114	14.079	14.048	14.019
5	16.258	13.274	12.06	11.392	10.967	10.672	10.456	10.289	10.158	10.051	9.963	9.888	9.825	9.77	9.722	9.68	9.643	9.609	9.58	9.553
6	13.745	10.925	9.78	9.148	8.746	8.466	8.26	8.102	7.976	7.874	7.79	7.718	7.657	7.605	7.559	7.519	7.483	7.451	7.422	7.396
7	12.246	9.547	8.451	7.847	7.46	7.191	6.993	6.84	6.719	6.62	6.538	6.469	6.41	6.359	6.314	6.275	6.24	6.209	6.181	6.155
8	11.259	8.649	7.591	7.006	6.632	6.371	6.178	6.029	5.911	5.814	5.734	5.667	5.609	5.559	5.515	5.477	5.442	5.412	5.384	5.359
9	10.562	8.022	6.992	6.422	6.057	5.802	5.613	5.467	5.351	5.257	5.178	5.111	5.055	5.005	4.962	4.924	4.89	4.86	4.833	4.808
10	10.044	7.559	6.552	5.994	5.636	5.386	5.2	5.057	4.942	4.849	4.772	4.706	4.65	4.601	4.558	4.52	4.487	4.457	4.43	4.405
11	9.646	7.206	6.217	5.668	5.316	5.069	4.886	4.744	4.632	4.539	4.462	4.397	4.342	4.293	4.251	4.213	4.18	4.15	4.123	4.099
12	9.33	6.927	5.953	5.412	5.064	4.821	4.64	4.499	4.388	4.296	4.22	4.155	4.1	4.052	4.01	3.972	3.939	3.91	3.883	3.858
13	9.074	6.701	5.739	5.205	4.862	4.62	4.441	4.302	4.191	4.1	4.025	3.96	3.905	3.857	3.815	3.778	3.745	3.716	3.689	3.665
14	8.862	6.515	5.564	5.035	4.695	4.456	4.278	4.14	4.03	3.939	3.864	3.8	3.745	3.698	3.656	3.619	3.586	3.556	3.529	3.505
15	8.683	6.359	5.417	4.893	4.556	4.318	4.142	4.004	3.895	3.805	3.73	3.666	3.612	3.564	3.522	3.485	3.452	3.423	3.396	3.372
16	8.531	6.226	5.292	4.773	4.437	4.202	4.026	3.89	3.78	3.691	3.616	3.553	3.498	3.451	3.409	3.372	3.339	3.31	3.283	3.259
17	8.4	6.112	5.185	4.669	4.336	4.101	3.927	3.791	3.682	3.593	3.518	3.455	3.401	3.353	3.312	3.275	3.242	3.212	3.186	3.162
18	8.285	6.013	5.092	4.579	4.248	4.015	3.841	3.705	3.597	3.508	3.434	3.371	3.316	3.269	3.227	3.19	3.158	3.128	3.101	3.077
19	8.185	5.926	5.01	4.5	4.171	3.939	3.765	3.631	3.523	3.434	3.36	3.297	3.242	3.195	3.153	3.116	3.084	3.054	3.027	3.003
20	8.096	5.849	4.938	4.431	4.103	3.871	3.699	3.564	3.457	3.368	3.294	3.231	3.177	3.13	3.088	3.051	3.018	2.989	2.962	2.938
21	8.017	5.78	4.874	4.369	4.042	3.812	3.64	3.506	3.398	3.31	3.236	3.173	3.119	3.072	3.03	2.993	2.96	2.931	2.904	2.88
22	7.945	5.719	4.817	4.313	3.988	3.758	3.587	3.453	3.346	3.258	3.184	3.121	3.067	3.019	2.978	2.941	2.908	2.879	2.852	2.827
23	7.881	5.664	4.765	4.264	3.939	3.71	3.539	3.406	3.299	3.211	3.137	3.074	3.02	2.973	2.931	2.894	2.861	2.832	2.805	2.78
24	7.823	5.614	4.718	4.218	3.895	3.667	3.496	3.363	3.256	3.168	3.094	3.032	2.977	2.93	2.889	2.852	2.819	2.789	2.762	2.738
25	7.77	5.568	4.675	4.177	3.855	3.627	3.457	3.324	3.217	3.129	3.056	2.993	2.939	2.892	2.85	2.813	2.78	2.751	2.724	2.699
26	7.721	5.526	4.637	4.14	3.818	3.591	3.421	3.288	3.182	3.094	3.021	2.958	2.904	2.857	2.815	2.778	2.745	2.715	2.688	2.664
27	7.677	5.488	4.601	4.106	3.785	3.558	3.388	3.256	3.149	3.062	2.988	2.926	2.872	2.824	2.783	2.746	2.713	2.683	2.656	2.632
28	7.636	5.453	4.568	4.074	3.754	3.528	3.358	3.226	3.12	3.032	2.959	2.896	2.842	2.795	2.753	2.716	2.683	2.653	2.626	2.602
29	7.598	5.42	4.538	4.045	3.725	3.499	3.33	3.198	3.092	3.005	2.931	2.868	2.814	2.767	2.726	2.689	2.656	2.626	2.599	2.574
30	7.562	5.39	4.51	4.018	3.699	3.473	3.305	3.173	3.067	2.979	2.906	2.843	2.789	2.742	2.7	2.663	2.63	2.6	2.573	2.549
40	7.314	5.178	4.313	3.828	3.514	3.291	3.124	2.993	2.888	2.801	2.727	2.665	2.611	2.563	2.522	2.484	2.451	2.421	2.394	2.369
50	7.171	5.057	4.199	3.72	3.408	3.186	3.02	2.89	2.785	2.698	2.625	2.563	2.508	2.461	2.419	2.382	2.348	2.318	2.29	2.265
60	7.077	4.977	4.126	3.649	3.339	3.119	2.953	2.823	2.718	2.632	2.559	2.496	2.442	2.394	2.352	2.315	2.281	2.251	2.223	2.198
70	7.011	4.922	4.074	3.6	3.291	3.071	2.906	2.777	2.672	2.585	2.512	2.45	2.395	2.348	2.306	2.268	2.234	2.204	2.176	2.15
80	6.963	4.881	4.036	3.563	3.255	3.036	2.871	2.742	2.637	2.551	2.478	2.415	2.361	2.313	2.271	2.233	2.199	2.169	2.141	2.115
90	6.925	4.849	4.007	3.535	3.228	3.009	2.845	2.715	2.611	2.524	2.451	2.389	2.334	2.286	2.244	2.206	2.172	2.142	2.114	2.088
100	6.895	4.824	3.984	3.513	3.206	2.988	2.823	2.694	2.59	2.503	2.43	2.368	2.313	2.265	2.223	2.185	2.151	2.12	2.092	2.067
200	6.763	4.713	3.881	3.414	3.11	2.893	2.73	2.601	2.497	2.411	2.338	2.275	2.22	2.172	2.129	2.091	2.057	2.026	1.997	1.971
500	6.686	4.648	3.821	3.357	3.054	2.838	2.675	2.547	2.443	2.358	2.283	2.22	2.166	2.117	2.075	2.036	2.002	1.97	1.942	1.915
1000	6.66	4.626	3.801	3.338	3.036	2.82	2.657	2.529	2.425	2.339	2.265	2.203	2.148	2.099	2.056	2.018	1.983	1.952	1.923	1.897

Figure G.1: Table of the Fisher's F-distribution with 99% of confidence limit.

Fisher Distribution

$\alpha = 99\%$

$F - distribution : F(\alpha, \nu_1, \nu_2)$

ν_1 = freedom degrees of the numerator

ν_2 = freedom degrees of the denominator

ν_2 \ ν_1	21	22	23	24	25	26	27	28	29	30	40	50	60	70	80	90	100	200	500	1000
1	6216.113	6223.097	6228.685	6234.273	6239.861	6244.518	6249.174	6252.9	6257.091	6260.35	6286.427	6302.26	6312.97	6320.886	6326.474	6330.665	6333.925	6349.757	6359.536	6362.796
2	99.451	99.455	99.455	99.455	99.459	99.462	99.462	99.462	99.462	99.466	99.477	99.477	99.484	99.484	99.484	99.488	99.491	99.491	99.499	99.499
3	26.684	26.639	26.617	26.597	26.579	26.562	26.546	26.531	26.517	26.504	26.411	26.354	26.316	26.289	26.269	26.253	26.241	26.183	26.148	26.137
4	13.994	13.97	13.949	13.929	13.911	13.894	13.878	13.864	13.85	13.838	13.745	13.69	13.652	13.626	13.605	13.59	13.577	13.52	13.486	13.475
5	9.528	9.508	9.485	9.466	9.449	9.433	9.418	9.404	9.391	9.379	9.291	9.238	9.202	9.176	9.157	9.142	9.13	9.075	9.042	9.032
6	7.372	7.351	7.331	7.313	7.298	7.281	7.266	7.253	7.24	7.229	7.143	7.091	7.057	7.032	7.013	6.998	6.987	6.934	6.901	6.891
7	6.132	6.111	6.092	6.074	6.058	6.043	6.029	6.016	6.003	5.992	5.908	5.858	5.824	5.799	5.781	5.766	5.755	5.702	5.671	5.66
8	5.336	5.316	5.297	5.279	5.263	5.248	5.234	5.221	5.209	5.198	5.116	5.065	5.032	5.007	4.989	4.975	4.963	4.911	4.88	4.869
9	4.786	4.765	4.746	4.729	4.713	4.698	4.684	4.672	4.66	4.649	4.567	4.517	4.483	4.459	4.441	4.426	4.415	4.363	4.332	4.321
10	4.383	4.363	4.344	4.327	4.311	4.296	4.283	4.27	4.258	4.247	4.165	4.115	4.082	4.058	4.039	4.025	4.014	3.962	3.93	3.92
11	4.077	4.057	4.038	4.021	4.005	3.99	3.977	3.964	3.952	3.941	3.86	3.81	3.776	3.752	3.734	3.719	3.708	3.656	3.624	3.613
12	3.838	3.818	3.798	3.78	3.765	3.75	3.736	3.724	3.712	3.701	3.619	3.569	3.535	3.511	3.493	3.478	3.467	3.414	3.382	3.372
13	3.643	3.622	3.604	3.587	3.571	3.556	3.543	3.53	3.518	3.507	3.425	3.375	3.341	3.317	3.298	3.284	3.272	3.219	3.187	3.176
14	3.483	3.463	3.444	3.427	3.412	3.397	3.383	3.371	3.359	3.348	3.266	3.215	3.181	3.157	3.138	3.124	3.112	3.059	3.026	3.015
15	3.35	3.33	3.311	3.294	3.278	3.264	3.25	3.237	3.225	3.214	3.132	3.081	3.047	3.022	3.004	2.989	2.977	2.923	2.891	2.88
16	3.237	3.216	3.198	3.181	3.165	3.15	3.137	3.124	3.112	3.101	3.018	2.967	2.933	2.908	2.889	2.875	2.863	2.808	2.775	2.764
17	3.139	3.119	3.101	3.083	3.068	3.053	3.039	3.026	3.014	3.003	2.92	2.869	2.835	2.81	2.791	2.776	2.764	2.709	2.676	2.664
18	3.055	3.035	3.016	2.999	2.983	2.968	2.955	2.942	2.93	2.919	2.835	2.784	2.749	2.724	2.705	2.69	2.678	2.623	2.589	2.577
19	2.981	2.961	2.942	2.925	2.909	2.894	2.88	2.868	2.855	2.844	2.761	2.709	2.674	2.649	2.63	2.614	2.602	2.547	2.512	2.501
20	2.916	2.895	2.877	2.859	2.843	2.829	2.815	2.802	2.79	2.778	2.695	2.643	2.608	2.582	2.563	2.548	2.535	2.479	2.445	2.433
21	2.857	2.837	2.818	2.801	2.785	2.77	2.756	2.743	2.731	2.72	2.638	2.584	2.548	2.523	2.503	2.488	2.476	2.419	2.384	2.372
22	2.805	2.785	2.766	2.749	2.733	2.718	2.704	2.691	2.679	2.667	2.583	2.531	2.495	2.469	2.45	2.434	2.422	2.365	2.329	2.317
23	2.758	2.738	2.719	2.702	2.686	2.671	2.657	2.644	2.632	2.62	2.538	2.483	2.447	2.421	2.401	2.386	2.373	2.316	2.28	2.268
24	2.716	2.695	2.676	2.659	2.643	2.628	2.614	2.601	2.589	2.577	2.492	2.44	2.403	2.377	2.357	2.342	2.329	2.271	2.235	2.223
25	2.677	2.657	2.638	2.62	2.604	2.589	2.575	2.562	2.55	2.538	2.453	2.4	2.364	2.337	2.317	2.302	2.289	2.23	2.194	2.182
26	2.642	2.621	2.602	2.585	2.569	2.554	2.54	2.526	2.514	2.503	2.417	2.364	2.327	2.301	2.281	2.265	2.252	2.193	2.158	2.144
27	2.609	2.589	2.57	2.552	2.536	2.521	2.507	2.494	2.481	2.47	2.384	2.33	2.294	2.267	2.247	2.231	2.218	2.159	2.122	2.109
28	2.579	2.559	2.54	2.522	2.506	2.491	2.477	2.464	2.451	2.44	2.354	2.3	2.263	2.238	2.216	2.2	2.187	2.127	2.09	2.077
29	2.552	2.531	2.512	2.495	2.478	2.463	2.449	2.436	2.423	2.412	2.325	2.271	2.234	2.207	2.187	2.171	2.158	2.097	2.06	2.047
30	2.526	2.506	2.487	2.469	2.453	2.437	2.423	2.41	2.398	2.386	2.299	2.245	2.208	2.181	2.16	2.144	2.131	2.07	2.032	2.019
40	2.346	2.325	2.306	2.288	2.271	2.256	2.241	2.228	2.215	2.203	2.114	2.058	2.019	1.991	1.969	1.952	1.938	1.874	1.833	1.819
50	2.242	2.221	2.202	2.183	2.167	2.151	2.138	2.123	2.11	2.098	2.007	1.949	1.909	1.88	1.857	1.839	1.825	1.757	1.713	1.698
60	2.175	2.153	2.134	2.115	2.098	2.083	2.068	2.054	2.041	2.028	1.938	1.877	1.838	1.808	1.783	1.764	1.749	1.678	1.633	1.617
70	2.127	2.108	2.088	2.087	2.05	2.034	2.019	2.005	1.992	1.98	1.888	1.826	1.785	1.754	1.73	1.711	1.695	1.622	1.574	1.558
80	2.092	2.07	2.05	2.032	2.015	1.999	1.983	1.969	1.956	1.944	1.849	1.788	1.748	1.714	1.69	1.671	1.655	1.579	1.53	1.512
90	2.065	2.043	2.023	2.004	1.987	1.971	1.956	1.942	1.928	1.916	1.82	1.759	1.716	1.684	1.659	1.639	1.623	1.546	1.494	1.476
100	2.043	2.021	2.001	1.983	1.965	1.949	1.934	1.919	1.908	1.893	1.797	1.735	1.692	1.659	1.634	1.614	1.598	1.518	1.468	1.447
200	1.947	1.925	1.905	1.886	1.868	1.851	1.836	1.821	1.807	1.794	1.694	1.629	1.583	1.548	1.521	1.499	1.481	1.391	1.328	1.304
500	1.891	1.869	1.848	1.829	1.81	1.794	1.778	1.763	1.749	1.735	1.633	1.568	1.517	1.481	1.452	1.428	1.408	1.308	1.232	1.201
1000	1.872	1.85	1.829	1.81	1.791	1.774	1.758	1.743	1.729	1.716	1.613	1.544	1.495	1.458	1.428	1.404	1.383	1.278	1.195	1.159

Figure G.2: (*continuing*) Table of the Fisher's F-distribution with 99% of confidence limit.