Fault Tolerant Control System Design for Distillation Processes

Sulaiman Ayobami Lawal

Chemical Engineering, School of Engineering

Newcastle University, United Kingdom

A thesis submitted for the degree of

Doctor of Philosophy

April 2018

Abstract

The complexity and sophistication of modern control systems deployed in the refinery operation, particularly the crude distillation unit as a result of increasing demand for higher performance and improved safety, are on the increase. This growing complexity comes with some level of vulnerabilities, part of which is the potential failure in some of the components that make up the control system, such as actuators and sensors. The interplay between these components and the control system needs to have some built-in robustness in the face of actuator and sensor faults, to guarantee higher reliability and improved safety of the control system and the plant respectively, which is fundamental to the economy and operation of the system. This thesis focuses on the application of frugally designed fault tolerant control systems (FTCS) with automatic actuator and sensor faults containment capabilities on distillation processes, particularly atmospheric crude distillation unit. A simple active actuator FTCS that used backup feedback signal, switchable references and restructurable PID controllers was designed and implemented on three distillation processes with varying complexities – methanol-water separation column, the benchmark Shell heavy oil fractionator, and an interactive dynamic crude distillation unit (CDU) to accommodate actuator faults. The fault detection and diagnosis (FDD) component of the actuator FTCS used dynamic principal component analysis (DPCA), a data-based fault diagnostic technique, because of its simplicity and ability to handle large amount of correlated process measurements. The reconfigurable structure of the PID controllers was achieved using relative gain array (RGA) and dynamic RGA system interaction analysis tools for possible inputs – outputs pairing with and without the occurrence of actuator faults. The interactive dynamic simulation of CDU was developed in HYSYS and integrated with MATLAB application through which the FDD and the actuator FTCS were implemented. The proposed actuator FTCS is proved being very effective in accommodating actuator faults in cases where

there are suitable inputs – outputs pairing after occurrence of an actuator fault.

Fault tolerant inferential controller (FTIC) was also designed and implemented on a binary distillation column and an interactive atmospheric CDU to accommodate sensor faults related to the controlled variables. The FTIC used dynamic partial least squared (DPLS) and dynamic principal component regression (DPCR) based soft sensor techniques to provide redundant controlled variable estimates, which are then used in place of faulty sensor outputs in the feedback loops to accommodate sensor faults and maintain the integrity of the entire control system. Implementation issues arising from the effects of a sensor fault on the secondary variables used for soft sensor estimation were addressed and the approach was shown to be very effective in accommodating all the sensor faults investigated in the distillation units. The actuator FTCS and the FTIC were then integrated with the DPCA FDD scheme to form a complete FTCS capable of accommodating successive actuator and sensor faults in the distillation processes investigated. The simulation results demonstrated the effectiveness of the proposed approach.

Lastly, fault tolerant model predictive control (FTMPC) with restructurable inputs – outputs pairing in the presence of actuator faults based on preassessed reconfigurable control structures was proposed, and implemented on an interactive dynamic CDU. The FTMPC system used a first order plus dead time (FOPDT) model of the plant for output prediction and RGA and DRGA tools to analyse possible control structure reconfiguration. The strategy helped improve the availability and performance of control systems in the presence of actuator faults, and can ultimately help prevent avoidable potential disasters in the refinery operation with improved bottom line – Profit. Overall, the proposed approaches are shown to be effective in handling actuator and sensor faults, when there are suitable manipulated variables and redundant analytical signals that could be used to contain the effects of the faults on the system.

This thesis is dedicated to God Almighty – the giver of life and knowledge – for His Rahman before, during and beyond this work.

To my wife Sekinat and my children Kamilah, Kamran and Karimah for their care, love and support.

Acknowledgements

I am first and foremost, grateful to God for His grace and for giving me life, and the strength that saw me through this work.

I am greatly appreciative of my supervisor Dr. Jie Zhang for his counsel, support and encouragement over the last four years. You have been immensely supportive, for which I am thankful and honoured to have been under your supervision. I am grateful to Dr. Wenxian Lv and Prof. Dexian Huang of the Institute of Process Control, Department of Automation, Tsinghua University, China for providing the original Crude Distillation Unit model that I modified and used in this work. I am grateful to the Chemical Engineering, School of Engineering, Newcastle University for the Teaching Scholarship Award that lessened my financial burden at the start and during the first three years of my PhD programme. The support of members of staff and students of the Chemical Engineering, School of Engineering, Newcastle University that made this work a success is greatly appreciated.

I am highly indebted to my employer, University of Lagos, Nigeria for granting me study leave to undertake this research and for providing funding for the first three years of my PhD through the Tertiary Education Trust Fund (TETFund) scholarship award. I am immeasurably grateful for your support. I would also like to express my deep appreciation to Petroleum Technology Development Fund (PTDF) for the Scholarship award at the later stage of my research programme. The award came at the right time, and I am greatly appreciative.

Finally, the love, care, supports, prayers and perseverance shown by my family; my wife Sekinat, my daughters Kamilah and Karimah, and my son Kamran throughout the entire duration of this research are immensely acknowledged, and I will always be indebted to you. I love you all. The prayers and supports of my parents and siblings have in part seen me through the challenging period of this work. I appreciate you all.

Contents

List of Figures

List of Tables

Nomenclature

α	Significance level
β, β_p, β_b	Weighting matrices for implementing the restructurable PID controller
Δ	Sampling interval
Λ	Relative gain array
λ	Eigenvalues of the covariance matrix Σ
Λ_{Fi}	RGA used to obtain fault i back-up controller reconfiguration
Σ	Covariance matrix for the normal process data
σ_j	Standard deviation of variable j
θ_i	Principal component scores/PCR model parameters

Symbols

$\hat{P}_k, \hat{P}_{k-1}$	Model coefficient estimates at time step k and $k-1$
$\hat{X}$	Estimate of the PCA model of X
$\hat{y}$	Controlled product quality variable estimates using DPCR/DPLS
$\tilde{x}$	Predicted trajectories of a nominal system under MPC
a	Number of principal components retained in the PCA model
A, B, C, D	Known system matrices of appropriate dimensions
avg_x_{kj}	Average variable contributions to the monitoring statistics
C, V	Parameter and projection matrices under parity space approach
c_α	Normal distribution at 100(1-α)% confidence level

$cont\_x_{k_{f,j}}$	Contribution of variable x_k to score vector t_j at point f
d, r	Residual signal and unknown input vector/disturbance
E, F	Residual terms of a PCA model/PLS model
e, f	Fault vector and state estimation error/control error
F, T, K, H	Unknown observer matrices
F_1, F_2	Systems under monitoring
f_s, f_a, f_c	Sensor, actuator and component fault vectors
$F_{a,n-a,\alpha}$	F Distribution with appropriate degree of freedom and confidence limit
G_c, G_b	Controller and pre-assessed back-up controller for faults accommodation
G_u, G_{rf}	Input and fault transfer matrices
$h(x)$	Lyapunov-based controller
H_u, H_y	Transfer matrices
$J(r(t))$	Residual evaluation function
K	Process gain matrix
N	Prediction horizon
n	Number of samples in X matrix
Ny	Linear function of output y
p	Number of process variables in matrix X
P	Static/dynamic weighting matrix or model coefficient vector relating directly to the physical parameters of a system
Q_{c1}, R_{c1}	Positive definite symmetric weighting matrices of an MPC objective function
R_1, R_2	Fault entry matrices
r_b	Back-up set point signal
r_r	Switchable references

T	Principal component scores
$T(t)$	Threshold function
T^2	Hotelling's T square value
T^2_{lim}, SPE_{lim}	T^2 and SPE confidence limits
U	Principal component loading vectors
u, u_R	Known system input vector and input vector to the Actuator
u_1, u_2	Sets of possible manipulated variable for MPC
u_b	Back-up manipulated variables
u_c	Controller output
$u^*_{d2}(\tau \mid t_k), u^*_{d1}(\tau \mid t_k)$	Optimal solutions to LMPC optimization problem
V	Projection matrix or Lyapunov function
$w, \hat{w}$	Vector of model parameters in terms of PCs and its estimates
W_0	Observability matrix of (C, A_1)
$x, \hat{x}$	State variable vector and estimated state vector
X, Y	Scaled measurements matrix and response variable(s)
y, y_R	Measured system output and real system output vector
y_b	Controlled variable back-up signal
y_p, y_s	Primary controlled output and measured uncontrolled secondary variables
y_y	Restructurable controlled outputs
y_{p_est}	DPCR/DPLS estimated primary controlled variable
z	Redundant system signal/Simulator output/Full-order observer state

Acronyms/Abbreviations

AEM	Abnormal even management
AFTCS	Active fault tolerant control system

AGO	Automotive gas oil
ASTM	American society for testing and materials
AUKF	Adaptive unscented Kalman filter
CDU	Crude distillation unit
CFZ	Crude flash zone
CV	Controlled variable
DMPC	Distributed model predictive control
DPCA	Dynamic principal component analysis
DPCR	Dynamic principal component regression
DPLS	Dynamic partial least square
DRGA	Dynamic relative gain array
DV	Disturbance variables
FDD	Fault detection and diagnosis
FDM	Fuzzy-based decision making
FIC	Flow controller
FOPDT	First order plus dead time
FTC	Fault tolerant control
FTCS	Fault tolerant control system
FTIC	Fault tolerant inferential controller
FTMPC	Fault tolerant model predictive control
GLR	Generalized likelihood ratio
LIC	Level controller
LMPC	Lyapunov-based model predictive control
LNG	Liquefied natural gas

LQ	Linear quadratic
LQG	Linear quadratic gaussian
MLR	Multiple linear regression
MPC	Model predictive control
MSPCA	Multiscale principal component analysis
MV	Manipulated variables
PA	Pumparound
PC	Principal component
PCA	Principal component analysis
PCR	Principal component regression
PFTCS	Passive fault tolerant control system
PI	Proportional integral
PID	Proportional integral derivative
PLS	Partial least square
PQV	Product quality variable
RGA	Relative gain array
SOPDT	Second order plus dead time
SPE	Squared prediction error
SPRT	Sequential probability ratio testing
SS	Side stripper
TIC	Temperature controller
UIO	Unknown input observer

Chapter 1

Introduction

1.1 Background and Motivation

It is inconceivable nowadays that any facility will be built or retrofitted in the oil and gas industry without a considerable level of automation. There is an increase in the complexity and sophistication of modern control systems deployed in the industries, especially on safety-critical systems. This growing complexity comes with some level of vulnerabilities, part of which is the potential failure in some of the components that make up the control system, such as actuators and sensors. The risk is even higher in complex chemical plants like refinery with hundreds or thousands of sensors and actuators. The interplay between these components and the control system needs to have some built-in robustness to guarantee high level of safety and reliability of the plant, which is fundamental to the operation of the system. More so, meeting the economic and operational targets of the system requires its continued safe operation even in the presence of faults in the system or some of its control system components.

In spite of the successes recorded in the last four decades or so with the use of computers for conventional and advanced process control systems in our various industries, the task of responding to abnormal situations (i.e. faults) is mostly performed manually. Billions of dollars are lost in the industries every year due to low productivity, loss of operational hours, occupational injuries and illnesses resulting from major and common minor accidents occurring on a daily basis (McGraw-Hill Economics, 1985; Bureau of Labor Statistics, 1998; National Safety Council, 1999). It was reported by Nimmo (1995) that United States petrochemical industry alone incurs approximately 20 billion US dollars in annual losses, while United Kingdom records up to 27 billion US dollars

losses every year (Laser, 2000) due to poor abnormal event management (AEM). It is also interesting to know that about 70% of industrial accidents are caused by human errors (Venkatasubramanian *et al.*, 2003c). Despite advances in computer-based control applications in the industries, the fact that some of the worst chemical and nuclear power plants accidents, namely Nuclear Tsunami of March 2011 (though caused by unforeseen natural disaster) that had devastating effect on Japanese economy; Santrachs LNG plant explosion (Skikda, Algeria) on January 19, 2004 where 27 people died, and 56 were injured; Kuwait Petrochemicals Mina Al-Almedi refinery in June 2000; Occidental Petroleums Piper Alpha accident (Lees, 1996) on July 6, 1988 that resulted in the death of 162 employees of the company; Chornobyl Nuclear Power Plant on April 26, 1986; Union Carbides Bhopal, India, accident of December 3, 1984 that caused 3,800 deaths and approximately 11,000 disabilities (Jackson, 1993), just to mention a few, all happened in the last three decades or so exposes the limitations of both current redundant architectures, control and safety systems in various industries the world over. It is inevitable that some processing equipment including actuators, sensors and control systems will breakdown or malfunction at some point during their operational life span. Hence, it will be desirable to have a control system that can accommodate those potential failures during operation while still maintaining acceptable level of performance, albeit with some graceful degradation. Having smart control systems with some fault tolerant capabilities on these plants would have offered some robustness in the overall control architecture, and ultimately give sufficient time to repair the impaired systems.

The increasing availability and application of intelligent actuators and sensors with built-in diagnostic capabilities in several industries, oil and gas inclusive also supports the efforts towards the development of smart plants. The demand for development and application of smart controllers with built-in diagnostics and reconfigurable capabilities for optimal operation and management of plants during normal and abnormal situations in the process industries is therefore on the increase. These smart controllers could be referred to as fault tolerant control systems (FTCS). FTCS is an advanced control system with automatic components containment capabilities. It is necessitated by the increasing demand for higher performance, improved safety, reliability and availability of control systems in the event of malfunctions in actuators, sensors and or other system components. FTCS is also expected to provide desirable performance on complex automated facilities when process equipment, actuators, and sensors breakdown or malfunction during operation.

Fault Tolerant Control Systems has received a great deal of interest in both the industry and in the academia, but its actualization has faced some challenges in terms of its applicability in the industry. FTCS has two major components — fault detection and diagnosis (FDD) and fault tolerant controllers (FTC). FDD is a matured research area. Researches in this area span over four decades with different and diverse techniques employed (Isermann, 1984; Frank, 1990; Patton and Chen, 1992a; Patton and Chen, 1992b; Isermann, 1993; Frank, 1996; Garca and Frank, 1997; Isermann and Ball, 1997; Patton and Chen, 1997; Zhang *et al.*, 1997; Edward *et al.*, 2000; Gomm *et al.*, 2000; Blanke *et al.*, 2001; Blanke *et al.*, 2003; Venkatasubramanian *et al.*, 2003a; Venkatasubramanian *et al.*, 2003b; Venkatsubramanian *et al.*, 2003c; Isermann, 2005; Zhang, 2006a; Sangha *et al.*, 2008; Zhang and Jiang, 2008; Chilin *et al.*, 2012a; Yu *et al.*, 2014). FDD mainly detects fault, isolates (determines location and type) and estimates fault(s) magnitude(s); feeding the information to FTC, an active and evolving research area, which then reconfigures as appropriate to ensure acceptable performance in the impaired system in an online real-time manner.

There are some commercial equipment monitoring and health management packages in use in the industries, such as Profit Sensor from Honeywell, Plant Triage from Expertune and AMS from Emerson. These packages employ techniques such as Multiple Linear Regression (MLR) and Principal Components Analysis (PCA) in monitoring the process variables and health of the system components, but have no integrated fault tolerant controllers to take corrective actions when fault(s) is/are detected. The architecture and integration of FDD and FTC to form FTCS sounds pretty straightforward theoretically, but in actual fact, its actualization has faced numerous challenges as most of the developed FDD techniques are for monitoring purposes rather than control purposes. Admittedly, significant effort has been made recently in FTCS, where many algorithms and methods have been developed in different application areas (Chandler, 1984; Vander Velde, 1984; Eterno *et al.*, 1985; Stengel, 1991; Rauch, 1994; Rauch, 1995; Blanke *et al.*, 1997; Blanke *et al.*, 2000; Blanke *et al.*, 2001; Diao and Passino, 2002; Isermann *et al.*, 2002; Mehrabi *et al.*, 2002; Blanke *et al.*, 2003; Bruccoleri *et al.*, 2003; Qin and Badgwell, 2003; Zhang and Jiang, 2003; Steinberg, 2005; Blanke *et al.*, 2006; Isermann, 2006; Wang *et al.*, 2007; Zhang and Jiang, 2008; Noura *et al.*, 2009; Chilin *et al.*, 2010a; Chilin *et al.*, 2010b; Chilin *et al.*, 2012a; Chilin *et al.*, 2012b; MacGregor and Cinar, 2012; Mirzaee and Salahshoor, 2012; Lao *et al.*, 2013). However, there are still so many issues to be addressed in the application of FTCS to oil and gas processes. Some of these challenges include the ability

of the FDD component to quickly and accurately detect and diagnose different faults (actuator, sensor and component faults); the mechanism for effective integration of FDD and FTC; the suitability of the FTCS to handle non-linear systems; its robustness to noise and uncertainties and the complexity of computation required during implementation.

Hence, with the challenges listed above, this thesis contributes to the furtherance of the development and application of FTCS to the oil and gas processes, particularly the distillation processing units with special focus in its computational complexity, ease of implementation and effective FDD and FTC integration mechanism. This involves the development and application of simple restructurable feedback controllers with backup feedback signals and switchable reference points to tolerate actuator faults; fault-tolerant model predictive controllers (FTMPC) and fault-tolerant inferential controllers (FTIC) to accommodate actuator and sensor faults respectively in the binary and crude distillation units.

1.2 Aim and Objectives

The main aim of this thesis is to design Fault Tolerant Control Systems for distillation processes with capabilities to automatically accommodate failures in components such as actuators and sensors. The proposed FTCS will help to maintain reliable desirable performance and stability of a crude distillation unit after the occurrence of faults, perhaps with acceptable graceful performance degradation. It will also ensure reliable and effective control system performance pre-fault era. This will involve strategies to monitor behaviour of the crude distillation unit, including failures in sensors and actuators, and means by which sub-optimal control strategies are designed and selected as circumstances change in the system.

1.2.1 Aim

The main aim of this thesis is to develop and implement fault-tolerant controllers – simple restructurable feedback controllers with backup feedback signals and switchable reference points, fault-tolerant model predictive controllers (FTMPC) and fault-tolerant inferential controllers (FTIC) to accommodate actuator and sensor faults in crude distillation units.

1.2.2 Objectives

The objectives pursued in this work in order to achieve the main aim are;

1. Review of current relevant FDD and FTC strategies in the oil and gas industry.
2. Identify the current FDD architectures and their suitability for integration with FTC towards building robust fault tolerant control systems for distillation processes.
3. Develop dynamic and interactive test-bed models in Matlab, Simulink, and Hysys for the distillation processes.
4. Develop and implement suitable data-based fault diagnostic scheme for the processes developed in (3).
5. Develop and implement simple reconfigurable fault-tolerant controller on the processes developed in (3) to accommodate actuator faults.
6. Develop and implement fault-tolerant inferential control (FTIC) system on the processes developed in (3) to accommodate sensor faults.
7. Integrate and implement FDD, the actuator fault-tolerant controller developed in (5) and FTIC on the processes developed in (3) to accommodate sensor and actuator faults respectively.
8. Develop and implement fault-tolerant model predictive control (FTMPC) on the processes developed in (3) to tolerate actuator faults.
9. Integrate and implement FDD, FTIC, and FTMPC on the processes developed in (3) to accommodate sensor and actuator faults respectively.

1.3 Scope of Study

The work reported in this thesis is limited to the development and application of FTCS to processes in the oil and gas industry, particularly distillation processes. The review of relevant FTCS components – FDD and FTC, is undertaken and suitable actuator and sensor fault-tolerant controllers are developed and applied to binary and crude distillation processing units.

1.4 Thesis Contributions

This thesis further advances the existing body of knowledge in the development and application of fault-tolerant control systems to the oil and gas processes. The main contributions are listed below.

- Simple active restructurable feedback controllers with backup feedback signals and switchable reference points are designed and integrated with dynamic principal components analysis fault detection and diagnostic model to accommodate actuator faults in binary and crude distillation processes. To achieve this, different reconfigurable control structures are analysed a priori using relative gain array (RGA) and dynamic relative gain array (DRGA) analysis to select possible switching options for the control system as circumstances change on the plant.

- The development of fault-tolerant inferential controller (FTIC) using dynamic principal component regression (DPCR) and dynamic partial least square (DPLS) techniques for controlled variable estimations, and integration with dynamic principal component analysis (DPCA) fault monitoring diagnostic model to accommodate sensor faults on simple and complex distillation processes is achieved in this work. The FTIC system has sensor fault diagnostic model integrated with sensor fault-tolerant control technique to contain the effects of sensor fault on the system.

- The integration of the FTIC and the restructurable feedback controllers with DPCA fault diagnostic models to accommodate both actuator and sensor faults in crude distillation unit as a complete fault-tolerant control system is also achieved in this work.

- The development and implementation of fault-tolerant model predictive control (FTMPC) integrated with both FTIC and DPCA fault diagnostic model to accommodate actuator and sensor faults in a dynamic crude distillation unit is another major contributions of this thesis.

These contributions have been published in reputable international journal and presented at different conferences as highlighted below.

Journal Publication

Lawal, S. A. and J. Zhang, J. (2017), 'Actuator fault monitoring and fault tolerant control in distillation columns', *International Journal of Automation and Computing*. 14(1), 80-92.

Conference Papers

Lawal, S. A. and Zhang, J. (2015) 'Actuator fault monitoring and fault tolerant control in distillation columns', *2015 21st International Conference on Automation and Computing (ICAC)*, pp. 329 334.

Lawal, S. A. and Zhang, J. (2016) 'Sensor Fault Detection and Fault Tolerant Control of a Crude Distillation Unit', in Kravanja, Z. and Bogataj, M. (eds.) *26th European Symposium on Computer Aided Process Engineering*. Amsterdam: Elsevier Science Bv, pp. 2091 2096.

Lawal, S. A. and Zhang, J. (2016) 'Fault monitoring and fault tolerant control in distillation columns', *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 865-870.

Lawal, S. A. and Zhang, J. (2017) 'Actuator and Sensor Fault Tolerant Control of a Crude Distillation Unit', in Espuna, A., Graells, M. and Puigianer, L. (eds.) *Computer Aided Chemical Engineering*. Elsevier, pp. 1705-1710.

1.5 Thesis Organisation

The thesis is organised into seven chapters. Details contained in the remaining six chapters are outlined as follows.

Chapter 2 presents a brief review of the different components of fault tolerant control system – fault detection and diagnosis (FDD) and fault tolerant controllers (FTC). Passive and active fault-tolerant control systems are first introduced, after which FDD which is a major component of active FTCS is discussed. Relevant state of the art fault detection and diagnosis techniques are summarily reviewed to assess their suitability for the development of the FTCS for complex chemical plants. Different techniques employed under model-based and data-based fault detection and diagnosis are also outlined in this section. The different approaches that have been researched in the development of fault-tolerant controllers for complex systems including model predictive control are assessed,

and the chapter is concluded with a brief discussion on techniques used in soft sensor estimation such as DPCR and DPLS.

Chapter 3 details the design of the proposed simplified fault-tolerant control system with its different components and how they are integrated. Fault-tolerant control system for actuator faults is first presented, which also includes the rationale for using data-based technique for the actuator fault detection and diagnosis part. The control strategies and the tools employed in identifying and analysing different control loops pairing pre and post-fault era in order to achieve a seamless switching and stability in the system post-fault era are then presented. This is followed by detailing the procedures used to accomplish the design of FTIC, the sensor fault-tolerant control system component of the FTCS. It includes how the faulty controlled variable measurements are estimated using DPCR and DPLS and the eventual integration of the predicted values into the FTCS. The chapter is concluded with the presentation of the combined actuator and sensor faults FTCS.

Chapter 4 focuses on the implementation of the developed FTCS for actuator faults accommodation on distillation processes. The system is applied to three different distillation processes – a binary distillation column, the Shell heavy oil fractionator unit and a crude distillation unit. The processes are appropriately described in this section, and of particular interest is the development of an interactive dynamic crude distillation unit to allow for implementation of the FTCS on a complex system. This is presented in such a way as to demonstrate the applicability of the FTCS to plants with varying complexities.

Chapter 5 presents the implementation of the FTIC for sensor faults accommodation for the same processes described in Chapter 4. This also highlights the effectiveness and the applicability of the approach to systems with varying complexities. In the concluding part of this section, implementation of the combined sensor (FTIC) and actuator components of the FTCS on a crude distillation unit is presented.

Chapter 6 details the design of fault tolerant model predictive control (FTMPC) and its integration with data-based fault detection and diagnosis. The implementation of the proposed FTMPC on a crude distillation unit is demonstrated afterward to show its effectiveness. Changes are made to the interactive dynamic crude distillation unit described in Chapter 4 to enable the implementation of FTMPC on the unit.

Chapter 7 presents a synopsis of the achievements and contributions of the thesis in light of the set objectives in Chapter 1. Some recommendations for future works are then discussed to conclude the thesis.

Chapter 2

Review of Fault Tolerant Control Systems

2.1 Introduction

A significant number of researches has been carried out in FTCS, leading to the proliferation of a wide range of techniques in different application areas (Chandler, 1984; Vander Velde, 1984; Eterno *et al.*, 1985; Stengel, 1991; Rauch, 1994; Rauch, 1995; Blanke *et al.*, 1997; Blanke *et al.*, 2000; Noura *et al.*, 2000; Blanke *et al.*, 2001; Diao and Passino, 2002; Isermann *et al.*, 2002; Mehrabi *et al.*, 2002; Blanke *et al.*, 2003; Bruccoleri *et al.*, 2003; Qin and Badgwell, 2003; Zhang and Jiang, 2003; Steinberg, 2005; Blanke *et al.*, 2006; Isermann, 2006; Wang *et al.*, 2007; Zhang and Jiang, 2008; Noura *et al.*, 2009; Chilin *et al.*, 2010a; Chilin *et al.*, 2010b; Chilin *et al.*, 2012a; MacGregor and Cinar, 2012; Mirzaee and Salahshoor, 2012; Lao *et al.*, 2013). FTCS is broadly classified into two types — passive and active fault-tolerant control systems (PFTCS and AFTCS). The classification is functional, based on how the controllers handle faults in systems. Passive FTCS have predesigned control laws that are made insensitive to some known faults and have limited capabilities on the range and magnitude of faults that can be handled. Active FTCS, on the other hand, have built-in fault monitoring diagnostic component that can detect the occurrence of faults in real-time and relay the information to the reconfigurable controller component of the control system to act, maintaining some level of acceptable performance in the system despite the fault. Further considerations on the classes of FTCS as mentioned above and their many different components, as well as the major relevant state of the art techniques that have been applied in the field of FTCS, will be discussed in this

chapter.

An outline of the rest of this chapter is as follows. A brief discussion on passive FTCS is presented next, followed by active FTCS with its constituent components in Section 2.3. Fault detection and diagnosis as the first major component of an active FTCS with its two broad classification under model-based and data-based FDD are presented in Section 2.4. The different design approaches that have been developed under each FDD category, including state and parameter estimation approaches under model-based FDD, and principal component analysis (PCA) and projection to latent structure (PLS) approaches under data-based FDD category are sufficiently discussed. A brief review of fault tolerant controllers (FTC) as the second major component of an active FTCS and some of the state-of-the-art techniques that have been developed in the field are then briefly discussed in Section 2.5. Fault tolerant model predictive control (FTMPC), distributed model predictive control (DMPC) and fault tolerant inferential control (FTIC) are all presented under FTC. This is followed by brief discussions on dynamic principal component regression (DPCR) and dynamic partial least square (DPLS) soft sensor estimation techniques in Section 2.6. The chapter is concluded with a summary of all the major techniques discussed therein.

2.2 Passive Fault Tolerant Control Systems

Passive fault tolerant controllers are also referred to as reliable controllers. They cannot be considered as smart controllers because they usually have fixed structure and are without built-in diagnostics to detect and diagnose faults in any system. Several authors have worked on PFTCS. Liang *et al.* (2000) worked on state feedback controllers that can accommodate a predefined set of actuator faults for nonlinear systems using Hamilton-Jacobi inequality without any fault diagnostic component. Hsieh (2002) proposed a unified gain margin constraint approach to develop a reliable, guaranteed cost controller using two-stage linear quadratic (LQ) reliable control technique. Veillette *et al.* (1992) presented the design of reliable centralised and decentralised control systems that guarantee stability and H-infinity performance pre and post-fault era for sensor or actuator faults in the centralised control system, and for control channel faults in the decentralised case. The design of an algebraic Riccati equation based reliable LQ state-feedback controller that guarantees system stability and known quadratic performance bound in the presence of a selected subset of actuator faults was also proposed by Veillette (1995). Siljak (1980) and

Yang *et al.* (1998) considered reliable control system design through the use of multiple identical controllers that guarantee internal stability and H-infinity performance before, during and after the occurrence of a sensor and or an actuator fault. Yang *et al.* (2000) presented the design of reliable LQG controllers for linear systems with sensor faults, covering normal operation, partial failure and complete failure, and Yang *et al.* (2001) considered the application of reliable LQ state-feedback regulators to provide stability for discrete-time systems with actuator failures. Zhao and Jiang (1998) proposed robust pole region assignment techniques using a dynamic pre-compensator to modify the dynamic characteristics of the redundant actuator control channels and offer reliable control performance. Yang and Zhang (1995) discussed a method that guarantees closed-loop stability and an H-infinity-norm bound using multiple similar controllers based on algebraic Riccati equation approach to accommodate actuator faults. Yang *et al.* (2001) also presented procedures for designing reliable H-infinity controllers that guarantee asymptotic stability and H-infinity performance during normal operation and in the presence of faults in sensors and actuators for linear systems. All these techniques need neither FDD system nor reconfigurable controllers to function, hence have limited capabilities in handling more serious faults. More insight and a brief review of researches in PFTCS can be found in Yu and Jiang (2015).

2.3 Active Fault Tolerant Control Systems

Active FTCS is an advanced control system with automatic components containment capabilities that provides desirable performance on complex automated facilities whether faults are present or not. There are many stakeholders in AFTCS research field, especially from within the academic community which reflects its multidisciplinary nature. The improved consideration the field has received recently was necessitated by the need to achieve a higher level of reliability, maintainability and performance in situations where controlled systems can have potentially damaging effects on the personnel, plant and the environment if faults occur in its or other system components (Patton, 1997a). Modern control systems are becoming increasingly complex and control algorithms even more sophisticated. Consequently, the issues of availability, cost efficiency, reliability, operating safety and environmental protection are of major importance (Chen and Patton, 1999). Active Fault Tolerant Control System (AFTCS) aims to prevent catastrophic consequences of fault by reconfiguring the control system to maintain satisfactory oper-

ational performance even with severe faults. Control actions are generated based on the observed faulty situations to achieve the process objectives using the process information from the remaining functional sensors and manipulating the available healthy actuators (MacGregor and Cinar, 2012).

AFTCS is of significant practical importance to the oil and gas industry – the focus of this research. It offers benefits in addition to those offered by advanced control systems through possession of diagnostics features that provide accurate timely information on the occurrence of faults, such as sensor and actuator faults and the capabilities to manage such failures in the control system components thereby maintaining the integrity of not just the control system, but of the entire operation. Fault tolerant with diagnostics capabilities is considered as one of the features of intelligent systems. According to Stengel (1991) "Fault tolerant control systems, by design or implementation are intelligent systems". Astrom (1991) is of the same opinion that fault diagnostic capability is an essential ingredient property of an intelligent system. One could argue that the use of FTCS in our various industries, especially in chemical and petroleum processing industry could be the norm in the next two decades or so, to take advantage of the increasing use of smart sensors and actuators. The main motivation for the application of FTCS in the chemical and oil and gas processing industries was driven historically by its application in the aircraft flight control systems (Steinberg, 2005). However, FTC has not been widely applied to the oil and gas industry, hence the need for this research. Another motivation for this research is the possible application of the proposed FTCS to the refinery operations in Nigeria for improved safety and higher economic rewards, particularly when faults occur.

AFTCS has two major discrete components: fault detection and diagnosis (FDD) and fault tolerant controllers (FTC), and a third – controller reconfiguration and switching mechanism which handles the interplay between FDD and FTC to achieve a seamless AFTCS that meets its design objectives as shown in Figure 2.1. The effectiveness or otherwise of an appropriate FDD component of the AFTCS, which essentially detects, isolates and identifies faults will in large part determine the success or otherwise of the whole AFTCS. Also, the ease of controller reconfiguration and switching mechanism, in addition to having suitable healthy actuators and alternative measurements sources for input-output restructuring will be crucial for the fault-tolerant effort. A detailed discussion of the different FDD techniques is given in the next section. Beyond this section, FTCS is used to refer to AFTCS for simplicity.

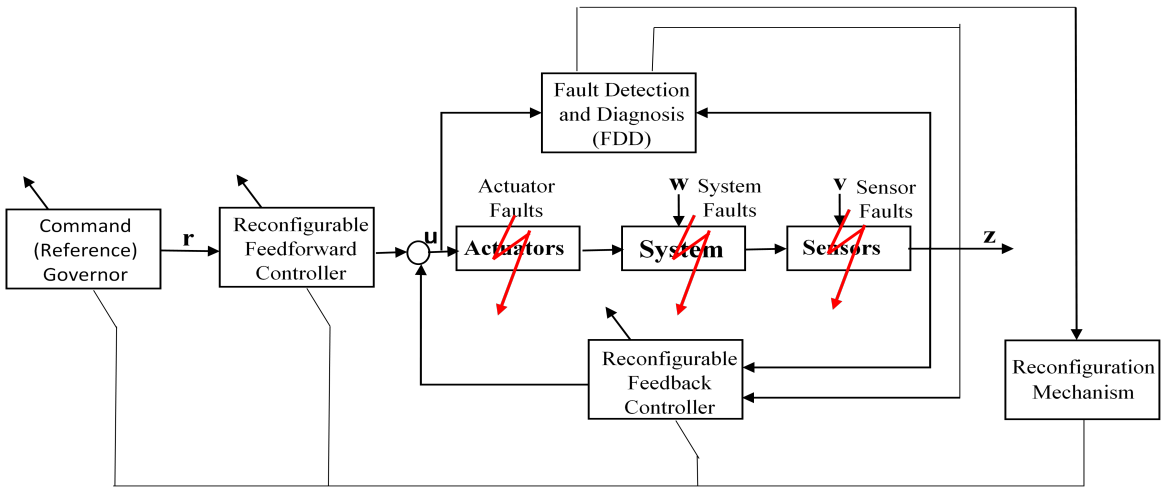


Figure 2.1: FTCS General Structure *(Source: Zhang and Jiang, 2008)*

2.4 Fault Detection and Diagnosis

Fault detection and diagnosis has been researched extensively in the last four decades or so, and several techniques have been developed. Some of the techniques are quantitative model based approaches (Willsky, 1976; Himmelblau, 1978; Isermann, 1984; Basseville, 1988; Patton, 1991; Isermann, 1993; Patton, 1993; Yu *et al.*, 1995; Frank, 1996; Patton and Chen, 1996; Patton, 1997a; Patton, 1997b; Patton and Chen, 1997; Venkatsubramanian *et al.*, 2003; Sangha *et al.*, 2008), qualitative model based approaches (Arkin and Vachtsevanos, 1990; Dvorak, 1992; Venkatasubramanian *et al.*, 2003a), data based/ process history based approaches (Zhang *et al.*, 1996; Zhang *et al.*, 1997; Gomm *et al.*, 2000; Venkatasubramanian *et al.*, 2003b; Zhang, 2006a; Yu *et al.*, 2014), and knowledge based approaches (Tzafestas, 1989). Several authors have adopted slightly varied and overlapping classifications of FDD. For example Zhang (2006a) adopted three broad classifications of FDD into model-based approaches, data analysis based approaches and knowledge-based approaches; Zhang and Jiang (2008) used two broad classifications of model-based and data-based methods with each method further classified into quantitative and qualitative methods; while Venkatsubramanian *et al.* (2003a) broadly classified FDD into three categories: quantitative model-based methods; qualitative model-based methods and process history based methods. They all almost refer to the same broad classification with slightly different nomenclatures. This work adopts the two broad classifications of Zhang and Jiang (2008) to summarise available FDD techniques, see Figure 2.2.

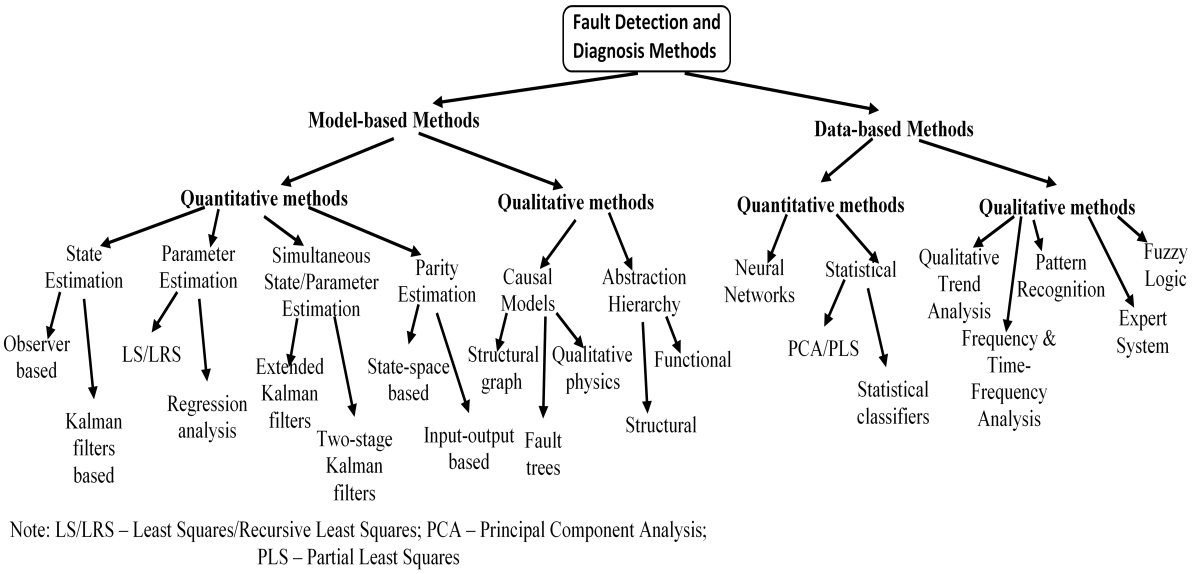


Figure 2.2: Classification of FDD techniques *(Source: Zhang and Jiang, 2008)*

FDD is a crucial component of an FTCS. Its effectiveness determines the applicability, effectiveness and overall functionality of the resulting FTCS. An FDD scheme has three main tasks: (1) *fault detection* which detects the presence of fault in a system and the time it occurs; (2) *fault isolation* that determines the kind, location and time of detection of a fault; and (3) *fault identification* which provides information on the size and time-variant of fault (Isermann and Ball, 1997; Zhang and Jiang, 2008). For clarity sake, FDD is used in this thesis to mean a combination of fault detection and isolation (FDI) plus the fault identification function (Isermann, 2006). Fault identification is the determination of type, size, location and time of detection of a fault (Isermann and Ball, 1997). There are certain minimum performance criteria a suitable FDD candidate must satisfy to fit into an overall structure of an active FTCS. Such desirable performance indices according to Zhang and Jiang (2008) are:

- Ability to handle different type of faults (actuator, sensor and system component faults)
- Ability to produce quick and accurate detection
- Isolability, which is being able to differentiate between different faults
- Identifiability
- Suitability for fault tolerant control system integration

- Identifiability for multiple faults
- Suitability for nonlinear systems
- Robustness to noise and uncertainties
- Computational complexity

The performance indices outlined above are the minimum requirements which an FDD scheme must satisfy, at least to a greater extent before one can hope for a practically applicable FTCS in the oil and gas industry. Model-based FDD is discussed next.

2.4.1 Model-based Fault Detection and Diagnosis

The traditional approach to fault diagnosis is based on hardware or physical redundancy with the application of a voting scheme. It employs multiple lanes of sensors, actuators, computers and software to measure and or control a particular variable (Chen and Patton, 1999). Imagine employing this approach in modern complex systems with hundreds, possibly thousands of variables to be measured, monitored and control. Indeed, the drawbacks of having extra equipment and the accompanying costs, additional space for installation and the costs of maintenance will be of serious concern. To overcome these problems, analytical redundancy had been developed. It mainly uses the redundant (or functional) relationships between various measured variables of the monitored system. Analytical redundancy is deemed to be potentially more reliable. It does not need additional hardware to generate residual signal. Hence, no additional hardware fault will be introduced (van Schrick, 1993). Figure 2.3 illustrates the concepts of hardware and analytical redundancy.

Patton and Chen (1997) defined model-based fault diagnosis as "the determination of faults in a system by comparing the available system measurements with a priori information represented by the system's mathematical model, through generation of residual quantities and their analysis." Faults are declared when the residuals generated as a result of the difference between the measured variables and their estimates from the mathematical models reach or exceed a set of fixed or variable thresholds on the particular residual. A set of residuals can be designed with each having a unique sensitivity to individual faults occurring in different location in the system. Fault isolation is then achieved with subsequent analysis of each residual after a threshold has been breached. Application of this

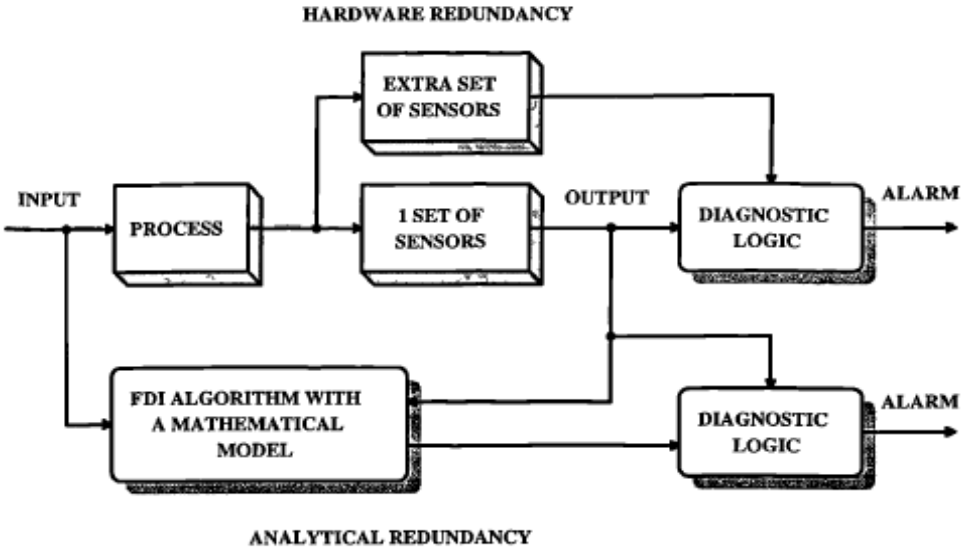


Figure 2.3: Hardware and analytical redundancy *(Source: Chen, 1995)*

Table 2.1: Features of different FDD techniques*(Source: Zhang and Jiang, 2008)*

Criteria/method	State estimation				Parameter estimation RLS & variant	Simultaneous state and parameter estimation		Parity space
	Single		Multiple			Extended	Two-stage	
	Observer	KF	Observer	KF		KF	KF	
Fault sensor	√	√	√	√	∗	√	√	√
Actuator	♠	♠	♠	√	√	√	√	♠
Type structure	♠	♠	♠	√	√	√	√	♠
Speed of detection	√	√	√	√	∗	√	√	√
Isolability	×	×	√	√	√	√	√	√
Identifiability	×	×	♠	∗	√	√	√	∗
Suitability for FTC	×	×	√	√	√	√	√	×
Multiple fault identifiability	∞	∞	√	√	√	√	√	∗
Nonlinear systems	×	×	♠	√	♠	√	√	√
Robustness	∞	∞	∗	∗	♠	♠	♠	√
Computational complexity	√	√	∗	∗	√	∗	√	√

Note: (√) favourable; (∗) less favourable; (×) not favourable; (♠) applicable; (∞) not applicable

approach hinges heavily on having a good knowledge of the process and the relationship between faults and model states or parameters.

Also, a complete and accurate mathematical model of the system is required, which is usually a constraint especially for complex chemical and petroleum processing facilities as considered in this work. Modelling the dynamic of a system becomes more difficult with increasing complexity with uncertainties in respect to the system's structure, its parameters and the effect of disturbances on the system. The primary tasks of an FDD; fault detection, isolation and diagnosis will be discussed under state estimation, parameter estimation and parity space techniques as the most frequently used model-based fault detection and isolation techniques. Table 2.1, extracted from Zhang and Jiang (2008)

summarises their features against the performance indices earlier mentioned. Another point worthy of mentioning is the issue of robustness in model based fault diagnosis. Robustness against modelling uncertainty that results from incomplete knowledge and understanding of the monitored processes is as important as the main objective for which the diagnostic scheme was designed. It has become an important research issue in recent time (Patton, 1997b; Patton and Chen, 1997; Chen and Patton, 1999). Model based fault diagnosis involves two main stages of residual generation and decision making. It was initially proposed by Chow and Willsky (1980) and is now generally accepted by the fault diagnosis community.

2.4.1.1 Residual Generation and Decision Making

I Residual Generation

It is a procedure for fault symptoms (residual signal) extraction from the system. It generates a fault indicating signal – residual, using the available input and output information from the monitored system. The generated signal should typically be zero or close to zero when no fault occurs, and it is designed such that possible occurrence of a fault is indicated from the onset. The value of the residual is significantly different from zero when fault occurs. The residual is characteristically independent of the system inputs and outputs in an ideal condition. It should typically only contain fault information to ensure reliable fault detection and isolation. The algorithm used to generate residual is referred to as residual generator (Chen, 1995; Chen and Patton, 1999). Figure 2.4 illustrates the different residual generation techniques available.

Mathematical Derivation of Residual Generator

Let z represent a redundant signal generated by the monitored system (F_1).

$$z(t) = F_1(u(t), y(t)) \tag{2.1}$$

where $u(t)$ and $y(t)$ are the input and output of the system F_1. Then, residual r is generated when z is compared with the system measured output $y(t)$.

$$r(t) = F_2(y(t), z(t)) = 0 \tag{2.2}$$

Ideally, the residual should be zero in a fault free system, but the above expression will be violated when a fault occurs. There are several other ways of generating residuals.

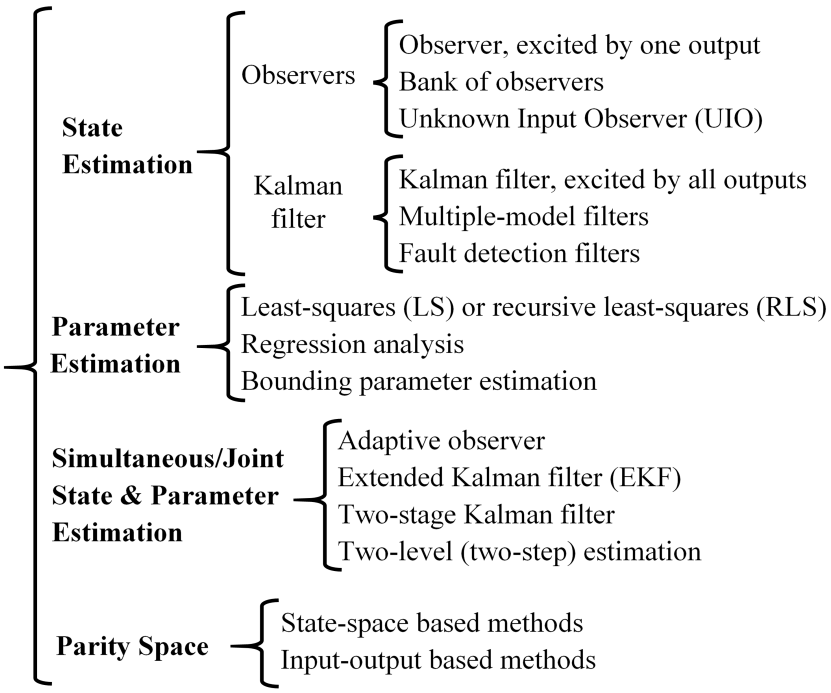


Figure 2.4: Residual generation techniques in model-based FDD *(Zhang and Jiang, 2008)*

One is by using simulator, in which only the system input will be required and the signal z is produced as the output of the simulator. This is then compared with the system monitored output y to generate the required residual. However, this system has a major drawback in that; its stability is not guaranteed when the monitored system is unstable. An output observer based residual generator could also be used. In this case, both the system input and output signals are required to generate an estimate of a linear function of output y, say Ny. The system F_2 can then be expressed as:

$$F_2(y, z) = P(z - Ny) \tag{2.3}$$

where P is a static or dynamic weighting matrix. Figure 2.5 below presents a general structure for residual generator, expressed mathematically as:

$$r(s) = \begin{bmatrix} H_u(s) & H_y(s) \end{bmatrix} \begin{bmatrix} u(s) \\ y(s) \end{bmatrix} = H_u(s)u(s) + H_y(s)y(s) \tag{2.4}$$

$$r(t) = 0 \quad if\ and\ only\ if\ f(t) = 0 \tag{2.5}$$

$$H_u(s) + H_y(s)G_u(s) = 0 \tag{2.6}$$

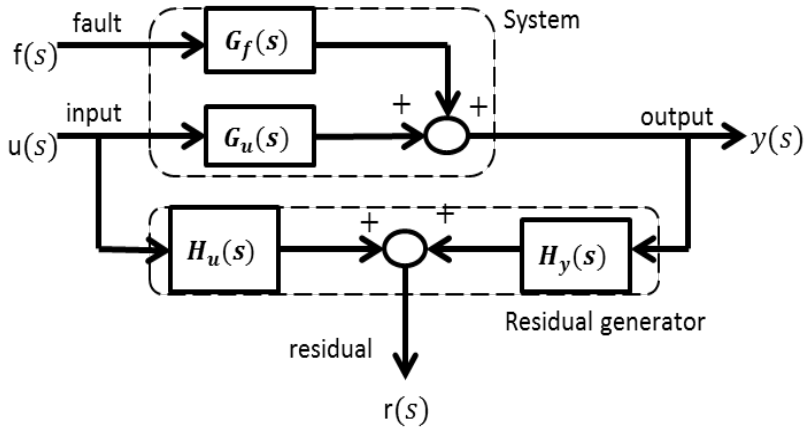


Figure 2.5: General structure of a residual generator

$H_u(s)$ and $H_y(s)$ are realizable transfer matrices through stable linear systems, and $f(t)$ is the fault. The residual is designed to be zero for a fault free system as expressed in equation 2.5 with constraint expressed in equation 2.6 satisfied. Certain desired residual performance is achieved through suitable choice of $H_u(s)$ and $H_y(s)$ which must satisfy equation 2.6. Detection of a fault is simply carried out by relating the residual evaluation function $J(r(t))$, (e.g. the norm of the residual vector) with the threshold function $T(t)$, chosen with some prior knowledge of the system. The expression is given below.

$$\begin{cases} J(r(t)) \leq T(t) & for \quad f(t) = 0 \\ J(r(t)) > T(t) & for \quad f(t) \neq 0 \end{cases} \tag{2.7}$$

The residual vector with the occurrence of faults is given as:

$$r(s) = H_y(s)G_f(s)f(s) = G_{rf}(s)f(s) = \sum_{(i=1)}^{g} [G_{rf}(s)]_i f_i(s) \tag{2.8}$$

where $G_{rf}(s) = H_y(s)G_f(s)$ is defined as the fault transfer matrix expressing the relationship between the residual and faults. $[G_{rf}(s)]_i$ (which must be non-zero for the i_{th} fault f_i to be detectable, i.e. $[G_{rf}(s)]_i \neq 0$) is the i_{th} column of the transfer matrix $G_{rf}(s)$ and $f_i(s)$ is the i_{th} component of $f(s)$ (Chen, 1995; Chen and Patton, 1999). For more reliable fault detectability, we can introduce this condition:

$$[G_{rf}(0)]_i \neq 0 \tag{2.9}$$

II Decision Making

The generated residuals are examined for likely occurrence of faults, and a decision rule is then applied to determine if any faults have occurred. This process may involve a simple threshold testing on the instantaneous values or moving average residual values, or it may consist of methods of statistical decision theory, e.g., generalized likelihood ratio (GLR) testing or sequential probability ratio testing (SPRT) (Willsky, 1976; Basseville, 1988; Tzafestas and Watanabe, 1990; Chen and Patton, 1999). The hard bit of the model-based fault diagnosis is done at the residual generation stage, which is the reason most works in this field focus on it. The decision making part is relatively easy.

2.4.1.2 Faulty System Model

Building a mathematical model of the system under investigation is the first step in model based fault diagnosis. A multiple-input multiple-output linear dynamic system is considered in this section; a model linearized around an operating point will be used for non-linear system. For the purpose of modelling a faulty system, an open-loop system is considered, which can be separated into three parts: actuators, sensors and system dynamics. Figure 2.6 presents the open loop system dynamics with actuator, sensor and component faults under consideration (Chen and Patton, 1999).

The state space model of the system dynamics block in Figure 2.6 without the fault component is given as:

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu_R(t) \\ y_R(t) &= Cx(t) + Du_R(t) \end{cases} \tag{2.10}$$

where $x \in \Re^n$ is the state vector, $u_R \in \Re^r$ is the input vector to the system, $y_R \in \Re^m$ is the real system output vector, $f_s \in \Re^m$ is the sensor fault vector and $f_a \in \Re^r$ is the actuator fault vector; A, B, C and D are the known system matrices with appropriate dimensions. Including component fault in equation 2.10 above results in:

$$\dot{x}(t) = Ax(t) + Bu_R(t) + f_c(t) \tag{2.11}$$

The component fault affects the dynamics of the original system and needs to be captured in the model. When such fault is represented as a change in the system parameter, such as a change in the i_{th} row and j_{th} column element of matrix A, then we have:

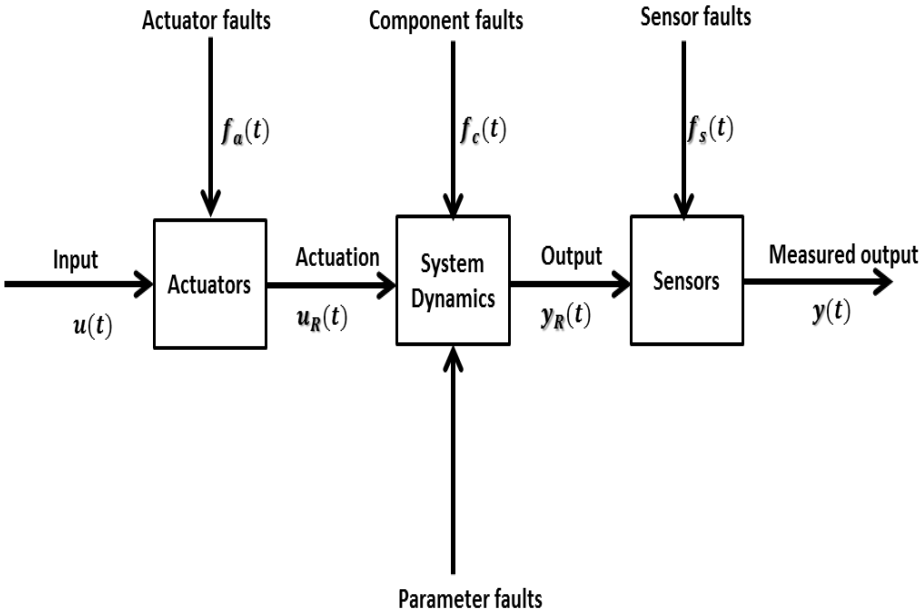


Figure 2.6: Open loop system dynamics

$$\dot{x}(t) = Ax(t) + Bu_R(t) + I_i \Delta a_{ij} x_j(t) \tag{2.12}$$

Here, $x_j(t)$ is the j_{th} element of vector $x(t)$ and I_i is an all zero n-dimensional vector except the i_{th} element being 1. The output of the system is described below, with the sensor and actuator dynamics ignored.

$$y(t) = y_R(t) + f_s(t) \tag{2.13}$$

$$u_R(t) = u(t) + f_a(t) \tag{2.14}$$

$$u(t) = u_R(t) + f_{is}(t) \tag{2.15}$$

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) + Bf_a(t) + f_c(t) \\ y(t) &= Cx(t) + Du(t) + Df_a(t) + f_s(t) \end{cases} \tag{2.16}$$

A correct choice of the sensor and actuator fault vectors as presented in equations 2.13 and 2.14 can describe all sensor and actuator fault situations. Equation 2.15 describes a system with an unknown input, for instance, an uncontrolled system. Instead, an input sensor is used to measure the input to the actuator.

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) + R_1 f(t) \\ y(t) &= Cx(t) + Du(t) + R_2 f(t) \end{cases} \tag{2.17}$$

$$y(s) = G_u(s)u(s) + G_f(s)f(s) \tag{2.18}$$

$$\begin{cases} G_u(s) &= C(sI - A)^{-1}B + D \\ G_f(s) &= C(sI - A)^{-1}R_1 + R_2 \end{cases} \tag{2.19}$$

Equation 2.16 presents a system with all possible actuator, component and sensor faults while 2.17 is the compact state space model of a system with all possible faults where $f(t) \in \Re^g$ is a fault vector with each $f_i(t)$ $(i = 1, 2, 3, ..., g)$ corresponding to a specific fault. The matrices R_1 and R_2 are the fault entry matrices and they represent the effect of faults on the system. $u(t)$ and $y(t)$ are both known for fault detection and diagnosis purpose. They are the input vector to the system (measured actuation) and the measured output vector respectively. Equations 2.17 and 2.18 represent general model representation for faulty system in time domain and frequency domain respectively. The faulty system representations are widely accepted in the fault diagnosis literature (Frank, 1990; Gertler, 1991; Frank, 1992; Patton and Chen, 1992b; Frank, 1994; Gertler and Kunwer, 1995; Patton and Chen, 1997; Chen and Patton, 1999).

2.4.1.3 State Estimation Approach

State estimation is one of the several approaches employed in the residual generation for fault detection and diagnosis purposes (Patton *et al.*, 1989; Frank, 1990; Patton and Chen, 1992a; Gertler and Kunwer, 1995). Output observer based residual generation is the most commonly used approach, and it is discussed here as a representative of the state estimation technique for residual generation.

I Observer based residual generation

Our interest here is outputs estimation using an observer to generate residual vectors. Output estimates are sufficient for this purpose, so the use of full state observer is not required. The approach estimates the outputs of the system through the measurements, or a subset of it using either Luenberger observer(s) in a deterministic setting or Kalman filter(s) in a stochastic setting. Then, the residual is the weighted output estimation error (or innovations in the stochastic case) (Patton *et al.*, 1989; Frank, 1990; Patton

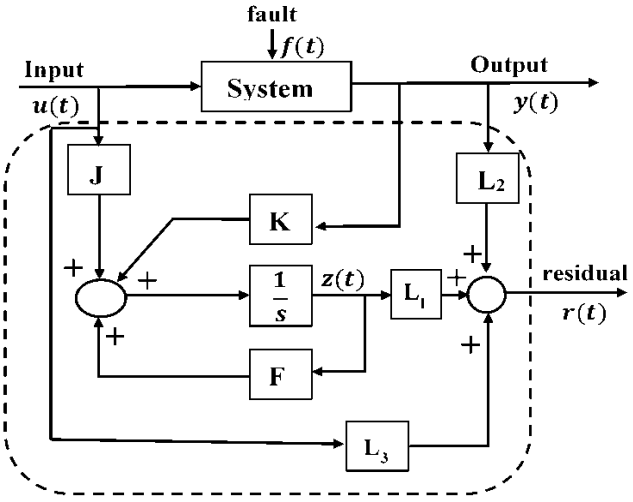


Figure 2.7: Generalized Luenberger observer residual generator *(Chen, 1995)*

and Chen, 1992a; Frank, 1994; Chen, 1995). Given below is a residual generator via generalised Luenberger observer, see Figure 2.7:

$$\begin{cases} \dot{z}(t) = Fz(t) + Ky(t) + Ju(t) \\ r(t) = L_1 z(t) + L_2 y(t) + L_3 u(t) \end{cases} \tag{2.20}$$

The matrices in the equation 2.20 above should satisfy:

$$\begin{cases} F \ has \ stable \ eigenvalues \\ TA - FT = KC \\ J = TB - KD \\ L_1 T + L_2 C = 0 \\ L_3 + L_2 D = 0 \end{cases} \tag{2.21}$$

When the residual generator, equation (2.20) is applied to the system, equation (2.17), the residual is:

$$\begin{cases} \dot{e}(t) = Fe(t) - TR_1 f(t) + KR_2 f(t) \\ r(t) = L_1 e(t) + L_2 R_2 f(t) \end{cases} \tag{2.22}$$

where $e(t) = z(t) - Tx(t)$. It is obvious from the above expressions that the residual depends completely on faults. The other option is to use full order observer with $T = I$.

A single residual is sufficient to detect fault, but a set of residual vectors (structured residual set) or directional residual vector will be required to isolate faults with the ob-

server based approach. The design of a structured residual set for sensor faults is pretty straightforward. For instance, if the output vector $y = (y_1, ..., y_m)$ is replaced with an output vector y $(y_1, ..., y_{(i-1)}, y_{(i+1)}, ..., y_m)$ without the single sensor measurement y_i, the residual will be insensitive to the fault in the i_{th} sensor. However, for isolating an actuator fault, the design of a structured residual set is not as straightforward and can be achieved through the use of unknown input observers (Viswanadham and Srichander, 1987; Frank, 1990) or eigenstructure assignment (Patton *et al.*, 1989; Chen, 1995). A fixed residual vector can be designed through fault detection filter invented by Beard (1971).

II Unknown input observer

Mathematical description of any system under consideration is at the heart of model-based fault detection and diagnosis. The more accurately the model represents the system, the better the reliability and performance of the corresponding fault diagnostic scheme. Modelling errors and disturbances are inevitable in such mathematical representation. Hence, there is need to develop robust residual generator. Robust residual generation is the most significant task in model-based fault diagnosis techniques, and unknown input observer (UIO) belongs to such class of robust residual generator. It works on the principle of decoupling the state estimation error from the unknown inputs (disturbances). By so doing, the residual can also get de-coupled from each disturbance; the residual is defined as a weighted output estimation error (Chen, 1995; Chen and Patton, 1999). Though the unknown input vector is unknown, its distribution matrix is assumed known. The approach was originally proposed by Watanabe and Himmelblau (1982), and the design problem of UIO dated back to 1975 (Wang *et al.*, 1975). Consider a dynamic system in which its uncertainty can be summarised as an additive unknown disturbance:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Ed(t) \\ y(t) = Cx(t) \end{cases} \tag{2.23}$$

Given the structure of a full-order observer described as:

$$\begin{cases} \dot{z}(t) = Fz(t) + TBu(t) + Ky(t) \\ \hat{x}(t) = z(t) + Hy(t) \end{cases} \tag{2.24}$$

where $x(t) \in \Re^n$ is the state vector, $y(t) \in \Re^m$ is the output vector, $u(t) \in \Re^r$ is the known input vector, $d(t) \in \Re^q$ is the unknown input (disturbance) vector, $\hat{x} \in \Re^n$ is the estimated state vector, $z \in \Re^n$ is the state of the full-order observer, A, B, C, E are known matrices

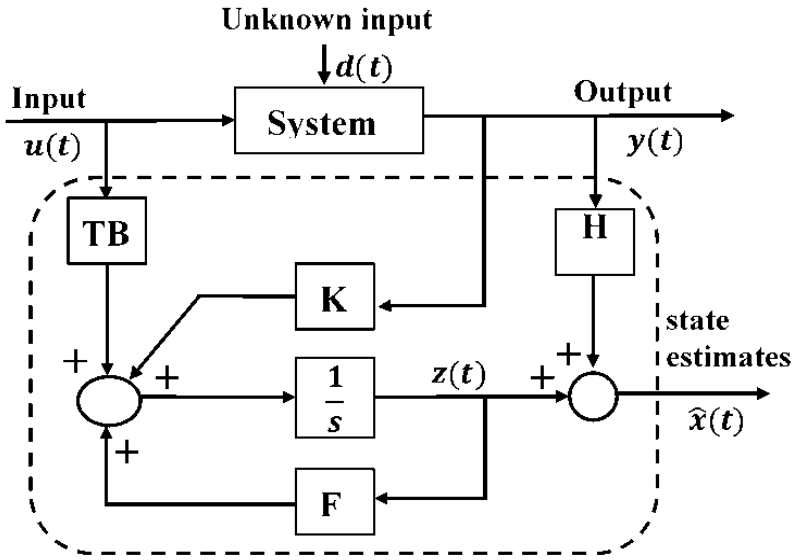


Figure 2.8: The structure of a full-order unknown input observer

with appropriate dimensions and F, T, K, H are matrices to be designed to achieve unknown input de-coupling and other design requirements. The observer described by equation 2.24 is shown in Figure 2.8.

When the observer (2.24) is applied to the system (2.23), the state estimation error $(e(t) = x(t) - \hat{x}(t))$ is governed by the expression

$$\dot{e}(t) = (A - HCA - K_1C)e(t) + [F - (A - HCA - K_1C)]z(t) + [K_2 - (A - HCA - K_1C)H]y(t) + [T - (I - HC)]Bu(t) + (HC - I)Ed(t) \quad (2.25)$$

where

$$K = K_1 + K_2 \quad (2.26)$$

Then, the state estimation error will be:

$$\dot{e}(t) = Fe(t) \quad (2.27)$$

If the following relations hold true:

$$\begin{cases} (HC - I)E = 0 \\ T = I - HC \\ F = A - HCA - K_1C \\ K_2 = FH \end{cases} \tag{2.28}$$

If all eigenvalues of F are negative, $e(t)$ will approach zero asymptotically, meaning $\hat{x} \to x$. It means that the observer (2.24) is an unknown input observer of the system (2.23). Hence, the design of UIO is to solve equations 2.26 and 2.28 and to make sure that all eigenvalues of the system matrix F are stable (Chen, 1995; Chen and Patton, 1999). Table 2.2 summarises the procedure and the necessary conditions required for the design of an unknown input observer.

2.4.1.4 Parameter Estimation Approach

Parameter estimation is one of the techniques employed in model-based fault diagnosis. Fault detection and diagnosis can be achieved through system identification techniques which presume that faults are reflected in the physical system parameters such as friction, mass, viscosity, resistance, capacitance, inductance, and so on (Isermann, 1984; Isermann, 1987; Isermann and Freyermuth, 1991a; Isermann and Freyermuth, 1991b; Isermann, 1997). Again, for this approach like many other an accurate model of the system is required. It uses the input-output model of the system. Since the goal is not only to detect process faults but also to investigate process faults, the process models should express as closely as possible the physical law which governs the system behaviour. Hence, the process models have to be first of all developed by theoretical modelling, which means stating the energy, mass and momentum balance equations, the phenomenological laws for any irreversible phenomena and the physical-chemical state equations of the system (Isermann, 1984). The models then appear in the continuous time domain, in partial or ordinary differential equations form.

The simple idea of the detection approach is that the parameters of the actual system are continuously estimated online using well-known parameter estimation method such as least squares. The results are then compared with the parameters of the reference fault free model conditions. Consider the system representation below:

$$y(t) = f(P, u(t)) \tag{2.29}$$

Table 2.2: Unknown input observer design procedure

Step 1 Check the rank condition for E and CE: If the $rank(CE) \neq rank(E)$, a UIO does not exist, go to step 10.

Step 2 Compute H, T and A_1:
$H = E[(CE)^T CE]^{-1}(CE)^T; \qquad T = I - HC; \qquad A_1 = TA$

Step 3 Check the observability: If (C, A_1) observable, a UIO exist and K_1 can be computed using pole placement, then go to step 9.

Step 4 Construct a transformation matrix P for the observable canonical decomposition: To select independent $n_1 = rank(W_0)(W_0$ is the observability matrix of $(C, A_1))$ row vector $p_1^T, ..., p_(n_1)^T$ from W_0, together with other $n - n_1$ row vector $p_{n_1+1}^T, ..., p_n^T$ to construct a non-singular matrix as:
$P = [p_1, ..., p_{n_0}; p_{n_0+1}, ..., p_n]^T$

Step 5 Perform an observable canonical decomposition on (C, A_1):
$$PA_1P^{-1} = \begin{bmatrix} A_{11} & 0 \\ A_{12} & A_{22} \end{bmatrix} \qquad CP^{-1} = [C^* \quad 0]$$

Step 6 Check the detectability of (C, A_1): If any one of the eigenvalues of A_{22} is unstable, a UIO does not exist and go to step 10.

Step 7 Select n_1 desirable eigenvalues and assign them to $A_{11} - K_p^1 C^*$ using pole placement.

Step 8 Compute $K_1 = P^{-1}K_p = P^{-1}[(K_p^1)^T \quad (K_p^2)^T]^T$, where K_p^2 can be any $(n - n_1) \times m$ matrix.

Step 9 Compute F and K: $\qquad F = A_1 - K_1 C, \qquad K = K_1 + K_2 = K_1 + FH.$

Step 10 STOP

where P is the model coefficient vector which is directly related to the physical parameters of the system. The function $f(.,.)$ can take either linear or nonlinear format. The basic procedure in using parameter estimation for fault detection and diagnosis is:

- Build the system model using physical relations.
- Determine the relationship between model coefficients and process physical parameters.
- Estimate the normal model coefficients.
- Calculate the normal process physical parameters.
- Then, determine the parameter changes which occur for the various fault scenarios.

A database of faults symptoms can be built through the execution of the last step for known faults. Coefficients of the system model are periodically identified during operation from the measurable inputs and outputs, which are then compared with the normal and faulty model parameters. To generate residuals using this method, an online parameter identification algorithm should be used. If the estimation of the model is obtained at time step $k-1$ as $\hat{P}_{(k-1)}$, the residual can be defined in either of the following ways (Chen, 1995; Chen and Patton, 1999):

$$\begin{cases} r(k) = \hat{P}_k - P_0 \\ r(k) = y(k) - f(\hat{P}_{(k-1)}, u(k)) \end{cases} \tag{2.30}$$

where P_0 is the normal model coefficient.
It is not easy to achieve fault isolation using parameter estimation approach. Simply because the parameters being identified are model parameters which cannot always be converted back to the system physical parameters (Isermann, 1984). However, the faults are expressed as the variations in physical parameters.

2.4.1.5 Parity Space Approach

Desai and Ray (1984), Chow and Willsky (1980), Lou *et al.* (1986) and Frank (1990) were the early contributors to the parity relation approach in fault detection and investigation. The essence is to check the consistency (parity) of the mathematical relations of the system by using the sensor measurements and the known process inputs. Parity relations are usually transformed variants of the input-output or state space models of the system

(Gertler and Singer, 1990; Gertler, 1991). Ideally, during steady state operation the value of the parity equations should be zero, but in reality, the residuals are non-zero due to model uncertainties, faults in the system, errors in sensors and actuators and model inaccuracies. So, the idea of this approach is to rearrange the structure of the model to get the best fault isolation. Willsky (1976) first introduced the idea of dynamic parity relation which was further developed by other authors (Gertler *et al.*, 1990; Gertler and Singer, 1990). Redundancy primarily offers freedom to achieve further fault isolation in the design of residual generation. Fault isolation requires the ability to generate residual vectors which are orthogonal to each other for different faults.

The idea of parity space method can be explained as follows (Desai and Ray, 1984; Frank, 1990). Given the expressions below:

$$y(t) = Cx(t) \tag{2.31}$$

$$y(t) = Cx(t) + \Delta y(t) \tag{2.32}$$

where $x(t) \in \Re^n$ is the true values of the state variable; $y(t) \in \Re^m$ is the measurement vector and $C \in \Re^{m \times n}$ is the parameter matrix. Redundancy exists if $m > n$. Equation 2.32 represents a faulty condition while 2.31 represents a fault free condition. Then, choose the projection matrix $V \in \Re^{(m-n) \times m}$ to satisfy:

$$VC = 0 \tag{2.33}$$

$$V^T V = I_m - C(C^T C)^{-1} C^T \tag{2.34}$$

The rows of V are required to be orthogonal, being a null space of C, i.e. $V^T V = I_{m-n}$. Combining the observation y into a parity vector p yields:

$$p(t) = Vy(t) = VCx(t) + V\Delta y(t) = V\Delta y(t) \tag{2.35}$$

$p(t) = Vy(t)$ is the parity equation set whose residuals carry the signature of the measurement faults. In the fault free case, $p = 0$. For a single i^{th} sensor fault: $\Delta y = [0\ 0\ 0\ .\ .\ \Delta y_i\ .\ .\ 0]^T$

$$V\Delta y = \Delta y_i \times (i^{th}\ column\ of\ V) \tag{2.36}$$

Thus the column V determines the m distinct directions associated with those m sensor faults, which enables the distinction of the m fault signature and hence their isolability. The above procedure assumes direct redundancy. Due to Chow and Willsky (1980), the following procedure provides a general scheme for both direct and temporal redundancy. Consider the standard discrete state space model below where A, B, C and D are parameter matrices of appropriate dimensions and $x(t)$ denotes the n dimensional state vector.

$$\begin{cases} x(t+1) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{2.37}$$

$$y(t+1) = CAx(t) + CBu(t) + Du(t+1) \tag{2.38}$$

The above equation (2.38) is the output at $t+1$, and for $s > 0$, $y(t+s)$ takes the form:

$$y(t+s) = CA^s x(t) + CA^{s-1}Bu(t) + \ldots + CBu(t+s-1) + Du(t+s) \tag{2.39}$$

Collecting the equations for $s = 0, 1, \ldots, m_1 \leq m$ and writing it in a compact form yields:

$$Y(t) = Qx(t-m_1) + RU(t) \tag{2.40}$$

Pre-multiply the above expression with a vector w^T of appropriate dimension yields a scalar equation:

$$w^T Y(t) = w^T Qx(t-m_1) + w^T RU(t) \tag{2.41}$$

The above expression will contain input, output and unknown state variables. It will qualify as parity equation only if the state variables disappear which requires:

$$w^T Q = 0 \tag{2.42}$$

This is a set of homogeneous linear equations, and if the system is observable, these n equations are independent. It has been shown that once the design objectives are selected, parity equation and observer-based designs lead to identical or equivalent residual generators (Gertler, 1991).

2.4.2 Data-based Fault Detection and Diagnosis

The difficulties faced in developing detailed first principle models for complex chemical processes with acceptable level of accuracy needed for fault monitoring and accommodation purposes limit the application of model-based FDD to well-understood systems like electro-mechanical systems. Data-based FDD, on the other hand, has been extensively used in the chemical industries for process monitoring and fault diagnosis because of its ability to provide reduced dimensional models for high dimensional processes. Its extensive usage also stems from its simplicity and ability to handle large amount of correlated process measurements. A large amount of process data collected from a system under normal and faulty conditions is required for the data-based FDD techniques. Using the classification of Zhang and Jiang (2008), data-based FDD is further classified into quantitative and qualitative methods.

The quantitative data based approaches extract features from the available process data through multivariate statistical and non-statistical means. Neural networks FDD approach is an example of the non-statistical method while principal component analysis (PCA), statistical pattern classifiers and partial least squares (PLS) are examples of the multivariate statistical methods. The qualitative data based FDD approaches, such as expert systems, fuzzy logic, pattern recognition, qualitative trend analysis and frequency and time frequency analysis, as presented in Figure 2.2 will not be discussed further in this thesis as our focus is on model-based and data-based FDD methods. Multivariate statistical approaches are powerful tools that are capable of compressing data to reduce its dimensionality and still retain as much variation as contained in the original data set for more straightforward analysis. The multivariate statistical techniques can efficiently handle noise and correlation in the original data during transformation into a much lower dimension.

2.4.2.1 Principal Component Analysis

PCA is a standard multivariate statistical technique that has been used for various analyses stretching over a century. It was originally proposed by Pearson (1901) and later developed by Hotelling (1947). Principal component analysis is based on orthogonal decomposition of the covariance matrix of the process variables along direction that explains the maximum variation of the data. Its main function is finding factors that have a much lower dimension than the original data set which accurately describes the major trend in

the original data set.

Let p denote the number of measured process variables; X be a $n \times p$ matrix of the scaled measurements of n samples and p variables with covariance matrix Σ. From matrix algebra, Σ may be reduced to a diagonal matrix L by a particular orthonormal $p \times p$ matrix U, i.e.,

$$\Sigma = ULU^T \tag{2.43}$$

where columns of U are the principal component loading vectors and the diagonal elements of L are the ordered eigenvalues of Σ which defines the amount of variance explained by the corresponding eigenvector. Then, the principal component transformation is given as:

$$T = XU \quad or \quad t_i = Xu_i \tag{2.44}$$

where t_i and u_i are the i^{th} column of T and U respectively. Equivalently, X can be decomposed by PCA as:

$$\hat{X} = TU^T = \sum_{i=1}^{p} t_i u_i' \tag{2.45}$$

The $n \times p$ matrix $T = (t_1, t_2, \ldots, t_p)$ contains the so-called principal component (PC) scores which are linear combinations of all the p variables. Typically, the first "a" principal components ($a < p$) will capture the most variation in the original data if they are correlated and can be used to represent the majority of data variation. There are different criteria available for the selection of number of principal components "a". In this work however, we select "a" which account for between 75% and 90% variation in the original data set and examine the suitability of different values of "a" for the FDD purpose using appropriate data sets.

$$\hat{X} = t_1 u_1' + t_2 u_2' + \ldots + t_a u_a' + E = \sum_{i=1}^{a} t_i u_i' + E \tag{2.46}$$

where E and $\hat{X}$ are the residual terms and the PCA model of X respectively. With an in-control model established based on historical data collected during normal operation, process monitoring is achieved by using the Hotellings T^2 and squared prediction error (SPE) monitoring statistics of the nominal model given below to detect fault from new measurements.

$$T_i^2 = \sum_{j=1}^{a} \frac{t_{i,j}^2}{\lambda_j} \tag{2.47}$$

where T_i^2 is the Hotelling's T^2 value for sample i, $t_{i,j}$ is the i^{th} element of principal component j, λ_j is the eigenvalue corresponding to principal component j and a is the number of principal components retained. SPE is simply the sum of squares of the difference between the original scaled data and their estimates ($\hat{X}$) from the PCA model. When the process is in normal operation, both SPE and T^2 monitoring statistics should be small and within their control limits. However, when a fault appears in the monitored process, the fault will cause some variables to have larger than normal magnitudes (large T^2 value) and change the variable correlations leading to large SPE values. The T^2 index indicates nonconformity with the expected behaviour of the process as captured by the diagnostic model while the SPE index presents deviations that result from events not described in the diagnostic model (MacGregor and Cinar, 2012). The fault then causes the monitoring statistics to violate their respective limits (thresholds) for some specified periods, before a fault is eventually declared. The control limits for SPE and T^2 are given by (2.48) and (2.49) respectively.

$$\begin{cases} SPE_{lim} = \theta_1 \left[\frac{c_\alpha h_0 \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right]^{\frac{1}{h_0}} \\ \theta_i = \sum_{j=a+1}^{p} \lambda_j^i \\ h_0 = 1 - \frac{2\theta_1 \theta_2}{3\theta_2} \end{cases} \tag{2.48}$$

$$T_{lim}^2 = \frac{a(n-1)}{(n-a)} F_{a,n-a,\alpha} \tag{2.49}$$

In (2.48) and (2.49) above, c_α is the value for normal distribution at $100(1-\alpha)\%$ confidence level and $F_{a,n-a,\alpha}$ is the F distribution with appropriate degrees of freedom and confidence level. Upon declaration of a fault, variable contribution plots are obtained for the SPE and the Hotelling's T^2 for further fault diagnosis to identify the component that has developed fault. This is done with some good understanding of the monitored process. The Hotelling's T^2 variable contribution plot can be obtained using (2.50) below (Kourti and MacGregor, 1996).

$$cont\_x_{k_{f,j}} = \frac{t_{f,j}}{\lambda_j} \, u_{k,j} \, x_{f,k} \tag{2.50}$$

where $cont\_x_{k_{f,j}}$ is the contribution of variable x_k to score vector t_j at point f (point of fault declaration); $t_{f,j}$, λ_j and $u_{k,j}$ are the score vector t_j, the corresponding eigenvalue and loading vector for k^{th} variable respectively at the faulty sample f, while $x_{f,k}$ is variable x_k also at point f. The SPE contribution plots can be easily obtained by taking the contributions of each variable to the large SPE value at the point of fault declaration.

2.4.2.2 Projection to Latent Structure

Projection to latent structure (PLS) originated from the pioneering work of Wold (1982) between the mid-1960s and early 1980s and was further developed by Wold and co-workers (Wold *et al.*, 1984a; Wold *et al.*, 1984b; Wold *et al.*, 1987). PLS, similar to PCA conceptually reduces the dimension of correlated process data by projecting them down onto a lower dimensional latent variable space. PLS however, works with additional data matrix Y, process quality variables together with the process variable X. PLS models the relationship between the two sets of data while compressing them simultaneously. It extracts the latent variables that explain the variation in process data X, at the same time the variation in X that is most predictive of the quality data Y. The first PLS latent variable is the linear combination of the process variables that maximises the covariance between them and the quality variable (Venkatasubramanian *et al.*, 2003b). PLS defines the high dimensional process variables (regressor) and process quality variables (response) (X and Y) in terms of a small number of latent variables (T) that defines the major directions of variation in the process data (MacGregor and Cinar, 2012). The basic model is defined as:

$$X = TU^T + E \tag{2.51}$$

$$Y = TC^T + F \tag{2.52}$$

where X and Y are $(n \times p)$ and $(n \times m)$ matrices of observed values, $T = XW^*$ is a $(n \times a)$ matrix of latent variable scores ($a \ll p$), U, C and W^* are matrices of loading estimated from the data, n is the number of observations, and p and m are the numbers of regressor and response variables respectively. The concept of dependent and independent variables has little place in latent variable model. E and F are errors associated with X and Y respectively. The choice of process variables and process quality variables are user defined (MacGregor and Cinar, 2012).

There are other variants of PCA and PLS techniques that have been used for faults investigation over the years. Nonlinear PCA had been used to handle system nonlin-

earity (Dong and McAvoy, 1996; Zhang *et al.*, 1997); incorporation of multivariate SPC with neural networks (Hoskins *et al.*, 1991; Nomikos and Macgregor, 1994); application of multiway PCA for batch processes (Nomikos and Macgregor, 1994); neural net PLS incorporated with feedforward networks for PLS modelling (Qin and Mcavoy, 1992); application of multiblock PLS (MacGregor *et al.*, 1994); integration of PCA with discriminant analysis techniques (Raich and Cinar, 1996); the use of recursive PCA (Li *et al.*, 2000); recursive PLS for adaptive modelling (Qin, 1998); the use of multiscale PCA (MSPCA) approach which integrate PCA and wavelet analysis (Bakshi, 1998); and combination of model based approach with multivariate statistical method (Gertler and McAvoy, 1998).

Multivariate statistical approaches are powerful tools capable of handling high dimensional process variables that have good correlation to reveal the presence of abnormalities in systems. From industrial successful application point of view, multivariate statistical process monitoring techniques are the most widely used techniques for fault diagnostics owing to their fast abnormal events detection, ease of implementation and little effort required for their modelling with very little a priori process knowledge. However, they do not possess 'fingerprint' or 'signature' properties for diagnosis due to their limited process knowledge.

2.4.2.3 Dynamic PCA

Dynamic PCA is a variant of PCA technique that incorporates time-lagged measurements in its model to capture the dynamic correlation behaviour of the system for effective fault propagation analysis. The technique is the same as PCA with the only difference being the increased dimension of the process variable p by a factor of l (the number of time lags considered) to give $(l+1)p$ process variables. In essence, this leads to an increase in the columns of X to a new dimension $n \times (l+1)p$ resulting in orthonormal matrix U in (2.43) having dimension $(l+1)p \times (l+1)p$. Consider an $(n \times p)$ process variable matrix X, the augmented matrix X for DPCA at any time instant t will be:

$$X = [X \quad X(t-1) \quad \ldots \quad X(t-l)] \tag{2.53}$$

If for instance, $p = 3$ and $l = 1$, we have

$$X(t) = [x_1(t) \quad x_2(t) \quad x_3(t) \quad x_1(t-1) \quad x_2(t-1) \quad x_3(t-1)] \tag{2.54}$$

where $x_i(t)$ and $x_i(t-1)$ are the process variables at time t and at $l = 1$. The procedure for the determination of the number of time lag (l) can be found in Ku *et al.* (1995).

2.4.2.4 Neural Networks

There are a number of papers on neural networks applications in process fault investigation (Venkatasubramanian and Chan, 1989; Watanabe *et al.*, 1989; Ungar *et al.*, 1990; Venkatasubramanian *et al.*, 1990; Hoskins *et al.*, 1991; Zhang and Roberts, 1991; Gomm *et al.*, 2000; Zhang, 2006a; Sangha *et al.*, 2008; Yu *et al.*, 2014). For this approach, the knowledge of training data set that covers normal operations, faults and symptoms is required. Through training the relationship between faults and their symptoms can be discovered and stored as network weights. The trained network is then used for fault diagnosis and abnormalities are matched with their corresponding faults. Neural networks are attractive techniques for fault diagnosis as they are easy to develop and also due to their ability to handle system nonlinearities. However, multiple neural networks may need to be used for improved diagnostic performance as single neural network can lack robustness, especially when there are insufficient data (Jacobs *et al.*, 1991; Jordan and Jacobs, 1994; Zhang *et al.*, 1997; Zhang, 2006a). Reliable and robust trained network is essential for reliable fault diagnosis. The reliability of a neural network depends on some factors, such as the network topology, the training algorithm used, and the size of available training data (Zhang, 2006a).

2.5 Fault Tolerant Controllers

This section discusses FTC as the other component of FTCS. Fault tolerant controllers belong to the class of smart or intelligent controllers with built-in diagnostics. They are capable of tolerating failures or malfunctions in system components, actuators and sensors and still deliver satisfactory performance despite those failures. Hence, the main purpose in FTC is to design a controller with suitable structure to achieve stability and satisfactory performance, whether or not all the system components including the control system itself are functioning correctly. An extensive number of researches has been carried out on FTC since the early 1980s (Chandler, 1984; Vander Velde, 1984; Eterno *et al.*, 1985; Stengel, 1991; Rauch, 1994; Rauch, 1995; Blanke *et al.*, 1997; Blanke *et al.*, 2001; Isermann *et al.*, 2002; Blanke *et al.*, 2003; Mahmoud *et al.*, 2003; Zhang and Jiang, 2003; Steinberg, 2005; Blanke *et al.*, 2006; Isermann, 2006). This was motivated by the need to give aircraft

control system much needed control capabilities to accommodate faults within the system and still be able to land the aircraft safely. Interest in the design and application of FTC grew in the other industries due to the increased safety and reliability demand beyond what conventional controllers offer. These industries include aerospace, nuclear power plants, automotive, manufacturing and chemical and process industries (Isermann *et al.*, 2002; Mehrabi *et al.*, 2002; Bruccoleri *et al.*, 2003; Zhang and Jiang, 2008).

Several techniques have been used in the design of fault tolerant controllers; Zhang and Jiang (2008) gave a detailed classification of such techniques as shown in Figure 2.9. They used criteria such as mathematical design tools, design approaches, reconfiguration mechanism, and the type of systems investigated. It is not surprising that most of the techniques that have been researched in FTC are concentrated in the aerospace and aviation industry due to its historical reasons. Some impressive results on the design and application of FTC have been published lately: application of distributed model predictive control (DMPC) to accommodate actuator faults in a three unit continuous stirred tank reactor (Chilin *et al.*, 2010a; Chilin *et al.*, 2010b; Chilin *et al.*, 2012b); the use of adaptive controller for FTC in General Electric XTE46 engine (Diao and Passino, 2002); combined model predictive control (MPC) and H_∞ robust controller (Mirzaee and Salahshoor, 2012) and the use of proactive fault tolerant Lyapunov-based MPC (Lao *et al.*, 2013) rather than reactive FTC that have been the norm over the last two decades or so. Many of the techniques employed in FTC as presented in Figure 2.9 rely on ideas that had been investigated in the past for other control purposes. Though well-known control design techniques are used, they face new challenges and problems that may not appear in the conventional controller design (Zhang and Jiang, 2008). It is essential that such control methods deliver some good level of performance in the impair system in an online real time manner. Owing to the demand and performance requirement of an FTC, it is not unusual for an FTC to have a combination of different control structures and control design algorithm. This thesis focuses on fault tolerant model predictive controller and simple restructurable active FTC that uses backup feedback signal design approaches for actuator fault accommodation, and fault tolerant inferential controller (FTIC) for sensor fault accommodation respectively.

2.5.1 Fault Tolerant Model Predictive Control (FTMPC)

Model predictive control is a high performing model-based process control strategy with ability to handle multivariable interactions, constraints on control inputs and system states, and optimisation requirements in a systematic manner. It is popular in the process control industry because the actual operating objectives and operating constraint can be represented explicitly in the optimisation problem solved at every control instant (Camponogara *et al.*, 2002). Several researchers have worked and continue to work on FTMPC with interest in the area growing daily. Mhaskar (2006) designed a robust model predictive controller to achieve fault-tolerant control of nonlinear systems subject to uncertainties, constraints and actuator fault. He used Lyapunov-based approach to formulate constraints that account for uncertainty explicitly in the predictive control law and also explicitly to allow the characterisation of initial conditions starting from where closed-loop stability is guaranteed. Zhang *et al.* (2014) used state space model predictive fault-tolerant control to accommodate partial actuator faults in batch processes with unknown disturbances. Zhang *et al.* (2014) propose an improved cost index that can aid selection of relevant weighting factors for better control performance. Tao *et al.* (2014) applied state space model predictive control to accommodate partial actuator fault in linear systems. Lao *et al.* (2013) proposed proactive Lyapunov-based fault-tolerant model predictive control to handle effectively incipient actuator fault in chemical processes.

Mirzaee and Salahshoor (2012) presented a unified robust fault tolerant control framework to effectively handle changes in unmeasured disturbance and model parameters, biases and drifts in sensors and actuators respectively. This was achieved through the use of adaptive unscented Kalman filters (AUKFs) and fuzzy-based decision making (FDM) algorithm for fault detection and isolation, and actuator and sensor faults diagnostics respectively. The AUKF and FDM schemes were integrated with H_∞ optimal robust controller and MPC using a fuzzy switch scheme for switching between MPC and robust controller for effective performance in actuator and sensor faults accommodation.
Generally, the design of an MPC has three main components:

- The model of the system under consideration. This is used essentially for the system open-loop future trajectory prediction and in large part plays a crucial role in the effectiveness or otherwise of the MPC.
- A control objective function to be minimised subject to constraints imposed by the system model, restrictions on control inputs, system states and others.

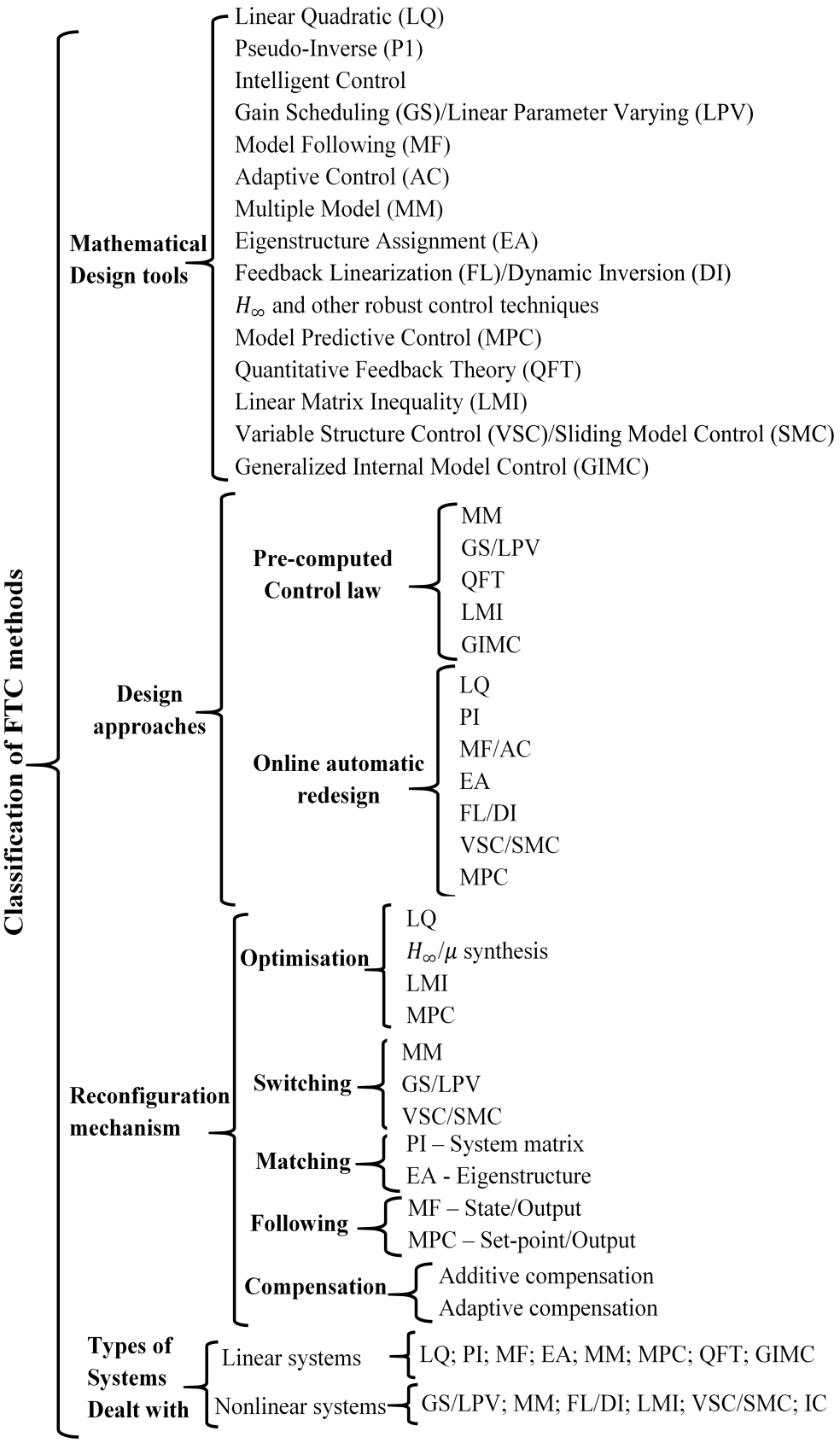


Figure 2.9: Classification of FTC *(Source: Zhang and Jiang, 2008)*

- A receding horizon scheme that introduces feedback into the control law for disturbances and model-mismatch compensation.

Consider the state space model of a system as given below:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0 \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{2.55}$$

where x, u and y are the state variables, inputs and outputs of the system respectively, A, B, C and D are matrices of appropriate dimensions. A brief description of a typical MPC formulation is given as (Garcia *et al.*, 1989):

$$\min_{u \in S(\Delta)} \int_{t_k}^{t_{k+N}} [\|\tilde{x}_i(\tau)\|^2_{Q_{C1}} + \|u_i(\tau)\|^2_{R_{C1}}]dt \tag{2.56a}$$

$$s.t. \quad \dot{\tilde{x}} = f(\tilde{x}, u(t)) \tag{2.56b}$$

$$u_i(t) \in U_i \tag{2.56c}$$

$$\tilde{x}(t_k) = x(t_k) \tag{2.56d}$$

where $S(\Delta)$, N and $\tilde{x}$ denote the family of piece-wise constant functions with sampling interval Δ, the prediction horizon and the predicted trajectories of the nominal system in (2.55) respectively, Q_{c1} and R_{c1} are positive definite symmetric weighting matrices. The objective function in (2.56a) is to be minimised subject to constraint (2.56b) which is supposed to have zero uncertainties in model (2.55) used to predict future trajectories of the system. Constraints (2.56c) and (2.56d) take into account the restrictions on the control inputs and the measured system states respectively. The first step of the optimal solution defined by (2.56), denoted as $u_i^*(t \mid t_k)$ is implemented and the whole procedure is repeated continuously.

2.5.2 Distributed Model Predictive Control

MPC typically works in a centralised fashion, but when dealing with complex systems, as we have in the chemical and oil and gas industry, for optimality, it may be better to have distributed control schemes where local control inputs are computed using local measurements and reduced-order of the sub-system dynamics. DMPC are used to coordinate the implementation of separate MPC controllers to achieve optimal input trajectories in a

distributed manner. It is a developing research area with interest from both the academia and the industry. A review of DMPC by Christofides *et al.* (2013) gave algorithmic details of the different approaches that have been used in the design and implementation of DMPC to provoke further researches in the area. Rawlings and Stewart (2008) presented cooperative DMPC to guarantee nominal stability and performance properties with high degree of communication between local controllers by using MPCs with modified objective functions. Mercangoz and Doyle III (2007) proposed a DMPC algorithm based on the work of Mutambara (1998) and implemented it for level control on an experimental four-tank system. Chilin *et al.* (2010a) demonstrated the application of DMPC for actuator faults; their work is briefly outlined below. Consider a nonlinear system described by a state-space model:

$$\dot{x}(t) = f(x(t), u_1(t), u_2(t), d(t)) \tag{2.57}$$

where $x(t) \in \Re^{n_x}$ denotes state variables vector, $d \in \Re^p$ is the model of the set of possible faults, $u_1(t) \in \Re^{n_{u1}}$ and $u_2(t) \in \Re^{n_{u2}}$ are the two different sets of possible manipulated inputs. The faults are unknown and d_j, $j = 1 \ldots p$, can take any value. The system is controlled by two sets of control input u_1 and u_2 (i.e. $u(t) = u_1(t) + u_2(t)$). They assumed a Lyapunov-based controller $u_1(t) = h(x)$ exists, which renders the origin of the fault-free closed-looped system asymptotically stable with $u_2(t) = 0$.

Then, they designed a DMPC structure (see Figure 2.10) to achieve closed-loop stability and performance using two Lyapunov-based MPC, LMPC2 and LMPC1 to compute control input trajectories u_2 and u_1 respectively (Chilin *et al.*, 2012a; Chilin *et al.*, 2012b). Consider the expressions for LMPC2 (equation 2.58a – 2.58e) and LMPC1 (equation 2.59a – 2.59d) below:

$$\min_{u_{d2} \in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T(\tau) Q_c \tilde{x}(\tau) + u_{d1}^T(\tau) R_{c1} u_{d1}(\tau) + u_{d2}^T(\tau) R_{c2} u_{d2}(\tau)] d(\tau) \tag{2.58a}$$

$$\dot{\tilde{x}}(\tau) = f(\tilde{x}(\tau), u_{d1}(\tau), u_{d2}(\tau), 0) \tag{2.58b}$$

$$u_{d1}(\tau) = h(\tilde{x}(j\Delta)), \quad \forall\, \tau \in [j\Delta, (j+1)\Delta), \quad j = 0 \ \ldots \ N-1 \tag{2.58c}$$

$$\tilde{x}(0) = x(t_k) \tag{2.58d}$$

$$\frac{\delta V(x)}{\delta x} f(x(t_k), h(x(t_k)), u_{d2}(0), 0) \leq \frac{\delta V(x)}{\delta x} f(x(t_k), h(x(t_k)), 0, 0) \tag{2.58e}$$

and

$$\min_{u_{d1}\in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T(\tau)Q_c\tilde{x}(\tau) + u_{d1}^T(\tau)R_{c1}u_{d1}(\tau) + u_{d2}^{*T}(\tau \mid t_k)R_{c2}u_{d2}^*(\tau \mid t_k)]d(\tau) \quad (2.59a)$$

$$\dot{\tilde{x}}(\tau) = f(\tilde{x}(\tau), u_{d1}(\tau), u_{d2}^*(\tau \mid t_k), 0) \quad (2.59b)$$

$$\tilde{x}(0) = x(t_k) \quad (2.59c)$$

$$\frac{\delta V(x)}{\delta x} f(x(t_k), u_{d1}(0), u_{d2}^*(0 \mid t_k), 0) \leq \frac{\delta V(x)}{\delta x} f(x(t_k), h(x(t_k)), u_{d2}^*(0 \mid t_k), 0) \quad (2.59d)$$

where V is the Lyapunov function, $\tilde{x}$ is the predicted trajectory for the fault-free system with u_2 being the input trajectory computed by the LMPC2 and u_1 being the Lyapunov-based controller $h(x)$ applied in a sample and hold fashion. The DMPC is implemented thus:

1. Both LMPC1 and LMPC2 receive the state measurement $x(t_k)$ from the sensor at each sampling instant t_k.

2. LMPC2 evaluates the optimal input trajectory of u_2 based on $x(t_k)$ and sends the first step input value to its corresponding actuators and the entire optimal input trajectory to LMPC1.

3. After receiving the entire input trajectory of u_2 together with $x(t_k)$, LMPC1 evaluates the future input trajectory of u_1.

4. LMPC1 then sends the first step input value of u_1 to its corresponding actuators.

$u_{d2}^*(\tau \mid t_k)$ and $u_{d1}^*(\tau \mid t_k)$ are the optimal solutions to the optimisation problems of LMPC2 and LMPC1 respectively. Hence, the manipulated inputs to the system are:

$$\begin{cases} u_1(t \mid x(t_k)) = u_{d1}^*(t - t_k \mid t_k), \quad \forall\ t \in [t_k, t_{k+1}) \\ \\ u_2(t \mid x(t_k)) = u_{d2}^*(t - t_k \mid t_k), \quad \forall\ t \in [t_k, t_{k+1}) \end{cases} \quad (2.60)$$

A non-zero residual is generated when fault occurs in a system. The residual is generated through this expression ($\hat{x}$ and x are the filter state for the fault-free system and the measured state respectively):

$$r(t) = \mid \hat{x}(t) - x(t) \mid \quad (2.61)$$

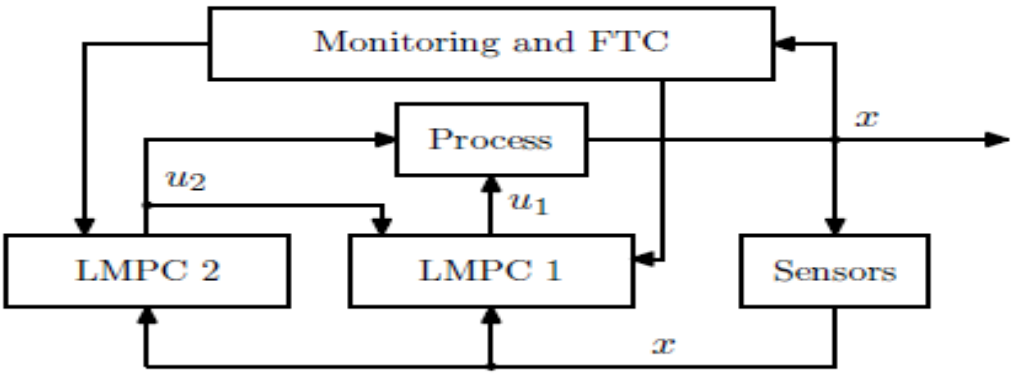


Figure 2.10: DMPC structure *(Source: Chilin et al, 2010)*

The main idea behind this approach is that there is an extra control input u_2 that can be called upon to stabilize the system in the presence of a fault. It is suspected that the effectiveness of the approach will depend on the type and severity of the fault.

2.5.3 Fault Tolerant Inferential Control

Inferential control is a robust strategy that has been deployed in the industry for decades. It is used to control variables whose measurements are not readily accessible (i.e. cannot be measured directly) or have substantial time delay (Kano *et al.*, 2000; Zhang, 2006b) and therefore need to be inferred from the measured secondary variables that have reasonable correlation with the unmeasured controlled variables. The inference is usually made through a variety of analytical techniques. The process through which the inference is achieved could be referred to as soft-sensing or soft-sensor. Bolf *et al.* (2008) used neural network-based soft sensor to estimate kerosene distillation end point and freezing point in a crude refinery operation. Zhou *et al.* (2012) proposed bootstrap aggregated neural network and bootstrap aggregated partial least square regression techniques to infer and control kerosene dry point in refineries with varying crudes. Kano *et al.* (2000) also employed dynamic partial least square regression to estimate distillate and bottom compositions for inferential control of a distillation system. Zhang (2006b) developed an off-set free inferential control strategy with principal component regression and partial least squares for distillation composition control.

The main idea of estimating certain variable of interest from other related variables has also been exploited in sensor fault accommodation. Deshpande *et al.* (2009) applied state estimator approach to provide controlled variable estimate for feedback control af-

ter declaration of a sensor fault. Manuja *et al.* (2008) used a combination of generalised likelihood ratio and reduced order models to accommodate sensor fault in an ideal binary distillation column. However, there are some application issues that need to be addressed during implementation, especially when the sensor fault effect has manifested in the secondary variables used for inference which could lead to inaccurate estimates. The use of inaccurate controlled variable estimate for feedback control purposes could worsen the already abnormal situation in the system and may cause the system to be unstable.

2.6 Soft Sensor Estimator

Soft sensor or software sensor generally refers to analytical ways of obtaining values of certain variable(s) of interest from several other measurements. There are several techniques available for prediction of an output from process measurements. Dynamic principal component regression (DPCR) and dynamic partial least square (DPLS) techniques are the focus of this thesis.

2.6.1 Dynamic Principal Component Regression

DPCR is a variant of principal component regression (PCR) and uses principal components (PCs) of the process variable matrix for its regression model development. DPCR is used to develop soft sensors for controlled variables whose sensors could potentially develop fault. The soft sensors can be developed using carefully selected process measurements that have good correlation with the controlled product quality variables. The input matrix X that is used for the regression comprises of process measurements at the current and the l previous sampling times to incorporate system dynamics into the DPCR model. A brief description of DPCR is given below. More information about PCR and DPCR can be found in Geladi and Kowalski (1986) and Ku *et al.* (1995). Given an $n \times (l+1)p$ matrix X with n samples, p process variables and l previous sampling time measurements. Then we can have a linear combination of the first a principal components (PC) of X as the model output given as:

$$T_a = XU_a = Xu_1 + Xu_2 + Xu_3 + \ \ldots \ + Xu_a \tag{2.62}$$

$$\begin{cases} \hat{y} = T_a w = XU_a w = Xu_1 w_1 + Xu_2 w_2 + Xu_3 w_3 + \ \ldots \ + Xu_a w_a \\ or \\ \hat{y} = XU_a w = X\theta = x_1\theta_1 + x_2\theta_2 + x_3\theta_3 + \ \ldots \ + x_{(l+1)p}\theta_{(l+1)p} \end{cases} \tag{2.63}$$

where $T_a = [t_1 \quad t_2 \quad \ldots t_a]$ are the PC scores, $U_a = [u_1 \quad u_2 \quad \ldots u_a]$ are the PC loading vectors, $\hat{y}$ is the controlled product quality variable estimate, w is a vector of model parameters in terms of PCs, x_1 to $x_{(l+1)p}$ are the model input (selected process variables), and θ_1 to $\theta_{(l+1)p}$ are the model parameters. The least squares estimation of w is:

$$\hat{w} = (T_a' T_a)^{-1} T_a' Y = (U_a' X' X U_a)^{-1} U_a' X' Y \tag{2.64}$$

Then, the model parameters in (2.63) can be obtained through DPCR as:

$$\hat{\theta} = U_a \hat{w} = U_a (U_a' X' X U_a)^{-1} U_a' X' Y \tag{2.65}$$

The data set that is used to develop the DPCR is usually partitioned into training and testing data sets, and models with different numbers of principal components would be developed using the training data and then tested with the testing data set. The model with the smallest testing error is then selected.

2.6.2 Dynamic Partial Least Square

Dynamic partial least square, also known as dynamic projection to latent structure (DPLS) is an extension of projection to latent structure discussed in Section 2.4.2.2. It incorporates time-lagged measurements of the regressor or process variables into the input matrix X to capture the dynamic behaviour of the system whose responses Y are to be predicted. This increases the columns of regressor matrix X, or the number of process variables "p" by a factor of l (number of time-lag), giving $n \times (l+1)p$ dimension for matrix X. The DPLS projects matrices X and Y onto a subset of latent variables t and u respectively. More details about the PLS approach can be found in Geladi and Kowalski (1986). Equations (2.51) and (2.52) in Sections 2.4.2.2 can also be presented as below:

$$X = \sum_{j=1}^{a} t_j p_j^T + E \tag{2.66}$$

$$Y = \sum_{j=1}^{a} u_j q_j^T + F \tag{2.67}$$

A linear relationship between the regressor and response variables is achieved by performing an ordinary least squares regression between each pair of the corresponding t and u vectors.

$$\hat{u}_j = t_j b_j \qquad j = 1, 2, \ldots, a \tag{2.68}$$

where b_j is the coefficient from the inner linear regression between the jth latent variables t_j and u_j, i.e.:

$$b_j = u_j^T \cdot t_j / (u_j^T \cdot u_j) \tag{2.69}$$

The number of latent variables to be retained can be determined through cross-validation.

2.7 Summary

In this chapter, fault tolerant control systems (FTCS) and its major components – fault detection and diagnosis (FDD) and fault tolerant controllers (FTC) were discussed. The two types of FTCS – passive and active FTCS and the latest researches and techniques that have been developed in these areas were also presented. Passive FTCS was briefly discussed while actives FTCS being the focus of this thesis was discussed in sufficient details. The two main categories of FDD – model based and data based FDD and some of the state-of-the-art approaches developed under the two categories were sufficiently discussed. Under the model based category, approaches such as residual generation techniques, state estimation approach, parameter estimation approach, parity space approach and unknown input observer; including the modelling of faulty system under model based were all discussed. Similarly, FDD approaches under the data based category comprising of dynamic principal component analysis, dynamic partial least squares and neural network were equally discussed.

The second component of FTCS, the reconfigurable fault tolerant controller (FTC) and the numerous researches undertaken in the field were also presented in this chapter. Different techniques used in achieving FTC were also discussed. However, only model predictive control (MPC) and distributed model predictive control (DMPC) were briefly outlined, as they form the basis of one of the approaches used for actuator fault accommodation in this thesis. A brief mention of fault tolerant inferential control (FTIC) for sensor faults accommodation and some of the techniques used in this thesis for soft sensor

estimation were similarly presented. All the relevant aspects of FTCS this thesis focuses on were all discussed in sufficient details.

Chapter 3

Simplified Fault Tolerant Control Systems

3.1 Introduction

The design of simple restructurable feedback controllers like proportion-integral-derivative (PID) with backup feedback signals and switchable reference points is discussed in this chapter. As discussed in the previous chapter, most of the current fault-tolerant controllers involve techniques with high level of complexity and computational tasks in their design and implementation. The proposed fault-tolerant control technique offers a simplified approach to the design and implementation of controllers capable of tolerating actuator and sensor faults in complex systems, with perhaps acceptable graceful performance degradation. The approach is expected to achieve results comparable to those employing complex computational tasks, though different possible control structures would have to be analysed a priori using tools like relative gain array (RGA) and dynamic relative gain array analysis to select possible switching options. As it is often the case that, for any given process, there are several ways of controlling it, some better than others, so selecting a sub-optimal strategy under faulty condition would be far more acceptable than process shut-down. However, the switchability and restructurability of a fault tolerant controller are process dependent as maintaining acceptable level of performance in some processes may not always be achievable due to lack of suitable controlled and manipulated variable pairing. This has to be carefully assessed taking into consideration the remaining healthy actuators and the process variables pairing for control purposes.

Simplified fault-tolerant control system for actuator faults that uses data-based tech-

nique for actuator fault detection and diagnosis is first presented in the next Section. The mechanism used to integrate the FDD and the simplified FTC with its different components is discussed. The control strategies and the tools employed in identifying and analysing different control loops pairing pre and post-fault era are then presented, in order to achieve a seamless switching and stability in the system post-fault era. This is followed by detailing the procedures used to achieve the design of FTIC, the sensor fault-tolerant control system component of the FTCS. It includes how the faulty controlled variable measurements are estimated using DPCR and DPLS and the eventual integration of the predicted values into the FTCS. The chapter is concluded with the presentation of the combined actuator and sensor faults FTCS.

3.2 Fault Tolerant Control System Design for Actuator Faults

The detailed procedure for the design of actuator fault-tolerant control system for complex chemical processes is presented in this section. The system is a conventional PID controller with extended restructurable capability for actuator fault accommodation. PID controllers are relatively easy to implement and are popular among plant operators. Its extended version with additional capabilities, particularly one that offers fault-tolerant control capability without any doubt will be an excellent addition to the list of growing FTCS for industrial applications. The block diagram of a conventional feedback controller for a single loop as presented in Figure 3.1 has four major components — the PID controller, actuator, plant and sensor. r, e, u_c, u and d in Figure 3.1 are set-point, deviation, controller output, manipulated variable and disturbance respectively, while y, y_p and y_s are actual controlled output, measured primary controlled output and measured uncontrolled secondary variables respectively.

Figure 3.2 presents the structure of the proposed FTC for actuator faults for a complex chemical process. The figure has some notable differences from that of a conventional feedback controller. There are additional blocks like DPCA FDD scheme, reconfiguration mechanism, two weighting matrices blocks and a reconfigurable PID controller block instead of an ordinary PID controller block. e, u_c, u_d, y, y_p and y_s are vectors, as previously defined while r_b, r_r, u_b, y_b and y_y are vectors of appropriate dimensions for back-up set point signal, switchable references, back-up manipulated variables, controlled variables

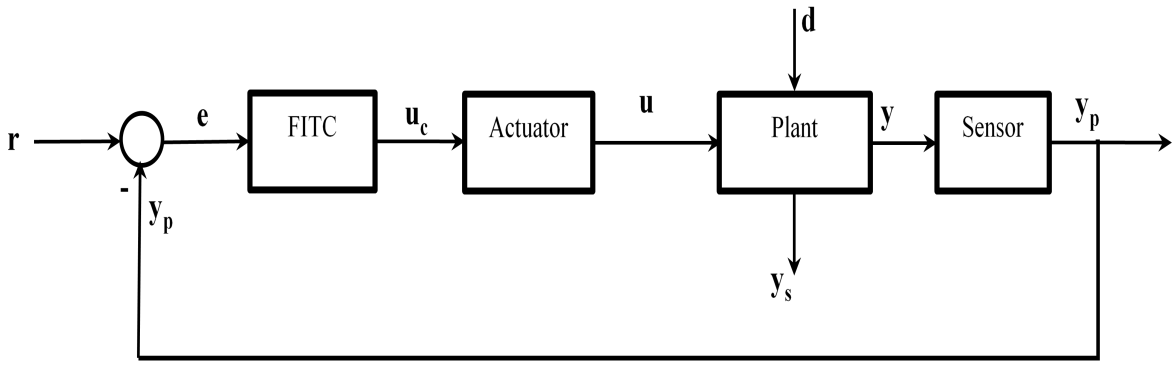


Figure 3.1: A conventional feedback controller block diagram

back-up signals and restructurable controlled outputs respectively. r_p is the vector form of r, which is the reference points for the primary controlled variables. The DPCA FDD scheme of the system deals with process monitoring for timely and accurate detection and diagnosis of actuator faults. The reconfiguration mechanism acts on the fault information received from the FDD scheme. It contains several possible controller switching options designed *a priori* based on rigorous analysis of a closed set of possible actuator faults using RGA and DGRA, including stability analysis of the entire system. The reconfigurable PID controller implements the selected reconfigurable option by reconfiguring its control structure after isolating the faulty actuator using the back-up signals for reference points and the primary controlled outputs. This is made possible with the use of the weighting matrices blocks for seamless implementation. Procedures involved in some of the major components of the FTCS are detailed in the next sections.

3.2.1 DPCA FDD Scheme

Dynamic PCA monitoring technique is used in the integrated actuator FDD scheme to identify possible actuator faults occurrence. In order to avoid repetition, the procedures presented in Section 2.4.2.4 will be augmented further to highlight how the FDD scheme functions in the whole FTCS system. The matrix of the scaled measurement "X" in equation 2.40 is given as:

$$X = [u \quad y_p \quad y_s] \tag{3.1}$$

where X is the matrix of past measurements of all the process variables to be included in the DPCA diagnostic model with dimension $n \times (l+1)p$, u is an $n \times (l+1)np$ matrix

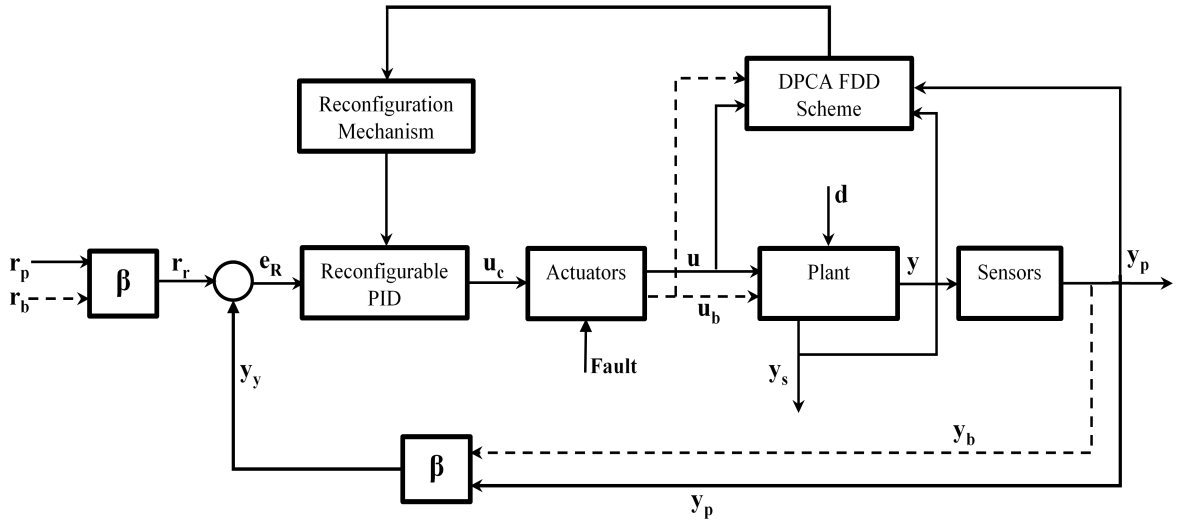


Figure 3.2: The proposed FTC for actuator faults

of manipulated variables, y_p is an $n \times (l+1)np$ matrix of primary controlled variables, y_s is an $n \times (l+1)ns$ matrix of measured secondary variables and p ($p = 2np + ns$) is the total number of variables included in the monitoring diagnostic model during normal operation. n, np, ns and l are the total number of samples, total number of primary controlled variables, number of measured secondary variables and the time lag considered respectively. It is assumed that the manipulated variable "u" is always available, otherwise it can be obtained from the knowledge of the controller output "u_c". The first phase of the FDD scheme is the development of an actuator fault detection scheme, as described in Section 2.4.2.4. The scheme is then used to monitor the process for possible actuator faults using the computed control limits for the Hotelling's T^2 and the SPE monitoring statistics as presented in equations 2.41 and 2.42. The second phase of the FDD scheme involves fault diagnosis to identify the particular faulty actuator through the use of contribution plots of the monitoring statistics.

Contribution plots are simply graphical representations depicting the contributions of each variable in the diagnostic model to the values of the Hotelling's T^2 and SPE monitoring statistics, particularly upon detection of a fault. In this thesis, excess contributions of each variable is used by first computing their total contributions to the monitoring statistics at the point of fault declaration and the following two consecutive sampling periods for proper diagnosis. Average contributions of each variable to the monitoring statistics during normal operation are also obtained and subtracted from the total contributions earlier computed to obtain the variable excess contribution. The variables that contribute

the most to the faulty situation are then mapped to a particular actuator fault based on the knowledge of the system. Hotelling's T^2 variable contributions to a faulty actuator are obtained using the following equations (Kourti and MacGregor, 1996). Let r be the number of score vectors that violate their limits ($r \leq a$).

$$cont\_x_{k_{f,j}} = \frac{t_{f,j}}{\lambda_j}\, u_{k,j}\, x_{f,k} \tag{3.2}$$

where $cont\_x_{k_{f,j}}$ is the contribution of variable x_k to score vector t_j at point f (point of fault declaration); a is the number of principal components, $t_{f,j}$, λ_j and $u_{k,j}$ are the score vector t_j, the corresponding eigenvalue and loading vector for k^{th} variable respectively at the faulty sample f, while $x_{f,k}$ is variable x_k also at point f. The value of $cont\_x_{k_{f,j}}$ represents $\frac{t_{f,j}^2}{\lambda_j}$ which should always be positive and it is set equal to zero if negative. The total contribution of variable x_k to the detected fault is given as:

$$cont\_x_k = \sum_{j=1}^{r} (cont\_x_{k_{f,j}}) \tag{3.3}$$

Average variable contributions to the monitoring statistics during normal operation at any instance is given as:

$$avg\_x_{k,j} = \frac{t_j}{\lambda_j} u_{k,j} x_k \tag{3.4}$$

$avg\_x_{k,j}$ should always be positive and is set to zero if negative. The overall average contributions of each variable to Hotelling's T^2 monitoring statistics pre-fault era is given as:

$$x_{k\_avg} = \sum_{j=1}^{n} (avg\_x_{k,j}) \tag{3.5}$$

Subtracting equation 3.5 from equation 3.3 gives the excess contributions of each variable to the out of control situation, which are then plotted to identify the variables indicative of the fault and then mapped unto a particular actuator fault. After successful detection and diagnosis of an actuator fault and subsequent implementation of the FTC, for continued process monitoring, np is reduced by 1 and p by 2. The dimension of X post-fault era now reduces to $n \times (l+1)(p-2)$. This reflects the isolation of the faulty actuator and subsequent removal of a controlled variable for further system monitoring. Output of the FDD monitoring scheme is passed on to the reconfiguration mechanism to reconfigure the input-output pairing for the whole system as appropriate.

3.2.2 Control Strategies and Loop Pairing Assessment

It is imperative that rigorous process interaction of the multivariable system is undertaken, in order to have good understanding of the effect of variable pairing reconfiguration on the stability of the system, particularly during faults accommodation. Different control strategies during normal operation and faulty conditions are investigated to determine the optimum and sub-optimal controlled variable-manipulated variable pairing for every potential actuator fault in the system. The task involved is non-trivial and it is achieved through the use of RGA and DRGA.

3.2.2.1 Relative Gain Array (RGA)

Relative gain array, developed by Bristol (1966) and extended by Mcavoy (1983) and Shinskey (1988) is used for the control loop interaction analysis. A brief description of the procedures involved in the analysis is given in this section. RGA gives a quantitative measure of the level of interaction amongst the loops of a multivariable control structure using the system process gains matrix, which defines the steady state open-loop relationship between the inputs and outputs. Let the relationship between outputs and inputs of a multivariable system be presented as below:

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ y_p(s) \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1p} \\ k_{21} & k_{22} & \cdots & k_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ k_{p1} & k_{p2} & \cdots & k_{pp} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \\ \vdots \\ u_p(s) \end{bmatrix} \tag{3.6}$$

Equation 3.6 can be presented in a compact form as:

$$y(s) = K \cdot u(s) \tag{3.7}$$

where $y(s)$, $u(s)$ and K are controlled outputs, manipulated inputs and the steady state process gain matrix respectively. K can be obtained by independently varying the manipulated inputs of the multivariable system one at a time and then allowing the system to reach a new steady state. Several changes can be made to individual manipulated variable over a reasonably long period of time during the process simulation to gather enough data, which can then be used to obtain a more accurate K matrix and dynamic models, in this case, transfer function models of the system using System Identification Toolbox in MATLAB. The RGA (Λ) of the system can then be obtained using:

$$\Lambda = K \cdot^* (K^T)^{-1} \tag{3.8}$$

where $\cdot^*$ represents element by element multiplication. Several Λ for different sets of K matrices will have to be analysed for each possible actuator fault and implemented on the system to assess the stability of the system under various degrees of actuator faults. The RGA analysis could involve several hundreds of different inputs-outputs pairing for all the possible actuator faults, particularly for complex system. This will help to determine an optimum/sub-optimal inputs-outputs pairing during controller reconfiguration in any faulty situation.

3.2.2.2 Dynamic Relative Gain Array (DRGA)

RGA has some limitations as it does not consider the transient behaviour and effect of presence of disturbances in the system. DRGA is used in conjunction with RGA for a more robust loop pairing and stability analysis. DRGA was first introduced by Witcher and Mcavoy (1977) and later by Bristol (1979) to address the perceived limitations of RGA by using the transfer function models of the system instead of the traditional steady state process gains. It can give more accurately the extent of interactions that is present amongst different loop pairing and more insight into the stability of the system, especially during controller reconfiguration. The denominator of the transfer function models provides an opportunity to evaluate the magnitude of the elements of relative gain at several frequencies by setting $s = jw$.

3.2.3 Reconfigurable PID Controllers

As it is often the case, for any given process, there are several possible sub-optimal control structures (input-output pairing) for the system, some more effective than others. The simple reconfigurable PID controller proposed here leverages on the opportunity of having more than one manipulated variables that can be used to effect control of an output. Several possible control structures will have to be assessed *a priori* as explained in the last section using RGA and DRGA, and then stored for possible implementation in the event of an actuator fault being identified. Let the control error generated by a conventional feedback control law in Figure 3.1 be:

$$e = r - y_p \tag{3.9}$$

and the control error with back-up feedback signal for an actuator fault in Figure 3.2 be

$$e_R = r_r - y_y \tag{3.10}$$

where

$$r_r = \beta[r_p^T \quad r_b^T]^T \quad and \quad y_y = \beta[y_p^T \quad y_b^T]^T \tag{3.11}$$

β is a weighting matrix block given as:

$$\beta = diag(\beta_p \quad \beta_b) = \begin{bmatrix} \beta_p & 0 \\ 0 & \beta_b \end{bmatrix} \tag{3.12}$$

β_p and β_b during normal operation, are identity and zero square weighting matrices with dimension $(np,\ np)$ for the primary controlled variables and back-up feedback signals respectively. The weighting matrices are used to deactivate and activate actual and backup feedback signal as appropriate during fault-tolerant controller reconfiguration. Substituting equation 3.11 and equation 3.12 into equation 3.10 gives

$$e_R = \begin{bmatrix} \beta_p & 0 \\ 0 & \beta_b \end{bmatrix} [r_p^T \quad r_b^T]^T - \begin{bmatrix} \beta_p & 0 \\ 0 & \beta_b \end{bmatrix} [y_p^T \quad y_b^T]^T \tag{3.13}$$

Let the reconfigurable PID controller be

$$G_R = [G_c^T \quad G_b^T]^T \tag{3.14}$$

where G_c and G_b are the actual controllers used during normal process operation and the pre-assessed backup feedback controllers to accommodate possible actuator fault occurrence respectively. Weighting matrix β is also introduced in equation 3.14 in order to implement the reconfigurable controller, which now becomes

$$G_{RC} = \beta G_R \tag{3.15}$$

The control law for the reconfigurable fault tolerant PID controller is then given as

$$u = G_{RC} e_R \tag{3.16}$$

The different possible manipulated and controlled variable pairing are assessed *a priori* to decide on the reconfiguration pairing upon detection and identification of a fault.

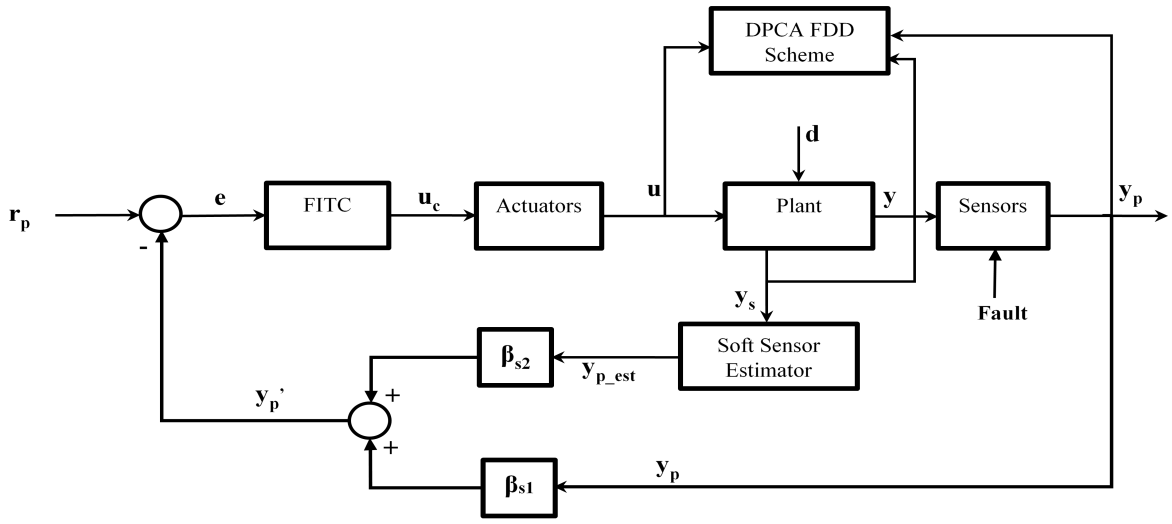


Figure 3.3: The proposed FTIC for sensor faults

Hence, accommodation of any individual fault is dependent on having a suitable healthy actuator that is able to provide satisfactory performance in the impaired system. Only a single fault-tolerant control system is considered in this thesis. However, the approach can also be applied to duplex FTCS structure. By single and duplex FTCS, we mean a single and double fault-tolerant control system backup for each pre-assessed actuator fault provided there are suitable restructurable manipulated and controlled variable pairings.

3.3 Fault Tolerant Control Systems Design for Sensor Faults

Faulty sensor output degrades the performance of a control system, and could significantly impact on the economy of the plant. It could also potentially lead to disaster if the fault effect is not quickly and adequately mitigated. Multiple sensors have been used in the past to accommodate the effects of any potential failure in sensors, particularly for safety-critical systems. The use of analytical means to achieve the same purpose has been on the increase in the last few decades. Analytic methods are also used in some cases to detect and identified sensor faults, particularly for the model-based FDD. This work uses DPCR and DPLS approaches to estimate primary controlled variables whose values could be affected by potential faulty sensors using appropriate measured uncontrolled secondary variables as part of a fault-tolerant control scheme. Figure 3.3 presents the structure of the proposed fault-tolerant control system for sensor fault accommodation. The sensor fault-

tolerant controller is referred to as fault-tolerant inferential controller (FTIC) and shares the same DPCA FDD scheme with FTC for actuator fault accommodation. Sensor faults are detected and diagnosed in the same way as actuator faults using Hotelling's T^2 and SPE monitoring statistics together with their contribution plots, as discussed in Section 3.2.1.

3.3.1 Soft Sensor Estimator Using DPCR and DPLS

The soft sensor estimator block in Figure 3.3 computes estimates of the primary controlled variables using both DPLS and DPCR with the procedures detailed in Section 2.6. The two approaches each produces an estimate for every controlled variable in the system, and the more accurate estimate, that is the estimate that is closest to the actual measured controlled output is selected for that particular controlled variable. This exercise is carried out with the data collected during normal operation of the system to determine the best approach for each controlled variable. In order to achieve this, several matrix blocks are first identified, each containing variables that have good correlation with and best describe a particular output. The matrix blocks are then used with the best approach as explicated in Section 2.6 to develop soft sensor estimates of the primary controlled variables (y_p) before, during and after occurrence of sensor faults. The expression below presents the relationship between the estimates and the actual measured controlled variables pre and post-fault era.

$$\begin{cases} y_{p\_est} \approx y_p & for \quad t < t_f \\ y_{p\_est} \neq y_p & for \quad t \geq t_f \end{cases} \tag{3.17}$$

where y_{p_est} and y_p are vectors of estimates and actual measured controlled outputs, t is a time instance during the simulation and t_f is the time a sensor fault is declared. The element of y_{p_est} corresponding to that of faulty y_p is then used in place of the faulty sensor output for continued safe and satisfactory operation of the concerned control loop. Next section gives more details on how this is achieved.

3.3.2 Fault Tolerant Inferential Controller

Inferential control is a control strategy that has been implemented in the industry for decades. It is usually employed in situations where the controlled variables are not readily accessible (i.e. cannot be measured directly) or have substantial time delay (Kano *et*

al., 2000; Zhang, 2006b) and need to be inferred from measured secondary variables that have strong correlation with the controlled variable. Its application in this thesis is rather for substitution of faulty sensor outputs to maintain the integrity of the whole control system. Essentially, empirical models developed from process data is used to achieve the inference. However, care must be taken when there is strong correlation among the measurement variables to be used for the estimation as collinearities will inflate the variances of the least squares estimators. This could in turn lead to inaccurate model parameters estimations and ultimately inaccurate inferred values for the controlled variables, which is why principal components (PC) are used to estimate the model parameters as presented in equations 2.64 and 2.65. In the proposed FTIC, a redundant controlled variable signal is always available through the soft sensor scheme which is then called upon after a sensor fault is identified. From Figure 3.3, let the controlled variables compared with the references r_p be

$$y'_p = \beta_{s1}\ y_p + \beta_{s2}\ y_{p\_est} \tag{3.18}$$

where β_{s1} and β_{s2} are weighting matrices used to implement the measured and estimated controlled variable signals. β_{s1} is defined as:

$$\begin{cases} \beta_{s1} = I & for \quad t < t_f \\ \beta_{s1} \neq I & for \quad t \geq t_f \end{cases} \tag{3.19}$$

During normal operation, β_{s1} is an identity matrix while β_{s2} is a zero matrix. However, when a sensor fault is declared, the faulty sensor is isolated and the diagonal element of β_{s1} corresponding to the faulty sensor changes to zero while its corresponding diagonal value in β_{s2} changes to 1. This mechanism is used to ensure that controlled variable feedback signals, whether measured or inferred are always available for control purposes thereby maintaining the integrity of the whole control system. The " y'_p " obtained in equation 3.18 is the vector containing the feedback signals for the controlled outputs.

3.4 Fault Tolerant Control System Design for Actuator and Sensor Faults

The complete FTCS that is capable of accommodating sensor and actuator faults is presented in Figure 3.4. It combines the capabilities of both the actuator FTC and the

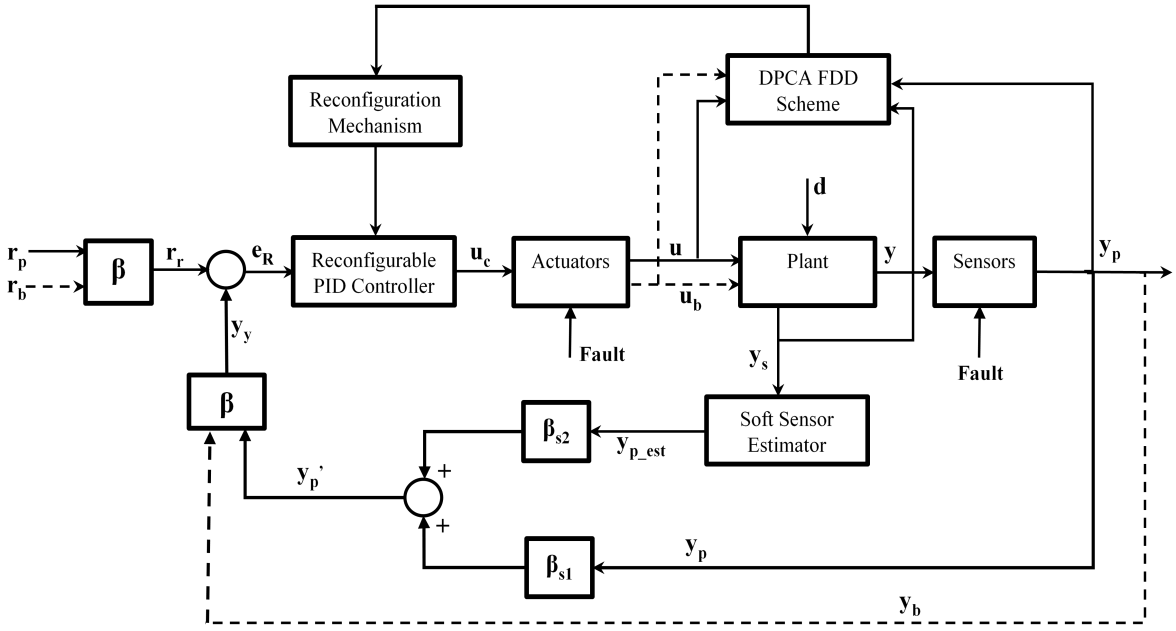


Figure 3.4: FTCS for combined actuator and sensor faults

sensor FTC. Using Figure 3.4, the controlled outputs feedback signal for the FTCS (y_y) in equation 3.11 now becomes:

$$y_y = \beta \begin{bmatrix} y_p'^T & y_b^T \end{bmatrix}^T = \beta \begin{bmatrix} (diag(\beta_{s1}\ y_p + \beta_{s2}\ y_{p\_est}))^T & y_b^T \end{bmatrix}^T \tag{3.20}$$

Hence, error generated by the FTCS in equation 3.10 is given as:

$$e_R = \beta(\begin{bmatrix} r_p^T & r_b^T \end{bmatrix}^T - \begin{bmatrix} (diag(\beta_{s1}\ y_p + \beta_{s2}\ y_{p\_est}))^T & y_b^T \end{bmatrix}^T) \tag{3.21}$$

Structure of the control law "u" presented in equation 3.13 remains unchanged and it is capable of tolerating both actuator and sensor fault. It is assumed that only one actuator fault can occur at any given time. However, the FTCS is able to accommodate sensor and actuator faults occurring successively. The DPCA FDD scheme is designed to detect and diagnose both actuator and sensor faults, and depending on the fault identified, the FTCS implements appropriate scheme using the reconfiguration mechanism part of the system. If an actuator fault is declared, the identified actuator fault is isolated and an appropriate control restructuring is undertaken to select a new set of controlled variable – manipulated variable pairing for the entire system, thereby reducing the dimension of the control structure by one. The predefined controller settings are then implemented while the FDD scheme continues to monitor the process post-fault era to guarantee its safe operation. If

the detected fault is a sensor fault, the feedback control loop involving the faulty sensor will not be functional and the fault-tolerant inferential control is implemented, by-passing the faulty sensor.

3.5 Summary

We have undertaken in this chapter, the design of simple restructurable feedback controller using backup feedback signals, switchable reference points, restructurable PID controllers and redundant controlled variable estimates to accommodate sensor and actuator faults. The FDD scheme that monitors the system was developed using DPCA data-based approach to detect and identify actuator and sensor faults. RGA and DRGA tools are used to analyse different possible control structures and stability of the system. We have also explained the mechanism through which controller reconfiguration is achieved including how the soft sensor estimates of the controlled variables are obtained. Procedures involved in the development and implementation of actuator FTC and sensor FTC were explained with clarity. Lastly, how the integrated FTCS functions as a complete system to accommodate actuator and sensor faults was described.

Chapter 4

Implementation of the Proposed FTCS for Actuator Faults Accommodation on Distillation Columns

4.1 Introduction

Design and implementation of FTCS for the oil and gas industry, particularly the distillation process is the main focus of this thesis. The distillation column is among the most common and energy intensive units in any refinery operation. It is fundamental to the chemical and process industries, which is why its dynamics and control has been studied extensively. Implementation of the actuator faults tolerant control system proposed in Chapter 3 on three different distillation processes with varying degrees of complexities under normal operation and faulty circumstances is presented in this chapter. This is to demonstrate the flexibility of the approach under various actuator faults. The developed FTCS for actuator fault, as described in Section 3.2, is first implemented on a binary distillation column with two primary control loops (Lawal and Zhang, 2015), and then on a fractionator, the Shell heavy oil fractionator with three primary control loops and four measured secondary variables (Lawal and Zhang, 2017b). Lastly, the control system is implemented on a crude distillation unit with several interactive primary control loops and numerous indirectly controlled secondary variables.

4.2 Application to a Binary Distillation Column

Figure 4.1 presents the binary distillation column, the first of the three case studies on which the actuator FTC is to be implemented.

4.2.1 Process Description and Control Strategies Prior Assessments

The binary distillation column studied here is a comprehensive nonlinear simulation of a methanol-water separation column. The column has ten theoretical stages and is used to separate feed with 50% weight fraction methanol into methanol with 95% weight fraction and water containing 5% weight fraction of methanol as the distillate and bottom products respectively. Disturbances in the system are changes in feed flow rate and feed compositions. A nonlinear stage-by-stage dynamic model of the column has been developed using mass and energy balances. The simulation has been validated against pilot plant test and is well known for its use in control system performance studies (Tham *et al.*, 1991a; Tham *et al.*, 1991b; Zhang and Agustriyanto, 2003). The assumptions imposed on the column model include negligible vapour hold-up, perfect mixing in each stage, and constant liquid hold-up. Table 4.1 presents a summary of the column steady-state conditions. There are five control loops in the column, as presented in Figure 4.1 – column pressure, condenser level, reboiler level, top and bottom composition control loops. However, only the top and bottom control loops are considered in the simulation as it is assumed that the column has perfect pressure control, and so do levels in the condenser and reboiler. Levels in the condenser and reboiler are controlled by the top and bottom flow rates respectively.

Table 4.1: Nominal column operating data

Column Parameters	Values
No of theoretical stages	10
Feed tray	5
Feed composition (Z)	50% methanol
Feed flowrate (F)	18.23 g/s
Top composition (Y_D)	0.95 (weight fraction)
Bottom composition (X_B)	0.05 (weight fraction)
Top product flow rate (D)	9.13 g/s
Bottom product flow rate (B)	9.1 g/s
Reflux flow rate (L)	10.11 g/s
Steam flow rate (V)	13.81 g/s

The column is controlled using reflux-vapour (LV) control strategy where the top com-

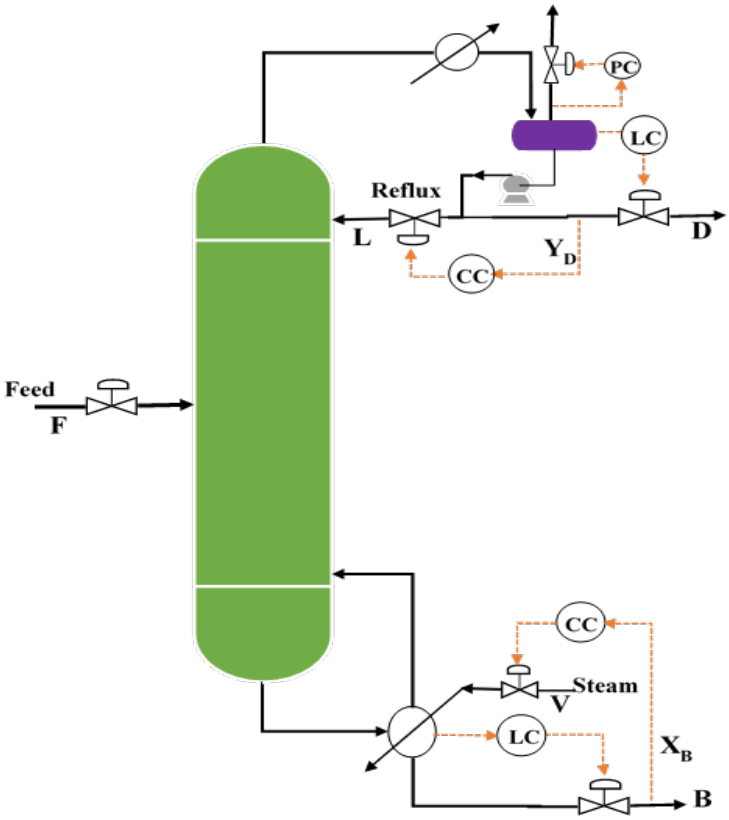


Figure 4.1: Binary distillation column

position (Y_D) is controlled by the reflux flow rate (L) and the bottom composition (X_B) by the steam flow rate (V) to the reboiler. The column has a 2 by 2 control structure with the process model given as (Zhang and Agustriyanto, 2003):

$$\begin{bmatrix} Y_D(s) \\ X_B(s) \end{bmatrix} = \begin{bmatrix} \frac{1.09e^{-5s}}{5.51s+1} & \frac{-1.30e^{-5s}}{13.72s+1} \\ \frac{2.27e^{-5s}}{17.15s+1} & \frac{-7.18e^{-5s}}{29.50s+1} \end{bmatrix} \begin{bmatrix} L(s) \\ V(s) \end{bmatrix} \tag{4.1}$$

Only RGA is used to assess the column control structure. Using equation 3.8 the elements of the RGA is obtained as:

$$\Lambda = \begin{bmatrix} 1.6053 & -0.6053 \\ -0.6053 & 1.6053 \end{bmatrix} \tag{4.2}$$

Elements of the RGA confirm the input-output pairing of the binary distillation column. A fault in one actuator will leave the other as the only healthy actuator that could be used to control either output. These possibilities are explored during the simulation, and the column PI controller settings during normal and faulty conditions are presented in Table 4.2.

Table 4.2: Distillation column controller settings

	Controller parameters			
	PI loop 1		PI loop 2	
	K_{p1}	T_{i1}	K_{p2}	T_{i2}
Normal operation	45	18.67	-20	18.4
Reflux actuator fault acc.	–	–	-101	13.5
Steam actuator fault acc.	70.7	21	–	–

4.2.2 Process Simulation under Normal and Faulty Conditions

The column was simulated in MATLAB for 1300 minutes with 30 second sampling time to give a total of 2600 data points for normal process conditions. The top and bottom product compositions are measured by composition analysers with 10 sampling time delay (5 minutes). A combination of 10% and 5% changes in feed flow rate and feed composition were randomly introduced into the system during simulation for both normal and faulty conditions. After collection of data for normal process condition, low and high magnitude faults were also introduced into the system at different times to simulate fault cases. This was achieved by restricting the flow of reflux and steam rates to represent stuck valves, thereby acting as actuator faults. Seven actuator faults were investigated for the reflux and steam actuators, and Figure 4.2 presents the column with faulty actuators. Details of the faults are presented in Table 4.4. Four low magnitude actuator faults ($F1$, $F3$, $F6$, and $F7$) were investigated, with values of the manipulated variables held close to their respective steady state values. The first 2 low magnitude faults ($F1$ and $F3$) were investigated for low magnitude fault detectability while the last 2 low magnitude faults ($F6$ and $F7$) were introduced to investigate effects of disturbances on low magnitude faults propagation and detectability. Also, two high magnitude actuator faults ($F2$ and $F4$) and a combination of the two high magnitude actuator faults ($F5$) were considered. The fault cases were each simulated for 750 minutes to collect 1500 samples. A total of 14 variables are monitored during simulation: the top and bottom product compositions, the manipulated variables – reflux and steam flowrates, and the ten tray temperatures, as presented in Table 4.3.

4.2.3 Actuator Faults Diagnostic Model Development and FDD

Random noises with zero means and 0.15 and 0.001 standard deviations are first added to the ten tray temperatures and the top and bottom compositions respectively, to represent

Table 4.3: Variables for the binary distillation column

Variables	Variable description
Variable 1	Top composition [weight fraction]
Variable 2	Bottom composition [weight fraction]
Variable 3	Reflux flow rate $[g/s]$
Variable 4	Steam flow rate $[g/s]$
Variable 5	Stage 10 temperature $[^oC]$
Variable 6	Stage 9 temperature $[^oC]$
Variable 7	Stage 8 temperature $[^oC]$
Variable 8	Stage 7 temperature $[^oC]$
Variable 9	Stage 6 temperature $[^oC]$
Variable 10	Stage 5 temperature $[^oC]$
Variable 11	Stage 4 temperature $[^oC]$
Variable 12	Stage 3 temperature $[^oC]$
Variable 13	Stage 2 temperature $[^oC]$
Variable 14	Stage 1 temperature $[^oC]$

Table 4.4: Distillation column fault list

Fault	Fault description
F1	Reflux valve stuck between 7-10 g/s after sample 750
F2	Reflux valve stuck between 5- 8 g/s after sample 750
F3	Steam valve stuck between 10-14 g/s after sample 750
F4	Steam valve stuck between 10-13 g/s after sample 750
F5	Reflux and steam valves stuck @ 8 g/s and 13 g/s after samples 750 and 1150 respectively
F6	F1 repeated with feed flow rate disturbance introduced after sample 900
F7	F3 repeated with feed flow rate disturbance introduced after sample 900

true measurements of the data collected during normal operating conditions in Section 4.2.2. Figure 4.3 presents the top and bottom compositions, their respective manipulated variables and the ten tray temperatures for normal operating conditions.

A DPCA diagnostic model with one time-lagged measurements was developed using the procedures described in Section 2.4.2 with the first 1600 samples out of the 2600 collected under normal operating conditions while the last 1000 samples were used for validation. The data was scaled to zero mean and unit variance to obtain matrix X in equation 3.1 where u includes reflux and steam flow rates, y_p includes the top and bottom compositions, and y_s comprises of the ten tray temperatures. Four principal components account for 82.17% variations in the original data and are sufficient to develop the DPCA diagnostic model. The diagnostic model developed for the fault free system is applied to the seven faulty data sets to detect faults. Figure 4.4 presents the T^2 and SPE monitoring plots for the fault-free system while Figure 4.5 and Figure 4.6 show those of the seven

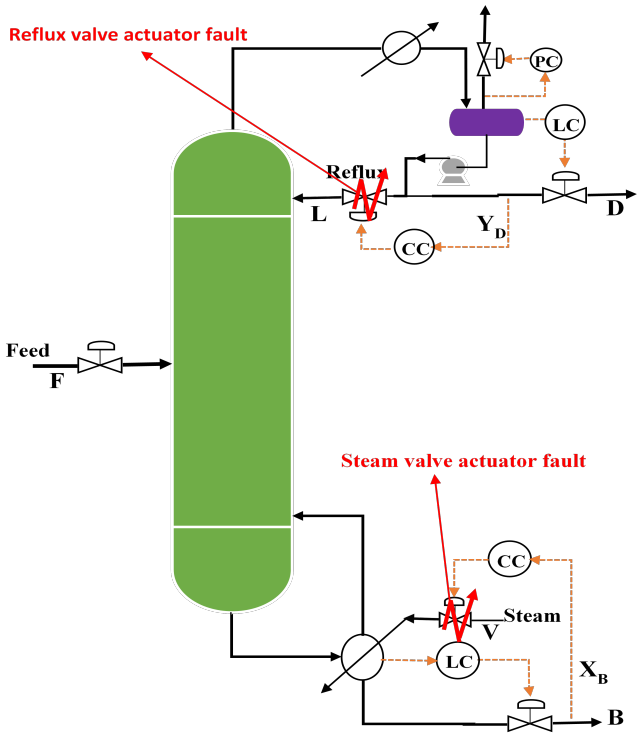


Figure 4.2: Binary distillation column with faulty actuators

fault cases ($F1$ – $F7$). The red line in Figures 4.4 – 4.6 is the 99% control limit. A fault is declared after the control limits are violated for four successive sampling times to reduce the occurrence of false alarms. Once the presence of an actuator fault is detected, further fault identification analysis is carried out through contribution plots as described in Section 3.2.1 to identify variables that are responsible for the fault, and ultimately isolate the fault. Figures 4.7 and 4.8 present the excess contribution plots for the fault cases detected, with the blue and yellow bars representing excess contributions of each variable at the point of fault declaration and the next sampling time respectively.

4.2.4 Implementation of Actuator FTC on Identified Actuator Faults

The proposed reconfigurable FTC scheme is implemented on the distillation column upon detection of an actuator fault. The two actuators investigated in the system are reflux flow and steam flow actuators with varying degrees of faults as presented in Table 4.4. The error signals generated by the system during normal operation for the top and bottom composition controllers (G_c) in equation 3.13 is:

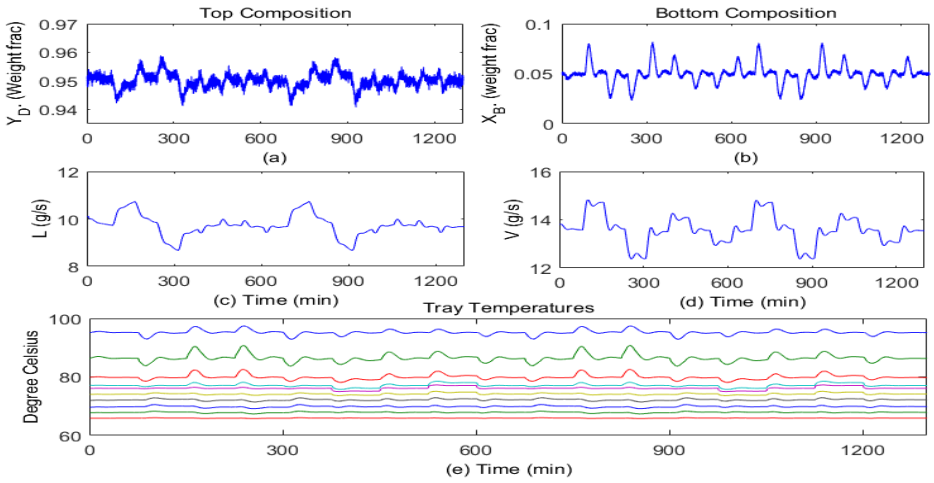


Figure 4.3: Top and bottom compositions, and the column temperature measurements

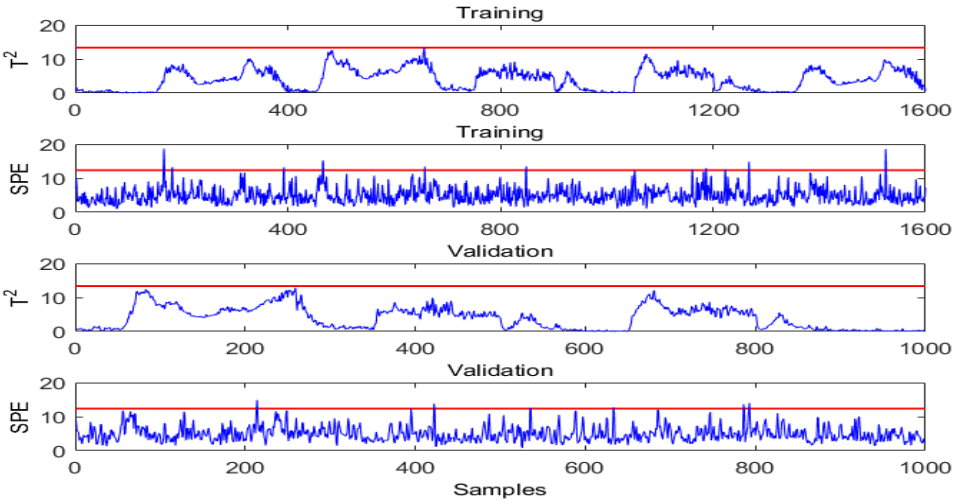


Figure 4.4: T^2 and SPE monitoring plots for the fault-free system

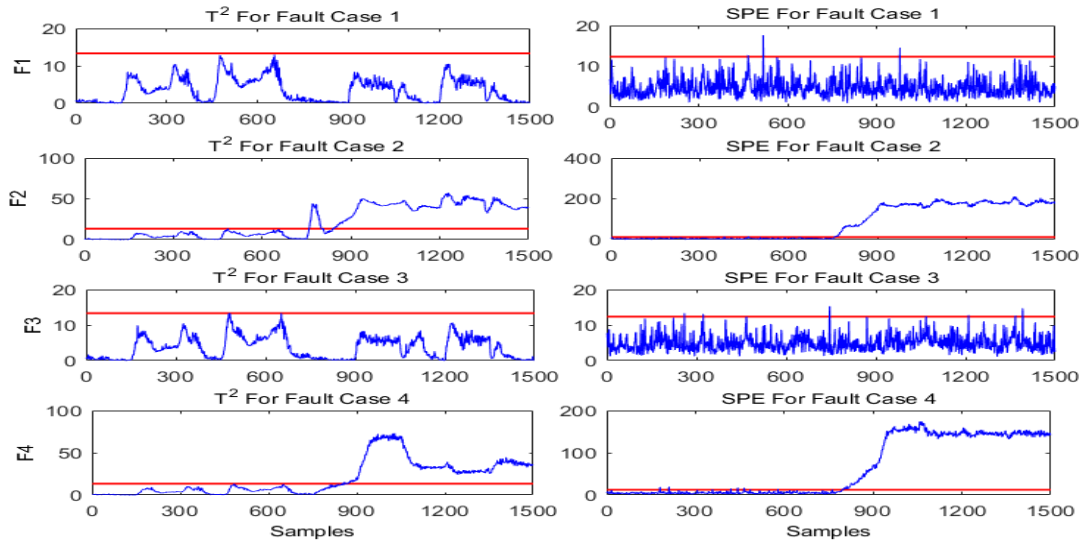


Figure 4.5: T^2 and SPE monitoring plots for fault cases 1 –4

Figure 4.6: T^2 and SPE monitoring plots for fault cases 5 –7

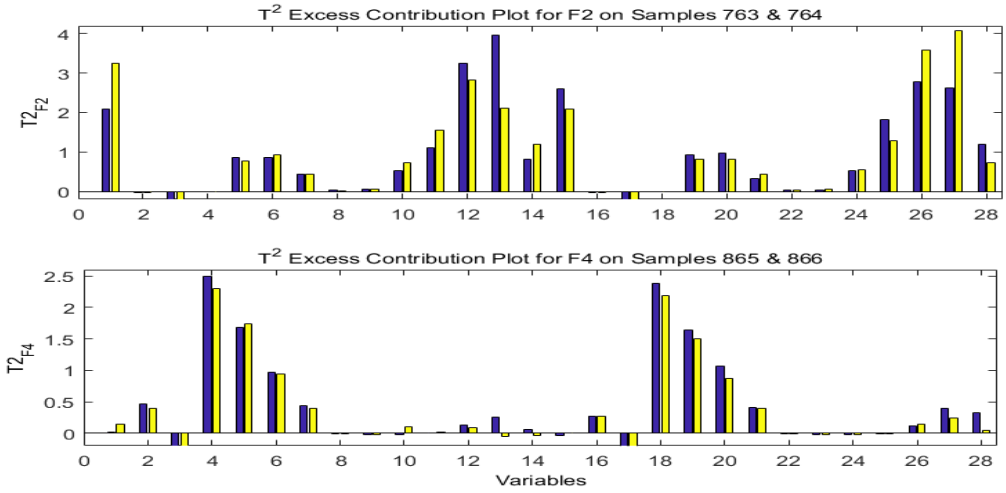


Figure 4.7: Excess Contribution plots for fault cases F2 and F4

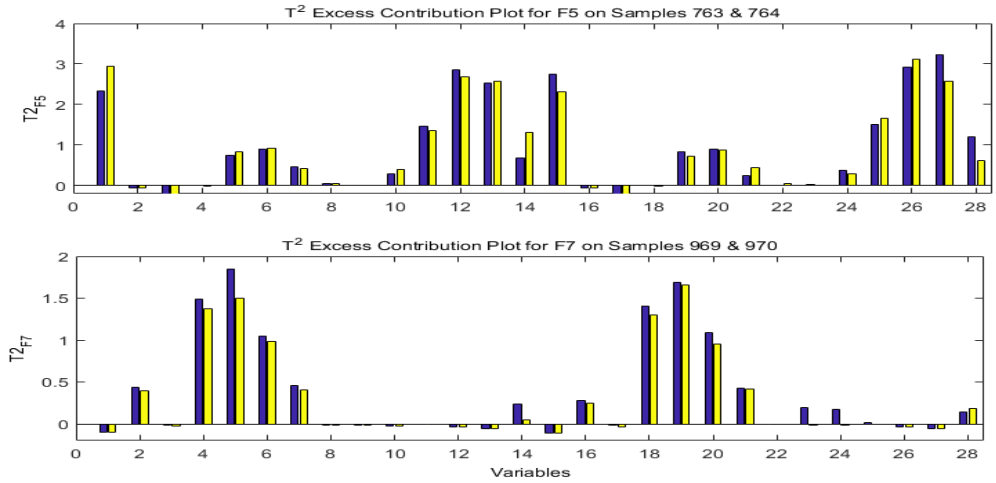


Figure 4.8: Excess Contribution plots for fault cases F5 and F7

$$e_R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{b1} \\ r_{b2} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{b1} \\ y_{b2} \end{bmatrix} \tag{4.3}$$

where r_{p1}, r_{p2}, y_{p1}, and y_{p2} are the reference points and outputs of the primary controlled variables while r_{b1}, r_{b2}, y_{b1}, and y_{b2} are the backup feedback signals for reference points and the primary controlled outputs respectively. It can be observed that weightings for the backup signals are zeros. When an actuator fault is detected and subsequently identified, for instance, the reflux flow actuator, the reconfigurable controller is then activated using the only healthy actuator, in this case the steam valve actuator provided the top composition is deemed more valuable. Equation 4.3 then becomes:

$$e_R = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{b1} \\ r_{b2} \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{b1} \\ y_{b2} \end{bmatrix} \tag{4.4}$$

and the control law for accommodating the reflux valve actuator fault is obtained using equation 3.16 as:

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_{b1} \\ u_{b2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & G_{b1} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ r_{b1} - y_{b1} \\ 0 \end{bmatrix} \tag{4.5}$$

Equation 4.5 presents the reconfigured controller that accommodates the reflux valve actuator fault using the steam valve actuator while the bottom composition is left uncontrolled. The procedure is the same if the steam valve actuator fault is declared.

4.2.5 Results and Discussions

Figures 4.5 and 4.6 show the T^2 and SPE plots for faults $F1$ to $F4$ and $F5$ to $F7$ respectively. Figure 4.7 shows the contribution plots for $F2$ and $F4$ while Figure 4.8 presents those of $F5$ and $F7$. Figures 4.7 and 4.8 present the excess contributions of each variable to the larger than normal value of T^2 at the point of fault declaration. By excess

contributions, we mean the difference between contributions of each variable to the values of T^2 at the point of fault declaration and their respective average contribution under fault free conditions. From the analysis of T^2 and SPE plots shown in Figures 4.5 and 4.6, the monitoring statistics for faults $F2, F4, F5$ and $F7$ exceeded their control limits at different times during the simulation, so these faults were detected. Faults $F2$ and $F5$ were detected 13 sampling times (6 mins 30 sec.) after introduction, at sample 763, while it took 115 sampling time (approximately 58 mins), at sample 865 for fault effect to manifest in $F4$ as presented in Figure 4.5. Fault 7 ($F7$) was detected at sample 969, approximately 110 minutes after it was introduced as shown in Figure 4.6. Note that faults $F3$ and $F7$ are the same with the exception of disturbances introduced after the faults to investigate the effects of the disturbances on the faults propagation and detectability. A 10% increase in feed composition disturbance was introduced at sample 900 in the case of $F3$ while the same magnitude of disturbance in feed flow rate was introduced in the case of $F7$, also at sample 900.

Basically, disturbances do affect fault propagation and detection in the column and could amplify a rather minor undetected fault as shown in the case of $F7$ which was detected 69 sampling times (approx. 35 minutes) after the disturbance was introduced. Further analyses were conducted to identify the faults using contribution plots upon declaration of a fault. The T^2 contribution plots gave a more consistent indication of the variables responsible for the faults; hence only T^2 contribution plots are used for fault identification in this instance. In the case of $F2$ and $F5$ where reflux actuation faults were identified, the contribution plot as shown in Figure 4.7 identified the top composition and the top four tray temperatures (variables 1, 11, 12, 13, 14, 25, 26, 27 and 28) as the major contributors to the out-of-control situation. Analysis of the T^2 contribution plots presented in Figures 4.7 and 4.8 combined with the process knowledge aided the fault identification. For instance, when the reflux actuator fault occurred (stuck reflux valve), and after it was detected, the contribution plot isolates variables indicative of the fault. The detected reflux actuator fault with reduced reflux flow caused the top tray temperature measurements to rise significantly which ultimately led to the top composition drifting out of control. Hence, the contributions of these variables (top composition and the top tray temperatures) to the T^2 monitoring statistics increased significantly as presented in Figures 4.7. The rise in the top tray temperature measurements as a consequence of reduced reflux flow is peculiar to the reflux actuation fault, which aided its isolation. Similarly, observing contribution plots for $F4$ and $F7$ as presented in Figures

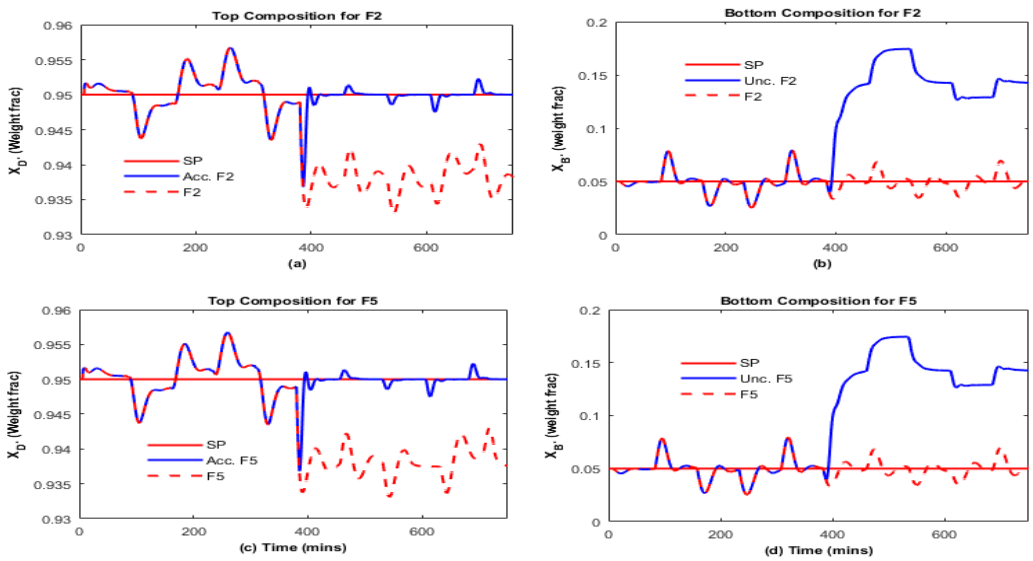


Figure 4.9: Responses of the top and bottom compositions to F2 and F5 reflux actuator faults accommodation

4.7 and 4.8, when the steam actuator fault occurred, the bottom composition drifted out of control which also affected the steam controller output and the bottom tray temperatures. These effects manifest in the larger than average contributions of these variables to the T^2 values at the point of fault declaration and beyond. This was the pattern exploited in the fault identifications as different faults show different variable contributions to the T^2 values after an occurrence of a fault.

As mentioned in Table 4.4, $F1, F2$ and $F6$ are all reflux actuator faults of different magnitudes; while $F3, F4$ and $F7$ are steam actuator faults, also of different magnitudes. $F5$ is a combination of reflux and steam actuator faults, with the reflux actuator fault occurring first. Observations from Figures 4.7 and 4.8 show different variable contribution patterns which aided fault isolation. Faults $F1, F3$ and $F6$ were not detected because only small changes were made to the values of the two actuators which were close to the nominal values of the two manipulated variables as shown in Table 4.1 under reflux and steam flow rates. The resulting values for the process variables were within normal operating conditions. Fault 6 ($F6$), a rather minor undetected fault in $F1$, was affected by the amplifying effect of the disturbance (increased in feed flow rate after sample 900) on its propagation which moved it to marginal stability.

Clearly, the conventional LV control strategy used for the binary column normal operation could not accommodate the actuator faults. Hence, the control strategy in the column is restructured by switching to a one-point control strategy where the only remaining healthy actuator, steam flow rate actuation in the case of $F2$ and $F5$ or reflux

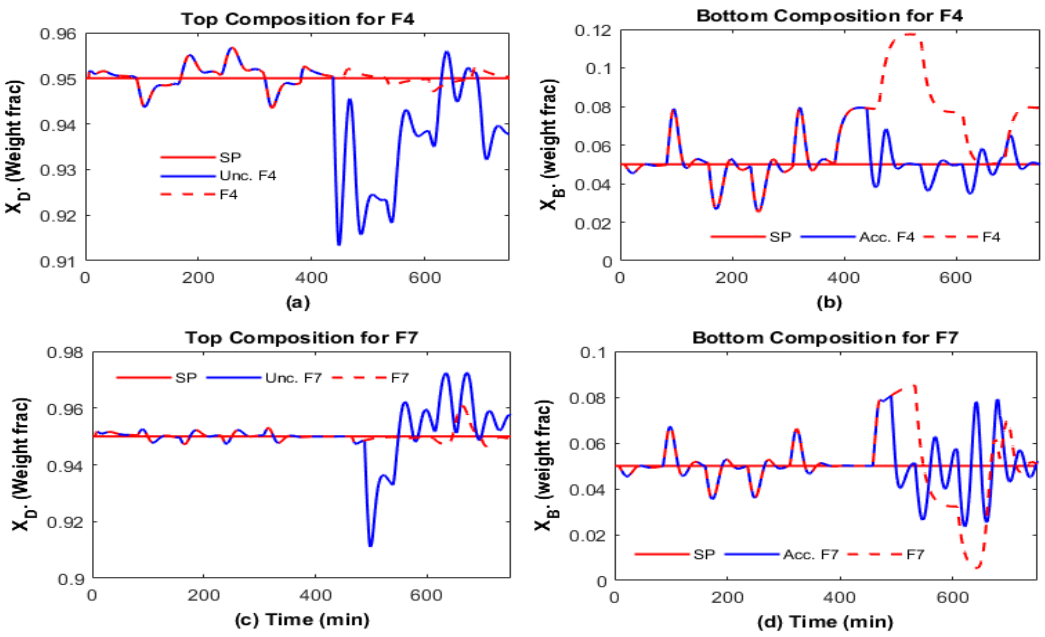


Figure 4.10: Responses of the top and bottom compositions to F4 and F7 steam actuator faults accommodation

flow rate actuation in the case of $F4$ and $F7$ is used to accommodate the faults and maintain the more valuable of the two compositions within an acceptable range while the other is uncontrolled. Steam flow actuation was immediately reconfigured and implemented to tolerate reflux valve faults, $F2$ and $F5$ by manipulating the steam flow rate to directly maintain the top composition at its set point thereby tolerating reflux valve actuator faults in $F2$ and $F5$ as presented in Figure 4.9. Table 4.2 presents the reconfigurable PI controller settings for the column under normal and faulty conditions. Similarly, upon detection of steam flow actuator faults, $F4$ and $F7$, the column control structure was reconfigured and immediately switched to reflux valve one-point control by manipulating reflux flow rate to directly maintain the bottom composition at set point if the bottom composition is deemed more important, thereby tolerating the steam flow actuation faults $F4$ and $F7$ as shown in Figure 4.10. It is worth mentioning at this point that, the fault accommodation approach implemented in this case is sub-optimal as it is practically impossible to use one manipulated variable to maintain both top and bottom compositions at their set points. The sub-optimal fault accommodation approach provides desirable performance and will be far more acceptable than shut-down. SP, Unc. and Acc. are used in Figures 4.9 and 4.10 to represent set point, uncontrolled fault and accommodated fault respectively. The effects of the disturbances, feed flow rates and the feed compositions after the faults were well compensated for by the actuator FTC scheme as can be observed in Figures 4.9 and 4.10.

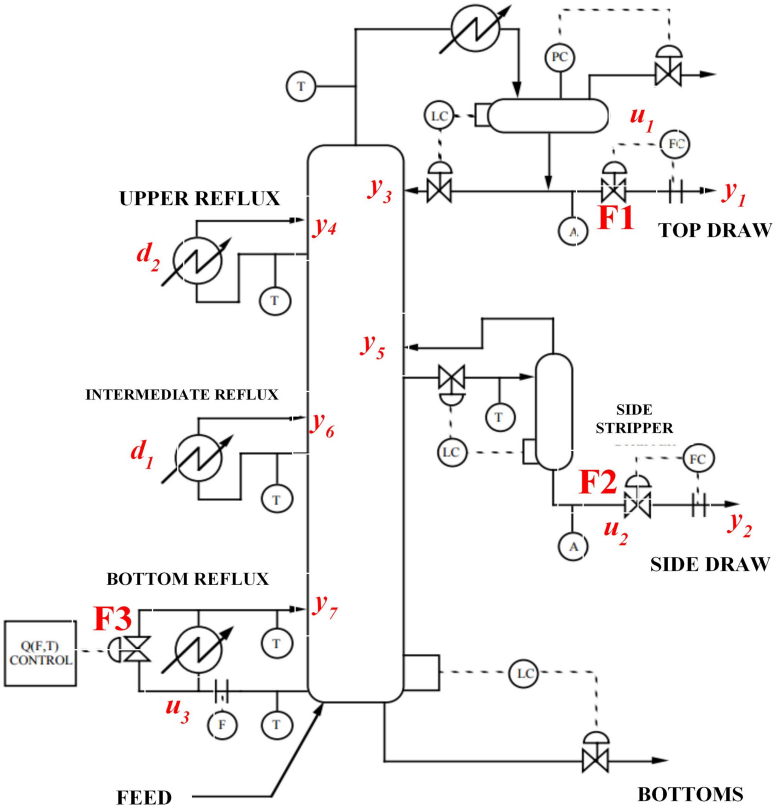


Figure 4.11: Shell heavy oil fractionator

4.3 Application to the Shell Heavy Oil Fractionator

The proposed actuator FTC scheme is applied to the Shell heavy oil fractionator in this section. It is the second case study to be considered under the proposed actuator FTC. Figure 4.11 presents the schematic diagram of the system. The system is relatively more complex than the binary distillation column considered in the previous section. There are more interactions amongst the control loops.

4.3.1 Process Description and Control Loop Pairing

The Shell heavy oil fractionator benchmark used here was developed by Shell Company as a test bed for the assessment of new control theories and technologies in 1986 (Prett and Morari., 1987; Vlachos *et al.*, 2002). It is a highly constrained multivariable process with large dead times and very strong interactions amongst its control loops. The original system is slightly modified in this study by relaxing some of its constraints for the purpose of actuator faults accommodation. The heavy oil fractionator has five inputs and seven outputs, and it provides a realistic test bed for control related studies. The process was modelled using a first-order plus dead time transfer function matrix. Three out

Table 4.5: Variables for heavy oil fractionator

Variables	Output variables
Variable 1	Top end point (y_1)
Variable 2	Side end point (y_2)
Variable 3	Top temperature (y_3)
Variable 4	Upper reflux temperature (y_4)
Variable 5	Side draw temperature (y_5)
Variable 6	Inter. reflux temperature (y_6)
Variable 7	Bottom reflux temperature (y_7)
	Input variables
Variable 8	Top draw (u_1)
Variable 9	Side draw (u_2)
Variable 10	Bottom reflux duty (u_3)
	Disturbance variables
	Inter. reflux duty (d_1)
	Upper reflux duty (d_2)

of the 5 inputs (top draw – u_1, side draw – u_2 and bottom reflux duty – u_3) into the system are used as manipulated variables, directly maintaining 3 process outputs (top end point – y_1, side end point – y_2 and bottom reflux temperature – y_7) at their set points while the remaining 2 inputs – intermediate reflux duty (d_1) and upper reflux duty (d_2) serve as unmeasured disturbances into the system. The other four outputs are not controlled. Table 4.5 gives the full listing of all the system variables. The manipulated variables are subject to saturation (± 0.5) and rate limit ($\pm$ 0.05 per sample time) actuator hard constraints, which introduce non-linearity into the system. The disturbances are bounded within absolute values not more than 0.5. The complete model of the system is given in Table 4.6 while Figure 4.12 presents the system with different back-up feedback signals (indicated by dashed lines) for possible implementation of actuator fault tolerant controller. The system is controlled using three reconfigurable PI controllers with integral anti-windup.

The input-output selection for the control configuration was achieved after careful analysis of the system coupled with the use of RGA analysis as described in Section 3.2.2.1. The transfer function matrix of the system given in equation 4.6 is used to obtain the steady state RGA for the system as shown in equation 4.7. Based on the RGA values, the manipulated variables u_1, u_2 and u_3 are used to control y_1, y_2 and y_7 respectively under normal operating conditions, producing a 3×3 control configuration. Possible controller reconfigurations are pre-assessed using the RGA tool for the input-output pairings under different faulty conditions.

Table 4.6: Shell heavy oil fractionator transfer function model parameters

	Top draw (u_1)			Side draw (u_2)			Bot. reflux duty (u_3)			Int. reflux duty (d_1)			Upper reflux duty (d_2)		
	K	τ	θ	K	τ	θ	K	τ	θ	K	τ	θ	K	τ	θ
Top end point (y_1)	4.05	50	27	1.77	60	28	5.88	50	27	1.20	45	27	1.44	40	27
Side end point (y_2)	5.39	50	18	5.72	60	14	6.90	40	15	1.52	25	15	1.83	20	15
Top temperature (y_3)	3.66	9	2	1.65	30	20	5.53	40	2	1.16	11	0	1.27	6	0
Upper reflux temp. (y_4)	5.92	12	11	2.54	27	12	8.10	20	2	1.73	5	0	1.79	19	0
Side draw temp. (y_5)	4.13	8	5	2.38	19	7	6.23	10	2	1.31	2	0	1.26	22	0
Inter. reflux temp. (y_6)	4.06	13	8	4.18	33	4	6.53	9	1	1.19	19	0	1.17	24	0
Bottom reflux temp. (y_7)	4.38	33	20	4.42	44	22	7.20	19	0	1.14	27	0	1.26	32	0

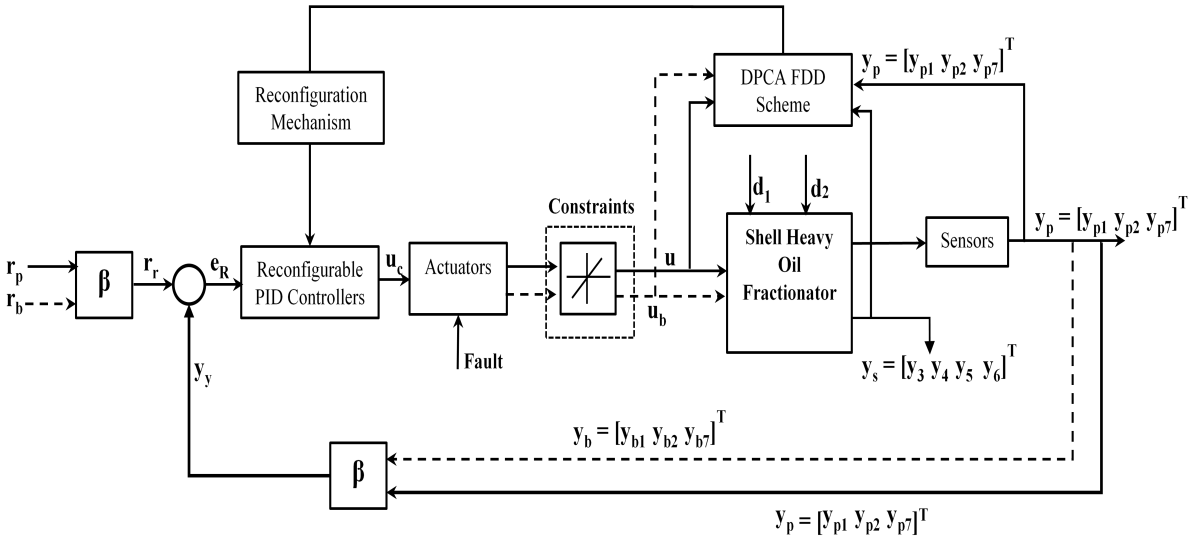


Figure 4.12: Schematic of the Shell heavy oil fractionator integrated with FTCS

$$G(s) = \begin{bmatrix} \frac{4.05e^{-27s}}{50s+1} & \frac{1.77e^{-28s}}{60s+1} & \frac{5.88e^{-27s}}{50s+1} \\ \frac{5.39e^{-18s}}{50s+1} & \frac{5.72e^{-14s}}{60s+1} & \frac{6.90e^{-15s}}{40s+1} \\ \frac{4.38e^{-20s}}{33s+1} & \frac{4.42e^{-22s}}{44s+1} & \frac{7.20}{19s+1} \end{bmatrix} \tag{4.6}$$

$$\Lambda = \begin{bmatrix} 2.0757 & -0.7289 & -0.3468 \\ 3.4242 & 0.9343 & -3.3585 \\ -4.4999 & 0.7946 & 4.7053 \end{bmatrix} \tag{4.7}$$

The input-output pairing for controller reconfiguration of the three actuator faults, $F1$ – top draw actuator fault; $F2$ – side draw actuator fault; and $F3$ – bottom reflux duty actuator fault investigated in this case study is also determined. When a fault is declared and identified, for instance top draw actuator fault ($F1$), we are left with just two healthy

Table 4.7: Controlled and manipulated variables pairing

	Manipulated Inputs			
Controlled Outputs	Normal	F1	F2	F3
Top end point (y_1)	u_1	u_3	u_3	–
Side end point (y_2)	u_2	u_2	u_1	u_2
Bot. Reflux Temp. (y_7)	u_3	–	–	u_1

Table 4.8: Shell heavy oil fractionator reconfigurable PI controller settings

	Controller parameters							
	Normal		F1		F2		F3	
Controlled Output Loop	K_p	T_i	K_p	T_i	K_p	T_i	K_p	T_i
Top end point	0.05	0.0215	0.2	0.004	0.21	0.005	–	–
Side end point	0.45	0.0160	0.45	0.016	0.20	0.001	0.45	0.016
Bot. Reflux Temp.	3	0.005	–	–	–	–	1	0.020

actuators, side draw and bottom reflux duty actuators (u_2 and u_3) to maintain three outputs at set points. This is unrealizable using the conventional PID control strategy. Therefore only two outputs are controlled directly while the third is uncontrolled. We have chosen the top draw and the side draw end points (y_1 and y_2) as the outputs to control after actuator fault ($F1$ – u_1) was declared by appropriately reconfiguring the remaining healthy actuators. An example of the RGA matrix obtained under $F1$ is given in equation 4.8 and Table 4.7 presents the inputs-outputs pairing for the three fault cases. Table 4.8 presents the PI controller settings for the reconfigured controllers under normal condition and each faulty actuator.

$$RGA_{F10} = \begin{bmatrix} -0.570 & 1.5702 \\ 1.5702 & -0.5702 \end{bmatrix} \tag{4.8}$$

where Λ_{F1} is the RGA for $F1$.

4.3.2 Process Simulation under Fault-Free and Faulty Conditions

The heavy oil fractionator was simulated without actuator faults in Simulink for 2000 minutes with 1 minute sampling time as shown in Figure 4.13 to collect 2000 samples of the seven outputs and three manipulated variables. Intermediate reflux duty (d_1) and upper reflux duty (d_2) serve as disturbances and were randomly introduced into the system during normal process operation. Gaussian noise of zero mean and 0.003 standard

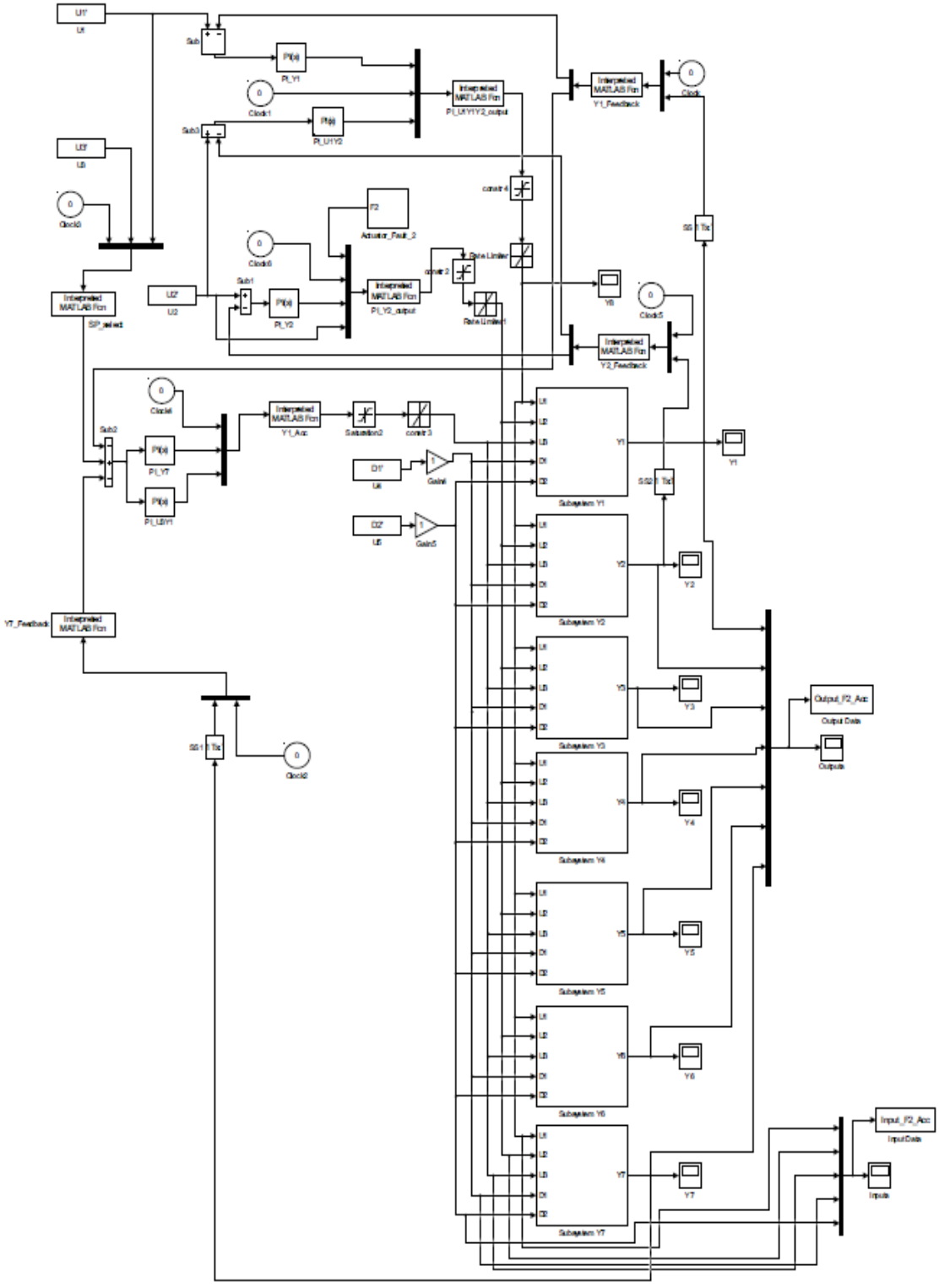


Figure 4.13: Heavy oil fractionator Simulink model

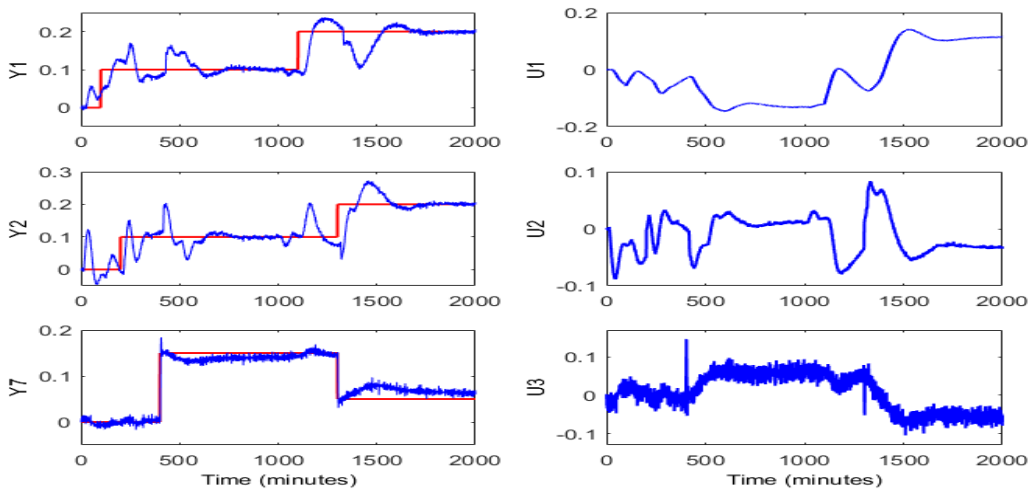


Figure 4.14: Input and output responses to set-point changes and disturbance

Table 4.9: Heavy oil fractionator fault list

Fault	Fault description
F1	Top draw actuator fault – control valve stuck at 0.5
F2	Side draw actuator fault – control valve stuck at 0.5
F3	Bottom reflux duty actuator fault – control valve stuck at 0.5

deviation were added to each of the 7 outputs to represent true measurements of the data collected. Figure 4.14 presents the system actuator outputs under normal operating conditions and their respective outputs responses to changes in set-points and introduction of disturbances. Details of the three actuator fault cases ($F1, F2$ and $F3$), one each for the three actuators (u_1, u_2 and u_3) are presented in Table 4.9. The fault was introduced in each case at 800 minutes as a constant value of 0.5 (i.e. control valve stuck to 0.5). The fault cases were each simulated for 2000 minutes to collect 2000 samples.

4.3.3 Actuator Fault Detection and Diagnosis

Precisely the same procedure applied in the methanol-water separation column to detect and diagnose actuator faults was used here. 1100 samples of the 2000 samples collected during normal operating conditions were used to develop the DPCA diagnostic model with one time lag while the remaining 900 samples were used for validation. The training data set was scaled to zero mean and unit variance. Three principal components which account for 86.95% variation ($a = 3$) in the original data are used to develop the DPCA diagnostic model for process monitoring and actuator FDD. Figure 4.15 shows the process

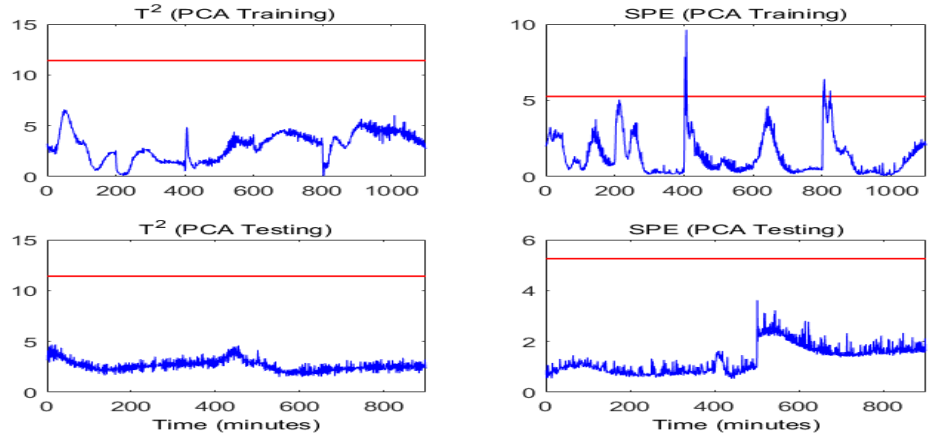


Figure 4.15: T^2 and SPE monitoring plots for training and testing data

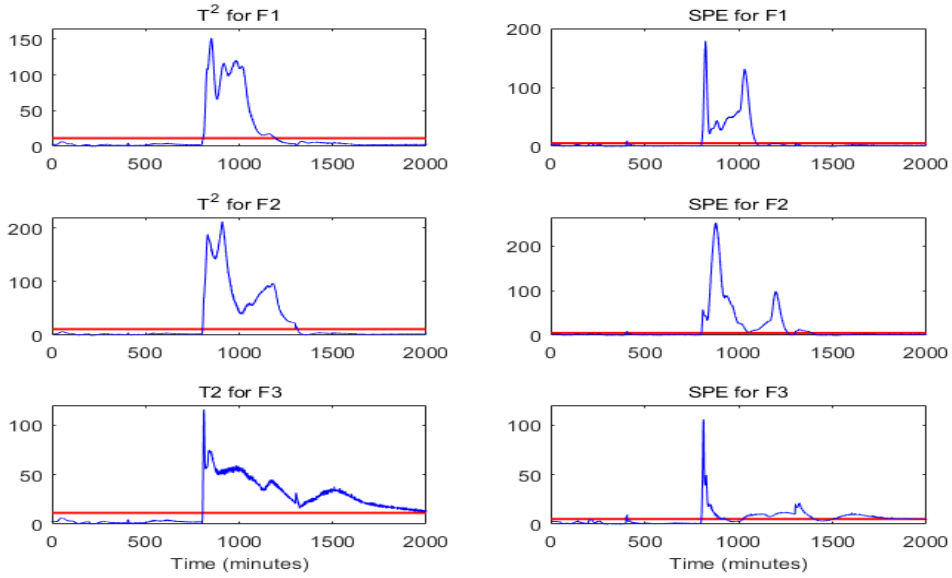


Figure 4.16: T^2 and SPE monitoring plots for faults F1 – F3

monitoring performance indices for the training and testing data sets. The developed diagnostic model is then applied to the three faulty actuator cases in the Shell heavy oil fractionator to detect possible fault occurrences. A fault is declared when the monitoring indices, T^2 and SPE violate their respective limits for four consecutive sampling times to ensure no false alarm is recorded. Figure 4.16 presents the Hotelling's T^2 and SPE process monitoring performance for the three faults ($F1$ – $F3$). After a fault is declared, its root cause is further investigated through contribution plots which provide information on the contribution of each variable to the faulty scenario thereby aiding its isolation.

4.3.4 Implementation of FTC on Identified Actuator Faults

When there is an actuator fault, the 3 by 3 control configuration used for normal process operation will have to be restructured, settings of the reconfigurable controllers retuned and the set points switched as appropriate upon detection and isolation of an actuator fault in order to maintain the integrity of the system. This is achieved through the feedback and set points backup signals as shown in Figure 4.12. The different pre-assessed input-output pairings presented in Table 4.7 and the appropriate reconfigured controller settings presented in Table 4.8 are implemented, depending on the fault identified. The error vector generated for the reconfigurable controller for the system during normal operation is obtained as equation 4.9 using equation 3.13.

$$e_R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{b1} \\ r_{b2} \\ r_{b3} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{b1} \\ y_{b2} \\ y_{b3} \end{bmatrix} \tag{4.9}$$

where r_{p1}, r_{p2}, r_{p3} are the reference points for the system outputs; y_{p1}, y_{p1}, y_{p1} are the outputs; r_{b1}, r_{b2}, r_{b3} are the backup signals for reference point and y_{b1}, y_{b2}, y_{b3} are the corresponding outputs backup feedback signals. When the top draw actuator fault ($F1$) is declared and the fault tolerant controller reconfigured as appropriate, equation 4.10 is obtained.

$$e_R = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{b1} \\ r_{b2} \\ r_{b3} \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{b1} \\ y_{b2} \\ y_{b3} \end{bmatrix} \tag{4.10}$$

Then, the fault tolerant control law under $F1$ is given as

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_{b1} \\ u_{b2} \\ u_{b3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & G_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{b1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ r_{p2} - y_{p2} \\ 0 \\ r_{b1} - y_{b1} \\ 0 \\ 0 \end{bmatrix} \tag{4.11}$$

As shown in equation 4.10 and equation 4.11 above, the weightings for different signals are activated and deactivated as appropriate to accommodate the fault declared and sub-optimally maintain the system within acceptable operating region.

4.3.5 Results and Discussions

The three actuator faults investigated in this system – top draw actuator fault (F1), side draw actuator fault ($F2$) and the bottom reflux duty actuator faults ($F3$) were all detected. The DPCA diagnostic model monitoring statistics, T^2 and SPE detected the top draw actuator fault ($F1$) 11 minutes and 8 minutes respectively after its introduction as presented in Figure 4.16. Side draw reflux actuator fault ($F2$) violated the T^2 and SPE monitoring limits at 809 and 807 minutes respectively while bottom reflux duty actuator fault ($F3$) was detected at 808 and 806 minutes respectively. Hotelling's T^2 and SPE variable contribution plots are analysed at the point an actuator fault is detected to investigate the root cause of the fault. The variable contribution plots shown in Figure 4.17 present excess contributions of each variable to the average values of T^2 and SPE that led to the fault being declared. Top temperature (variable 3) and top draw (variable 8) contributed significantly to the fault, as identified by T^2 contribution plot. The SPE contribution plot shows side end point, top temperature, upper reflux temperature, side draw and bottom reflux duty (variables 2, 3, 4, 5, 8 and 10) as the major contributors to the faulty situation recorded. A critical analysis of the effect of top draw actuator fault ($F1$), depending on the magnitude of the fault shows a similar effect on the variables identified by the diagnostic model as being responsible for the fault.

Though the T^2 and SPE contribution plots give indications of the likely causes of the fault, however an understanding of the system is still required to make the connections between the fault detected and the variables identified by the isolation technique. The side draw actuator fault ($F2$) was caused by a significantly large value of side draw (variable

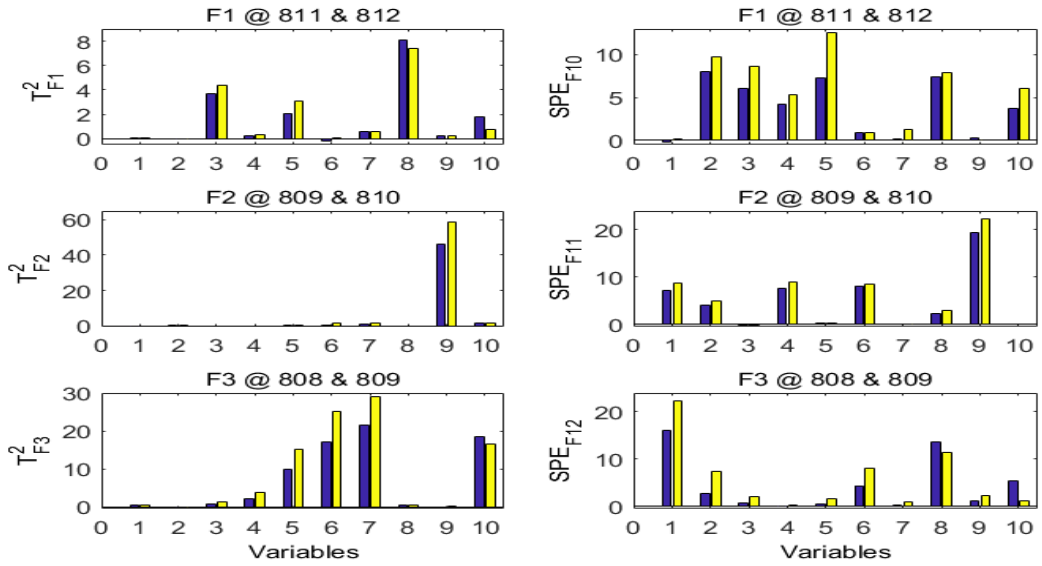


Figure 4.17: T^2 and SPE excess contribution plots for faults F1 – F3

9) which is the output of the faulty actuator as identified by the contribution plots. SPE contribution plots in addition to the faulty actuator output also show top end point, side end point, upper reflux temperature and intermediate reflux temperature (variables 1, 2, 4, 6 and 9) as the variables responsible for the fault, as presented in Figure 4.17. Similarly, side draw temperature, intermediate reflux temperature, bottom reflux temperature and bottom reflux duty actuator output (variables 5, 6, 7, and 10) are identified by the T^2 contribution plots as the variables responsible for the bottom reflux duty actuator fault ($F3$). SPE contribution plots indicate the top end point and the top draw actuator (variables 1 and 8) as the root causes of the fault. The pattern observed in this system for the bottom reflux duty actuator fault ($F3$) is similar to the one observed in the steam valve actuator fault for the methanol-water separation column.

After the fault is detected and isolated as either being top draw actuator fault (u_1), side draw actuator fault (u_2) or bottom reflux duty actuator fault (u_3), it has to be accommodated in order to stabilise the system and ensure its continued safe operation, at least sub-optimally. When top draw actuator fault (u_1) occurs, clearly the 3 by 3 control structure will not be functional and depending on the severity of the fault, one of the remaining two healthy actuators, side draw actuator (u_2) and the bottom reflux duty actuator (u_3) are reconfigured to control the top end point (y_1) as presented in Table 4.7. Reflux duty actuator (u_3) is reconfigured to maintain the top end point at set point, leaving the bottom reflux temperature (y_7) uncontrolled, as shown in Figure 4.18.

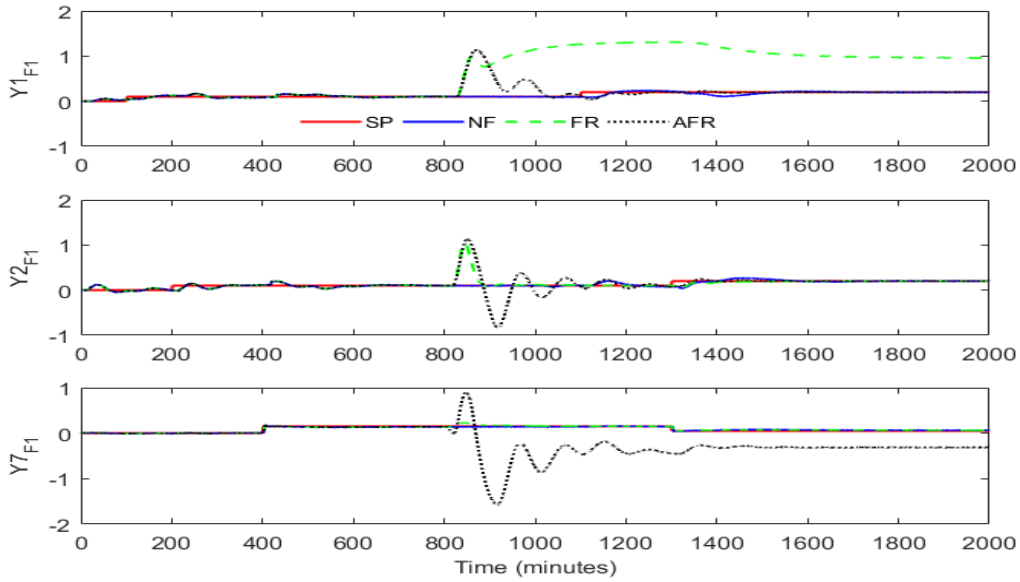


Figure 4.18: Output responses of accommodated actuator fault 1 (F1)

The control structure reconfiguration was achieved through the backup feedback signals presented in Figure 4.12. Appropriate backup feedback signals, in this case r_{b1} and y_{b1} were activated by changing their weightings from 0 to 1 and at the same time changing the weightings of the corresponding feedback signal to zero, as shown in equation 4.10. SP, NF, FR and AFR in Figures 4.18 – 4.20 represent set points, no faults, fault responses and accommodated fault responses respectively. It can be observed from Figure 4.18 that the bottom reflux duty was able to maintain the top end point at set point despite the influence of disturbances. Also, the performance of side end point control loop was slightly affected due to the strong interaction in the system.

In the case of side draw actuator fault ($F2$), none of the two remaining healthy actuators (u_1 and u_3) was able to accommodate the fault. Though the RGA analysis suggests the top draw (u_1) should be able to maintain the side end point (y_2) at set point, however its performance was very poor as can be observed from Figure 4.19. FTC was reconfigured to a 2 by 2 structure controlling the top end point (y_1) and side end point (y_2) by manipulating bottom reflux duty (u_3) and top draw (u_1) respectively making use of the backup feedback signals and reference point reconfiguration mechanism. The bottom reflux duty was able to keep the top end point at set point. However, the top draw was not effective in maintaining side end point at set point. The bottom reflux temperature is uncontrolled having reduced the control configuration to 2 by 2.

The same scenario was observed when the bottom reflux duty actuator fault (u_3) was

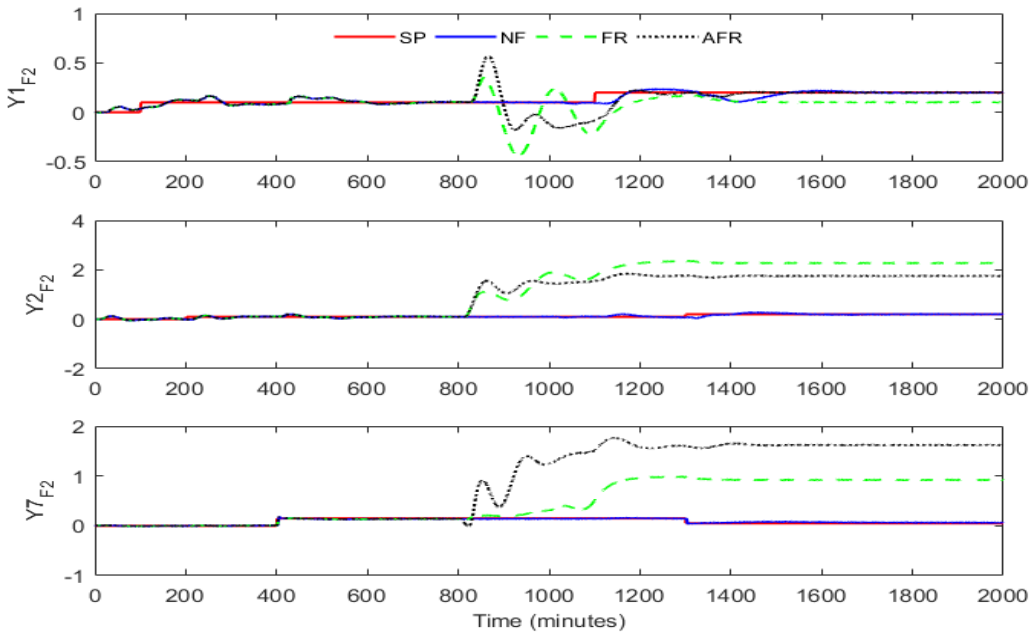


Figure 4.19: Output responses of accommodated actuator fault 2 (F2)

declared. Neither of the two remaining healthy actuators, top and side draw actuators (u_1 and u_2) were able to control the bottom reflux temperature. Figure 4.20 presents the fault tolerant controller performance for the bottom reflux duty actuator fault ($F3$) where top draw actuator (u_1) was reconfigured to control y_7. Observations from Figure 4.20 shows that implementation of fault tolerant controller in this particular case could not improve the system performance.

4.4 Application to Crude Distillation Unit

The preservation of the integrity of crude distillation unit in the presence of actuator faults through the implementation of the proposed actuator FTC is presented in this section. The crude distillation unit is a complex energy intensive industrial distillation process with substantial time lag and severe interaction amongst its control loops. The use of a control system in CDU only ensures system stability and consistent production of quality products as long as no fault occurs. However, in the presence of control system component faults, such as actuator fault, a more robust control system with automatic components containment capabilities will be required to provide desirable performance in the system. Implementation of the proposed actuator fault tolerant control system, as detailed in Chapter 3, on such a complex process demonstrates the simplicity, effectiveness

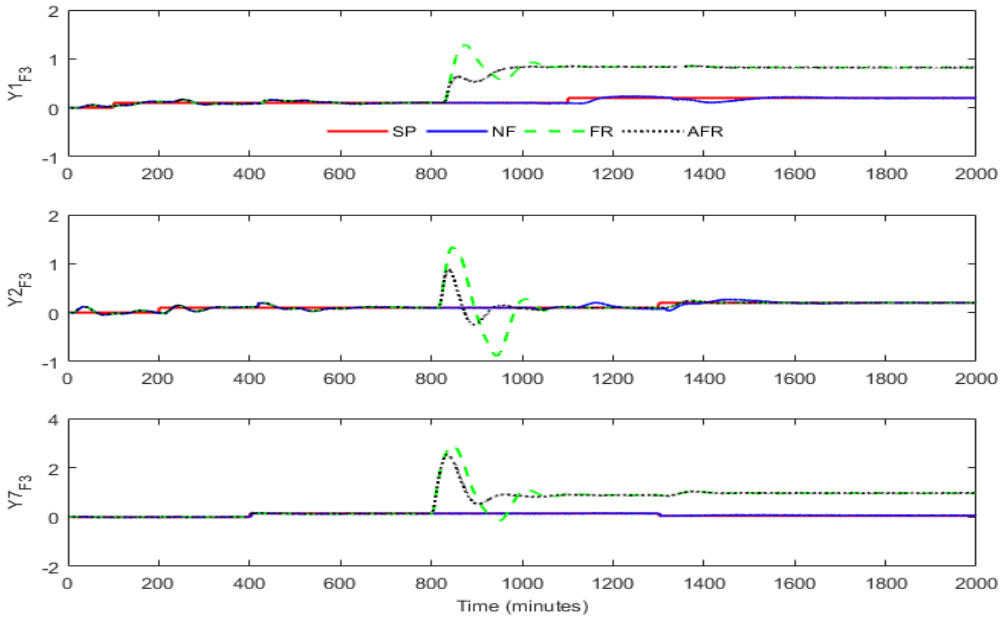


Figure 4.20: Output responses of accommodated actuator fault 3 (F3)

and applicability of the accommodating strategy in the presence of actuator faults. A dynamic HYSYS model of the crude distillation unit investigated in this thesis is presented in Figure 4.21. The dynamic CDU model has been used in previous works by Yu *et al.* (2008) (in Chinese) and Zhou *et al.* (2012) to investigate multi-objective optimization of industrial CDU and inferential estimation of kerosene dry point respectively.

4.4.1 Crude Distillation Unit Process Description

The crude distillation unit in HYSYS consists of a train of heat exchangers, an atmospheric CDU with a 3-phase condenser attached, a vacuum CDU, three pumparound cooling circuits, three side draws with stripper attached to each, crude furnace, several separator vessels and 29 control loops, as presented in Figure 4.21. Three different crudes designated as standard, middle and heavy are created in HYSYS. However, only standard crude was used throughout the simulation with random introduction of middle and heavy crudes during simulation, representing approximately 2% of the total volume of crude charged into the system. This introduction serves as disturbances in the system, representing changes in composition of the standard crude. The crude is heated to 185^oC through series of exchangers by exchange with hot intermediate streams from the crude and vacuum columns before entering the furnace where its temperature is raised to 360^oC, the

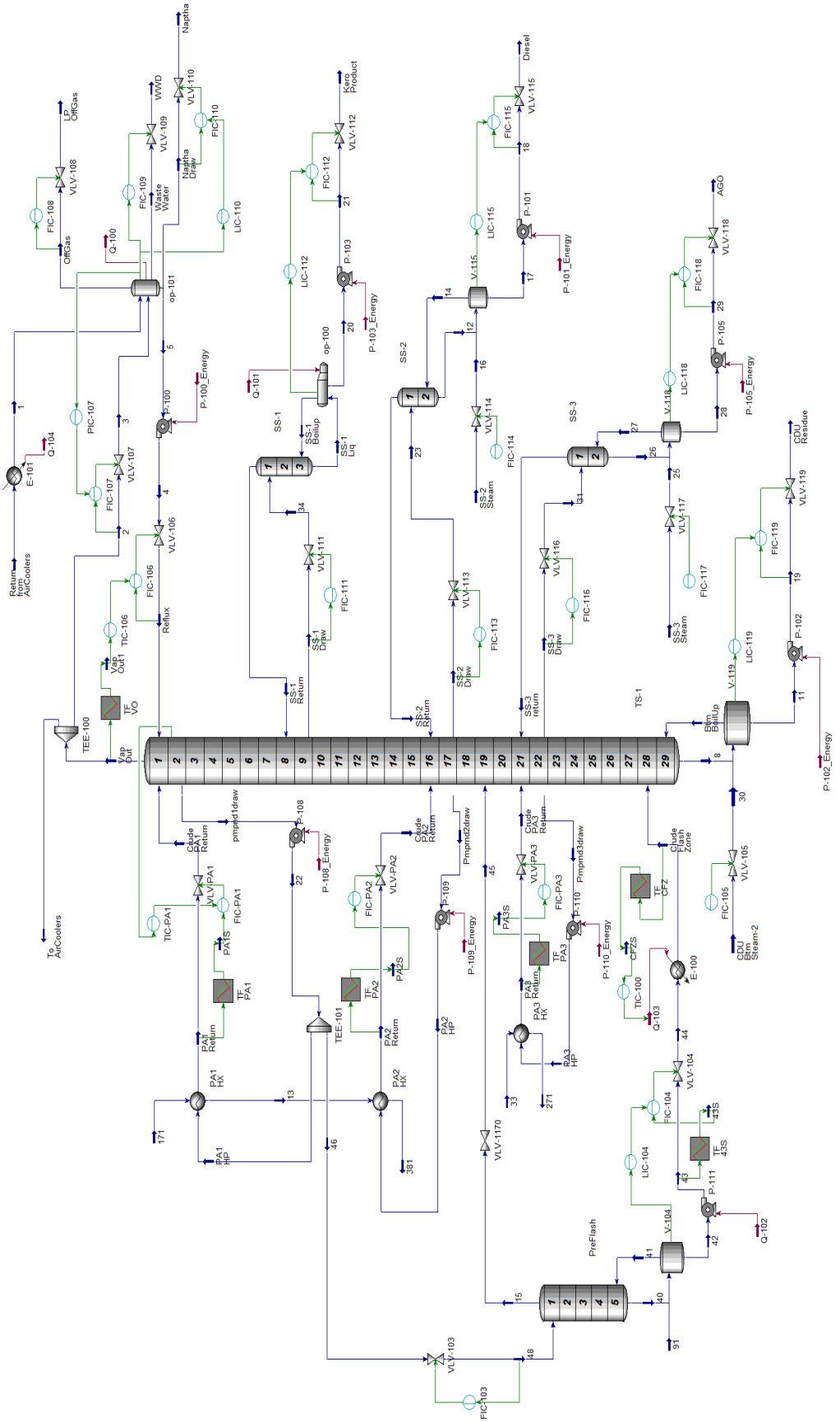


Figure 4.21: Crude distillation unit

temperature at which it enters the atmospheric column flash zone. The focus of this thesis is mainly on the atmospheric crude unit, hence operations and control of the vacuum unit is not discussed.

The crude mixes with the bottom boil-up vapour and the steam injected into the bottom separator vessel to strip the lightest hydrocarbons from the column bottom residue. As the hot vapour from the flash zone rises, it is contacted by the colder reflux flowing down the column. The pumparound circuits and the overhead condenser provide the reflux that is used to condense the side liquid products. The products from the system are naphtha, kerosene, diesel, atmospheric gas oil (AGO) and the CDU residue. The side liquid products are kerosene, diesel and AGO; and are drawn from column stages 9, 17 and 22 respectively. The side products are transferred into their respective side strippers with attached reboiler and separator vessels with steam injection lines to strip the products off lighter hydrocarbons. The pumparound streams are drawn from stages 2, 17 and 22; and the cooled pumparound streams are returned to stages 1, 16 and 21 respectively.

The column has 29 control loops – 20 flow control loops, 6 level control loops, 2 temperature control loops and 1 pressure control loop. The flow control loops are used to maintain the flowrates of specific streams including pumparound, side draw, steam and products rate. The level controllers are used as master controllers together with some flow controllers in a cascade control settings to control liquid percentage levels in the separators attached to the pre-flash column, the main column, side strippers and liquid level in the overhead condenser. The temperature controllers control the temperature of the crude entering the flash zone and that of the vapour leaving the top of the column, while the pressure controller controls the pressure of the 3-phase condenser. Details of all the controlled variables (y_i) – manipulated variables (u_i) pairing and their respective nominal operating conditions are presented in Table 4.10, while Table 4.11 presents the column nominal values for some selected process variables. The complete list of all the seventy-one process variables monitored in the system is presented in Appendix A. The controllers are used to maintain the flow, temperature and pressure profile of the column, which in turn maintain the specified product quality variables for the system. The product quality variables used for the CDU are ASTM D1160 cut-points at 0% and 100% for kerosene, ASTM D1160 cut-points at 90% and 95% for diesel, ASTM D93 flash points for kerosene and AGO, and AGO viscosity at 210F.

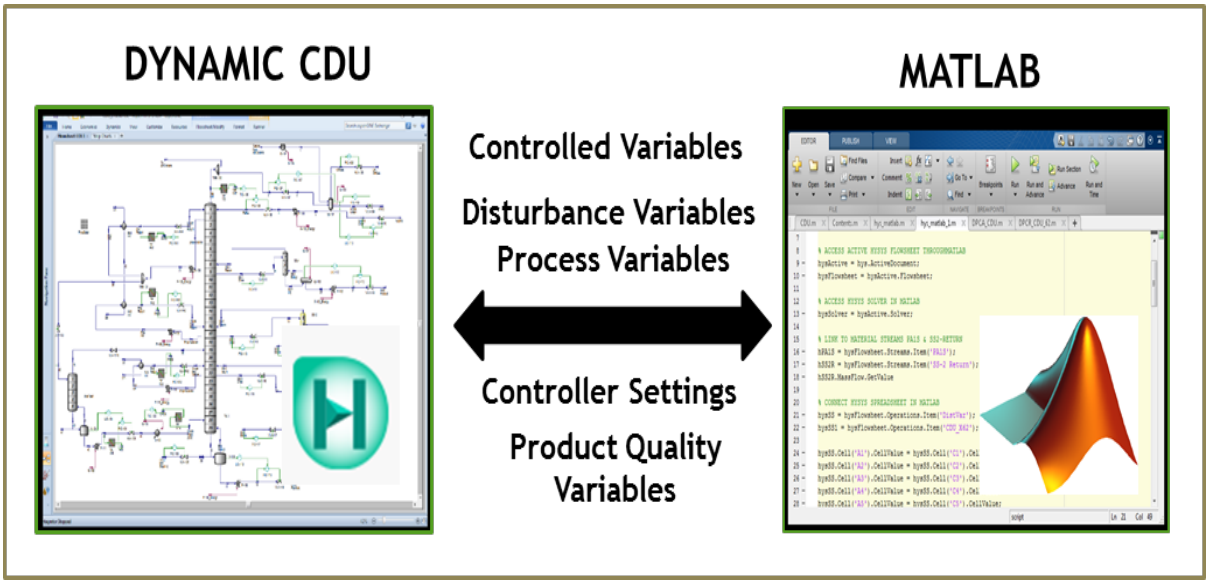


Figure 4.22: Interactions between HYSYS and MATLAB application

4.4.2 Development and Simulation of Interactive Dynamic Crude Distillation Unit

The dynamic crude distillation model in HYSYS is integrated with MATLAB programme to create an interactive dynamic crude distillation simulator, through which effective implementation of the proposed actuator fault tolerant control system could be achieved. This requirement is fundamental to successfully implement the actuator FTC because it allows flow of information between the two applications. The CDU model to monitor and control resides in HYSYS while the actuator FTC system is developed in MATLAB. Hence, the requirement to automate the operation, monitoring and implementation of the proposed FTCS on the dynamic CDU model in HYSYS becomes necessary. Figure 4.22 illustrates the interface between the two applications. First, an active connection is created in MATLAB that allows it to connect and simulate the dynamic CDU model in HYSYS for a specified period of time. Then different sub connections are created in MATLAB to access objects in the CDU HYSYS model that contain the variables of interest. These are process variables that need to be monitored, manipulated and controlled in some cases to ensure that actuator faults are quickly detected, identified and accommodated, depending on the severity of the fault identified.

To minimise the number of object connections to be created in MATLAB, an appropriate number of spreadsheet objects are created in HYSYS that contain all the different variables and parameters that may need to be adjusted as appropriate during the simula-

Table 4.10: CDU control structure

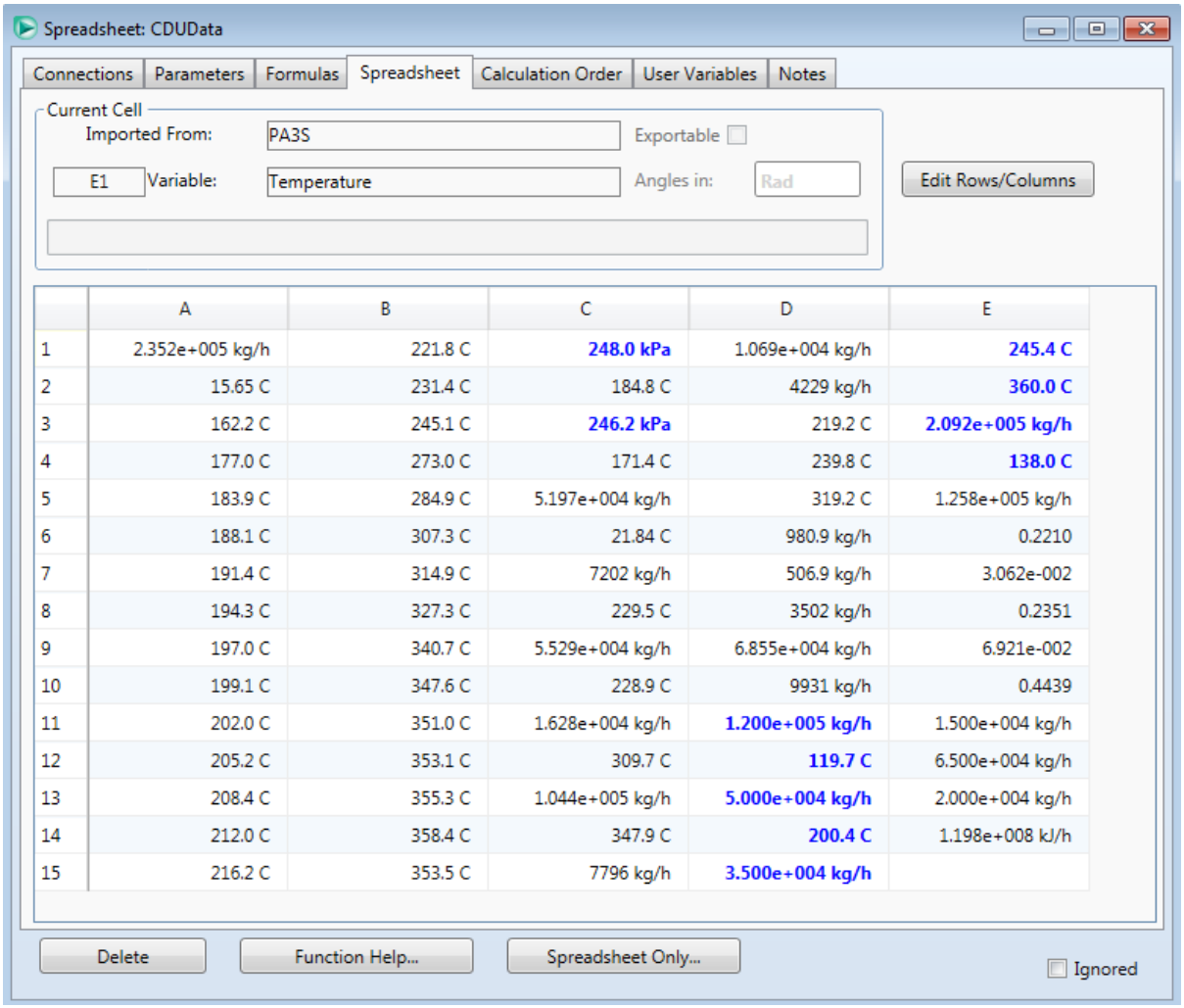
No	Controllers	Controlled Variables	Manipulated Variables	Nominal Valve Opening
1	FIC-PA1	1st pumparound mass flow	Valve PA1 desired position	85.70%
2	FIC-PA2	2nd pumparound mass flow	Valve PA2 desired position	77.33%
3	FIC-PA3	3rd pumparound mass flow	Valve PA3 desired position	56.65%
4	TIC-100	Crude flash zone temperature	Q-103 control valve	48.51%
5	FIC-103	Stream 48 mass flow	Valve 103 desired position	32.10%
6	LIC-104	V-104 liquid percent level	Separator V104 liquid level	49.37%
7	FIC-104	Stream 43 mass flow	Valve 104 desired position	41.59%
8	FIC-105	CDU steam mass flow	Valve 105 desired position	50.83%
9	TIC-106	Vap out temperature	FIC-106 SP	34.27%
10	FIC-106	Reflux mass flow	Valve 106 desired position	32.66%
11	PIC-107	OP-101 Condenser Pressure	FIC-107 SP	92.61%
12	FIC-107	Stream 2 mass flow	Valve 107 desired position	90%
13	FIC-108	LP Offgas mass flow	Valve 108 desired position	0%
14	LIC-110	OP-101 liquid percent level	FIC-110 SP	59.50%
15	FIC-110	Naphtha draw flow	Valve 110 desired position	30.97%
16	FIC-111	Side draw 1 mass flow	Valve 111 desired position	39.13%
17	LIC-112	OP-100 reboiler liquid percent level	FIC-112 SP	7.58%
18	FIC-112	Kerosene mass flow	Valve 112 desired position	21.27%
19	FIC-113	Side draw 2 mass flow	Valve 113 desired position	79.07%
20	FIC-114	Side stripper 2 steam flow	Valve 114 desired position	35.35%
21	LIC-115	Separator V-115 liquid level	FIC-115 SP	57.13%
22	FIC-115	Diesel mass flow	Valve 115 desired position	73.34%
23	FIC-116	Side draw 3 mass flow	Valve 116 desired position	74.16%
24	FIC-117	Side stripper 3 steam flow	Valve 117 desired position	76.73%
25	LIC-118	Separator V-118 liquid level	FIC-118 SP	82.62%
26	FIC-118	AGO mass flow	Valve 118 desired position	90.00%
27	LIC-119	Separator V-119 liquid level	FIC-119 SP	47.46%
28	FIC-119	CDU residue mass flow	Valve 119 desired position	42.29%

tion. The spreadsheets created in HYSYS include spreadsheets for all the process variables of interest, the manipulated variables, controlled variables, disturbance variables, process quality variables, and the percentage maximum control valve openings. Figures 4.23 shows the spreadsheet of all the process variables monitored in the system. Data in the spreadsheets are accessed and stored in MATLAB during simulation. The disturbance variables spreadsheet is used to randomly introduce disturbances into the system during simulation while the actuator fault spreadsheet is also used to introduce faults into the system. The process variables collected during the simulation include temperature and flow rate measurements of the crude flash zone, pump-arounds, side draws, reflux stream, and the temperature measurements of all the 29 stages in the column. Flow rates and temperatures of naphtha, kerosene, diesel, AGO, the CDU residue and the ratios of the feed rate

Table 4.11: Nominal crude distillation unit operating conditions

Selected Process Variables	Values
Crude mass flow	235,200 kg/hr
Crude temperature	15.65^oC
Crude flash zone flow	209,200 kg/hr
Crude flash zone temperature	360^oC
Reflux flow rate	68,550 kg/hr
Reflux flow temperature	22^oC
Vapour out flow	125,800 kg/hr
Vapour out temperature	138^oC
Column bottom boil-up	9,931 kg/hr
First pumparound flowrate	121,600 kg/hr
First pumparound return temperature	119^oC
Second pumparound flowrate	50,000 kg/hr
Second pumparound return temperature	200^oC
Third pumparound flowrate	35,000 kg/hr
Third pumparound return temperature	245^oC
First side draw flow rate	15,000 kg/hr
First side draw flow rate	199^oC
Second side draw flow rate	65,000 kg/hr
Second side draw flow rate	245^oC
Third side draw flow rate	20,000 kg/hr
Third side draw flow rate	327.2^oC
Naphtha to crude feed ratio	0.221
Kerosene to crude feed ratio	0.03062
Diesel to crude feed ratio	0.2351
AGO to crude feed ratio	0.06921
CDU residue to crude feed ratio	0.4439
Heat flow to the furnace	1.198e+008 kJ/h

to each of the products flow rates are also included. The automated HYSYS-MATLAB CDU model is simulated for 600 minutes with 30 seconds sampling time to collect 1200 data points under normal operating conditions. An example of the MATLAB code used for the simulation is presented in Appendix B. A total of seventy-one variables including the controlled variables, some manipulated variables and the disturbance variables are monitored during the simulation. Figures 4.24 and 4.25 present the plots of some select process variables and those of the product quality variables for the system during normal operating conditions. Table 4.12 presents the nominal values of the product quality variables. The system is also simulated for 400 minutes with 30 seconds sampling time for the five different actuator faults investigated in this section ($F1$ – $F5$) as shown in Figure 4.26. Details of the actuator faults are discussed in Section 4.4.4.



	A	B	C	D	E
1	2.352e+005 kg/h	221.8 C	248.0 kPa	1.069e+004 kg/h	245.4 C
2	15.65 C	231.4 C	184.8 C	4229 kg/h	360.0 C
3	162.2 C	245.1 C	246.2 kPa	219.2 C	2.092e+005 kg/h
4	177.0 C	273.0 C	171.4 C	239.8 C	138.0 C
5	183.9 C	284.9 C	5.197e+004 kg/h	319.2 C	1.258e+005 kg/h
6	188.1 C	307.3 C	21.84 C	980.9 kg/h	0.2210
7	191.4 C	314.9 C	7202 kg/h	506.9 kg/h	3.062e-002
8	194.3 C	327.3 C	229.5 C	3502 kg/h	0.2351
9	197.0 C	340.7 C	5.529e+004 kg/h	6.855e+004 kg/h	6.921e-002
10	199.1 C	347.6 C	228.9 C	9931 kg/h	0.4439
11	202.0 C	351.0 C	1.628e+004 kg/h	1.200e+005 kg/h	1.500e+004 kg/h
12	205.2 C	353.1 C	309.7 C	119.7 C	6.500e+004 kg/h
13	208.4 C	355.3 C	1.044e+005 kg/h	5.000e+004 kg/h	2.000e+004 kg/h
14	212.0 C	358.4 C	347.9 C	200.4 C	1.198e+008 kJ/h
15	216.2 C	353.5 C	7796 kg/h	3.500e+004 kg/h	

Figure 4.23: Spreadsheet for all the monitored process variables

4.4.3 Control Strategies Prior Assessment

The current control structure of the system as presented in Figure 4.21 and discussed in Section 4.4.1 needs to be modified to allow for implementation of our proposed actuator FTC on the system. Majority of the controlled variables that we are interested in are indirectly controlled by maintaining the flow rates of some of the streams including pump-arounds, side draws and products draw rates. After careful consideration of the CDU current control structure and taking into consideration advice from experts, we decided to restructure some of the control loops to directly control some variables using suitable manipulated variables. At first, through consultation of relevant literature and careful observation of the system, 12 controlled variables which include temperatures of the stages where side products are drawn and the flow rates and temperature of some of the products were selected. Also, 12 possible manipulated variables that could be used to directly control

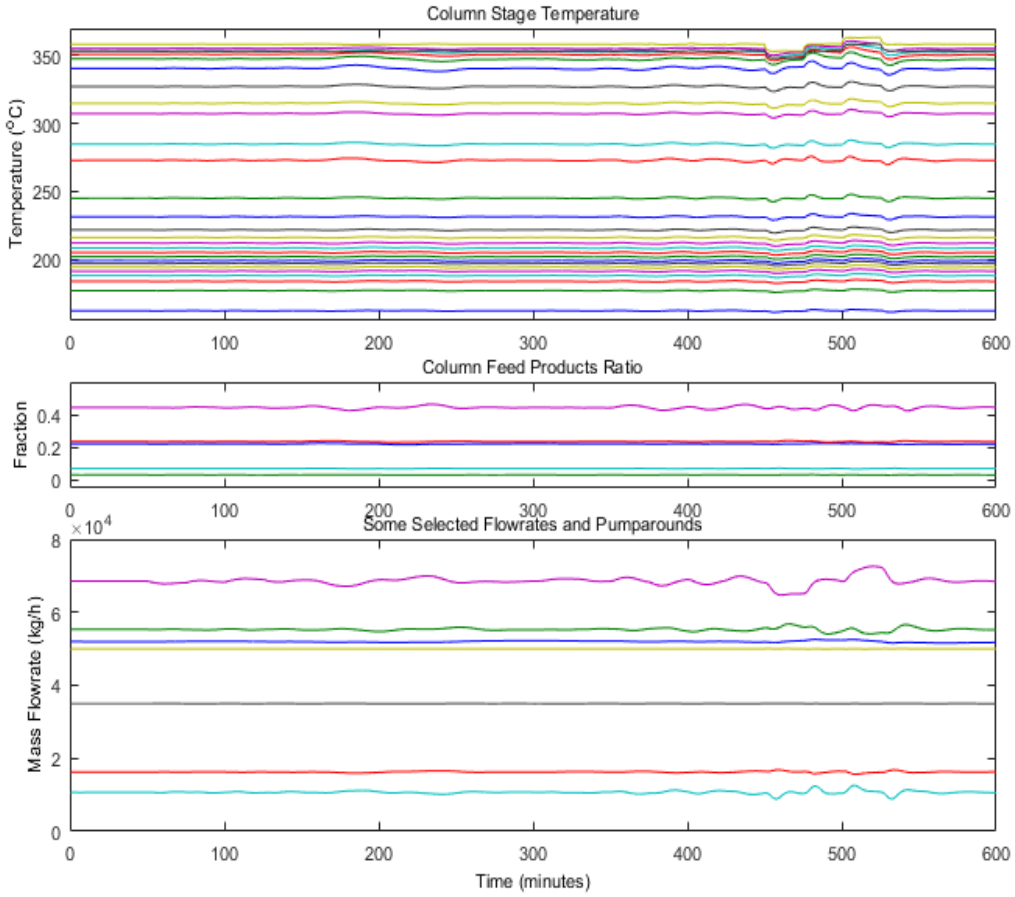


Figure 4.24: Plot of some selected process variables

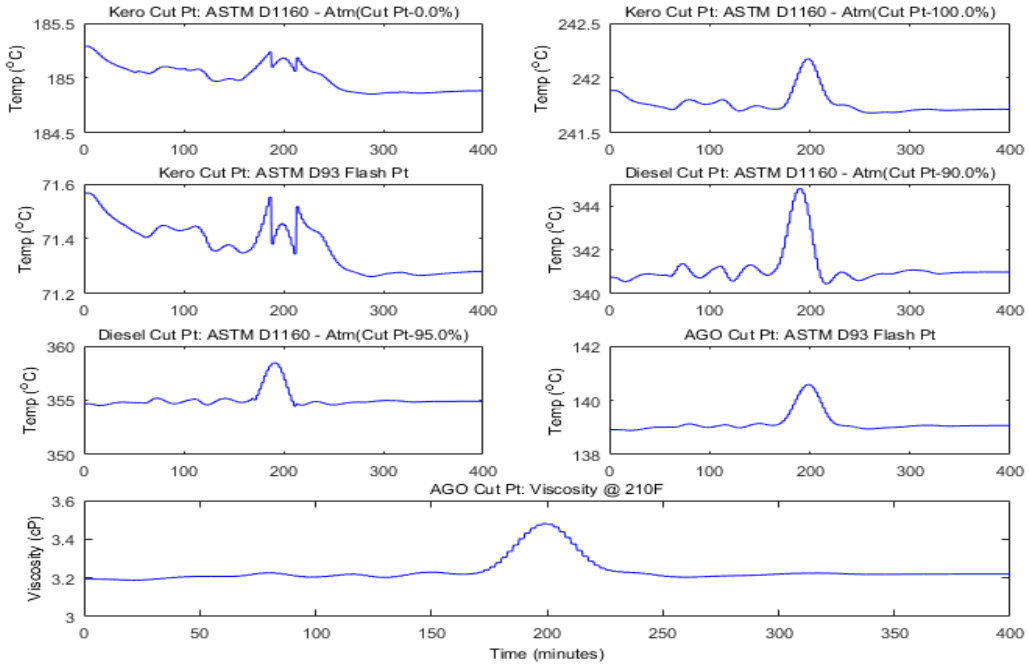


Figure 4.25: Dynamic CDU product quality variables

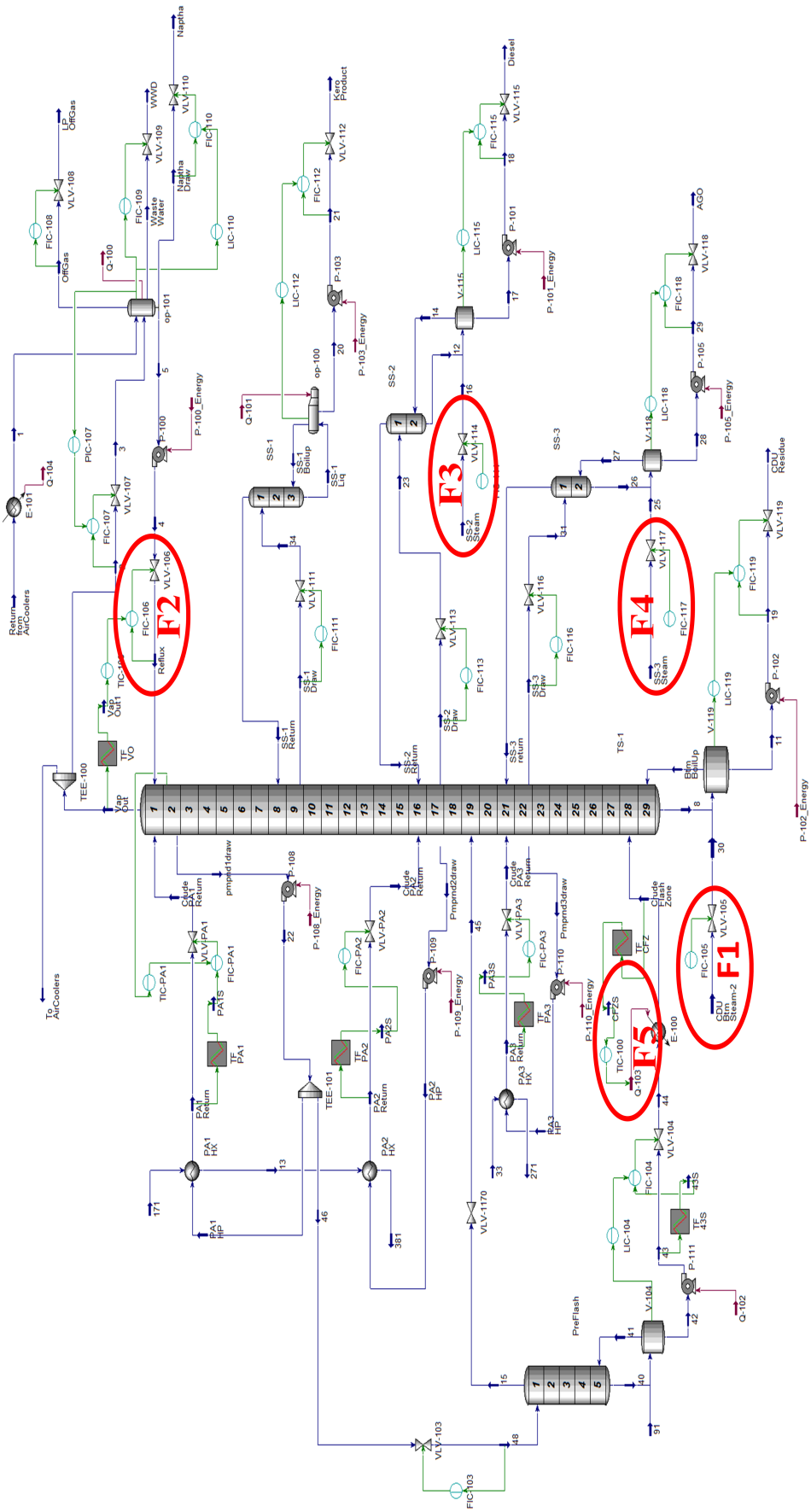


Figure 4.26: Dynamic CDU model in HYSYS showing faults F1 – F5

Table 4.12: CDU process quality variables

Process Quality Variables	Values
Kero Cut Pt: ASTM D1160 – Atm (Cut Pt-0.0%)	184.9^oC
Kero Cut Pt: ASTM D1160 – Atm (Cut Pt-100.0%)	241.7
Kero Cut Pt: ASTM D93 Flash Pt	71.28^oC
Diesel Cut Pt: ASTM D1160 – Atm (Cut Pt-90.0%)	341^oC
Diesel Cut Pt: ASTM D1160 – Atm (Cut Pt-95.0%)	354.9^oC
AGO Cut Pt: ASTM D93 Flash Pt	139.1^oC
AGO Cut Pt: Viscosity @ 210F	3.22 cP

the selected controlled variable were selected as presented in Table 4.13. The twelve selected control loops are each simulated in open loop for 700 minutes with 30 second sampling time, making necessary changes to their set points to collect sufficient data for model identification. Data for the open loop responses to changes in the manipulated variables collected during simulation was then used to develop a set of first and second order plus dead time (FOPDT & SOPDT) models using System Identification Toolbox in MATLAB.

Table 4.13: Initial 12 by 12 inputs – outputs reconfiguration for actuator FTC

No	Controllers	Controlled Variables	Manipulated Variable
1	FIC-PA1	SS-2 return flow rate	1st pumparound mass flow
2	FIC-PA2	Stage 17 temperature	2nd pumparound mass flow
3	FIC-PA3	Stage 22 temperature	3rd pumparound mass flow
4	FIC-104	CDU residue mass flow	Crude flash zone mass flow
5	FIC-105	Bottom boil-up flow	CDU bottom steam 2
6	FIC-106	Vap out temperature	Reflux mass flow
7	FIC-111	Kerosene product mass flow	SS-1 draw rate
8	FIC-114	Diesel temperature	SS-2 steam
9	FIC-117	AGO temperature	SS-3 steam
10	FIC-113	Diesel product mass flow	SS-2 draw rate
11	FIC-116	AGO product mass flow	SS-3 draw rate
12	TIC-100	Crude flash zone temperature	Furnace heat flow

The 12 by 12 model of the system is presented in Appendix C and was used to investigate the level of interactions amongst the loops using RGA and DRGA analysis. Several rigorous simulations and analysis of the system were undertaken based on the results of the RGA and DRGA suggested controlled variable (y_i) – manipulated variable (u_i) pairing. A number of the pairing suggested by the tools drive the system to marginal

stability and in some cases make the system unstable. Reasonable efforts were made to reduce the model to a manageable 5 by 5 system in order to implement the proposed actuator FTC on the automated CDU system.

Table 4.14: Reduced 5 by 5 inputs – outputs pairing

No	Controllers	Controlled Variables	Manipulated Variable
1	FIC-105	Bottom boil-up flow (9,931 kg/h)	CDU bottom steam 2 (3,500 kg/h)
2	FIC-106	Stage 1 temperature (138 ^oC)	Reflux mass flow (68,550 kg/h)
3	FIC-114	Diesel temperature (228.9 ^oC)	SS-2 steam flow (1,000 kg/h)
4	FIC-117	AGO temperature (309.7 ^oC)	SS-3 steam flow (500 kg/h)
5	TIC-100	CFZ temperature (360 ^oC)	Furnace heat flow (1.198e+08 kJ/h)

Details of the reduced model inputs – outputs pairing is shown in Table 4.14 and equations 4.12 and 4.13 present the FOPDT models and the RGA results respectively. The reduced fault free system has bottom boil-up flow (y_1), stage 1 temperature (y_2), diesel temperature (y_3), AGO temperature (y_4) and crude flash zone temperature (y_5) being directly controlled by CDU bottom steam (u_1), reflux flow rate (u_2), side stripper 2 (SS-2) steam flow rate (u_3), side stripper 3 (SS-3) steam flow rate (u_4) and furnace heat output (u_5) respectively. The new reconfigured pairing using the manipulated variables to directly control the selected controlled variables is again simulated to ensure effective control of the outputs. The performances of the new control structure using PID controllers tuned with the IMC tuning tool in HYSYS is compared with those of the original structure for the selected controlled variables and presented in Figure 4.27. Table 4.15 presents the PID controllers' settings used for the reconfigured system during fault free and faulty situations.

$$G_5(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.0535e^{-2.59s}}{1+1.414s} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.191}{1+1.012s} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \quad (4.12)$$

$$\Lambda_{G5}(s) = \begin{bmatrix} 1.1472 & 0.0427 & 0.0191 & -0.0015 & -0.2075 \\ 0.0344 & 0.5563 & 0.3548 & 0.0723 & -0.0178 \\ 0.0138 & 0.0647 & 0.6097 & 0.1529 & 0.1589 \\ -0.0473 & 0.0580 & 0.0205 & 0.7805 & 0.1882 \\ -0.1481 & 0.2783 & -0.0042 & -0.0042 & 0.8782 \end{bmatrix} \tag{4.13}$$

where G_5 and Λ_{G5} are the transfer function models of the reduced 5 by 5 system and the corresponding RGA respectively.

Having established a stable operation of the fault free system with the restructured controllers and achieved effective control of the selected outputs, possible reconfiguration of the control structure is undertaken in the event of an actuator fault occurring in any of the five control loops. RGA and DRGA tools as described in Sections 3.2.2.1 and 3.2.2.2 are used *a priori* to investigate possible control structure reconfiguration upon detection of an actuator fault. For instance, a fault in the CDU bottom steam (u_1) control valve will reduce the system to a 5 by 4 control structure where four manipulated variables are available to maintain five controlled variables at desired set points. Non-squared RGA is first used to eliminate the least effective controlled variable thereby reducing the system to 4 by 4 after which RGA and DRGA are used to select possible input-output pairing. The decision on the controlled variable to leave out in the effect of an actuator fault could also be due to economic reasons, and most importantly what is physically and technically achievable given the circumstance. Equations 4.14 – 4.23 show the reduced 4 by 4 models for each of the five faults and their respective RGA results. Example application of the DRGA system interaction analysis tool for the input-output pairing and reconfiguration of the fault-free system and the reduced system under an actuator fault respectively is presented in Appendix D. However, not all the faults can be accommodated by switching the manipulated variables, even for the fault-free system as observed during the initial fault free simulation. Table 4.16 presents the possible inputs – outputs reconfiguration upon detection and identification of an actuator fault.

$$G_{F1}(s) = \begin{bmatrix} \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{-0.0535e^{-3.59s}}{1+1.414s} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.191}{1+1.012s} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{4.14}$$

Table 4.15: Reconfigurable actuator FTC PID settings

		y_1	y_2	y_3	y_4	y_5
	K_p	0.78	0.50	8.35	8.48	0.51
Normal	T_I	0.04	0.30	1.74	3.90	0.69
	T_D	–	–	–	–	–
	K_p	–	0.50	8.35	8.48	0.51
F1	T_I	–	0.30	1.74	3.90	0.69
	T_D	–	–	–	–	–
	K_p	0.78	0.25	8.35	8.48	–
F2	T_I	0.04	13.1	1.74	3.90	–
	T_D	–	0.25	–	–	–
	K_p	0.78	–	0.45	8.48	0.51
F3	T_I	0.04	–	5.00	3.90	0.69
	T_D	–	–	0.79	–	–
	K_p	0.78	0.50	–	0.58	0.51
F4	T_I	0.04	0.30	–	2.00	0.69
	T_D	–	–	–	1.66	–
	K_p	0.78	0.50	8.35	8.48	–
F5	T_I	0.04	0.30	1.74	3.90	–
	T_D	–	–	–	–	–

$$\Lambda_{F1} = \begin{bmatrix} 0.5681 & 0.3480 & 0.0685 & 0.0153 \\ 0.0675 & 0.6130 & 0.1515 & 0.1680 \\ 0.0508 & 0.0176 & 0.7690 & 0.1625 \\ 0.3136 & 0.0214 & 0.0110 & 0.6541 \end{bmatrix} \tag{4.15}$$

Table 4.16: Possible inputs – outputs reconfiguration

	Manipulated Inputs					
Controlled Outputs	Normal	$F1$	$F2$	$F3$	$F4$	$F5$
Bottom boil-up flow (y_1)	u_1	–	u_1	u_1	u_1	u_1
Stage 1 temperature (y_2)	u_2	u_2	u_5	–	u_2	u_2
Diesel temperature (y_3)	u_3	u_3	u_3	u_2	–	u_3
AGO temperature (y_4)	u_4	u_4	u_4	u_4	u_3	u_4
Crude flash zone temp. (y_5)	u_5	u_5	–	u_5	u_5	–

G_{F1} and Λ_{F1} are the transfer function models and the corresponding RGA values for the reduced 4 by 4 system under actuator fault one ($F1$). The resulting $y_i - u_i$ pairings are

$y_2 - u_2$, $y_3 - u_3$, $y_4 - u_4$ and $y_5 - u_5$ after isolating u_1 (CDU bottom steam control valve) as there is no suitable manipulated variable to control y_1.

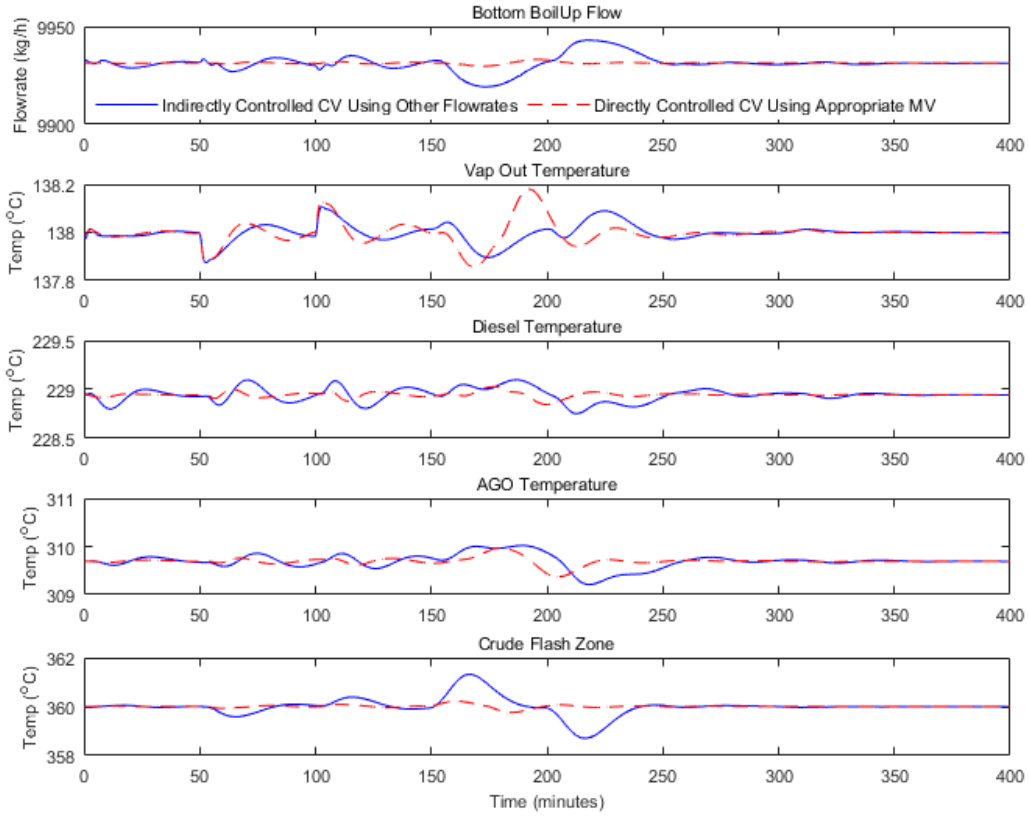


Figure 4.27: Controlled variable plot for the reconfigured fault-free system

$$G_{F2}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \end{bmatrix} \tag{4.16}$$

$$\Lambda_{F2} = \begin{bmatrix} 1.2927 & 0.0195 & -0.0016 & -0.3107 \\ -0.1711 & 0.3482 & 0.0661 & 0.7568 \\ -0.0262 & 0.6113 & 0.1518 & 0.2632 \\ -0.0954 & 0.0210 & 0.7837 & 0.2907 \end{bmatrix} \tag{4.17}$$

G_{F2} and Λ_{F2} are the transfer function models and the corresponding RGA values for the reduced 4 by 4 system under actuator fault two ($F2$). The reconfigured control structure under $F2$ for the inputs – outputs pairings are $y_1 - u_1$, $y_2 - u_5$, $y_3 - u_3$ and $y_4 - u_4$ after isolating u_2 (reflux flow control valve fault), leaving y_5 uncontrolled.

$$G_{F3}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.03556}{s+0.098} & \frac{-0.055931e^{-3.6s}}{1+1.001s} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{4.18}$$

$$\Lambda_{F3} = \begin{bmatrix} 1.1254 & 0.0810 & -0.0010 & -0.2054 \\ 0.0632 & 0.5421 & 0.2534 & 0.1412 \\ -0.0409 & 0.1038 & 0.7508 & 0.1864 \\ -0.1477 & 0.2731 & -0.0032 & 0.8779 \end{bmatrix} \tag{4.19}$$

G_{F3} and Λ_{F3} are the transfer function models and the corresponding RGA values for the reduced 4 by 4 system under actuator fault three ($F3$) where u_1, u_2, u_4 and u_5 are reconfigured to control y_1, y_3, y_4, and y_5 respectively after isolating the faulty control valve (u_3), leaving y_2 uncontrolled.

$$G_{F4}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.191}{1+1.02s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{4.20}$$

$$\Lambda_{F4} = \begin{bmatrix} 1.1363 & 0.0520 & 0.0533 & -0.2417 \\ 0.0281 & 0.5068 & 0.5879 & -0.1228 \\ -0.0167 & 0.1651 & 0.3755 & 0.4761 \\ -0.1478 & 0.2762 & -0.0168 & 0.8884 \end{bmatrix} \tag{4.21}$$

G_{F4} and Λ_{F4} are the reduced 4 by 4 transfer function models and the corresponding RGA values for the CDU system under actuator fault four ($F4$) where y_1, y_2, y_4, and y_5 are controlled directly using u_1, u_2, u_3 and u_5 respectively after control structure reconfiguration with SS-3 steam control valve (u_4) isolated and y_3 uncontrolled.

$$G_{F5}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} \\ \frac{0.03556}{s+0.098} & \frac{-0.0535e^{-3.59s}}{1+1.414s} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} \end{bmatrix} \tag{4.22}$$

$$\Lambda_{F5} = \begin{bmatrix} 0.8544 & 0.1287 & 0.0183 & -0.0014 \\ 0.0297 & 0.5435 & 0.3546 & 0.0721 \\ 0.0747 & 0.1632 & 0.6074 & 0.1547 \\ 0.0412 & 0.1646 & 0.0197 & 0.7746 \end{bmatrix} \tag{4.23}$$

G_{F5} and Λ_{F5} are the transfer function models and the corresponding RGA values for the CDU system under actuator fault five ($F5$). There is no suitable manipulated variable that could be used to directly maintain y_5 at set point and the system control structure is reduced to 4 by 4 with $y_1 - u_1, y_2 - u_2, y_3 - u_3$ and $y_4 - u_4$ inputs – outputs pairing respectively.

Table 4.17: Crude distillation unit fault list

Fault	Fault description
F1: FIC 105	CDU bottom steam control valve maximum opening restricted to 30% (50.84**)
F2: FIC 106	Reflux flow control valve maximum opening restricted to 28% (32.66**)
F3: FIC 114	Side stripper-2 steam control valve maximum opening restricted to 25% (35.35**)
F4: FIC 117	Side stripper-3 steam control valve maximum opening restricted to 50% (76.73**)
F5: TIC 100	Furnace heat flow control valve maximum opening restricted to 44.82% (48.51**)

Note: ** – Nominal operating condition

4.4.4 Introduction of Actuator Faults

The five actuator faults investigated in this CDU case study are presented in Table 4.17. F1 is the bottom boil-up control valve fault; $F2$ is the reflux flow control valve fault; $F3$ is the SS-2 steam control valve fault; $F4$ is the $SS-3$ steam control valve fault, and $F5$ is the furnace heat flow control valve fault. The full range of throttling of the control valves is restricted one at a time to values below their nominal operating conditions on 150 minutes (sample 300) during the simulation. These restrictions limit the ability of the individual control valve to maintain their respective controlled variables at set point leading to detection of faults. Effects of the individual faults on the five controlled variables and the seven product quality variables presented in Tables 4.13 and 4.11 respectively are shown in Figures 4.28 to 4.37. Figures 4.28 and 4.29 show the effects of F1 on each of the controlled variables and the process quality variables respectively; Figures 4.30 and 4.31 show those of $F2$; Figures 4.32 and 4.33 show the effect of $F3$ on the controlled and product quality variables in that order; Figures 4.34 and 4.35 present the effect of $F4$ on the same variables and Figures 4.36 and 4.37 show the effect of $F5$ on both sets of

controlled variables and product quality variables respectively.

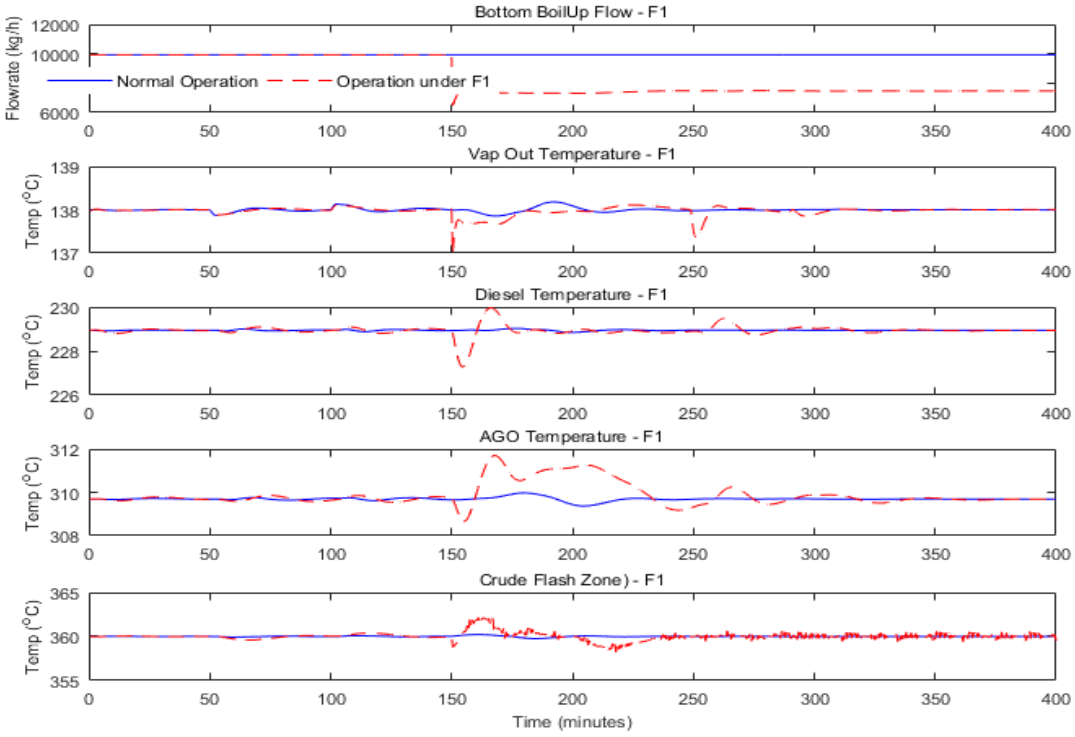


Figure 4.28: Plots of the controlled variables under F1

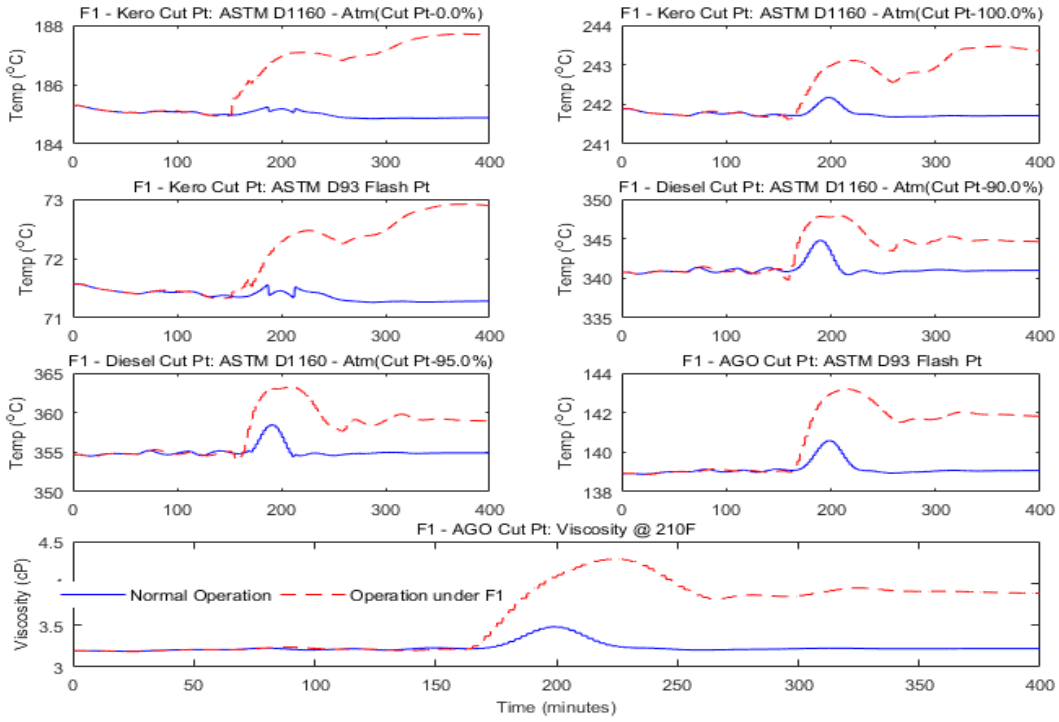


Figure 4.29: Plots of the product quality variables under F1

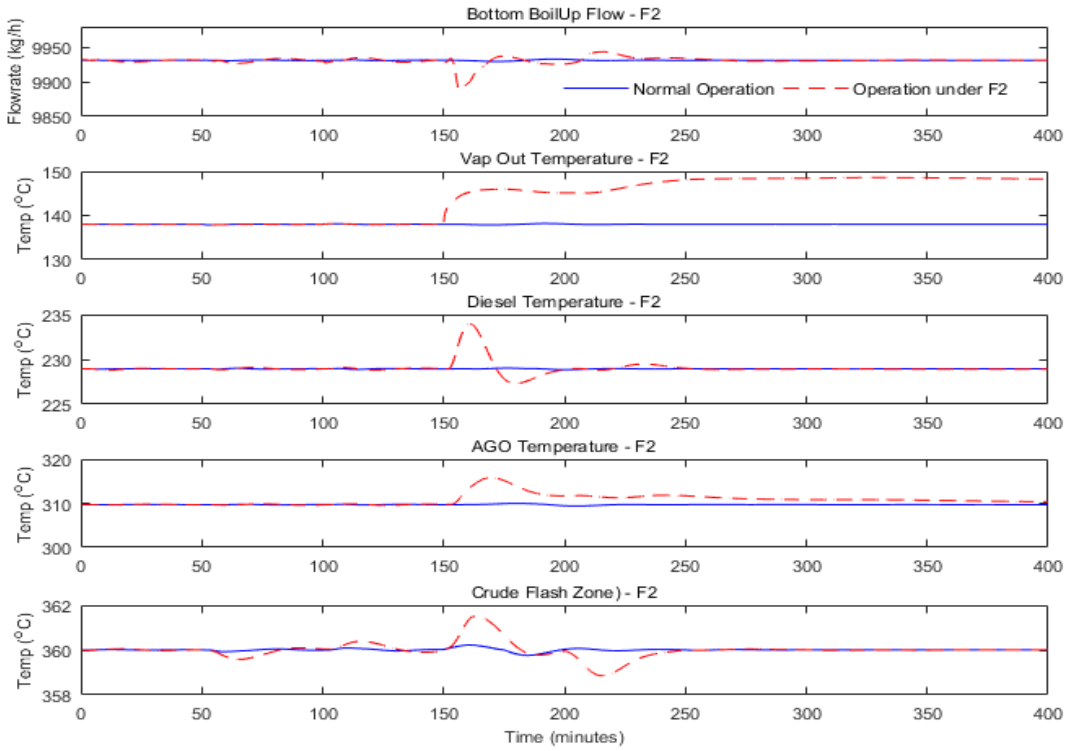


Figure 4.30: Plots of the controlled variables under F2

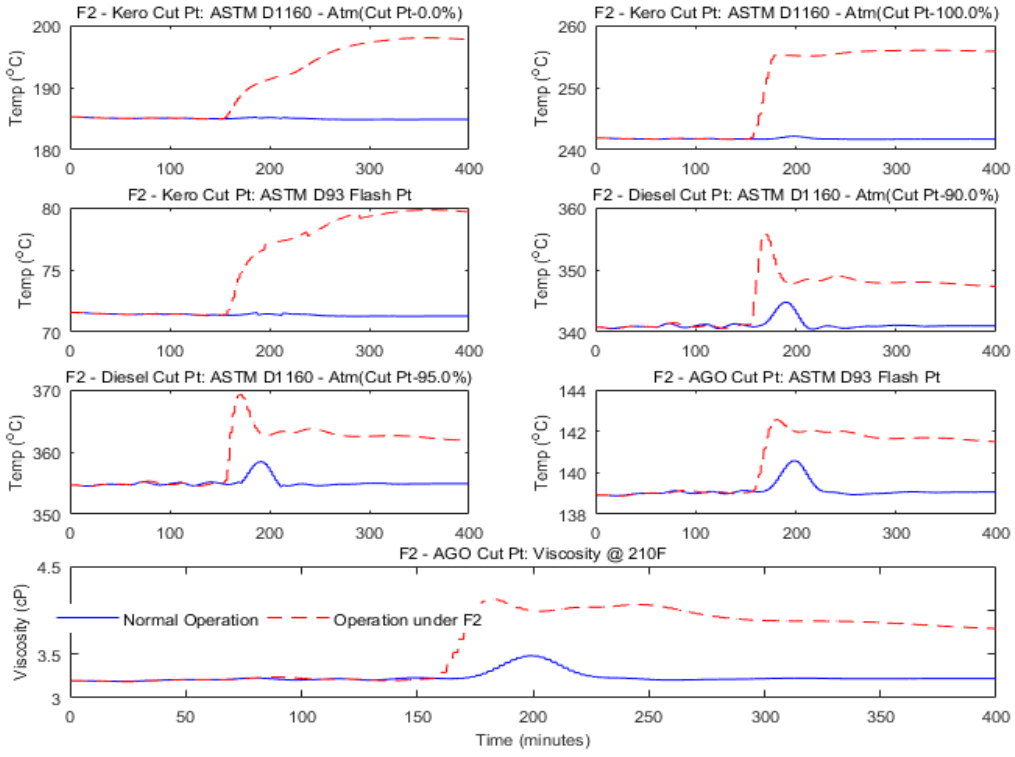


Figure 4.31: Plots of the product quality variables under F2

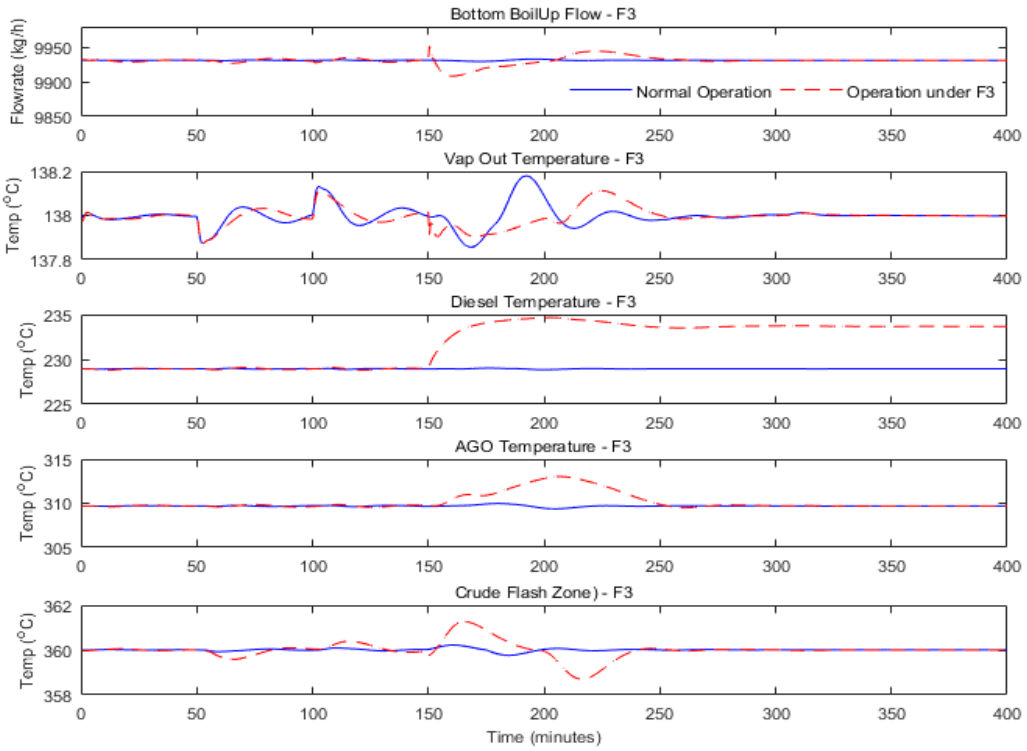


Figure 4.32: Plots of the controlled variables under F3

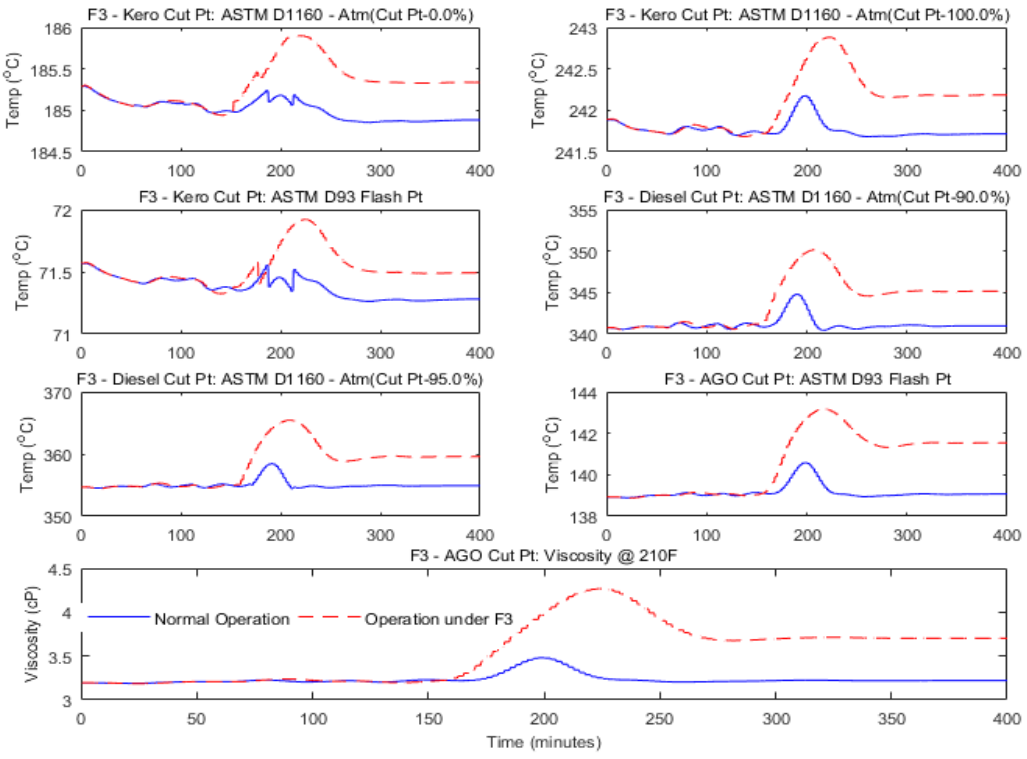


Figure 4.33: Plots of the product quality variables under F3

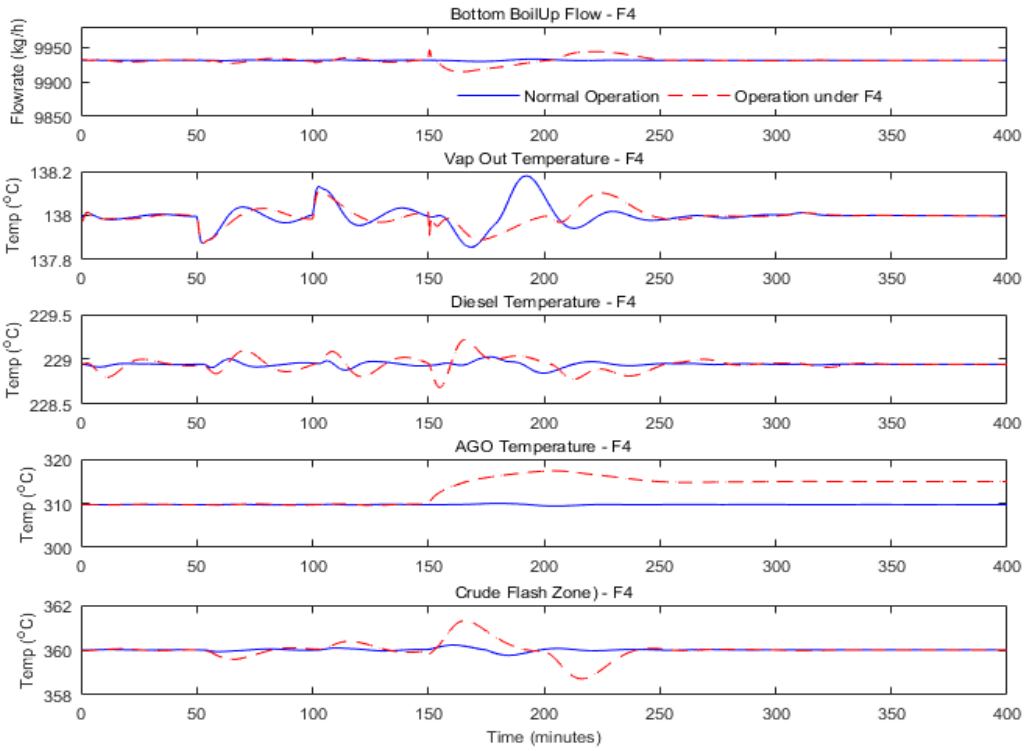


Figure 4.34: Plots of the controlled variables under F4

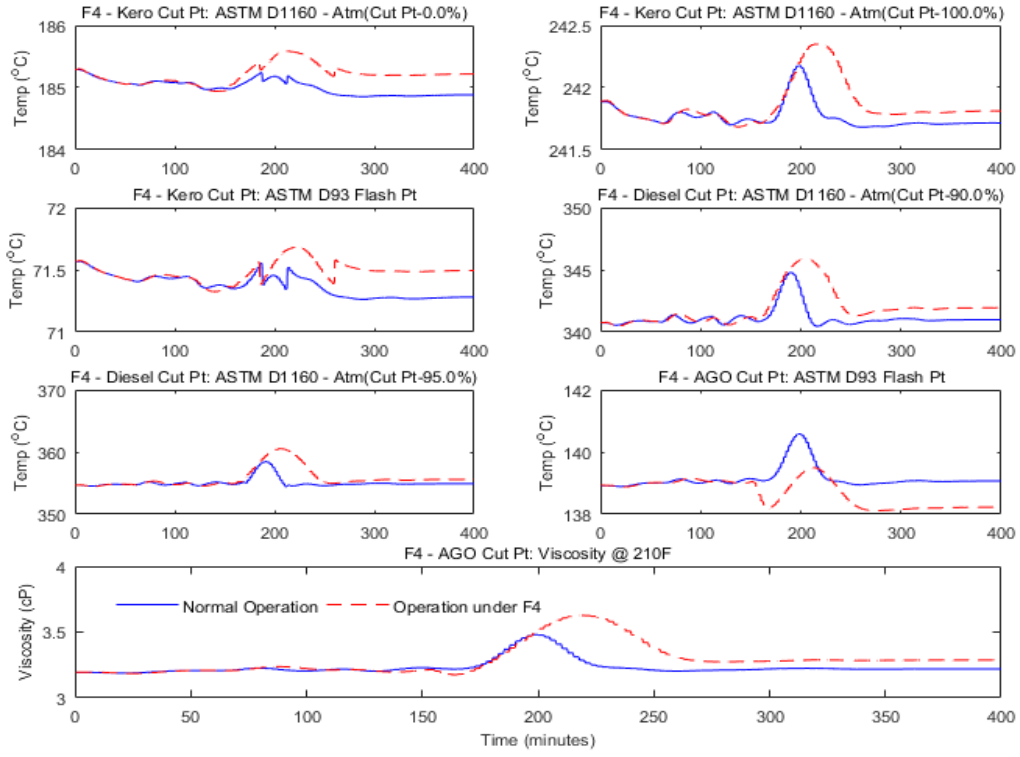


Figure 4.35: Plots of the product quality variables under F4

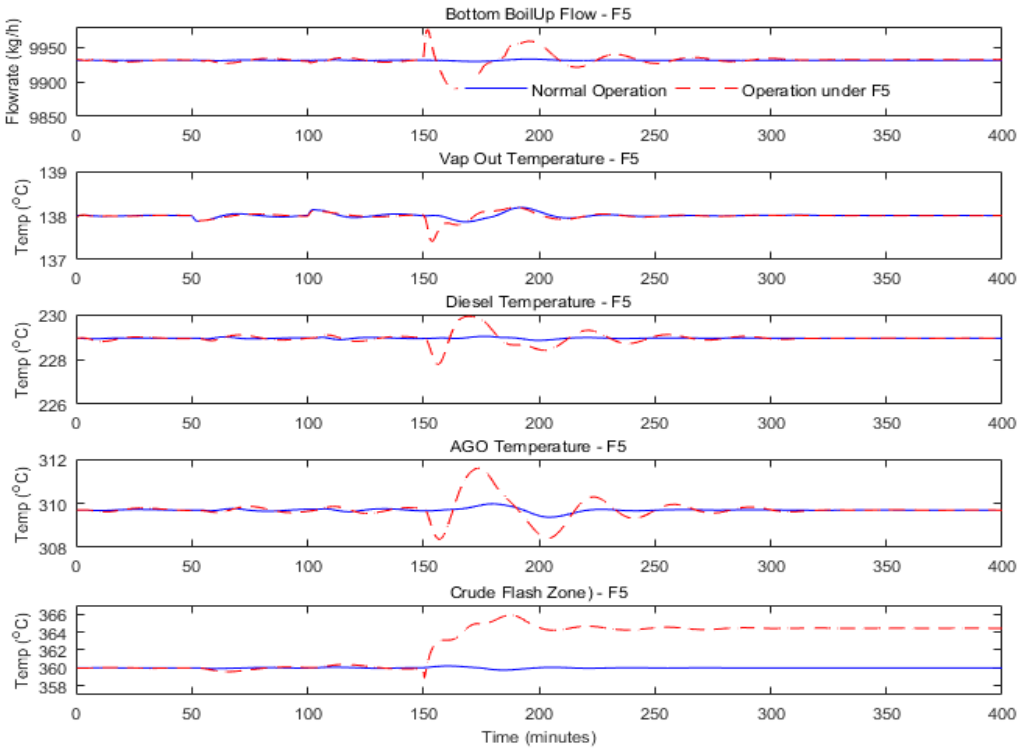


Figure 4.36: Plots of the controlled variables under F5

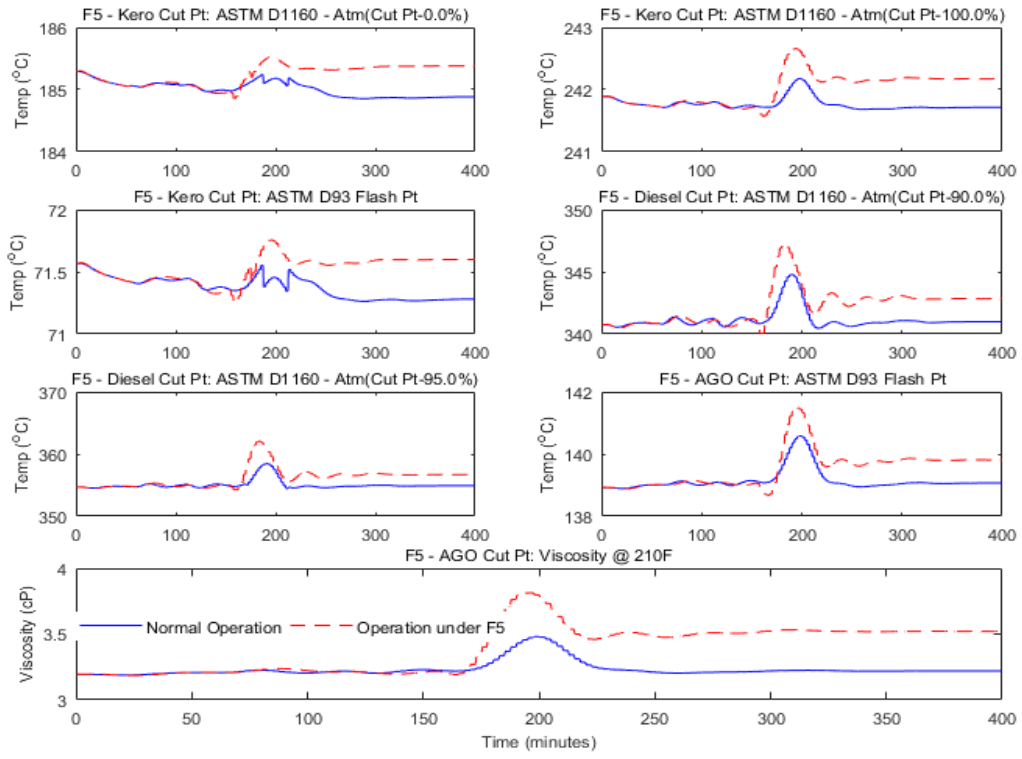


Figure 4.37: Plots of the product quality variables under F5

4.4.5 Diagnostic Model Development and Faults Detection and Identification

The 1200 data points collected for all the seventy-one process variables monitored during the CDU fault-free simulation in MATLAB are used to develop a DPCA diagnostic model for the system. First, measurement noises are added to all the variables except the feed charged to product flow ratios. The data is then scaled to zero means and unit variance. The procedures described in Section 2.4.2 are followed using 800 samples out of the 1200 collected during normal operating conditions to develop the fault detection and diagnostic model, while the remaining 400 data points are used to validate the model. Five principal components which account for 85.85% variation in the original data set with one time lag ($l = 1$) are sufficient to develop the dynamic PCA diagnostics model. The diagnostic model is then used to monitor the operation of the interactive dynamic CDU system under the five faulty conditions to detect and identify possible occurrence of actuator faults. Figures 4.38 and 4.39 present the T^2 and SPE monitoring statistics with control limit (red line) for the training and validating data sets and those of the five fault cases ($F1-F5$) respectively. The values of the T^2 and SPE monitoring statistics should be small and within their control limits in the absence of fault, but large enough to be detected as fault when one is present. An actuator fault is declared after the limits of both monitoring statistics are violated for eight consecutive sampling times (4 minutes) simultaneously. Actuator faults could also be declared faster if the values of the monitoring statistics are more than double those of their respective limits for two consecutive sampling period. These criteria are appropriate to eliminate declaration of false alarm, and also because of the complexity of the system being investigated.

The moment an actuator fault is detected, further fault diagnostic is carried out through contribution plots of the monitoring statistics to identify the fault. When the fault is declared, each of the principal components (PC) used to develop the diagnostic model is checked, in this case five principal components, at the point of fault declaration to identify the PC that violates its limit ($\pm$ limit for each PC). Figures 4.40 to 4.44 present the PC plots for each of the fault cases. The cumulative effects of the variables responsible for the PCs going outside their bounds as shown in Figures 4.40 to 4.44 are then presented pictorially in the contribution plots. Figures 4.45 to 4.49 present the contribution plots of the five fault cases.

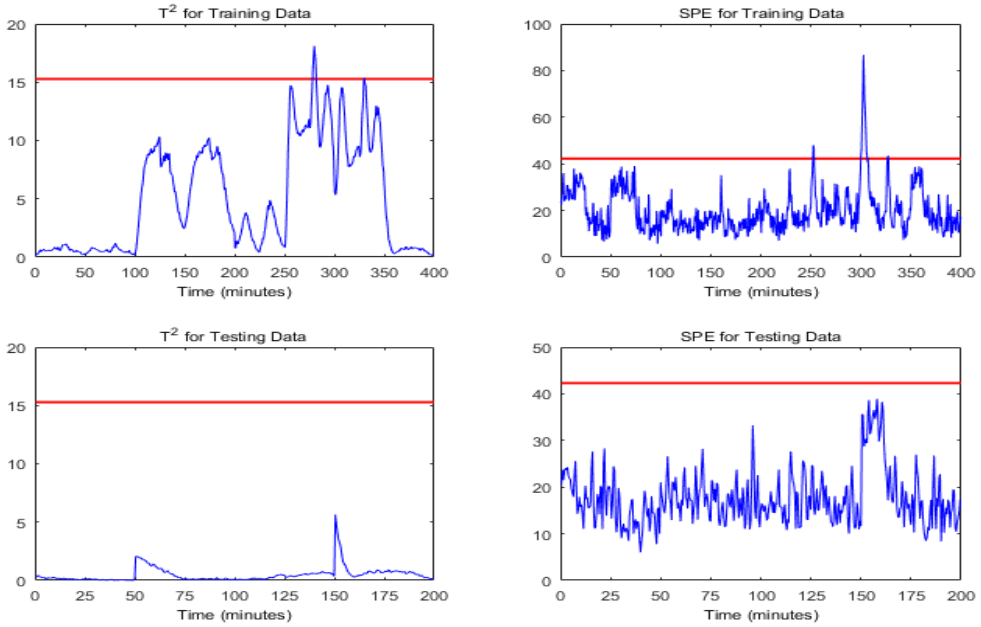


Figure 4.38: T^2 and SPE plots for the training and validating data sets

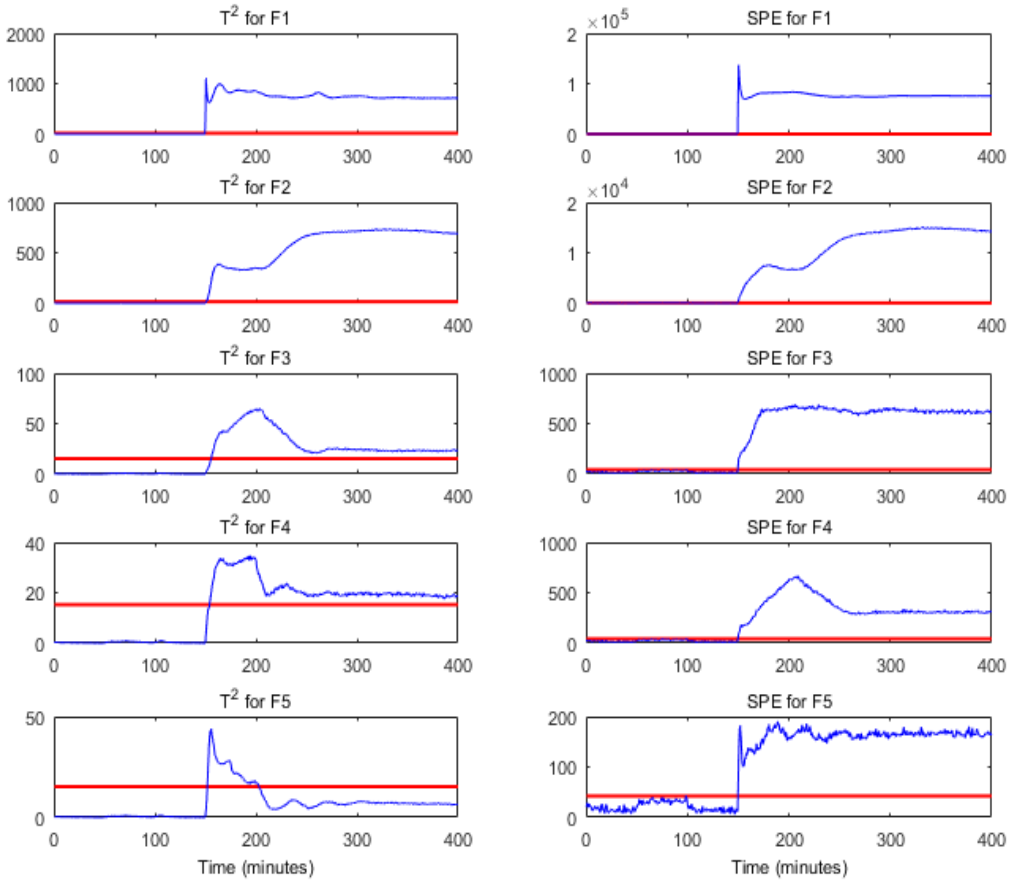


Figure 4.39: T^2 and SPE plots for faults 1 – 5 (F1 – F5)

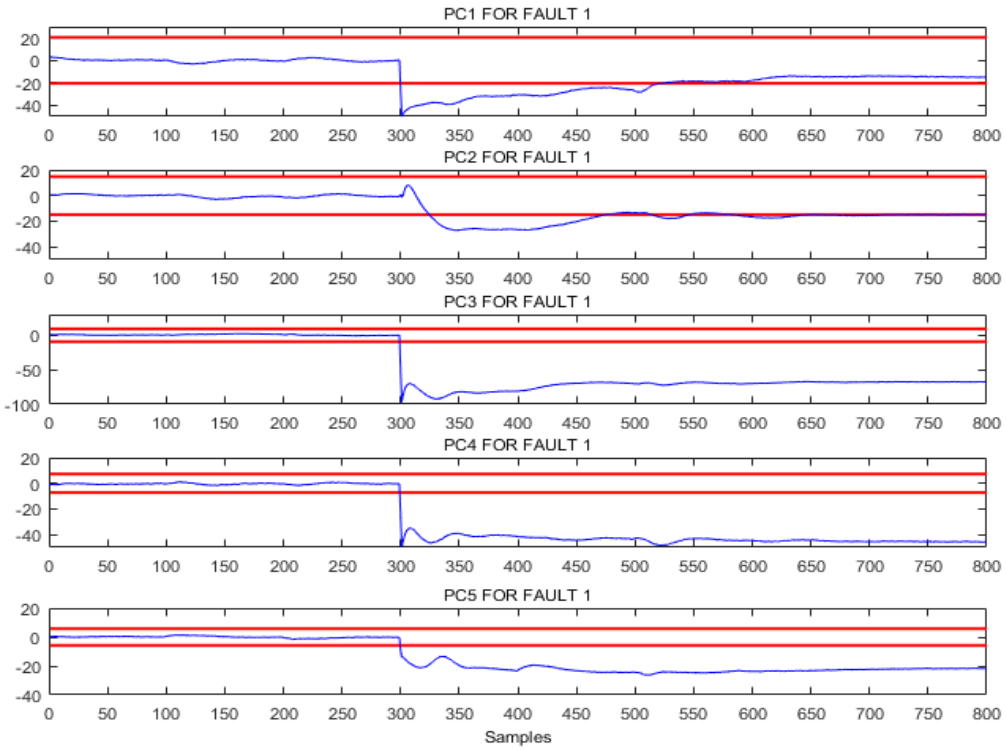


Figure 4.40: PC plots for fault F1

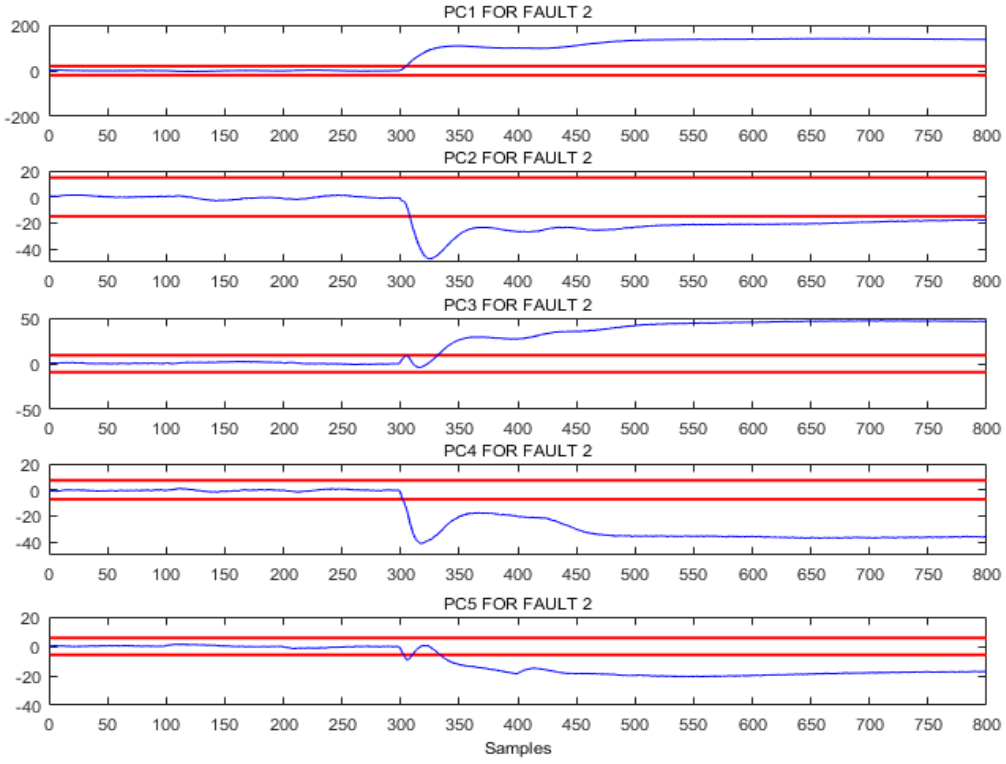


Figure 4.41: PC plots for fault F2

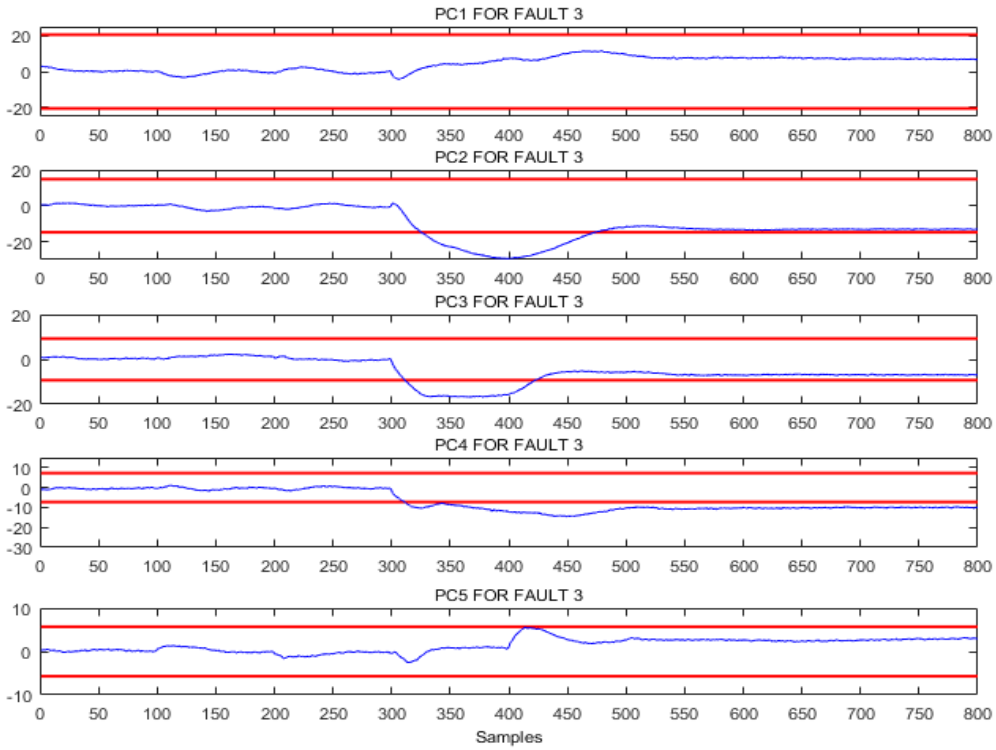


Figure 4.42: PC plots for fault F3

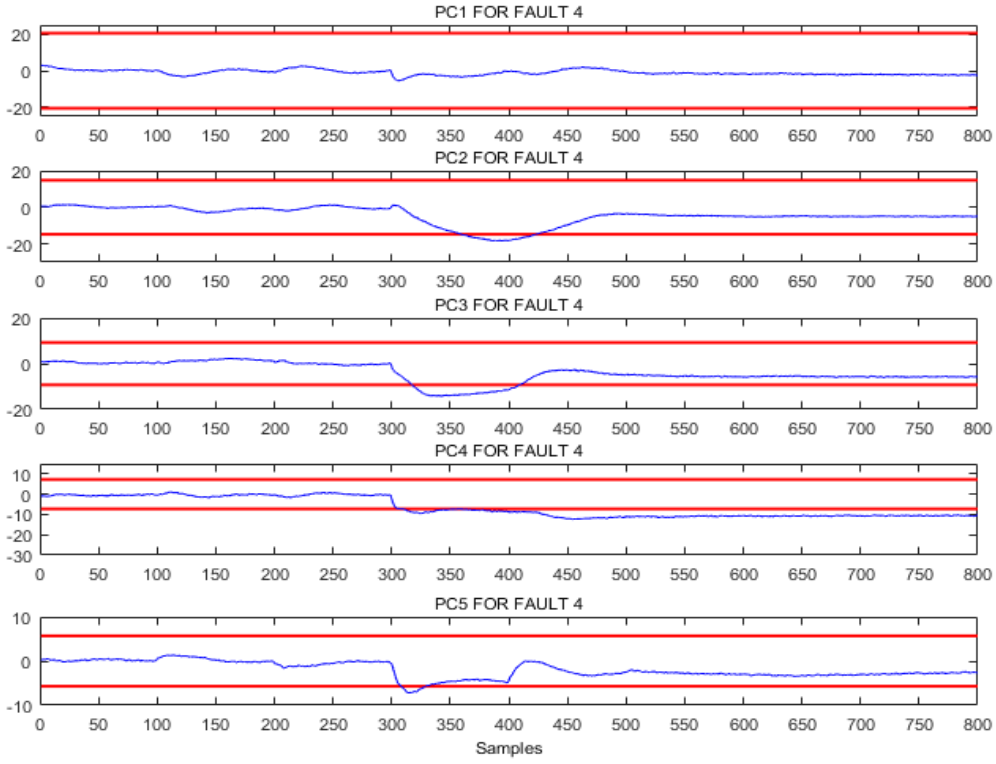


Figure 4.43: PC plots for fault F4

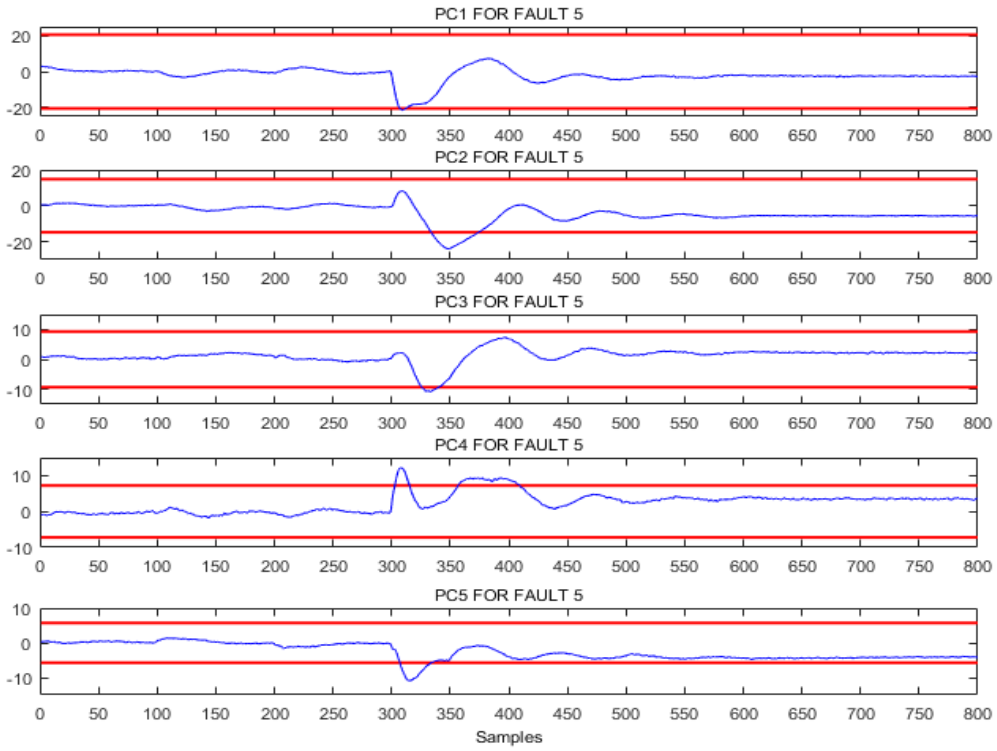


Figure 4.44: PC plots for fault F5

4.4.6 Implementation of the Actuator FTC on CDU for the Identified Actuator Faults

The structure of the control system pre-fault era may need to be reconfigured upon declaration and identification of an actuator fault, so as to preserve the integrity of the control system and most importantly ensure the CDU system is continued to be operated safely and economically in spite of the fault. Settings of some of the reconfigured controllers may also need to be adjusted and the back-up feedback signals switched as appropriate. As discussed in the previous sections, not all of the five actuator faults investigated in this case study can be accommodated. This is due to the non-availability of suitable manipulated variable that could be used to accommodate the faults. Faults $F1$ and $F5$ as described in Table 4.17 could not be accommodated as there are no suitable manipulated variables that could be used to sub-optimally control them. However, fault cases $F2 - F4$ are sub-optimally accommodated when identified using different manipulated variables as presented in Table 4.16. To accommodate the $F2$ actuator fault, the error signal generated by the reconfigurable actuator FTC for the fault-free automated CDU system during normal operation is given as:

$$e = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{p4} \\ r_{p5} \\ r_{b1} \\ r_{b2} \\ r_{b3} \\ r_{b4} \\ r_{b5} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{p4} \\ y_{p5} \\ y_{b1} \\ y_{b2} \\ y_{b3} \\ y_{b4} \\ y_{b5} \end{bmatrix} \tag{4.24}$$

where r_{pi} and y_{pi} $(i = 1 \ldots 5)$ are the system reference points and the primary controlled outputs respectively; r_{bi} are the reference points backup signals and y_{bi} are the corresponding outputs backup feedback signals. To accommodate reflux flow actuator fault $(F2)$, the furnace heat flow control valve (u_5) is reconfigured and its controller settings (settings in the rectangular boxes) adjusted as presented in Table 4.15. The error signal for the reconfigured actuator FTC under $F2$ fault is obtained as:

$$e_{RF2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{p4} \\ r_{p5} \\ r_{b1} \\ r_{b2} \\ r_{b3} \\ r_{b4} \\ r_{b5} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{p4} \\ y_{p5} \\ y_{b1} \\ y_{b2} \\ y_{b3} \\ y_{b4} \\ y_{b5} \end{bmatrix} \tag{4.25}$$

The fault tolerant control law to accommodate $F2$ actuator fault is then given as

$$u_{F2} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_{b1} \\ u_{b2} \\ u_{b3} \\ u_{b4} \\ u_{b5} \end{bmatrix} = \begin{bmatrix} G_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & G_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & G_{b2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} - y_{p1} \\ 0 \\ r_{p3} - y_{p3} \\ r_{p4} - y_{p4} \\ 0 \\ 0 \\ r_{b2} - r_{b2} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.26}$$

Similarly, the error signals ($e_{RFi}, i = 3, 4$) and the fault tolerant control laws (u_{Fi}) under actuator faults $F3$ and $F4$ are obtained as

$$e_{RF3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{p4} \\ r_{p5} \\ r_{b1} \\ r_{b2} \\ r_{b3} \\ r_{b4} \\ r_{b5} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{p4} \\ y_{p5} \\ y_{b1} \\ y_{b2} \\ y_{b3} \\ y_{b4} \\ y_{b5} \end{bmatrix} \tag{4.27}$$

$$u_{F3} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_{b1} \\ u_{b2} \\ u_{b3} \\ u_{b4} \\ u_{b5} \end{bmatrix} = \begin{bmatrix} G_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & G_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{b3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} - y_{p1} \\ 0 \\ 0 \\ r_{p4} - y_{p4} \\ r_{p5} - y_{p5} \\ 0 \\ 0 \\ r_{b3} - y_{b3} \\ 0 \\ 0 \end{bmatrix} \tag{4.28}$$

$$e_{RF4} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} \\ r_{p2} \\ r_{p3} \\ r_{p4} \\ r_{p5} \\ r_{b1} \\ r_{b2} \\ r_{b3} \\ r_{b4} \\ r_{b5} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \\ y_{p3} \\ y_{p4} \\ y_{p5} \\ y_{b1} \\ y_{b2} \\ y_{b3} \\ y_{b4} \\ y_{b5} \end{bmatrix} \tag{4.29}$$

$$u_{F4} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_{b1} \\ u_{b2} \\ u_{b3} \\ u_{b4} \\ u_{b5} \end{bmatrix} = \begin{bmatrix} G_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & G_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & G_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & G_{b4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{p1} - y_{p1} \\ r_{p2} - y_{p2} \\ 0 \\ 0 \\ r_{p5} - y_{p5} \\ 0 \\ 0 \\ 0 \\ r_{b4} - y_{b4} \\ 0 \end{bmatrix} \tag{4.30}$$

The weightings for the primary and backup feedback signals are deactivated and activated

as appropriate in order to accommodate the identified actuator faults as shown in equations 4.24 to 4.30 so as to sub-optimally maintain the system within acceptable operating region and preserve the integrity of both the control system and the process.

4.4.7 Discussion of Results

All the five actuator faults investigated in the dynamic CDU as presented in Table 4.17 are detected using the DPCA monitoring statistics developed in Section 4.4.5 with data collected during the fault free simulation of the system. Fault case 1 ($F1$ – bottom boil-up control valve fault) was detected at sample 301, a minute after introduction on both T^2 and SPE monitoring statistics as presented in the fault monitoring statistics of the 5 fault cases in Figure 4.39. Fault case 2 ($F2$ – reflux control valve actuator fault) was detected at sample 305 (2 minutes 30 seconds after introduction) for T^2 monitoring statistic and at sample 302 (1 minute after introduction) for SPE monitoring statistic as presented in Figure 4.39. Fault case 3 ($F3$ – SS-2 steam actuator fault) was detected 9 minutes and 1 minute 30 seconds after introduction, at samples 317 and 302 for T^2 and SPE monitoring statistics respectively. Actuator fault $F4$ ($SS-3$ steam actuator fault) was detected at samples 315 and 302 for the T^2 and SPE monitoring statistics respectively, 8 minutes and 1 minute 30 second after introduction for the T^2 and SPE monitoring statistics respectively. The fifth actuator fault case ($F5$ – furnace heat flow actuator fault) was detected 4 minutes and 1 minute 30 seconds after introduction, at samples 308 and 302 for the T^2 and SPE monitoring statistics respectively.

After an actuator fault is declared, the contributions of each variable monitored in the system to the fault recorded are further investigated through the variable contribution plots of the two monitoring statistics. Contributions in excess of normal average variable contributions to the monitoring statistics at the point of fault declaration are examined to identify variables responsible for the fault. Normal average variable contributions are the average contributions recorded for each variable during normal operating conditions. To achieve this, the plot of each PC used to develop the monitoring statistics, in this case 5 PCs are examined to identify the PC that violates their limits. Figures 4.40 to 4.44 present the PC plots for the 5 fault cases respectively. Table 4.18 summarises the PCs that violate their bounds ($\pm$ limit for each PC), and the cumulative contributions of the PCs for each actuator fault are presented in Figures 4.45 to 4.49.

Figure 4.45 presents the contribution plots for $F1$, and observation from the T^2 con-

tribution plot reveals variables 55 and 53, bottom boil-up mass flow (y_1) and the CDU bottom steam flow (u_1) as the two major variables that contributed to the faulty situation.

Table 4.18: List of PCs that violate their limits for faults F1 – F5

Faults	Principal Components
F1	PC1, PC3, PC4, PC5
F2	PC1, PC2, PC3, PC4, PC5
F3	PC2, PC3, PC4
F4	PC3, PC4, PC5
F5	PC4, PC5

The SPE contribution plot also reveals variables 40, 49 and 69, that is diesel temperature, SS-2 return temperature and AGO feed ratio in addition to the actual controlled variable for the control loop as being responsible for the fault. The fault was easily identified using the two contribution plots due to significantly large changes in the contributions of those variables associated with the control loop, particularly variables 53 and 55. These two variables are directly linked to the faulty actuator. The contribution plots for $F2$ as presented in Figure 4.46 show variables 64, 49, 3, 17, 18 and 40; which are temperature of the vapour leaving the column (y_2), SS-2 return temperature, stages 2, 16 and 17 temperatures and diesel temperature respectively as being responsible for the fault according to the T^2 contribution plot. The SPE contribution plot for $F2$ also shows variable 54; the reflux mass flow (u_2) as variable with significantly large contribution in addition to variables 64, 3, and 49 which were already picked up by the T^2 contribution plots. These variables are closely connected to the faulty control loop. For instance, reduced reflux mass flow (variable 54) is a consequence of the faulty valve which had a direct negative impact on variables 64 and 3 – temperature of stages 1 and 2, leading to the identification of the actuator fault.

For the fault case 3 ($F3$), the T^2 contribution plot as shown in Figure 4.47 identifies variables 40, 49, 46 and 51; which are diesel temperature (y_3), SS-2 return temperature and mass flow, and SS-2 steam flow (u_3) respectively, as the variables with larger than average contributions to the monitoring statistics. The SPE contribution plot also identified the same variables as being responsible for the fault. Again, the identified variables are closely linked to the faulty control valve and can be mapped to the fault. Significant changes in the values of the contributions of these variables to the monitoring statistics are the direct consequence of the actuator fault $F3$. Figure 4.48 presents the contribution plots

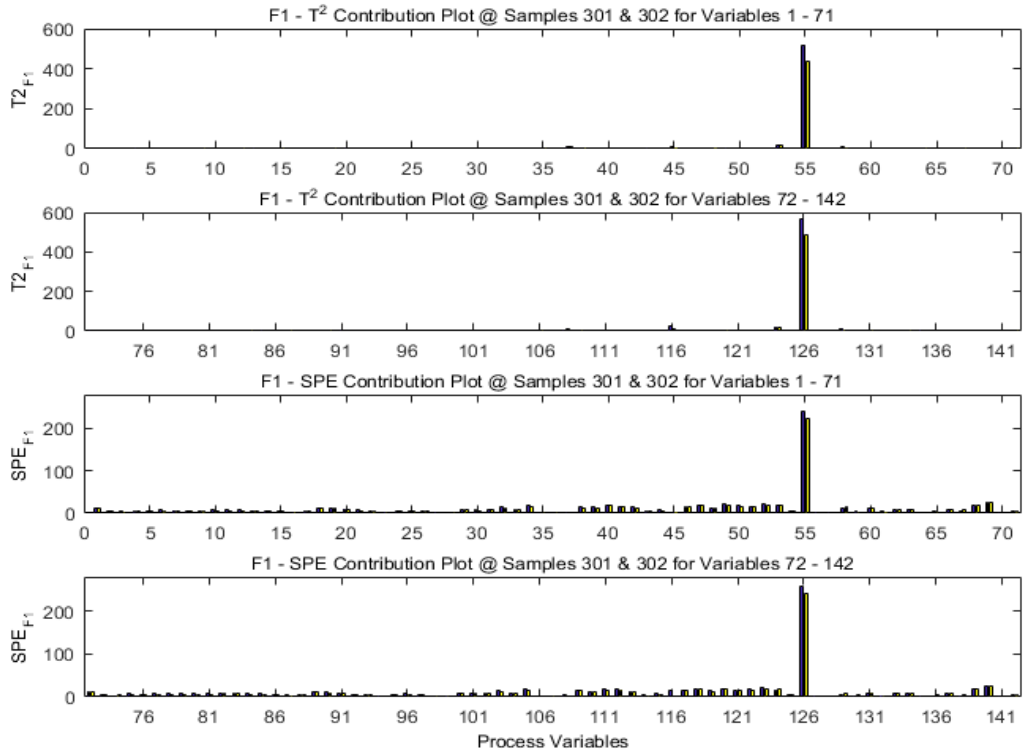


Figure 4.45: T^2 and SPE excess contribution plots for fault F1

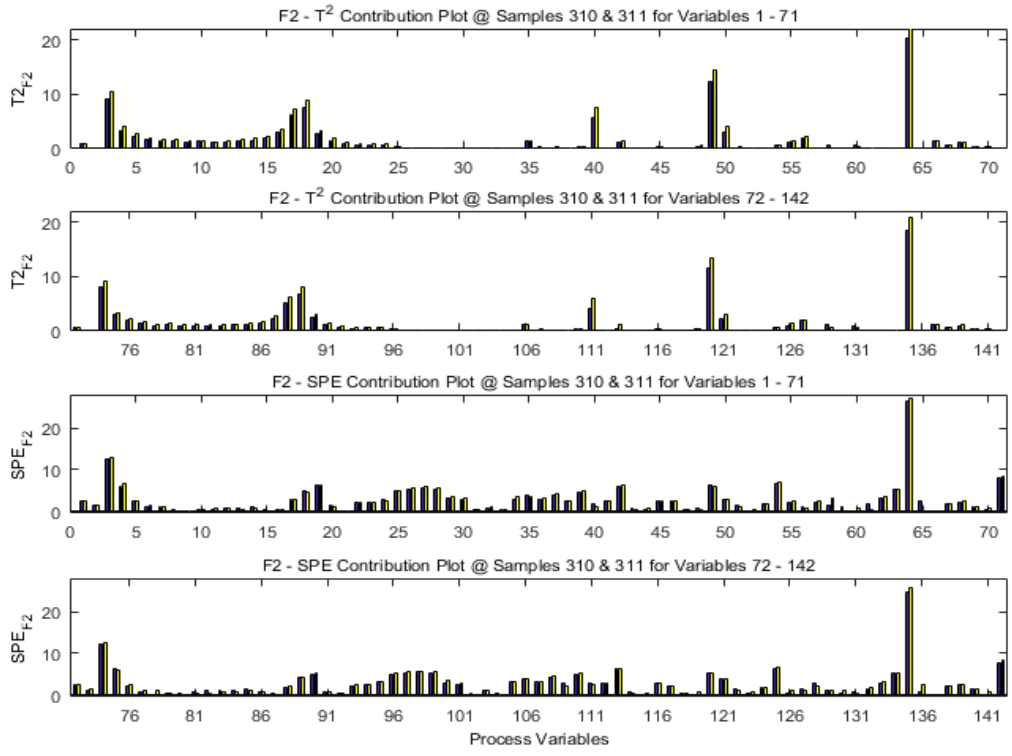


Figure 4.46: T^2 and SPE excess contribution plots for fault F2

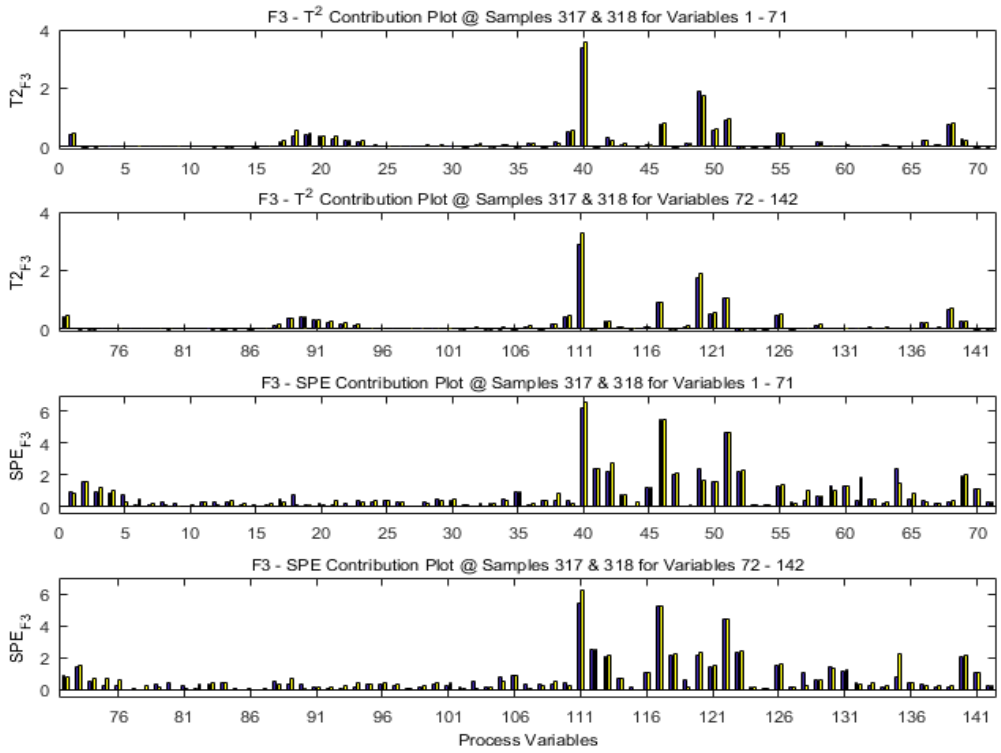


Figure 4.47: T^2 and SPE excess contribution plots for fault F3

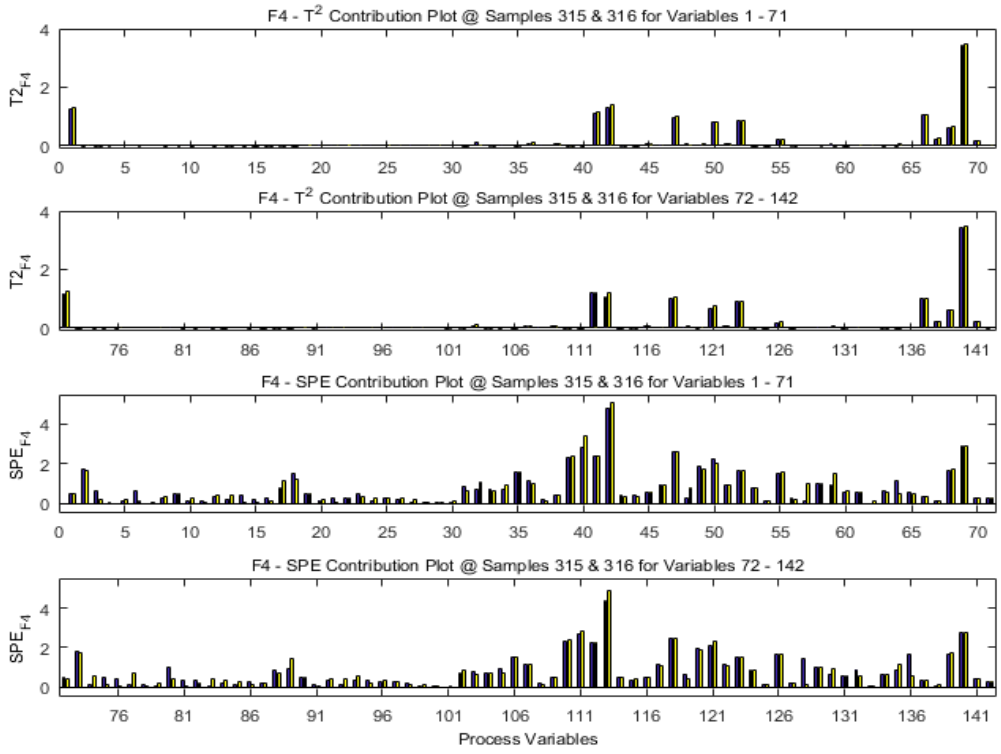


Figure 4.48: T^2 and SPE excess contribution plots for fault F4

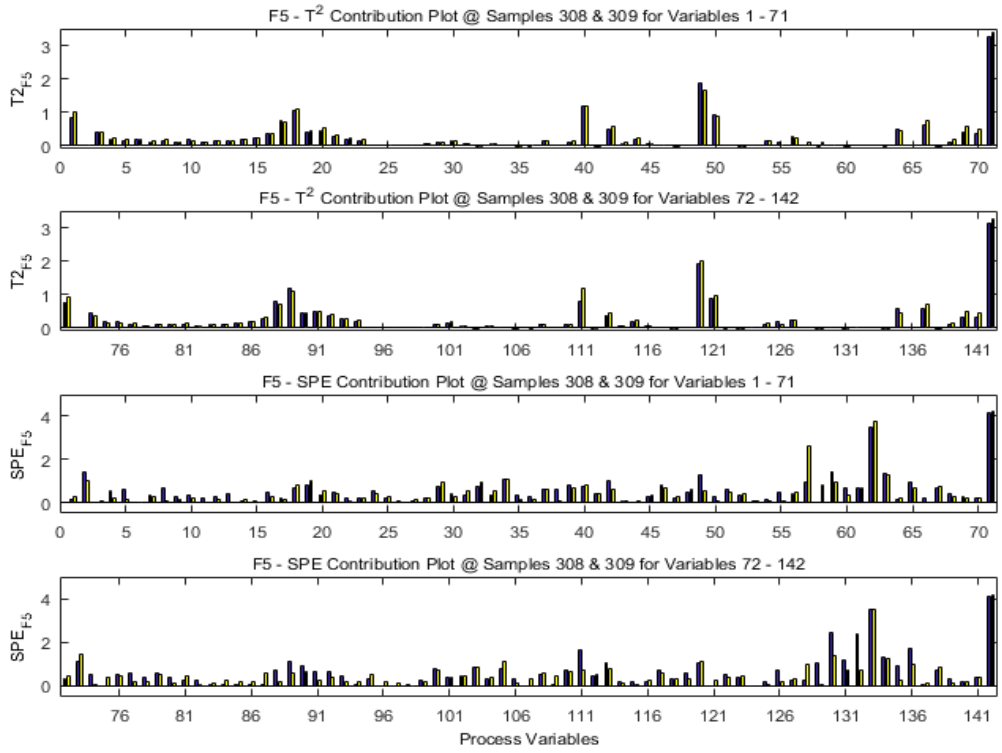


Figure 4.49: T^2 and SPE excess contribution plots for fault F5

for $F4$. The T^2 contribution plot reveals variables 1, 69, 41, 42, 47 and 66; the crude oil mass flow, AGO feed ratio, AGO mass flow, AGO temperature (y_4), SS-3 return mass flow and naphtha feed ratio respectively as being responsible for the fault. SPE contribution plot also reveals variables 41, 42, and 69 in addition to variables 40 (diesel temperature), 39 (diesel mass flow), 50 (SS-3 return temperature) and 49 ($SS-2$ return temperature) as being the contributing variables to the fault. A critical analysis of the effect of fault $F4$ on those variables shows good cause to associate the variables to the fault as they are closely linked to the faulty control loop. The T^2 and SPE contribution plots for $F5$ are shown in Figure 4.49. It reveals variable 71, the furnace heat flow (u_5) to the crude flash line as the major contributor to the fault. The T^2 contribution plot in addition to variable 71 (u_5) also shows variables 40 and 49, diesel temperature and SS-2 return temperature as being contributors while SPE contribution plot reveals variable 62, the crude flash zone temperature (y_5) as another major contributor to the fault recorded. A fault in the furnace heat flow valve directly affects the crude flash zone temperature (y_5), making the variables – fault mapping easily achievable. Adequate knowledge of the system being investigated is still required to make the connections between the variables identified by the contribution plots and the faults declared by the monitoring statistics.

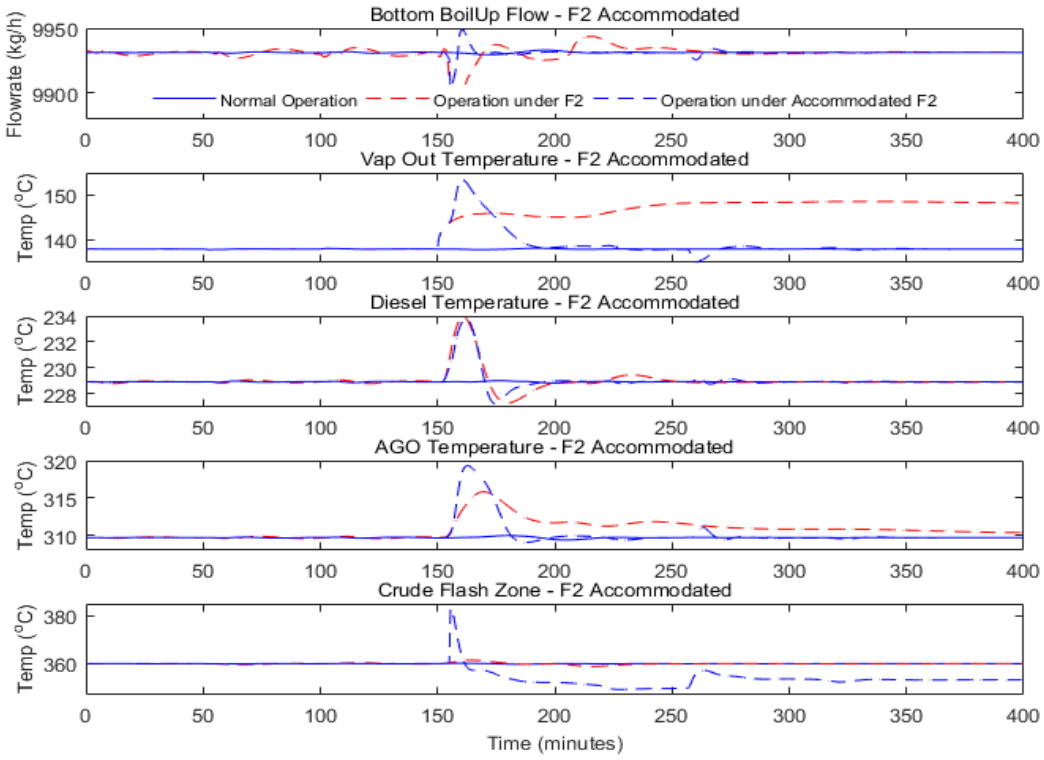


Figure 4.50: Controlled variables response plot for accommodated F2

Table 4.19 summarises the list of variables responsible for each fault.

The identified faults are accommodated according to the possible actuator FTC reconfiguration presented in Table 4.16. Fault $F1$ could not be accommodated as there was no suitable manipulated variable that could keep it at set point, even in the absence of any actuator fault, as observed during the system rigorous fault free simulation. When fault $F2$ (u_2 – faulty reflux flow control valve) is identified and isolated, non-square RGA analysis suggested y_2 (stage 1 temperature) is left uncontrolled and the remaining four controlled loops are maintained. However, further input-output pairing investigation through RGA and DRGA reveals y_2 (stage 1 temperature or temperature of the vapour leaving the column) could be controlled by manipulating u_5 (furnace heat flow rate) as presented in Table 4.16. Figures 4.50 and 4.51 show the responses of the five controlled variables and the seven products quality variables to the implementation of the actuator FTC on the dynamic CDU system to accommodate $F2$. The solid blue lines in the figures are the responses of the controlled variables and products quality variables during normal operating conditions; the dashed red lines are the responses of the same variables under faults while the dashed blue lines are the responses of the controlled variables and the product quality variables to the implementation of the actuator FTC strategy. The actu-

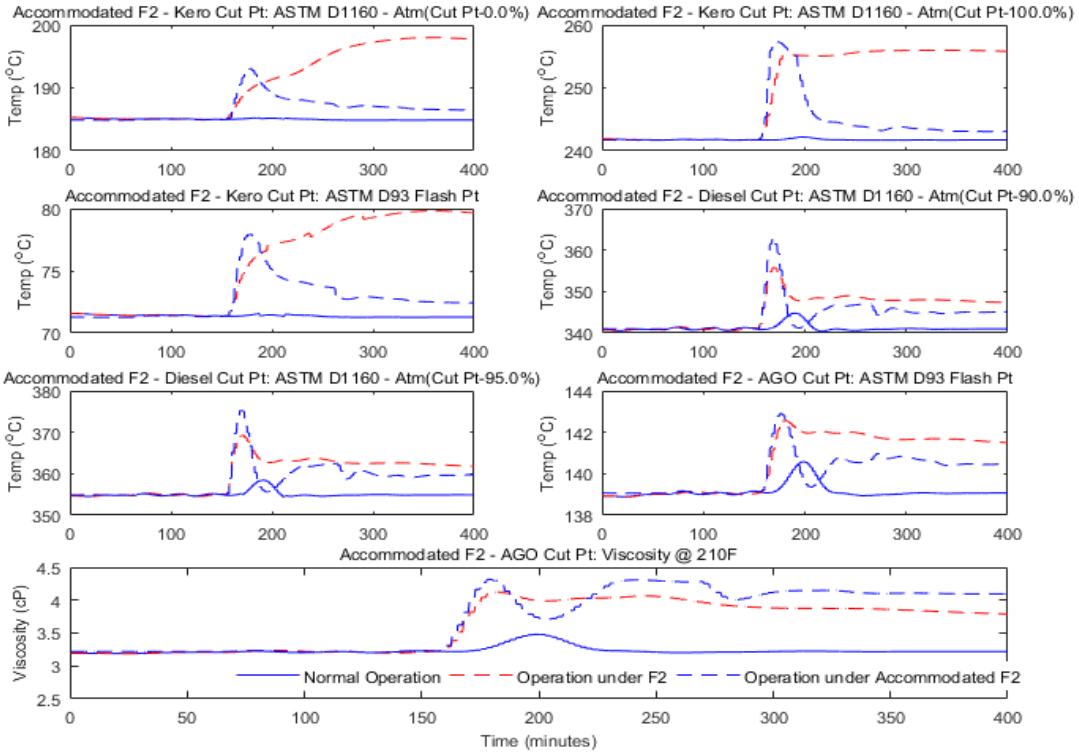


Figure 4.51: Process quality variables response plot for accommodated F2

ator FTC system is able to accommodate fault $F2$, maintaining y_2 (stage 1 temperature) at set point and reduced the effects of the fault on other controlled variables while y_5 (crude flash zone temperature) is left uncontrolled, as presented in Figure 4.50. Similarly, the effects of the fault on the product quality variables are reduced greatly, particularly for the ASTM D1160 cut points at 0% and 100% for kerosene, ASTM D93 flash point for kerosene and diesel, and ASTM D1160 cut points at 90% and 95% for diesel, as presented in Figure 4.51. No improvement is recorded on the viscosity at 210F for AGO.

Fault $F3$ (u_3 – faulty SS-2 steam control valve) is accommodated by reconfiguring the actuator FTC using u_2 (reflux flow control valve) to directly maintain y_3 (diesel temperature) at set point and the controller settings tuned as appropriate, as presented in Table 4.15. Figures 4.52 and 4.53 present the responses of the five controlled variables and the seven product quality variables to the implementation of the accommodating actuator FTC. The curves are as previously defined above. The actuator FTC is very effective in accommodating $F3$ by quickly returning y_3 (diesel temperature) to its set point and reducing the effect of the fault on other controlled variables, except for y_2 (stage 1 temperature) which is uncontrolled, as presented in Figure 4.52. The strategy is not so effective in reducing the effects of the fault on all the product quality variables as can be

Table 4.19: Variables responsible for faults F1 – F5

Faults		Variables
F1	T^2	55, 53
	SPE	55, 40, 49, 69
F2	T^2	64, 49, 3, 17, 18, 40
	SPE	64, 3, 71, 49, 54
F3	T^2	40, 49, 46, 51
	SPE	40, 46, 51, 41, 42, 52, 59
F4	T^2	69, 1, 41, 42, 47, 66
	SPE	42, 40, 39, 41, 47, 49, 69
F5	T^2	71, 49, 40, 18, 1
	SPE	71, 62, 59, 2

observed from Figure 4.53. The strategy was only able to reduce the fault effect on ASTM D1160 cut points at 90% and 95% for diesel, ASTM D93 flash point for diesel and viscosity at 210F for AGO; while ASTM D1160 cut points at 0% and 100% for kerosene and ASTM D93 flash point for kerosene further drifted away from their respective nominal values. This is because the controller reconfigured to directly maintain y_3 (diesel temperature) at set point is direct acting and increased reflux flow rate (u_2) in order to maintain y_3 as set point. This action led to reduced temperature on the top stages of the column which invariably made the product quality variables to drift further away from their nominal values. This is a decision that will be made based on the economy of the plant.

Fault case 4 ($F4$) is accommodated upon identification by reconfiguring the actuator FTC using u_3 (SS-2 steam control valve) to directly maintain y_4 (AGO temperature) at set point, leaving y_3 (diesel temperature) uncontrolled. Figures 4.54 and 4.55 present responses of the controlled and product quality variables to the actuator fault accommodating strategy respectively. The plots are as previously defined. It can be observed from Figure 4.54 that the reconfigured actuator FTC was able to maintain y_4 (AGO temperature) at set point and reduced the effects of the faults on other controlled variables. However, the results of the accommodating strategy on the product quality variables suggest it is not effective as all the product quality variables further drifted away from their respective nominal values, as presented in Figure 4.55. The responses of the product quality variables under $F4$ are better than under the accommodating strategy, and it might be better to leave the fault ($F4$) uncontrolled as suggested by the inputs – outputs pairing tools, RGA and DRGA. Fault case 5 ($F5$ – faulty furnace heat flow control valve)

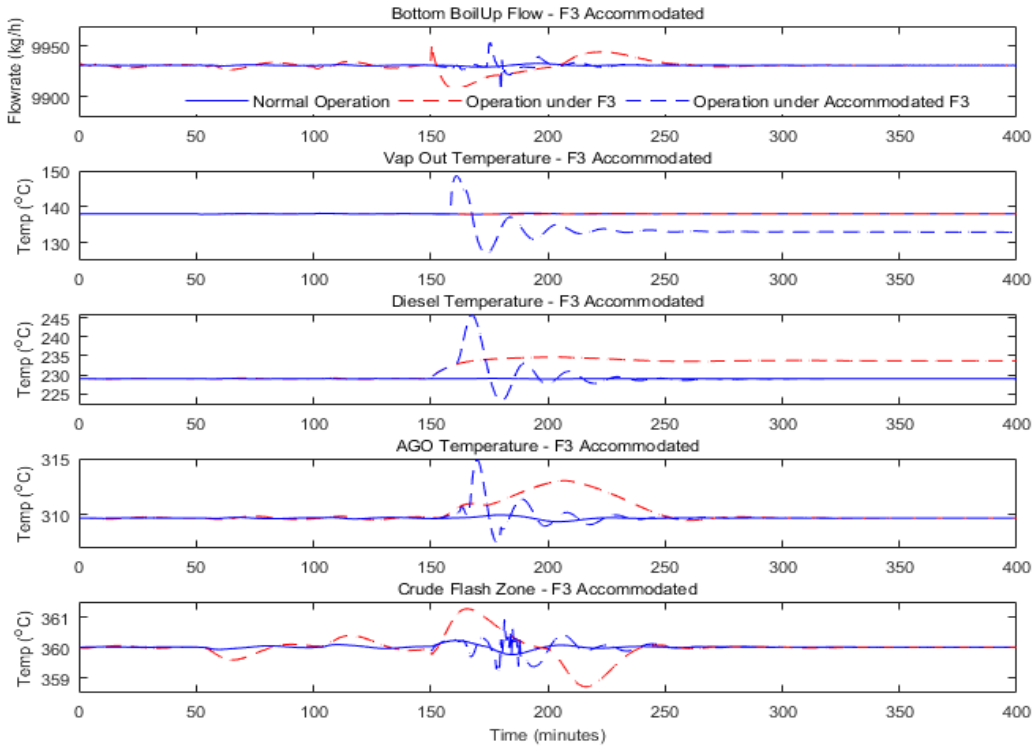


Figure 4.52: Controlled variables response plot for accommodated fault F3

could not be accommodated as there is no suitable manipulated variable to reconfigure to keep it at set point.

The actuator FTC works well in situations where there are suitable manipulated variables that could be reconfigured to accommodate the identified actuator fault. However, the reconfiguration is not always possible as observed in this case study where $F1$ and $F5$ could not be accommodated due to non-availability of suitable manipulated variable pairing. Hence, the proposed actuator FTC provides an opportunity to sub-optimally maintain the integrity of control systems in the presence of actuator faults by reconfiguring the structure of the control system to minimise the impact of the fault on the system. The sub-optimal actuator FTC strategy is system dependent and it needs no additional hardware, but needs the possible control reconfiguration structure pre-assessed and used as back-up when necessary. The accommodating strategy is not always applicable, not only when there are no suitable manipulated variables, but also when the available pairing during faulty circumstance cannot effectively accommodate the identified fault as evident in the accommodation of $F4$ (faulty SS-3 steam control valve) using u_3 (SS-2 steam control valve).

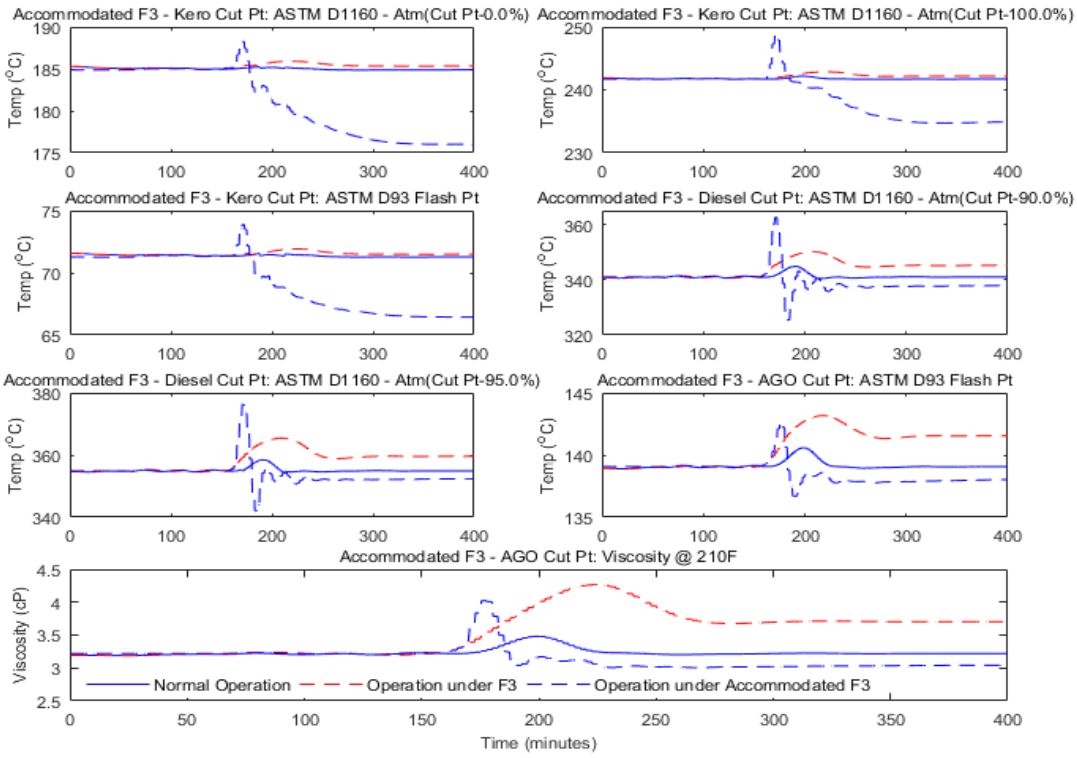


Figure 4.53: Product quality variables response plot for accommodated fault F3

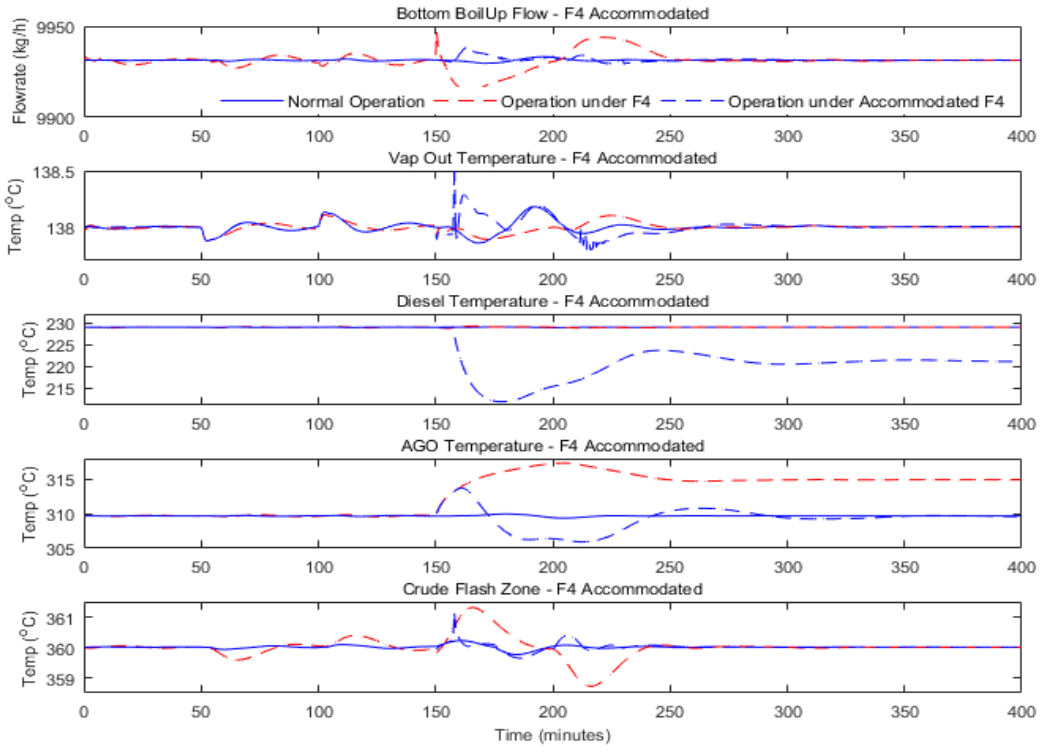


Figure 4.54: Controlled variables response plot for accommodated fault F4

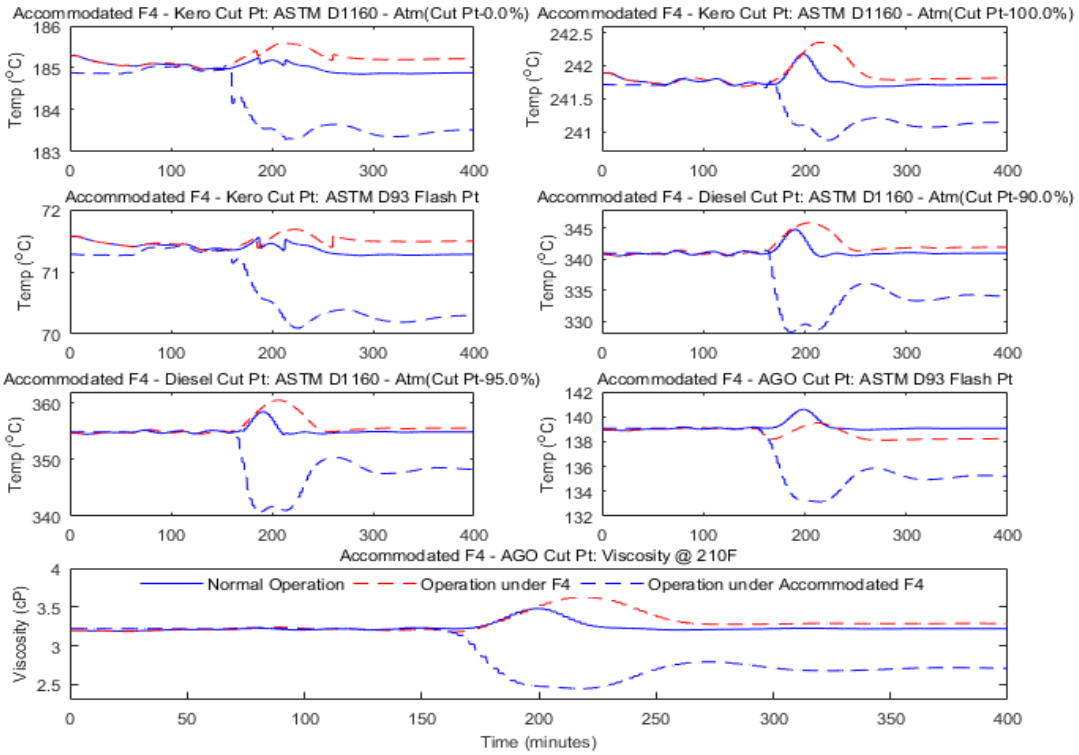


Figure 4.55: Product quality variables response plot for accommodated fault F4

4.5 Summary

This chapter details application of the proposed actuator fault tolerant controller on three distillation processes of varying complexities. The implementation of the actuator FTC on a binary distillation column to accommodate steam and reflux control valves faults was first discussed, and the strategy proved very effective in accommodating either fault using the only available manipulated variable based on the plants economic preferences. The application of the accommodating strategy on the Shell heavy oil fractionator with relatively more complex interactions than the binary column was also presented. The strategy was able to accommodate effectively the top end point (y_1) using the reflux duty actuator control valve (u_3) upon identification of top draw actuator fault (u_1), though not so effective for the other faults as presented in Section 4.3.5. And lastly, implementation of the actuator FTC on an interactive dynamic CDU was presented. The CDU represents a very complex system with severe control loops interaction, and the actuator FTC proved effective in some cases and not so effective in others, particularly when there are no suitable inputs – outputs pairing after occurrence of an actuator fault.

This strategy can help improve the availability and performance of control systems

in the presence of actuator faults and ultimately prevent avoidable potential disasters in the refinery operation with improved bottom line, profit by sub-optimally maintaining continued safe operation of the plant during abnormal events.

Chapter 5

Implementation of Proposed FTCS for Sensor Faults Accommodation on Distillation Columns

5.1 Introduction

Sensors play a critical role in any control system, in fact no good control can be achieved without accurate information on the state of the system. The health of sensors in control systems is very important, and the presence of faults in such sensors needs to be quickly accommodated, either through information from redundant sensors or through analytical means. The use of analytical means to provide alternative measurements of controlled variables could be achieved through different approaches, and it forms the core of our proposed sensor fault accommodating strategy. This thesis used dynamic principal component regression (DPCR) and dynamic partial least square (DPLS) techniques to infer the controlled variable estimates as described in Section 2.6. Implementation of the proposed fault-tolerant inferential controller (FITC) to accommodate sensor faults in distillation processes is presented in this chapter. Applications of the control scheme to binary distillation column (Lawal and Zhang, 2016a) and crude distillation unit (Lawal and Zhang, 2016b), as presented here show the effectiveness of the proposed scheme. The FTIC is a part of a complete FTCS that also includes accommodation of actuator faults. Implementation of the whole FTCS on sensor and actuator faults in a crude distillation unit is also presented.

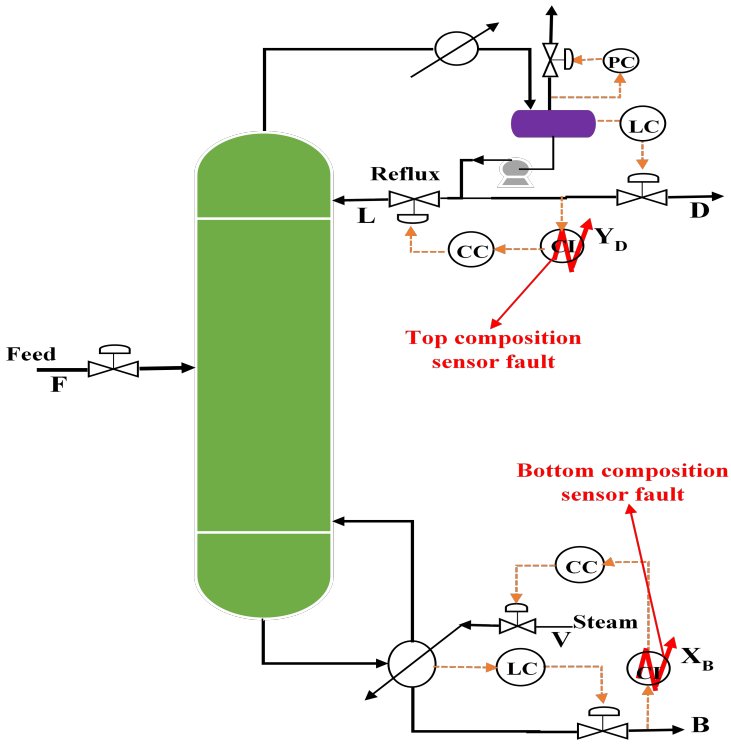


Figure 5.1: Binary distillation column with faulty Sensors

5.2 Application to Binary Distillation Column

The binary distillation column investigated in this section is the methanol-water separation column already discussed in Section 4.2.1. Figure 5.1 presents the column with faults in its top and bottom composition sensors, which are to be accommodated using the proposed FTIC.

5.2.1 Process Simulation and Faults Introduction

The distillation column is simulated during normal operating conditions for 1300 minutes with 30 second sampling time using the reflux-vapour (LV) control strategy. The top composition (Y_D) is controlled by the reflux flow rate (L) and the bottom composition (X_B) by the steam flow rate (V) to the reboiler. The column is simulated in MATLAB with a total of 2600 data points collected under the fault-free conditions. Top and bottom product compositions are measured directly by composition analysers with 5 minutes time delay. Approximately ten percent step changes in feed composition and flow rate are introduced as disturbances into the system during simulation in order to generate data

that is robust enough for fault diagnosis and to also prevent false faults detection. Four sensor faults are introduced into the system – $F1, F2, F3$, and $F4$; one at a time, two each for the top and bottom composition sensors respectively. The top and bottom composition sensor readings are held at constant outputs of 85% and 3% methanol weight fractions for $F1$ and $F2$ respectively. This represents step changes of approximately -10% and -40% for the top and bottom composition sensors respectively. In fault cases F3 and F4, the top and bottom compositions measured values are multiplied by random variables with values between 0.8 and 0.85 for top composition and values between 0.3 and 0.7 for bottom composition. This implies random biases, loss of efficiencies with relative magnitudes of between 10 – 15% and 30 – 70% for the top and bottom composition sensors respectively. Table 5.1 presents details of the sensor faults. The sensor fault cases are each simulated for 750 minutes to collect 1500 samples. Fourteen variables are monitored during the simulation – the top and bottom product compositions, the manipulated variables (reflux and steam flow rates), and the ten tray temperature measurements. Random noises with zero means and standard deviations of 0.15 and 0.001 are added to the ten tray temperature measurements and the top and bottom compositions respectively, to present the measurements as real plant data. The plots of the ten tray temperature measurements and the measured top and bottom compositions during normal operations are presented in Figure 5.2.

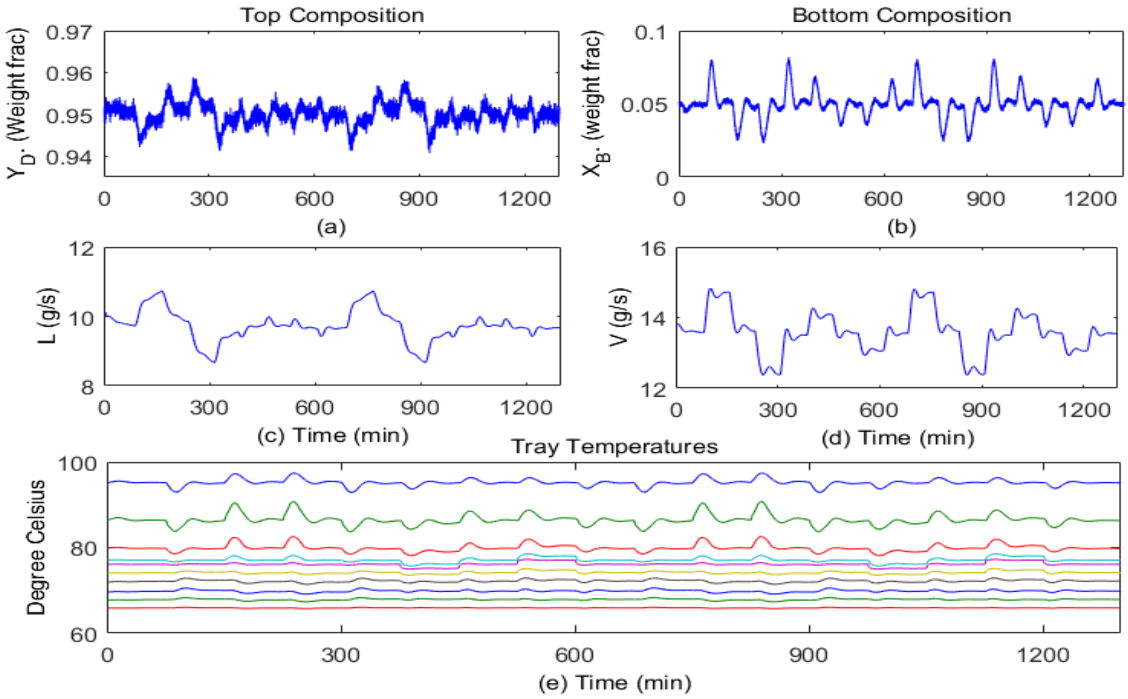


Figure 5.2: Tray temperature measurements and top and bottom compositions

Table 5.1: Binary distillation column fault list

Fault	Fault description
F1	Top composition sensor fault with sensor output stuck at 0.85 after sample 750 (static)
F2	Bottom composition sensor fault with sensor output stuck at 0.03 after sample 750 (static)
F3	Top composition sensor fault with random sensor outputs between 0.85 - 0.80 introduced after sample 750
F4	Bottom composition sensor fault with random sensor outputs between 0.035 - 0.015 introduced after sample 750

5.2.2 Soft Sensor Estimation

Estimates of the top and bottom product compositions at time t are computed using the measured uncontrolled secondary variables from the column; these variables are the ten tray temperature measurements at times t and $t-1$ through DPLS and DPCR based soft sensors. One time lag ($l = 1$) was sufficient to adequately capture the system dynamics in this case. The first 2000 samples collected during normal operating conditions are used to develop the soft sensors, with 1200 samples used for training and the remaining 800 samples used for the predictive model validation. The data is first scaled to zero mean and unit variance to ensure that all the variables have similar magnitudes. The procedures detailed in Sections 2.6 and 3.3.1 are followed to obtain the soft sensor estimates. Figure 5.3 presents the top and bottom composition estimates for the training and validating data sets obtained using both the DPLS and DPCR. It can be observed from the plots that the two techniques produced approximately the same estimates with little or no difference at all. Hence, only the estimates obtained through DPCR is used for sensor fault accommodation in this case study.

Table 5.2 presents the top and bottom compositions model parameters (θ_{di} and θ_{bi}) and their respective prediction errors on testing data obtained using DPCR as discussed in Sections 2.6. Since the model is developed using scaled data, and the estimates are to be used on-line and in real time, there is need for the estimates to be converted back to the original scale before being implemented for fault tolerant inferential control purposes. The unscaled identified models are:

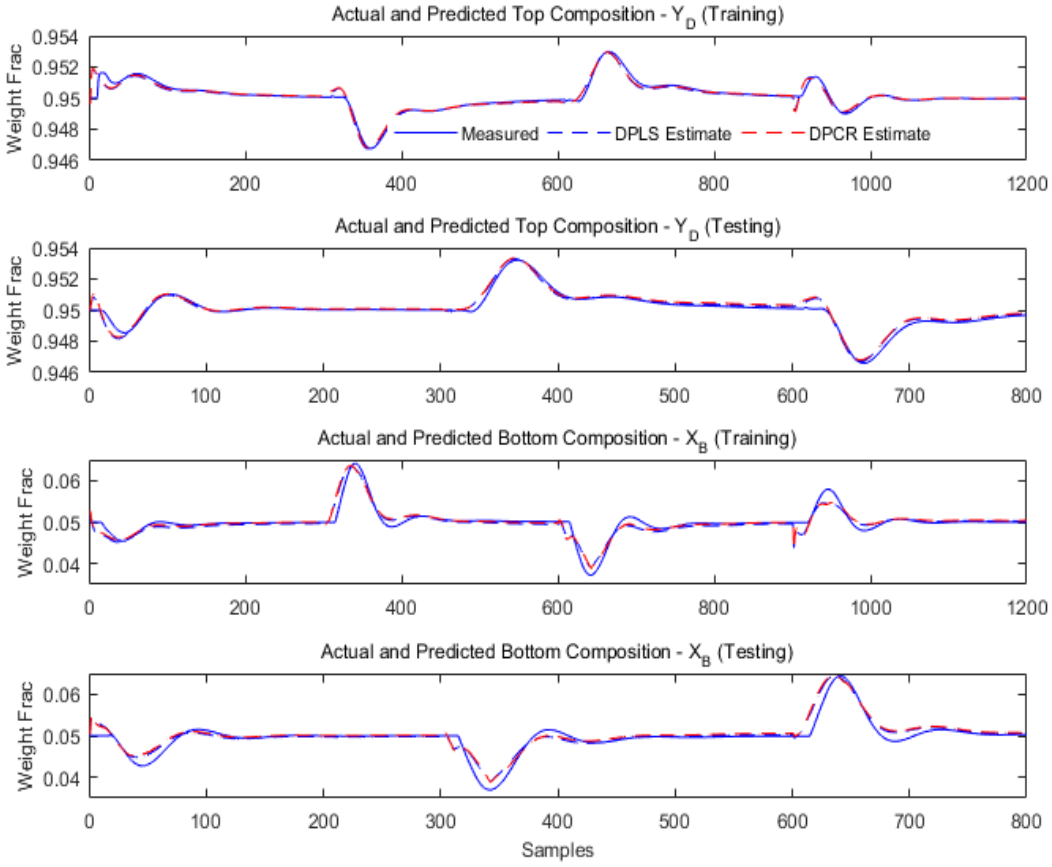


Figure 5.3: Measured and predicted top and bottom compositions

$$\hat{Y}_D(t) = \bar{Y}_D + \theta_{d1}\Delta T_1(t) + \theta_{d2}\Delta T_2(t) + \cdots + \theta_{d10}\Delta T_{10}(t) + \theta_{d11}\Delta T_1(t-1) + \theta_{d12}\Delta T_2(t-1) + \cdots + \theta_{d20}\Delta T_{10}(t-1) \tag{5.1}$$

$$\hat{X}_B(t) = \bar{X}_B + \theta_{b1}\Delta T_1(t) + \theta_{b2}\Delta T_2(t) + \cdots + \theta_{b10}\Delta T_{10}(t) + \theta_{b11}\Delta T_1(t-1) + \theta_{b12}\Delta T_2(t-1) + \cdots + \theta_{b20}\Delta T_{10}(t-1) \tag{5.2}$$

where $\hat{Y}_D(t)$ and $\hat{X}_B(t)$ are the top and bottom composition estimates at time t, $\bar{Y}_D$ and $\bar{X}_B$ are means of the top and bottom compositions, θ_{d1} to θ_{d20} and θ_{b1} to θ_{b20} are their respective model parameters; $\Delta T_i(t)$ and $\Delta T_i(t-1)$ are the deviations of the 10 tray temperature measurements from their nominal mean values at times t and $t-1$.

Table 5.2: DPCR composition model parameters

	Testing data SSE		Top Composition	Bottom Composition
No(i)	Top comp.	Bot. comp.	θ_{di}	θ_{bi}
1	3758.335	1463.879	-0.0958	-0.3259
2	1885.141	1704.118	-0.0197	-0.0135
3	610.680	272.977	0.0880	0.0063
4	596.131	282.444	0.1628	-0.0082
5	540.099	299.222	0.2160	0.0448
6	540.009	302.801	-0.1512	0.0625
7	473.246	307.980	-0.2823	0.0870
8	473.928	307.891	-0.2590	0.0628
9	427.133	306.718	-0.1345	0.0301
10	384.548	313.749	-0.0544	0.0056
11	348.722	312.184	-0.0510	-0.3986
12	338.5243**	312.247	0.0030	-0.3440
13	364.322	268.817	0.0803	-0.0877
14	366.678	267.869	0.1823	-0.0776
15	368.386	265.899	0.2149	0.0175
16	367.722	265.896	-0.1511	0.0399
17	378.624	257.507	-0.1689	0.0339
18	381.404	254.430	-0.2594	0.0652
19	381.871	252.188	-0.1595	0.0406
20	399.469	245.1322**	-0.0345	0.0256

SSE: Sum of squared error; **: Smallest SSE

5.2.3 Composition Sensor Faults Detection and Identification

The procedures described in Section 2.4.2 is used to develop a diagnostic model with one time-lagged measurements, using the first 1600 samples out of the 2600 collected under normal operating conditions while the last 1000 samples are being used for model validation. The data was scaled to zero mean and unit variance to obtain matrix X in equation 3.1 where u includes reflux and steam flow rates, y_p includes the top and bottom compositions and y_s comprises of the ten tray temperatures. Four principal components account for 82.17% variations in the original data and are sufficient to develop the DPCA diagnostic model. The diagnostic model developed for the fault free system is applied to the four faulty data sets to detect sensor faults.

Figure 5.4: T^2 and SPE monitoring plots for the fault-free system

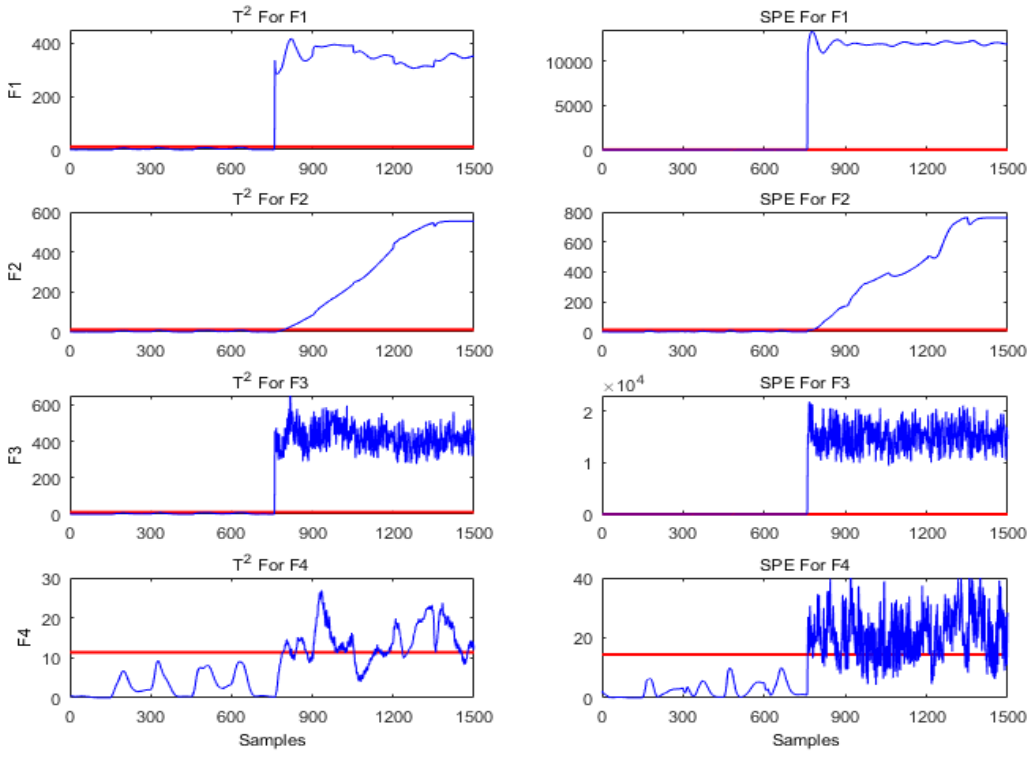


Figure 5.5: T^2 and SPE monitoring plots for the fault cases F1 – F4

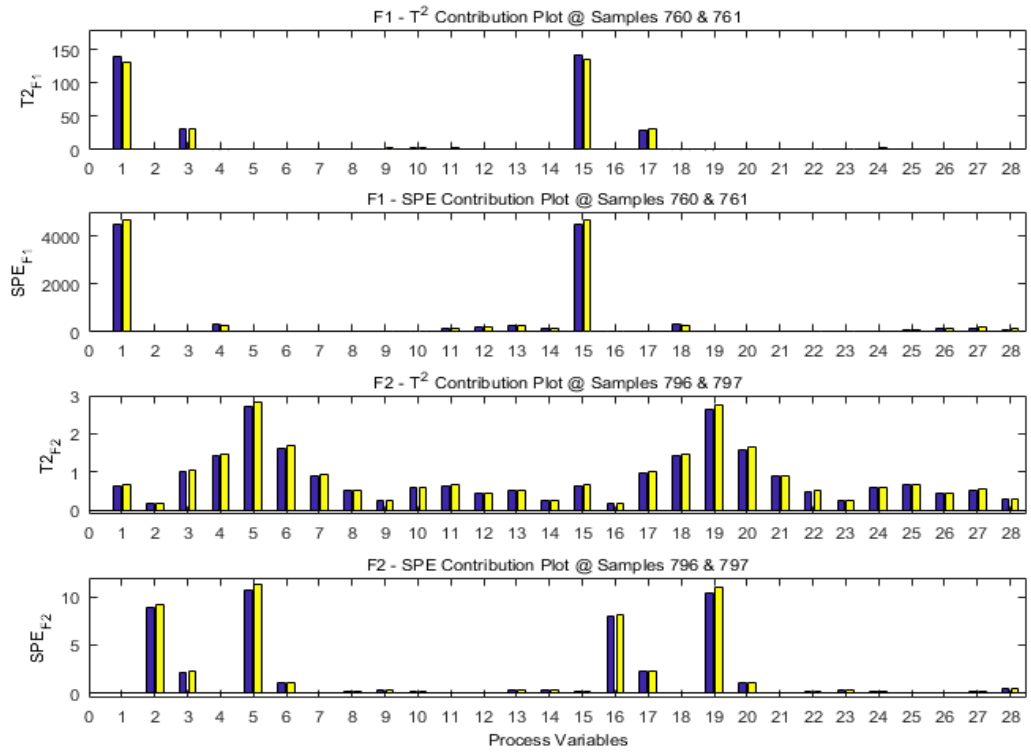


Figure 5.6: T^2 and SPE excess contribution plots for the fault cases F1 – F2

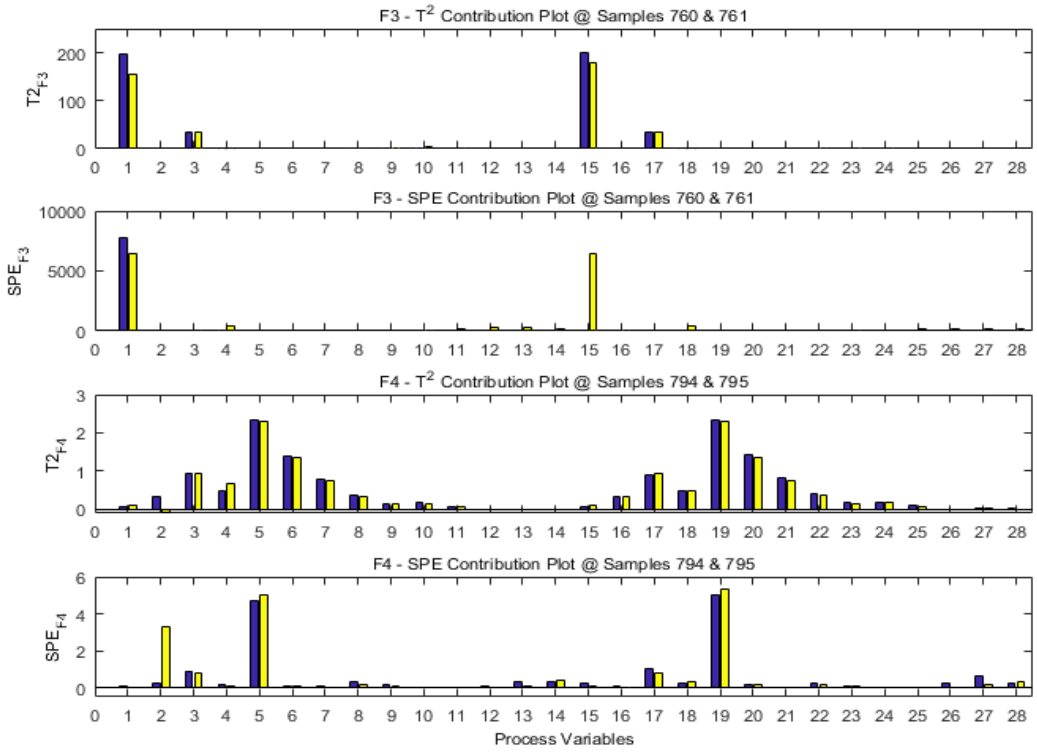


Figure 5.7: T^2 and SPE excess contribution plots for the fault cases F3 – F4

Figure 5.4 presents the T^2 and SPE monitoring plots for the fault-free system while Figure 5.5 shows those of the four fault cases ($F1 - F4$). The red line in Figures 5.4 and 5.5 is the 99% control limit. A fault is declared when the control limits for SPE or T^2 are violated for four sampling times consecutively. This helps to reduce the occurrence of false alarms. Once a sensor fault is detected, further fault diagnostic is undertaken through contribution plots to identify variables that are responsible for the fault, and ultimately isolate the fault. Figures. 5.6 and 5.7 present the contribution plots for the detected sensor fault cases. The contribution plots give indication of variables that have contributed excessively to the values of the monitoring statistics, thereby causing the monitoring statistics to go outside their normal operating bounds. Details of how the contribution plots aided the sensor fault identification are given in Section 5.2.5.

5.2.4 Composition Sensor Faults Accommodation Using FTIC

The proposed sensor fault tolerant controller is implemented on the distillation column the moment a sensor fault is identified. This is made possible through the use of the relevant redundant controlled variable signal, which is the estimate provided by the DPCR based soft sensor and is used in place of the faulty sensor output in the feedback control loop as presented in Figure 3.3. The controlled variable feedback signals (y'_p) used during normal operation is obtained using equation 3.18 as:

$$y'_p = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Y_D \\ X_B \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{Y}_{D\_DPCR} \\ \hat{X}_{B\_DPCR} \end{bmatrix} \tag{5.3}$$

When a sensor fault is identified, for instance, a top composition sensor fault (faulty Y_D value), the diagonal element corresponding to the faulty top composition sensor output changes to zero to isolate it while the corresponding diagonal element in the redundant backup feedback signal (DPCR estimate) is activated accordingly. The resulting controlled variable feedback signal during the sensor fault accommodation is then given as:

$$y'_p = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} Y_D \\ X_B \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{Y}_{D\_DPCR} \\ \hat{X}_{B\_DPCR} \end{bmatrix} \tag{5.4}$$

The sensor FTIC strategy simply replaces the faulty sensor output with the DPCR inferred estimates to maintain the integrity of the control system and that of the plant, which in this case is a binary distillation column. The same procedure is followed if the bottom

composition sensor fault is identified. Figures 5.8 and 5.9 present the accommodation of sensor fault cases $F1 - F2$ and $F3 - F4$ respectively using FTIC.

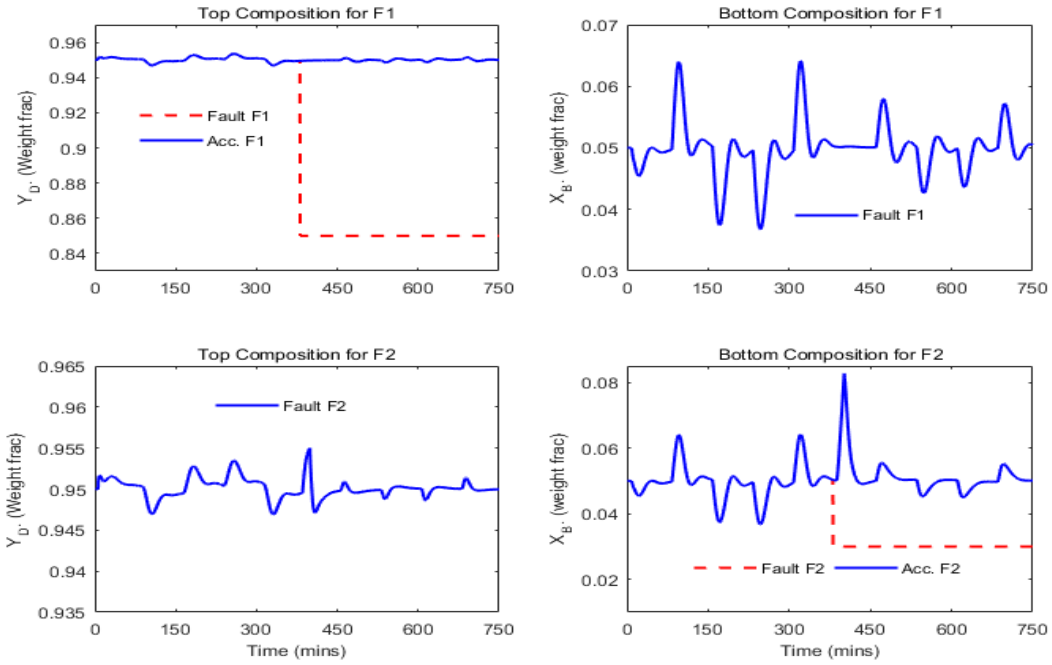


Figure 5.8: F1 and F2 sensor faults accommodation

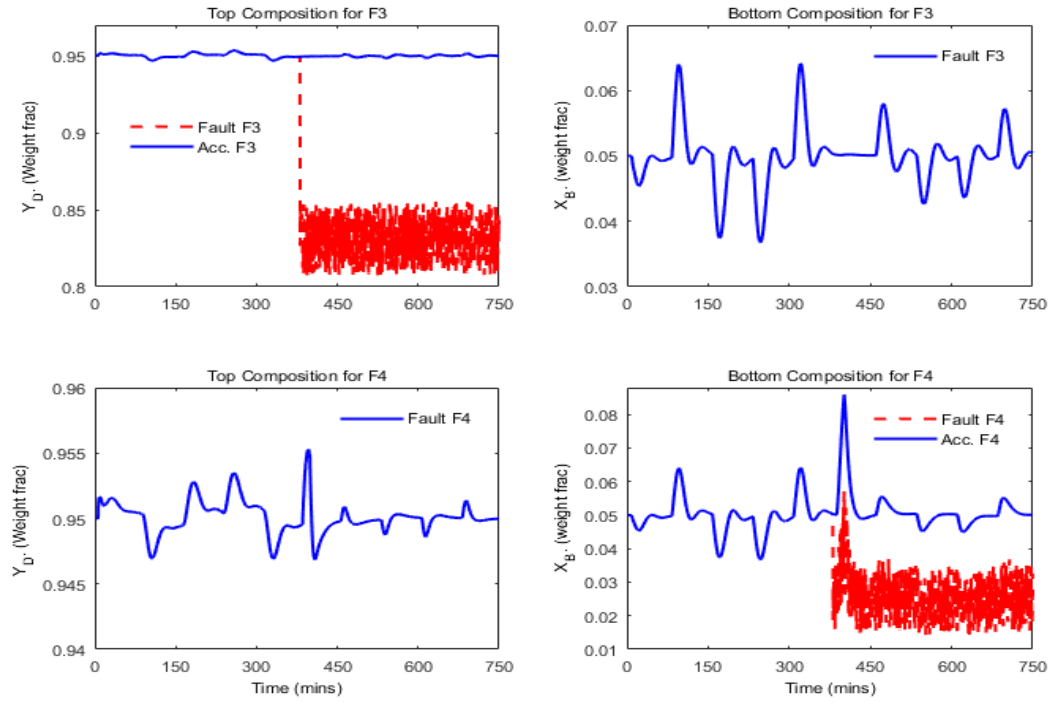


Figure 5.9: F3 and F4 sensor faults accommodation

5.2.5 Results and Discussions

All the four sensor faults investigated in the binary distillation column are detected and properly identified. From the analysis of the T^2 and SPE monitoring plots presented in Figure 5.5, sensor faults $F1 - F4$ were all detected. $F1$, top composition sensor fault, as presented in Table 5.1 was detected at sample 760 on both T^2 and SPE monitoring plots, 5 minutes after it was introduced at sample 750. Fault $F2$, bottom composition sensor fault was detected and identified at samples 796 and 781, some 23 minutes and 15 minutes 30 seconds after introduction on T^2 and SPE monitoring plots respectively. Fault $F3$ has similar characteristics with $F1$, and it was identified 5 minutes after introduction, at sample 760 on both T^2 and SPE monitoring plots. It took 22 minutes and 7 minutes at samples 794 and 764 on T^2 and SPE monitoring plots respectively, for fault $F4$ to be detected, as presented in Figure 5.5.

Upon detection of a fault, further diagnostics are undertaken to identify the actual fault that has just been flagged through the use of T^2 and SPE contribution plots, aided by a good understanding of the process under investigation. Observations from the T^2 and SPE contribution plots shown in Figure 5.6 show that top composition (variables 1 and 15) is the major contributor to the sensor fault $F1$. The fault was easily detected due to a sudden change in the top composition sensor output, causing the system to drift out of acceptable operating conditions. A similar situation was observed in sensor fault $F3$. The top composition sensor is highly sensitive, even a 5% drift in its output will result in a declaration of a sensor fault. The contribution of reflux flow rate (variables 3 and 17) to faults $F1$ and $F3$ is also significant, but much less than those of variables 1 and 15. Bottom composition sensor faults $F2$ and $F4$, as presented in Figures 5.6 and 5.7 show variables 3, 4, 5, 6, and 7 (reflux flow rate, steam flow rate, temperature of stages 10, 9 and 8) as the major contributors to the fault declared from the T^2 contribution plot. The SPE contribution plot for $F2$ is rather more conclusive, indicating variables 2 (bottom composition) and 5 (bottom stage temperature) as perhaps the only contributing variables to the faulty situation. Good knowledge of the process together with contribution plots aided the fault identification.

The proposed sensor FTIC is implemented on the binary distillation column for the fault cases $F1 - F4$ upon identification. Figures 5.8 and 5.9 present the responses of the top and bottom compositions under the sensor fault accommodating strategy. The inferential control strategy used the soft sensor estimates in place of the faulty sensor

measurements for feedback control, thereby accommodating the sensor fault. The effects of feed flow and feed composition disturbances after the faults were well compensated for by the fault tolerating control approach, as can be observed from Figures 5.8 and 5.9. The sensor FTIC strategy works quite well in preserving the integrity of the control system.

5.3 Application to Crude Distillation Unit

Real-time sensor faults detection, identification and implementation of the proposed sensor fault tolerant controller, also referred to as fault tolerant inferential controller (FTIC) on a dynamic crude distillation unit to accommodate sensor faults is presented in this section. As demonstrated here, faulty sensors need to be quickly identified and isolated in order to preserve the integrity of both the control system and the process. A detailed description of the dynamic crude distillation unit on which the sensor FTIC is implemented was given in Section 4.4.1, and Figure 5.10 presents the unit with the four sensor faults investigated in this section. Only the important parts of the CDU system that relate directly to the implementation of the sensor FTIC is described here to avoid repetition. Figure 5.11 gives a summary of the CDU schematic showing all the input and output variables into the system, including the disturbance variables (DV) and the products quality variables (PQV).

The crude distillation unit has a total of 71 variables which include flow rates and temperatures of all the streams and temperature measurements of all the column stages. There are three disturbances in the system, namely the crude composition, temperature and flow rate as shown in Figure 5.11. Crude oil is fed into the atmospheric distillation unit developed in HYSYS at a temperature of around 15^oC. The crude is then heated to 185^oC through series of heat exchangers by exchange with hot intermediate streams from the crude and vacuum columns, before entering the furnace where its temperature is raised to 360^oC, the temperature at which it enters the atmospheric column flash zone. The column has naphtha, kerosene, diesel, atmospheric gas oil (AGO) and the CDU residue as its products. ASTM D1160 cut-points at 0% and 100% for kerosene, ASTM D1160 cut-points at 90% and 95% for diesel, ASTM D93 flash points for kerosene and AGO, and AGO viscosity at $210F$ are the product quality variables used to determine the quality of the products. The nominal values for the product quality variables are presented in Table 5.3. Details of the sensor faults investigated in the system as shown in Figure 5.10 are given in Section 5.3.2.

5.3.1 Process Simulation and Controlled Variables Estimation

The CDU model in HYSYS is first integrated with MATLAB programme to build an interactive dynamic system that will, in effect aid the implementation of the sensor FTIC on the crude distillation unit. The FDD system that is used to detect and identify sensor faults and the sensor FTIC are built in MATLAB and implemented on the CDU system in HYSYS. Hence, the automation of the simulator for effective sensor FTIC implementation becomes necessary. The two systems are integrated as described in Section 4.4.2. A disturbance variable spreadsheet created in the CDU HYSYS model is accessed through MATLAB programme, and it is used to randomly introduce disturbances into the system during simulation. The same spreadsheet is used to introduce sensor faults into the system, details of the sensor faults introduction is given in Section 5.3.2. The automated CDU simulator is simulated for 600 minutes with 30 seconds sampling time under normal operating conditions to collect 1200 data points.

Table 5.3: CDU product quality variables

Process Quality Variables	Values
Kero Cut Pt: ASTM D1160 – Atm (Cut Pt-0.0%)	184.9^oC
Kero Cut Pt: ASTM D1160 – Atm (Cut Pt-100.0%)	241.7
Kero Cut Pt: ASTM D93 Flash Pt	71.28^oC
Diesel Cut Pt: ASTM D1160 – Atm (Cut Pt-90.0%)	341^oC
Diesel Cut Pt: ASTM D1160 – Atm (Cut Pt-95.0%)	354.9^oC
AGO Cut Pt: ASTM D93 Flash Pt	139.1^oC
AGO Cut Pt: Viscosity @ 210F	3.22 cP

A total of seventy-one variables including the temperature and flow rate measurements of the crude flash zone, pump-arounds, side draws, reflux stream, and the temperature measurements of all the 29 stages in the column, flow rates and temperatures of naphtha, kerosene, diesel, AGO, the CDU residue and the ratios of the feed rate to each of the products flow rates are monitored during the simulation. The system product quality variables during nominal operating conditions and a summary of the flow of information between the HYSYS model and MATLAB during simulation for FDD and on-line soft sensor estimation of the controlled variables are presented in Figures 5.12 and 5.13.

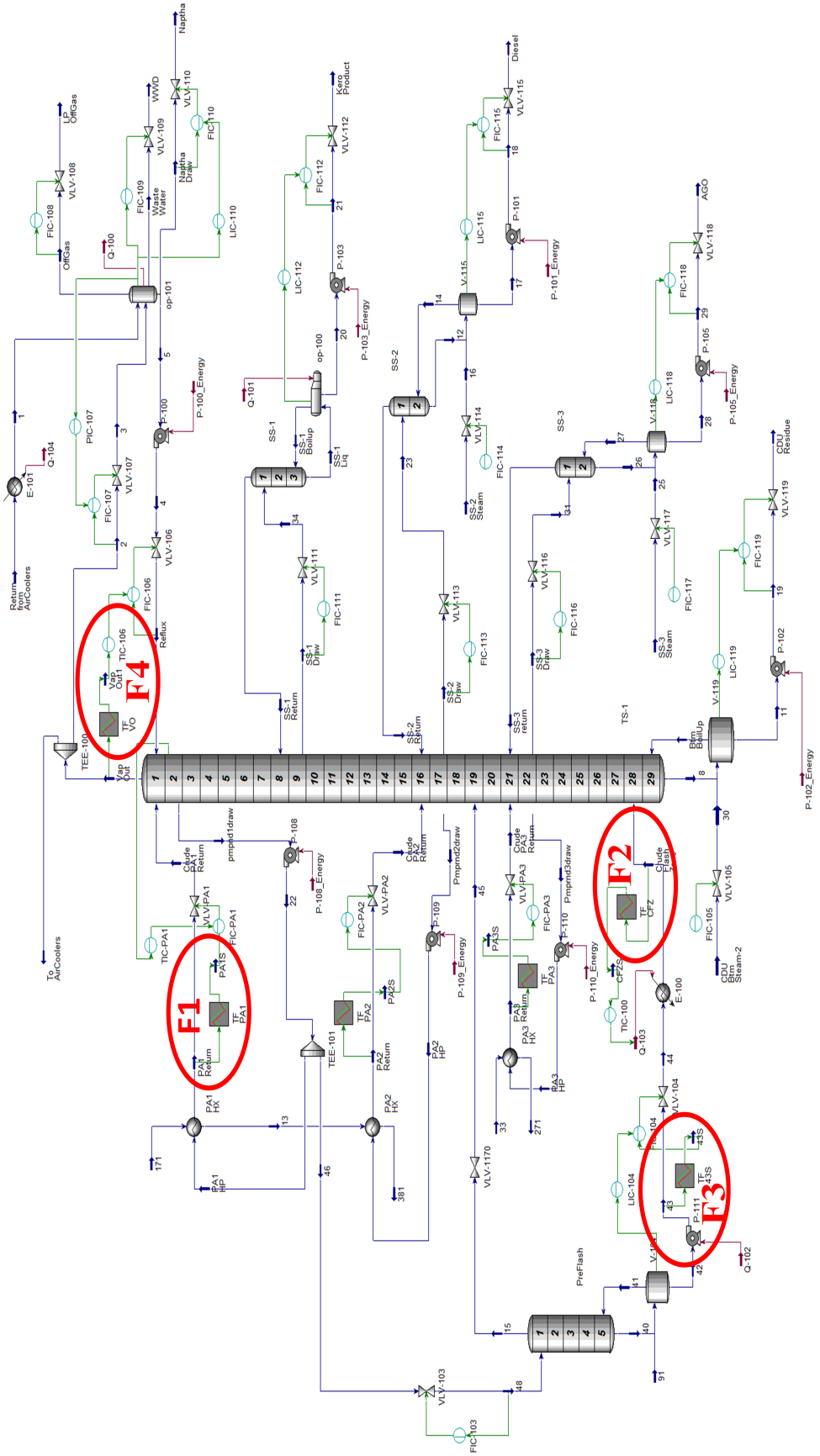


Figure 5.10: Dynamic CDU system with four sensor faults

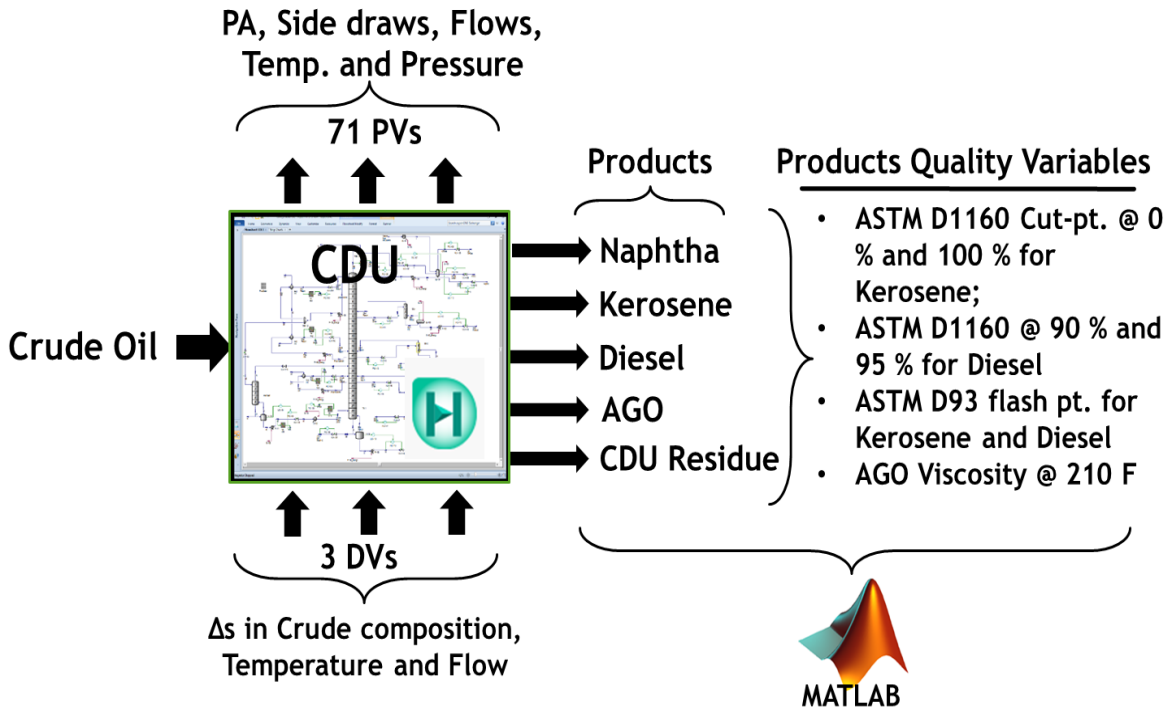


Figure 5.11: Schematic of the interactive dynamic CDU

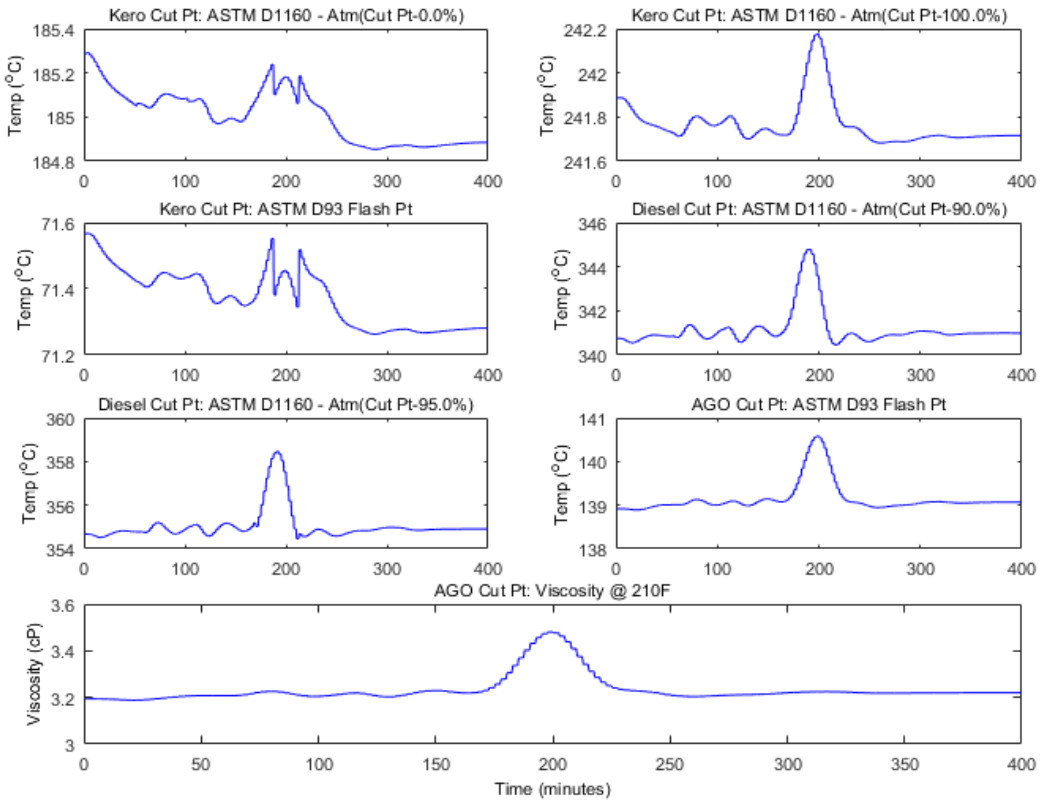


Figure 5.12: Product quality variables during normal operation

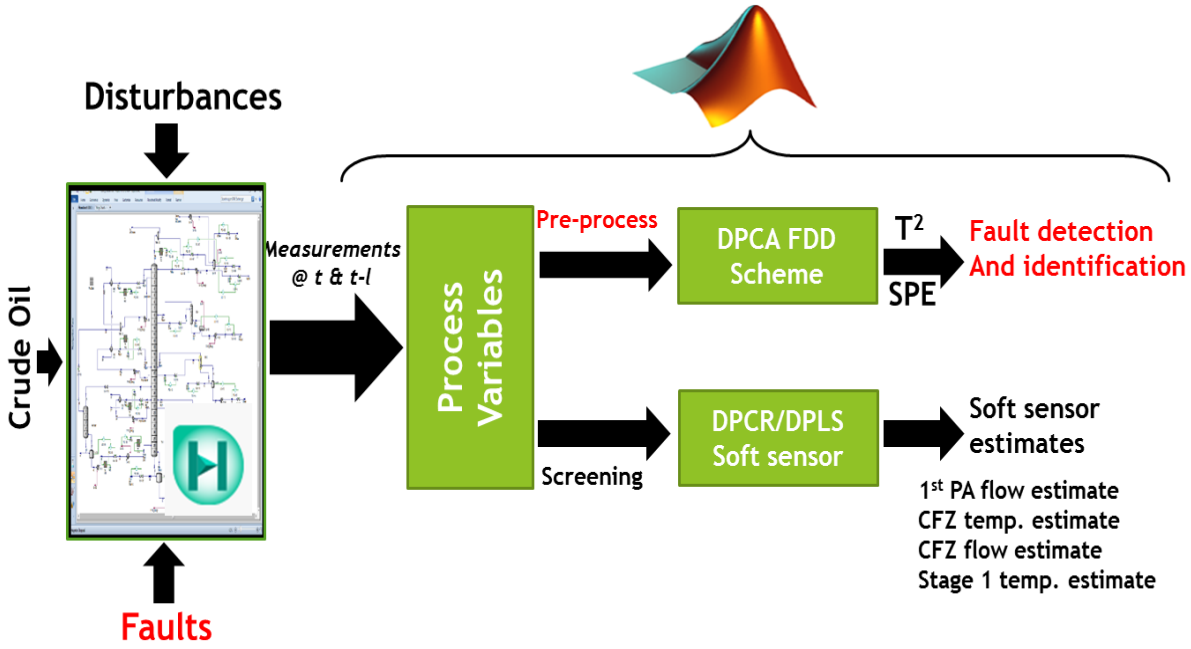


Figure 5.13: Simplified procedures for FDD and CDU PQV soft sensor estimates

Four controlled variables sensor faults are investigated in this section – (1) first pumparound flowrate; (2) CFZ temperature; (3) CFZ flowrate; (4) stage one temperature/vapour out temperature. Estimates of the controlled variables are obtained using carefully selected measured uncontrolled secondary variables at times t and $t-l$ through DPLS and DPCR based soft sensors. One time lag ($l = 1$) was sufficient to adequately capture the system dynamics in this case. 800 out of the 1200 samples collected during nominal operating conditions are used to develop the soft sensors, where 600 samples are used for training and 200 samples used to validate the soft sensor models. The data is first scaled to zero mean and unit variance to ensure that all the variables have equal weighting in the soft sensor estimates.

The procedures detailed in Sections 2.6 and 3.3.1 are followed to obtain the soft sensor estimates. The DPLS soft sensor gives more accurate estimates of the four controlled variables as presented in Figures 5.14 and 5.15, and are used to accommodate the sensor faults through the proposed sensor FTIC. However, soft sensor estimates during normal process operating conditions and faulty sensor conditions could be quite different. The quality of the soft sensor estimates is affected by the propagating fault effect on the variables used for the controlled variable estimation, just before and at the moment a fault is identified and isolated. Therefore, certain measures need to be taken during implementation of the sensor FTIC to ensure system stability and effective sensor FTIC performance.

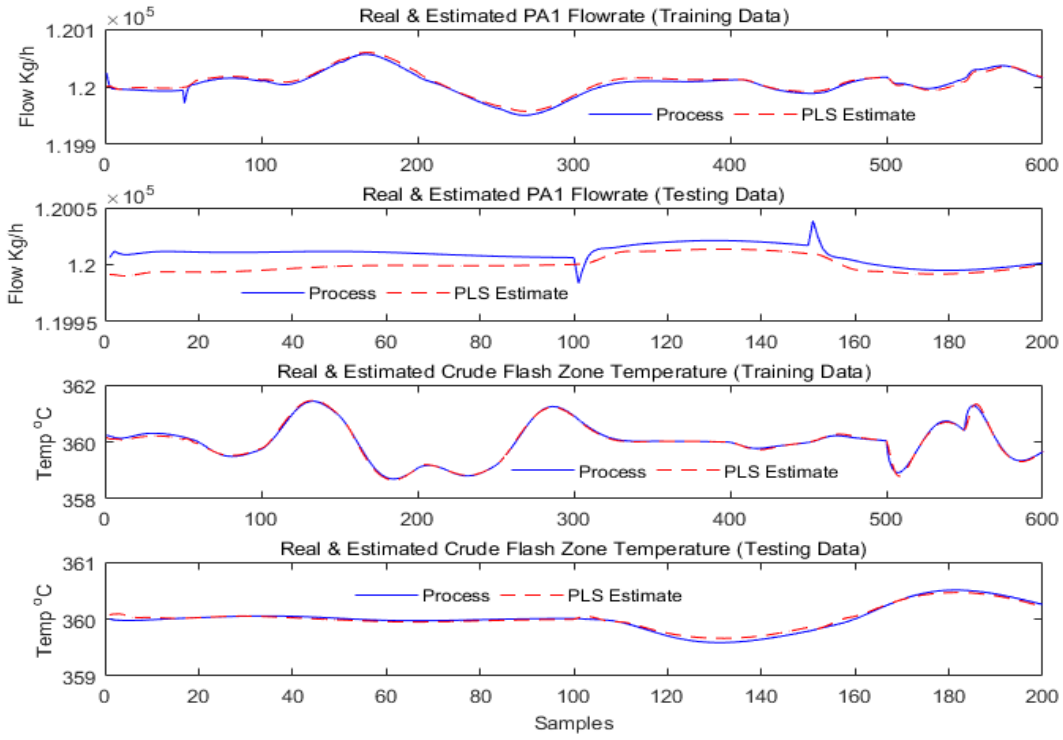


Figure 5.14: DPLS estimates of controlled variables 1 & 2

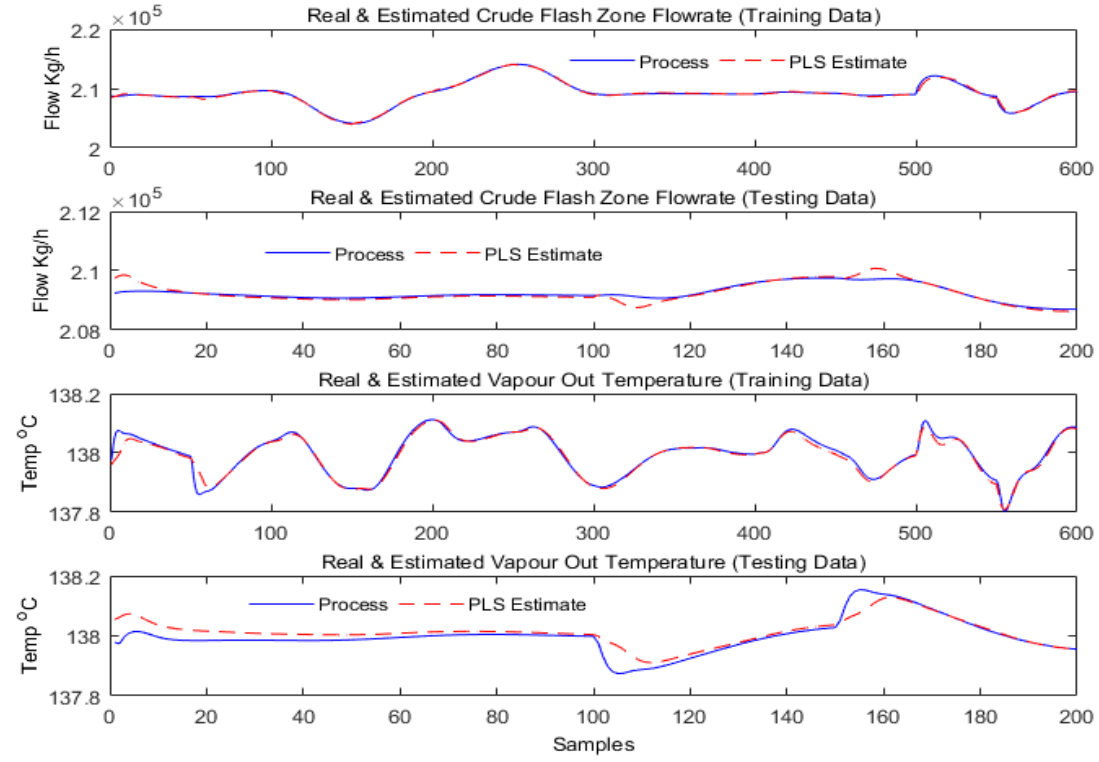


Figure 5.15: DPLS estimates of controlled variables 3 & 4

5.3.2 Flow and Temperature Sensor Faults Introduction, Detection and Identification

Two sensor faults each for flow and temperature sensor measurements are investigated in this section, as presented in Table 5.4. Sensor fault $F1$ – first pumparound flow sensor fault was introduced into the system at sample 301. The value of the sensor output is reduced by 20%, signifying a 20% bias. The second sensor fault $F2$ – CFZ temperature sensor fault also had a 20% reduction in its actual measured value and was equally introduced at sample 301. Sensor faults $F3$ and $F4$, CFZ flow rate and stage 1 temperature sensor faults both had 20% biases on their outputs and were also introduced at sample 301. A gain of 0.8 value was added to each control loop, and the sensor outputs were each multiplied by the gain to introduce the faults, as presented in Figure 5.10. The system is then simulated for 400 minutes with 30 seconds sampling time, one at a time for the four sensor faults. Figure 5.16 shows the effects of each sensor fault on their respective controlled variable, while Figures 5.17 to 5.20 present the responses of all the product quality variables to each of the sensor faults.

Table 5.4: CDU sensor fault list

Fault	Fault description
F1	1st pumparound flow sensor fault with 20% bias introduced @ sample 301
F2	CFZ temperature sensor fault with 20% bias introduced @ sample 301
F3	CFZ flow sensor fault with 20% bias introduced @ sample 301
F4	Stage 1 temperature sensor fault with 20% bias introduced @ sample 301

After the sensor faults have been introduced into the system, they need to be detected and isolated as quickly as possible to ensure the system stability and preserve the integrity of the control system. A DPCA diagnostic model is developed for the dynamic CDU using the 1200 data points collected for all the seventy-one process variables monitored during the system fault-free simulation. The procedures described in Section 4.4.5 are followed using 800 samples out of 1200 collected during normal operating conditions to develop the fault detection and diagnostic model, while the remaining 400 data points are used to validate the model. Five principal components (PC) which account for 85.85% variation in the original data set with one-time lag ($l = 1$) are sufficient to develop the dynamic PCA diagnostics model. The diagnostic model is then used to monitor the operation of the interactive dynamic CDU system to detect and identify possible occurrence of

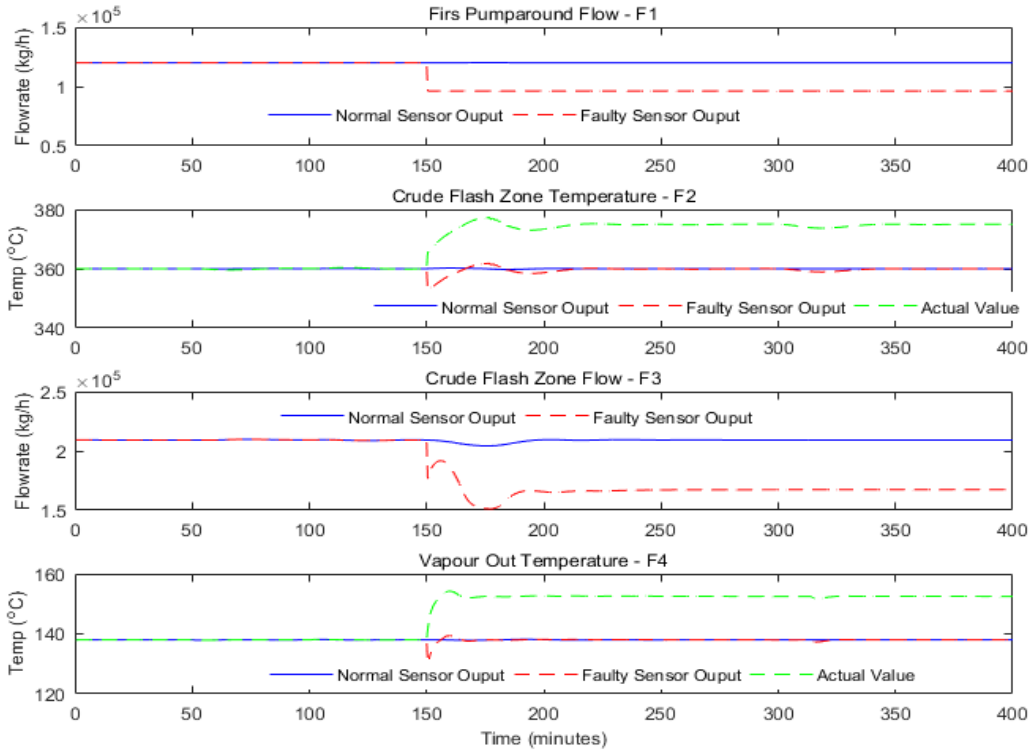


Figure 5.16: Controlled variables responses to sensor faults F1 – F4

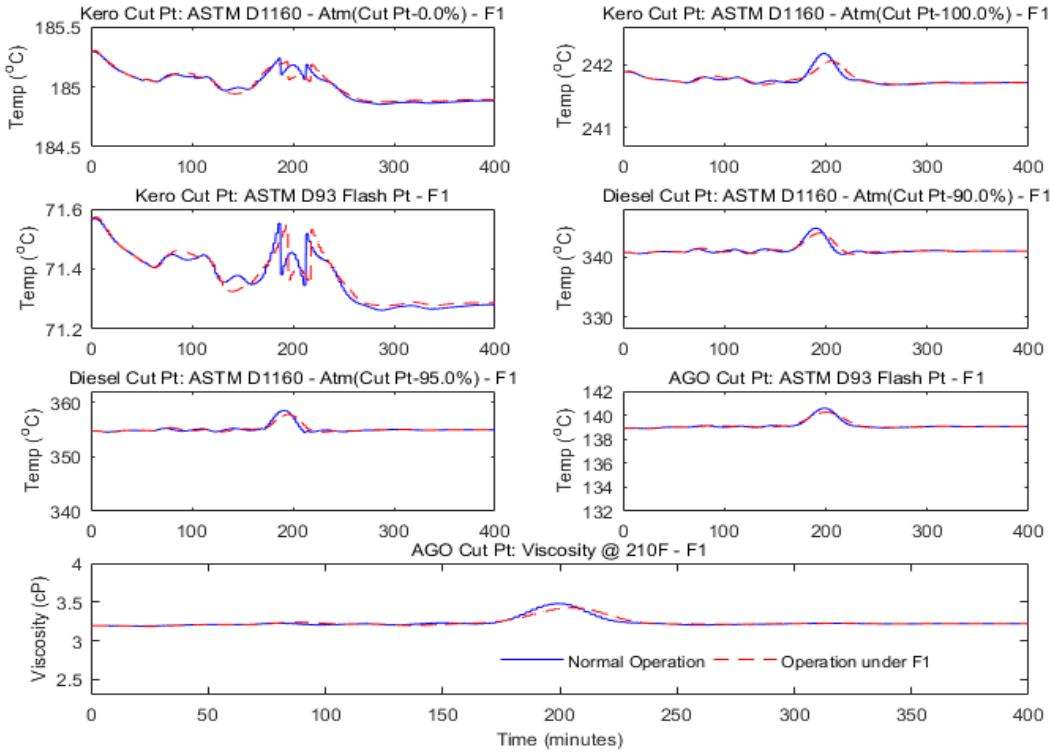


Figure 5.17: Responses of product quality variables to sensor fault F1

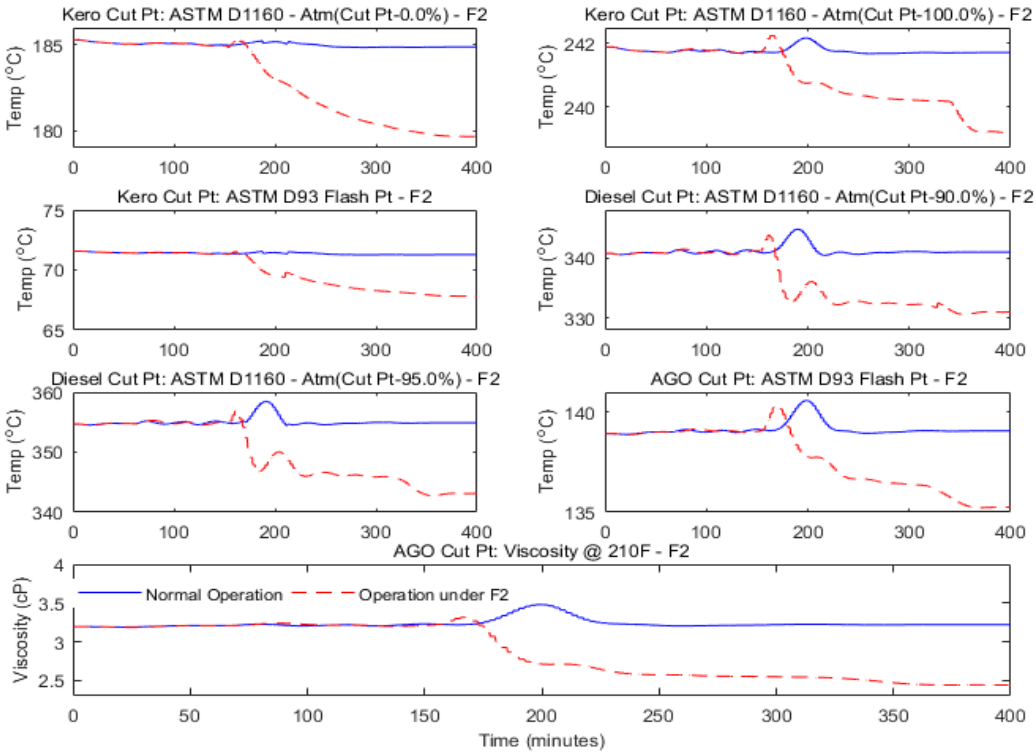


Figure 5.18: Responses of product quality variables to sensor fault F2

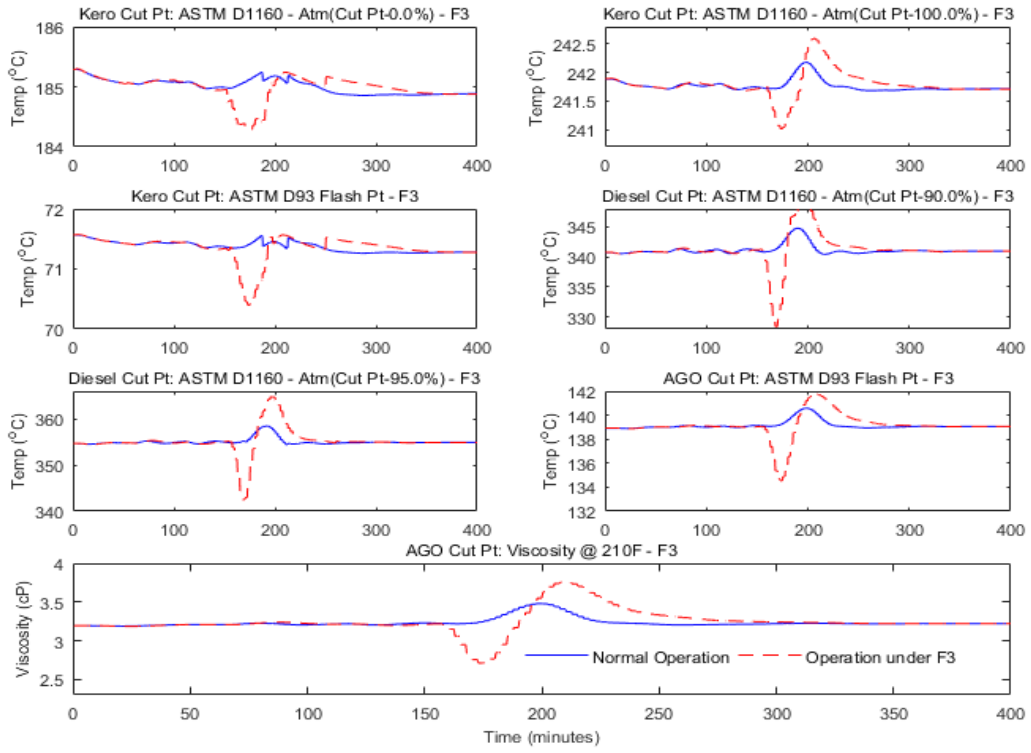


Figure 5.19: Responses of product quality variables to sensor fault F3

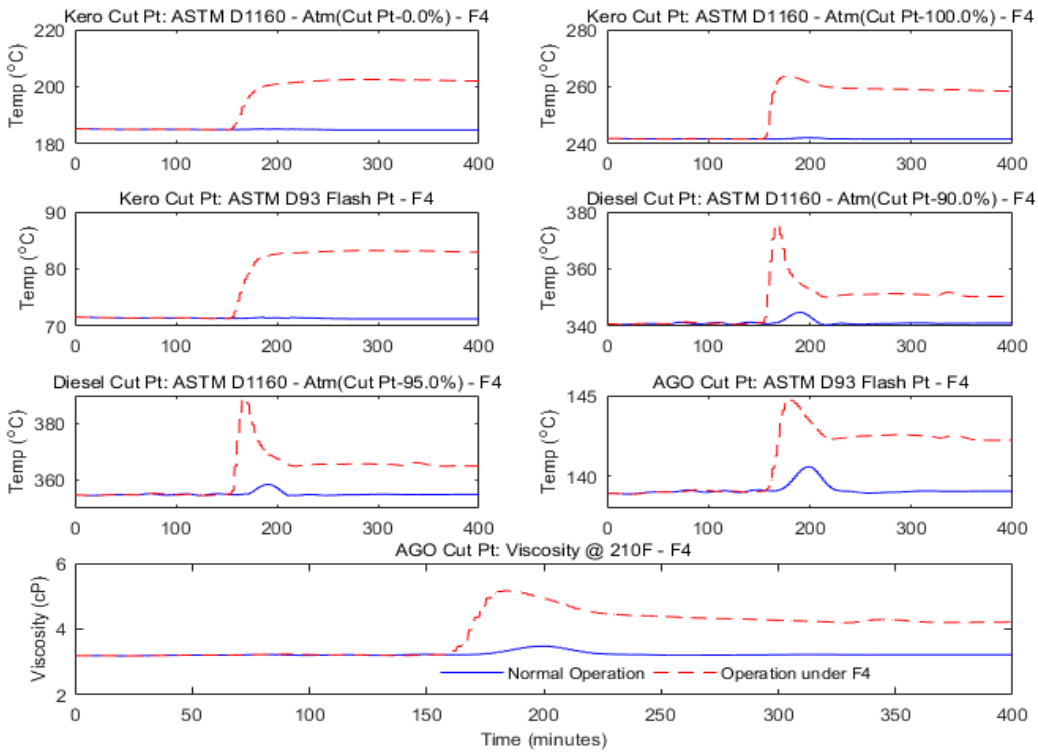


Figure 5.20: Responses of product quality variables to sensor fault F4

sensor faults. The T^2 and SPE monitoring statistics for the training and testing data sets and those of the four sensor fault cases $F1 - F4$ are presented in Figures 5.21 and 5.22 respectively. A sensor fault is declared after the limits of the monitoring statistics are violated simultaneously for four consecutive sampling times (2 minutes), or when the values of the monitoring statistics are more than double those of their respective limits for two consecutive sampling period. The criteria are to help eliminate possible declaration of false alarm.

Immediately a sensor fault is detected, the contribution plots of the monitoring statistics are used to identify the fault. Each principal component (PC) used to develop the diagnostic model is checked at the point of fault declaration to identify the PC that violates its limit. The PC plots for the four sensor faults as presented in Figures 5.23 to 5.26 show plots depicting the principal components being plotted within their respective ±99% control limits for ease of detecting fault and computing contributions of each variable to the faulty situation. Figures 5.27 – 5.30 present the contribution plots of the four sensor faults. The contribution plots show the cumulative effect of the variables responsible for the PCs that violate their limits. For each PC that violates its limit at the point of fault declaration, the variables that contribute majorly to that PC value are identified through

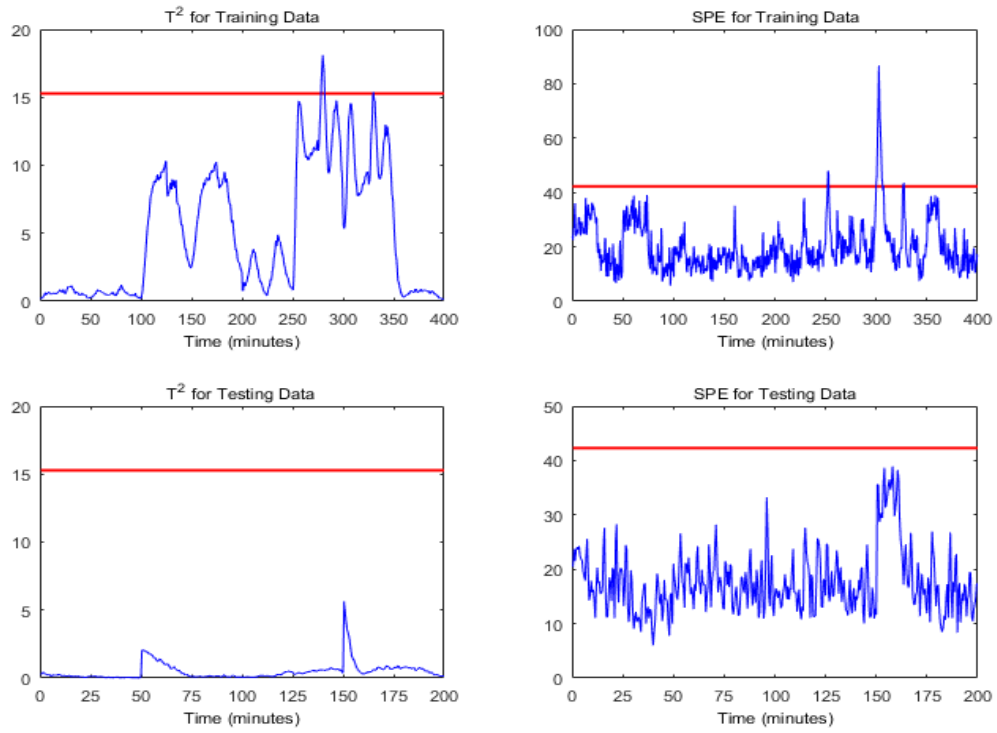


Figure 5.21: T^2 and SPE monitoring plots for normal operation

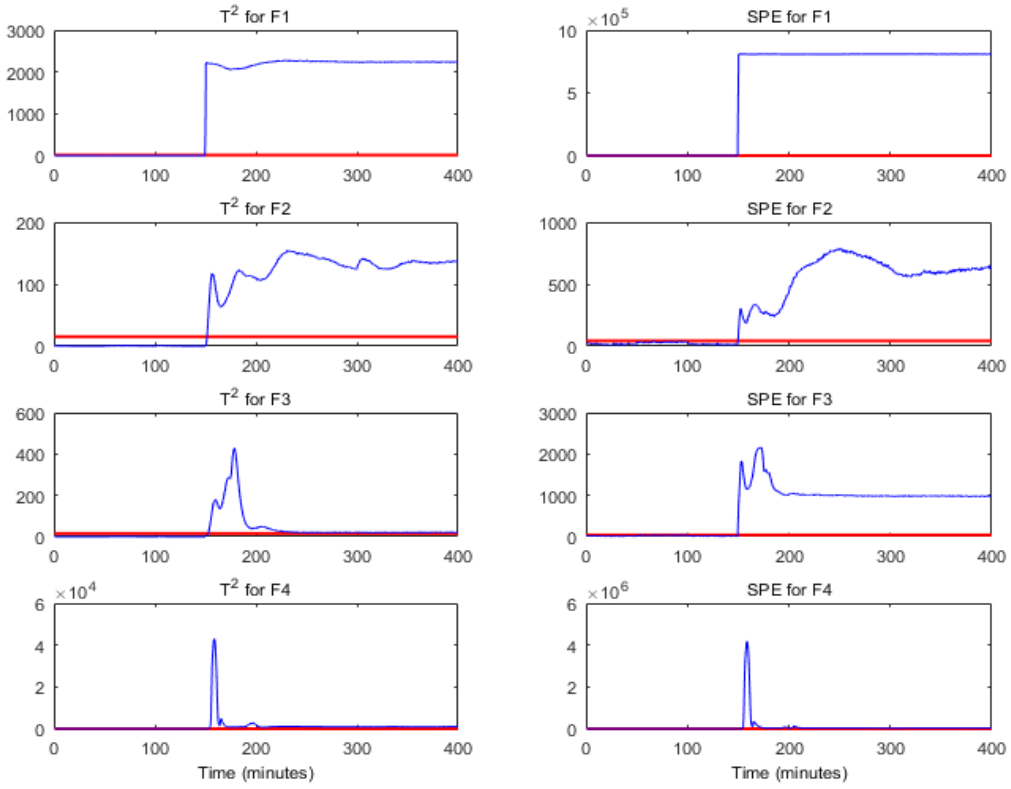


Figure 5.22: T^2 and SPE monitoring plots for sensor faults F1 – F4

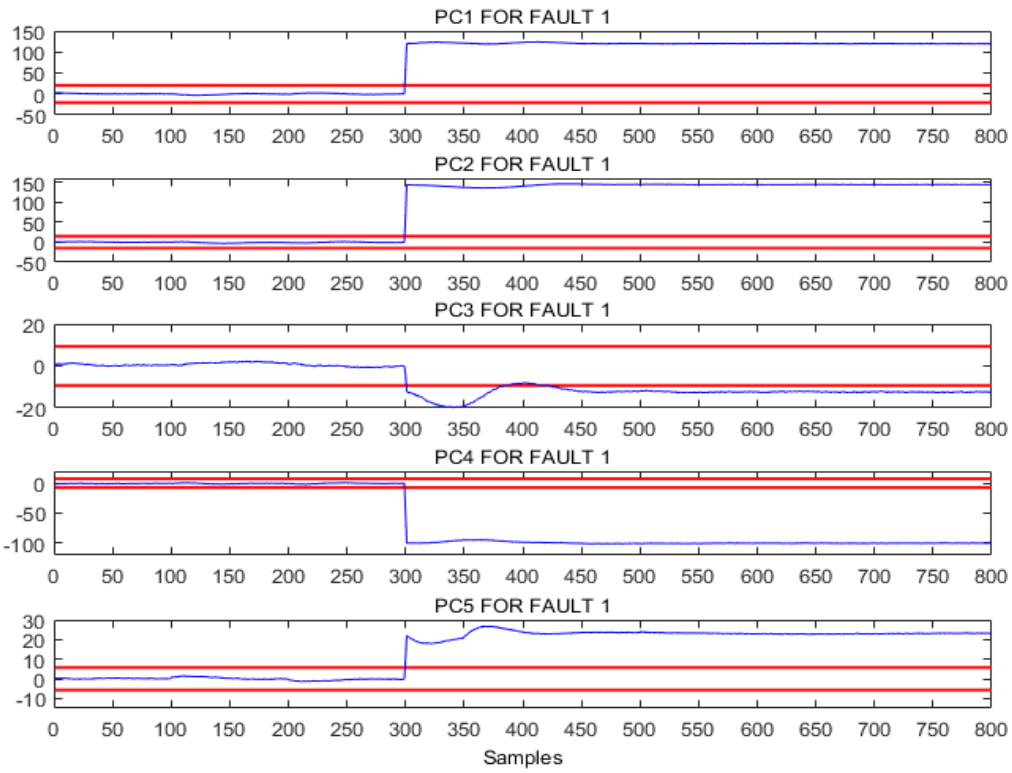


Figure 5.23: PC plots for sensor fault F1

contribution plots. Take for instance, sensor fault $F1$ has all its PCs violating their limits as presented in Figure 5.23. The variables responsible for those violations are picked up in the contribution plot of the same fault $F1$ shown in Figure 5.27. More detailed discussion of how the PC plots and the contribution plots are used to identify the sensor faults are given in Section 5.3.4.

5.3.3 Accommodation of Identified Sensor Faults through Implementation of Sensor FTIC

Upon detection and identification of a sensor fault, the proposed sensor FTIC is implemented on the dynamic crude distillation unit. The relevant redundant soft-sensor estimate of the controlled variable whose sensor developed fault is used in the feedback control loop instead of the faulty sensor output. The controlled variable feedback signals (y'_p) used during the system normal operation is obtained using equation 3.18 as:

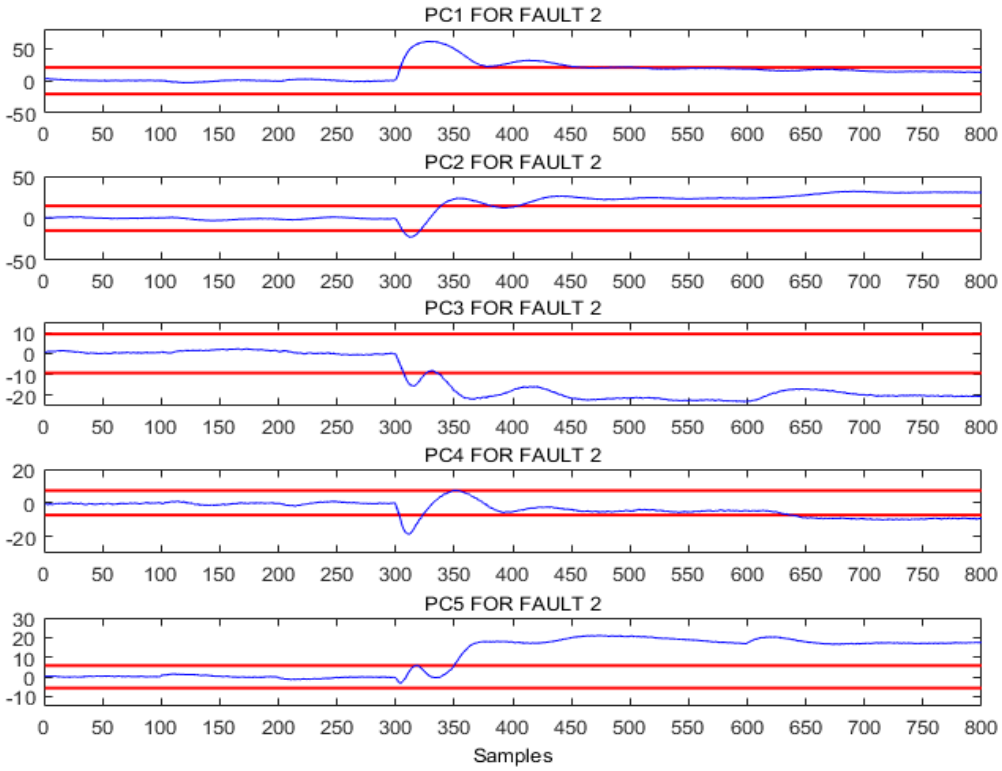


Figure 5.24: PC plots for sensor fault F2

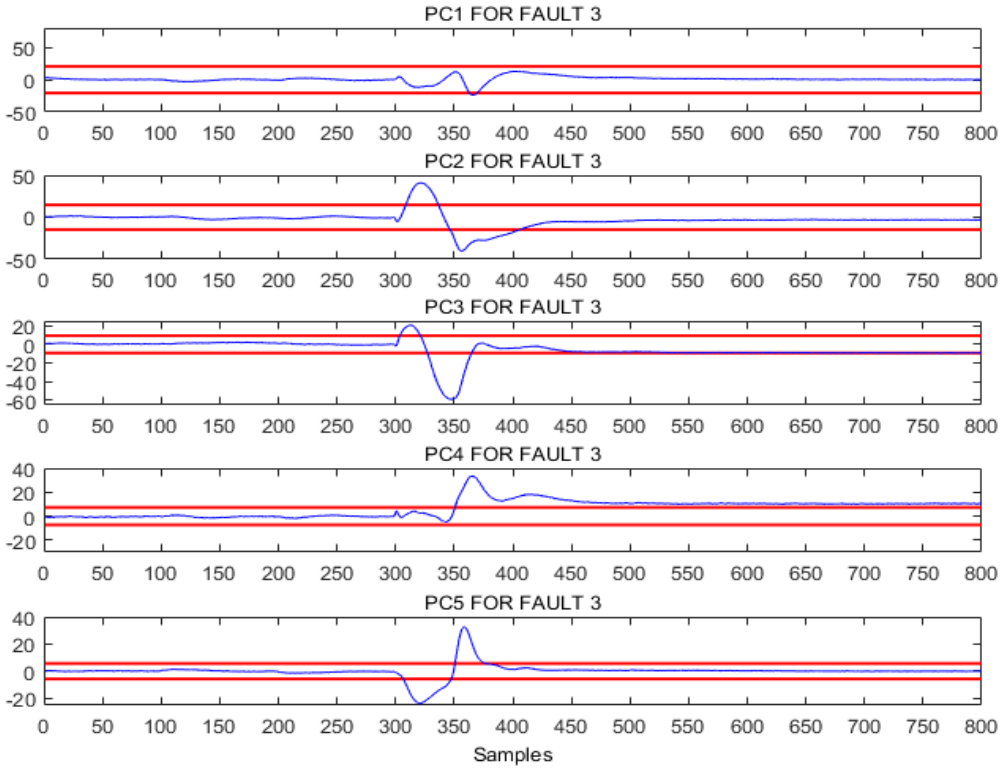


Figure 5.25: PC plots for sensor fault F3

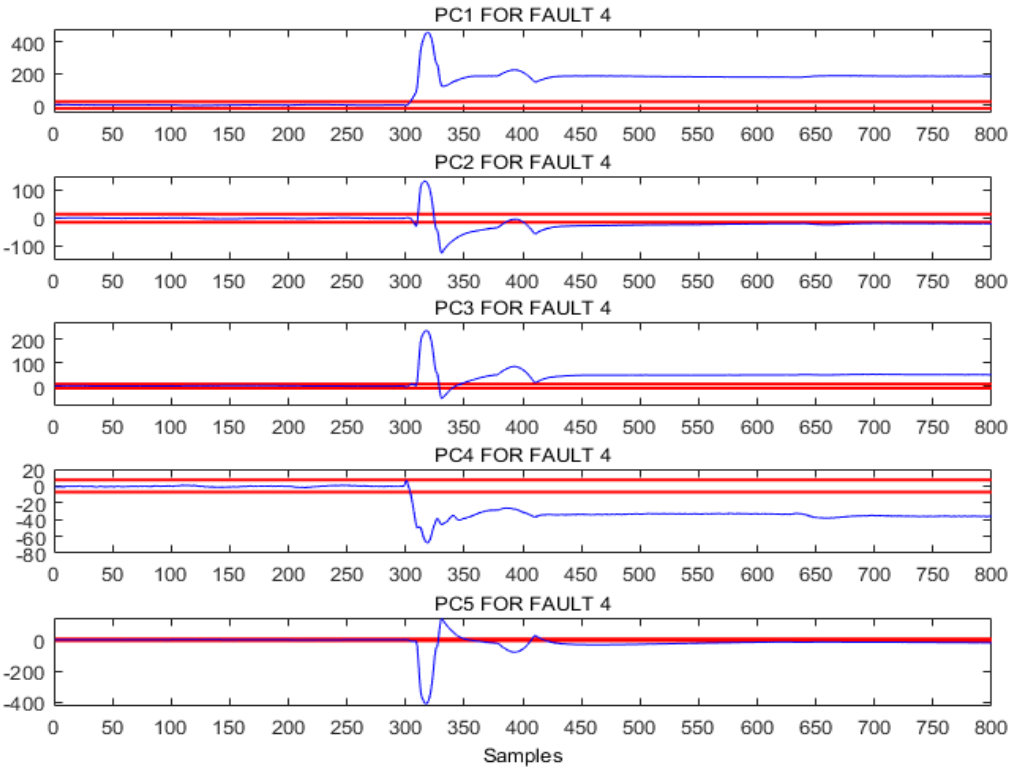


Figure 5.26: PC plots for sensor fault F4

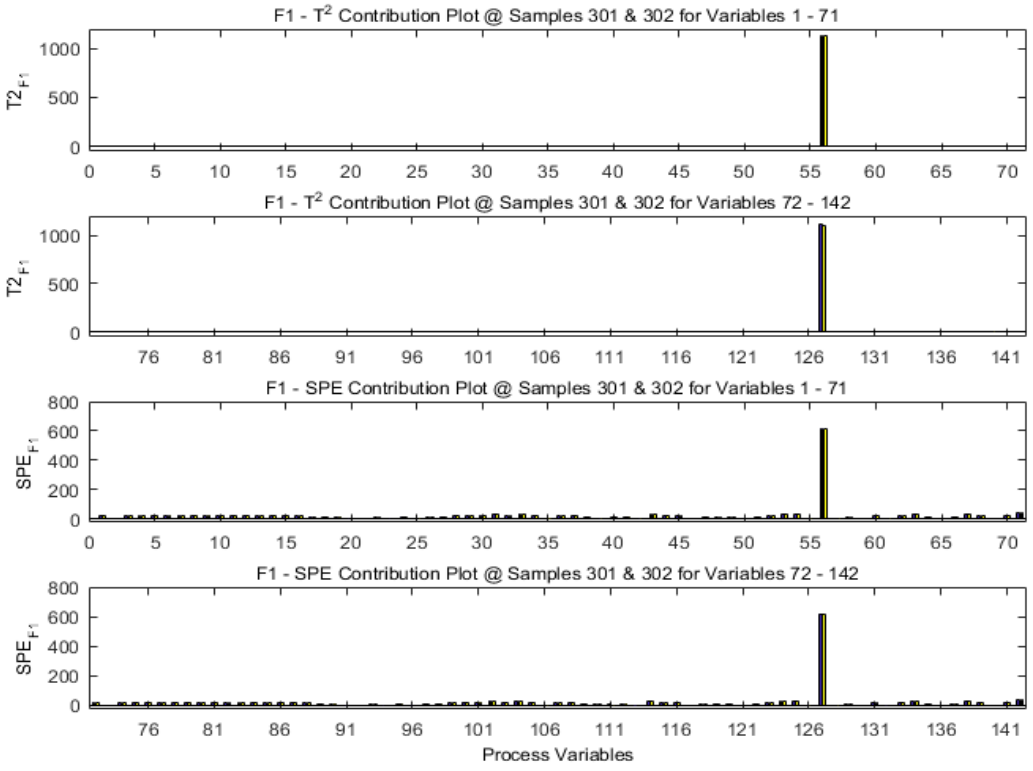


Figure 5.27: Excess contributions plots for sensor fault F1

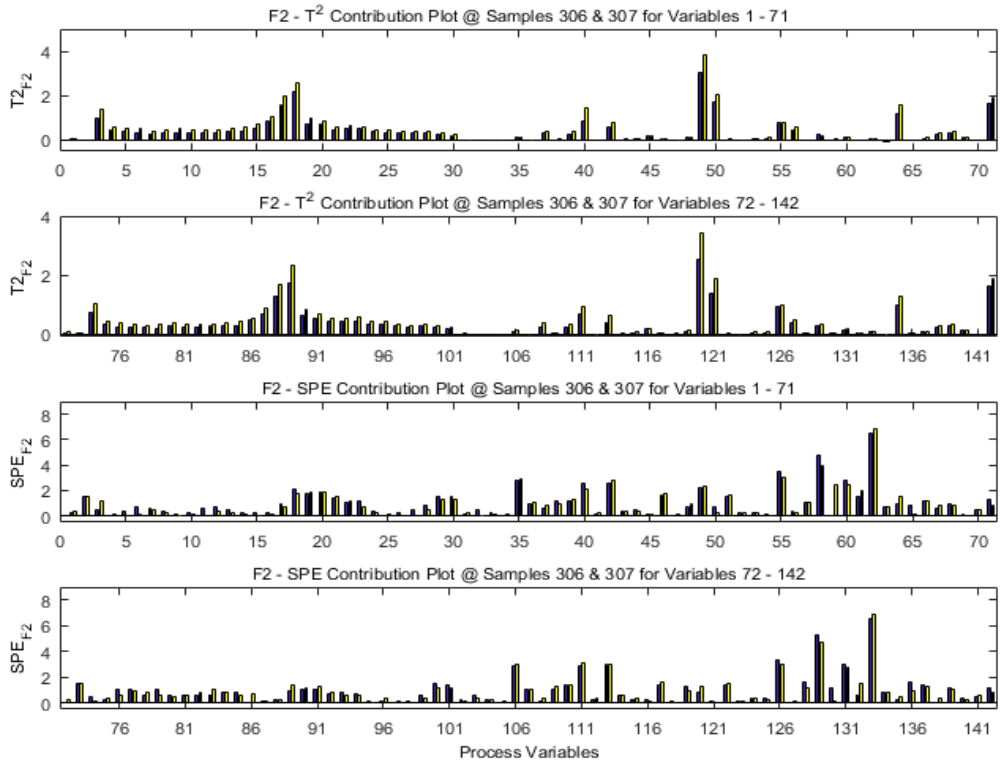


Figure 5.28: Excess contributions plots for sensor fault F2

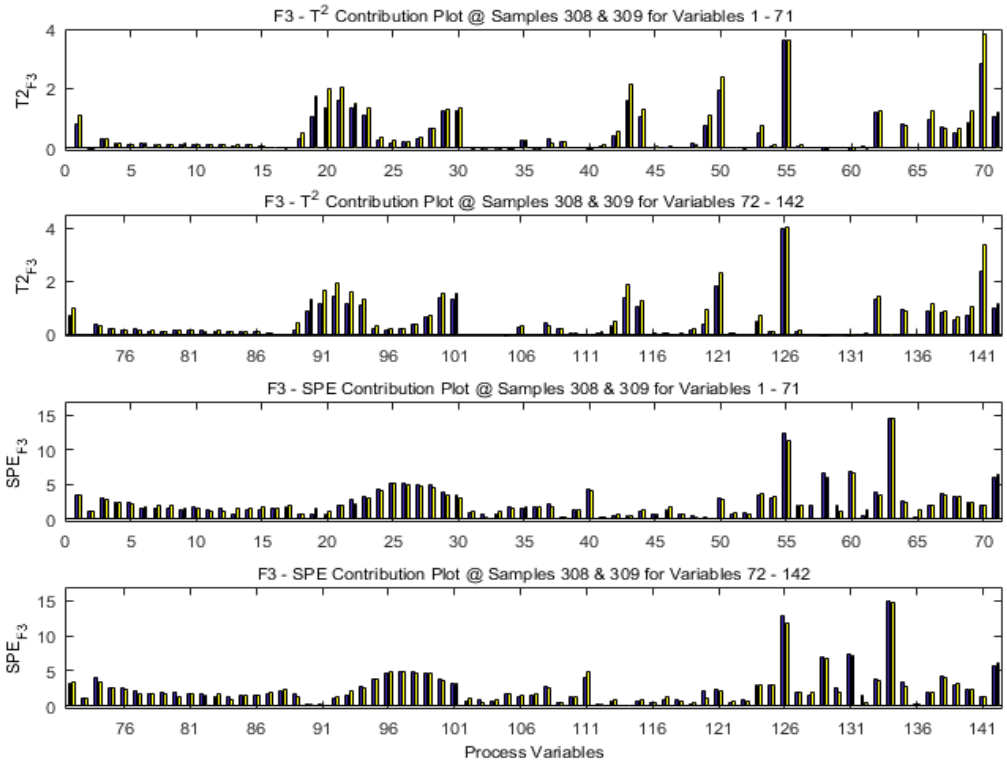


Figure 5.29: Excess contributions plots for sensor fault F3

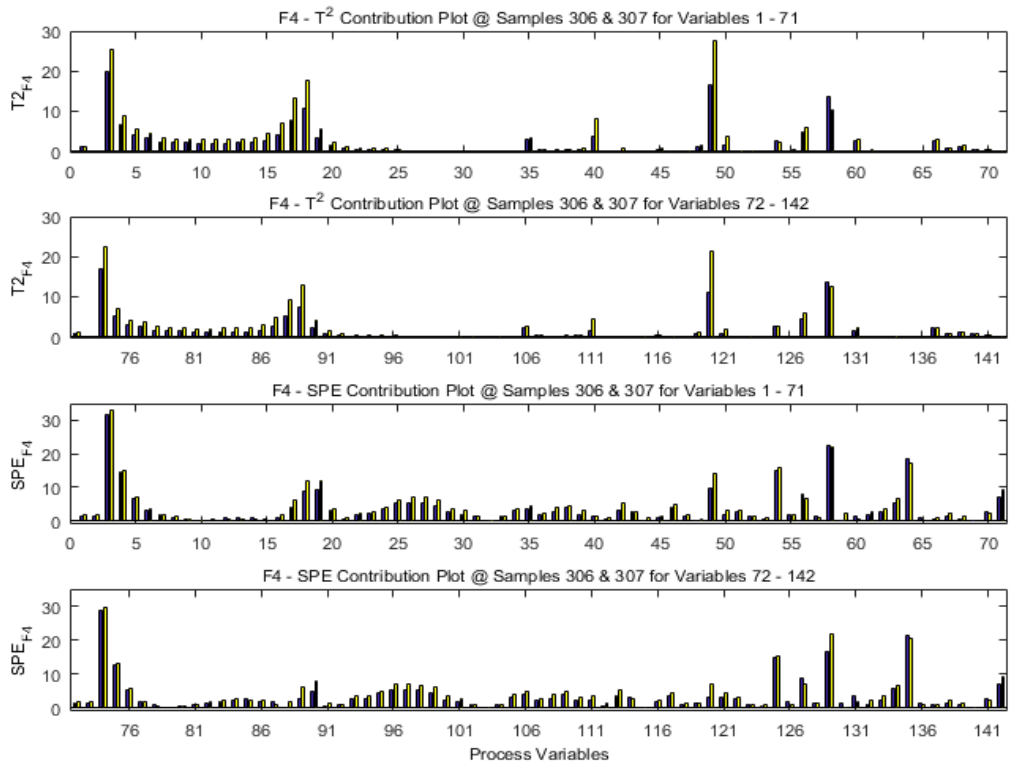


Figure 5.30: Excess contributions plots for sensor fault F4

$$y'_p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{y}_{DPLS_1} \\ \hat{y}_{DPLS_2} \\ \hat{y}_{DPLS_3} \\ \hat{y}_{DPLS_4} \end{bmatrix} \tag{5.5}$$

y_1 to y_4 and $\hat{y}_{DPLS_1}$ to $\hat{y}_{DPLS_4}$ are the controlled variables sensor outputs and their respective redundant soft sensor estimates. When sensor fault $F1$ – 1st pumparound flow sensor fault (y_1) is identified, its diagonal element changes to zero, isolating the faulty sensor while the corresponding diagonal element in the redundant soft sensor backup feedback signal is activated accordingly. Then, the controlled variables feedback signals used to accommodate the sensor fault is given as:

$$y'_p = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{y}_{DPLS_1} \\ \hat{y}_{DPLS_2} \\ \hat{y}_{DPLS_3} \\ \hat{y}_{DPLS_4} \end{bmatrix} \tag{5.6}$$

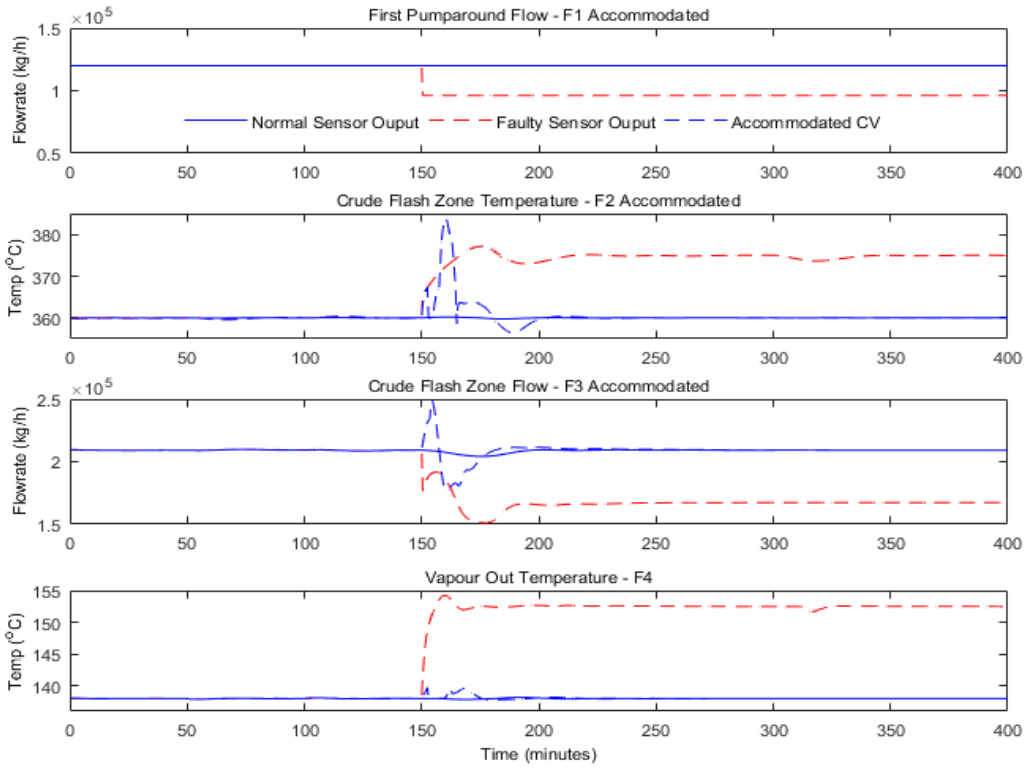


Figure 5.31: Accommodated sensor faults F1 – F4 controlled variables

This procedure is applied to other sensor faults ($F2, F3$ and $F4$) when they are identified. The responses of the controlled variables to the implementation of the sensor FTIC to accommodate faults $F1 - F4$ are presented in Figure 5.31. The solid blue lines are the measured values of the controlled variables during normal operation; the dashed red lines are the faulty sensor outputs of the controlled variables, while the dashed blue lines are the responses of the controlled variables to the implementation of sensor FTIC. Figures 5.32 – 5.35 present the responses of the product quality variables, green dashed lines, to the same accommodating sensor fault strategy. Red and blue lines are as previously defined.

5.3.4 Results and Discussions

The sensor faults investigated in the dynamic CDU were all detected, using the T^2 and SPE monitoring plots presented in Figure 5.22. The first pumparound flow sensor fault – $F1$ was detected a minute after introduction at sample 302 on both T^2 and SPE monitoring plots. Sensor fault $F2$ – CFZ temperature sensor fault was detected 3 minutes and 1 minute after introduction at samples 306 and 302 on T^2 and SPE monitoring plots

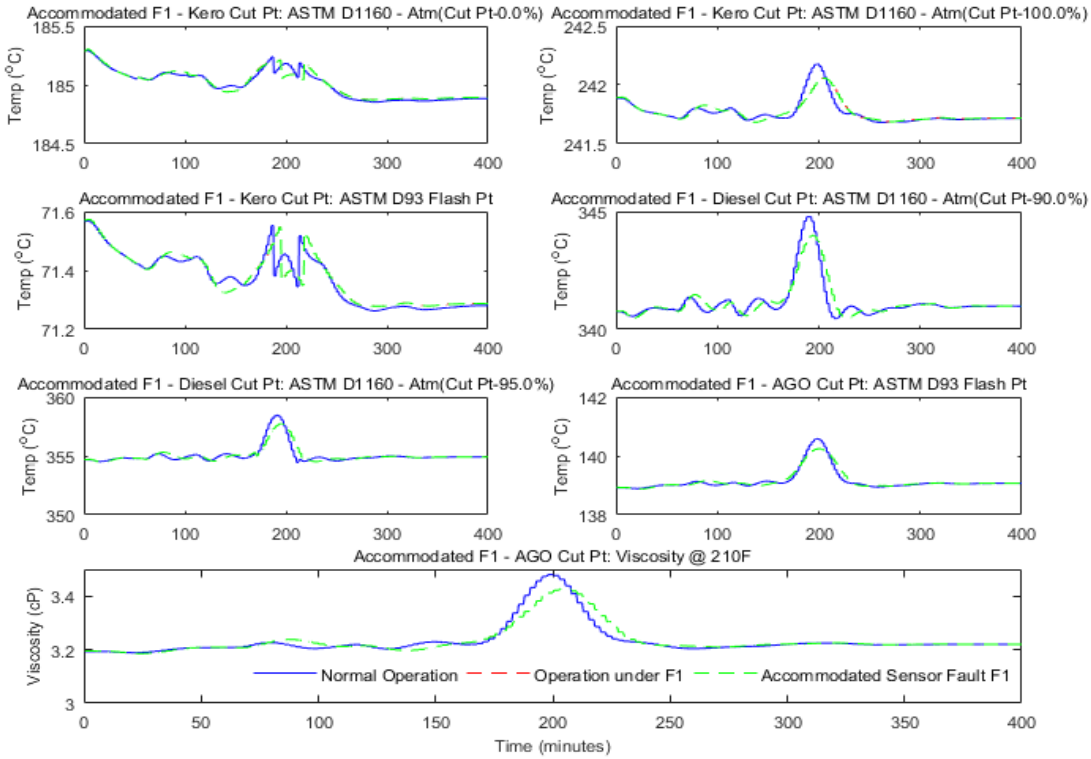


Figure 5.32: Responses of PQV to implementation of sensor FTIC on sensor fault F1

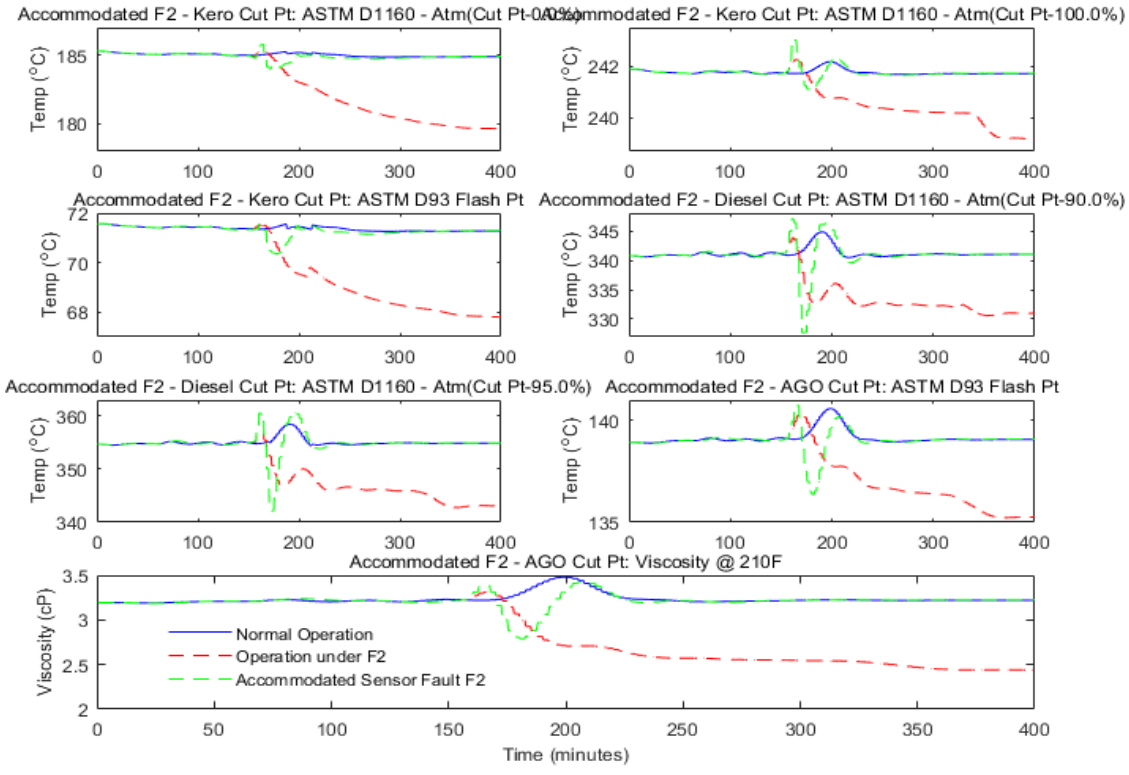


Figure 5.33: Responses of PQV to implementation of sensor FTIC on sensor fault F2

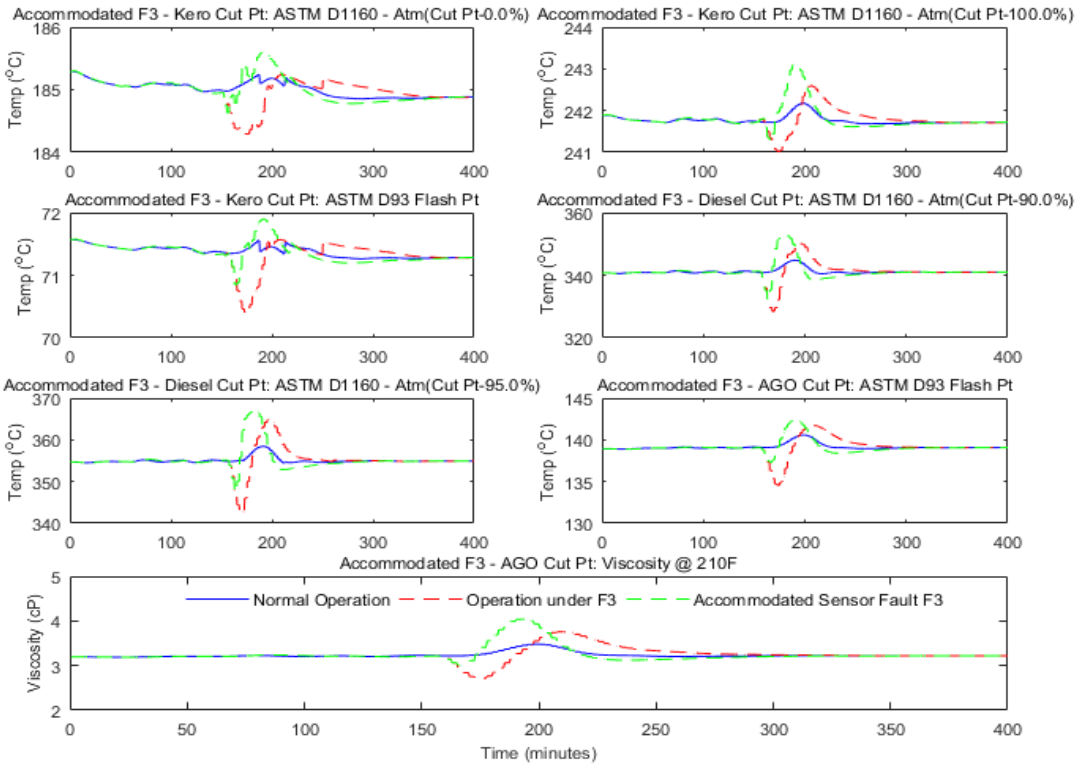


Figure 5.34: Responses of PQV to implementation of sensor FTIC on sensor fault F3

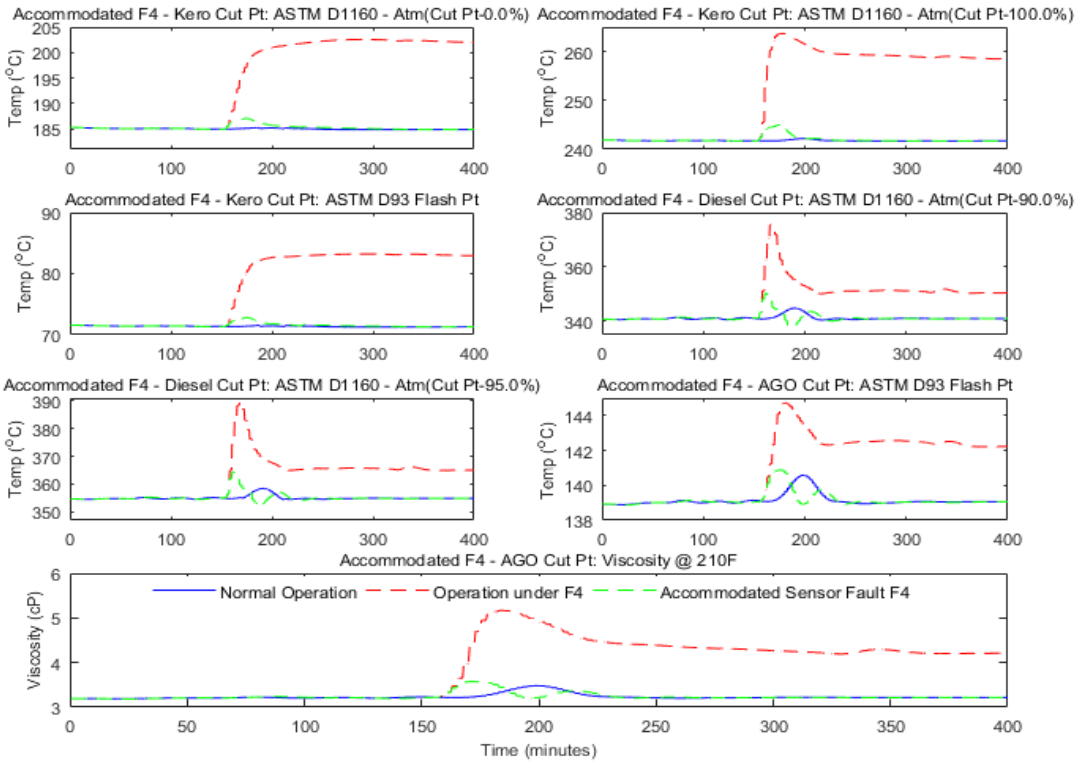


Figure 5.35: Responses of PQV to implementation of sensor FTIC on sensor fault F4

respectively. $F3$ – CFZ flow sensor fault was detected at samples 308 and 302, 4 minutes and 1 minute on T^2 and SPE monitoring plots respectively. Sensor fault $F4$ – stage 1 temperature sensor fault was detected on T^2 and SPE monitoring plots at samples 306 and 302 after introduction, 3 minutes and 1 minute respectively.

Immediately the sensor faults are detected, further diagnostics are carried out to identify the faults through the principal component (PC) plots and the monitoring statistics contribution plots. The PC plots shown in Figures 5.23 to 5.26 identified the PCs that violate their limits. Information from the PC plots is used to compute the contributions of each variable to the faulty situation and presented in the T^2 and SPE contribution plots shown in Figures 5.27 to 5.30. Both T^2 and SPE contribution plots picked up variable 56, the first pumparound flow sensor output as the only contributing variable to the fault $F1$ as presented in Figure 5.27. Its value drifted well out of the operating region. Variables 17, 18, 49, 50, 64 and 71, which are stages 16 and 17 temperatures, side streams (SS) 2 and 3 return temperatures, stage 1 temperature and furnace heat flow respectively, have high contributions to the large T^2 value for fault $F2$. The SPE contribution plot shows variables 18, 19, 35, 55, 58 and 62, which are respectively temperatures of stages 17 and 18, Naphtha mass flow, bottom boil-up mass flow, second pumparound mass flow and CFZ temperature, as being responsible for the high SPE values, as presented in Figure 5.28. The variables identified by the T^2 and SPE contribution plots have a direct connection with the sensor output. For instance, when the CFZ temperature sensor displays a value below set-point, the manipulated variable that is used to keep the controlled variable at set-point, in this case, the furnace heat input into the column increases in order to maintain the CFZ temperature at a desired value. Also, increased CFZ temperature will affect virtually all the column temperatures, particularly the ones near the bottom, as observed here.

Sensor fault $F3$ has variables 19, 20, 21, 29, 30, 44, 50, 55, 62 and 70, which are temperatures of stages 18, 19, 20, 28 and 29, CDU residue temperature, SS-3 return temperature, bottom boil-up mass flow, CFZ temperature and the crude feed to CDU residue ratio respectively, as the major contributors to the faulty situation according to T^2 contribution plot. SPE contribution plot has variables 25, 26, 27, 55, 58, 60, 63 and 71, which are respectively temperatures of stages 24, 25 and 26, bottom boil-up mass flow, second pumparound mass flow, third pumparound mass flow, CFZ mass flow and the furnace heat flow as being responsible for the fault, as presented in Figure 5.29. Table 5.5 presents the variables responsible for each sensor fault. Fault in CFZ mass flow sensor has

a direct impact on most flows in the CDU, especially the ones closely linked to the amount of feed charged into the column like bottom boil-up mass flow, the pumparound flows and the sensor displayed value for the CFZ mass flow as identified by the contribution plots. Also, an increase or otherwise in the CFZ mass flow as a consequence of its faulty sensor will affect the furnace heat input and the temperatures of the bottom stages in the column. These are the variables identified by the contribution plots as being symptomatic of the sensor fault.

Sensor fault $F4$ according to T^2 contribution plot has variables 3, 4, 17, 18, 49 and 58, which are temperatures of stages 2, 3, 16 and 17, SS-2 return temperature and second pumparound mass flow respectively as being symptomatic of the fault. The SPE contribution plot shows variables 3, 4, 18, 49, 54, 58, 64 and 71, which are respectively temperatures of stages 2, 3, and 17, SS-2 return temperature, reflux mass flow, second pumparound mass flow, temperature of the vapour leaving the column (stage 1 temperature) and the furnace heat flow as the major contributing variables to the abnormal situation. Figure 5.30 presents the contribution plots for sensor fault $F4$. Sensor fault $F4$ obviously has a direct effect on the reflux mass flow which is being used to control it. It also affects the temperatures of the top stages and the flows closely associated with the reflux flow like first and second pumparound flows, as identified by the contribution plots. The variables indicative of each fault, together with good knowledge of the process are used to identify the faults.

Table 5.5: Variables responsible for faults F1 – F4

Faults		Variables
F1	T^2	56
	SPE	56
F2	T^2	17, 18, 49, 50, 64, 71
	SPE	18, 19, 35, 55, 58, 62
F3	T^2	19, 20, 21, 29, 30, 44, 50, 55, 62, 70
	SPE	25, 26, 27, 55, 58, 60, 63, 71
F4	T^2	3, 4, 17, 18, 49, 58
	SPE	3, 4, 18, 49, 54, 58, 64, 71

Sensor FTIC is implemented after the identification of a sensor fault using the backup soft sensor estimate in the feedback control loop, in place of the faulty sensor output as described in the last section. It can be observed from Figure 5.31 and Figures 5.32 – 5.35

that the proposed sensor FTIC worked quite well in accommodating the sensor faults, maintaining the product quality variables within their normal operating regions. The nature of a sensor fault and its effect on the system determine whether or not it should be accommodated. Take sensor fault $F1$ for instance, due to the way the system is designed a 20% loss of efficiency means the value recorded is 20% lower than the actual measured value. The controller responded by sending a corrective signal to the valve to open wider, however, due to the operating condition and physical restriction of the valve, opening it by 100% made little difference to the actual flow going through it. The amount of fluid flowing through the valve did not change much despite the fault and did not affect the system, as presented in Figure 5.32. The same scenario also played out in sensor fault $F3$; the FTIC was able to improve the system performance in this case, but not significantly as the effect of the fault did not drive the system to instability as presented in Figure 5.34.

It is important to note that the faults also affect the secondary variables used to infer the back-up soft sensor estimates of the controlled variables at the point of fault declaration t_f. Hence, the secondary variables shifted backwards by n sampling times (i.e. measurements at $t_f - n$ sampling times) are used to first stabilise the system up until sampling time $t_f - 2$. Value of n is chosen based on the knowledge of the system, particularly the system settling time.

5.4 Sensor and Actuator Faults Accommodation in Crude Distillation Unit

The effectiveness of the proposed fault tolerant control system (FTCS) in accommodating both actuator and sensor faults is demonstrated in this section. The combined actuator and sensor fault tolerant controllers are implemented on the crude distillation unit to accommodate successive sensor and actuator faults. Like most of the existing FTCS, it is based on the assumption that no two faults are occurring simultaneously. However, the proposed simplified accommodating strategy can accommodate successive sensor faults and combined sensor and actuator faults occurring consecutively. Only three faults are investigated under this section, a sensor fault, an actuator fault and a combined sensor and actuator as presented in Table 5.6 and Figure 5.36. This will avoid repetition as the techniques employed in accommodating individual fault are also applied when dealing

with two or more faults occurring successively. The interactive dynamic CDU described in Sections 4.4.1 and 5.3.1 is used for this purpose.

Table 5.6: CDU sensor and actuator fault list

Fault	Fault description
F1	CFZ temperature sensor fault with 20% bias introduced @ sample 201
F2	Reflux flow control valve maximum opening restricted to 28%, introduced @ sample 351
F3	Combined F1 and F2 faults introduced @ samples 201 and 351 respectively

5.4.1 Sensor and Actuator Faults Introduction, Detection and Identification

As listed in Table 5.6, fault $F1$ is CFZ temperature sensor fault introduced into the system at sample 201, the magnitude and manner of introduction are similar to the CFZ temperature sensor fault $F2$ investigated in Section 5.3.2, Table 5.4. Faults $F2$ is reflux flow control valve fault as described in Section 4.4.4 under introduction of actuator fault, while fault $F3$ is the combined sensor fault $F1$ and actuator fault $F2$ introduced at samples 201 and 351 respectively. After the faults were introduced, the DPCA fault detection and diagnostics scheme developed with five PC in Section 4.4.5 is employed to detect and identify the faults. Figure 5.37 presents the T^2 and SPE monitoring statistics for the training and validating data sets and those of fault $F3$. Faults $F1$ and $F2$ produced similar plots to those presented in Sections 5.3.2 and 4.4.5, Figures 5.23 and 4.39 respectively. Hence they are not presented here. It is very important to note that the sensor fault $F1$ that was first introduced has to be isolated and accommodated using the sensor FTIC described in Section 5.3 before the actuator fault that was later introduced could be detected and identified. Otherwise, the actuator fault will be masked by the effect of the sensor fault on the system, as can be observed from Figure 5.37.

Figure 5.38 shows the effect of the fault on their respective controlled variables, while Figure 5.39 presents the responses of the system product quality variables to fault $F3$. Upon detection of a fault, whether a sensor or an actuator fault, the contribution plots of the T^2 and SPE monitoring statistics are used to identify the fault, by associating certain variables that have unusually large contributions to the abnormal situation to a particular fault.

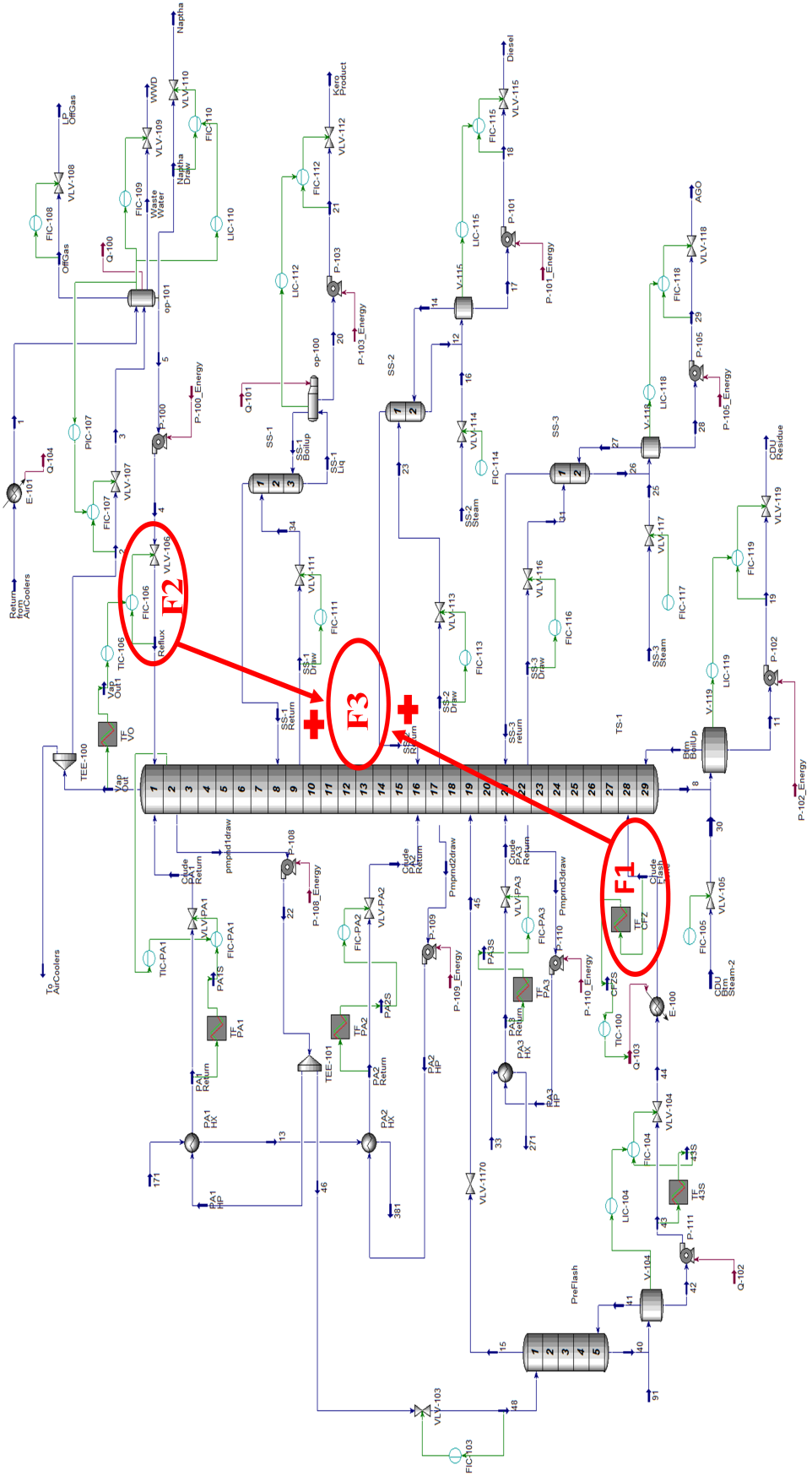


Figure 5.36: Dynamic CDU with faults F1 – F3

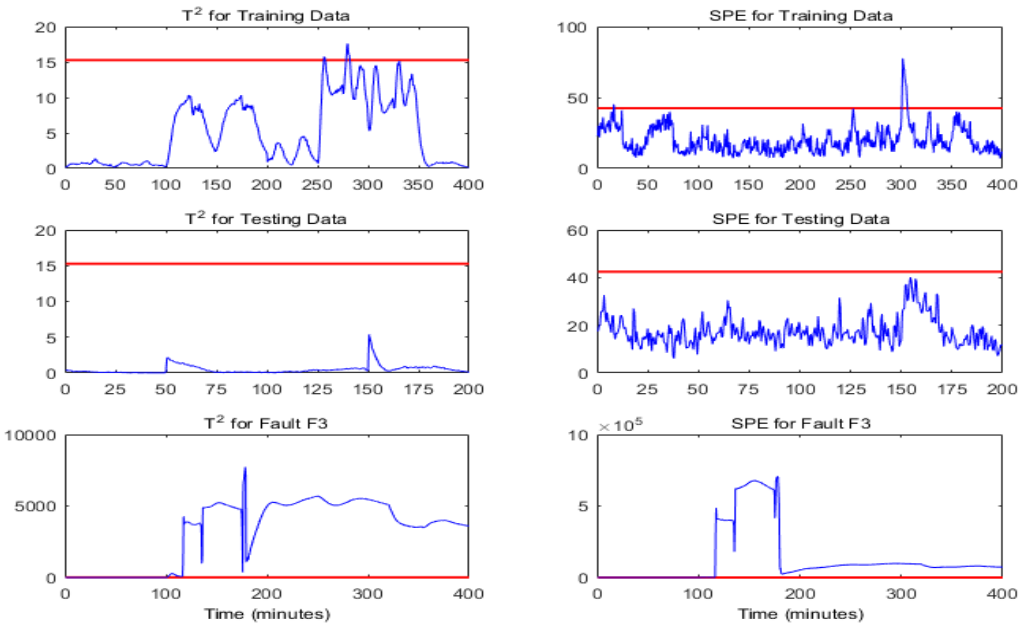


Figure 5.37: T^2 and SPE for training and testing data sets, and fault F3

For the combined sensor and actuator fault $F3$, the individual faults were separately identified as discussed in Section 5.3.4 (the sensor part of $F3$) and 4.4.7 (the actuator part of $F3$); the variables contributing to each fault are the same as previously explained.

5.4.2 Sensor and Actuator Faults Accommodation Using FTIC and FTC

Accommodation of sensor fault $F1$ and actuator fault $F2$ are not discussed separately to avoid repetition as they are discussed under fault $F3$. To accommodate the sensor part of fault $F3$, the DPLS soft sensor estimate of the CFZ temperature is used in the feedback loop to replace the faulty sensor output, to maintain the system stability. This was discussed in Section 5.3.4. Having accommodated the sensor fault, the actuator part of fault $F3$ can be accommodated by first isolating the faulty valve, and then switch the manipulated variable of the faulty loop to another one that can effectively be used to keep the concern controlled variable within an acceptable operating region. The pre-assessed input-output reconfiguration pairing presented in Table 4.15 is used to reconfigure the furnace heat flow into the system to directly control the temperature of the vapour leaving stage 1 (stage 1 temperature), leaving the CFZ temperature uncontrolled. This automatically leaves the sensor part of the same fault ineffective, because the CFZ temperature is no longer being controlled. However, the CFZ temperature soft sensor estimate is still

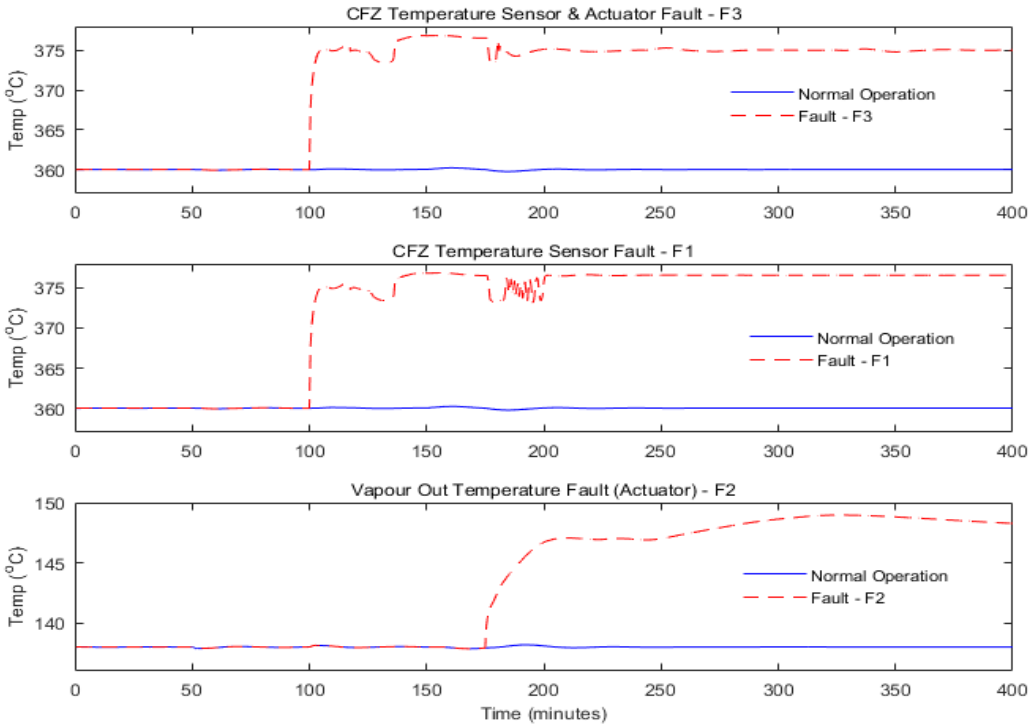


Figure 5.38: Controlled variable responses to faults F1 – F3

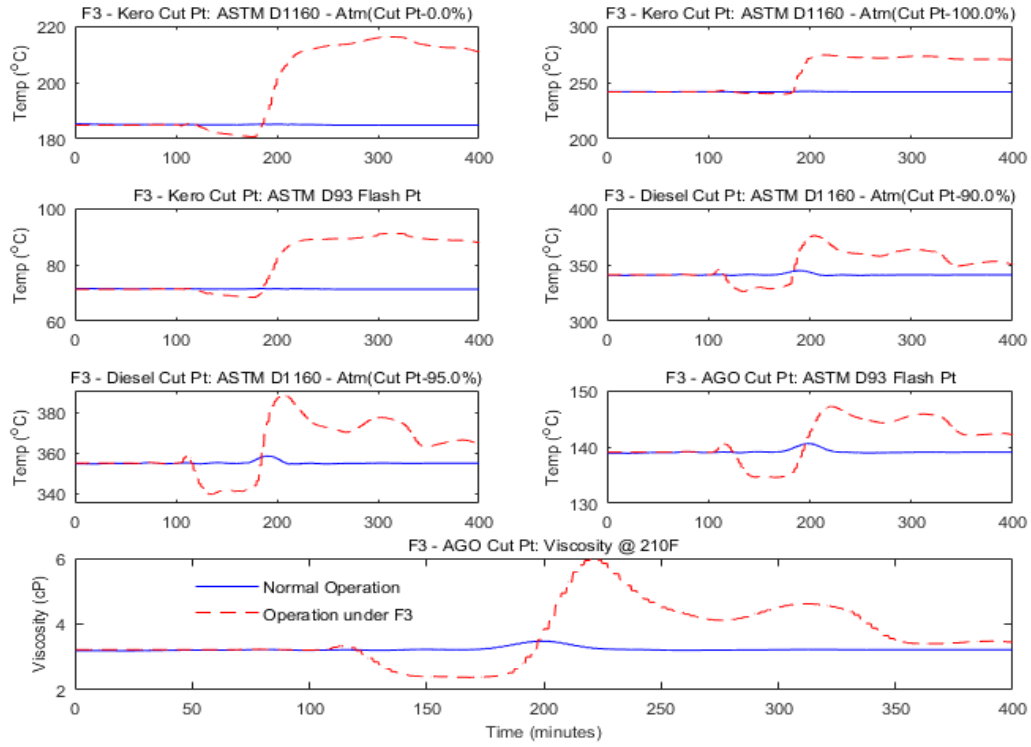


Figure 5.39: Responses of the PQV to fault F3

being used as an indicator. The sensor part of fault F3 is accommodated as stated below.

$$y'_p = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_{p1} \\ y_{p2} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{y}_{DPLS_1} \\ \hat{y}_{DPLS_2} \end{bmatrix} = \begin{bmatrix} \hat{y}_{DPLS_1} \\ y_{p2} \end{bmatrix} \tag{5.7}$$

where y'_p is a vector of the controlled variable feedback signals, y_{p1} and y_{p2} are CFZ temperature and the stage 1 temperature respectively, while $\hat{y}_{DPLS_1}$ and $\hat{y}_{DPLS_1}$ are their corresponding DPLS based soft sensor estimates. The feedback controlled variables y_y in equation 3.20 is for the combined feedback signal y'_p and the backup feedback signal y_{bi} as presented in Figure 3.4. It is given as:

$$y_y = \begin{bmatrix} \hat{y}_{DPLS_1} \\ y_{p2} \\ y_{b1} \\ y_{b2} \end{bmatrix} \tag{5.8}$$

The error signal generated to accommodate fault $F3$ according to equation 3.21 is obtained as:

$$e_{F3} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{p1} - \hat{y}_{DPLS_1} \\ r_{p2} - y_{p2} \\ r_{b1} - y_{b1} \\ r_{b2} - y_{b2} \end{bmatrix} \tag{5.9}$$

The fault tolerant control law used to accommodate the actuator part of fault $F3$ is then given as:

$$u_{F3} = \begin{bmatrix} u_1 \\ u_2 \\ u_{b1} \\ u_{b2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{b2} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ r_{b2} - y_{b2} \end{bmatrix} \tag{5.10}$$

5.4.3 Results and Discussions

The three fault cases investigated in this section were all identified using the DPCA FDD scheme described in Sections 4.4.5 and 5.3.2. Sensor fault $F1$ is the same as the sensor part of fault $F3$ and was detected 3 minutes and 1 minute after introduction, at samples 206 and 202 on T^2 and SPE monitoring plots respectively. Actuator fault $F2$ is the same

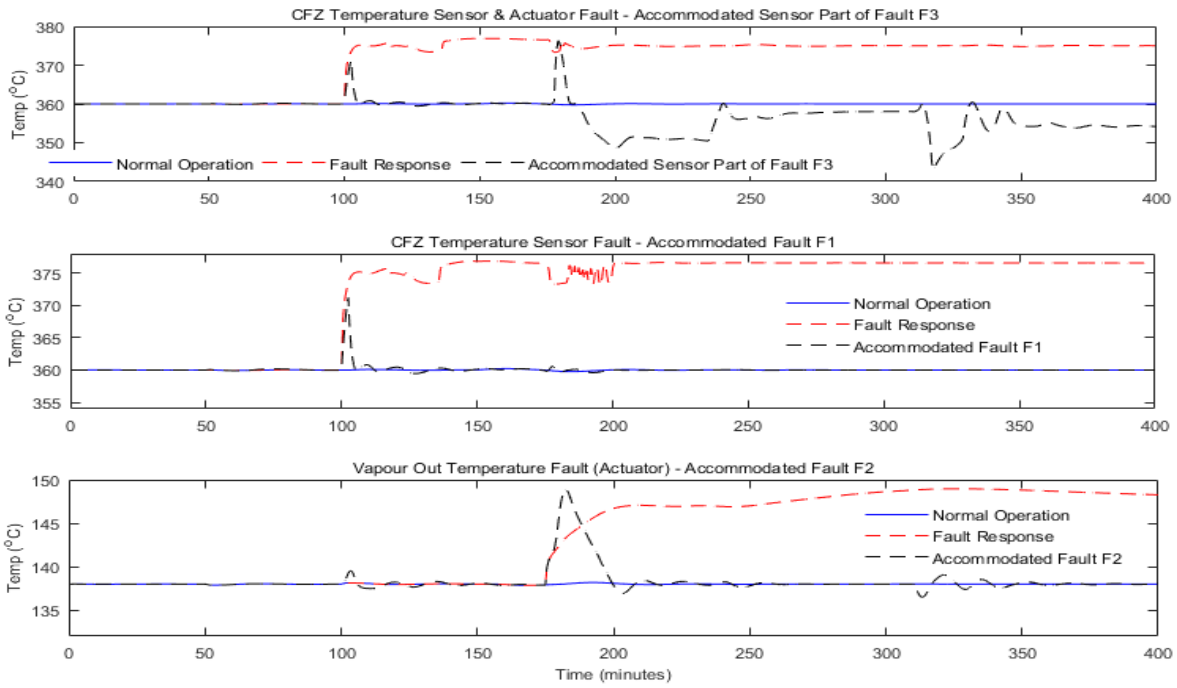


Figure 5.40: Accommodated controlled variables for faults F1 – F3

as the actuator part of fault $F3$ and was detected at samples 355 and 352, 2 minutes 30 seconds and 1 minute after introduction on T^2 and SPE monitoring plots respectively. The faults are identified accordingly using the appropriate contribution plots as discussed for sensor fault $F2$ in Section 5.3.4 and for actuator fault $F2$ in Section 4.4.7 respectively. The same variables are responsible for the faults as previously discussed.

The combined actuator and sensor faults $F3$ is accommodated as presented in Figures 5.40 and 5.41. After the sensor part of fault $F3$ was declared, its redundant soft sensor estimate is used in the feedback loop in place of the faulty sensor measurement to accommodate the fault. It can be observed in the first two rows of Figure 5.40 that the sensor fault was well accommodated. The actuator part of fault $F3$ is accommodated by reconfiguring the FTCS, switching its manipulated variable to the furnace heat flow into the system and the controller settings changed as appropriate. This enables the stage 1 temperature to be controlled directly using the furnace heat flow and leaving the CFZ temperature uncontrolled; this effectively discontinued the use of CFZ temperature soft sensor estimate to accommodate the sensor fault. Figure 5.41 shows the responses of the product quality variables to the implementation of the simplified FTCS on the CDU system.

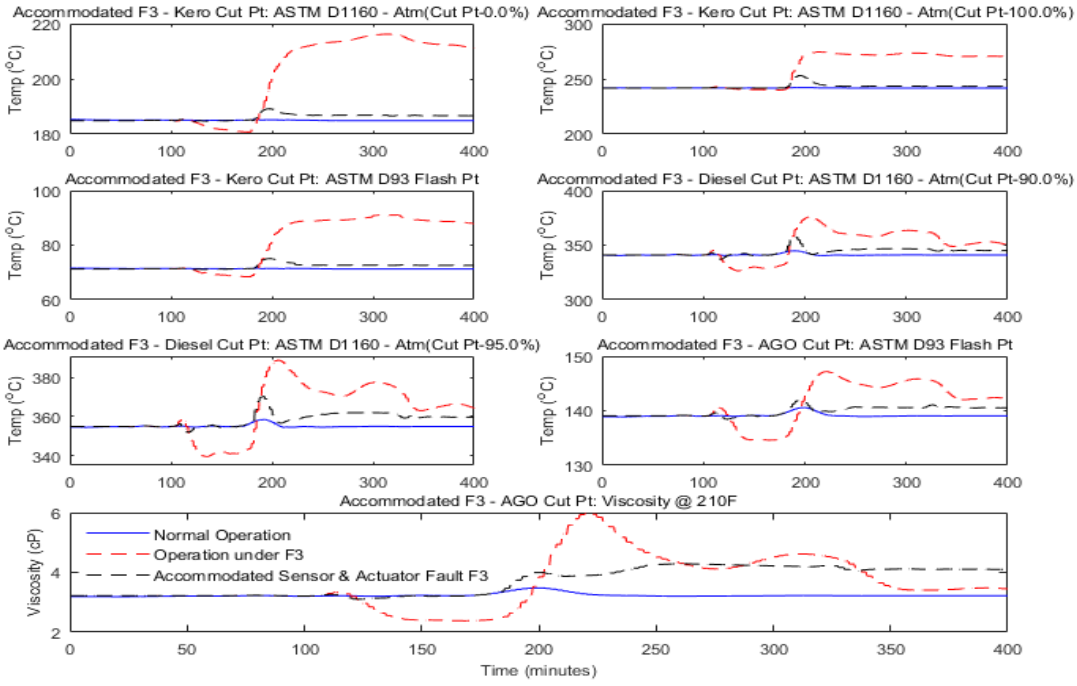


Figure 5.41: Accommodated PQV for fault F3

5.5 Summary

The implementation of the sensor fault tolerant inferential controller (FTIC) on industrial distillation processes have been reported in this section. The effectiveness of the approach was first demonstrated on a binary distillation column, using the proposed sensor FTIC to accommodate sensor faults through the use of soft sensor estimates of the appropriate controlled variable instead of the faulty sensor output to maintain the integrity of the control system and that of the process. The approach was also implemented on an interactive dynamic crude distillation unit to further demonstrate its applicability and efficacy on complex industrial distillation processes. There are possible implementation issues that could arise as a result of using the soft sensor estimates of the affected controlled variable sensor outputs for FTIC, which could worsen the abnormal situation if not quickly addressed. This is because, the effect of the sensor fault on the secondary variables to be used to estimate the controlled variable will lead to inaccurate estimates, and when used in the feedback loop creates further problem for the system. This is addressed by using nominal values of the secondary variables to estimate the output for a specified period of time and also holding the appropriate manipulated variable at its nominal value to first stabilise the system before switching to the FTIC mode.

The proposed FTCS is then implemented on the CDU to accommodate successive sensor and actuator fault. The simplified restructurable fault tolerant control system worked effectively in handling the faults introduced, and without any doubt will improve the availability and robustness of control systems employed in the operation of refineries. It will also help to quickly arrest abnormal situations arising from the occurrence of faults in sensors and actuators.

Chapter 6

Faults Tolerant Model Predictive Control

6.1 Introduction

This chapter explores the use of restructurable model predictive control strategy, referred to in this thesis as "Fault-Tolerant Model Predictive Control (FTMPC)" to accommodate actuator faults in a crude distillation unit. Model predictive control (MPC) has gained wide acceptance in the industry because of its constraints handling capabilities, which is one of its main features leveraged on in this thesis to accommodate actuator faults. Its design involves obtaining over the control horizon, the optimal sequence of adjustments to the manipulated variables that will over the prediction horizon minimise the predicted control errors. The first move of the sequence of adjustments to the manipulated variable is implemented, and the whole process is repeated. An actuator fault presents a severe constraint to the effective good performance of a control system in any plant, more so in a complex system like crude distillation unit. FTMPC belongs to the class of constraint optimisation based control algorithm with extended capabilities of possible control structure reconfiguration in the presence of faults. The model used for output prediction in FTMPC is updated, its objective function and constraints are also adjusted in the face of actuator faults for safe and continued economic operation of the system.

This chapter is organised as follows. The next section describes the design of the FTMPC, the development of the first order plus dead-time (FOPDT) model of the system investigated and the possible control structure reconfiguration pre and post-fault era. Integration of the fault detection and diagnosis (FDD) scheme with the proposed FTMPC

for actuator faults diagnosis and accommodation is also discussed. This is followed by the integration of fault-tolerant inferential control (FTIC) with the FDD scheme and the proposed FTMPC to form a complete FTCS to accommodate both actuator and sensor faults. Application of the FTMPC together with FTIC to the crude distillation system discussed in Chapter 4, to accommodate actuator and sensor faults is then presented in Section 6.4, followed by the discussion of results in Section 6.5 and a summary of the chapter in the final section.

6.2 Design of Fault-Tolerant Model Predictive Control

The fault tolerant model predictive control (FTMPC) system discussed in this section is a constrained optimisation based control algorithm. It possesses the ability to accommodate actuator faults by reconfiguring its control structure based on pre-assessed sub-optimal control structure reconfiguration. Different approaches have been used in the design and implementation of FTMPC on complex chemical processes. Chilin *et al.* (2010b) applied distributed model predictive control to accommodate actuator faults in a reactor-separator process using redundant control input to offer extra control flexibility. MacGregor and Cinar (2012) implemented FTMPC on injection-moulding and batch polymerisation processes by treating actuator faults as unmeasured disturbances without necessarily changing the input-output control structure of the system. Kettunen (2010) in his PhD thesis used control allocation (CA), an idea first presented by Buffington and Enns (1996) for FTMPC control reconfiguration to accommodate actuator faults in a complex de-aromatisation process. These approaches have some form of actuation redundancy to provide extra capabilities for actuator fault accommodation when it occurs. In this work however, the implementation of FTMPC on CDU with reduced control structure dimension (restructured input-output pairing) and the non-trivial task of model-updating for the re-assigned FTMPC configuration during implementation are considered (Lawal and Zhang, 2017a). The integration of FTMPC and FTIC to accommodate actuator and sensor fault is also considered.

In the design of the FTMPC, it is assumed that the plant model is linear, in the form of step response model; that the performance index is quadratic and that the constraints are of linear inequalities form. Given the performance index J at time instant k for a

fault-free system as:

$$J(k) = \sum_{i=1}^{P} \|\hat{y}(k+i \mid k) - r(k+i \mid k)\|^2_{\Gamma_{u(i)}} + \sum_{i=1}^{C} \|\Delta\hat{u}(k+i \mid k)\|^2_{\Gamma_{y(i)}} \tag{6.1}$$

where $\hat{y}$, r and $\Delta\hat{u}$ are the predicted output vector, reference trajectory of the output and changes of the input vector respectively; P and C are the prediction horizon and control horizon respectively; while Γ_u and Γ_y are the positive-definite weighting matrices for predicted errors and control moves respectively. Γ_u and Γ_y are used effectively as tuning parameters to give satisfactory dynamic performance. For the FTMPC design, it is assumed the number of manipulated variables equals that of controlled variables, giving a square step response model in a matrix form. The MPC usually serves as master controller, setting the reference points for the lower level controllers, mostly PID controllers for effective control. The control performance of an MPC controller relies mainly on the predictive performance of the model. In the presence of an actuator fault, the dimension of the manipulated variable reduces, and there may be the need to reconfigure the output-input pairing structure of the MPC in absence of redundant manipulated variable to compensate for the loss. This is achieved by using the pre-assessed control structure reconfiguration when certain control input becomes unavailable. The reconfiguration seeks to pair outputs with the most effective inputs in the presence of the fault using tools like RGA and dynamic RGA, so as to minimise loss of effectiveness in the control system. This approach is similar to the one employed in Chapter 4 to accommodate actuator fault. Subsequently, the MPC objective function is adjusted as appropriate to reflect the isolated control input and the restructured output-input pairing. The resulting FTMPC maintains the essential features of the original MPC – making use of the system reduced explicit model to predict output response, the use of receding horizon idea and optimizing the predicted plant behaviour through computation of control signal.

6.3 Integration of FDD, FTIC and FTMPC

The proposed FTMPC only accommodates actuator faults, and it is integrated with the FTIC scheme proposed in Chapter 3 for a complete fault-tolerant control system that is capable of accommodating both actuator and sensor faults. The two control strategies share the same FDD scheme for initial detection and identification of actuator and sensor faults respectively. Figure 6.1 presents the complete FTCS with FTIC and FDD inte-

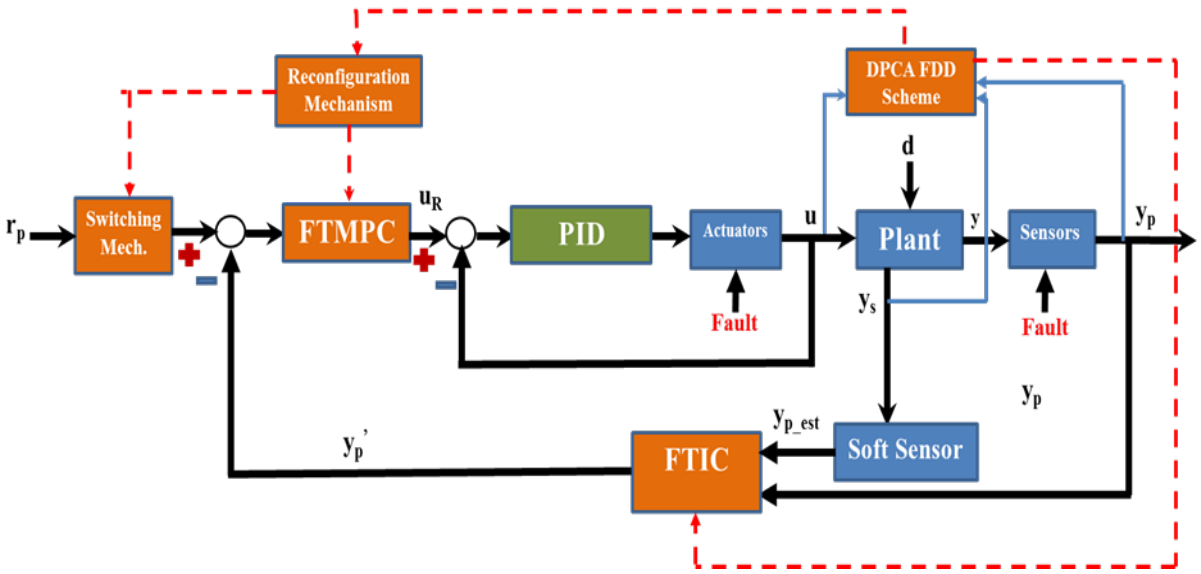


Figure 6.1: FTCS with integrated FTMPC, FTIC and FDD

grated, similar to Figure 3.4. The notable difference is the replacement of reconfigurable PID controller with FTMPC and PID controllers as master and slave controllers respectively, and the way the control structure reconfiguration mechanism is achieved. The fault detection and identification scheme integrated with the FTMPC and FTIC was discussed in Section 3.2.1, and the FTIC was as developed in Section 3.3.2 using either dynamic principal component regression or dynamic partial least squares technique for controlled output estimation.

6.4 Application of the proposed FTMPC to Crude Distillation Unit

The proposed FTMPC, integrated with FTIC and FDD is implemented on the atmospheric crude distillation unit described in Chapter 4, Sections 4.4.1 and 4.4.2 to accommodate actuator and sensor faults respectively (Lawal and Zhang, 2017a). The interactive dynamic CDU was developed in HYSYS and integrated with MATLAB application to simulate the unit. It consists of a train of heat exchangers, an atmospheric CDU with a 3-phase condenser attached, a vacuum CDU, three pumparound cooling circuits, three side draws with stripper attached to each, crude furnace, several separator vessels and 29 control loops, as presented in Figure 4.21. The CDU has five product streams – naphtha, kerosene, diesel, atmospheric gas oil (AGO) and the CDU residue. Seven products quality variables – ASTM D1160 cut-points at 0% and 100% for kerosene, ASTM D1160

cut-points at 90% and 95% for diesel, ASTM D93 flash points for kerosene and AGO, and AGO viscosity at $210F$ are monitored to ascertain the quality of the products, as presented in Table 4.12.

Five control loops are used to investigate the effectiveness of the FTMPC approach. The controlled variables – bottom boil-up flow (y_1), stage 1 temperature (y_2), diesel temperature (y_3), AGO temperature (y_4) and crude flash zone temperature (y_5) are directly controlled by the CDU bottom steam (u_1), reflux flow rate (u_2), side stripper 2 (SS-2) steam flow rate (u_3), side stripper 3 (SS-3) steam flow rate (u_4) and the furnace heat output (u_5) respectively, as discussed in Section 4.4.3 and presented in Table 4.14. The five control loops are each simulated in open loop for 700 minutes with 30 seconds sampling time, making necessary changes to their set points to collect sufficient data for model identification. Data for the open loop responses to changes in the manipulated variables collected during simulation is then used to develop a set of first order plus dead time (FOPDT) models using the System Identification Toolbox in MATLAB. Equation 6.2 below presents the 5 by 5 transfer function models of the system.

$$G_5(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.0535e^{-3.59s}}{1+1.414s} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.191}{1+1.012s} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{6.2}$$

A model predictive controller is developed in HYSYS using the models obtained above, with appropriate constraints applied to the controlled variables and the manipulated variable moves. The MPC serves as master controller to the five low-level PID controllers, setting their set-points in a cascade control structure. The MPC controller in HYSYS is based on optimisation of a quadratic objective function involving the predicted errors. Figures 6.2 and 6.3 show the settings for the FTMPC in HYSYS for the fault-free system, while Figure 6.4 presents the performance of the FTMPC to set-point changes and disturbances.

The fault-free system under FTMPC is simulated for 400 minutes with 30 seconds sampling time through MATLAB using the settings presented in Figures 6.2 and 6.3, and Figure 6.5 presents its product quality variables. With the established stable operation of

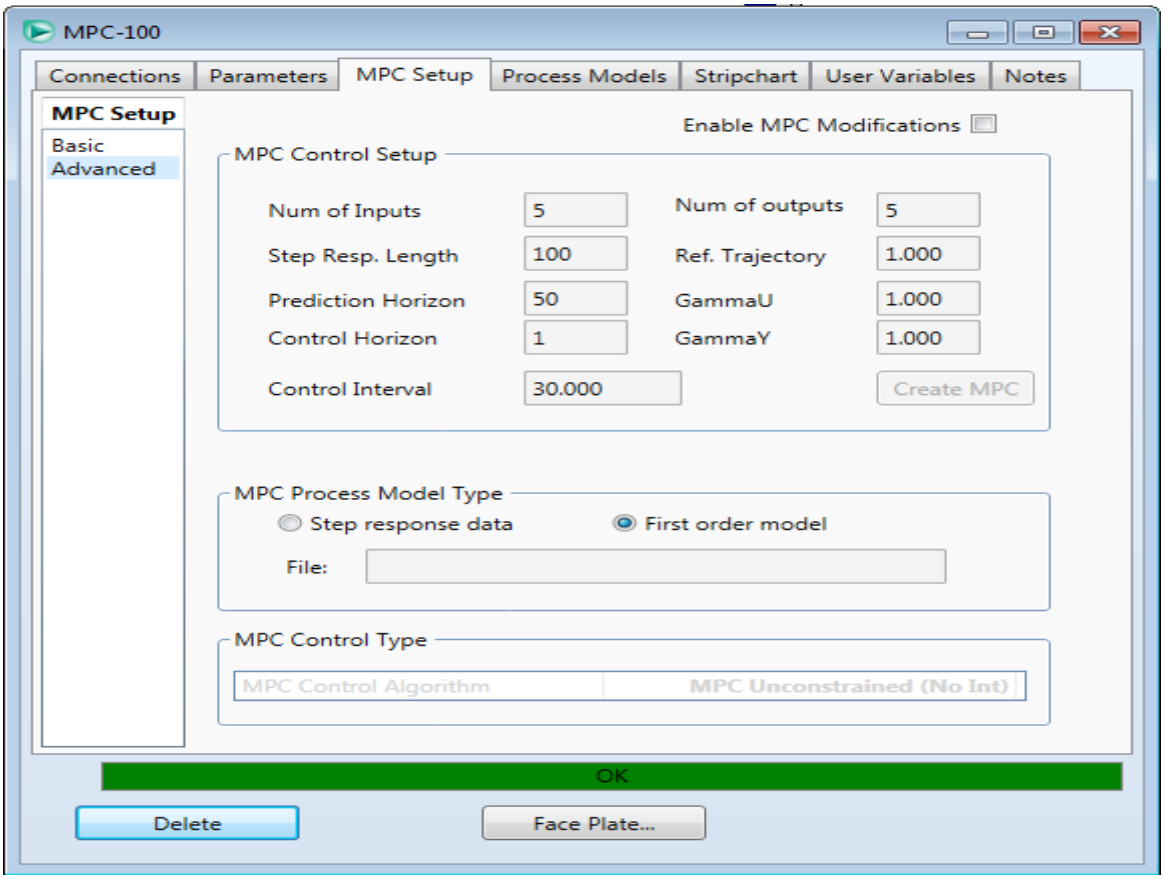


Figure 6.2: FTMPC developed in HYSYS

the system, possible FTMPC control structure reconfiguration under faulty actuators is undertaken using RGA and DRGA tools and presented in Tables 6.1 and 6.2, as discussed in Section 4.4.3. $F1$ and $F2$ in Tables 6.1 and 6.2 are faults $F1$ and $F2$ investigated in this chapter and defined in Section 6.4.2.

6.4.1 FDD Model Development

The diagnostic model developed in Section 4.4.5 with five principal components and one time-lag ($l = 1$) for the actuator and sensor faults detection and identification is used in this chapter. Though the data used for the diagnostic model development was collected when the plant was being controlled using individual PID controllers, however, the plant's behaviour and operating conditions are the same with the same type and magnitude of disturbances added. T^2 and SPE monitoring statistics are used to detect faults, and their contribution plots are used for fault identification. The values of the monitoring indices are small and within their control limits in the absence of a fault, but are unusually large for a sustained period when a fault is present. A fault is declared after the limits of either monitoring statistics are violated for eight consecutive sampling times (4 minutes).

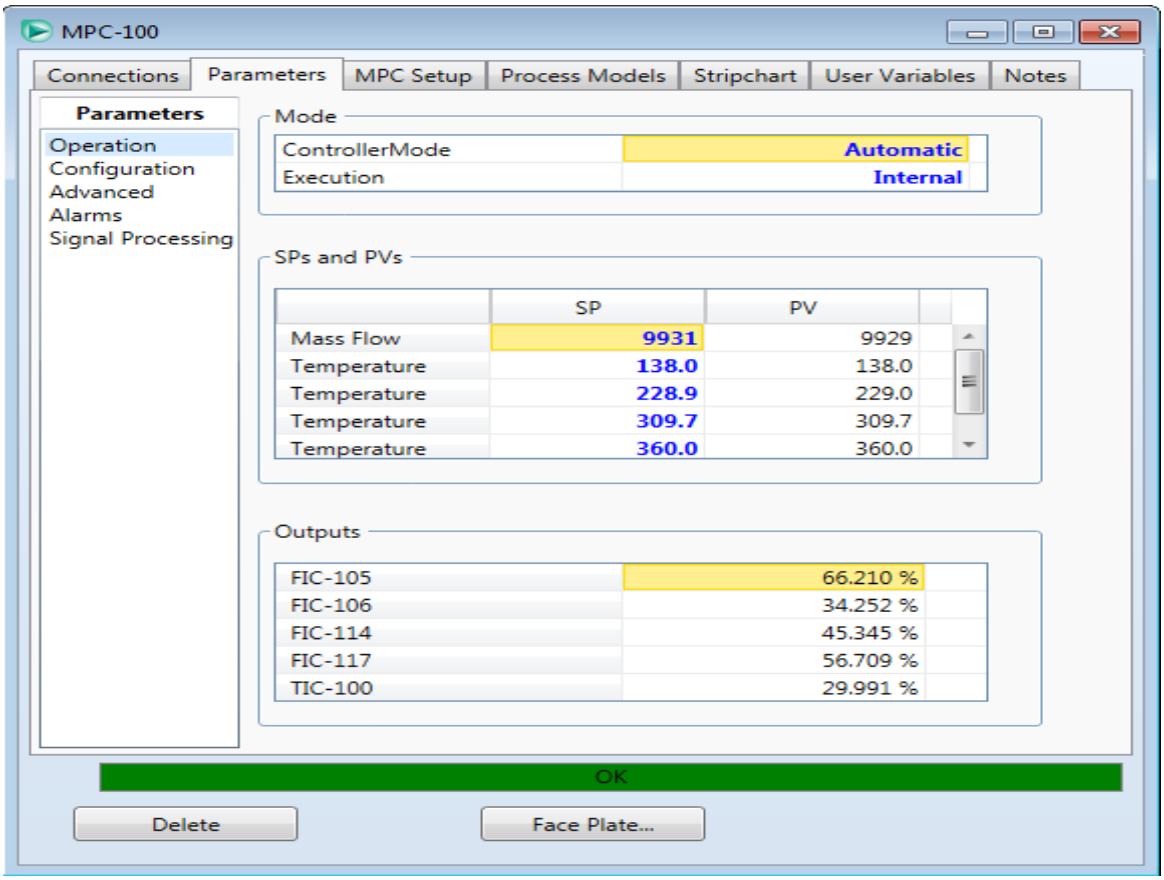


Figure 6.3: Control structure of the FTMPC in HYSYS

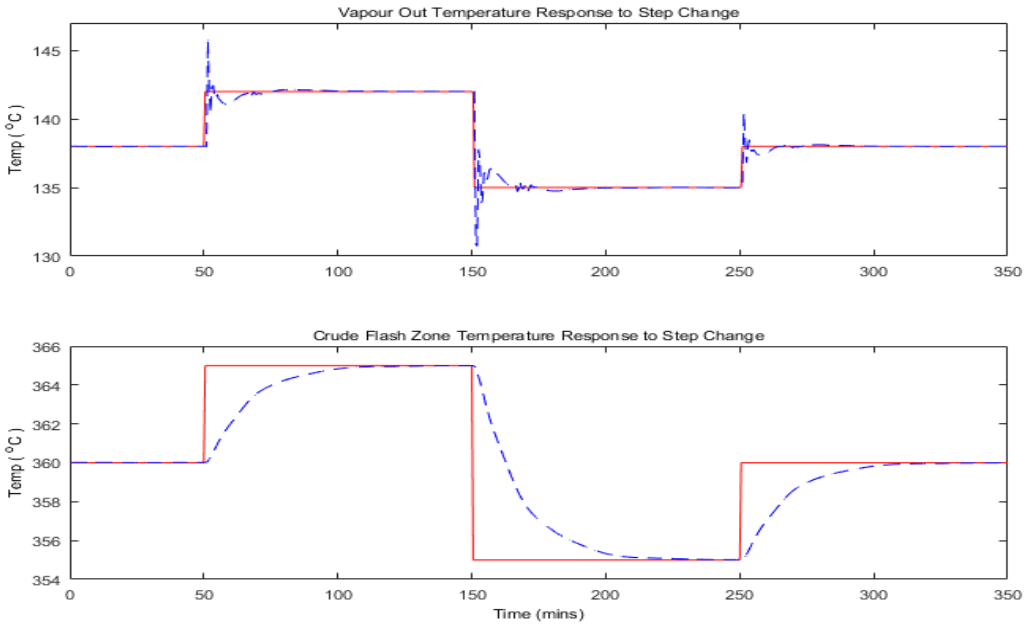


Figure 6.4: Responses to set point change for stage 1 and CFZ temperatures

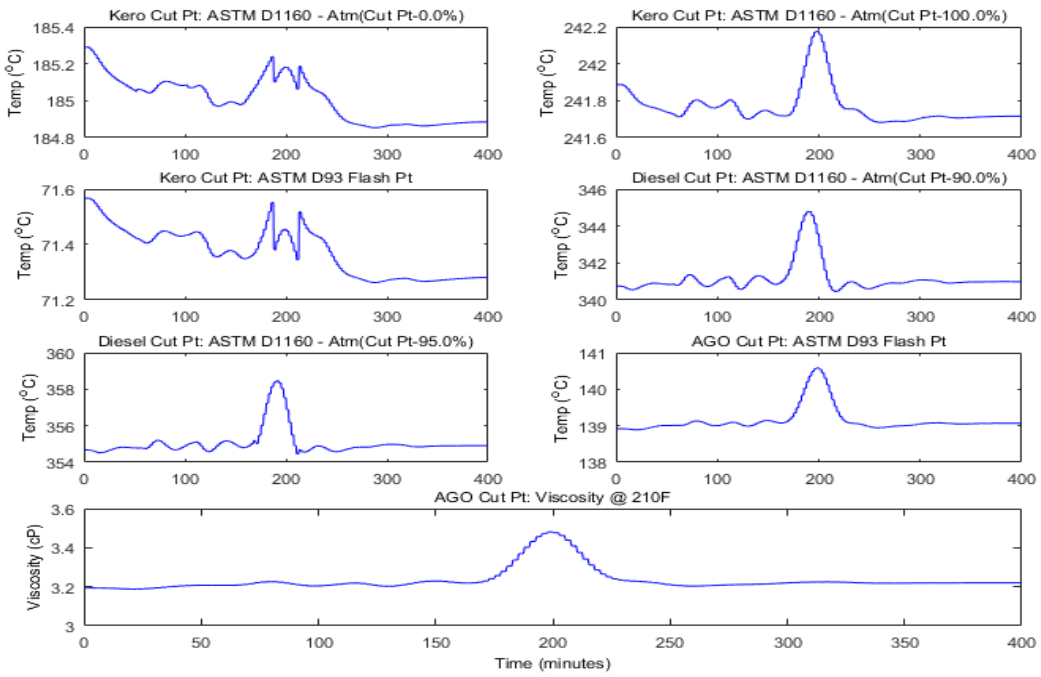


Figure 6.5: Process quality variables

Faults could also be declared faster if values of the monitoring statistics are more than double those of their respective limits for two consecutive sampling period. These criteria are deemed appropriate to ensure that no false alarm is declared, and also because of the complexity of the system being investigated. Figure 6.6 shows the T^2 and SPE monitoring statistics with control limit (red line) for the training and validating data sets under normal process operation.

6.4.2 Fault Introduction and Accommodation

Two faults each for the actuator ($F1$ and $F2$) and sensor ($F3$ and $F4$) faults are investigated in this chapter. The faults are $F1$ – reflux flow control valve fault; $F2$ – SS-2 steam control valve fault; $F3$ – crude flash zone temperature sensor fault; and $F4$ – vapour out/stage 1 temperature sensor fault, as described in Table 6.3. The two actuator faults were part of the faults investigated in Chapter 4, while the two sensor faults were previously investigated in Chapter 5. This allows for comparison of the effectiveness of the different approaches given similar fault scenarios. The full range of throttling of the actuators are restricted one at a time to values below their nominal operating conditions at 150 minutes (sample 300) during the simulation. These restrictions limit the ability of the individual control valve to maintain their respective controlled variables at set point. To

Table 6.1: Reconfigurable FTMPC PID settings

		y_1	y_2	y_3	y_4	y_5
Normal	K_p	0.78	0.50	8.35	8.48	0.51
	T_I	0.04	0.30	1.74	3.90	0.69
	T_D	–	–	–	–	–
F1	K_p	0.78	0.1000	8.35	8.48	–
	T_I	0.04	10.000	1.74	3.90	–
	T_D	–	0.0043	–	–	–
F2	K_p	0.78	–	0.45	8.48	0.51
	T_I	0.04	–	3.21	3.90	0.69
	T_D	–	–	0.10	–	–

Table 6.2: Possible inputs – outputs reconfiguration

	Manipulated Inputs		
Controlled Outputs	Normal	$F1$	$F2$
Bottom boil-up flow (y_1)	u_1	u_1	u_1
Stage 1 temperature (y_2)	u_2	u_5	–
Diesel temperature (y_3)	u_3	u_3	u_2
AGO temperature (y_4)	u_4	u_4	u_4
Crude flash zone temp. (y_5)	u_5	–	u_5

introduce the sensor faults, their respective sensor outputs were multiplied from sample 301 by a factor of 0.8 in their control loop, signifying 20% biases. Each fault was simulated for 400 minutes with 30 seconds sampling time. The responses of the five controlled variables to the actuator faults $F1$ and $F2$ as presented in Figures 6.7 and 6.8 respectively are slightly different from similar faults investigated in Chapter 4, Figures 4.30 and 4.32 respectively. The effects of the sensor faults $F3$ and $F4$ on their respective outputs are as presented in Chapter 5, Figure 5.16. Responses of the seven product quality variables to faults $F1$ and $F2$ are presented in Figures 6.9 and 6.10 respectively, while those of faults $F3$ and $F4$ are as presented in Chapter 5, Figures 5.18 and 5.20 respectively. The solid blue line and the dashed red line in Figures 6.9 and 6.10 are the normal and the fault responses respectively.

The diagnostic model described in the previous chapter is applied to monitor the operation of the interactive dynamic CDU system under the four faulty conditions to detect and identify possible occurrence of faults. Figure 6.11 presents the T^2 and SPE

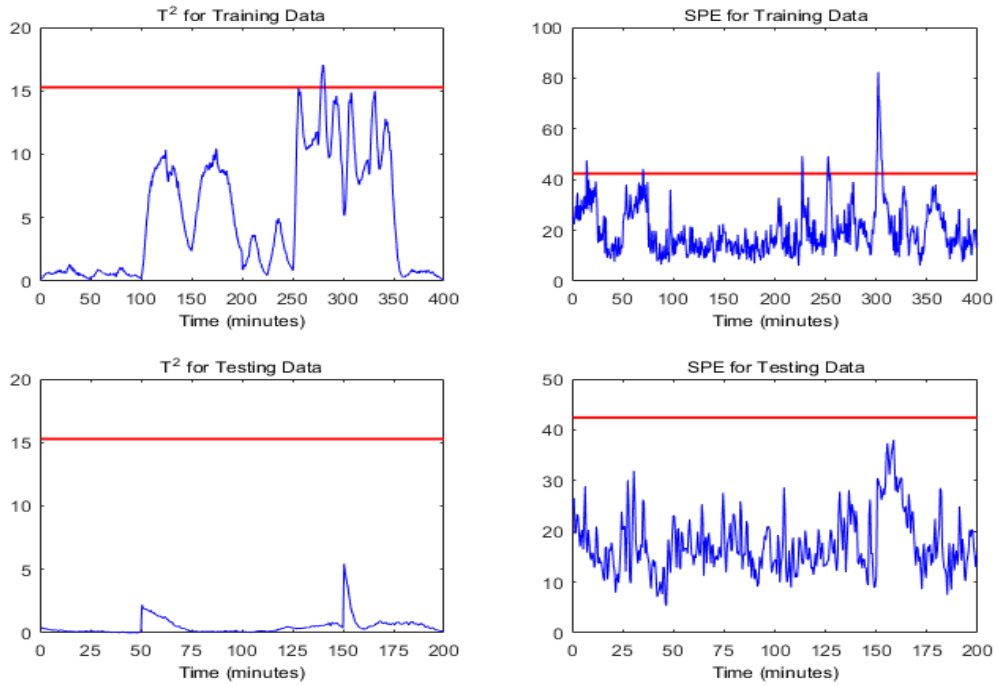


Figure 6.6: T^2 and SPE monitoring plots for normal operation

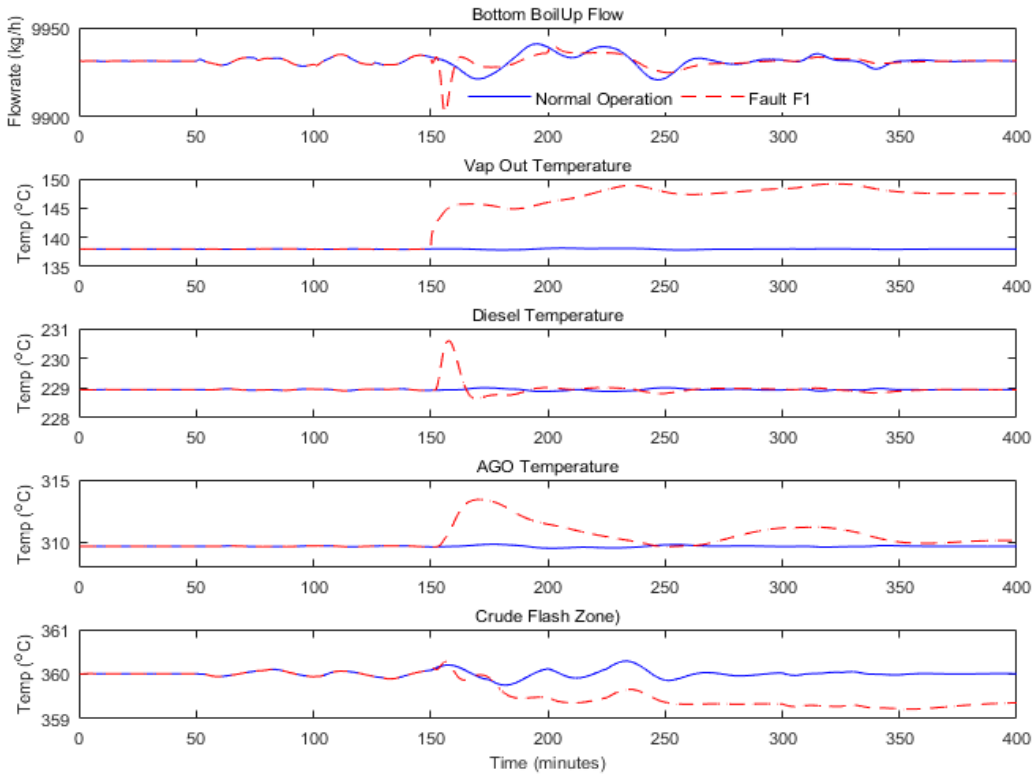


Figure 6.7: Controlled variables responses to actuator fault F1

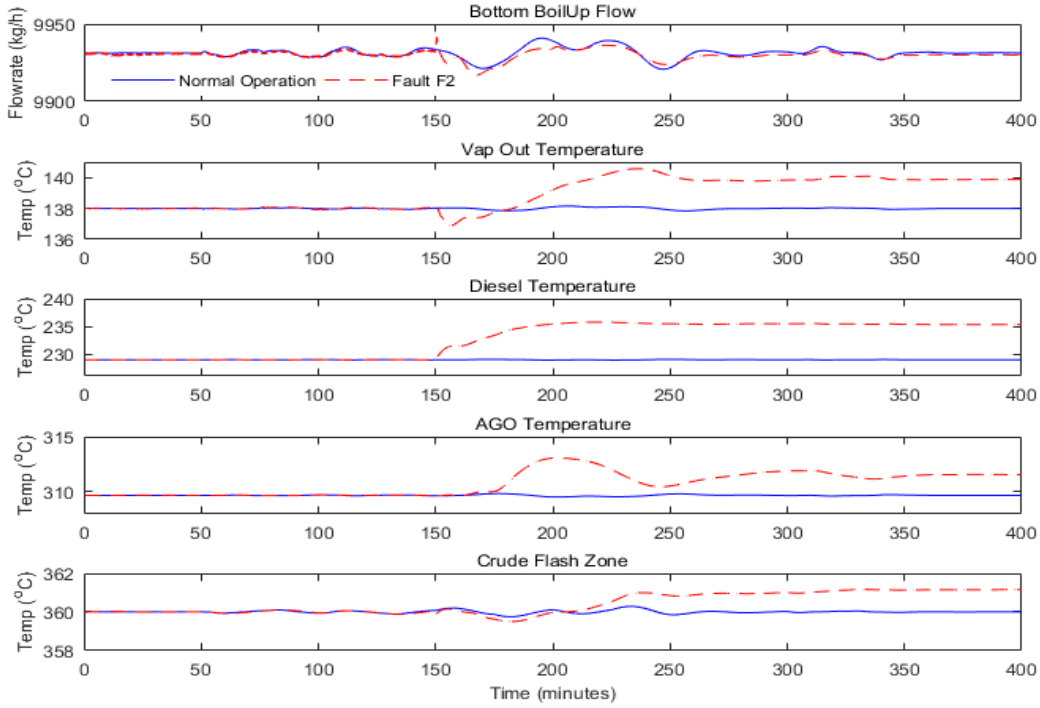


Figure 6.8: Controlled variables responses to actuator fault F2

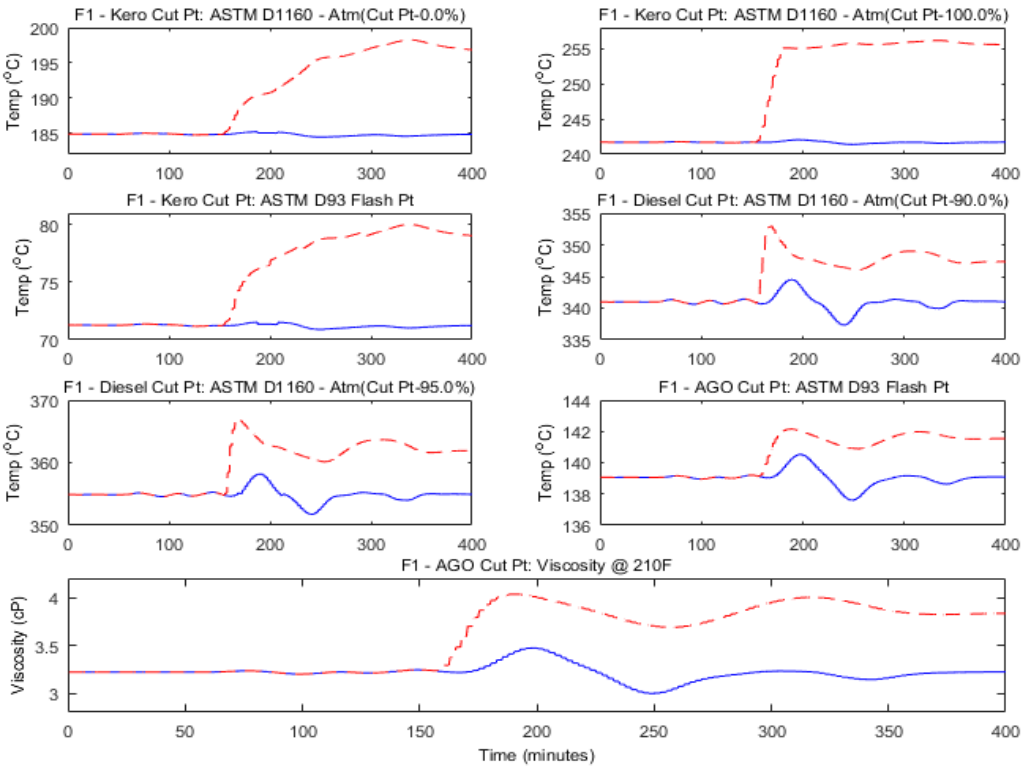


Figure 6.9: Responses of product quality variables to actuator fault F1

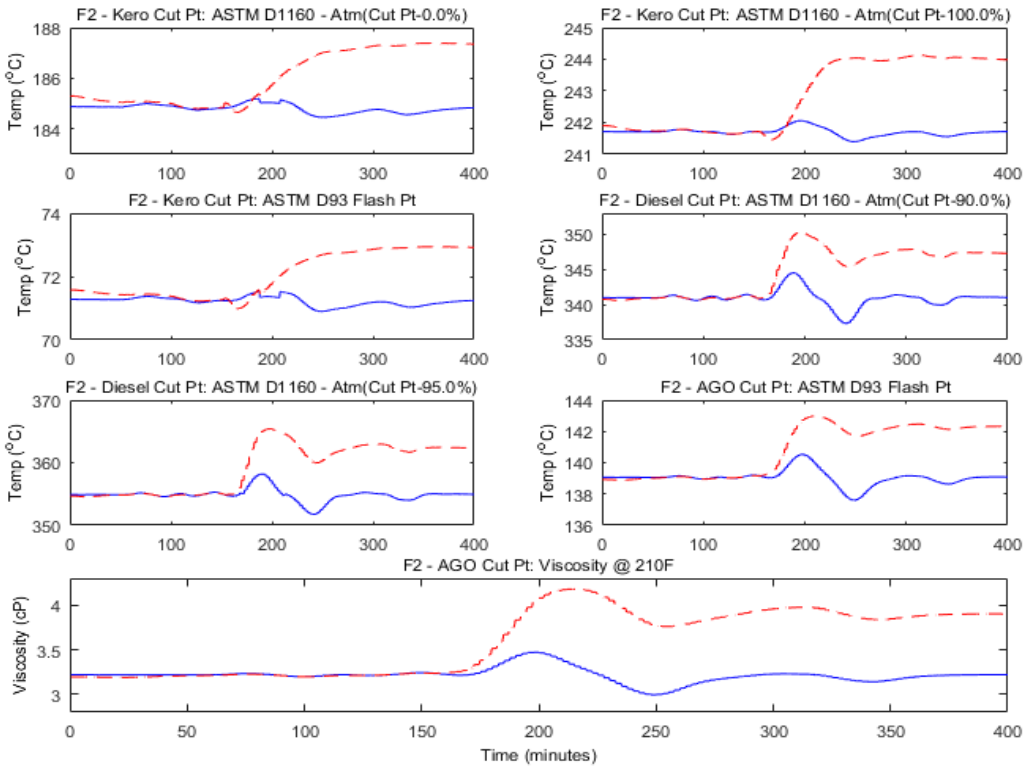


Figure 6.10: Responses of product quality variables to actuator fault F2

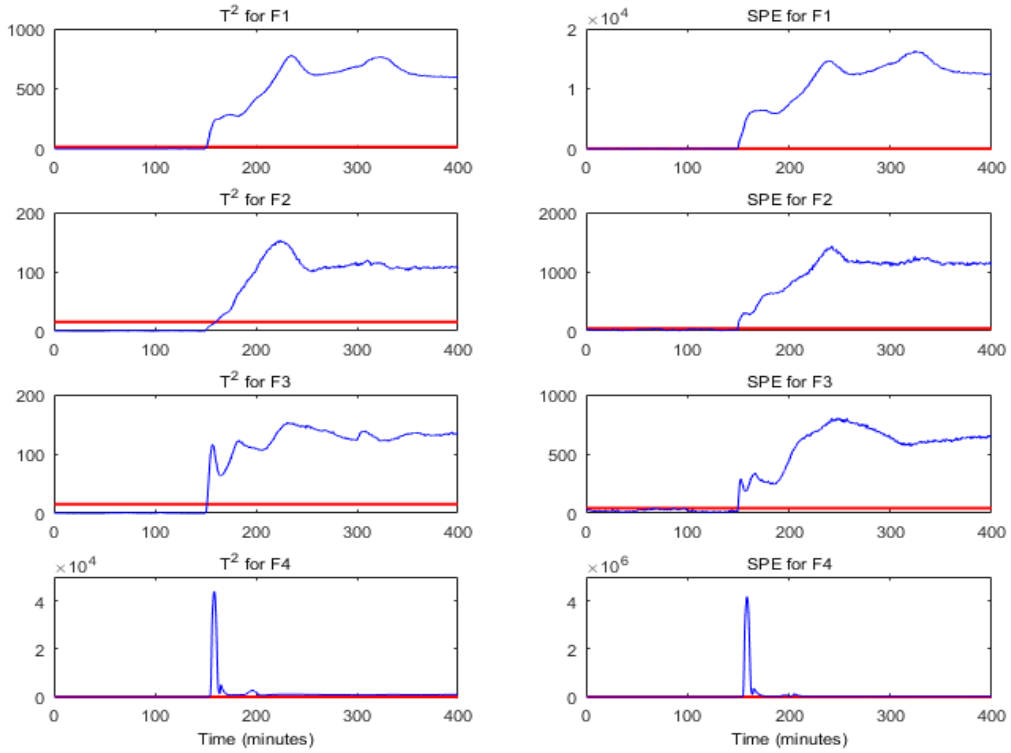


Figure 6.11: T^2 and SPE monitoring plots for faults F1 – F4

Table 6.3: Crude distillation unit fault list

Fault	Fault description
F1: FIC 106	Reflux flow control valve maximum opening restricted to 28% @ Sample 301
F2: FIC 114	Side stripper-2 steam control valve maximum opening restricted to 25% @ Sample 301
F3	CFZ temperature sensor fault with 20% loss of efficiency introduced @ sample 301
F4	Stage 1 temperature sensor fault with 20% loss of efficiency introduced @ sample 301

monitoring statistics of the fault cases $F1 - F4$. Further diagnostics are undertaken through the monitoring statistics contribution plots to identify a fault as soon as it is declared. The diagnostic procedure is as presented in the previous chapters. Figures 6.12 and 6.13 present the PCs plots for faults $F1$ and $F2$ as they are slightly different to similar faults investigated in Chapter 4, in Figures 4.41 and 4.42 respectively due to different control strategies used in each case. The PC plots for faults $F3$ and $F4$ are the same with similar faults investigated in Chapter 5, in Figures 5.24 and 5.26 respectively. The PCs that violate their limits in each fault detected are identified and used to obtain the cumulative effect of each variable's contribution to the fault. The contribution plots for faults $F1$ and $F2$ are presented in Figures 6.14 and 6.15 respectively, while those of faults $F3$ and $F4$ are as presented in Chapter 5, in Figures 5.28 and 5.30 respectively.

6.4.3 Implementation of FTMPC and FTIC

When an actuator fault is detected and subsequently isolated, say for instance fault $F1$ – reflux valve actuator fault, the dimension of the available manipulated variables reduces by 1. The performance of the impaired FTMPC system is then improved by restructuring its outputs – inputs pairing, particularly the controlled variables whose set-points are controlled directly by the FTMPC, while minimising the effects of the fault on the entire system. The dimension of the process models used for error prediction, as presented in equation 6.2 also changes from 5 by 5 to 5 by 4, taking into account the isolated manipulated variable.

Under fault $F1$, the furnace heat flow control valve (u_5) is paired with the vapour out temperature (y_2), and the five controlled variables are now being controlled by the remaining four manipulated variables (u_1, u_3, u_4 and u_5) having isolated the reflux control valve (u_2) as presented in Table 6.2. Equation 6.3 presents the reduced models of the system post-fault era. The constraints are also adjusted as appropriate in HYSYS for sub-

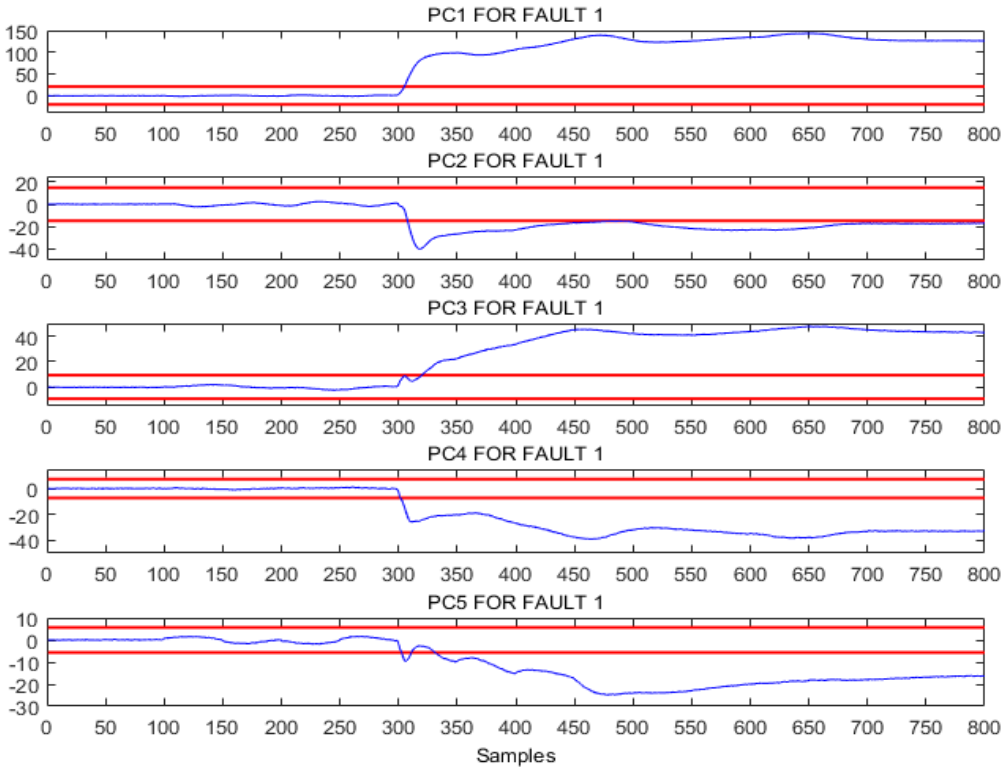


Figure 6.12: PC plots for actuator fault F1

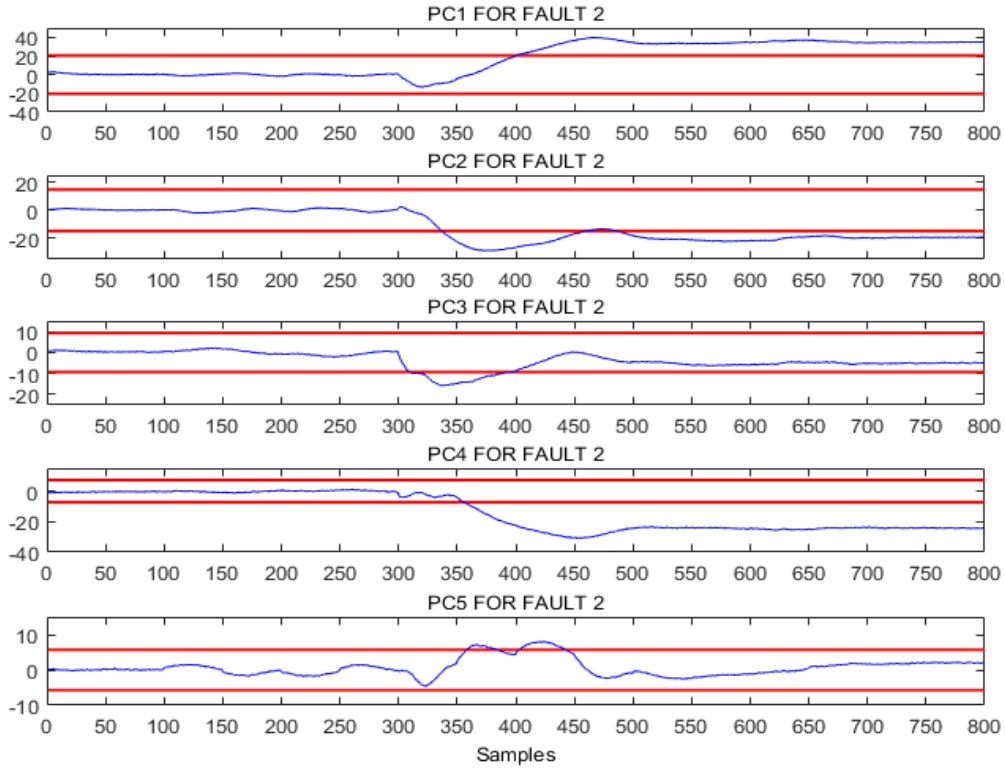


Figure 6.13: PC plots for actuator fault F2

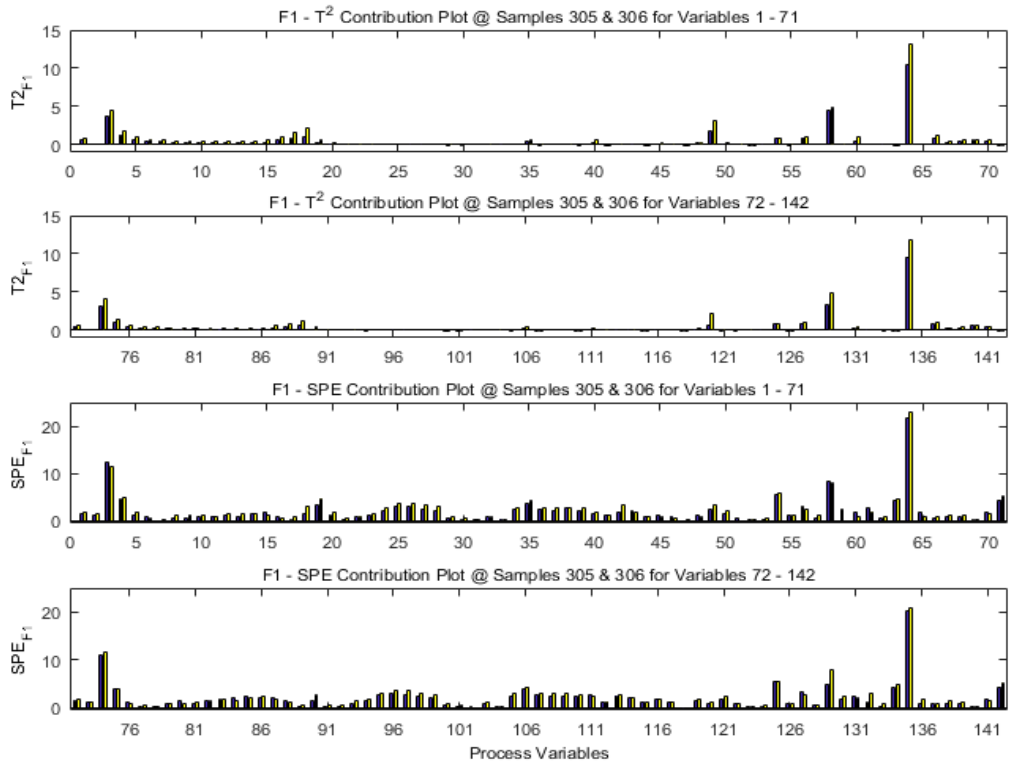


Figure 6.14: Excess contribution plots for actuator fault F1

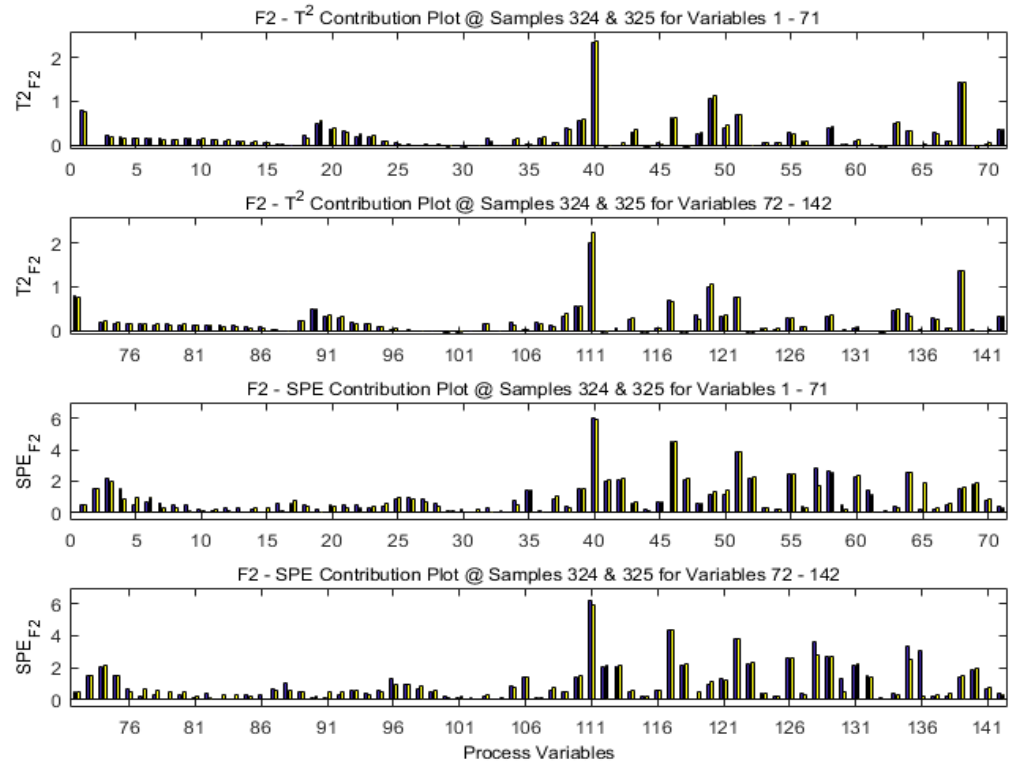


Figure 6.15: Excess contribution plots for actuator fault F2

optimal performance of the FTMPC. Similarly, actuator fault $F2$ (SS-2 steam actuator fault) is accommodated according to the reconfiguration structure presented in Table 6.2, similar to the reconfiguration structure used in Chapter 4. The reduced models of the system used for error prediction is updated and presented in equation 6.4. Figures 6.16 and 6.17 present the responses of the five controlled variables to the implementation of FTMPC on actuator faults $F1$ and $F2$ respectively, while Figures 6.18 and 6.19 present those of the seven product quality variables for the two actuator faults respectively.

$$G_{F1}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.191}{1+1.012s} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{6.3}$$

$$G_{F2}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.0535e^{-3.59s}}{1+1.414s} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{6.4}$$

Sensor faults $F3$ and $F4$ are accommodated after identification as described in Section 5.3.3 where the relevant redundant soft-sensor estimate of the controlled variable whose sensor developed fault is used in the feedback control loop instead of the faulty sensor output. In the case of sensor fault $F3$ – crude flash zone temperature sensor fault (y_5), the diagonal element corresponding to its output changes to zero, as presented in equation 5.5, isolating the faulty sensor while the corresponding diagonal element in the redundant soft sensor backup feedback signal is activated accordingly. The same procedure is applied to accommodate sensor fault $F4$ – vapour out temperature sensor fault when declared. The responses of the implementation of FTIC on faults $F3$ and $F4$ to the five controlled variables and the seven product quality variables are presented in Figures 6.20 and 6.21, and Figures 6.22 and 6.23 respectively.

6.5 Results and Discussions

The four faults investigated in this chapter were all detected and identified as previously presented in Section 4.4.7 and Section 5.3.4 in Chapters 4 and 5 respectively. The actuator faults $F1$ – reflux control valve actuator fault and $F2$ – SS-2 steam actuator faults were detected at samples 305 and 324, 2 minutes 30 seconds and 12 minutes respectively after introduction from T^2 monitoring statistic as presented in Figure 6.11; and at sample 301 for the two faults on SPE monitoring statistic. Fault cases $F3$ – CFZ temperature sensor fault and $F4$ – vapour out temperature sensor fault were detected at samples 307 and 305, 3 minutes 30 seconds and 2 minutes 30 seconds respectively after introduction on T^2 monitoring statistic. SPE monitoring plots picked up the faults at sample 301 in both cases.

There are minor variations in the fault detection time and the magnitudes of contributions of the variables responsible for the unusually large values of the T^2 and SPE monitoring statistics for faults $F1$ and $F2$, as presented in Figures 6.14 and 6.15 respectively. This is in comparison to the discussions of similar faults presented in Section 4.4.7 (Figures 4.46 and 4.47 respectively). However, the same variables responsible for similar faults investigated in Chapter 4 ($F2$ and $F3$), as presented in Table 4.19 are responsible for the faults $F1$ and $F2$ respectively. Similarly, faults $F3$ and $F4$ have the same variables presented in Table 5.5 ($F2$ and $F4$ respectively) as being symptomatic of the faults.

After the identification of the faults, FTMPC was implemented on the dynamic CDU to accommodate faults $F1$ and $F2$ according to the reconfiguration structure presented in Table 6.2. The responses of the FTMPC implementation to the five controlled variables for $F1$ and $F2$ are presented in Figures 6.16 and 6.17 while their corresponding product quality variables are presented in Figures 6.18 and 6.19 respectively. The solid blue line, the dashed red line and the dashed blue line in Figures 6.16 – 6.19 represent the normal process operation, faulty responses and responses of the controlled and product quality variables to the implementation of the FTMPC respectively. The FTMPC was able to accommodate the faults and minimise their effects on the other controlled variables. Since some of the outputs – inputs pairing are sub-optimal, it is inevitable to have some offset owing to the effect of the faults on the system. The FTMPC approach was able to significantly reduced the impact of the faults on the system, judging by the responses of the product quality variables as presented in Figures 6.18 and 6.19, notably the ASTM D1160 cut points at 0% and 100% for kerosene and ASTM D93 flash point for kerosene.

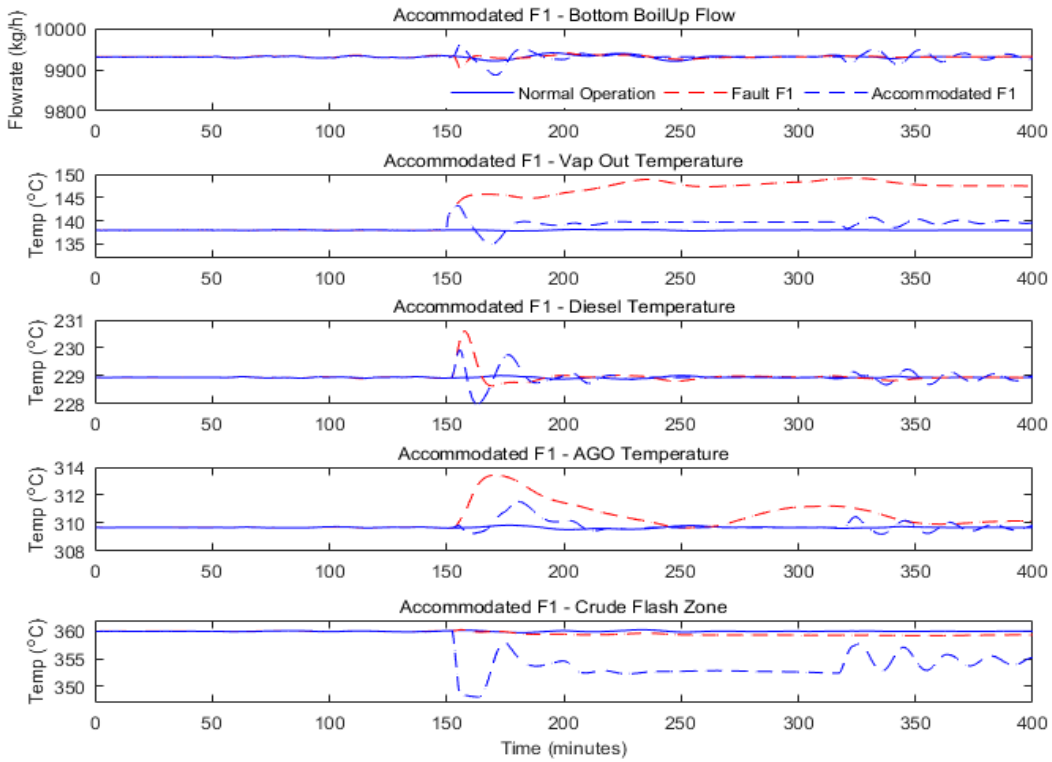


Figure 6.16: Responses of the five controlled variables to the accommodated fault F1

The results obtained are comparable to those presented in Chapter 4 as can be observed from the sum of squared error (SSE) of the control errors of the two actuator faults accommodating strategies presented in Table 6.4.

Table 6.4 gives a summary of the relative effectiveness of the two actuator FTCS – restructurable PID controllers and FTMPC – proposed in this work. As can be observed from Table 6.4, FTMPC performed better with reduced errors between the nominal operation conditions of the controlled variables before the fault and the responses achieved during fault accommodation for the reflux control valve actuator fault ($F1$). Four out of the five controlled variables recorded reduced errors compared with their responses under the restructurable PID controllers, with the exception of the Bottom boil-up (y_1) whose error of 50,332 was much larger under FTMPC. However, the effects of the implementation of FTMPC on the product quality variables during $F1$ actuator fault accommodation were not so profound. Only two (kerosene cut point at 100% and diesel cut point at 90%) out of the seven product quality variables recorded lower errors when compared to their responses under the restructurable PID controllers investigated in Chapter 4. Similarly, FTMPC performed better than the simple restructurable PID controllers in the case of

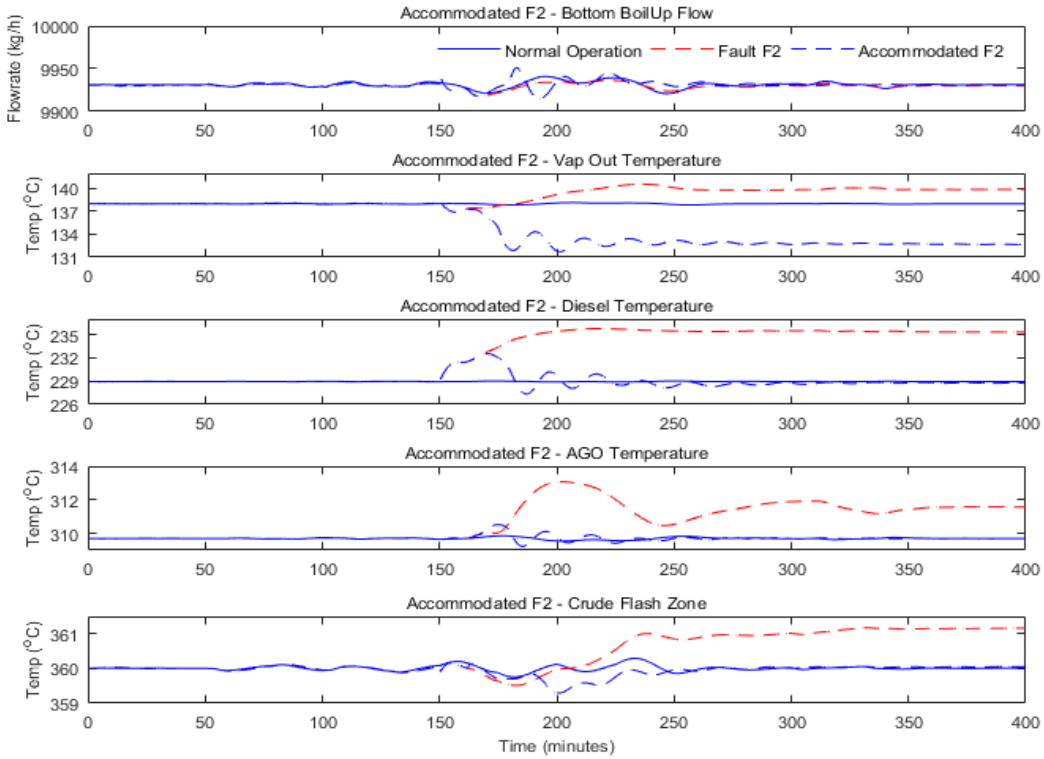


Figure 6.17: Responses of the five controlled variables to the accommodated fault F2

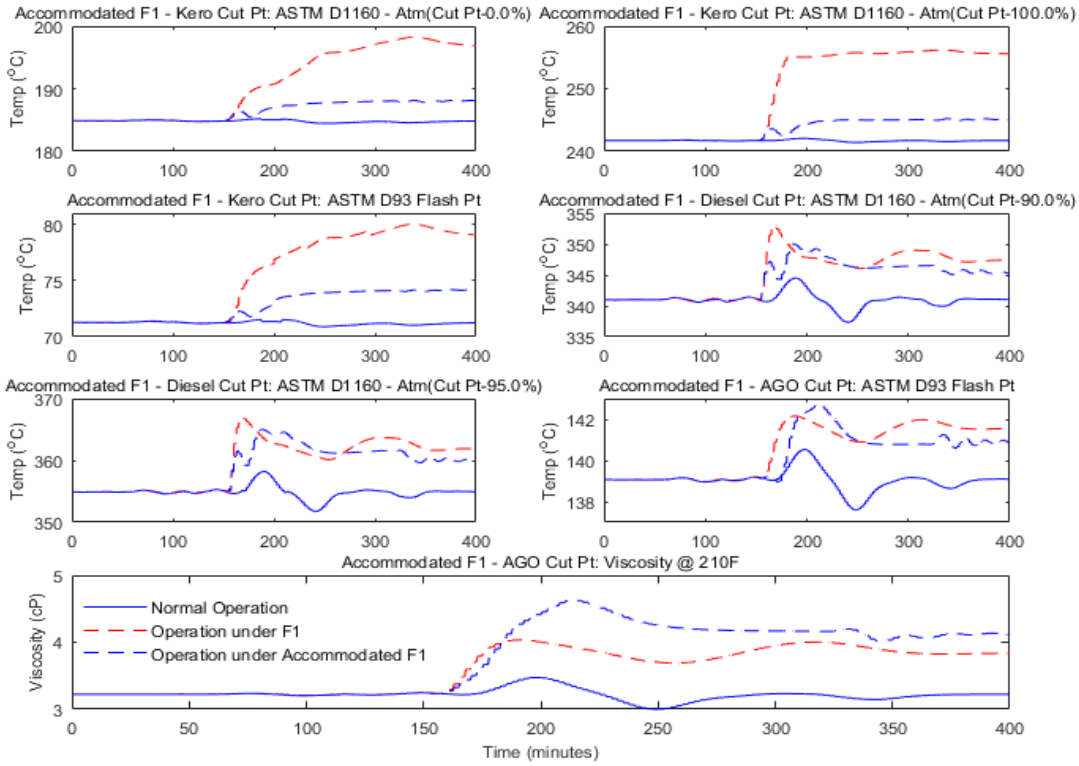


Figure 6.18: Responses of PQV to implementation of FTMPC on fault F1

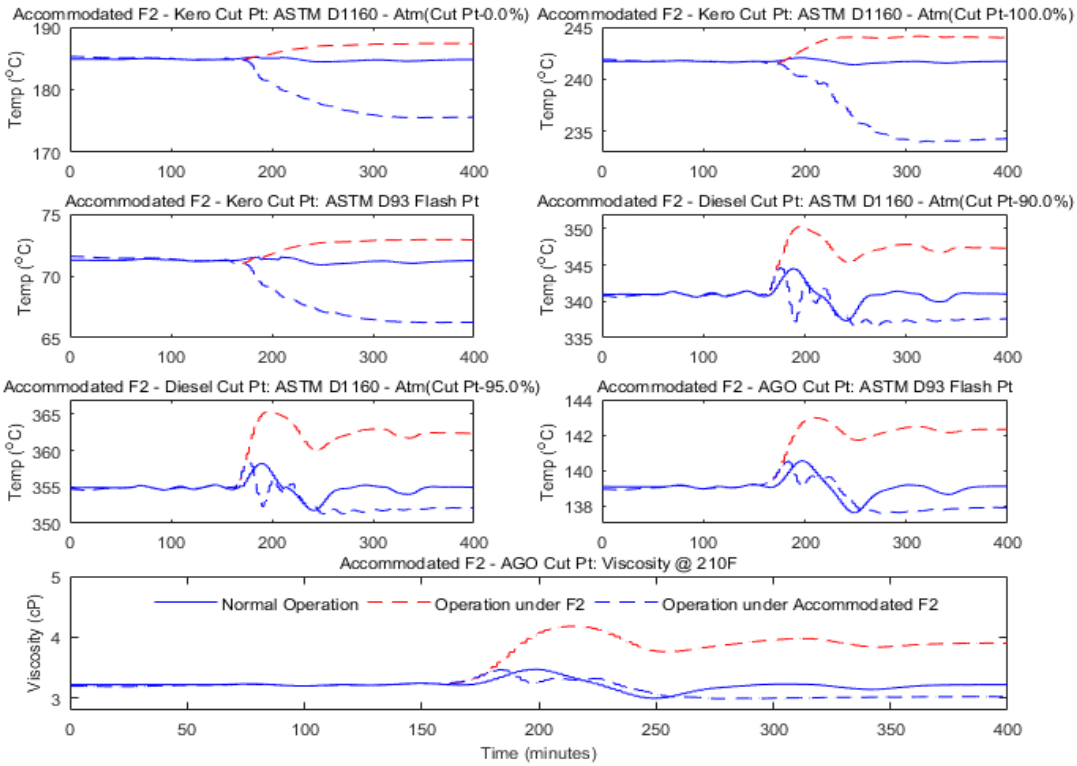


Figure 6.19: Responses of PQV to implementation of FTMPC on fault F2

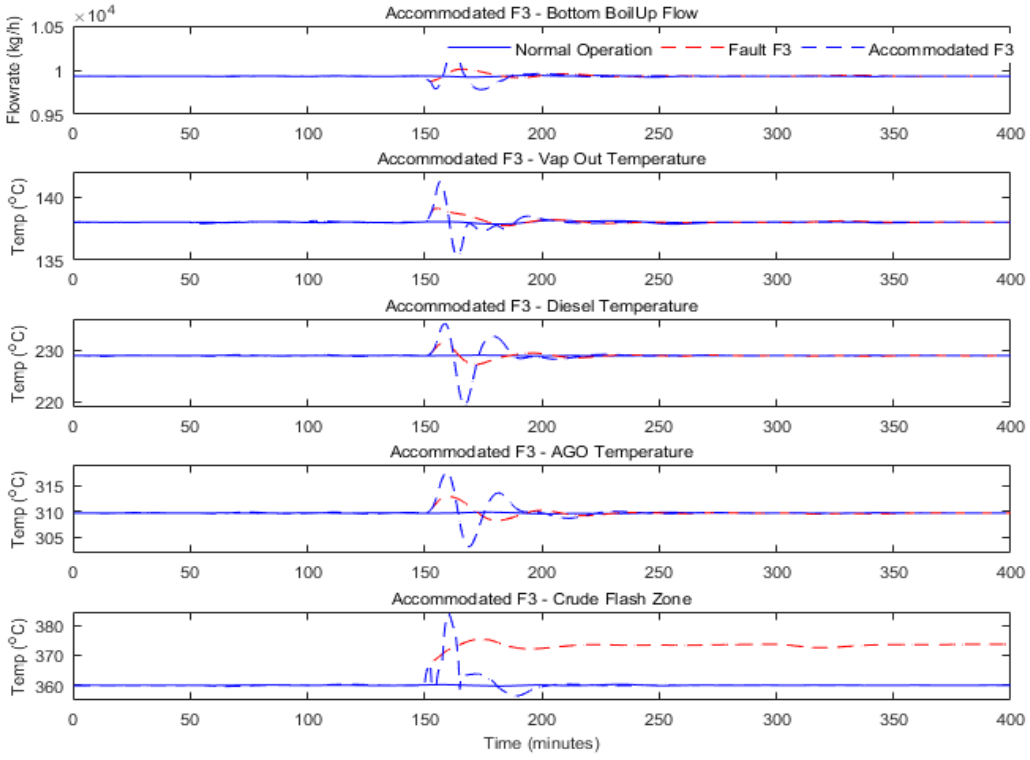


Figure 6.20: Responses of the five controlled variables to the accommodated fault F3

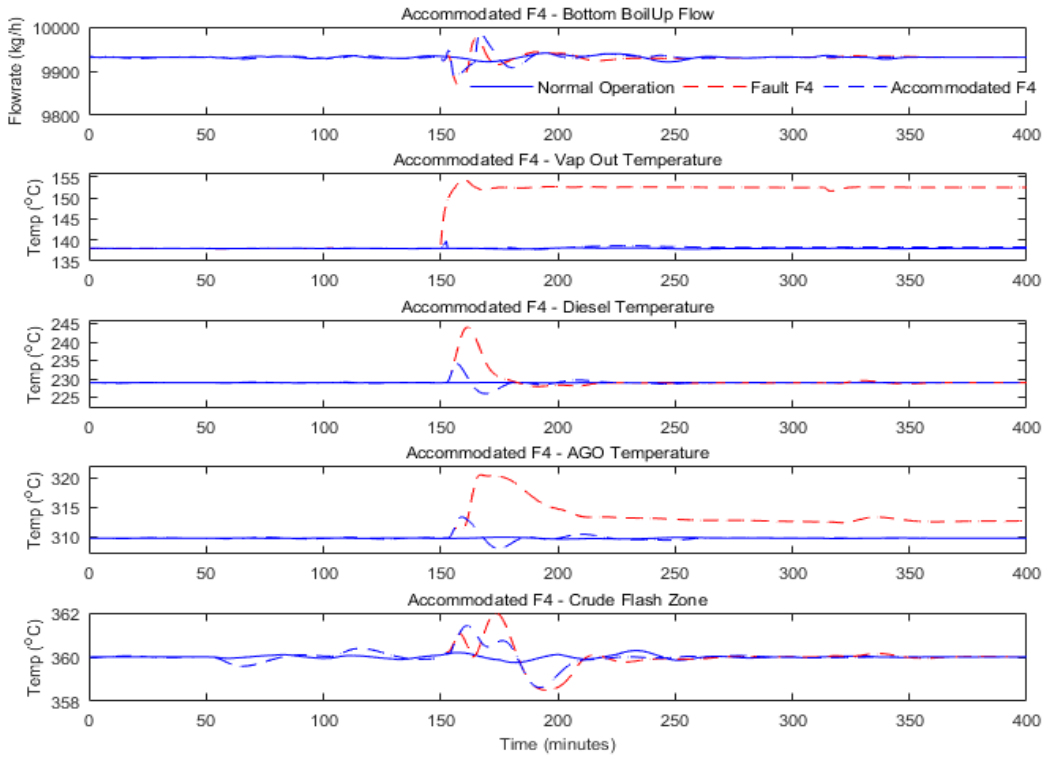


Figure 6.21: Responses of the five controlled variables to the accommodated fault F4

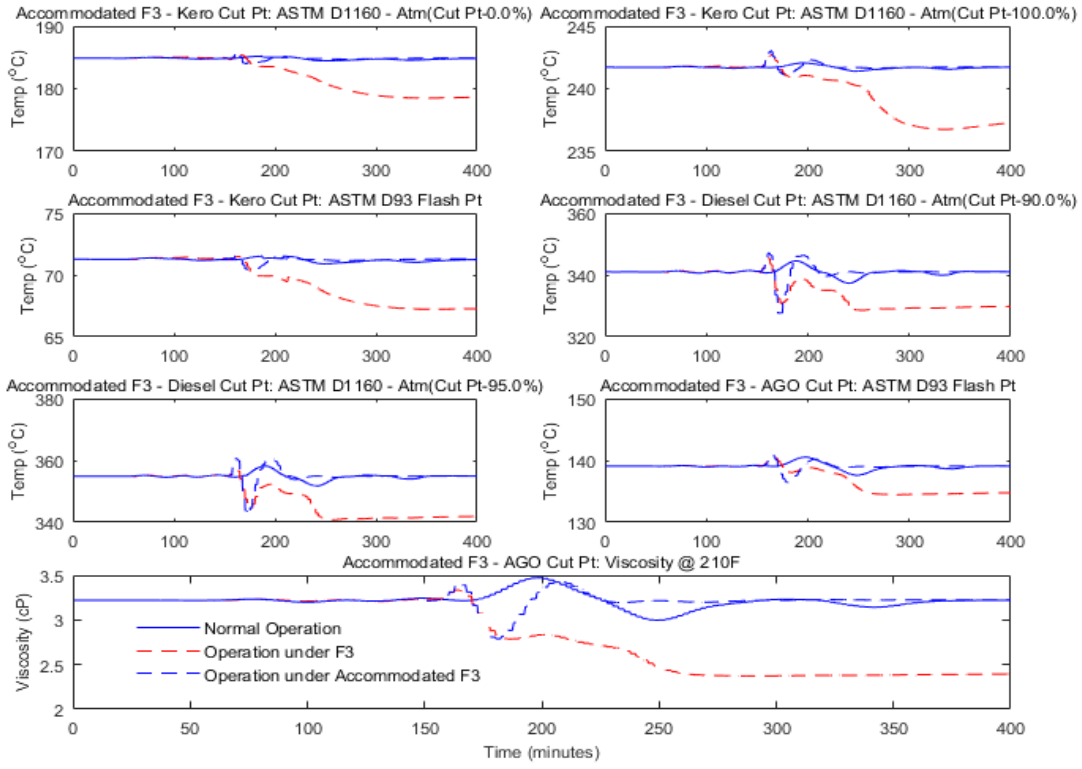


Figure 6.22: Responses of PQV to implementation of sensor FTIC on fault F3

Table 6.4: SSE of control errors for FTMPC & restructurable PID for F1 & F2

	Reflux control valve actuator fault (F1)		SS-2 steam control actuator fault (F2)	
Controlled variables	Restructurable controller (Chapter 4)	FTMPC (Chapter 6)	Restructurable controller (Chapter 4)	FTMPC (Chapter 6)
Bottom boil-up flow (y_1)	5,266	50,332	4,017	18,254
Stage 1 temperature (y_2)	6,077	1,702	13,496	12,246
Diesel temperature (y_3)	417	28	4,476	535
AGO temperature (y_4)	2,235	91	303	16
Crude flash zone temp. (y_5)	27,963	22,623	13	24
Product quality variable				
Kerosene Cut Pt: (Cut Pt-0.0%)	4,889	6,276	24,599	26,444
Kerosene Cut Pt: (Cut Pt-100.0%)	15,474	9,366	14,659	16,957
Kero Cut Pt: ASTM D93 Flash Pt	3,474	4,876	642	7,823
Diesel Cut Pt: (Cut Pt-90.0%)	19,484	18,068	15,550	4,747
Diesel Cut Pt: (Cut Pt-95.0%)	22,380	24,478	12,917	3,231
AGO Cut Pt: ASTM D93 Flash Pt	1,308	2,679	1,028	481
AGO Cut Pt: Viscosity @ 210F	360	481	29	12

SS-2 steam control valve actuator fault ($F2$) with reduced errors in three out of the five controlled variables investigated in this section. Four out of the seven product quality variables investigated under $F2$ also recorded reduced errors compared to when restructurable PID controllers were implemented to accommodate the same fault, as presented in Table 6.4.

Faults F3 and F4 are accommodated as described in Section 5.3.4 by using the backup soft sensor estimate in the feedback control loop in place of the faulty sensor output. The backup soft sensor estimates for the CFZ temperature and the vapour out temperature are as presented in Figures 5.14 and 5.15 respectively. The responses of the five controlled variables and the seven product quality variables for faults F3 and F4 are presented in Figures 6.20 and 6.21, and Figures 6.22 and 6.23 respectively, and are comparable to those obtained in Chapter 5, Section 5.3.4.

6.6 Summary

The development and application of fault-tolerant model predictive control (FTMPC), integrated with an FDD scheme and FTIC to accommodate actuator and sensor faults

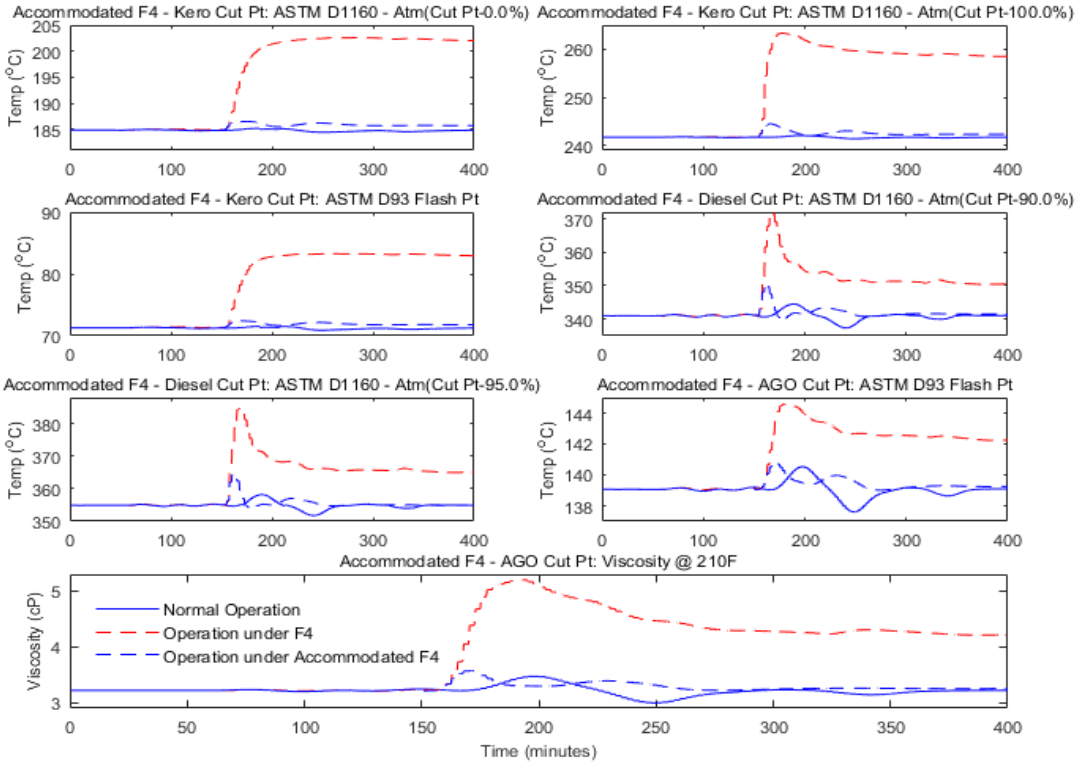


Figure 6.23: Responses of PQV to implementation of sensor FTIC on fault F4

in a crude distillation unit is presented in this chapter. The design and restructurability of the FTMPC system were first discussed. The restructurability and the effectiveness of the approach, just like most active fault-tolerant control system depend on the possibility of being able to reconfigure its input-output pairing upon detection and identification of a fault. The FTMPC system used a first order plus dead time (FOPDT) model of the plant for output prediction and RGA and DRGA tools to analyse possible control structure reconfiguration. Its FDD scheme used the DPCA technique described in Chapter 3 and FTIC also presented in Chapter 3. The proposed FTMPC was then implemented on a CDU with two faults each for both actuator and sensor. The approach was able to detect, identify and accommodate all the faults investigated, and the results compare favourably with similar faults investigated in Chapters 4 and 5. Overall, the approach proves effective in handling actuator fault sub-optimally, when there are suitable manipulated variables that could be used to lessen the effects of fault on the system.

Chapter 7

Conclusions and Recommendations for Future Works

7.1 Conclusions

This thesis investigated, and contributed to the furtherance of the development and application of fault tolerant control system (FTCS) to the oil and gas processes, particularly the distillation processing units. First, a simple restructurable feedback controller with backup signals and switchable references (actuator FTCS) was developed and applied to three different distillation processes with varying complexities – binary distillation column, the Shell heavy oil fractionator and a crude distillation unit to accommodate actuator faults in the systems. Second, a fault tolerant inferential controller (FTIC) was also developed and implemented on a binary and crude distillation units to accommodate sensor faults – temperature and flow sensor faults. Third, both the simple restructurable feedback controller (actuator FTCS) and the FTIC are integrated with a dynamic principal component analysis (DPCA) based fault detection and diagnosis (FDD) scheme to form a complete FTCS, which was then implemented on a dynamic crude distillation unit to accommodate both actuator and sensor faults. Lastly, a fault tolerant model predictive controller (FTMPC) was developed, integrated with FTIC and implemented on a dynamic crude distillation unit to accommodate actuator and sensor faults respectively on the system.

Firstly, this thesis used dynamic principal component analysis technique based FDD to quickly and effectively detect and identify faults. The DPCA technique was chosen based on its many strengths, and most importantly due to the complexities of the sys-

tems being dealt with. The FDD scheme employed T^2 and SPE monitoring statistics and their respective contribution plots together with good understanding of the processes, to effectively detect and identify faults. This was a crucial first step in the design of an effective FTCS for the distillation processes investigated. Then a simple restructurable proportional-integral-derivative (PID) controller with backup signals and switchable references (actuator FTCS) was developed, integrated with the FDD scheme and implemented on a nonlinear methanol-water separation column, the benchmark Shell heavy oil fractionator, and an interactive dynamic crude distillation unit to accommodate actuator faults. The effect of disturbances on actuator faults propagation was also reported in this thesis. Disturbances, depending on its direction and magnitude, can amplify a rather minor undetected fault as evident in the results presented on the methanol-water separation column in Chapter 4. The restructurability of the developed FTCS was based on pre-assessed possible control structure reconfiguration using RGA and DRGA system interaction analysis tools. The approach proved effective in accommodating actuator faults in the three distillation processes considered. However, implementation of the proposed actuator FTCS is system dependent as different layers of performances have to be critically analysed for the whole system and configured as appropriate. Actuator fault accommodation is not always possible as observed in the cases of the Shell heavy oil fractionator and the crude distillation unit. Another important factor that should be critically considered in the design and implementation of actuator FTCS on any system is a very good understanding of the process in terms of its products quality and economics. This is very important as it determines the level of degradation that is acceptable in the system or its products before the operation becomes unprofitable.

Secondly, the FTIC developed in this thesis used inferred primary controlled variables from the measured uncontrolled secondary variables, based on DPLS and DPCR soft sensor estimation techniques. The inferred controlled variables were used in place of the faulty sensor outputs in the feedback control loops to maintain the integrity of the methanol-water separation column and the crude distillation unit investigated in this thesis. The FTIC also made use of the DCPA based FDD to first detect and identify the flow and temperature sensor faults investigated in the systems for prompt and effective faults isolation. The proposed FTIC approach is proved to be very effective in accommodating all the sensor faults investigated, and was able to maintain the integrity of each control loop, thereby preserving the specified products quality variables in each case.

Thirdly, the proposed simple actuator FTCS and the FTIC developed were both inte-

grated with the DPCA based FDD, to form a complete FTCS capable of accommodating actuator and sensor faults, and applied to the interactive dynamic crude distillation unit. Successive sensor and actuator faults were detected, identified and accommodated in the dynamic crude distillation unit using the complete FTCS.

Lastly, fault-tolerant model predictive control (FTMPC) was developed, and integrated with the FTIC and FDD schemes to form another variant of a complete FTCS to accommodate actuator and sensor faults respectively. The complete FTCS was again implemented on a crude distillation unit as presented in Chapter 6. The approaches were able to accommodate actuator and sensor faults in the simulated system, when suitable input-output pairing was available after declaration of a fault according to the pre-assessed control reconfiguration structure. The results obtained in this case are comparable to the ones obtained when actuator FTCS and FTIC were used. However, actuator fault accommodation is not always possible without suitable back-up or satisfactory input-output pairing, irrespective of the level of sophistication of the control system. It will be possible to implement this approach for systems with tens of input-output pairings, though several hundreds of possible back-up models will be required for possible reconfiguration, or at the very least pre-assessed.

In conclusion, the primary aim of this thesis, which is the development and implementation of fault-tolerant control systems – simple restructurable feedback controllers with backup feedback signals and switchable reference points (actuator FTCS), fault-tolerant model predictive controllers (FTMPC) and fault-tolerant inferential controllers (FTIC) to accommodate actuator and sensor faults on distillation processes, particularly crude distillation unit was achieved. All the nine objectives set out at the onset of this thesis were achieved, and the thesis is able to further advance the existing body of knowledge in the development and application of fault-tolerant control systems to industrial distillation processes with the contributions listed below.

- Simple active restructurable feedback controllers with backup feedback signals and switchable reference points (actuator FTCS) was designed and integrated with DPCA FDD scheme to accommodate actuator faults in binary and crude distillation processes.

- Fault-tolerant inferential controller (FTIC) that uses dynamic principal component regression (DPCR) and dynamic partial least square (DPLS) techniques for controlled variables estimations from measured uncontrolled secondary variables was

developed, and integration with the DPCA based FDD scheme to accommodate sensor faults on simple and complex distillation processes.

- The integration of the FTIC and the actuator FTCS with DPCA FDD scheme as a complete fault-tolerant control system (FTCS) was achieved and implemented on a crude distillation unit to accommodate both actuator and sensor faults.

- The development and implementation of fault-tolerant model predictive control (FTMPC) integrated with both FTIC and DPCA fault diagnostic model to accommodate both actuator and sensor faults in a crude distillation unit was also achieved.

7.2 Recommendations for future works

This thesis and several other works on the design and application of FTCS are premised on an assumption that, only one actuator fault occurs at any given time as it is very unlikely that two or more actuators could fail simultaneously. However, it will be interesting to further investigate strategies that could be deployed to accommodate possible simultaneous multiple actuator failures in the oil and gas processing industry. This will without a doubt involve the design of fault diagnostic scheme that is able to detect and identify multiple simultaneous faults, whether actuators, sensors or other system components. It will be an interesting research area as the combined effects of the faults, particularly actuators may make it difficult to determine the severity of individual actuator fault and make their accommodation even more difficult. Our future work in relation to distillation processes will look at the possibility of detecting and accommodating multiple simultaneous actuator faults in crude distillation units.

Similarly, accommodation of multiple sensor faults occurring simultaneously is worth further investigation. The combined effects of the multiple sensor faults on the secondary variables that would be used to predict the concern controlled variables might be difficult to isolate, which could result in inaccurate predictions of the controlled variables of interest. The possible implementation issues that might arise as a result of using the existing techniques like the FTIC strategy presented in this thesis to accommodate multiple sensor faults could be looked at in our future researches on sensor fault accommodation is distillation processes.

The effectiveness of most fault tolerant control systems, especially the ones designed

for actuator faults accommodation, rely heavily on the availability of suitable redundant or essential manipulated variable that could be activated or switched as appropriate in the face of an actuator fault to enhance the availability and reliability of the accommodating control system. To have redundant actuators in complex chemical processes like oil refineries in anticipation of possible faults to some critical actuators might not be desirable due to potential large number of such actuators, the cost implication and other factors. There could be ways in which the faulty actuator(s), depending on the nature and severity of the fault could still be used sub-optimally in maintaining the integrity of both the process and the control system. It is certainly an area that is worth investigating further.

Despite their strength, the use of data-based FDD techniques like DPCA and DPLS for fault diagnosis also have limitations in the sense that, multiple faults could have similar variables contributing to their larger than usual monitoring statistics, leading to a very difficult fault identification process. More so, for complex systems like the crude distillation unit investigated in this thesis, further investigation for exhaustive possible faults and the probable use of sectionalised DPCA for fault diagnosis might be required. The sectionalised DPCA could be used to investigate faults in different parts of the process as against lumping the entire process together, and this could be further investigated.

The use of combined model-based and data-based fault diagnostic scheme for actuator and sensor fault detection and accommodation in distillation processes is another fascinating research area that could be explored in the future. The combined FDD technique could leverage on the strength of data-based techniques such as the ease of computation, its ability to handle nonlinearity and large amount of process data. The challenge of obtaining a high fidelity model for the combined FDD could be solved by using process data from the system to build suitable models like state space to combine with the DPCA model from the data-based approach. Though, the combined FDD approach have the potential to make fault detection and identification less effective as any missing information in the data used to develop the model could be amplified.

Another research area that is worth exploring is the application of DMPC technique to accommodate faults in a crude distillation unit where two sets of manipulated variables could be used to accommodate any potential actuator fault. The entire refinery operation that includes vacuum and atmospheric crude distillation units and fluid catalytic cracking unit could all be investigated for actuator and sensor faults using this approach. It is our believe that advances in some of these future researches will have great influence on the

application of FTCS in the refinery operation.

References

Arkin, R. C. & Vachtsevanos, G. (1990), 'Qualitative fault-propagation in complex-systems', *Proceedings of the 29th IEEE Conference on Decision and Control, Vols 1-6* pp. 1509–1510.

Astrom, K. J. (1991), 'Intelligent control', *Proc. First European Control Conference. Grenoble,* France pp. 2328–2339.

Bakshi, B. R. (1998), 'Multiscale pca with application to multivariate statistical process monitoring', *AIChE Journal* **44**(7), 1596–1610.

Basseville, M. (1988), 'Detecting changes in signals and systems - a survey', *Automatica* **24**(3), 309–326.

Beard, R. V. (1971), Failure accommodation in linear systems through self-reorganization, Report, Man Vehicle Lab, MIT: Cambridge, Massachusetts.

Blanke, M., Frei, C., Kraus, F., Patton, R. J. & Staroswiecki, M. (2000), 'What is fault-tolerant control?', *In Proceedings of the 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Process.* Budapest, Hungary. pp. 40–51.

Blanke, M., Izadi-Zamanabadi, R., Bgh, S. A. & Lunau, C. P. (1997), 'Fault-tolerant control systems — a holistic view', *Control Engineering Practice* **5**(5), 693–702.

Blanke, M., Kinnaert, M., Lunze, J. & Staroswiecki, M. (2003), *Diagnosis and fault-tolerant control*, Vol. 22 of *Robotica*, Springer — Verlag, Berlin, Germany.

Blanke, M., Kinnaert, M., Lunze, J. & Staroswiecki, M. (2006), *Diagnosis and Fault-Tolerant Control*, 2 edn, Springer-Verlag Berlin Heidelberg, Berlin, Germany.

Blanke, M., Staroswiecki, M. & Wu, N. E. (2001), 'Concepts and methods in fault-tolerant control', *Proceedings of the 2001 American Control Conference, Vols 1-6* pp. 2606–2620.

Bolf, N., Ivandic, M. & Galinec, G. (2008), 'Soft sensors for crude distillation unit product properties estimation and control', *2008 Mediterranean Conference on Control Automation, Vols 1-4* pp. 790–795.

Bristol, E. (1966), 'On a new measure of interaction for multivariable process control', *IEEE Trans. Auto. Control* **AC-11**, 133 – 134.

Bristol, E. (1979), 'Recent results on interactions in multivariable process control', *Presented at 71st Annual AIChE Meeting.* Houston, TX. .

Bruccoleri, M., Amico, M. & Perrone, G. (2003), 'Distributed intelligent control of exceptions in reconfigurable manufacturing systems', *International Journal of Production Research* **41**, 1393–1412.

Buffington, J. M. & Enns, D. F. (1996), 'Lyapunov stability analysis of daisy chain control allocation', *Journal of Guidance Control and Dynamics* **19**(6), 1226–1230.

Bureau of Labor Statistics (1998), Occupational Injuries and Illnesses in the United States by Industry, Report, Washington, DC: Government Printing Office.

Camponogara, E., Jia, D., Krogh, B. H. & Talukdar, S. (2002), 'Distributed model predictive control', *IEEE Control Systems Magazine* **22**(1), 44–52.

Chandler, P. R. (1984), 'Self-repairing flight control system reliability and maintainability program – executive overview', *Proceedings of the IEEE national aerospace and electronics conference* pp. 586–590.

Chen, J. (1995), Robust Residual Generation for Model-Based Fault Diagnosis of Dynamic Systems, PhD thesis, University of York, UK.

Chen, J. & Patton, R. J. (1999), *Robust Model Based Fault Diagnosis for Dynamic Systems*, Asian Studies in Computer and Information Science, Kluwer Academic Publishers, Boston, USA.

Chilin, D., Liu, J., Chen, X. & Christofides, P. D. (2012a), 'Fault detection and isolation and fault tolerant control of a catalytic alkylation of benzene process', *Chemical Engineering Science* **78**, 155–166.

Chilin, D., Liu, J., Davis, J. F. & Christofides, P. D. (2012b), 'Data-based monitoring and reconfiguration of a distributed model predictive control system', *International Journal of Robust and Nonlinear Control* **22**(1), 68–88.

Chilin, D., Liu, J. F., de la Pena, D. M., Christofides, P. D. & Davis, J. F. (2010b), 'Monitoring and handling of actuator faults in a distributed model predictive control system', *2010 American Control Conference* pp. 2847–2854.

Chilin, D., Liu, J., Munoz de la Pena, D., Christofides, P. D. & davis, J. F. (2010a), 'Detection, isolation and handling of actuator faults in distributed model predictive control systems', *Journal of Process Control* **20**, 1059–1075.

Chow, E. Y. & Willsky, A. S. (1980), 'Issues in the development of a general algorithm for reliable failure detection', *Proc. of the 19th Conf. on Decision & Control.* Albuquerque, NM. .

Christofides, P. D., Scattolini, R., de la Pena, D. M. & Liu, J. F. (2013), 'Distributed model predictive control: A tutorial review and future research directions', *Computers and Chemical Engineering* **51**, 21–41.

Desai, M. & Ray, A. (1984), 'A fault detection and isolation methodology theory and application', *Proceeding of 1984 American Control Conference.* pp. 262–270.

Deshpande, A. P., Zamad, U. & Patwardhan, S. C. (2009), 'Online sensor/actuator failure isolation and reconfigurable control using the generalized likelihood ratio method', *Industrial and Engineering Chemistry Research* **48**, 1522–1535.

Diao, Y. & Passino, K. M. (2002), 'Intelligent fault-tolerant control using adaptive and learning methods', *Control Engineering Practice* **10**(8), 801–817.

Dong, D. & McAvoy, T. J. (1996), 'Nonlinear principal component analysisbased on principal curves and neural networks', *Computers and Chemical Engineering* **20**(1), 65–78.

Dvorak, D. L. (1992), Monitoring and Diagnosis of Continuous Dynamic Systems Using Semi-Quantitative Simulation, PhD thesis, The University of Texas at Austin, Austin, Texas 78712, USA.

Edward, P. G., Meadows, E. S., Wang, C. & Doyle, F. J. (2000), 'Model based control of a four-tank system', *Computers and Chemical Engineering* **24**, 1503–1509.

Eterno, J. S., Weiss, J. L., Looze, D. P. & Willsky, A. S. (1985), 'Design issues for fault tolerant-restructurable aircraft control', *In Proceedings of the 24th IEEE Conference on Decision and Control.* pp. 900–905.

Frank, P. M. (1990), 'Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results', *Automatica* **26**(3), 459–474.

Frank, P. M. (1992), 'Enhancement of robustness in observer-based fault-detection', *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (Safeprocess 91)* **1992**, 99–111.

Frank, P. M. (1994), 'Enhancement of robustness in observer-based fault-detection', *International Journal of Control* **59**(4), 955–981.

Frank, P. M. (1996), 'Analytical and qualitative model-based fault diagnosis a survey and some new results', *European Journal of Control* **2**(1), 6–28.

Garcia, A. E. & Frank, P. M. (1997), 'Deterministic nonlinear observer-based approaches to fault diagnosis: A survey', *Control Engineering Practice* **5**(5), 663–670.

Garcia, C. E., Prett, D. M. & Morari, M. (1989), 'Model predictive control - theory and practice - a survey', *Automatica* **25**(3), 335–348.

Geladi, P. & Kowalski, B. R. (1986), 'Partial least-squares regression - a tutorial', *Analytica Chimica Acta* **185**, 1–17.

Gertler, J. (1991), 'Analytical redundancy methods in failure detection and isolation', *IFA CIMA CS Symposium SAFE PROCESS '91.* Baden-Baden, Germany. pp. 9–21.

Gertler, J., Fang, X. W. & Luo, Q. (1990), *Detection and diagnosis of plant failures; the orthogonal parity equation approach*, Vol. 37, Academic Press, pp. 157–216. Edited by Leondes, C.

Gertler, J. J. & Kunwer, M. M. (1995), 'Optimal residual decoupling for robust fault-diagnosis', *International Journal of Control* **61**(2), 395–421.

Gertler, J. & McAvoy, T. J. (1998), 'Principal component analysis and parity relations - a strong duality', *(Safeprocess'97): Fault Detection, Supervision and Safety for Technical Processes 1997, Vols 1-3* pp. 833–838.

Gertler, J. & Singer, D. (1990), 'A new structural framework for parity equation-based failure-detection and isolation', *Automatica* **26**(2), 381–388.

Gomm, J. B., Weerasinghe, M. & Williams, D. (2000), 'Diagnosis of process faults with neural networks and principal component analysis', *Proceedings of the Institution of Mechanical Engineers Part E-Journal of Process Mechanical Engineering* **214**(E2), 131–143.

Himmelblau, D. M. (1978), *Fault Detection and Diagnosis in Chemical and Petrochemical Processes*, Elsevier Scientific Publishing Company, New York.

Hoskins, J. C., Kaliyur, K. M. & Himmelblau, D. M. (1991), 'Fault-diagnosis in complex chemical-plants using artificial neural networks', *AIChE Journal* **37**(1), 137–141.

Hotelling, H. (1947), *Multivariate Quality Control Illustrated by Air Testing of Sample Bombsights*, McGraw Hill, New York, pp. 111–184. Edited by Eisenhart, C. and Hastay, M.W. and Wallis, W.A.

Hsieh, C. S. (2002), 'Performance gain margins of the two-stage lq reliable control', *Automatica* **38**(11), 1985–1990.

Isermann, R. (1984), 'Process fault detection based on modeling and estimation methods — A survey', *Automatica* **20**(4), 387–404.

Isermann, R. (1987), Experiences with process fault detection via parameter estimation, *in* S. G. Tzafestas, M. G. Singh & G. Schmidt, eds, 'System Fault Diagnostics, Reliability Related Knowledge-based Approaches', Vol. 1, D. Reidel Publishing Company, Dordrecht, Holland, pp. 3–33.

Isermann, R. (1993), 'Fault diagnosis of machines via parameter estimation and knowledge processingtutorial paper', *Automatica* **29**(4), 815–835.

Isermann, R. (1997), 'Supervision, fault-detection and fault-diagnosis methods an introduction', *Control Engineering Practice* **5**(5), 639–652.

Isermann, R. (2005), 'Model-based fault-detection and diagnosis status and applications', *Annual Reviews in Control* **29**(1), 71–85.

Isermann, R. (2006), *Fault-diagnosis systems: An introduction from fault detection to fault tolerance*, Springer-Verlag Berlin Heidelberg, Berlin, Germany.

Isermann, R. & Ball, P. (1997), 'Trends in the application of model-based fault detection and diagnosis of technical processes', *Control Engineering Practice* **5**(5), 709–719.

Isermann, R. & Freyermuth, B. (1991*a*), 'Process fault-diagnosis based on process model knowledge .1. principles for fault-diagnosis with parameter-estimation', *Journal of Dynamic Systems Measurement and Control-Transactions of the ASME* **113**(4), 620–626.

Isermann, R. & Freyermuth, B. (1991*b*), 'Process fault-diagnosis based on process model knowledge .2. case-study experiments', *Journal of Dynamic Systems Measurement and Control-Transactions of the ASME* **113**(4), 627–633.

Isermann, R., Schwarz, R. & Stolzl, S. (2002), 'Fault-tolerant drive-by-wire systems', *IEEE Control Systems Magazine* **22**(5), 64–81.

Jackson, B. B. (1993), Union carbide: Disaster at bhopal, Report, Union Carbide Corporation.

Jacobs, R. A., Jordan, M. I., Nowlan, A. S. J. & Hinton, A. G. E. (1991), 'Adaptive mixtures of local experts', *Neural Computing and Applications* **3**(1), 79–87.

Jordan, M. I. & Jacobs, R. A. (1994), 'Hierarchical mixtures of experts and the em algorithm', *Neural Computation* **6**(2), 181–214.

Kano, M., Miyazaki, K., Hasebe, S. & Hashimoto, I. (2000), 'Inferential control system of distillation compositions using dynamic partial least squares regression', *Journal of Process Control* **10**(2-3), 157–166.

Kettunen, M. (2010), Data-based fault-tolerant model predictive controller: An application to a complex dearomatizarion process, PhD thesis, Aalto University.

Kourti, T. & MacGregor, J. F. (1996), 'Multivariate SPC methods for process and product monitoring', *Journal of Quality Technology* **28**(4), 409–428.

Ku, W. F., Storer, R. H. & Georgakis, C. (1995), 'Disturbance detection and isolation by dynamic principal component analysis', *Chemometrics and Intelligent Laboratory Systems* **30**(1), 179–196.

Lao, L., Ellis, M. & Christofides, P. D. (2013), 'Proactive fault-tolerant model predictive control', *AIChE Journal* **59**(8), 2810–2820.

Laser, M. (2000), 'Recent safety and environmental legislation', *Transactions of the Institution of Chemical Engineers* **78**(B), 419–422.

Lawal, S. A. & Zhang, J. (2015), 'Actuator fault monitoring and fault tolerant control in distillation columns', *2015 21st International Conference on Automation and Computing (ICAC)* pp. 329–334.

Lawal, S. A. & Zhang, J. (2016*a*), 'Fault monitoring and fault tolerant control in distillation columns', *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)* pp. 865–870.

Lawal, S. A. & Zhang, J. (2016*b*), Sensor fault detection and fault tolerant control of a crude distillation unit, *in* Z. Kravanja & M. Bogataj, eds, '26th European Symposium on Computer Aided Process Engineering', Vol. 38B, Elsevier Science Bv, Amsterdam, pp. 2091–2096.

Lawal, S. A. & Zhang, J. (2017*a*), Actuator and sensor fault tolerant control of a crude distillation unit, *in* A. Espua, M. Graells & L. Puigjaner, eds, 'Computer Aided Process Engineering', Elsevier, pp. 1705–1710.

Lawal, S. A. & Zhang, J. (2017*b*), 'Actuator fault monitoring and fault tolerant control in distillation columns', *International Journal of Automation and Computing* **14**(1), 80–92.

Lees, F. P. (1996), *Loss prevention in the process industries hazard identification, assessment, and control*, 2nd ed.. edn, Boston : Butterworth-Heinemann, Boston.

Li, W. H., Yue, H. H., Valle-Cervantes, S. & Qin, S. J. (2000), 'Recursive pca for adaptive process monitoring', *Journal of Process Control* **10**(5), 471–486.

Liang, Y. W., Liaw, D. C. & Lee, T. C. (2000), 'Reliable control of nonlinear systems', *IEEE Transactions on Automatic Control* **45**(4), 706–710.

Lou, X. C., Willsky, A. S. & Verghese, G. C. (1986), 'Optimally robust redundancy relations for failure-detection in uncertain systems', *Automatica* **22**(3), 333–344.

MacGregor, J. & Cinar, A. (2012), 'Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods', *Computers and Chemical Engineering* **47**, 111–120.

MacGregor, J. F., Jacckle, C., Kiparissides, C. & Koutondi, M. (1994), 'Process monitoring and diagnosis by multiblock pls methods.', *American Institute of Chemical Engineers Journal* **40**(5), 826–838.

Mahmoud, M., Jiang, J. & Zhang, Y. M. (2003), 'Stabilization of active fault tolerant control systems with imperfect fault detection and diagnosis', *Stochastic Analysis and Applications* **21**(3), 673–701.

Manuja, S., Narasimhan, S. & Patwardhan, S. (2008), 'Fault diagnosis and fault tolerant control using reduced order models', *Canadian Journal of Chemical Engineering* **86**(4), 791–803.

Mcavoy, T. J. (1983), 'Some results on dynamic interaction analysis of complex control-systems', *Industrial and Engineering Chemistry Process Design and Development* **22**(1), 42–49.

McGraw-Hill Economics (1985), Survey of investment in employee safety and health, Report, New York: McGraw-Hill Publishing Co.

Mehrabi, M. G., Ulsoy, A. G., Koren, Y. & Heytler, P. (2002), 'Trends and perspectives in flexible and reconfigurable manufacturing systems.', *Journal of Intelligent Manufacturing* **13**(2), 23–44.

Mercangoz, M. & Doyle III, F. J. (2007), 'Distributed model predictive control of an experimental four-tank system', *Journal of Process Control* **17**(3), 297–308.

Mhaskar, P. (2006), 'Robust model predictive control design for fault tolerant control of process systems', *Industrial and Engineering Chemistry Research* **45**(25), 8565–8574.

Mirzaee, A. & Salahshoor, K. (2012), 'Fault diagnosis and accommodation of nonlinear systems based on multiple-model adaptive unscented kalman filter and switched mpc and H-infinity loop-shaping controller', *Journal of Process Control* **22**(3), 626–634.

Mutambara, A. (1998), *Decentralized Estimation and Control for Multisensor Systems*, CRC Press, Boca Raton.

National Safety Council (1999), Injury facts 1999 Edition, Press release, National Safety Council.

Nimmo, I. (1995), 'Adequately address abnormal situation operations', *Chemical Engineering Progress* **91**(9), 36–45.

Nomikos, P. & Macgregor, J. F. (1994), 'Monitoring batch processes using multiway principal component analysis', *AIChE Journal* **40**(8), 1361–1375.

Noura, H., Theilliol, D., Ponsant, J. C. & Chamseddine, A. (2009), *Fault Tolerant Control Systems: Design and Practical Applications*, Vol. XXI, 233, Springer-Verlag London, London.

Noura, H., Theilliol, D. & Sauter, D. (2000), 'Actuator fault-tolerant control design: demonstration on a three-tank-system', *International Journal of Systems Science* **31**(9), 1143–1155.

Patton, R. (1997*a*), 'Fault-tolerant control system: The 1997 situation', *Proceedings of SAFEPROCESS.* Kingston Upon Hull, UK. pp. 1033–1054.

Patton, R. J. (1991), 'Fault detection and diagnosis in aerospace systems using analytical redundancy', *Computing and Control Engineering Journal* **2**(3), 127–136.

Patton, R. J. (1993), 'Robustness issues in fault-tolerant control', *Proceedings of the IEE Colloquium on Fault Diagnosis and Control System Reconfiguration.* pp. 1–25.

Patton, R. J. (1997*b*), 'Robustness in model-based fault diagnosis: The 1995 situation', *Annual Reviews in Control* **21**, 103–123.

Patton, R. J. & Chen, J. (1992*a*), 'Optimal selection of unknown input distribution matrix in the design of robust observers for fault-diagnosis', *Fault Detection, Supervision and Safety for Technical Processes (Safeprocess 91)* **1992**, 229–234.

Patton, R. J. & Chen, J. (1992*b*), 'A review of parity space approaches to fault-diagnosis', *Fault Detection, Supervision and Safety for Technical Processes (Safeprocess 91)* **1992**, 65–81.

Patton, R. J. & Chen, J. (1996), *Robust fault detection and isolation (FDI) systems*, Vol. Volume 74, Academic Press, pp. 171–224. Edited by Cornelius, T. Leondes.

Patton, R. J. & Chen, J. (1997), 'Observer-based fault detection and isolation: Robustness and applications', *Control Engineering Practice* **5**(5), 671–682.

Patton, R. J., Frank, P. M. & Clark, R. N. (1989), *Fault Diagnosis in Dynamic Systems, Theory and Application*, Control Engineering Series, Prentice Hall, New York.

Pearson, K. (1901), 'On lines and planes of closest fit to systems of points in space', *Philosophical Magazine* **2**, 559–572.

Prett, D. M. & Morari., M. (1987), *The Shell Process Control Workshop*, Butterworths, London.

Qin, S. J. (1998), 'Recursive pls algorithm for adaptive data monitoring', *Computers and Chemical Engineering* **22**(4/5), 503–514.

Qin, S. J. & Badgwell, T. A. (2003), 'A survey of industrial model predictive control technology', *Control Engineering Practice* **11**(7), 733–764.

Qin, S. J. & Mcavoy, T. J. (1992), 'Nonlinear pls modeling using neural networks', *Computers and Chemical Engineering* **16**(4), 379–391.

Raich, A. & Cinar, A. (1996), 'Statistical process monitoring and disturbance diagnosis in multivariable continuous processes', *AIChE Journal* **42**(4), 995–1009.

Rauch, H. E. (1994), 'Intelligent fault diagnosis and control reconfiguration', *IEEE Control Systems* **14**(3), 6–12.

Rauch, H. E. (1995), 'Autonomous control reconfiguration', *IEEE Control Systems* **15**(6), 37–48.

Rawlings, J. B. & Stewart, B. T. (2008), 'Coordinating multiple optimization-based controllers: New opportunities and challenges', *Journal of Process Control* **18**(9), 839–845.

Sangha, M., Yu, D. & Gomm, J. B. (2008), 'Robustness assessment and Adaptive FDi for car engine', *International Journal of Automation and Computing* **5**(2), 109–118.

Shinskey, F. G. (1988), *Process Control System*, McGraw-Hill, New York.

Siljak, D. D. (1980), 'Reliable control using multiple control-systems', *International Journal of Control* **31**(2), 303–329.

Steinberg, M. (2005), 'Historical overview of research in reconfigurable flight control', *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **219**(4), 263–275.

Stengel, R. F. (1991), 'Intelligent failure-tolerant control', *IEEE Control Systems* **11**(4), 14–23.

Tao, J., Zhu, Y. & Fan, Q. (2014), 'Improved state space model predictive control design for linear systems with partial actuator failure', *Industrial and Engineering Chemistry Research* **53**, 3578 – 3586.

Tham, M. T., Vagi, F., Morris, A. J. & Wood, R. K. (1991*a*), 'Multivariable and multirate self-tuning control - a distillation column case-study', *IEE Proceedings-D Control Theory and Applications* **138**(1), 9–24.

Tham, M. T., Vagi, F., Morris, A. J. & Wood, R. K. (1991*b*), 'Online multivariable adaptive-control of a binary distillation column', *Canadian Journal of Chemical Engineering* **69**(4), 997–1009.

Tzafestas, S. G. (1989), System fault diagnosis using the knowledge-based methodology, *in* R. J. Patton, P. M. Frank & R. N. Clark, eds, 'Fault Diagnosis in Dynamic Systems, Theory and Application', Prentice Hall, chapter 15, pp. 509–572.

Tzafestas, S. G. & Watanabe, K. (1990), 'Modern approaches to system/sensor fault detection and diagnosis', *Journal A* **31**(4), 42–57.

Ungar, L. H., Powell, B. A. & Kamens, S. N. (1990), 'Adaptive networks for fault-diagnosis and process-control', *Computers and Chemical Engineering* **14**(4-5), 561–572.

van Schrick, D. (1993), Zustandsschtzer' siemen zur Fehlerkenung, deren Zuverhissigkeit und Anwendung auf den sp gefiuihrten Omnibus, PhD thesis, University of Wuppertal.

Vander Velde, W. E. (1984), 'Control system reconfiguration', *Proceedings of 1984 American control conference.* San Diego, CA. **3**, 1741–1745.

Veillette, R. J. (1995), 'Reliable linear-quadratic state-feedback control', *Automatica* **31**(1), 137–143.

Veillette, R. J., Medanic, J. V. & Perkins, W. R. (1992), 'Design of reliable control-systems', *IEEE Transactions on Automatic Control* **37**(3), 290–304.

Venkatasubramanian, V. & Chan, K. (1989), 'A neural network methodology for process fault-diagnosis', *AIChE Journal* **35**(12), 1993–2002.

Venkatasubramanian, V., Rengaswamy, R. & Kavuri, S. N. (2003a), 'A review of process fault detection and diagnosis Part II: Quantitative model and search strategies', *Computers and Chemical Engineering* **27**(3), 313–326.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N. & Yin, K. (2003b), 'A review of process fault detection and diagnosis Part III: Process history based methods', *Computers and Chemical Engineering* **27**(3), 327–346.

Venkatasubramanian, V., Vaidyanathan, R. & Yamamoto, Y. (1990), 'Process fault-detection and diagnosis using neural networks .1. steady-state processes', *Computers and Chemical Engineering* **14**(7), 699–712.

Venkatsubramanian, V., Rengaswamy, R., Yin, K. & Kavuri, S. N. (2003c), 'A review of process fault detection and diagnosis Part I: Quantitative model-based methods', *Computers and Chemical Engineering* **27**(3), 293–311.

Viswanadham, N. & Srichander, R. (1987), 'Fault detection using unknown input observers', *Control - Theory and Advanced Technology* **3**(2), 91–101.

Vlachos, C., Williams, D. & Gomm, J. B. (2002), 'Solution to the shell standard control problem using genetically tuned pid controllers', *Control Engineering Practice* **10**(2), 151–163.

Wang, S. H., Davison, E. J. & Dorato, P. (1975), 'Observing states of systems with unmeasurable disturbances', *IEEE Transactions on Automatic Control* **20**(5), 716–717.

Wang, Y., Yang, Y., Zhou, D. & Gao, F. (2007), 'Active fault-tolerant control of nonlinear batch processes with sensor faults', *Industrial and Engineering Chemistry Research* **46**(26), 9158–9169.

Watanabe, K. & Himmelblau, D. M. (1982), 'Instrument fault detection in systems with uncertainties', *International Journal of Systems Science* **13**(2), 137–158.

Watanabe, K., Matsura, I., Abe, M., Kubota, M. & Himmelblau, D. M. (1989), 'Incipient fault diagnosis of chemical processes via artificial neural networks', *American Institute of Chemical Engineers Journal* **35**(11), 1803–1812.

Willsky, A. S. (1976), 'A survey of design methods for failure detection in dynamic systems', *Automatica* **12**(6), 601–611.

Witcher, M. F. & Mcavoy, T. J. (1977), 'Interacting control-systems - steady-state and dynamic measurement of interaction', *ISA Transactions* **16**(3), 35–41.

Wold, H. (1982), *Soft modeling, the basic design and some extensions*, System under Indirect Observations, North Holland, Amsterdam. Edited by Joreskog, K. and Wold, H.

Wold, S., Albano, C., Dunn III, W. J., Edland, U., Esbensen, K., Geladi, P., Hellberg, S., Johansson, E., Lindberg, W. & Sjostrom, M. (1984a), Multivariate data analysis in chemistry, *in* B. Kowalski, ed., 'Chemometrics; mathematics and statistics in chemistry', Reidel Publishing Co, Dordrecht.

Wold, S., Esbensen, K. & Geladi, P. (1987), 'Principal component analysis', *Chemometrics and Intelligent Laboratory Systems* **2**(1-3), 37–52.

Wold, S., Ruhe, A., Wold, H. & Dunn, W. J. (1984b), 'The collinearity problem in linear-regression - the partial least-squares (PLS) approach to generalized inverses', *SIAM Journal on Scientific and Statistical Computing* **5**(3), 735–743.

Yang, G. H., Wang, J. L. & Soh, Y. C. (2000), 'Reliable lqg control with sensor failures', *IEE Proceedings-Control Theory and Applications* **147**(4), 433–439.

Yang, G. H., Wang, J. L. & Soh, Y. C. (2001), 'Reliable h(infinity) controller design for linear systems', *Automatica* **37**(5), 717–725.

Yang, G. H. & Zhang, S. Y. (1995), 'Design of reliable control systems by using multiple similar controllers', *Proceedings of the 34th IEEE Conference on Decision and Control, Vols 1-4* pp. 969–972.

Yang, G. H., Zhang, S. Y., Lam, J. & Wang, J. L. (1998), 'Reliable control using redundant controllers', *IEEE Transactions on Automatic Control* **43**(11), 1588–1593.

Yu, D., Hamad, A., Gomm, J. & Sangha, M. (2014), 'Dynamic fault detection and isolation for automotive engine air path by independent neural network model', *International Journal of Engine Research* **15**(1), 87–100.

Yu, D., Williams, D., Shield, D. & Gomm, J. B. (1995), 'A parity space method of fault detection for bilinear systems', *Proceedings of the 1995 American Control Conference* **1**(6), 1132–1133.

Yu, X. & Jiang, J. (2015), 'A survey of fault-tolerant controller based on safety related issues', *Annual Review in Control* **39**, 46–57.

Yu, X., Lu, W., Huang, D. & Jin, Y. (2008), 'Multi-objective optimization of industrial crude distillation unit based on HYSYS and NSGA-II', *Huagong Xuebao/CIESC Journal* **59**(7), 1646–1649.

Zhang, J. (2006*a*), 'Improved on-line process fault diagnosis through information fusion in multiple neural networks', *Computers and Chemical Engineering* **30**(3), 558–571.

Zhang, J. (2006*b*), 'Offset-free inferential feedback control of distillation compositions based on pcr and pls models', *Chemical Engineering and Technology* **29**(5), 560–566.

Zhang, J. & Agustriyanto, R. (2003), 'Multivariate inferential feed-forward control', *Industrial and Engineering Chemistry Research* **42**, 4186–4197.

Zhang, J., Martin, E. B. & Morris, A. J. (1996), 'Fault detection and diagnosis using multivariate statistical techniques', *Chemical Engineering Research and Design* **74**(A1), 89–96.

Zhang, J., Martin, E. B. & Morris, A. J. (1997), 'Process monitoring using non-linear statistical techniques', *Chemical Engineering Journal* **67**(3), 181–189.

Zhang, J. & Roberts, P. (1991), 'Process fault diagnosis with diagnostic rules based on structural decomposition', *Journal of Process Control* **1**(5), 259–269.

Zhang, R. D., Lu, J. Y., Qu, H. Y. & Gao, F. R. (2014), 'State space model predictive fault-tolerant control for batch processes with partial actuator failure', *Journal of Process Control* **24**(5), 613–620.

Zhang, Y. & Jiang, J. (2008), 'Bibliographical review on reconfigurable fault-tolerant control systems', *Annual Reviews in Control* **32**(2), 229–252.

Zhang, Y. M. & Jiang, J. (2003), 'Fault tolerant control system design with explicit consideration of performance degradation', *IEEE Transactions on Aerospace and Electronic Systems* **39**(3), 838–848.

Zhao, Q. & Jiang, J. (1998), 'Reliable state feedback control system design against actuator failures', *Automatica* **34**(10), 1267–1272.

Zhou, C., Liu, Q. Y., Huang, D. X. & Zhang, J. (2012), 'Inferential estimation of kerosene dry point in refineries with varying crudes', *Journal of Process Control* **22**(6), 1122–1126.

Appendix A

Crude Distillation Unit Variable List

The table below presents all the process variables monitored in the crude distillation unit.

Variables	Description	Unit
1	CO Feed - Mass Flow	[kg/h]
2	CO Feed - Temperature	$[^oC]$
3	TS-1 - Stage 2 Temperature	$[^oC]$
4	TS-1 - Stage 3 Temperature	$[^oC]$
5	TS-1 - Stage 4 Temperature	$[^oC]$
6	TS-1 - Stage 5 Temperature	$[^oC]$
7	TS-1 - Stage 6 Temperature	$[^oC]$
8	TS-1 - Stage 7 Temperature	$[^oC]$
9	TS-1 - Stage 8 Temperature	$[^oC]$
10	TS-1 - Stage 9 Temperature	$[^oC]$
11	TS-1 - Stage 10 Temperature	$[^oC]$
12	TS-1 - Stage 11 Temperature	$[^oC]$
13	TS-1 - Stage 12 Temperature	$[^oC]$
14	TS-1 - Stage 13 Temperature	$[^oC]$
15	TS-1 - Stage 14 Temperature	$[^oC]$
16	TS-1 - Stage 15 Temperature	$[^oC]$
17	TS-1 - Stage 16 Temperature	$[^oC]$
18	TS-1 - Stage 17 Temperature	$[^oC]$
19	TS-1 - Stage 18 Temperature	$[^oC]$
20	TS-1 - Stage 19 Temperature	$[^oC]$
21	TS-1 - Stage 20 Temperature	$[^oC]$
22	TS-1 - Stage 21 Temperature	$[^oC]$
23	TS-1 - Stage 22 Temperature	$[^oC]$

Variables	Description	Unit
24	TS-1 - Stage 23 Temperature	$[^oC]$
25	TS-1 - Stage 24 Temperature	$[^oC]$
26	TS-1 - Stage 25 Temperature	$[^oC]$
27	TS-1 - Stage 26 Temperature	$[^oC]$
28	TS-1 - Stage 27 Temperature	$[^oC]$
29	TS-1 - Stage 28 Temperature	$[^oC]$
30	TS-1 - Stage 29 Temperature	$[^oC]$
31	PreFlash - Bottom Stage Pressure	[kPa]
32	PreFlash - Bottom Stage Temperature	$[^oC]$
33	PreFlash - Top Stage Pressure	[kPa]
34	PreFlash - Top Stage Temperature	$[^oC]$
35	Naptha - Mass Flow	[kg/h]
36	Naptha - Temperature	$[^oC]$
37	Kero Product - Temperature	$[^oC]$
38	Kero Product - Mass Flow	[kg/h]
39	Diesel - Mass Flow	[kg/h]
40	Diesel - Temperature	$[^oC]$
41	AGO - Temperature	$[^oC]$
42	AGO - Mass Flow	[kg/h]
43	CDU Residue - Mass Flow	[kg/h]
44	CDU Residue - Temperature	$[^oC]$
45	SS-1 Return - Mass Flow	[kg/h]
46	SS-2 Return - Mass Flow	[kg/h]
47	SS-3 return - Mass Flow	[kg/h]
48	SS-3 return - Temperature	$[^oC]$
49	SS-2 Return - Temperature	$[^oC]$
50	SS-1 Return - Temperature	$[^oC]$
51	SS-2 Steam - Mass Flow	[kg/h]
52	SS-3 Steam - Mass Flow	[kg/h]
53	CDU Btm Steam-2 - Mass Flow	[kg/h]
54	Reflux - Mass Flow	[kg/h]
55	Btm BoilUp - Mass Flow	[kg/h]
56	PA1 S - Mass Flow	[kg/h]
57	PA1 S - Temperature	$[^oC]$
58	PA2 S - Mass Flow	[kg/h]
59	PA2 S - Temperature	$[^oC]$
60	PA3 S - Mass Flow	[kg/h]

Variables	Description	Unit
61	PA3 S - Temperature	$[^oC]$
62	CFZ TS - Temperature	$[^oC]$
63	43 S - Mass Flow	[kg/h]
64	Vap Out1 - Temperature	$[^oC]$
65	Vap Out1 - Mass Flow	[kg/h]
66	ProdRatio - B2: Naphtha Feed Ratio	
67	ProdRatio - B3: Kero Feed Ratio	
68	ProdRatio - B4: Diesel Feed Ratio	
69	ProdRatio - B5: AGO Feed Ratio	
70	ProdRatio - B6: CDU Residue Feed Ratio	
71	Heat flow to the furnace	[kJ/h]

Appendix B

Sample MATLAB Code for CDU Simulation

An example of the MATLAB code used to simulate the dynamic interactive crude distillation unit is shown below.

```
% NROMAL SYSTEM SIMULATION WITH ALL DISTURBANCES AND NO FAULT
% HYSYS - MATLAB INTERFACE

    % LINK HYSYS TO MATLAB
hys = actxserver('HYSYS.Application');

    % ACCESS ACTIVE HYSYS FLOWSHEET THROUGH MATLAB
hysActive = hys.ActiveDocument;
hysFlowsheet = hysActive.Flowsheet;

    % CONNECT TO SUB-FLOWSHEET COL1 (ATM CDU)
hysSubFlowsheets = hysFlowsheet.Flowsheets;
hysCOL1 = hysSubFlowsheets.Item('COL1');

    % ACCESS HYSYS SOLVER IN MATLAB
hysSolver = hysActive.Solver;

    % LINK TO MATERIAL STREAMS PA1S & SS2-RETURN IN COL1
hPA1S = hysCOL1.Streams.Item('PA1S');
hSS2R = hysCOL1.Streams.Item('SS-2 Return');
%hSS2R.MassFlow.GetValue*3600;
```

```
    % ACCESS HYSYS SPREADSHEETS IN MATLAB
h_MV = hysFlowsheet.Operations.Item('MVs');
hysSS = hysFlowsheet.Operations.Item('DistVar');
h_X = hysFlowsheet.Operations.Item('CDUData');
h_PQV = hysFlowsheet.Operations.Item('Process QVs');

    % INITIAL SIMULATION WITH NOMINAL OPERATING CONDITIONS
hysSS.Cell('A1').CellValue = hysSS.Cell('C1').CellValue;
hysSS.Cell('A2').CellValue = hysSS.Cell('C2').CellValue;
hysSS.Cell('A3').CellValue = hysSS.Cell('C3').CellValue;
hysSS.Cell('A4').CellValue = hysSS.Cell('C4').CellValue;
hysSS.Cell('A5').CellValue = hysSS.Cell('C5').CellValue;
hysSS.Cell('A6').CellValue = hysSS.Cell('C6').CellValue;

    % SPECIFY SOME EMPTY ARRAYS
N = 3600; PQV = [];
X = [];

    % Initialising the simulation with nominal values
runTime=0.5;
allowToRun=1;

    for runSim=1:1200
if (allowToRun == 1)
% Initialising the simulation with nominal values
if (runSim>=1 && runSim<101)
hysSolver.Integrator.RunFor(runTime,'minutes'); % 1-50 mins
end
% FIRST DISTURBANCE INTRODUCED: CRUDE TEMP (15.6oC)
% Make +/- 20 % changes to Crude Temperature.
if (runSim>= 101 && runSim<151)
hysSS.Cell('A1').CellValue = hysSS.Cell('B1').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 51-75 mins
end
if (runSim>=151 && runSim<201)
hysSS.Cell('A1').CellValue = hysSS.Cell('C1').CellValue;
```

```
hysSolver.Integrator.RunFor(runTime,'minutes'); % 76-100 mins
end
if (runSim>=201 && runSim<251)
hysSS.Cell('A1').CellValue = hysSS.Cell('D1').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 101-125 mins
end
if (runSim>=251 && runSim<301)
hysSS.Cell('A1').CellValue = hysSS.Cell('C1').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 126-150 mins
end


    % SECOND DISTURBANCE INTRODUCED (2.352E+005 KG/H)
% Make +/- 20 % changes to Crude Flow Rate By Changing FIC-100 SP.
if (runSim>=301 && runSim<351)
hysSS.Cell('A2').CellValue = hysSS.Cell('B2').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 151-175 mins
end
if (runSim>=351 && runSim<401)
hysSS.Cell('A2').CellValue = hysSS.Cell('C2').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 176-200 mins
end
if (runSim>=401 && runSim<451)
hysSS.Cell('A2').CellValue = hysSS.Cell('D2').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 201-225 mins
end
if (runSim>=451 && runSim<501)
hysSS.Cell('A2').CellValue = hysSS.Cell('C2').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 226-250 mins
end


    % THIRD DISTURBANCE INTRODUCED (CO Feed Comp)
% Make +/- 20 % changes to Crude Feed Composition
if (runSim>=501 && runSim<551)
```

```
hysSS.Cell('A2').CellValue = hysSS.Cell('B2').CellValue;
hysSS.Cell('A3').CellValue = hysSS.Cell('B3').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 251-275 mins
end
if (runSim>=551 && runSim<601)
hysSS.Cell('A2').CellValue = hysSS.Cell('C2').CellValue;
hysSS.Cell('A3').CellValue = hysSS.Cell('C3').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 276-300 mins
end
if (runSim>=601 && runSim<651)
hysSS.Cell('A2').CellValue = hysSS.Cell('B2').CellValue;
hysSS.Cell('A4').CellValue = hysSS.Cell('B4').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 301-325 mins
end
if (runSim>=651 && runSim<701)
hysSS.Cell('A2').CellValue = hysSS.Cell('C2').CellValue;
hysSS.Cell('A4').CellValue = hysSS.Cell('C4').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 326-350 mins
end



    % FOURTH DISTURBANCE INTRODUCED (50 %)
% Make +/- 20 % changes to CFZ Feed Rate By Changing LIC-104 SP.
if (runSim>=701 && runSim<751)
hysSS.Cell('A5').CellValue = hysSS.Cell('B5').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 351-375 mins
end
if (runSim>=751 && runSim<801)
hysSS.Cell('A5').CellValue = hysSS.Cell('C5').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 376-400 mins
end
if (runSim>=801 && runSim<851)
hysSS.Cell('A5').CellValue = hysSS.Cell('D5').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 401-425 mins
```

```
end
if (runSim>=851 && runSim<901)
hysSS.Cell('A5').CellValue = hysSS.Cell('C5').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 426-450 mins
end
```

% FIFTH DISTURBANCE INTRODUCED (360^{oC} %) % Make +/- 20 % changes to CFZ Temp.

```
if (runSim>=901 && runSim<951)
hysSS.Cell('A6').CellValue = hysSS.Cell('B6').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 451-475 mins
end
if (runSim>=951 && runSim<1001)
hysSS.Cell('A6').CellValue = hysSS.Cell('C6').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 476-500 mins
end
if (runSim>=1001 && runSim<1051)
hysSS.Cell('A6').CellValue = hysSS.Cell('D6').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 501-525 mins
end
if (runSim>=1051 && runSim<1201)
hysSS.Cell('A6').CellValue = hysSS.Cell('C6').CellValue;
hysSolver.Integrator.RunFor(runTime,'minutes'); % 526-600 mins
end
```

% SOME CELLS ARE MULTIPLIED BY 3600 TO CONVERT THEIR UNITS TO KG/H

```
X0 = [h_X.Cell('A1').CellValue*N h_X.Cell('A2').CellValue
h_X.Cell('A3').CellValue h_X.Cell('A4').CellValue h_X.Cell('A5').CellValue
h_X.Cell('A6').CellValue h_X.Cell('A7').CellValue h_X.Cell('A8').CellValue
h_X.Cell('A9').CellValue h_X.Cell('A10').CellValue h_X.Cell('A11').CellValue
h_X.Cell('A12').CellValue h_X.Cell('A13').CellValue h_X.Cell('A14').CellValue
h_X.Cell('A15').CellValue h_X.Cell('B1').CellValue h_X.Cell('B2').CellValue
```

```
h_X.Cell('B3').CellValue h_X.Cell('B4').CellValue h_X.Cell('B5').CellValue
h_X.Cell('B6').CellValue h_X.Cell('B7').CellValue h_X.Cell('B8').CellValue
h_X.Cell('B9').CellValue h_X.Cell('B10').CellValue h_X.Cell('B11').CellValue
h_X.Cell('B12').CellValue h_X.Cell('B13').CellValue h_X.Cell('B14').CellValue
h_X.Cell('B15').CellValue h_X.Cell('C1').CellValue h_X.Cell('C2').CellValue
h_X.Cell('C3').CellValue h_X.Cell('C4').CellValue h_X.Cell('C5').CellValue*N
h_X.Cell('C6').CellValue h_X.Cell('C7').CellValue*N h_X.Cell('C8').CellValue
h_X.Cell('C9').CellValue*N h_X.Cell('C10').CellValue h_X.Cell('C11').CellValue*N
h_X.Cell('C12').CellValue h_X.Cell('C13').CellValue*N h_X.Cell('C14').CellValue
h_X.Cell('C15').CellValue*N h_X.Cell('D1').CellValue*N h_X.Cell('D2').CellValue*N
h_X.Cell('D3').CellValue h_X.Cell('D4').CellValue h_X.Cell('D5').CellValue
h_X.Cell('D6').CellValue*N h_X.Cell('D7').CellValue*N h_X.Cell('D8').CellValue*N
h_X.Cell('D9').CellValue*N h_X.Cell('D10').CellValue*N h_X.Cell('D11').CellValue*N
h_X.Cell('D12').CellValue h_X.Cell('D13').CellValue*N h_X.Cell('D14').CellValue
h_X.Cell('D15').CellValue*N h_X.Cell('E1').CellValue h_X.Cell('E2').CellValue
h_X.Cell('E3').CellValue*N h_X.Cell('E4').CellValue h_X.Cell('E5').CellValue*N
h_X.Cell('E6').CellValue h_X.Cell('E7').CellValue h_X.Cell('E8').CellValue
h_X.Cell('E9').CellValue h_X.Cell('E10').CellValue h_X.Cell('E11').CellValue*N
h_X.Cell('E12').CellValue*N h_X.Cell('E13').CellValue*N h_X.Cell('E14').CellValue*N];
X = [X; X0]; % SOME CELLS ARE MULTIPLIED BY 3600 TO CONVERT THEIR

UNITS TO KG/H
PQV_0 = [h_PQV.Cell('A1').CellValue h_PQV.Cell('A2').CellValue
h_PQV.Cell('A3').CellValue h_PQV.Cell('A4').CellValue h_PQV.Cell('A5').CellValue
h_PQV.Cell('A6').CellValue h_PQV.Cell('A7').CellValue];
    PQV = [PQV; PQV_0];
    end
end
```

Appendix C

12 × 12 Transfer Function Models of the Dynamic CDU

$$G_{12}(s) = \begin{bmatrix} G_A(s) & G_B(s) \\ G_C(s) & G_D(s) \end{bmatrix} \tag{C.1}$$

$$G_A(s) = \begin{bmatrix} \frac{0.00029e^{-2s}}{s^2+0.18s+0.095} & \frac{0.00026}{s^2+0.14s+0.098} & \frac{-0.00007e^{-14.5s}}{s+0.044} & \frac{0.0065e^{-5.57s}}{s^2+84s+1} & \frac{-0.493}{0.385s+1} & \frac{0.000003e^{-s}}{s^2+0.31s+0.278} \\ \frac{-0.12}{3.79s+1} & \frac{-0.000059}{3.51s+1} & \frac{-0.00017e^{-0.97s}}{0.069s+1} & \frac{-0.00015}{s+0.476} & \frac{0.00079}{s+0.165} & \frac{-0.00083e^{-3.35s}}{0.063s+1} \\ \frac{-0.015e^{-3s}}{s^2+0.2s+0.11} & \frac{-0.0000016e^{-12.5s}}{s+0.185} & \frac{-0.00014}{s+0.538} & \frac{-0.00019}{s+0.248} & \frac{0.00077}{s+0.133} & \frac{-0.00074e^{-2.22s}}{0.849s+1} \\ \frac{0.0029e^{-10.8s}}{0.038s+1} & \frac{0.012}{s+0.23} & \frac{0.269e^{-6.44s}}{0.091s+1} & \frac{1.002e^{-1.22s}}{0.938s+1} & \frac{-6.74e^{-1.7s}}{1.17s+1} & \frac{0.438}{0.0064s+1} \\ \frac{0.00029e^{-6.14s}}{1.75s+1} & \frac{0.00068}{s+0.233} & \frac{0.0043e^{-0.5s}}{s+0.239} & \frac{0.056}{s+1.2} & \frac{52.34}{s+43.82} & \frac{0.032e^{-5.22s}}{0.154s+1} \\ \frac{-0.023}{s^2+0.27s+0.15} & \frac{-0.000056}{3.33s+1} & \frac{-0.000092e^{-0.69s}}{0.424s+1} & \frac{-0.0000046e^{-0.5s}}{s+0.043} & \frac{0.00024}{s+0.064} & \frac{-0.00093}{0.977s+1} \end{bmatrix} \tag{C.2}$$

$$G_B(s) = \begin{bmatrix} \frac{-0.015e^{-0.5s}}{s+0.19} & \frac{1.326}{s+40.78} & \frac{-0.000034}{s^2+0.043s+0.044} & \frac{341.3}{s+50.47} & \frac{-9.76}{s+21.26} & \frac{0.00000019}{s^2+0.062s+0.0037} \\ \frac{0.00016}{s+0.116} & \frac{0.00033}{s+0.722} & \frac{0.00000098e^{-8.5s}}{0.062s+1} & \frac{0.000512e^{-4s}}{s+0.267} & \frac{0.00125}{s+0.24} & \frac{0.00000044}{s+0.247} \\ \frac{0.00012e^{-s}}{s+0.089} & \frac{0.00016}{s+0.102} & \frac{0.0000065}{s+0.1008} & \frac{-0.000075e^{-14s}}{s+0.045} & \frac{0.0008}{s+0.2424} & \frac{0.00000071}{s+0.374} \\ \frac{-0.875e^{-4.04}}{2.74s+1} & \frac{-43.87e^{-10s}}{s+47.38} & \frac{-0.93e^{-3.14s}}{1.33s+1} & \frac{-0.581e^{-14.9s}}{0.00013s+1} & \frac{-0.51e^{-0.5s}}{s+0.249} & \frac{-0.00512e^{-3.65s}}{0.0836s+1} \\ \frac{-0.0087e^{-2.5s}}{s+0.126} & \frac{-0.012}{s+0.166} & \frac{-0.011e^{-0.5s}}{s+0.162} & \frac{0.0066e^{-1.5s}}{s+0.0535} & \frac{-0.0058e^{-1.5s}}{s+0.367} & \frac{0.00000062e^{-12s}}{s+0.0065} \\ \frac{-4.325e^{-6.5s}}{s+0.086} & \frac{0.000098e^{-1.6s}}{0.202s+1} & \frac{0.0000015}{s+0.0398} & \frac{0.000452e^{-0.5s}}{s+0.0993} & \frac{0.000412}{s+0.101} & \frac{0.00000012e^{-0.5s}}{s+0.076} \end{bmatrix} \tag{C.3}$$

$$G_C(s) = \begin{bmatrix} \frac{0.00089e^{-0.22s}}{0.0124s+1} & \frac{-0.000075e^{-10.5s}}{s+0.0568} & \frac{-0.00011e^{-9.5s}}{s+0.0383} & \frac{-0.00017e^{-12s}}{s+0.0315} & \frac{0.2325}{0.00011s+1} & \frac{-0.00062e^{-6.5s}}{s+0.011} \\ \frac{-0.0083e^{-3.5s}}{s^2+0.21s+0.063} & \frac{-0.000057}{6.63s+1} & \frac{-0.00017e^{-1.26s}}{1.7s+1} & \frac{-0.00005e^{-1.5s}}{s+0.148} & \frac{0.00059}{s+0.098} & \frac{-0.00089e^{-3.6s}}{1.414s+1} \\ \frac{-0.175e^{-8.01s}}{2.4s+1} & \frac{-0.0000044}{s+0.134} & \frac{-0.000037e^{-0.5s}}{s+0.1183} & \frac{-0.000072e^{-0.5s}}{s+0.0844} & \frac{0.00061}{s+0.0788} & \frac{-0.031e^{-14.5s}}{s+36.5} \\ \frac{0.000089}{s^2+0.066s+0.012} & \frac{0.0399e^{-6.14s}}{2.55s+1} & \frac{0.00086e^{-2.5s}}{s+0.255} & \frac{0.0019e^{-6s}}{s+0.215} & \frac{0.396e^{-0.19s}}{0.096s+1} & \frac{2.829e^{-14.5s}}{s+43.7} \\ \frac{0.00046e^{-6.43s}}{0.069s+1} & \frac{0.00094e^{-5.4s}}{0.065s+1} & \frac{0.015e^{-1.9s}}{0.073s+1} & \frac{0.0088}{0.0518s+1} & \frac{56.61e^{-2.5s}}{s+166.9} & \frac{-0.00057e^{-9.5s}}{s+0.0000037} \\ \frac{0.000428}{s^2+0.32s+0.0043} & \frac{0.000126e^{-15s}}{623s+1} & \frac{0.0000033e^{-s}}{s+0.12} & \frac{-0.0169e^{-1.5s}}{s+23.38} & \frac{0.003}{0.401s+1} & \frac{0.0000059e^{-6s}}{s+0.0114} \end{bmatrix} \tag{C.4}$$

$$G_D(s) = \begin{bmatrix} \frac{1.36}{s+1.072} & \frac{0.0025e^{-15s}}{3s+1} & \frac{-0.00061e^{-6.5s}}{0.02s+1} & \frac{0.218e^{-0.223s}}{0.101s+1} & \frac{0.243e^{-0.81s}}{0.096s+1} & \frac{0.000049}{58.7s+1} \\ \frac{0.00011e^{-1.5s}}{s+0.073} & \frac{0.00014e^{-0.5s}}{s+0.21} & \frac{0.000014e^{-8.46s}}{0.06s+1} & \frac{-0.00502}{s+0.575} & \frac{0.00081}{s+0.129} & \frac{0.0000003e^{-2s}}{s+0.154} \\ \frac{0.00011e^{-3.5s}}{s+0.063} & \frac{0.000138e^{-2.5s}}{s+0.0704} & \frac{0.00013}{s+0.091} & \frac{-0.000091e^{-3.5s}}{s+0.0313} & \frac{-0.00473}{s+0.256} & \frac{0.000002e^{-10.5s}}{0.000001s+1} \\ \frac{-0.012e^{-5.5s}}{s+0.095} & \frac{0.149e^{-s}}{s+0.135} & \frac{-0.0064e^{-10.3s}}{0.072s+1} & \frac{-1.21e^{-1.5s}}{s+0.168} & \frac{0.067e^{-2.5s}}{s+0.144} & \frac{0.0000009e^{-14.5s}}{s+0.002} \\ \frac{-0.026e^{-1.2s}}{0.013s+1} & \frac{-0.0412e^{-2.44s}}{0.088s+1} & \frac{13.37}{s+14.13} & \frac{-0.176e^{-10.65s}}{2.15s+1} & \frac{-14.64}{s+3.36} & \frac{0.0000004e^{-12.5s}}{s+0.000006} \\ \frac{-0.0052e^{-14.5s}}{s+24.9} & \frac{-0.000036e^{-9s}}{0.000001s+1} & \frac{-0.000025e^{-6.45s}}{2.32s+1} & \frac{0.00318}{s+0.043} & \frac{0.003}{1.34s+1} & \frac{-0.0000023}{s+1.18} \end{bmatrix} \tag{C.5}$$

where $G_{12}(s)$ is the 12 by 12 models of the crude distillation unit whose inputs-outputs pairing is presented in Table 4.13, Section 4.4.3; $G_A(s), G_B(s), G_C(s)$ and $G_D(s)$ are the subsets of the $G_{12}(s)$ models as presented above.

Appendix D

Dynamic RGA

The dynamic RGA was used together with the steady state RGA to analyse various control loops interaction in the CDU to obtain the best input-output pairing configuration. This of course was followed by rigorous simulations of the suggested pairing and reconfigurations by RGA and DRGA to achieve the best results. Examples of the application of the DRGA to the crude distillation unit control loops pairing for the fault free system and the reduced system models under reflux flow control valve actuator fault ($F2$) are given below. Given the transfer function models of the 5 by 5 fault free system as presented in equation 4.12 as:

$$G_5(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0321e^{-5.34s}}{1+0.086s} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{-0.055931}{1+1.001s} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.0535e^{-2.59s}}{1+1.414s} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-2.095e^{-14.5s}}{s+41.2} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1+1e{-}06s} \\ \frac{0.17986}{1+2.95s} & \frac{0.00035e^{-6s}}{s+0.0114} & \frac{0.191}{1+1.012s} & \frac{0.182}{1+1.37s} & \frac{0.000173}{s+1.46} \end{bmatrix} \tag{D.1}$$

If s in the denominators of the transfer function models in (D.1) above are replaced by jw (i.e. $s = jw$), where $j = \sqrt{-1}$ and w is the frequency range under which the control loops interactions are to be investigated. A frequency range of 10^2 to 10^1 is assumed. The dynamic RGA of the system in (D.1) can be obtained by replacing the steady state gain matrix in equation 3.8 with the system transfer function models ($G_5(s)$) and testing over the frequency range 10^2 to 10^1.

$$\Lambda_{Dynamic} = G_5.^*(G_5^T)^{-1} \tag{D.2}$$

where $\Lambda_{Dynamic}$ is the dynamic RGA of the fault free system. Figure D.1 below presents the plots of the relative gains of the CDU input-output pairing presented in Table 4.14. It can be observed from Figure D.1 that the relative gains for the input-output pairing across the frequency range support the input-output pairing. The results of the dynamic RGA are supported by those of the RGA given in equation 4.13 and also confirmed through rigorous simulations of the different inputs-outputs pairing combinations.

Similarly, using the transfer function models of the reduced 4 by 4 system under fault $F2$ as presented in equation 4.16 (D.3) and applying equation (D.2) using $G_{F2}(s)$ over the same frequency range gives the dynamic RGA (Λ_{DF2}) for the control loop reconfiguration under $F2$ shown in Figure D.2. The relative gains in Figure D.2 confirm the reconfiguration pairing used to accommodate actuator fault $F2$, and also support the RGA results in equation 4.17.

$$G_{F2}(s) = \begin{bmatrix} \frac{30.85}{s+26.11} & \frac{0.0065e^{-1.5s}}{s+0.054} & \frac{0.00014e^{-1.5s}}{s+0.361} & \frac{6.18e{-}07e^{-12s}}{s+0.0068} \\ \frac{0.0145}{s+0.064} & \frac{0.0271e^{-0.5s}}{s+0.099} & \frac{0.0247}{s+0.101} & \frac{7.39e{-}06e^{-0.5s}}{s+0.0778} \\ \frac{0.03556}{s+0.098} & \frac{-0.301}{s+0.575} & \frac{0.0485}{s+0.129} & \frac{1.61e{-}05e^{-2s}}{s+0.1536} \\ \frac{0.03637}{s+0.078} & \frac{-0.0055e^{-3.5s}}{s+0.031} & \frac{-0.2837}{s+0.256} & \frac{0.00011e^{-10.5s}}{1{+}1e{-}06s} \end{bmatrix} \tag{D.3}$$

$$\Lambda_{DF2} = G_{F2}.^{*}(G_{F2}^{T})^{-1} \tag{D.4}$$

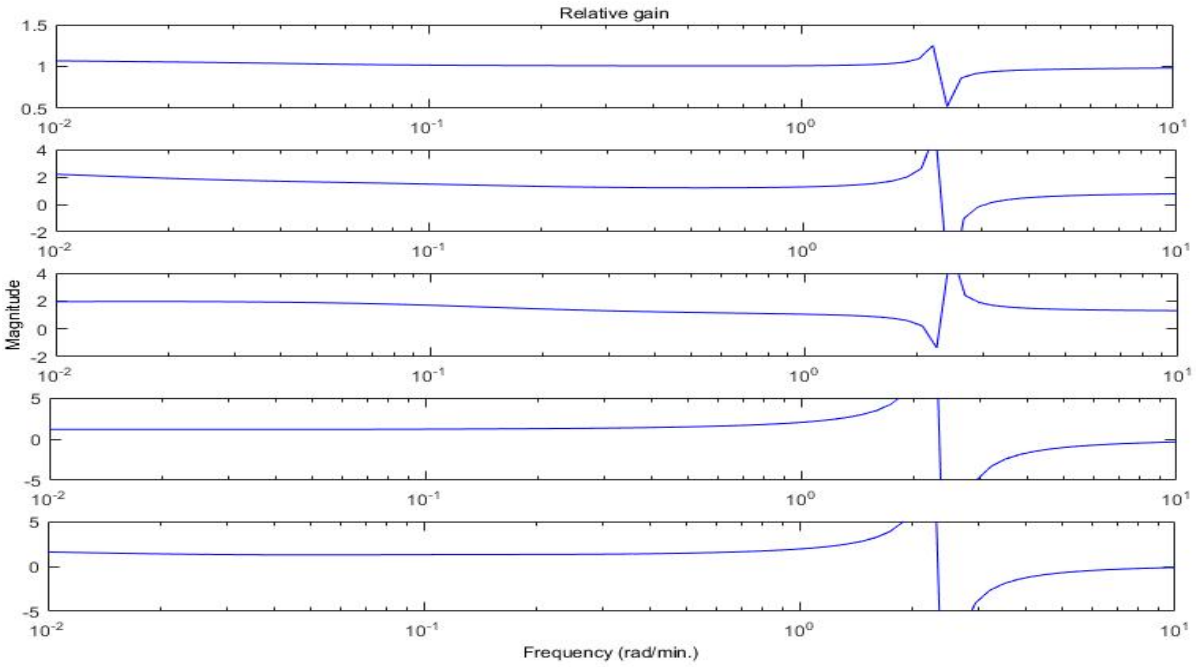


Figure D.1: Frequency response of dynamic RGA for the fault-free system

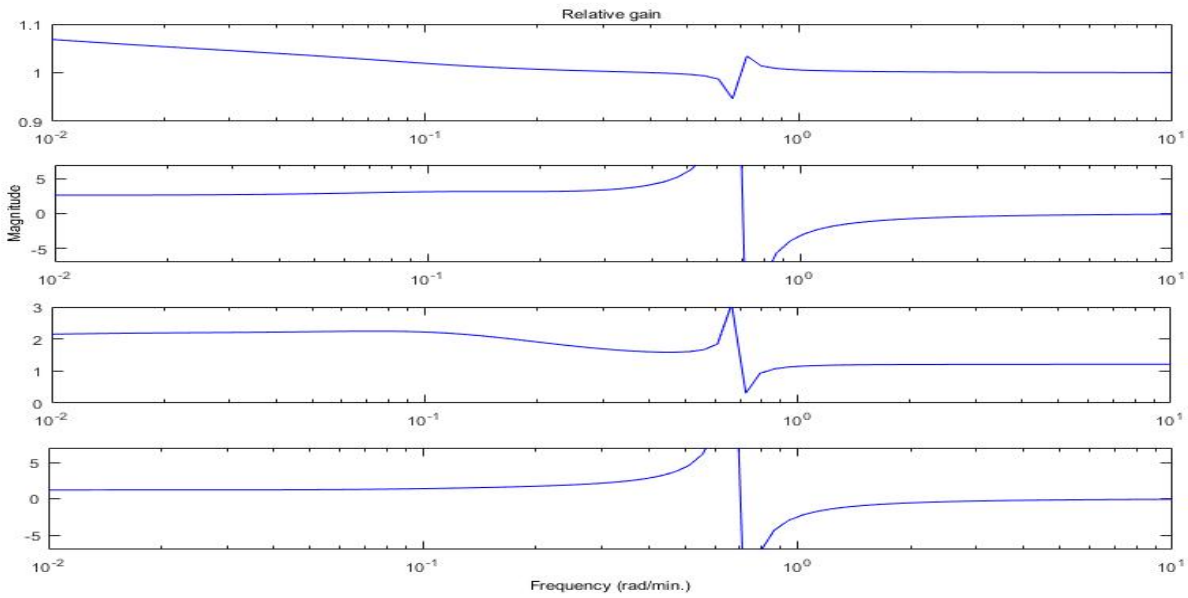


Figure D.2: Frequency response of dynamic RGA for the CDU under fault F2