Robust Fault Tolerant Detection and Isolation Techniques for Actuator Failures in Dynamical Systems

Nikolaos Tantouris

Department of Informatics
University of Piraeus

This dissertation is submitted for the degree of
Doctor of Philosophy

January 2017

University of Piraeus
Department of Informatics

Doctoral Dissertation

"Robust Fault Tolerant Detection and Isolation Techniques for Actuator Failures in Dynamical Systems"

by **Nikolaos M. Tantouris**

Advisory Committee: Despina Polemi, **Associate Professor, University of Piraeus**

Lena Valavani, **Distinguished Member of the Technical Staff, Draper; Fmr. Boeing Associate Professor of Aeronautics and Astronautics, MIT**

Christos Douligeris, **Professor, University of Piraeus**

Panagiotis Kotzanikolaou, **Assistant Professor, University of Piraeus**

Approved by the seven (7) member committee in **20-01-2017**

Despina Polemi
Associate Professor
University of Piraeus

Christos Douligeris
Professor
University of Piraeus

Panagiotis Kotzanikolaou
Assistant Professor
University of Piraeus

Gregory Chondrokoukis
Professor
University of Piraeus

Michael Psarakis
Assistant Professor
University of Piraeus

Dimitrios Tseles
Professor
Piraeus University of
Applied Sciences

Michalis Papoutsidakis
Assistant Professor
Piraeus University of
Applied Sciences

For my father and mother, who helped me in all things great and small ...

Declaration

This dissertation is the result of my own work and includes nothing, which is the outcome of work done in collaboration except where specifically indicated in the text. It has not been previously submitted, in part or as a whole, to any university of institution for any degree, diploma, or other qualification.

Nikolaos Tantouris
January 2017

Acknowledgements

It is extremely hard for me to gratitude within few lines of text, every single person that helped me during this journey.

Taking advantage of a dictum by Alexander the Great "*I am indebted to my father for living, but to my teacher for living well*", I would like to thank first of all, my family, who supports me in every aspect of my life since the day that I was born, and then every educationist that has shared inexhaustible knowledge with me, so far. Also, I would like to warmly thank my friends who honored me with their friendship, understanding and unwavering patience, especially the last four years.

In conclusion, I would like to ensure everyone that I will keep trying to uncover the reason of me being here and I hope that during this life journey I will find more people to share even more knowledge and experiences because as Socrates said "*One thing I know, that I know nothing*".

Abstract

This doctoral dissertation addresses the inefficiencies of the currently applied 'Fault Detection and Isolation' (FDI) techniques, calling for the generation of algorithms which can handle a wide variety of fault profiles and scenarios more effectively. The overarching objective of this dissertation was to develop alternative state-of-the-art methodologies which can be robust, simple to implement, and able to function reliably, in real-time and on a continual basis. To this effect, three novel algorithms fulfilling all the above requirements are derived and presented herein. These three filters are applied to the Guidance Navigation and Control (GNC) designs for autonomous spacecraft and validated within the Lisa Pathfinder (LPF) context in a simulation environment provided by the European Space Agency (ESA). The theoretical basis for their development and application is also amenable to other spacecraft, since robust FDI is critical for operational autonomy. The presented FDI techniques, namely the Diagnosis, Euresis and Euphoria Filters, are generated over two individual phases. Following a comprehensive literature review of the subject matter, initially, two fundamentally different model-based approaches are formulated and pursued, reflective of the two main trends of the existing model-based methods: the geometric approach and the multiple model approach. The novelty of the Diagnosis Filter (geometric approach) lies in that, in contrast to conventional methods, it does not only generate structured residuals, but also it employs directionality rather than residuals' magnitude as a diagnostic tool for faults (deterministic approach). This filter exhibits remarkable robustness and performs exceptionally well, regardless of the thruster's failure severity. In comparison, the Euresis Filter (multiple model approach) exhibits similar robustness, albeit for total failures only. However, it has the advantage of being applicable to non-linear design models unlike the Diagnosis Filter. During the second phase, directionality is incorporated in another optimisation-based geometric approach (i.e. Euphoria filter). In this case, directionality is not designed in at the outset but it is rather recovered by an adjoint operation on the H_2 filter residuals (optimal stochastic approach). The Euphoria Filter outperforms the other two in most evaluation criteria and it can

be applied in both linear and non-linear FDI scenarios. In summary, this thesis addresses the weaknesses of the currently applicable FDI techniques, outlines the objectives of this dissertation, and presents three novel FDI methods, extensively tested via Monte Carlo simulations and ultimately validated on the LPF model. The thesis concludes with recommendations for future work, potential developments in the field and alternative applications of these techniques in other domains such as IoT and Cyber-Physical systems.

Table of contents

List of figures **xv**

List of tables **xxi**

Nomenclature **xxv**

1 Introduction **1**

2 System Characteristics, Design Requirements and Objectives **9**
2.1 LPF: System Engineering Characteristics and Requirements 11
2.1.1 Multivariable Analysis of the Overall System Characteristics . 11
2.1.2 Specifications and Requirements 14
2.1.3 The Singular Values Analysis and State Space Characteristics 16
2.1.4 Singular Value Plots 17
2.1.5 Controllers, Thruster Failures and Stabilization 30
2.2 FDI Objectives 31
2.3 Types of Faults Considered 34
2.4 Evaluation Criteria for the FDI Algorithms 36

3 A Critical Survey of Model-based FDI Methods **39**
3.1 The Geometric and Analytical Redundancy Approach 39
3.2 The Multiple Model Approach 44
3.3 The Optimization Based Approach to FDI Using H_∞ 47
3.4 FDI Method Suitability - Selection Criteria 48
3.5 Rationale for Specific Architecture(s) Selection 49
3.5.1 Linear Model Availability 49
3.5.2 Capability for Fault Identification 50
3.5.3 Robustness 50
3.5.4 Real-Time Capability 51

3.5.5 Method Genericity- Suitability for Other Space Applications . 51

4 FDI Techniques 55
4.1 Design Method 1: Diagnosis Filter 57
4.1.1 Theoretical Foundation 57
4.1.2 Outline of the Diagnosis-Filter Operation 59
4.2 Design Method 2: The Euresis Filter 66
4.2.1 Theoretical Foundation 66
4.2.2 Outline of the Euresis Filter Operation 68
4.3 Design Method 3: The Euphoria Filter 72
4.3.1 Theoretical Foundation 72
4.3.2 Outline of the Euphoria-Filter Operation 77
4.4 The Failure Severity Algorithm 81

5 Evaluation of the FDI Techniques 85
5.1 Validation Plan 85
5.1.1 Performance Evaluation 85
5.1.2 Sensitivity versus Robustness 86
5.2 Design Method 1: Diagnosis Filter 91
5.3 Design Method 2: The Euresis Filter 112
5.4 Design Method 3: Euphoria Filter 125

6 Conclusions and Future Work 163
6.1 Conclusions 163
6.2 Future Work 164

References 173

Appendix A State Space Descriptions of Open Loop Dynamics 181

Appendix B Drag Free and Suspension Loop Dynamics (Combined) for the Detection Filter - Modified B Matrix 223

Appendix C Diagnosis Filter (The true identification of the Failed Thruster was missed in 430 Cases, of which 302 with Negative Severity) 229

List of figures

1.1 The Lisa Pathfinder Satellite (taken from ESA [1]) 3
1.2 The Thrusters of the Lisa Pathfinder (taken from ESA [1]) 4

2.1 LPF Design Model (Control Loops) (taken from simulation environment) 11
2.2 DF, SUS, and ATT Control Principle - adapted from [2] 12
2.3 Singular values of the DF open loop (Astrium) 18
2.4 Singular values of the DF open loop (simulation environment) 18
2.5 Singular Values of the DF Compensator (Astrium) 19
2.6 The DF loop gain (Astrium) 20
2.7 The DF loop gain (simulation environment) 20
2.8 Singular Values of the SUS open loop (Astrium) 22
2.9 Singular Values of the SUS open loop (simulation environment) . . . 22
2.10 Singular Values of the SUS Loop Compensator (Astrium) 23
2.11 Singular Values of the SUS Loop Gain (Astrium) 24
2.12 Singular Values of SUS Loop Gain (simulation environment) 24
2.13 Singular Values of the ATT Open Loop (Astrium) 25
2.14 Singular Values of the ATT Open Loop (simulation environment) . . 26
2.15 Singular Values of the ATT Loop Compensator (Astrium) 27
2.16 Singular Values of the ATT Loop Gain (Astrium) 27
2.17 Singular Values of the ATT Loop Gain (simulation environment) . . . 28
2.18 Singular Values of the DF and SUS loops 29
2.19 Structural Diagram of the LPF's FDI Generic Concept 32

4.1 The Structure of the Diagnosis Filter 58
4.2 The Combined Dynamics of DF and SUS loops 61
4.3 Overall Operation of the Diagnosis Filter Algorithm 62
4.4 The Structure of the Euresis Filter 67

4.5 The generic design framework for H_2, H_∞ 72
4.6 Euphoria Filter Design Architecture 77
4.7 Severity Calculation Flow Diagram 84

5.1 Diagnosis Filter - Tuning Methodology 88
5.2 Euresis Filter - Tuning Methodology 89
5.3 Euphoria Filter - Tuning Methodology 90
5.4 Diagnosis Filter - Thrusters Colour Code 91
5.5 Diagnosis Filter - Thrusters without Failure 92
5.6 Diagnosis Filter - 1^{st} Thruster Fault (Total Failure) 92
5.7 Diagnosis Filter - 2^{nd} Thruster Fault (Total Failure) 93
5.8 Diagnosis Filter - 3^{rd} Thruster Fault (Total Failure) 93
5.9 Diagnosis Filter - 4^{th} Thruster Fault (Total Failure) 94
5.10 Diagnosis Filter - 5^{th} Thruster Fault (Total Failure) 94
5.11 Diagnosis Filter - 6^{th} Thruster Fault (Total Failure) 95
5.12 Diagnosis Filter - 1^{st} and 2^{nd} Thruster Fault (Total Failure) 97
5.13 Diagnosis Filter - 1^{st} and 5^{th} Thruster Fault (Total Failure) 97
5.14 Diagnosis Filter - 2^{nd} and 3^{rd} Thruster Fault (Total Failure) 98
5.15 Diagnosis Filter - 2^{nd} and 5^{th} Thruster Fault (Total Failure) 98
5.16 Diagnosis Filter - 2^{nd} and 6^{th} Thruster Fault (Total Failure) 99
5.17 Diagnosis Filter - 3^{rd} and 4^{th} Thruster Fault (Total Failure) 99
5.18 Diagnosis Filter - 3^{rd} and 6^{th} Thruster Fault (Total Failure) 100
5.19 Diagnosis Filter - 4^{th} and 5^{th} Thruster Fault (Total Failure) 100
5.20 Diagnosis Filter - 2^{nd}, 3^{rd} and 6^{th} Thruster Fault (Total Failure) 101
5.21 Diagnosis Filter - 1^{st}, 4^{th} and 5^{th} Thruster Fault (Total Failure) 101
5.22 Diagnosis Filter - Distribution of Failed Thrusters for 4th Monte Carlo Set 105
5.23 Diagnosis Filter - Distribution of Severity Magnitudes for 4th Monte Carlo Set 105
5.24 FDI Detection Timing Profiles (Rounded Off) 106
5.25 Diagnosis Filter - Timing Profile showing Instability Inception Point Beyond which FDI is Jeopardised for 1^{st} case 108
5.26 Diagnosis Filter - Zooming in to Detection & Identification Time for 1^{st} case 109
5.27 Diagnosis Filter - Further Zooming in to Detection & Identification Time Showing the Effects of Noise for 1^{st} case 109

5.28 Diagnosis Filter - Timing Profile showing Instability Inception Point Beyond which FDI is Jeopardised for 2^{nd} case 110
5.29 Diagnosis Filter - Zooming in to Detection & Identification Time for 2^{nd} case 111
5.30 Diagnosis Filter - Further Zooming in to Detection & Identification Time Showing the Effects of Noise for 2^{nd} case 111
5.31 Euresis Filter - Thrusters Without Failure 113
5.32 Euresis Filter – 1^{st} Thruster (Total Failure) 114
5.33 Euresis Filter – 2^{nd} Thruster (Total Failure) 114
5.34 Euresis Filter – 3^{rd} Thruster (Total Failure) - Different Scale to Show Presence of Disturbances in Residual 115
5.35 Euresis Filter – 3^{rd} Thruster (Total Failure) 116
5.36 Euresis Filter – 4^{th} Thruster (Total Failure) 116
5.37 Euresis Filter – 5^{th} Thruster (Total Failure) 117
5.38 Euresis Filter – 6^{th} Thruster (Total Failure) 117
5.39 Euresis Filter – 7^{th} Thruster (Total Failure) 118
5.40 Euresis Filter – 8^{th} Thruster (Total Failure) 118
5.41 Euresis Filter – 1^{st} and 2^{nd} Thruster (Total Failure) 119
5.42 Euresis Filter – 1^{st} and 8^{th} Thruster (Total Failure) 119
5.43 Euresis Filter – 1^{st} , 5^{th} and 8^{th} Thruster (Total Failure) 120
5.44 Euresis Filter - Distribution of Severity Profiles 122
5.45 Euresis Filter - Distribution of Thrusters' Failures 123
5.46 Euresis Filter - Distribution of Correctly Identified Failed Thrusters . 123
5.47 Euphoria Filter - Distribution of Failed Thrusters for the 1^{st} case . . . 127
5.48 Euphoria Filter - Distribution of Severity Magnitudes for the 1^{st} case 127
5.49 Euphoria Filter - Distribution of Failed Thrusters for the 2^{nd} case . . . 129
5.50 Euphoria Filter - Distribution of Severity Magnitudes for the 2^{nd} case 129
5.51 Euphoria Filter - Distribution of Failed Thrusters for the 3^{rd} case . . . 130
5.52 Euphoria Filter - Distribution of Severity Magnitudes for the 3^{rd} case 130
5.53 Euphoria Filter - Distribution of Failed Thrusters for the 4^{th} case . . . 132
5.54 Euphoria Filter - Distribution of Severity Magnitudes for the 4^{th} case 132
5.55 Euphoria Filter - Distribution of Failed Thrusters for the 5^{th} case . . . 133
5.56 Euphoria Filter - Distribution of Severity Magnitudes for the 5^{th} case 134
5.57 Euphoria Filter - Distribution of Failed Thrusters for the 6^{th} case . . . 136
5.58 Euphoria Filter - Distribution of Severity Magnitudes for the 6^{th} case 137
5.59 Euphoria Filter - FDI Timing Profile for 2^{nd} Case 138

5.60 Euphoria Filter – Thrusters' Colour Code for Leakage Test Runs . . . 140
5.61 Euphoria Filter - Graph of Leakage for 1^{st} Set . . . 141
5.62 Euphoria Filter - Graph of Severity for 1^{st} Set . . . 141
5.63 Euphoria Filter - Graph of Residuals for 1^{st} Set . . . 142
5.64 Euphoria Filter – Thrusters' Angles for 1^{st} Set . . . 142
5.65 Euphoria Filter - Graph of Leakage for 2^{nd} Set . . . 143
5.66 Euphoria Filter - Graph of Severity for 2^{nd} Set . . . 143
5.67 Euphoria Filter - Graph of Residuals for 2^{nd} Set . . . 144
5.68 Euphoria Filter – Thrusters' Angles for 2^{nd} Set . . . 144
5.69 Euphoria Filter - Graph of Leakage for 3^{rd} Set . . . 145
5.70 Euphoria Filter - Graph of Severity for 3^{rd} Set . . . 145
5.71 Euphoria Filter - Graph of Residuals for 3^{rd} Set . . . 146
5.72 Euphoria Filter – Thrusters' Angles for 3^{rd} Set . . . 146
5.73 Euphoria Filter - Graph of Leakage for 4^{th} Set . . . 147
5.74 Euphoria Filter - Graph of Severity for 4^{th} Set . . . 147
5.75 Euphoria Filter - Graph of Residuals for 4^{th} Set . . . 148
5.76 Euphoria Filter – Thrusters' Angles for 4^{th} Set . . . 148
5.77 Euphoria Filter - Graph of Leakage for 5^{th} Set . . . 149
5.78 Euphoria Filter - Graph of Severity for 5^{th} Set . . . 149
5.79 Euphoria Filter - Graph of Residuals for 5^{th} Set . . . 150
5.80 Euphoria Filter – Thrusters' Angles for 5^{th} Set . . . 150
5.81 Euphoria Filter - Graph of Leakage for 6^{th} Set . . . 151
5.82 Euphoria Filter - Graph of Severity for 6^{th} Set . . . 151
5.83 Euphoria Filter - Graph of Residuals for 6^{th} Set . . . 152
5.84 Euphoria Filter – Thrusters' Angles for 6^{th} Set . . . 152
5.85 Euphoria Filter - Graph of Leakage for 7^{th} Set . . . 153
5.86 Euphoria Filter - Graph of Severity for 7^{th} Set . . . 153
5.87 Euphoria Filter - Graph of Residuals for 7^{th} Set . . . 154
5.88 Euphoria Filter – Thrusters' Angles for 7^{th} Set . . . 154
5.89 Euphoria Filter - Graph of Leakage for 8^{th} Set . . . 155
5.90 Euphoria Filter - Graph of Severity for 8^{th} Set . . . 155
5.91 Euphoria Filter - Graph of Residuals for 8^{th} Set . . . 156
5.92 Euphoria Filter – Thrusters' Angles for 8^{th} Set . . . 156
5.93 Euphoria Filter - Graph of Leakage for 9^{th} Set . . . 157
5.94 Euphoria Filter - Graph of Severity for 9^{th} Set . . . 157
5.95 Euphoria Filter - Graph of Residuals for 9^{th} Set . . . 158

5.96 Euphoria Filter – Thrusters' Angles for 9^{th} Set 158
5.97 Euphoria Filter - Singular Values of G_{FOL} 161
5.98 Euphoria Filter - Singular Values of G_{KF} 161
5.99 Euphoria Filter - H_2 matrix 162

6.1 Components of Modelling Methods (Karagiannis & Kühn [3]) 169

A.1 Attitude Open Loop - Matrix A of Astrium's TF 182
A.2 Attitude Open Loop - Matrix B of Astrium's TF 183
A.3 Attitude Open Loop - Matrix C of Astrium's TF 184
A.4 Attitude Open Loop - Matrix D of Astrium's TF 185
A.5 Attitude Open Loop - Matrix A of Astrium Compensator's TF 186
A.6 Attitude Open Loop - Matrix B of Astrium Compensator's TF 187
A.7 Attitude Open Loop - Matrix C of Astrium Compensator's TF 188
A.8 Attitude Open Loop - Matrix D of Astrium Compensator's TF 189
A.9 Attitude Open Loop - Matrix A of Simulation Environment 190
A.10 Attitude Open Loop - Matrix B of Simulation Environment 191
A.11 Attitude Open Loop - Matrix C of Simulation Environment 192
A.12 Attitude Open Loop - Matrix D of Simulation Environment 193
A.13 Drag Free Open Loop - Matrix A of Astrium's TF 194
A.14 Drag Free Open Loop - Matrix B of Astrium's TF 195
A.15 Drag Free Open Loop - Matrix C of Astrium's TF 196
A.16 Drag Free Open Loop - Matrix D of Astrium's TF 197
A.17 Drag Free Open Loop - Matrix A of Astrium Compensator's TF . . . 198
A.18 Drag Free Open Loop - Matrix B of Astrium Compensator's TF . . . 199
A.19 Drag Free Open Loop - Matrix C of Astrium Compensator's TF . . . 200
A.20 Drag Free Open Loop - Matrix D of Astrium Compensator's TF . . . 201
A.21 Drag Free Open Loop - Matrix A of Simulation Environment 202
A.22 Drag Free Open Loop - Matrix B of Simulation Environment 203
A.23 Drag Free Open Loop - Matrix C of Simulation Environment 204
A.24 Drag Free Open Loop - Matrix D of Simulation Environment 205
A.25 Suspension Open Loop - Matrix A of Astrium's TF 206
A.26 Suspension Open Loop - Matrix B of Astrium's TF 207
A.27 Suspension Open Loop - Matrix C of Astrium's TF 208
A.28 Suspension Open Loop - Matrix D of Astrium's TF 209
A.29 Suspension Open Loop - Matrix A of Astrium Compensator's TF . . 210
A.30 Suspension Open Loop - Matrix B of Astrium Compensator's TF . . 211

A.31 Suspension Open Loop - Matrix C of Astrium Compensator's TF . . 212
A.32 Suspension Open Loop - Matrix D of Astrium Compensator's TF . . 213
A.33 Suspension Open Loop - Matrix A of Simulation Environment 214
A.34 Suspension Open Loop - Matrix B of Simulation Environment 215
A.35 Suspension Open Loop - Matrix C of Simulation Environment 216
A.36 Suspension Open Loop - Matrix D of Simulation Environment 217
A.37 Combination of Drag Free and Suspension Loops - Matrix A of Simulation Environment 218
A.38 Combination of Drag Free and Suspension Loops - Matrix B of Simulation Environment 219
A.39 Combination of Drag Free and Suspension Loops - Matrix C of Simulation Environment 220
A.40 Combination of Drag Free and Suspension Loops - Matrix D of Simulation Environment 221

B.1 Combined Drag Free and Suspension Loop - Matrix A of FDI Techniques224
B.2 Combined Drag Free and Suspension Loop - Matrix B of FDI Techniques225
B.3 Combined Drag Free and Suspension Loop - Matrix C of FDI Techniques226
B.4 Combined Drag Free and Suspension Loop - Matrix D of FDI Techniques227

List of tables

2.1 Nature of errors that affect the LPF's thrusters 35

3.1 Comparison of Methods based on Selective Criteria 53

4.1 The Output Separable Failure Directions of DF_1 63
4.2 The Output Separable Failure Directions of DF_2 64

5.1 Diagnosis Filter - Severity Magnitudes and Noise levels of Validation Plan 102
5.2 Diagnosis Filter - Leakage Magnitudes and Noise Levels of Validation Plan 102
5.3 Diagnosis Filter – Summarized Results of Four Sets of Monte Carlo Analysis 107
5.4 Diagnosis Filter - min Angle Evolution for 1^{st} case 108
5.5 Diagnosis Filter - min Angle Evolution for 2^{nd} case 110
5.6 Euphoria Filter - Severity Magnitudes and Noise levels of Validation Plan 125
5.7 Euphoria Filter - Leakage Magnitudes and Noise Levels of Validation Plan 125
5.8 Euphoria Filter - Missed Identification Cases for the 1^{st} case 128
5.9 Euphoria Filter - Missed Identification Cases for the 3^{rd} case 131
5.10 Euphoria Filter - Missed Identification Cases for the 4^{th} case 133
5.11 Euphoria Filter - Missed Identification Cases for the 5^{th} case 135
5.12 Euphoria Filter – Summarized Results of Six Sets of Monte Carlo Analysis 138
5.13 Euphoria Filter – Leakage Cases - Time to reach Severity 159

6.1 Comparison of the Three FDI Techniques based on Design Characteristics 165

6.2 Comparison of the Three FDI Techniques based on Evaluation Characteristics 166
6.3 Comparison of the Three FDI Techniques based on Application Characteristics 167

C.1 Euphoria Filter - Missed Identification Cases for the 6^{th} case 230

Nomenclature

Acronyms / Abbreviations

ATT Attitude

CARE Control Algebraic Riccati Equationa

CPS Cyber-Physical Systems

DF Diagnosis Filter

DF Drag-Free

DFACS Drag-Free and Attitude Control System

DRS Disturbance Reduction System

EADS European Aeronautic Defence and Space

EKF Extended Kalman Filter

EO Earth Observation

ESA European Space Agency

ESTEC European Space Research and Technology Centre

FARE Filter Algebraic Riccati Equation

FDI Fault Detection and Isolation

FDIR Fault Detection, Isolation and Recovery

FEEP Field Emission Electric Propulsion

FSO Full State Observers

GNC Guidance Navigation and Control

GSF Gaussian Sum Filter

HIMM Hierarchical Interactive Multiple Model

HIMMML Hierarchical Interactive Multiple Model Maximum Likelihood Estimation

IMU Inertial Measurement Unit

IoT Internet of Things

KF Kalman Filter

KFDE Kalman Frequency Domain Equality

LFT Linear Fractional Transformation

LMI Linear Matrix Inequality

LPF Lisa Pathfinder

LPV Linear Parameter Varying

LTI Linear Time Invariant

LTP Lisa Technology Package

MM Multiple Model

MMKF Multiple Model Kalman Filter

NF Noise Factor

RHS Right-Hand Side

SUS Suspension

SVD Singular Value Decomposition

TF Transfer Function

UAV Unmanned Aerial Vehicle

UIO Unknown Input Observer

UKF Unscented Kalman Filter

$ULPE$ Unknown Linear Parametric Estimators

$V\&V$ Verification and Validation

Chapter 1

Introduction

A fault (malfunction) is defined as a deviation from a system's normal function, which detrimentally disturbs and impairs its performance, adversely affecting its overall operation. A fault diagnosis should be achieved as early as possible even if it could be initially tolerated by the system, to prevent further function deterioration and serious risks for the mission [4]. A fault diagnosis system comprises the following tasks as outlined in [5]:

- Fault Detection: detection of the time of fault occurrence in the functional units of the process, leading to undesired or unacceptable behaviour of the whole system.
- Fault Isolation: localization (classification) of different faults.
- Fault Analysis or Identification: determination of the exact nature, magnitude and cause of the fault.

A fault diagnosis system, depending on its performance, is called Fault Detection and Isolation (FDI) [6]. The Robust Fault Detection and Isolation has become a critical property for virtually all space missions and applications, since operational autonomy is of paramount importance. Such systems are required to function reliably and autonomously, to varying degrees, which presupposes their ability to not only robustly detect and identify failures but to also have the ability to reconfigure and continue operation in the event of a malfunction in their control systems (sensors or actuators). The Robust Fault Detection and Isolation can be challenging because of modelling inaccuracies in the space system design and their operating in noisy environments with considerable disturbances and uncertainty. This issue is further exacerbated in the context of smaller, less expensive spacecraft, where hardware

complexity, cost, and mass are to be kept to a minimum [7–11]. The key critical features of space missions where autonomy is a prime consideration, can therefore be identified as scalability, detectability and identifiability, re-configurability and robustness to failures [12–14].

The main purpose of the FDI function is to maintain the vehicle operational and safe. FDI requirements are strongly dependent on the mission type. For instance, Earth Observation (EO) missions ask for simplicity and can tolerate certain outages while commercial telecommunication satellites have a strong requirement regarding the outage duration. For science observation missions, the FDI requirements are very similar to those of the EO missions, while exploration missions put emphasis on availability during critical phases such as in orbit insertion, in proximity operations, or in the terminal rendezvous phase [15].

Lisa Pathfinder (LPF), a technology demonstrator spacecraft mission initiated by the European Space Agency (ESA) (Figure 1.1), falls into this last category, with stringent requirements for performance reliability and autonomy. The objective of LPF has been the testing and verification of key technologies and innovative instruments for the observation of gravitational waves - fluctuations in the fabric of spacetime - from space. In the core of the Lisa Technology Package (LTP) two test masses lie, that consist of two identical 2 kg, 46 mm, gold-platinum cubes, 38 cm apart, each surrounded by a sensor cage. Launched on December 3^{rd}, 2015, LPF has now exceeded its operational goal of placing the two test masses in the most precise gravitational free-fall ever (at 500000 by 800000 - km halo orbit around the first Sun - Earth Lagrange point, at 1.5 million km from Earth).

The test masses were shielded to an unprecedented degree from all internal and external forces by the spacecraft's sensing their motion and manoeuvring around them, while their position and attitude were measured with utmost accuracy by a laser interferometer. The derived results essentially pave the way for the development of large scale, space-borne observatories capable of detecting gravitational waves. The LPF concluded its first operational phase on the 25^{th} of June 2016, and has started running the Disturbance Reduction System (DRS), an additional experiment provided by NASA [1].

DRS is a NASA generated system of thrusters, advanced avionics, and software. Although most thrusters serve for spacecraft propulsion, DRS has a distinct purpose within the context of LPF: to hold it as perfectly still as possible. This enables the testing of technologies used in the detection of gravitational waves, which requires extreme steadiness and stability. DRS possesses eight thrusters positioned on either

side of the LPF spacecraft (Figure 1.2). Each thruster emits microscopic ionised liquid droplets (colloid electrospray), charged through an electric field and accelerated by a second one with an opposite charge. The generated force provides the thrust which stabilises the spacecraft [16].

The entire above highlight the importance of thrusters in a spacecraft, the LPF in this case, and the requirement for optimal FDI, which in turn will warrant the operational viability of a mission. Current FDI architectures are hierarchical, i.e. various FDI solutions are implemented at unit, subsystem, survival, and safety level. As exhibited in cases of orbit failure, this architecture has some inherent weaknesses, such as being sensitive to dormant fault, ground-programming errors and requiring extra hardware to ensure safety after a second failure. Although, standard space program FDI practices are very efficient to detect faults at unit level - using mostly model-based techniques [17, 18] - they are rarely used at system level. This is largely attributed to the following reasons: first, the system model is much more complex than a single unit model and, secondly, it is rarely required, due to the current avionics and Guidance, Navigation and Control (GNC) architecture. However, accumulated experience has revealed the distinct advantages of the model-based methods. For instance, a model-based system level FDI can be useful for critical safety applications such as for autonomous rendezvous, where an active robust FDI

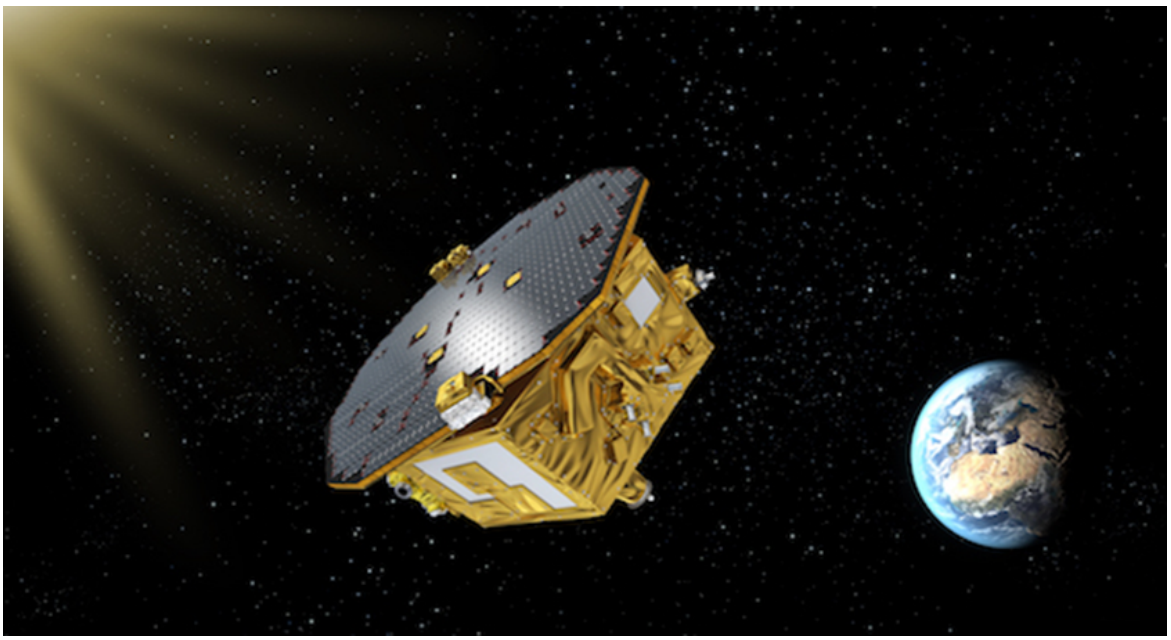


Fig. 1.1 The Lisa Pathfinder Satellite (taken from ESA [1])

in the safeguard function would limit or even eliminate the need for extra hardware and reduce reliance on the operational software [19].

The different approaches to fault diagnosis (reviewed and compared in [20–22]) can be classified into three main categories:

i. Model-based techniques.

ii. Knowledge-based techniques.

iii. Signal Processing-based techniques.

For space applications, detailed satellite models are usually available; therefore, model-based techniques appear to be the most appropriate solutions for advancing the state-of-the-art in FDI for autonomous satellites. In this context, the robust model-based FDIR (Fault Detection, Isolation and Recovery) hinges on the purity of the signals used, most often referred to as robust residuals, which are to be devoid of any noise or disturbance contamination, to the extent possible. Additionally, FDIR requires sufficient knowledge of the possible faults that could arise, and a good system model. These are the basic elements for a model-based approach, which

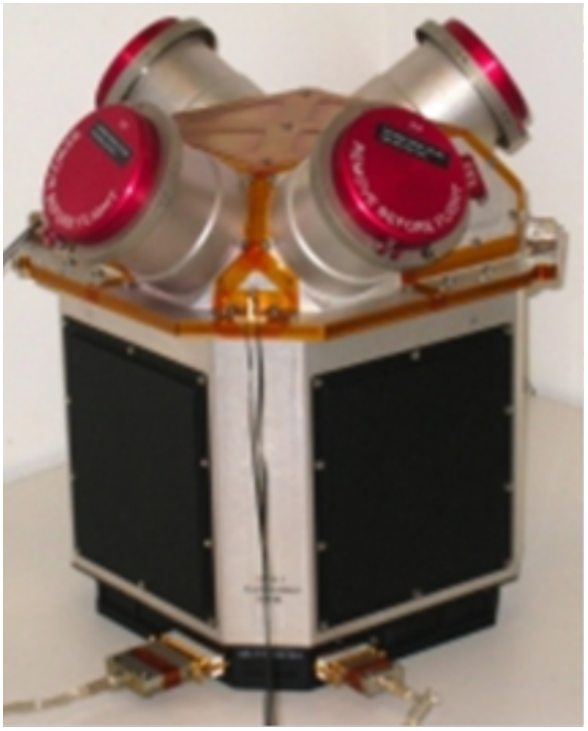


Fig. 1.2 The Thrusters of the Lisa Pathfinder (taken from ESA [1])

explicitly makes use of the underlying system model in the various fault detection designs employed. Robust FDI, then, refers to the desired correct fault/failure detection and isolation/identification in the presence of both unstructured (dynamic) or structured (parametric) uncertainty and exogenous system disturbances and sensor noise.

Given the technical specifications of LPF, which is the application of interest to ESA and the mission's requirements, it becomes clear that the model-based FDI is the indicated technology and, therefore, the focus of this thesis. The research presented here was geared for application to LPF, with the requirement that the resulting algorithms be real-time, robust, and able to handle a wide variety of thruster failures. LPF should ideally be able to detect, identify and remedy its own failed components [6, 23] most frequently found in actuators or sensors – the thrusters in the case of this study. Software and sensor failures come under an entirely different framework and are not addressed in this thesis; however, the presented mathematical model can be also used as the basis for handling such failures.

Hardware redundancy and associated voting mechanisms have been employed in the past, and represent a brute force way of failure detection. More recently, model-based analytical redundancy techniques, based on a well-developed theory for FDI, represent the way of the future. A high level of hardware redundancy just for FDI is not only bulky and heavy for space, but also costly [4, 24].

Analytical redundancy methods rely on a spacecraft model whose responses are compared to those of the plant (spacecraft) under the same inputs. Consequently, such methods have also become known as model-based. If a failure occurs, in a sensor or in an actuator, the plant will be affected. Consequently, the plant will display a different behaviour than that of the model under identical input conditions. In such a case, the detection of failure comprises the first step; the ensuing failure isolation and identification are much harder, and not always achievable. However, currently available theoretical results provide the capability to distinguish between sensor and actuator failures and to identify the failed component(s). Moreover, it is also possible to estimate the failure severity. The available FDI theory prescribes the conditions under which the identification of a failed component can be feasible, and points to the way of modifying, where possible, the measurement set in order to achieve this [25].

A model-based FDI method performance is only as good as the model itself. If the design model is not a perfect replica of the plant, then a difference in response(s)

is not necessarily due to a failure but could be due to a model mismatch; hence, failure detection is not feasible without an inherently high degree of a likely error (e.g. false alarm, missed detection, etc.). Consequently, even though the prime source of error is modelling inaccuracies, in addition sensor noise, and various exogenous disturbances in the operating environment, which are not included in the replica model.

A good FDI scheme has to be robust to such modelling inaccuracies and disturbances in order to be able to generate robust residuals resulting from component failures only, without taking into account modelling inaccuracies or disturbances acting on the plant. The objective of the thesis is to develop and apply state-of-the-art FDI techniques that are robust, simple to implement, can function reliably in real-time, on a continual basis, and which can be applicable to the GNC designs for autonomous spacecraft, particularly to small satellites and specifically to LPF.

Following a comprehensive literature review of the subject matter, initially two fundamentally different model-based approaches were formulated and pursued, reflective of the two main trends of the existing model-based methods: the geometric approach and the multiple model approach [4, 24]. The three techniques presented in this dissertation are the following:

- The Diagnosis Filter (Geometric Approach).
- The Euresis Filter (Multiple Model Approach).
- The Euphoria Filter (Geometric Approach - H_2 based Filter).

The novelty of the Diagnosis Filter (geometric approach) lies in that, in contrast to conventional methods, it does not only generate structured residuals, but it also employs directionality rather than the residuals magnitude as a diagnostic tool for faults (deterministic approach). It is shown that this filter exhibits remarkable robustness and performs exceptionally well, regardless of the thruster's failure severity. In comparison, the Euresis Filter (multiple model approach) exhibits similar robustness, albeit for total failures only. However, it has the advantage of being applicable to non-linear design models unlike the Diagnosis Filter. Subsequently, directionality is incorporated in another optimisation based geometric approach (i.e. Euphoria filter). In this case, directionality is not "designed-in" at the outset but it is rather recovered by an adjoint operation on the H_2 filter residuals (optimal stochastic approach). The Euphoria Filter outperforms the other two in most of the evaluation criteria and it can be applied in both linear and non-linear FDI scenarios.

Structure of the Thesis and Contribution

The thesis begins with a brief introduction (Chapter 1), providing the definitions of the main topics handled, and outlining the problems and challenges for a more effective fault identification which is the main objective of the research conducted. The resulting three newly developed algorithms derived by different methodologies, and the corresponding outcomes in the context of fault identification efficiency are briefly mentioned.

In Chapter 2, the used model system, i.e. LPF, is presented in more detail, regarding its technical engineering specifications and requirements. A multivariable analysis and the presentation of singular values plots for the better comprehension of the system dynamics are outlined. The thrusters' failures in their various types are then discussed, which along with the objectives of the Fault Identification systems are at the very core of the research of this thesis, setting the framework and the criteria for the generation of relevant algorithms that address the current inefficiencies.

A critical survey of the two main model-based FDI methods, i.e. the geometric and analytical approaches, follows in Chapter 3. Particular emphasis is given on the selection criteria for the particular architecture pursued in this thesis, the model-based FDI methods, with respect to its linear model availability, its capability for fault identification, its robustness and real-time function capabilities and its applicability on other space missions.

In Chapter 4, the three design methods that were ultimately generated, namely the Diagnosis, the Euresis and the Euphoria filters, are presented in detail with the theoretical background and implementation steps of each one. The theoretical foundation of each of the methods is discussed, and a fourth algorithm specific to failure severity is presented at the end of the chapter.

In Chapter 5, the above-mentioned FDI techniques are evaluated. First, the exact validation plan is explained in detail, along with the criteria of the performance evaluation, which is carried out with a special focus on sensitivity and robustness. After a comprehensive performance evaluation of all three algorithms against the set criteria, a further analysis of the Euphoria filter in particular with respect to Kalman Frequency Domain Equality (KFDE) and FDI timing is provided.

Chapter 6, summarises the main observations and conclusions drawn regarding the three FDI methods' performance and addresses the challenges and practicalities of their implementation.

The thesis concludes with an overview of the main findings of the research conducted, considerations, future directions stemming and applications of the current research in other fields and scientific contexts.

Chapter 2

System Characteristics, Design Requirements and Objectives

In this section, only the key system characteristics and requirements that have a direct impact on the FDI design are considered. The main objective of this dissertation was to design robust FDI algorithms for thruster failures. There are three main system loops around, for which, individual controllers with eight thrusters have been designed, functioning as the actuators of the system under study:

i. the Drag-Free loop (DF).

ii. the Attitude dynamics loop (ATT).

iii. the Suspension loop (SUS).

The drag-free and attitude control loops are strongly coupled, with the former serving as the inner loop due to its faster dynamics. There is also coupling of these two to the suspension loop but not vice versa [26, 27]. The controllers have been designed using the "loop-by-loop" approach, which is based on the assumption that no coupling between the individual loops occurs. However, this does not apply in all cases in the present system. As a result, the coupling between the loops compromises *a priori* such a controller design, as explicitly stated by Fertin in [27]: *"...the controllers designed by Astrium Gmbh for Drag-Free and Attitude Control System (DFACS) with Field Emission Electric Propulsion (FEEP) are not suitable for colloidal thrusters: there is too much rejection of thrusters' noise and not enough minimization of thrusters' step"*. Indeed, the thrusters' step or bias is a low frequency noise, exactly where the thrusters' characteristic frequencies are located, i.e. in the range $[10^{-10}, 10^{-14}] \mathrm{m\,s^{-2}\,Hz^{-1/2}}$.

On the other hand, the measurement noise(s) based on the LPF's specifications from ESA are of high frequency and reproduced with $[10^{-3}, 10]$ Hz values. The corresponding acceleration noises are in much lower frequencies and very close to the thrusters' characteristic ones. The system bandwidth appears to be at about 10^{-3} Hz, right where the measurement noises begin to emerge [27].

There is plenty of evidence that disturbances are not adequately rejected by the control system [27]. For example, for high frequency disturbance attenuation, the complementary sensitivities (closed loop transfer functions) should be considerably below 0 dB right before the frequency 10^{-3}dB, which is not the case. Respectively, for disturbance rejection, they should be much higher than 0 dB at the low frequency range.

A number of attempts to tune the controllers and improve upon these characteristics have been reported. However, the nature of the control design makes loop shaping a very tedious and time consuming and of questionable results process. In fact, there is no guarantee that even if individual loop constraints are met, the overall system behaviour will retain these characteristics in its integrated multi-loop function. Because of these difficulties, "only drag-free and suspension measurements are selected for FDI because of their low noise characteristics" according to Fertin [27]. On the other hand, a truly multivariable design allows for overall loop shaping with guaranteed performance and robustness. Most importantly, it can introduce automatically and by construction the plant (actuator) redundancies [28] and make the system fault tolerant to multiple actuator failures. Notwithstanding the above comment, the controller design is now fixed and FDI has to be designed around this. However, even with the existing design, some disturbances will inevitably occur both at the low and at the high frequency ends. For this reason, thresholds need to be accurately set and the simulation environment has to reach, firstly, steady-state (2×10^4s) condition and then to initialize the Unknown Input Observers (UIO) (3×10^4s), before the application of the FDI algorithm [29].

A class of alternative model-based solutions that would be more likely to overcome the shortcomings identified earlier, should realize a separation principle between robust residual generation and robust failure detection. This robust residual generation could be (a) the result of appropriate signal processing on the plant input(s) and output(s) in order to realize exogenously desirable loop shape characteristics before they are employed in an FDI scheme and (b) the utilization of appropriate filters, which optimally reject plant disturbances.

2.1 LPF: System Engineering Characteristics and Requirements

2.1.1 Multivariable Analysis of the Overall System Characteristics

The overall LPF design model in its current form, is presented in the diagram of Figure 2.1. The blue rectangles represent on-board functions (controllers and thrusters dispatching) and the light blue rectangles are the dynamic transfer functions of the drag-free, suspension and attitude degrees of freedom.

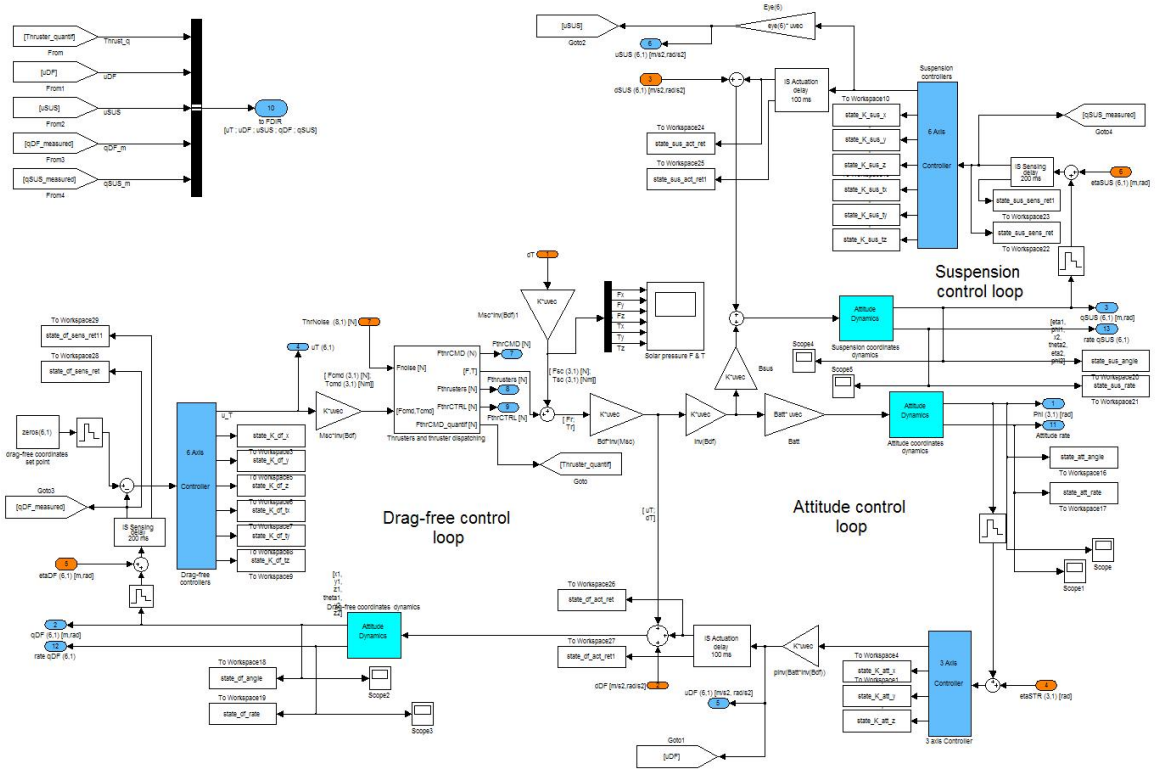


Fig. 2.1 LPF Design Model (Control Loops) (taken from simulation environment)

The LPF design model comprises the drag-free attitude dynamics and the suspension loops for which corresponding controllers have been separately designed, assuming non-interaction between the loop dynamics (Figure 2.2). The eight thrusters in the drag-free loop serve as actuators. This is the system for which robust FDI algorithms are to be designed, such that they are non-intrusive and do not interfere with the rest of the system function. Therefore, FDI has to function independently as a plug-in on the LPF.

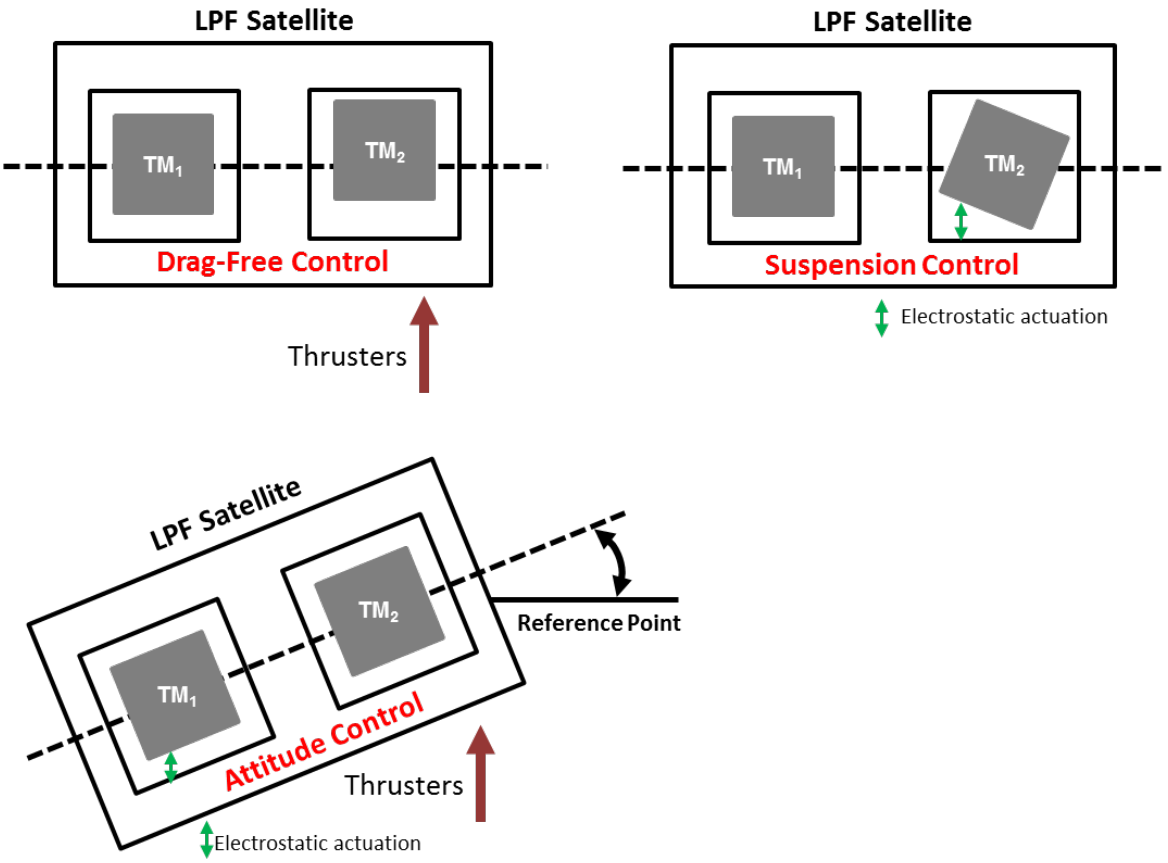


Fig. 2.2 DF, SUS, and ATT Control Principle - adapted from [2]

As mentioned earlier, the DF and ATT control loops are strongly coupled. Because of this non-ideality in the controller design, it is possible that the interaction between the loops can exacerbate the disturbance effects [30]. Any corrections or modifications to improve each individual loop performance would have an effect on the others. Even though this effect is observable, it can not be handled directly.

The process of loop-characteristics-improvement is, therefore, a purely trial-and-error-approach, with no predictable or guaranteed results; this can also be garnered from the descriptions in Fertin [27] and Valavani [26], where an improvement in a particular loop characteristic can adversely affect another loop. This shortcoming is commonly manifested in designs of this type and is not specific to the LPF system.

Previous works on LPF [15, 27, 29] did not include an overall multivariable picture of the entire plant as there is no mention of multivariable crossover frequencies (max and min), or overall multivariable system gains (max and min). Therefore, no matter how accurately the disturbances are known or estimated, it would still be not

possible to predict their overall impact on the multivariable system, how effectively its controllers would reject them, and what would remain for threshold setting. This might be the reason why disturbance identification was proposed in previous works, which would not have been required in a truly multivariable analysis based on the Singular Value Decomposition (SVD). The latter was carried out at the very start of this project [31, 32] in order to accurately establish the controlled system boundaries.

In the absence of an SVD, no conclusion can be drawn on whether and how the system inputs and outputs need to be further processed to provide information to the FDI scheme, since there is no accurate description of the multivariable disturbance rejection properties of the plant. Even if the controller design is carried out loop by loop, the system still has to function as an integral multivariable entity. These non-idealities have an impact on the overall system ability to process disturbances and sensor noise, which, in turn, influences the FDI design framework and places limitations on the achievable performance. Despite the above, the controller design is now considered to be fixed and FDI can be developed around this premise.

The singular value analysis we perform on the overall design model, complete with controllers, and is presented in section 2.1.3, clearly corroborates the observations previously reported by other researchers [15, 27, 29] in performing FDI on the LPF system and highlights potential problems. Our findings [31, 32] then make it possible to rectify the system shortcomings in terms of noise and disturbance rejection, and to avoid tedious and time consuming procedures of online noise estimation for threshold setting. Taken together, a sharp loop gain attenuation (i.e. a roll-off filter) is necessary around $\omega = 10^{-3}$Hz, preferably starting at the middle of the system bandwidth $[10^{-4}, 10^{-3}]$Hz, in order to avoid sharp effects and fully exploit the system capabilities. Although the need for such a roll-off could have been envisaged (based on some of the individual transfer functions between specific input-output pairs in the two loops), it does not necessarily apply to all such input-output pairs and, definitely, not for the overall system. The exact nature and characteristics of the roll-off may differ for different input-output pairs. The fact that this roll-off emerges for the entire system and is easily quantifiable as a one design parameter makes it possible to remedy the existing design for FDI purposes. As a consequence, all the input signals to the FDI methods need to be processed through a pre-processing filter designed according to the following characteristics: $G_{pre\_filter} - \frac{1}{s(s+0.5\times10^{-3})}$, which is incorporated in the simulation environment. However, given the sheer amount of noise that can go through such a loop, even if the thresholds are accurately set on a magnitude scale, this does not preclude

the possibility that the entire magnitude of such a threshold is solely due to noise and not due to a failure. Consequently, the possibility of missed or late detection is very distinct. Therefore, additional signal processing is imperative for the signals that will trigger the FDI process. In addition, methods where FDI is predicated on residuals magnitude rather than on direction, can be particularly sensitive to the pre-mentioned effects. A notable exception is the following. Since the Diagnosis Filter and the Euphoria Filter (H_2 based Filter) use residual directions rather than magnitudes as FDI diagnostics, they are practically immune to such effects and, therefore, they do not require any additional signal conditioning, simplifying implementation.

A simple observation of the combined, DF and SUS open loop singular value analysis reveals eight significant singular values, given that the effective transfer function G_{df_sus} is a (11×8) matrix. Two of these are separated from the others at the -200 dB levels. This suggests a system structure singularity, which also affects the Diagnosis Filter's structure. The eight thruster failure directions were identified through linear independent evaluation, that were in a later step separated into two output separable-groups (i.e. sets of linearly independent vectors/directions): one with six of these directions and the other with two [27].

In summary, regardless of the control design methodology, the overall plant remains a truly multivariable system. It has certain overall characteristics, in terms of its command-following and disturbance-rejection properties, which need to be identified, before any FDI scheme is successfully implemented. It should be noted that the FDI scheme must be totally non-intrusive to the controller design and that the overall system characteristics (the controllers included) are accurately represented.

2.1.2 Specifications and Requirements

A careful examination of Fertin's work [27] and the simulation environment model reveals that the overall system operating bandwidth of interest, lies in the interval $[10^{-4}, 10^{-3}]$ Hz, while the most significant disturbances (process and sensor) begin to occur at $\omega = 10^{-3}$ Hz and extend further within the high frequency range. It is also evident, on most individual transfer function plots, that the complementary sensitivity function continues to have a magnitude close to or larger than one even beyond this frequency, where the sensor noise is mostly pronounced. An effective loop shape should have a complementary sensitivity magnitude below one in this range, to attenuate, instead of amplify, sensor noise. This is not the case in the present design.

Fertin [27] shows that the thruster-induced-acceleration noise on the drag-free coordinates extends in the interval $[10^{-4}, 10^{0}]$ $\mathrm{m\,s^{-2}\,Hz^{-1/2}}$, with a much larger magnitude in the frequency range between $[10^{-4}, 10^{-3}]$ $\mathrm{m\,s^{-2}\,Hz^{-1/2}}$. Because of this direct conflict with the system operating bandwidth, the (higher) sensor noise can be regarded as consisting of two parts. One that is identical in function to low-frequency-disturbances, requiring a high enough gain in the range $[10^{-4}, 10^{-3}]$ $\mathrm{m\,s^{-2}\,Hz^{-1/2}}$ for good disturbance rejection and a second high frequency part (starting from 10^{-3}), compatible with a sensor noise profile requiring low magnitude of the closed loop transfer function/complementary sensitivity.

These tight and rather conflicting requirements (for purposes of best noise rejection) require that the complementary sensitivity's roll-off is very fast, occurring at around 10^{-3}, while retaining its high magnitude up to almost that point. According to the satellite's specifications, this roll-off does not occur and, even worse, in the majority of cases the magnitude is retained or increased beyond this frequency. This shortcoming is bound to impact the robustness of the FDI process, as the signals used will contain a significant contribution of the non-attenuated noises, which could confuse FDI, and trigger false alarms. If residual magnitudes are used as the diagnostic, this can significantly impair the actual FDI performance. Equally, an online noise-identification process would not necessarily provide the solution, as this objective is hampered by an inherently noisy signal. The result is a much longer time to noise estimation, with less accuracy, jeopardizing system performance in case of a failure, which cannot be promptly detected for effective corrective action.

On the other hand, the individual elements of the transfer function matrix of the overall multivariable system, show a possible gain inadequacy in the low frequency region, compromising the system's ability to reject disturbances and attain good command, following Fertin [27].

As exemplified in Fertin [27], the characteristics of each individual transfer function, in each loop separately, do not necessarily apply to the entire system (which is truly multivariable), despite the fact that the control design is carried out loop by loop. Consequently, in order to pinpoint the true multivariable characteristics governing the system behaviour, a SVD analysis was carried out to accurately capture the actual system characteristics.

2.1.3 The Singular Values Analysis and State Space Characteristics

In order to calculate the system singular values for various configurations, a state space description of each is required. This will also be needed for the Model-based FDI algorithms to be designed for the system.

There are three sub loops, the drag-free loop, where the thrusters are located, and which is, therefore, a central element, the suspension loop, and the attitude loop. The corresponding open loop transfer functions are given as G_{DF}, G_{SUS} and G_{SC_c}. Similarly, the controller transfer functions are specified as K_{DF}, K_{SUS}, and K_{ATT}, for each loop respectively.

Firstly, the state space description of each loop without the compensator (open loop) needs to be defined in order for the frequency domain of the natural open loop characteristics to be determined. Subsequently, the state space description of the compensator in each of the three loops is required for the calculation of the loop gain transfer function/matrix. The loop gain is then derived from the product of $G_i(s)$ and $K_i(s)$, where $K_i(s)$ is the compensator transfer function and the open loop plant transfer function $G_i(s)$. These in turn are calculated as follows:

$$\begin{aligned} K_i(s) &= C_{ki}[sI - A_{ki}]^{-1}B_{ki} \\ G_i(s) &= C_i[sI - A_{ki}]^{-1}B_i \end{aligned} \tag{2.1}$$

where the index i, can signify any of the three loops: drag-free, suspension, attitude, or the combined drag-free and suspension loops and the additional index k the corresponding compensator quantities. The matrices $A_i, A_{ki}, B_i, B_{ki}, C_i, C_{ki}$ represent the canonical triplet for the plant and compensator respectively in their sate space description. The state space descriptions for the open loop dynamics of each of the loops are derived in two ways:

1. By getting the state space description that corresponds to the transfer functions G_{DF}, G_{SUS} and G_{SC_c}, as given in Fertin [27] and used by provided simulation model by ESA. These are denoted with the $'_{AST'}$ index additionally attached to the corresponding state space element.

2. Using the Linearize Mode in the simulation environment, which is denoted with the $'_{LINMOD'}$ index attached to the corresponding state space element. Apparently, the simulation environment derives these linearized models from

the overall system information, including implementation information, such as the sampling time. Consequently, the resulting state space descriptions differ in this regard. For example, while the compensators have been designed based on continuous plant dynamics, they need to be discretised in the simulation environment before implementation. Therefore, the compensators directly derived from the linearization of the model ('LINMOD'[1]) differ in order from their corresponding state space descriptions generated from their transfer functions.

The transfer function derived representations were used for the loop gain calculations, as it is always preferable to carry out the design process on a continuous time basis and then to discretize for implementation. This offers the flexibility of choosing different sampling times, right before implementation, without the need to repeat the design process itself, as would have been in the case of a discrete time design approach applied at the outset.

The state space descriptions for the attitude loop, the drag-free loop, the suspension loop and the drag-free and suspension loops combined are given in Appendix A. In each of the three (sub) loops, we first give the state space description for the open loop and the compensator as derived from the Astrium transfer function and then the state space description of the open loop dynamics as derived via the simulation environment 'LINMOD'. For the drag-free and suspension loops combination, the state space description is as derived from the simulation environment 'LINMOD', since there is no combined transfer function description given in the input from Astrium.

2.1.4 Singular Value Plots

The singular values for the DF open loop are shown in Figure 2.3 and Figure 2.4, first as derived from the Astrium Transfer Function (TF) and then as generated from the 'LINMOD' function of the simulation environment. In the first case, all the six values coincide, as they are essentially six separate double integrators. However, in the second case, even though the maximum singular value levels are the same as before, at 160 dB, there is a split at the lower frequency end of the spectrum, slightly below 120 dB for the minimum singular value. This can be attributed to round-off errors due to the discretization at implementation.

[1] 'LINMOD': the function of the simulation environment used to extract linear models in the form of the state-space matrices A, B, C, and D.

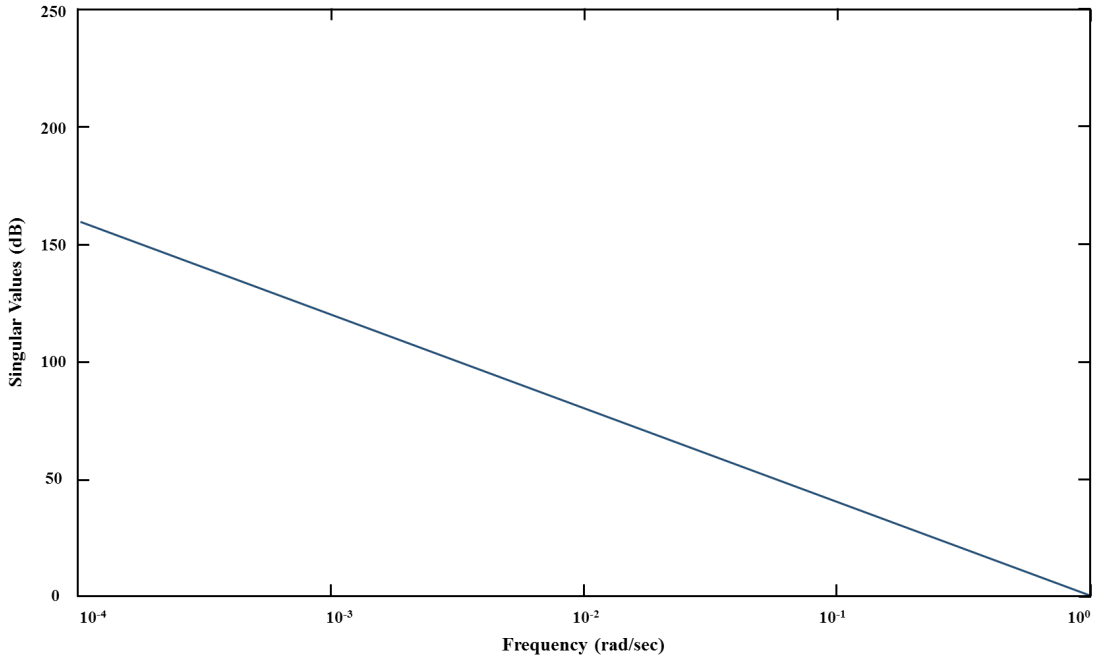


Fig. 2.3 Singular values of the DF open loop (Astrium)

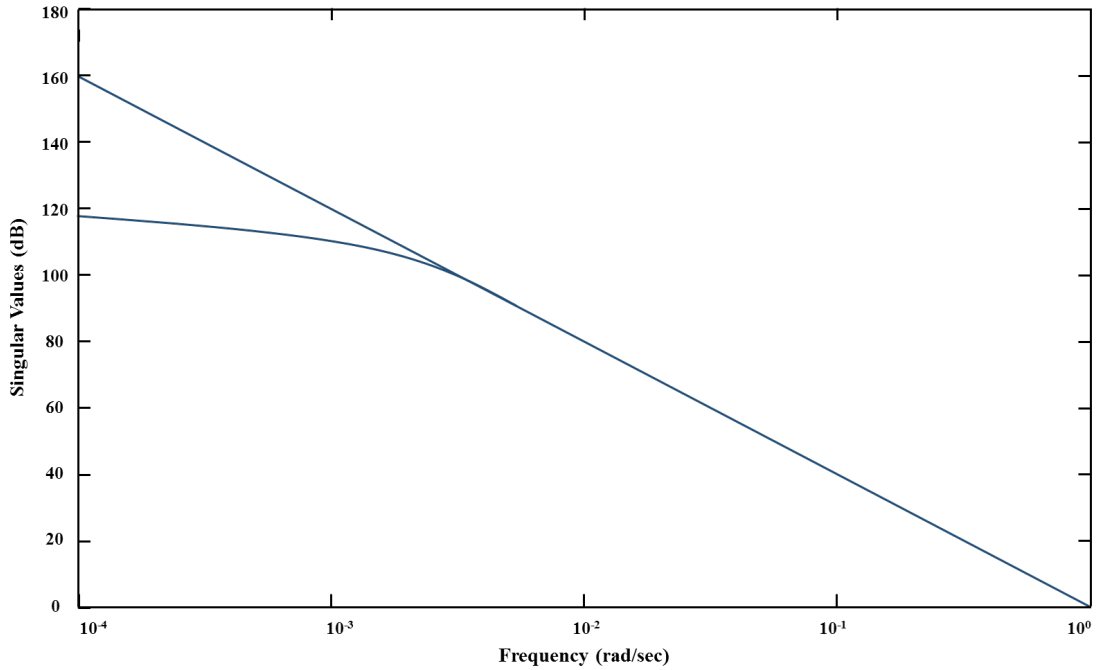


Fig. 2.4 Singular values of the DF open loop (simulation environment)

The open loop has high gains at low frequencies, which is a desirable characteristic; this high gain profile extends to the lower frequencies, 10^{-3} Hz and above, which can give rise to problems for measurements that have been corrupted by sensor noise. Figure 2.5 shows the drag-free compensator singular values while the corresponding loop gains are presented in Figure 2.6 and Figure 2.7; the values were derived from the continuous time quantities for open loop and compensator dynamics while the loop gains were generated from the respective 'LINMOD' discretized system.

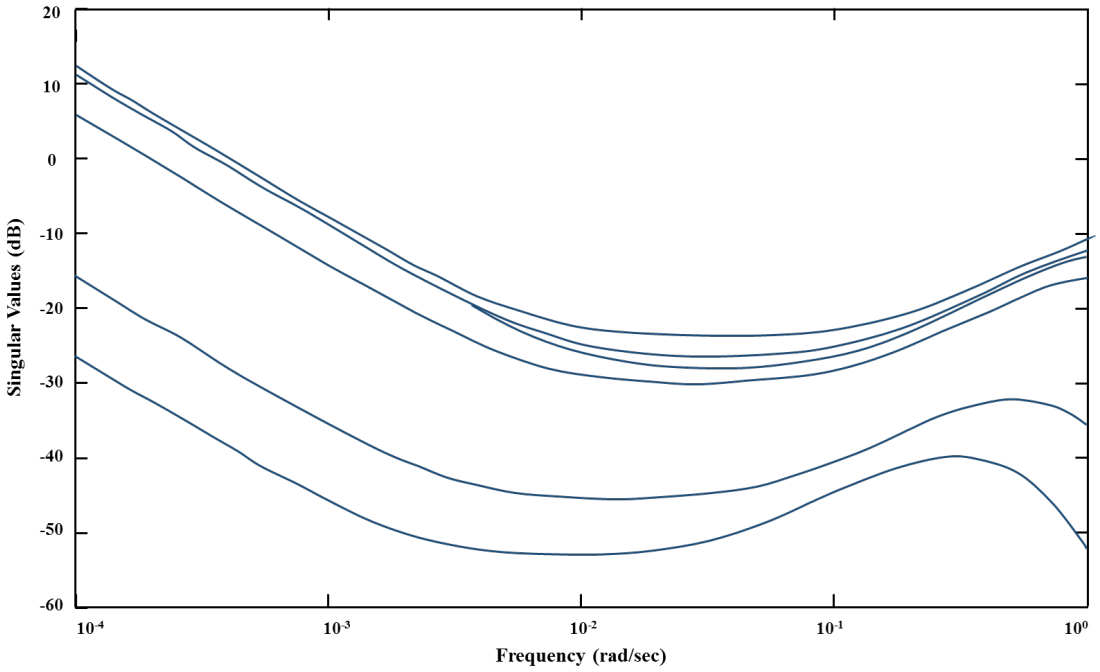


Fig. 2.5 Singular Values of the DF Compensator (Astrium)

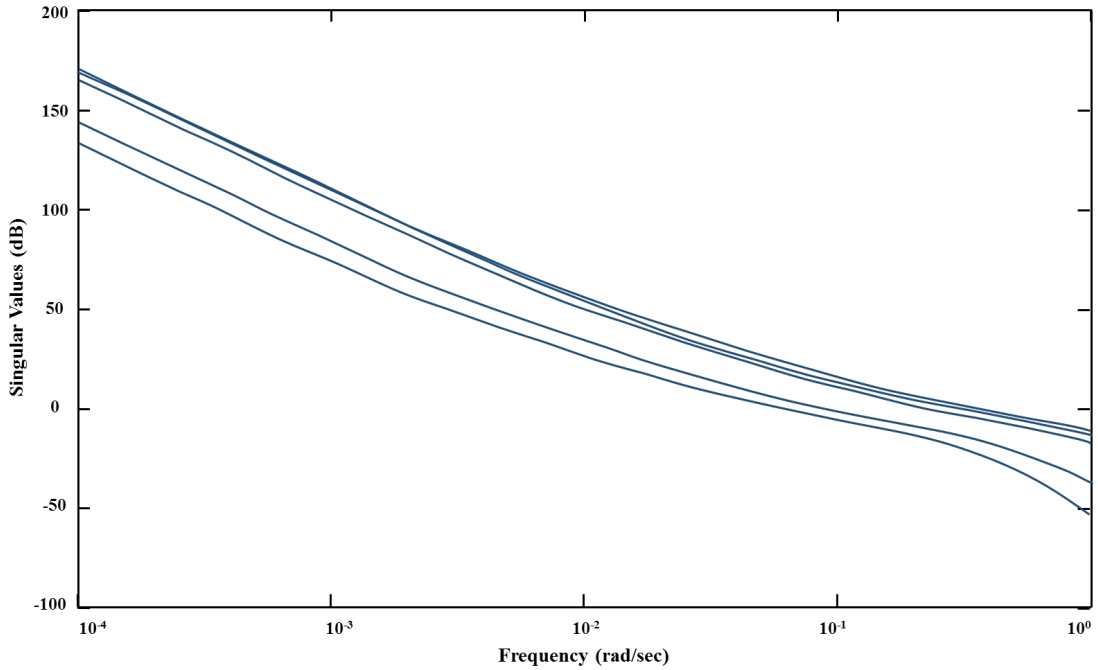


Fig. 2.6 The DF loop gain (Astrium)

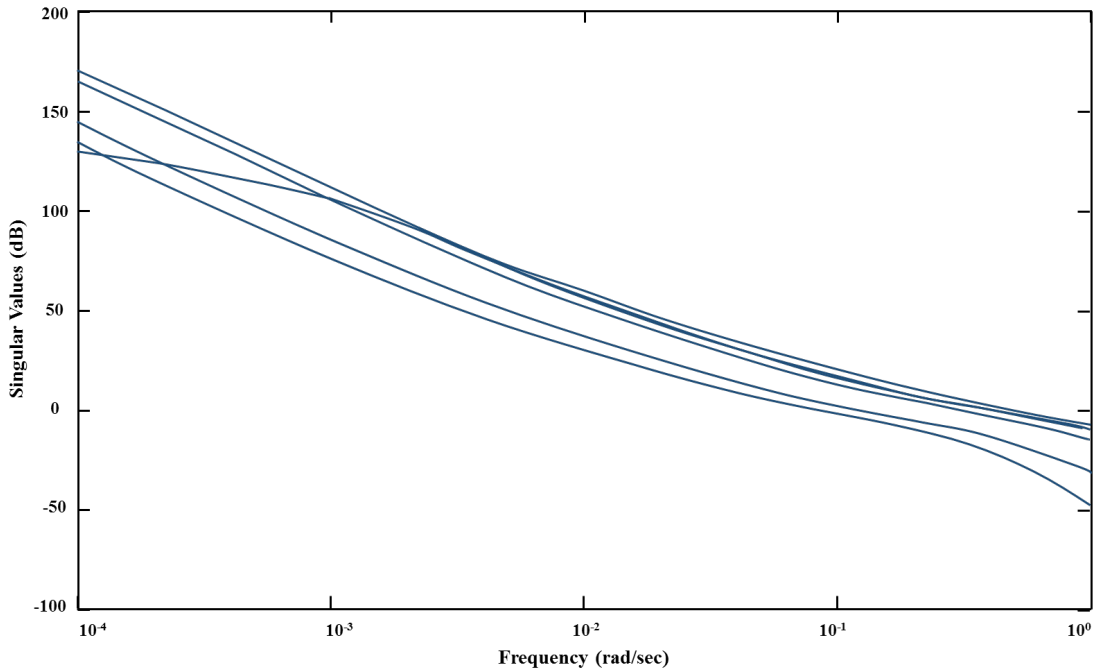


Fig. 2.7 The DF loop gain (simulation environment)

The similarity between Figure 2.6 and Figure 2.7 in terms of operating loop gain envelope (i.e. the gain profiles defined by the maximum and minimum singular values) is striking; the fact that the trajectories of the two different DF loops in the plots above are slightly different is inconsequential as far as the overall system characteristics relevant to performance and robustness are concerned. This further corroborates the fact that the implementation was carried out correctly and the system functions as it should with a sampling rate of $T_s = 1\,\mathrm{s}$, which is roughly two orders of magnitude larger than the system time constants. With this sampling rate, the discretized system is effectively an alternative continuous rendition of the original. Additionally, the loop gains remain very high at the higher frequencies despite the presence of the compensator; this is probably the reason why in the study by Patton [15]) an additional disturbance estimation was needed for threshold setting.

Figure 2.8 and Figure 2.9 feature the singular values of the Suspension (SUS) open loop dynamics from the Astrium TF and 'LINMOD'. It is interesting to note that in the TF case, the six singular values split into three in the lower frequencies, ranging from slightly below 120 dB to just above 100 dB, while in the 'LINMOD' case a similar split occurs between 160 db and about 105 db; probably attributable to the same reasons as in the DF case. However, in the 'LINMOD' case, the split is similar to the corresponding DF case, which could mean that it is attributable to similar round off or modelling approximation factors. Even though the lower magnitude bounds show a good agreement at about slightly above 100 dB, the upper bounds show a considerable departure from the 160 dB in the first case to just a little below 120 dB in the second. Again, in this range of frequencies, unlike the lower bound, the upper bound is rather inconsequential for the design.

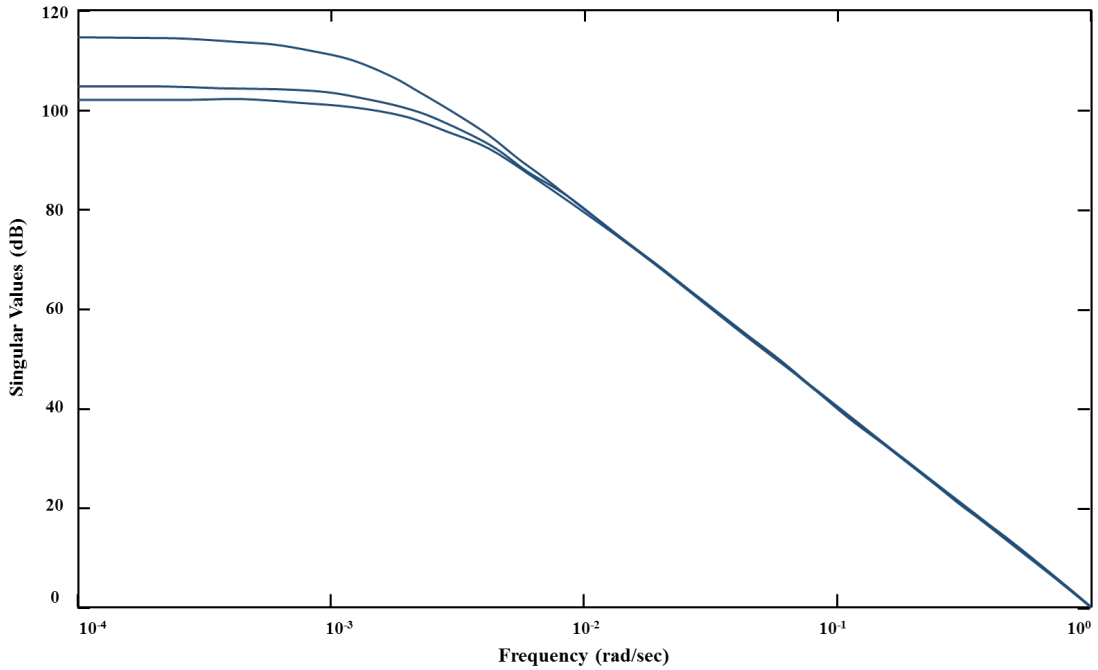


Fig. 2.8 Singular Values of the SUS open loop (Astrium)

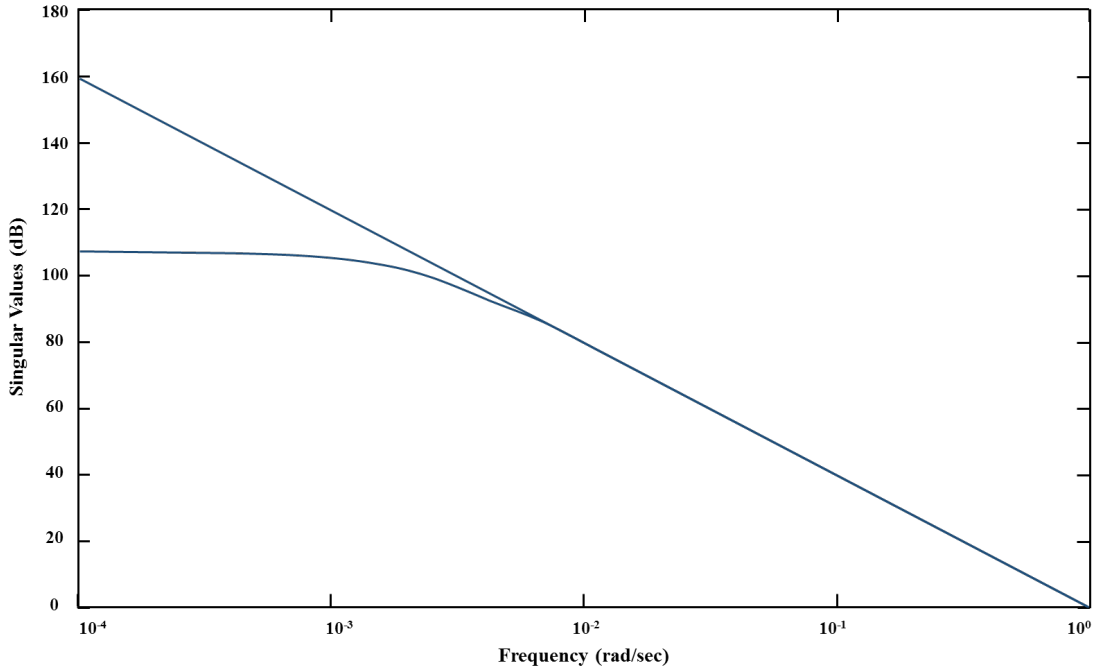


Fig. 2.9 Singular Values of the SUS open loop (simulation environment)

The compensator singular values for the SUS loop as well as the two loop gains, the Astrium TF-based gain and the 'LINMOD' gain, are presented in Figure 2.10, Figure 2.11 and Figure 2.12, respectively.

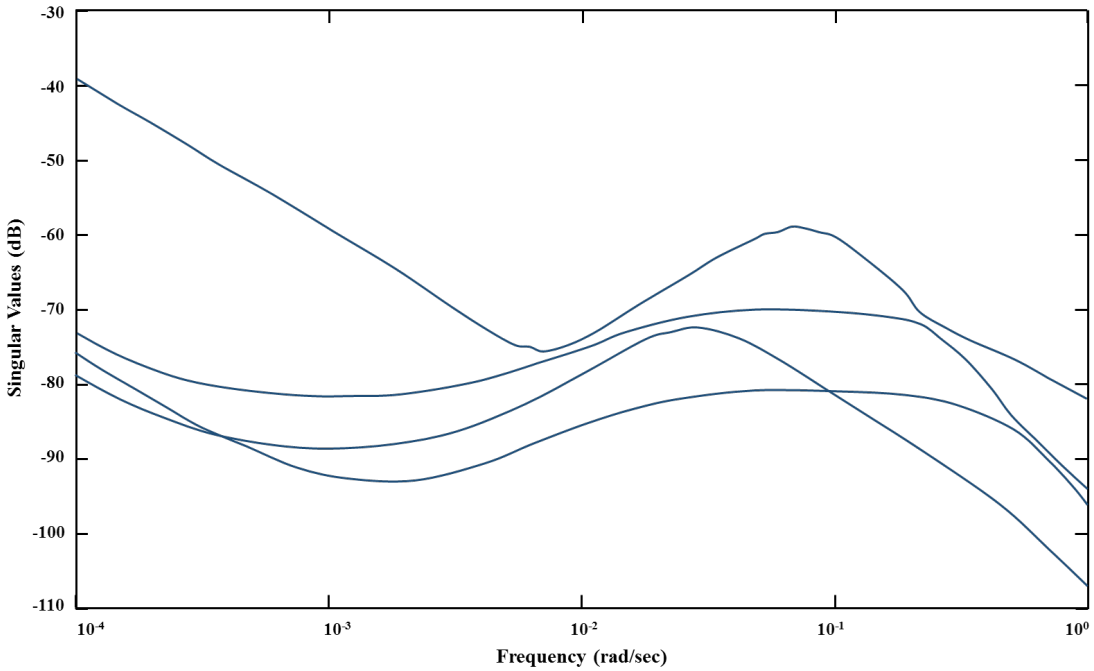


Fig. 2.10 Singular Values of the SUS Loop Compensator (Astrium)

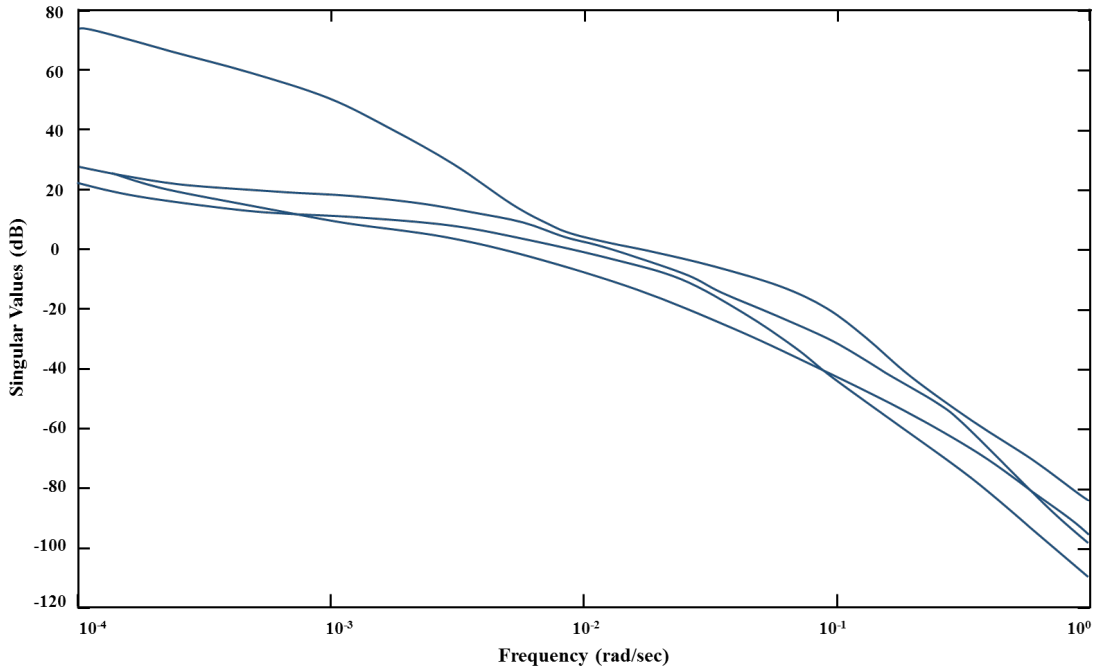


Fig. 2.11 Singular Values of the SUS Loop Gain (Astrium)

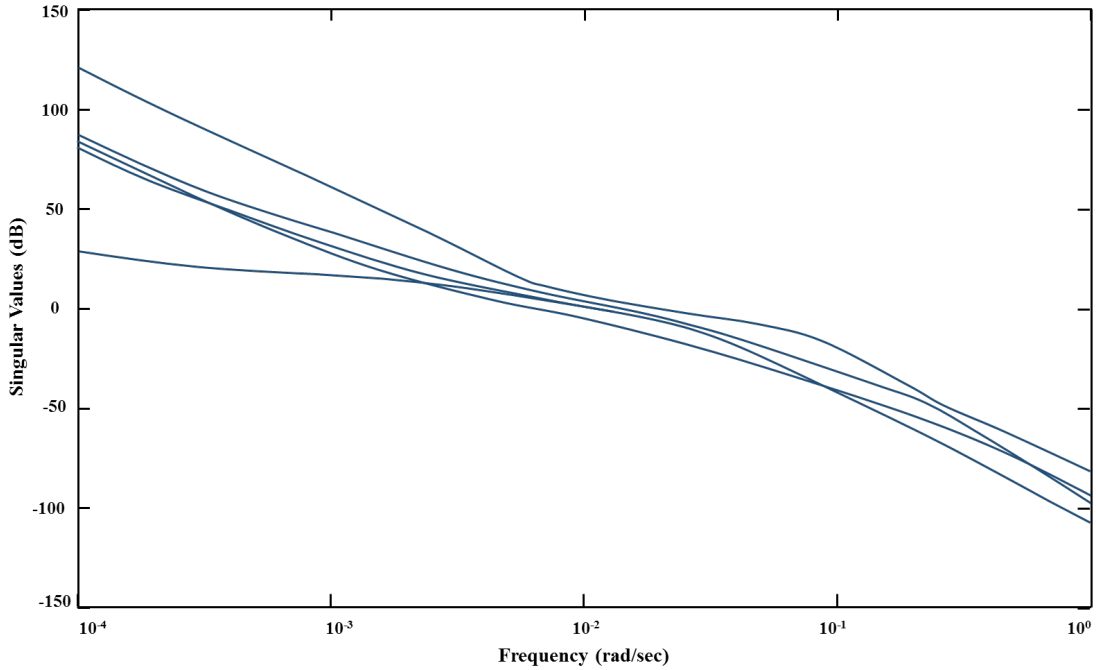


Fig. 2.12 Singular Values of SUS Loop Gain (simulation environment)

Marked differences in shape and magnitude are again observed on the upper bound side, while the lower bound magnitudes show a relatively good agreement, both at about 25 dB at $\omega = 10^{-4}$ rad/s and at the same levels (about 18 dB) at rad/s. However, as in the DF loop gain case, even though the open loop profile is improved (reduced) by almost four orders of magnitude in the higher frequencies, it is still not adequate for noise rejection, as it remains well above 0 dB, at about 18 dB. The open loop singular values for the Attitude loop, via TF and 'LINMOD' are shown in Figure 2.13 and Figure 2.14, respectively.

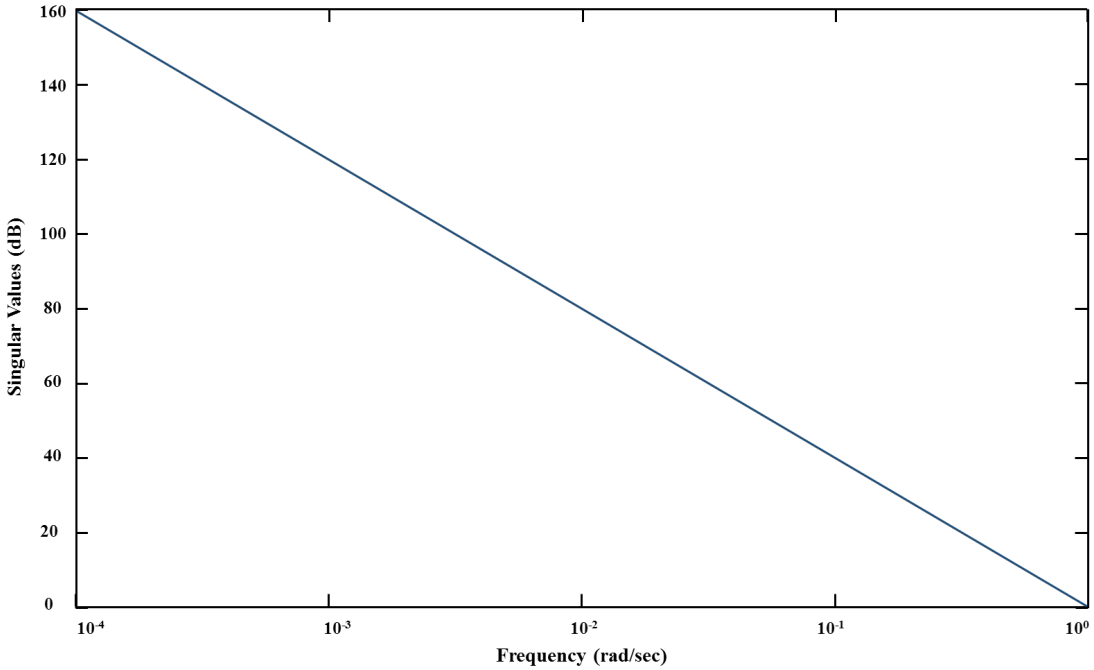


Fig. 2.13 Singular Values of the ATT Open Loop (Astrium)

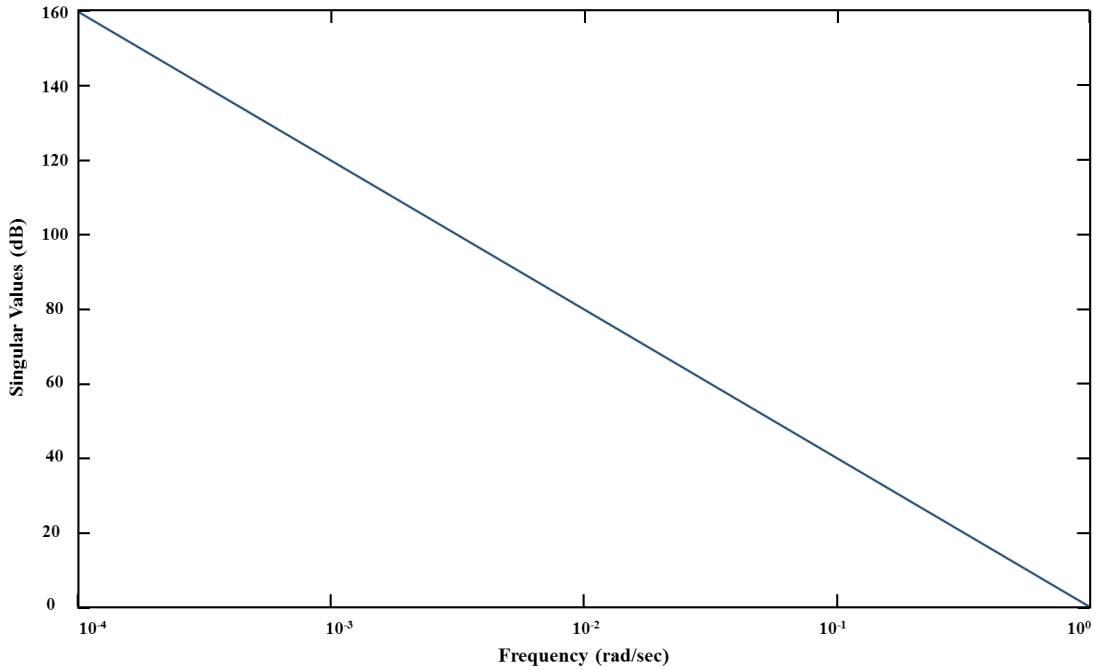


Fig. 2.14 Singular Values of the ATT Open Loop (simulation environment)

It is striking that both sets of the (three) singular values are identical possibly due to the fact that the ATT loop may be less prone to computational round-offs and modelling approximations. Figure 2.15, Figure 2.16 and Figure 2.17 present the singular values of the compensator and the attitude loop gains of the Astrium and 'LINMOD' environments, respectively.

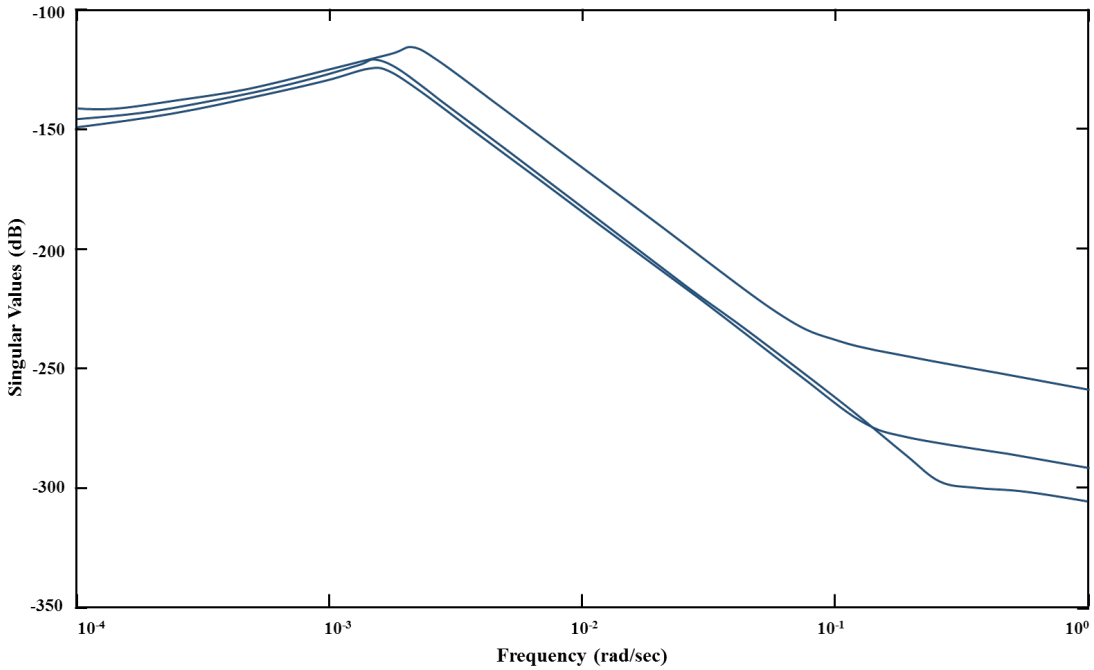


Fig. 2.15 Singular Values of the ATT Loop Compensator (Astrium)

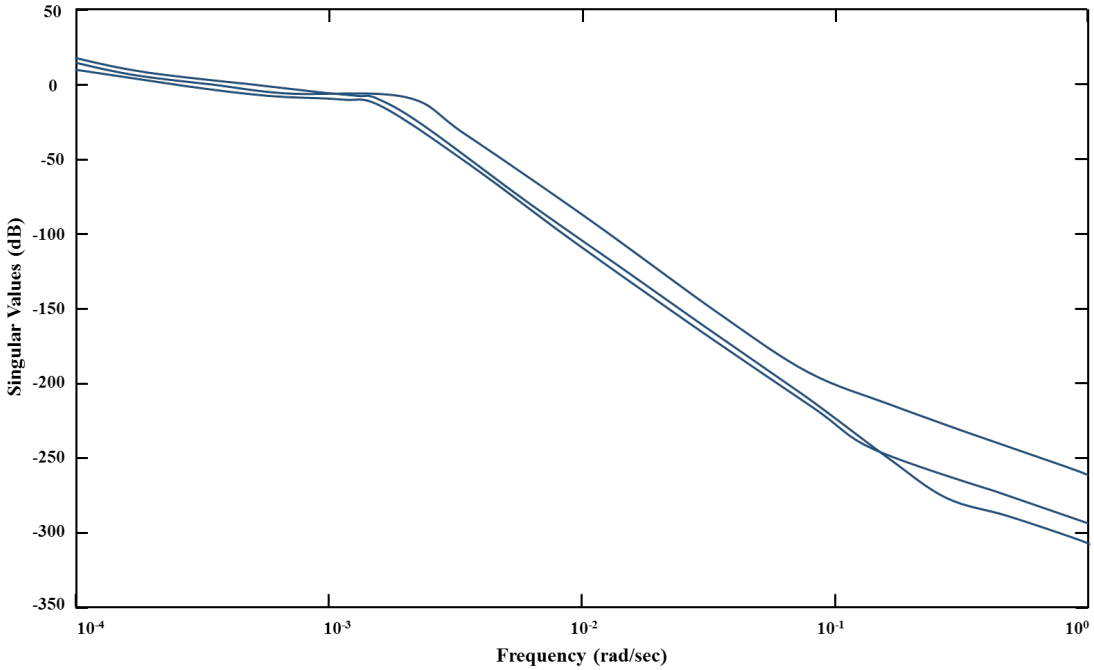


Fig. 2.16 Singular Values of the ATT Loop Gain (Astrium)

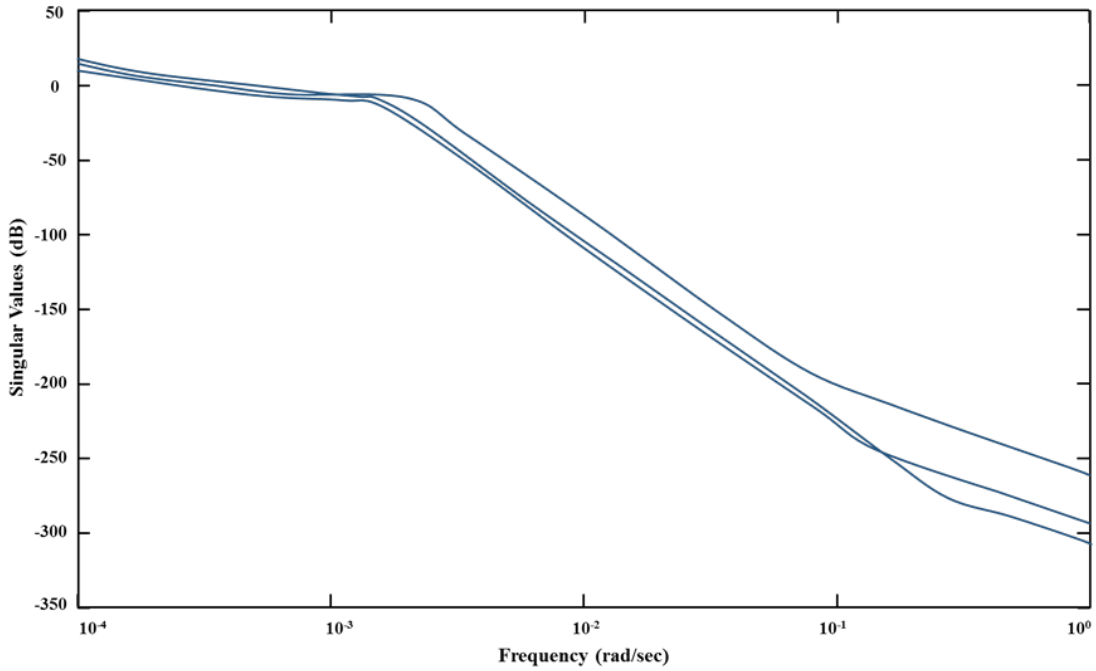


Fig. 2.17 Singular Values of the ATT Loop Gain (simulation environment)

The loop gains are identical in both the TF and 'LINMOD' case. However, they also show control loop inadequacy: they are extremely low (well below 20 dB) at the low frequencies, where they should be high for adequate disturbance rejection.

In addition, they are still high (as high as the SUS loop gains at about 20 dB) at the higher frequencies, where they should be low for measurement noise attenuation. This is the reason why the ATT loop measurements were previously discarded as too noisy [29], and only the DF and suspension measurements were retained.

Indeed, the ATT measurements carry a considerable amount of the low frequency disturbances, which can not be rejected by the control loop. Again, the ATT open loop singular values all coincide at 160 dB in a double integrator profile (-40 dB/decade), while for the other two, only the maximum singular value retains this profile in the 'LINMOD' case. The double integrator profile in the open loop is as expected from the modelling details in [27].

However, the minimum singular value shapes of the DF and SUS loops in 'LINMOD' are indicative not just of round-off errors, but also of modelling approximations. These may be precisely the cross coupling, on the one hand, between the DF and ATT loop dynamics and on the other hand, of both of them with the SUS loop. Either of the two (DF and SUS loop) does not affect the ATT loop. Despite their

shortcomings in performance and robustness, the compensators should stabilize all three loops, which are inherently unstable. This is particularly important in the design of the FDI process.

Lastly, the combined DF-SUS loop singular values are presented. Measurements from these two loops are used during detection since the effective transfer function G_{df_sus} is a (11×8) matrix; eight significant singular values exist as shown in Figure 2.18. Two of the eight values are separated from the others at -200 dB levels. This suggests a system structure singularity which also affects the FDI structure, as discussed in the following section.

It is indicative that the eight thruster failure directions are separated into two output-separable groups (i.e. sets of linearly independent vectors/directions): one with six of these directions and the other with two.

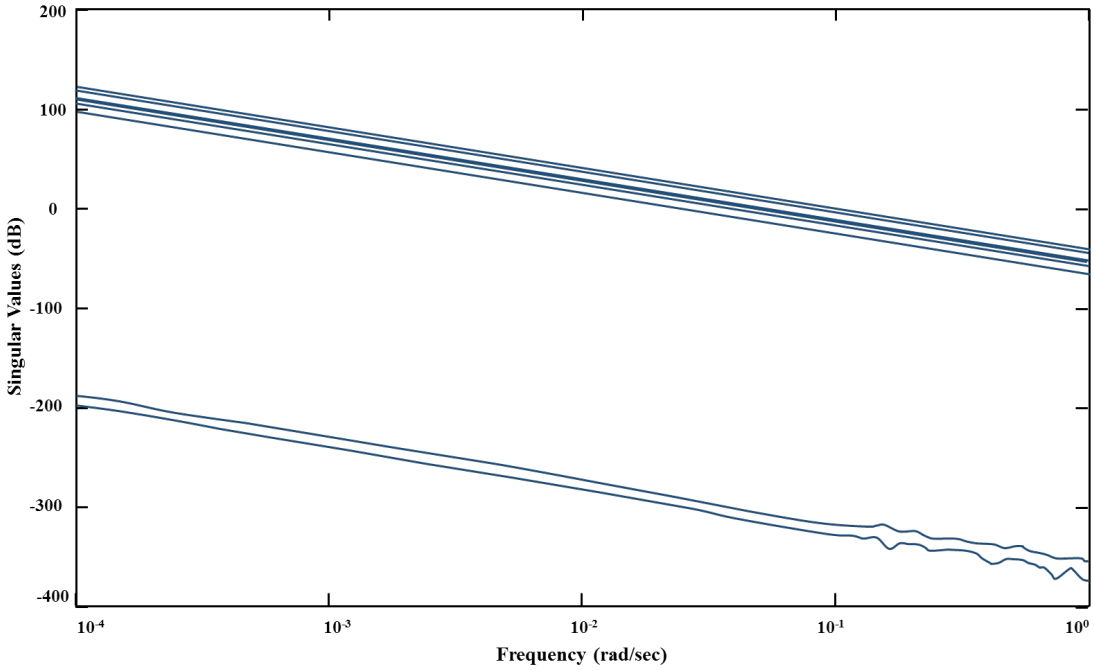


Fig. 2.18 Singular Values of the DF and SUS loops

2.1.5 Controllers, Thruster Failures and Stabilization

As a final step in the pre-design analysis, the compensator and corresponding closed loop eigenvalues for each subsystem were considered. All compensators show some marginally stable to slightly unstable eigenvalues; the same holds true for the corresponding closed loop subsystems. This means, that the compensators do not stabilize each loop separately. The eigenvalues of the three loops together, complete with compensators (hence the overall closed loop system), also show marginal instabilities. This is not surprising. In fact, even if each compensator alone stabilized its corresponding loop, this does not imply that all three together will stabilize the overall system. This can be explained by the fact that the loop-by-loop design is based on the assumption of loop independence, which is only an approximation for the purposes of this particular design method, and this assumption does not actually hold true for the real system as, at best, small cross feeds and cross correlations between loops tend to be ignored. These approximations, however, may compromise the stability of the nominal design as well as the overall system robustness.

Marginal instabilities may not necessarily be detrimental to the system function, even though the performance itself may be compromised. As a trade-off, the instability does provide enhanced agility and leverage for the system to respond faster to a command, a property which is useful in cases of frequent manoeuvres and set point changes. On the other hand, extra caution is required to monitor and set the (marginally unstable) controller initial conditions very accurately for a smooth system function, as one cannot count on a fast convergence to the values of the controller state that are compatible with the system state at any particular time, when the control kicks in. However, the FDI process is not affected, as the gain matrix D in the Diagnosis Filter case and the Kalman Filter gain matrix H in the Euresis Filter stabilize the residual dynamics.

Another interesting fact is that any thruster failure can destabilize an otherwise stable closed loop. The loop-by-loop closure is much more prone to this, as each loop has to rely on its own control system, and it cannot use any leverage from the others that might have enough slack to compensate for the particular failure. This is precisely where a truly multivariable control design has a distinct advantage as it can leverage the overall system capabilities and make use of any actuation redundancies to compensate for functional impairments. In fact, the simulation environment model behaviour [29] corroborates this fact: as soon as a (thruster) failure occurs (or is injected for experimental purposes), the rest of the (thrusters) control inputs

max out to almost reach saturation levels, with the FDI process functioning properly, nonetheless.

2.2 FDI Objectives

The primary objective of this thesis is to design robust failure detection algorithms for the thruster (actuator) failures using model-based schemes, which employ a dynamic detection model. If an actuator fails, then its effect can be observed by an anomaly in the vehicle dynamics, which in turn are observed via sensor measurements.

If one allows both sensor and actuator failures, the task of failure identification and isolation becomes increasingly more complex. However, the advanced detection theory, as described in Chapter 3 and in the theoretical foundation of the FDI techniques of Chapter 4, can handle both these eventualities at the expense of additional computation. For the purpose of this thesis, it is assumed that only actuator failures occur, while the sensors' function remains normal at all times.

In the LPF reference application; there are eight thrusters, in the typical configuration of two separate branches of four thrusters each. The thrusters are arranged to control the high-level disturbing torque of the main engine; only four thrusters are used in a nominal situation.

The structural diagram of the LPF FDI generic concept is shown in Figure 2.19.

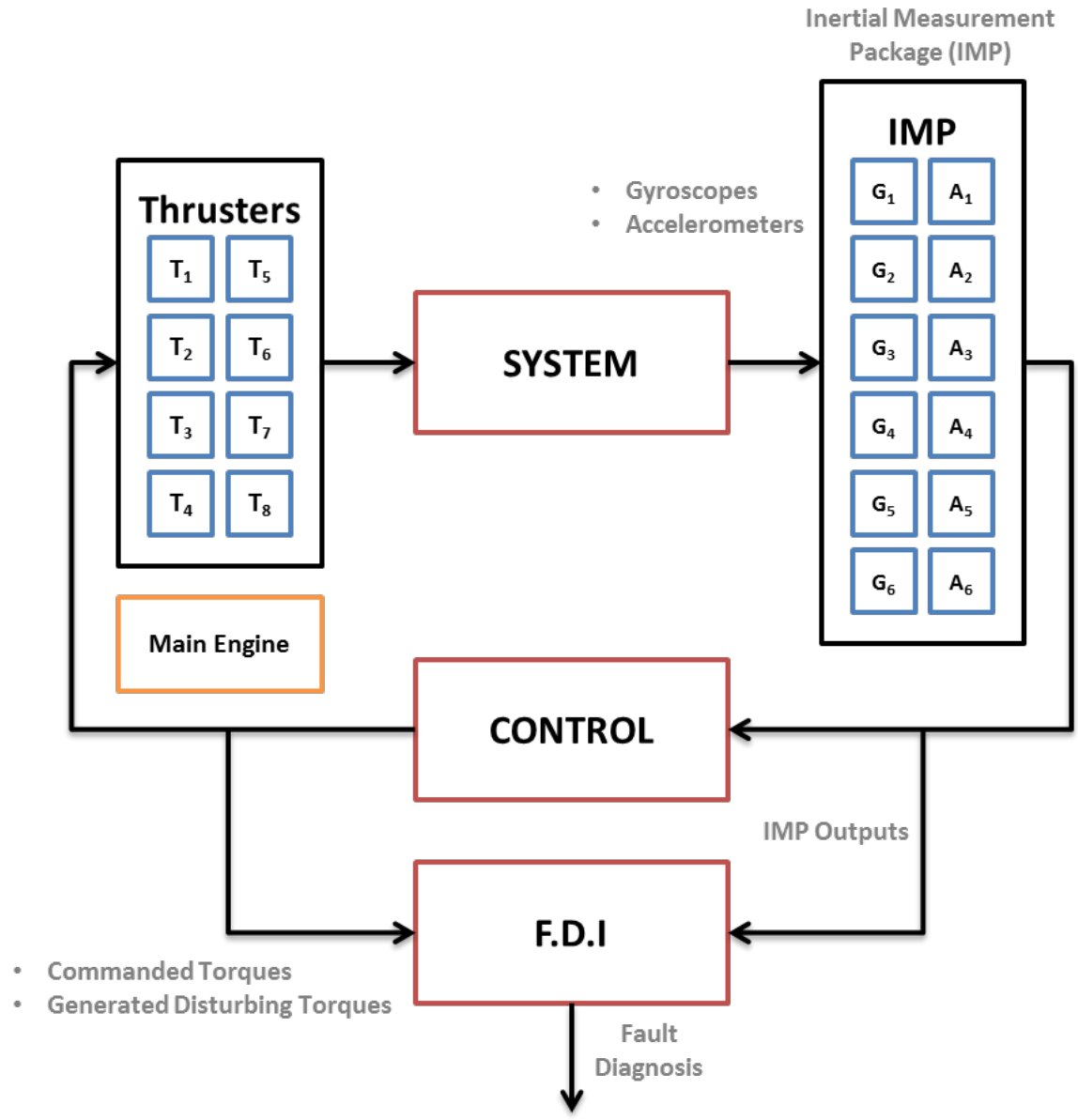


Fig. 2.19 Structural Diagram of the LPF's FDI Generic Concept

In this dissertation, the three selected FDI approaches and their corresponding algorithms are tailored in such a way so that they can detect and isolate thruster failures on-line and in real-time for different trimming points during the system operation. The algorithms are designed for linear system models, but Euresis and Euphoria Filters are modifiable to also handle nonlinear dynamics. Since the overall

nonlinear system is always around an operating/trimming point, depending on the mission scenario, there is always a respective linear(ized) model which can be readily derived automatically using appropriate simulation procedures. The design procedure(s) for all these three methods can be set to automatically create the appropriate FDI schemes in different cases of fault and severity. The definition of key parameters in the new design methods can be set to be carried out on-line with the created code and to operate as plug-ins without interfering with the rest of the system operation.

The LPF reference application focuses on actuator (thruster) faults only. It does not consider the possibility of sensor (gyro, accelerometer) faults. The underlying assumption in the current thesis is that the sensors operate perfectly. However, this premise may be proved too limiting and unrealistic, as the sensors quite often malfunction. This malfunction, in turn, confuses not only the FDI per se, but also the control system, compromising its performance; for this reason, in some scenarios there are additional tests which include two or more actuator failure simultaneously. The robustness of an FDI system can be considerably enhanced if designed to detect sensor and actuator failures simultaneously, without presuppositions of partial integrity of one set versus the other.

The developed Diagnosis Filter, which is a geometrical/analytical redundancy approach, encompasses not only both kinds of failure but also uncertainties (parametric) in the dynamics, without necessarily an increase in dimensionality, i.e. the need for construction of additional Diagnosis Filters; this depends on the specific system topology, which may allow for all faults to be detected by one filter only. The same also applies for the Euphoria Filter, the so-called H_2 based Filter. The Euresis Filter accommodates sensor failures as well, at the expense of the incorporation of additional models to reflect the sensor failures.

In the Diagnosis and Euphoria filters, even if the measurement based constraints indicate more than one filter to cover all the failures, the requisite number will still be considerably lower than if one detection filter corresponded to exactly one failure alone [15, 29]. This would alleviate on-board computational requirements and make the design implementable real-time.

The FDI objectives for this study are the following:

- Identify a failed thruster in any thruster configuration.
- Pin-point thruster failure severity.

- Identify two or possibly more simultaneous thruster failures in any configuration.

Such a capability to identify, on-line and in real-time, any of the eight thruster failures, occurring at any point during system operation, with any degree of severity and covering a wide range of fault profiles, has not been possible using the currently available methods [15, 29], which usually require a disturbance identification stage. This clearly distinguishes the work presented in this thesis from previous studies and defines its contribution in the field, specifically within the context of the proposed FDI techniques.

2.3 Types of Faults Considered

The thrusters can be functioning perfectly (open) or have a failure (closed). Thus, the model of a thruster fault in the LPF simulator is:

$$u_r = \delta u_T \tag{2.2}$$

where u_r is the real thrust and u_T the commanded thrust. $\delta \in [0,1]$ represents the thruster effectiveness, and its complement, $\gamma = (1-\delta)$ the percentage of failure, or failure severity. Thus,

- $\delta = 0$ represents the non-faulty case.
- $\delta = 1$ represents the case of complete loss of thrust.
- $\delta < 1$ represents intermediate degrees of thruster effectiveness (or complementarily, failure severity).
- $\delta =$ intermittent step functions of varying magnitudes, captures intermittent failures of different severities.

Identifying the thruster effectiveness factor δ, or its complement γ, which represents the failure severity, is fully addressed in Section 4.4. If δ is constant, then its complement represents a fixed percentage degradation in thruster function. However, the parameter δ need not be constant or intermittently constant. Another type of faults are the leakages that can also be described by appropriate functional representations of the parameter as, for example:

$$\delta = \frac{1}{1+\alpha t} \tag{2.3}$$

where α signifies the leakage rate, which can be set to a very low value. For example, for $\alpha = 0.001$, the thruster effectiveness is almost 100% at the beginning and is reduced to 99% at t =100 s. This would signify the presence of a slow leakage in the thruster. Other functional representations corresponding to different modes of failure, such as a linear degradation of a thruster function, are also possible. In all the three design methods (Diagnosis, Euresis or Euphoria Filters) once a thruster failure has been isolated, the failure severity can then be identified via a simple procedure as described in Section 4.4. This is possible with either a constant or a time varying δ. Table 2.1 summarizes the nature of the modelling errors.

Table 2.1 Nature of errors that affect the LPF's thrusters

Actuator uncertainties	
Thrusters	Bias (leakage), noise, high frequency dynamics Position and rotation misalignment Maximum thrust command step, quantization
Dynamics uncertainties	
Parametric uncertainties	Inertia matrix Mapping of spacecraft acceleration into drag-free acceleration Mapping of spacecraft acceleration into suspension acceleration Suspension stiffness Drag-Free stiffness and matrix of influence of thrusters on spacecraft acceleration centre of mass
Dynamics uncertainties	Coupling between axes, high frequency uncertainties
Sensor uncertainties	
Suspension sensors **Drag-Free sensors**	Noise, high frequency dynamical uncertainties
External disturbances	
Suspension actuator disturbances **Thrusters disturbances** **Drag-Free disturbances**	Bias, magnitude (leakage)

2.4 Evaluation Criteria for the FDI Algorithms

In terms of evaluation criteria, it is clear that the FDI algorithms should achieve fault sensitivity maximization, with a consequent minimization of false alarms and missed detection rates. The true identification of a specific faulty thruster and an accurate severity identification are also required.

More specifically, the performance characteristics for each FDI scheme can be quantified in terms of applying the following evaluation criteria:

1. False Alarm Rate r_{fa}
2. Missed Detection r_{md}

These important indices capture the accuracy of the FDI process. It is important that both are kept at an absolute minimum. The false alarms rate r_{fa} is the ratio between the false detections when there is no fault over the total number of actual failures; the missed detection rate r_{md} is the ratio of the missed failures over the total number of actual failures. These have been quantified from the totality of the simulation experiments on the simulation environment. The true detection rate r_{td} can then be computed as the difference $(1 - r_{fa} - r_{md})$. The criterion r_{td} is the true measure of FDI accuracy; ideally, it should remain close to one.

3. Correct Isolation Rate r_{ic}
4. Wrong Isolation Rate r_{iw}

These indices can be computed as the ratio between the numbers of correctly and incorrectly identified failures over the total number of failures, again as an average over a set of experiments, respectively. The true isolation rate r_{ti} is then the difference between r_{ic} and r_{iw}. It should be noted that the sum of r_{ic} and r_{iw} is not necessarily equal to one, as inconclusive isolation instances may occur. If we allow those to be classified as wrong isolations, then the two indices are complementary.

5. Time to Detection t_d
6. Time to achieve Isolation t_i

The above indices constitute the main criterion for the algorithm effectiveness. These have also been quantified for each method from simulation runs on the simulation environment, for various thruster faults and under different scenarios. As

each run may vary in the time it takes to detect and isolate a failure, the corresponding average detection time t_{da} and average isolation time t_{ia} from the collection of experiments on the simulation environment, has been computed and will be considered.

Of the above indices, 1-4 address the accuracy of the FDI scheme, while 5 and 6 its efficiency. They should all be minimized except from criterion 3, while the two derivative indices, r_{td} and r_{ti} should be maximized, as well as the correct isolation rate r_{ic}. These definitions are generic and method independent; however, the values they assume depend on the specific used FDI algorithm whose performance they characterize. These indices have been computed as a result of extensive Monte Carlo runs on the complete FDI system.

Chapter 3

A Critical Survey of Model-based FDI Methods

3.1 The Geometric and Analytical Redundancy Approach

Model-based FDI has been of interest in aerospace and other safety critical fields since the late 1970s. Some of the earliest work on FDI [33–35] focused on sensor failures and utilized the so called 'analytic redundancy' - essentially employing an analytic model of the sensor(s) for generating the necessary 'parity relations'/residuals, for fault detection and isolation. These were the first Model-based FDI techniques, which, by making use of analytical models for redundancy, enabled FDI on systems of critical importance without the need for additional hardware.

By the early 1980s, the first Detection Filter was designed [36], which employed a model of the system and made use of its structural/geometrical properties, in order to generate residuals in specific directions corresponding to actuator and sensor failures. Later, in 1986, a complete and very elegant (geometric) mathematical theory for Detection Filters, was presented by Massoumnia [37]. This became the catalyst for subsequent work on Model-based FDI using the geometric approach [38–48].

The Detection Filter is very similar in structure to the Kalman Filter or even to the Luenberger Observer [49]. It assumes a linear plant and it can provide information for more than one fault simultaneously. However, it differs from the above mentioned observers in the manner of its gain matrix design: in the Kalman Filter the gain matrix is chosen such that the effect of process and measurement noise is minimized in the state estimate, while in the Detection Filter the gain matrix is chosen such that the failure(s) are related to the residuals in a very specific way,

that allows not only fault detection, but also - where the system topology allows - to identify it; in fact, it can simultaneously identify more than one faults.

This chapter presents some of the basic background principles of analytic redundancy and detection filter structure. In the context of failure detection, a residual is a function of time, which is nominally zero, or close to zero when no failure is present, but it is distinguishably different from zero when a component of the system fails. For example, the difference between the outputs of two identical sensors measuring the same quantity is the simplest form of a residual. The process of generating the residuals from relationships among instantaneous outputs of sensors is usually called direct redundancy. However, it is also possible to generate the residuals using temporal redundancy, which is the process of exploiting the relationship among the sensor outputs and actuator inputs. This requires a hypothesized model of the dynamics of the system to correlate sensor outputs and actuator inputs at different instants of time.

A simple typical first order discrete system serves as an illustrative example. Consider the following system:

$$\begin{aligned} x(t+1) &= \alpha x(t) + bu(t) \\ y(t) &= cx(t) \end{aligned} \tag{3.1}$$

If the system is functioning properly and no failure is present, then a simple computation shows that:

$$y(t) - \alpha y(t-1) - cbu(t-1) = 0 \tag{3.2}$$

Relations like the above are known as generalized parity relations. A parity relation by itself is used to generate a residual $r(t)$. In the above example, simply take:

$$r(t) = y(t) - \alpha y(t-1) - cbu(t-1) \tag{3.3}$$

Assuming that the actuator is perfect and no measurement noise is present, a nonzero $r(t)$ indicates a sensor failure.

Full State Observers (FSO) are another class of processors, which use temporal redundancy to generate the residuals. Consider, for example, the following Linear Time Invariant (LTI) system with two actuator inputs:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_1 m_1(t) + B_2 m_2(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3.4}$$

The term $B_1 m_1(t)$ characterizes a failure of the first actuator, while $B_2 m_2(t)$ of the second. The functions $m_i(t)$ are assumed to be completely unknown, with $m_i(t) = 0$ when there is no failure present.

Next, we design a full order observer:

$$\begin{aligned} \hat{\dot{x}}(t) &= (A + DC)\hat{x}(t) - Dy(t) + Bu(t) \\ z(t) &= C\hat{x}(t) \end{aligned} \tag{3.5}$$

When a failure is present, the innovation $z(t) - y(t)$ will start to grow and, by placing a threshold on its magnitude, the presence of a failure in the system can be detected.

Next, through an appropriate choice of the gain matrix D, in conjunction with linear transformations on the innovation/residuals, the latter can be constrained to have a fixed direction in the output space, and, in fact, to lie in independent subspaces for different actuator failures. In particular, defining two linear transformations on the innovation $r_1(t)$, $r_2(t)$, as follows:

$$\begin{aligned} r_1(t) &= H_1[z(t) - y(t)] \\ r_2(t) &= H_2[z(t) - y(t)] \end{aligned} \tag{3.6}$$

matrices D, H_1, H_2 can be found such that the failure of the first actuator shows up in $r_1(t)$ but has no effect on $r_2(t)$, and similarly for the failure of the second actuator. Clearly, if the innovation growth is constrained to independent subspaces, then H_1 and H_2 can simply be taken as the projection matrices onto these subspaces.

From elementary system theory, for a nonzero $m_2(t)$ not to affect $r_1(t)$, the image of B_2 should be in the unobservable subspace of the system $(H_1C, A + DC)$. In addition, for a nonzero $m_1(t)$ to show up in $r_2(t)$, the image of B_1 should not intersect the

unobservable subspace of $(H_1C, A + DC)$. Similarly, for the unobservable subspace of $(H_2C, A + DC)$.

By a proper choice of the matrices D, H_1, H_2, the observability properties of the system relating the failure events to the residuals can be modified. The unobservable subspace of $(H_1C, A + DC)$ is simply the subspace spanned by those eigenvectors of $(A + DC)$ which are in the null space of H_1C. In addition, the column vector B_2 should be a linear combination of those eigenvectors, since the second actuator failure should not show up in the first residual. The problem, therefore, is to use the freedom in assigning the eigenvectors of $A + DC$ to satisfy the failure detection and identification requirements.

The work of Massoumnia [37] contains an elegant, complete, and thorough development of all the necessary mathematical apparatus for designing different classes of detection filters. The pertinent details for the design of a detection filter for thruster faults will be given in the presentation of the preferred architecture(s) for FDI, in the next chapter. However, the purpose of presenting this simple example here is to elucidate some of the terminology used interchangeably in the open literature, as well as its associated concepts. In addition, some of the characteristic properties and limitations of the detection filters can also be as following:

1. Detection filters require linear plants as there is no mathematical theory for Detection Filter design in nonlinear systems.

2. Detection filters are special type observers, which can also be termed Unknown Input Observers or even Unknown Linear Parametric Estimators (ULPE), since observers can also be set up as parameter estimators. This model-based observer does not need to have any knowledge of the input to the system in order to perfectly estimate the system state. Even if the actuator fails and its behaviour, i.e. the terms $m_1(t)$ and $m_2(t)$ in equation (3.2), is unknown, the observer perfectly identifies the system state, but also retains the (nonzero) residual information that corresponds to the failure(s).

3. Subspace Projections and the Eigen Structure assignment are an inextricable part of the FDI design framework in this context and cannot be viewed separately. They are essentially, synonymous and an alternative statement pertaining to the crux of the design per se.

4. Depending on the specific system topology, there are limitations as to how many faults can be identified simultaneously with one detection filter. In such cases, more than one filters become necessary [36, 37].

5. By construction, the detection filters are not designed to simultaneously handle (stochastic) disturbances and measurement noise, or modelling uncertainty (structured or unstructured). Hence, the issue of robust residual generation, devoid of any noise, disturbance, and uncertainty contamination, is key to robust FDI. This is precisely the challenge to the designer, which needs to be addressed separately as well as in parallel and simultaneously.

The fifth characteristic can be a serious limitation to a high performance robust FDI. In this setting, the only way a designer can address the issue is to:

i. either carefully set (fixed) thresholds [27, 50, 51], that are accurate measures of the expected disturbance and uncertainty contributions to the residuals; anything above the threshold would then be interpreted as a failure. However, this carries the risk of either being overly conservative, in which case missed detections would occur, or, in the opposite case, the rate of false alarms would increase. We also note here that fixed thresholds may not always work, as the disturbance magnitudes vary with the operating conditions. Using spectral content information to more accurately set the thresholds, on the other hand, would be computationally intensive and not useful for the real-time implementation of the FDI algorithms

ii. or do appropriate loop shaping through the control design, so as to ensure maximal disturbance rejection, and avoid threshold setting

iii. or do appropriate signal (pre)processing of the inputs to the FDI algorithms (of the plant input(s) and output(s))

In European Aeronautic Defence and Space (EADS) [52], an intelligent scheme is devised. It uses UIOs, with the unknown input provided by the failure(s), which is tantamount to a typical [37] Detection Filter structure, in order to identify failures. In this work, which follows closely the work by Uppal [51] and Patton [15]), only drag-free and suspension measurements are used for FDI because they have low noise. The algorithm performance is very good, even though a closer inspection of the results suggests that there is room for improvement, in terms of more accurate setting error thresholds using ideas in (iii) above, since, for the current system, the controllers are already set, so that we cannot exploit a truly multivariable controller structure, which could have appreciable built-in robustness and disturbance rejection properties, given the strong coupling from drag-free and attitude loop to the suspension loop [51].

In Patton [15], a combination of disturbance identification algorithm for threshold selection is run in parallel to a Detection Filter scheme, assuming linearity, with good performance. However, the problem of identifying disturbances in parallel to the FDI scheme can result in inaccuracies and loss of robustness; as a result, the thresholds will have the shortcomings mentioned above. Although the algorithms are not presented in sufficient detail, it seems, in light of the preceding discussion almost certain that further improvements can be achieved.

Remark: None of the works cited in the references [8, 11, 17, 18, 20–22, 33–48], or even in the open literature some of which is additionally cited in Uppal [51], address the issue of FDI in conjunction with a good control design where multivariable loop-shaping can offer tremendous advantages. Moreover, given that, most systems have built-in redundancies (especially in control effectors); a good control design can greatly alleviate the FDIR process, by automatically remedying the problem by exploiting the system redundancy.

3.2 The Multiple Model Approach

The Multiple Model (MM) FDI methods started to develop with a few years delay from the geometric and generalized parity methods of the mid-1980s, and are a natural extension of the Multiple Model Kalman Filters (MMKF) of the 1970s. Here, however, instead of each filter in the bank of Kalman Filters (KF) representing a specific parameter value within its associated uncertainty interval, each of the filters corresponds to the particular failure that it models. The relative probabilities are updated in a completely analogous manner to those in the MMKF case, until convergence occurs to the most probable member in the set, which corresponds to the actual failure.

The Multiple Model concept is attractive, as it directly models each fault in a corresponding KF model in the bank of filters for all failure modes. If the faults are not exactly known, both in direction and in magnitude, convergence to the correct fault mode becomes a challenging issue. This convergence depends also on the nature of the underlying process, which may exacerbate the problem. This is completely analogous to the convergence problems encountered in the Multiple Model (Adaptive) Kalman Filters of the 1970s, when none of the filters chosen exactly matched the actual (unknown) parameter value of the estimated plant. However, in the FDI setup, given that directionality is a dominant dynamic feature, even if the actual failure level is not known, fault detection - but not fault severity - can still be

achieved reliably, provided that at least the directionality is accurately reflected in the bank of the chosen KF models.

This alternative FDI design philosophy, using the KFs as optimal state estimators, allows for the optimal (stochastic) disturbance rejection from the residuals and, if the topology is right, it can ensure a robust fault detection; thus, minimizing false alarms or missed detections that would be attributable to the (stochastic) disturbance factors in a standard Detection Filter setting, which, by construction, is deterministic. In addition, this alternative FDI architecture can handle nonlinear plants, by simply substituting the Kalman Filters with various nonlinear state estimators, such as the Extended Kalman Filter (EKF), the Gaussian Sum Filter (GSF) and the Unscented Kalman Filter (UKF) [53–55]. The UKF filter proposed in Valavani [56] has excellent performance and avoids the divergence problems of other nonlinear estimators [57].

The FDI architecture(s) implemented in the LPF simulation model are fundamental Detection Filter implementations which use many non interacting models - one for each of the eight thrusters whose failures they are supposed to be monitored. In previous studies [17, 18, 58, 59] the Interactive Multiple Model (IMM) approach is employed, making use of banks of UKFs that reflect all the failure modes. A probabilistic approach is then used to lead to the correct model, which represents a particular failure which would thus be identified. More specifically, they utilize a standard multiple model structure as in (Adaptive) Multiple Model Kalman Filtering, where the individual models have been selected to conform to prescribed failures. Each model, then, realizes a UKF, by combining the FDI structure with signal filtering at the same time. The individual model probabilities are then updated, in order to determine the most probable model, which would correspond to the specific failure mode for which it was designed. The reference application is reaction wheels in a satellite Attitude Control System [17, 18] and sensor failures in Unmanned Aerial Vehicles (UAV) [58]. In Cork [58], the theoretical development allows for model switching, which has never been tried in simulation. The IMM-UKF algorithm performed adequately for some (milder) sensor faults - only single sensor faults were considered - with the results not indicating the accuracy of the mode estimation. The authors also point out that faults such as drift, bias, or ramp would cause UKF to diverge. According to their own admission, to avoid divergence requires knowledge of the true mode of the system. In Tudoroiu [17, 18], the authors also emphasize the importance of designing correctly the set of modes that represent the possible system behaviours, an exercise that can be unwieldy in a nonlinear setting.

In concept, this approach seems attractive but in reality, it can be impractical and ineffective due to a variety of reasons. In addition to the computational requirements, there are issues of convergence and robustness, as well as of possible UKF filter divergence, which can be exacerbated by the nature of failures themselves, without even considering all the other sources of uncertainty. This however, is a nonlinear scenario; the divergence of nonlinear estimators has been well documented in the literature [57] even in the absence of failures. If on such a sensitive structure one superimposes the requirement that a failure occurs, especially if this is a mild one at its onset, this failure information might be cloaked/overlooked and misinterpreted in the setup as there is no provision for signal content separation in the overall scheme.

Ru and Li [59] employed various schemes: MM, IMM, Hierarchical Interactive Multiple Model (HIMM) and Hierarchical Interactive Multiple Model Maximum Likelihood Estimation (HIMMML), in which the underlying model is linearized. It is claimed that HIMMML has the best performance; there is not enough detail to have any further insights, though most of the comments made above would also apply here as well. In Li [60]), an attempt to remedy some of the inherent shortcomings is made, by using Expected Mode Augmentation algorithms, which focus on the best choice of the true models to be implemented on an IMM structure. Briefly, the proposed technique starts with a specific number of models that form a grid, within which a new model set choice is evolved based on the probabilistic synthesis of the results in the previous model set. Thus, the scheme has the capability of evolving to new model sets based on current information, without requiring an excessive enhancement of the number of models; hence the characterization 'Variable Structure' and 'Expected Mode Augmentation'. This approach depends on the requirement of the mode space being continuous, which would make it unsuitable for failures that violate this in general, which is a rather large class of failures. There does not appear to be a significant performance or robustness improvement and the issues pertaining to computational complexity, convergence, and performance robustness still remain. However, as in the case of LPF the failures seem to be of limited number and well defined, the usage of an IMM approach may be actually promising, as it provides a more direct way of failure identification, since each of the individual models chosen in an IMM structure would directly correspond to a specific failure mode. This will be further discussed in the next section, where we suggest our preferred FDI architectures for the LPF thrusters' faults.

3.3 The Optimization Based Approach to FDI Using H_∞

The optimization based methods to robust filtering and FDI employed in this dissertation are based on the seminal paper by Doyle [61] in which the authors provided, for the first time, state space solutions for the control problem. They used modified Riccati equations with off-diagonal matrix blocks, which essentially forced a joint and simultaneous solution of the Filter Riccati and Control Riccati equations. In addition, they were able to prescribe a desired error norm bound γ, which the H_∞ solution was designed to satisfy, and which could be improved via iteration under appropriate conditions.

These control solutions ushered in the dual solutions to the robust filtering problem, which optimized the H_∞ norm, and were followed by considerable activity in the late 1980s to mid-1990s [62–67]. There was a great deal of expectation regarding the new design method, especially since it was now possible to include in the design framework, for optimization, various types of disturbances and constraints by plant augmentation and to assign appropriate weights to the corresponding blocks. Parametric uncertainty was also included in this manner. However, in 1991 it was established that the two methods, i.e. H_2 and H_∞, were essentially equivalent as far as the control design was concerned [28]; they provided stability robustness to the unstructured uncertainty that could maintain the prescribed performance under one actuator failure, and with some degradation, two and even three simultaneous failures. This was possible since the control design automatically made use of the system's built-in redundancy. However, they were both extremely sensitive to parametric uncertainty, and conservative with respect to attainable performance, because all the directionality information was lost in the solution, given the spherical nature of the norm expressions. Some slight improvement in performance for both filtering and control was achievable by $\mu - synthesis$, and later by near Linear Matrix Inequality (LMI) and Linear Parameter Varying (LPV) designs. However, in the latter two, the guaranteed robustness was compromised.

These optimization methods may be excellent for robust control and filtering under unstructured uncertainty and (non-parametric) disturbances and for the minimization of fault effects because they can be included in the design from the start as additional plant augmentation blocks. They are not however appropriate for FDI, if used alone, as they tend to confuse the directionality information, which is key to the detection and isolation of faults.

Previous work [68, 69], confirms to some extent the above remark. The Robust FDI is attempted for the thruster faults of the MICROSCOPE[1], using H_∞ estimators. In the study by Valavani and Voulgaris [28], the authors state that the proposed FDI scheme fails to isolate the faults successfully. In addition, in [68], the authors note that this solution has been tested but it has revealed worse FDI performance than a bank of twelve filters, one for each FEEP thruster fault, with some additional post filtering of the residuals. It is not made clear in the paper whether the 12 FDI filters, which should rather be termed robust residual generating filters for FDI, interact and how a fault determination is made amongst them. Additionally, as discussed earlier, the assumption of insufficiently defined system parameters and nonlinearity can seriously hamper such a design, which would have to be quite complex given its dimensionality, from including fault and parameter considerations in the design model.

3.4 FDI Method Suitability - Selection Criteria

The following generic factors determine the suitability of an FDI method for a particular application/mission [15, 51]:

- Simple and Systematic Design Procedure.
- Suitability for detecting and isolating sensor/actuator failures.
- Simple implementation requirements; low complexity (also weight, bulk considerations).
- Observability and information processing requirements.
- Capability to function in real-time reliably and continually.
- Capability to handle multiple faults (whether simultaneous or successive).
- Capability for fault isolation (structured/robust residuals).
- Suitability for detecting and isolating sensor/actuator failures.

[1]The Micro-Satellite à traînée Compensée pour l'Observation du Principe d'Equivalence (MICROSCOPE) is a 300-kilogram (660 lb) class minisatellite operated by CNES to test the universality of free fall (the equivalence principle) with a precision to the order of 10-15, 100 times more precise than can be achieved on Earth. It was launched on 25 April 2016 alongside Sentinel-1B and other small satellites.

- Capability for easy tuning of the FDI (due to parameter and trimming point changes).
- Reproducibility and consistency in fault isolation.
- Capability to handle non-linearity.
- Robustness to modelling errors and disturbances, yet sensitive to faults.
- Transparency for easy troubleshooting for safety and reliability.
- Capability for (analytical) validation.
- Non-intrusive to existing system; plug in capability to operating system.
- Adaptation and learning not a requirement; detection being distinct from identification.

In terms of more specific FDI performance, the objectives of a good FDI procedure should include:

- Fast detection time/low detection delay.
- Minimal false alarms rate.
- Minimal missed detection rate.
- Fault identifiability and isolation.
- Robustness to parameter variations, uncertainty and noise.

The two sets of criteria above, consisting of both generic factors and evaluation/performance criteria, comprise the basic prerequisites to achieve a space mission objective. While all are important for the purposes of this dissertation, the most important weight criterion is the robustness and fault isolation capability, and the FDI performance requirements as described above.

3.5 Rationale for Specific Architecture(s) Selection

3.5.1 Linear Model Availability

The overall system with the three sub-loops (the DF, SUS and ATT Loops), and the corresponding controllers is very well represented by linear (transfer function)

models in the LPF simulation environment. From those, state space descriptions are extracted for further analysis and FDI algorithm synthesis. The linear model descriptions also include a thorough disturbance profile to capture the effects of modelling inaccuracies and nonlinearities, as well as sensor noise profiles, compatible with linear model descriptions.

The Diagnosis and Euphoria Filters are based on the linearity of the plant dynamics, while the Euresis Filter not only ensures optimal performance for linear plants but it can also handle non-linear ones. In the latter case, the MMKFs would have to be substituted with UKFs which are explicitly designed to handle non-linearity. Therefore, the Euresis Filter is also applicable to a non-linear plant description, if available in a simulation environment.

3.5.2 Capability for Fault Identification

Diagnosis and Euphoria Filters are developed to generate structured residuals which correspond to specific (thruster) failure directions. As a result, they directly identify which thruster has failed based on the corresponding residual direction as generated during system operation.

The Euresis Filter although it does not explicitly generating structured residuals, it does encompass specific models corresponding to each thruster failure. By construction, the method converges to the right model corresponding to the specific failure mode.

In fact, all of them can be valid for any operating condition around which a linear model is obtained, depending on the specific mission requirements. Their FDI architectures are readily adaptable within a simulation environment as long as the failure directions are well defined. In addition, they represent a plug-in capability for FDI without interfering with the rest of the system functions and work in parallel, utilizing the available signals and measurements from the operating system.

3.5.3 Robustness

Robustness to modelling inaccuracies, parametric uncertainty, disturbances and noisy measurements is key in a robust residual generation process, in order to ensure correct fault detection and identification, without any false alarms and missed detection.

All three designed FDI techniques can incorporate additional robustness characteristics to eliminate/minimise the effects of the above-mentioned error sources.

These characteristics can be incorporated following a truly multi-variable Singular Value Decomposition (SVD) analysis of the entire system to reveal the vulnerabilities in certain frequency ranges of interest, which can be then mitigated by adding appropriate filters to the FDI signals before they are fed into the FDI blocks. More details are given in Chapter 4, where the systematic design is described in detail.

3.5.4 Real-Time Capability

All the filters are designed to run in real time, and with the simplest possible computation, as they are realized on existing system models in a simulation environment. There are no delays in identifying disturbances and setting thresholds, as was the case in work previously carried out by others. The current designs have built-in capabilities to reject all sources of error that are irrelevant to the FDI process during system operation.

3.5.5 Method Genericity- Suitability for Other Space Applications

Since all the proposed techniques are model-based, they are applicable to any other space application where a model is available.

The comparison criteria, as listed below, include some of the generic FDI suitability factors [15, 40]:

1. Design synthesis complexity.
2. Suitability for detecting and isolating sensor/actuator failures.
3. Implementation requirements, complexity.
4. Observability and information processing requirements.
5. Real-time performance and software validation.
6. Capability to handle multiple faults.
7. Fault diagnosis efficiency, accuracy and reliability.
8. Robustness to modelling errors, parametric uncertainty, noise and disturbances, yet sensitive to faults.
9. Transparency of scheme for easy troubleshooting for safety and reliability.

10. Compliance with system engineering requirements.
11. Suitability for other space applications.
12. Algorithm dependence on subsystem/unit characteristics.
13. Applicability to the LPF thrusters FDI.

Table 3.1 summarizes the above mentioned characteristics of the three model-based FDI approaches.

Table 3.1 Comparison of Methods based on Selective Criteria

	Methodologies		
Criteria	**Geometric/Analytic Redundancy**	**Multiple Model /Randomized Algorithms**	**Optimization Based H_2, H_∞**
Design Synthesis Complexity	Relatively simple to design; very systematic	Simple design; fairly ad hoc, especially in the case of 'randomized algorithms	Very systematic design process
Suitable for Sensor/Actuator Failure Detection	Very suitable.,Provides for both sensor & actuator FDI in same design	Suitable, if appropriately set up	Very suitable in a multiple model setting
Implementation Reqs/Complexity	Very simple, low complexity	Though simple to set up, computationally demanding	Simple to set up. Requires as many design models,as the number of expected failures
Observability and Information Processing Requirements	Very good; depending on available measurements	Good; depending on available measurements	Very good; depending on available measurements
Real-time performance and software validation	Very good in general; software validation straightforward	Good to fair, depending on specific algorithm class.,SW validation generally very involved	Very good.,SW validation straightforward
Multiple Faults FDI (more than one failures handled)	Very suitable; quite often, just one filter can be used to identify all failures	Suitable but requiring additional computational complexity (more models, one for each failure)	Suitable, but requiring more than one 'design models', one for each failure
Fault diagnosis criteria	Very good	Good	Very good
Robustness to Modelling Errors, Uncertainty, Disturbances	Not provided for in design considerations	Modelling errors are not handled	Robustness to certain types of structured and unstructured uncertainty; no stochastics handled
Transparency for Easy Troubleshooting	Very good	Fair	Very good
Compliance with LPF system engineering requirements	Generic, hence compatible	Randomized algorithms not suitable	Very compatible
Suitability for other space applications	Generic, hence applicable	Model-based; applicable where dynamic models exist	Model-based; applicable where dynamic models exist
Algorithm dependency on subsystem/unit characteristics	Model-based, hence dependent	Model-based, hence dependent	Model-based, hence dependent
Applicability to LPF Thrusters FDI	Generic, model-based, hence applicable	MM applicable; Randomized algorithms not applicable	Very good

Chapter 4

FDI Techniques

In terms of the overall design philosophy, the majority of FDI methods fall into one of two major categories:

i. Geometric/Analytical Redundancy methods, which include the Diagnosis and Euphoria Filters.

ii. Multiple Model/Randomized and Optimization Based (H_2, H_∞) Methods (within a Multiple Model Framework), which include the Euresis Filter.

The majority of methods in either category are Model-based and they all seek to generate robust residuals to support their specific diagnostic criteria for FDI. Robustness is widely accepted here to mean minimal sensitivity with respect to plant disturbances, sensor noise, and parametric uncertainty but not necessarily to failure directions, which is the main FDI objective. In this respect, the structured residuals are key factors along with their associated diagnostic criteria, which are primarily generated by the methods in category (i). More specifically, these methods configure their observer gains to minimize/maximize residuals in specific failure directions or to generate residuals whose transformed directions via the observer dynamics, uniquely correspond to specific failure directions. In the first case, such residual magnitude optimization is achieved via orthogonal projection type methods onto observer range/spans [15, 29] or via adjoint directionality recovery methods on otherwise optimized residuals (H_2 based Filter). In the methods of category (i), the structured residuals generated, preserve the directionality invariance with respect to the corresponding failure directions regardless of their magnitude. The available literature on these methods is limited. Almost invariably, the residuals generated by each algorithm are used to detect and identify a failure in both categories. While the

residuals magnitudes are used in the detection diagnosis criterion, in the Diagnosis and the Euphoria Filters, which are classified in category (i), residuals directions are used instead.

The residuals direction, rather than the magnitude, offers a much more robust diagnosis criterion, since direction is considerably less sensitive to (small) parametric uncertainty and cannot be easily affected by random noise, which is not typically contained in any specific direction and, therefore, is not expected to exclusively disturb any one direction alone. In contrast, the residuals magnitude is directly affected by noise and modelling inaccuracies, and although it is much more easier to handle analytically. Even though residual magnitude measurement is more conservative, it is less robust to both structured and unstructured uncertainties. This is why disturbance estimation and accurate threshold setting are needed in most of the available algorithms in order to minimize false alarms, while a directionality diagnostic criterion has no such requirements. Indeed, the direction captures phase information and is a more accurate and robust measure to be used. Therefore, a method that employs a residual direction diagnostic criterion for FDI is preferable to methods that use a residual magnitude criterion. Invariably, state of the art control design techniques use control error magnitudes according to specified norms, which they seek to minimize in the design process via an appropriate controller synthesis. Such designs, which result from the minimization of a particular error norm, are known to be extremely sensitive to parametric variation. In addition, they also tend to be conservative, as the directionality information cannot be reflected in a norm magnitude; for example, the error magnitude alone does not specify the error sign, which encapsulates direction. To date, the issue of how to include directionality information to enhance the performance, robustify, and desensitize the design has not been satisfactorily addressed in a systematic manner in the control area. Fortunately, the FDI problem framework does allow for the direct incorporation of directionality information in the category of geometric approaches, which, if appropriately handled in the theoretical development, can retain all the pertinent information for a robust diagnosis. The advantage here is that the FDI framework requires fault detection only, and not fault magnitude minimization. However, a good diagnosis procedure should also provide reliable information for accurate fault magnitude identification (fault severity).

4.1 Design Method 1: Diagnosis Filter

4.1.1 Theoretical Foundation

The Diagnosis Filter relies on system fault controllability. It assumes that the component failure - actuator or sensor, and thrusters in the case of LPF - affects the system response (state) and is reflected in the system response measurements (output observability). Thus, its development depends on appropriately tracing and transforming such fault controllability and observability subspaces by designing the Diagnosis Filter gain such that each fault direction has in the end a unique and invariant reflection in the residual direction.

The Diagnosis Filter is very different from the Kalman Filter concept, which is an optimal stochastic state observer in the sense of minimizing stochastic disturbances. Also, it is an observer, albeit a very specialized one, set in a deterministic framework. It assumes a linear time invariant plant, which can sustain a sensor or actuator failure, or even a parametric failure/jump. This filter also provides an estimate of the plant state along with the detection of a failure; the specific failed sensor or actuator should be readily identified from the residual direction itself. This is achieved by designing the residuals in such a way as to correspond to pre-specified failure directions. This is realized by the choice of the gain matrix with which the residuals are entered in the filter, such that a failure in a specific component is reflected in the Diagnosis Filter residual along a pre-specified direction. Thus, the Diagnosis Filter is not only able to detect a failure, but also to identify it automatically, as its resulting residual assumes a pre-specified failure direction which has been appropriately transformed in the course of the design process, based on the system observability and controllability characteristics. In fact, the Diagnosis Filter can simultaneously identify more than one failures.

In some cases, if the model structure allows it, only one Diagnosis Filter can be employed to detect all possible failures. Even if the design model topology for the plant does not always support this scenario (only one Diagnosis Filter), a second diagnosis filter usually completes the set of all the failures that can be identified. All such filters have an identical structure; however, each one possesses a different gain matrix that satisfies the structured residual directionality conditions that correspond to the failure directions it is designed for.

The overall Diagnosis Filter structure is outlined in the block diagram in Figure 4.1, where Diagnosis Filter 1 (DF_1) and Diagnosis Filter 2 (DF_2) represent the Detection Filters specifically designed for the plant, and whose dynamics are identi-

cal to those of the plant where the failures occur. The design includes one additional filter (DF_1), since two Diagnosis Filters are required in this particular instance: one that can handle six thruster failures, and a second for the remaining two, as derived from the SVD analysis and the output separability analysis carried out in Section 2. In Figure 4.1, $u(t)$ is the input to the plant, at the controller output (closed loop control input) and $y(t)$ its measured output; $u_f(t)$ and $y_f(t)$ are their corresponding filtered versions (through pre-filter 1 and pre-filter 2); r_1 and r_2 are the structured residuals – the difference between the real plant output or its pre-filtered version and the detection filter(s) output(s). Pre-filter 1 and pre-filter 2 can be identical; in fact, in the specific application they are chosen to be so. As discussed in Chapter 2, the pre-filter plays the role of a roll-off filter before crossover to further reject/attenuate disturbances and sensor noise for robust residual generation.

A pre-filter can be viewed as an enhancement of the robustness properties of the plant controller, if these have not been adequately designed for in the first place. Our choice of a pre-filter is a second order filter of the form: $\frac{1}{s\times(s+0.0005)}$ based on the discussion in Chapter 2. Adding this pre-filter to the FDI process, that is, filtering the plant inputs and outputs through it, before they are entered into the FDI block, does not affect the existing plant design or its compensators. These signals are just extracted as the plant operates, they are additionally processed through the pre-filters, and then they are injected into the FDI mechanism for fault detection.

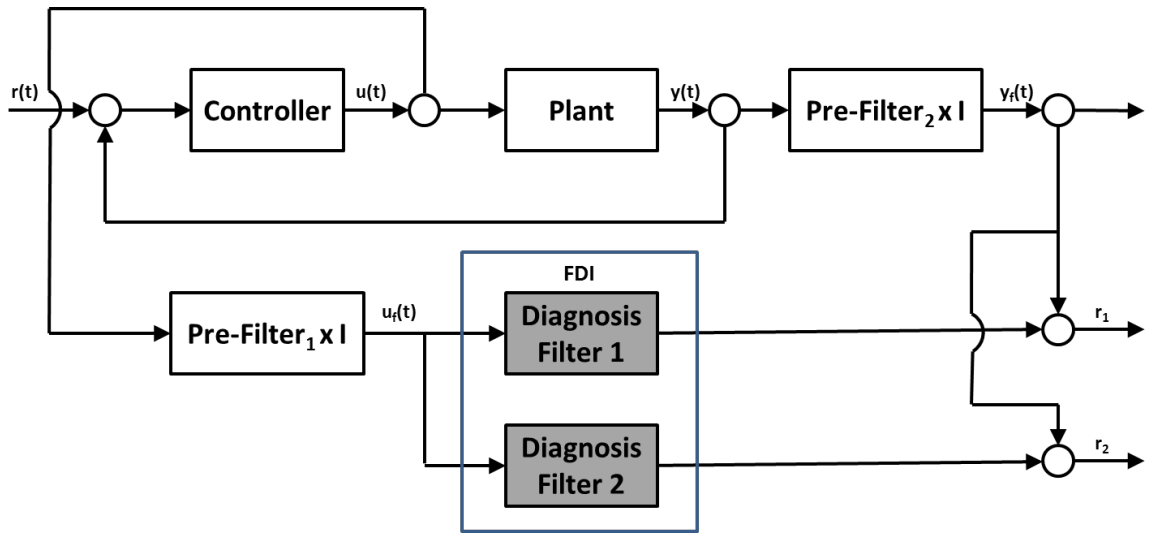


Fig. 4.1 The Structure of the Diagnosis Filter

In this sense, the FDI process functions as a plug-in, or a parallel process that runs on its own, autonomously and non-intrusively. It utilizes only the plant inputs and outputs as they evolve, in order to trigger the detection mechanism, and achieve thruster fault identification.

The architecture shown above, corresponds to the augmented/robustified version, based on a truly multivariable overall system analysis, applying singular values for the overall system, complete with controller design. This designing method functions similarly to an UIO, with the specific property of structuring its residuals along (pre-specified) failure directions helps to identify the specific failure. The basic principles of the theoretical development of the Diagnosis Filter, which are residual generation, analytic redundancy, and directionality relations, have been described in Chapter 3.

4.1.2 Outline of the Diagnosis-Filter Operation

This dissertation focuses on the actuator failures[1]. There are eight actuators in total, in the LPF spacecraft, corresponding to each of the first eight columns of the B matrix associated with a respective failure direction in the case of actuator failure. The Diagnosis Filter objective is to design the gain matrix D in such a way that the system residuals, defined as the difference between the plant and the filter outputs, are stationary with respect to the failure directions. In this way, each failure direction, processed through the filter, maps the residual into a unique direction in the output space, which is directly associated with the said failure. These failure signatures of the residual are distinct, because of the way the D matrix is designed. This section presents the sequence of the steps of the Diagnosis Filter design method.

Considering for example the Input Failure Model, which addresses actuator failures and dynamic changes, the model representation is :

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + fn(t) \\ y(t) &= Cx(t) \end{aligned} \tag{4.1}$$

where, f represents the failure direction (known) and $n(t)$ its magnitude (unknown). Since the failure magnitude is unknown, the detection filters, which are designed to have the same classical observer structure, have been referred to in the literature as UIOs.

[1] please see Chapter 2

The Diagnosis Filter is then a plant replica (classical observer structure) with an additional term (measurement update), as seen in the equation 4.2:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + D[y(t) - \hat{y}(t)] \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{4.2}$$

The gain matrix D is chosen so that the residuals are constrained to have a fixed direction in the output space and to lie in independent subspaces for different actuator failures. Thus, a residual in a particular direction signifies a specific failure that is immediately identifiable.

Within this framework, the theory allows more than one failure to be identified simultaneously. Their associated directions will be persistently present in the residual, and will be readily identified by simple operations (e.g. by taking the inner product between residual and failure directions to check for coincidence).

Furthermore, if one failure occurs, the filter residual inner product with the pre-specified directions will then contain a result aligned with one of the failures (minimum dot product angle). If more than one failure occurs, the residual inner product with the corresponding failure directions will yield as many alignments as the number of failures (minimal dot product angles). By associating these with the specific failure directions in the inner product, the failures are readily identifiable[2]. In addition, depending on the specific problem, it is possible to safely identify more than one failure simultaneously, but probably not all, according to theory predictions.

For our purposes, like in Dinh and Polle [29], we use the combination of DF and SUS loops dynamics, as shown in Figure 4.2. The necessary input information for the Detection Filter is contained in the signals that are the (controlled) plant input taken right after the thrusters and the plant output.

[2]This specific problem requires two Diagnosis Filters: one that can handle six actuator failures, and the second for the remaining two.

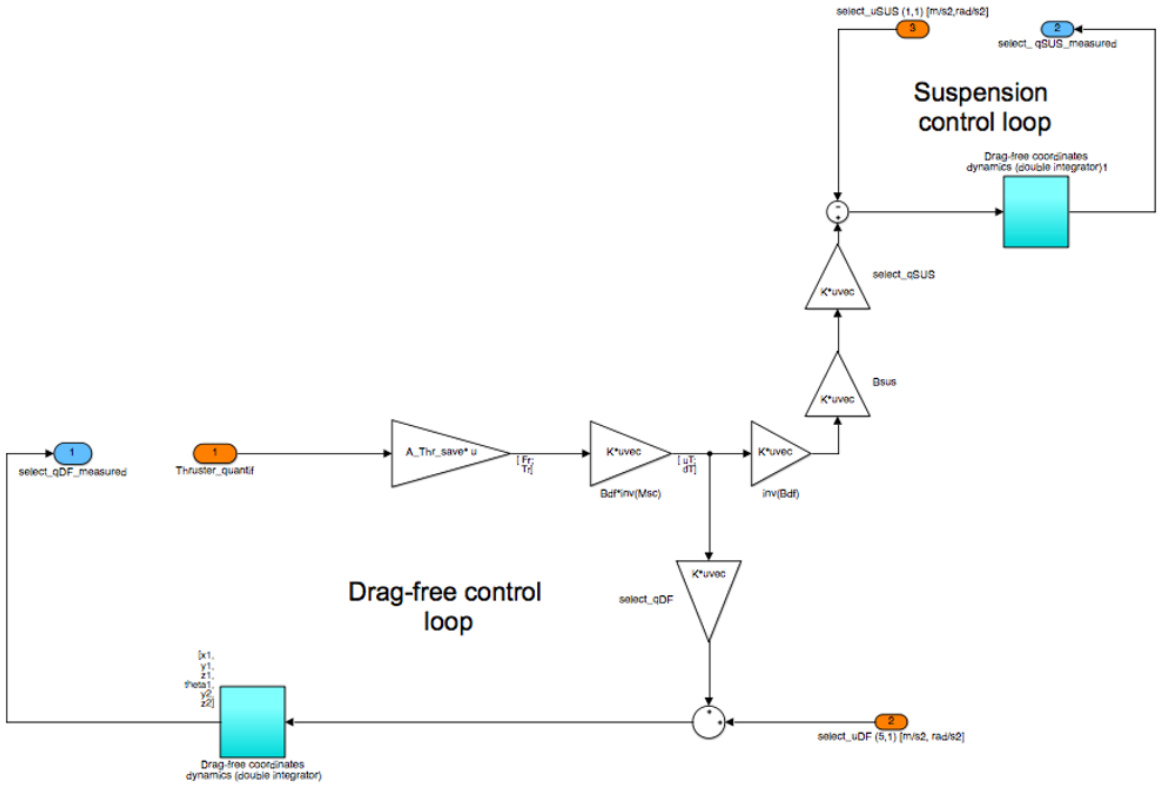


Fig. 4.2 The Combined Dynamics of DF and SUS loops

More specifically, these signals are the following:

$$u(t) = \begin{bmatrix} u_T^T & u_{DF}^T & u_{SUS}^T \end{bmatrix} \text{ and } y(t) = \begin{bmatrix} q_{DF}^T & q_{SUS}^T \end{bmatrix}$$

where

$$q_{DF}^T = \begin{bmatrix} x_1 & y_1 & z_1 & y_2 & z_2 & \theta_1 \end{bmatrix} \text{ and } q_{SUS}^T = \begin{bmatrix} x_2 & \eta_1 & \phi_1 & \theta_2 & \eta_2 & \phi_2 \end{bmatrix}$$

The first vector is actually taken at the thrusters output, while the second is the set of the DF and SUS measurements. These are schematically and symbolically represented as one input vector to the Diagnosis Filter, as shown in the upper left hand corner of Figure 2.1. However, in the actual filter implementation, $u(t)$ is a separate input, which enters the filter through matrix B (corresponding to the thrusters structure), while $y(t)$ forms the residual which enters the filter through a different input matrix, namely the filter gain matrix D. The state space representation

used for the Diagnosis Filter corresponds to the DF and SUS loops combined, as in Figure 4.2 and is given in Appendix B.

Note: As is also explained in [29], due to the rates' usage of q_{DF}^T, q_{SUS}^T for the state, the system model has an uncontrollable mode due to $x_1 - x_2$. A fully controllable model can be attained by omitting either x_1 or x_2 in the state $x_1 - x_2$. Because x_1 has a larger measurement noise, it is the one which is omitted. This reduces the model to eleven states and, hence, matrix A becomes (22×22) in dimension. The controllability of $[A, B]$ is very important since the actuator failures can ultimately be 'detected' only if their effect impacts the system state.

If we define $r(t) = y(t) - \hat{y}(t)$ then residual dynamics are governed by the following equation, which is obtained by subtracting equation 4.2 from equation 4.1:

$$\dot{r}(t) = [A - DC]r(t) + fn(t) \tag{4.3}$$

At this stage it should be noted that the residual dynamics are always stable for $[A, C]$ to be observable, even if A is an unstable matrix. The design steps of matrix D (outlined in Figure 4.3) are as follows:

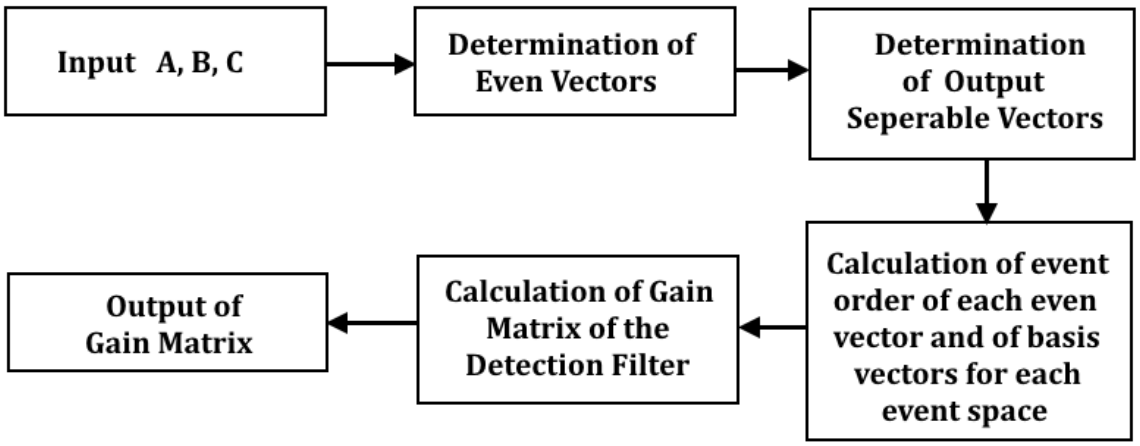


Fig. 4.3 Overall Operation of the Diagnosis Filter Algorithm

The Detection Filter theory makes it possible to find a matrix D, for any f, such that $r(t)$ assumes a fixed direction in response to $fn(t)$, regardless of the specific magnitude of $n(t)$.

- Step 1: Once the linear model is established, the first task is to find an event (failure direction) vector for each component to be monitored. Suppose f is such an event vector; if $Cf = 0$, then multiply by A until $CA^\mu f \neq 0$. Then $CA^\mu f$ becomes

the failure direction for the specific component or system. In our system, the event vectors are the columns of the B matrix: $Cb^i = 0$ $(i = 1,2,\ldots 8)$.

This essentially comprises the raw failure directions as represented by the columns of the B matrix, unobservable in the output: $CAb^i \neq 0$. In fact, the operation above yields eight vectors that define the failure directions f_i of the actuators in the output space.

Table 4.1 and Table 4.2 illustrate the output separable failure directions used for the design of the Diagnosis Filters DF_1 and DF_2.

Table 4.1 The Output Separable Failure Directions of DF_1

Six Output Separable Failure Directions used for the Design of DF_1					
0.002641	−0.00118	−0.00038	0.002521	−0.00108	−0.00038
0.00093	0.000866	−0.0015	−0.00156	0.000507	0.000602
0.000822	0.001061	−0.00192	−0.0015	−0.00063	−0.00125
0.001053	0.000399	0.000867	0.001294	−0.00266	0.001211
0.000523	0.000586	−0.00093	−0.00087	0.000946	0.000851
−0.00108	−0.00074	0.001497	0.001824	0.001166	0.000663
0.004222	−0.00419	−0.00332	0.003264	0.004207	−0.00424
−0.00039	0.001857	0.001907	0.00023	0.000402	−0.00187
0.000822	0.001061	−0.00192	−0.0015	−0.00063	−0.00125
−0.00108	−0.00074	0.001497	0.001824	0.001166	0.000663
0.004222	−0.00419	−0.00332	0.003264	0.004207	−0.00424

Table 4.2 The Output Separable Failure Directions of DF_2

Two Output Separable Failure Directions used for the Design of DF_2	
−0.000830	−0.0013
−0.000920	−0.00088
0.001769	0.00165
0.000389	−0.00255
−0.001510	−0.00155
−0.001560	−0.00176
−0.003250	0.003311
−0.001920	−0.00022
0.001769	0.00165
−0.001560	−0.00176
−0.003250	0.003311

- *Step 2*: Conventional linear independence tests would seem to yield a linear independence of all the eight-failure directions; however, a more careful inspection using singular values has concluded that only six of them are truly linearly independent. The remaining two are also linearly independent, but require a separate Diagnosis Filter (DF_2). These two sets of vectors (failure directions) were shown in ***Step 1***. Here, the interference of modelling inaccuracies and round-off errors influence the outcome values of the combination of the loops A and C. Matrix D should be chosen such that each B^i will yield a residual solely along its projection CB^i (or, in this case, CAB^i) in the event space F. When one of these vectors is given an unidirectional residual, it is referred as output stationary. There will be v_i freely assignable eigenvalues of $[A - DC]$ associated with B^i and F. The event space F comprises of all such 'projected' failure directions in output space.

- *Step 3*: Determine the event order v of each f, and compute basis vectors for each event space. If an event space is one-dimensional (which may frequently be the case) the only basis vector is the event vector itself. The event order for each of these failure directions is checked to be equal to one. Thus, each failure direction f_i is a basis vector for its own event space.

- *Step 4*: Any set of output separable vectors that leaves all the eigenvalues freely assignable is non-restrictive. If the number of the output separable vectors equals the rank of C (the number of independent actuators), then the number of nonassignable eigenvalues equals to the difference between the dimension of the model and the

sum of the event orders, i.e. the difference $n - \sum_{i-1}^{S} v^i$. If there is no difference, all the eigenvalues are assignable. If the number of independent measurements is the maximum possible and equal to the number of state variables of the model, all the possible sets of output separable vectors are non-restrictive (i.e. all eigenvalues are assignable). This is a fully measured system.

- *Step 5*: The equations that D must satisfy are derived from the set $\{f_1, \ldots f_t\}$. These equations are based on the sets of basis vectors assembled in ***Step 3*** and contain as parameters the assignable eigenvalues of $[A - DC]$. The event space is defined by $f = f_0 + f_1$, Cf_0 and $Af_0 = f_1$. The set of basis vectors used for computing D is $\{g, Ag, \ldots A^{v-1}g\}$. For g, the event generator of f, we have:

$$Cg = 0, CAg = 0, \ldots CA^{v-1}g = Cf$$

which can be written in terms of the basis vectors as:

$$f = \alpha_1 g + \alpha_2 Ag + \cdots + \alpha_v A^{v-1} g$$

i.e. where f is in its own event space. Most often, $\alpha_v = 1$ and $v = 1$, or at most, 2 or 3. This happens because the reference model is constructed with as few state variables as possible and with as many measurements as possible, which in turn minimizes the order of the event space. Since, each f_i is a basis vector (***Step 3***), then $f_i = g_i$. The operations listed there ultimately aim at expressing each f_i in terms of a set of basis-vectors in the event space.

- *Step 6*: The calculation of D is very simple for our specific application, as D satisfies an eigenvalue equation where the eigenvectors are the f_i and the eigenvalues can be set by the designer. Choosing all eigenvalues λ_i to be consistent with the overall system time constant characteristics and taking into account that the filter has to function at a faster time constant (one order of magnitude): $\lambda_i = -0.005$ $(i = 1,2,\ldots 8)$ for DF_1 and DF_2 for relevant actuators, where: $D_1 f_i = -0.005 f_i$ where $(i = 1,2,\ldots 6)$ and $D_2 f_i = -0.005 f_i$ where $(i = 1,2)$, respectively.

4.2 Design Method 2: The Euresis Filter

4.2.1 Theoretical Foundation

The idea behind the Euresis Filter (Multiple Model FDI) is simple. One constructs as many replica models of the plant as there are failures, each reflecting, at the right place, the specific failure. These models are almost identical, except for the place where the specific failure is represented, as if it were the true plant. In our specific application, there are eight possible failures, regardless of severity, corresponding to each thruster. There is also the possibility of no failure, which is also represented by one (unfailed) model, for a total of, originally, nine models. These models can be designed as, H_2 or H_∞ based optimal observers.

In the first case, the models mentioned above, get the form of Kalman Filters, which can optimally handle stochastic disturbances and measurement noise. In the second case, the H_∞ observers can optimally handle certain types of structured and unstructured uncertainty. In this dissertation, Kalman Filters are selected for the Euresis Filter design method.

A bank of models for FDI is created with the rationale that the residual will be minimal for the specific model whose failed representation matches the specific plant failure at least in direction, as represented by the failed B column in the case of a thruster failure. Therefore, if model #2 in the bank has a failed B^1 column and the plant happens to have a failed thruster #1 regardless of actual severity, then the residual between their two outputs should be minimal, as compared to the residuals formed between the plant output and the rest of the models in the bank, since there is a model match.

Figure 4.4 is a schematic representation of this concept.

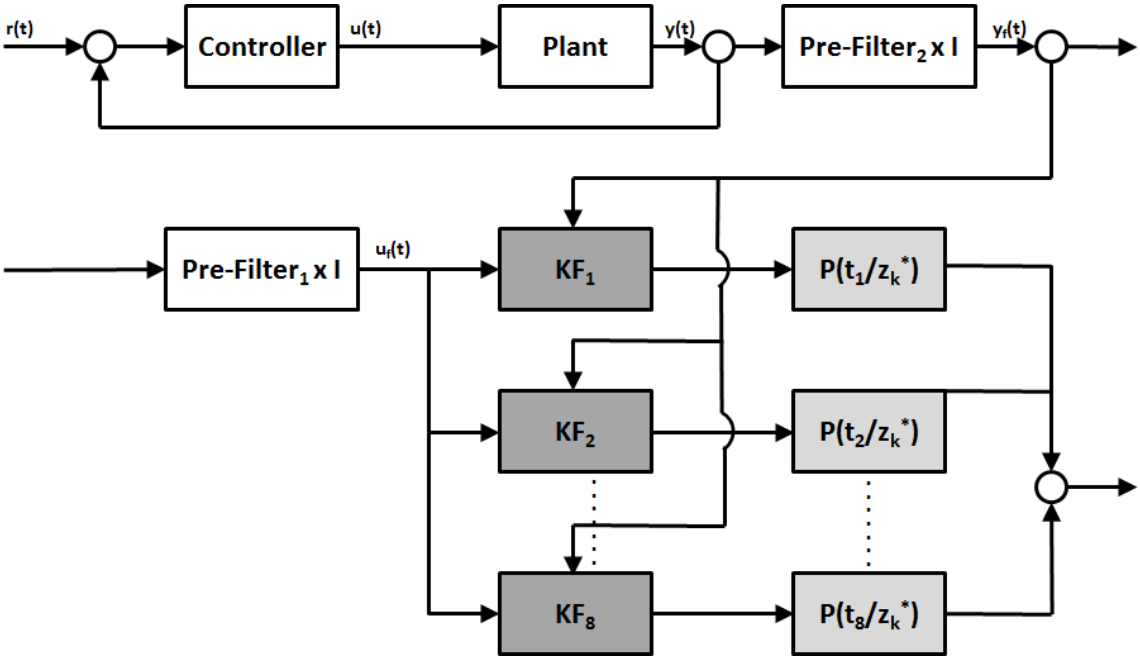


Fig. 4.4 The Structure of the Euresis Filter

Each filter in Figure 4.4 computes its own conditional probability density function (weight), and the one with the maximum probabilistic weight $p(t_i/z_k^*)$, where $i = 1,\ldots 9$ and z_k^* are the measurements, is precisely the one that corresponds to the particular plant mode with or without a failed thruster and $j = 1,\ldots 8$ are the corresponding Kalman Filter models of each thruster. Thus, a failure is identified by convergence to the most probable member in the set, which corresponds to the actual failure.

The probability weight for each KF (possible failure eventuality) is computed according to the formula:

$$p(t_i/z_k^*) = \frac{p(z_k^*/t_i)p(t_i)}{\sum_{j=1}^{8} p(z_k^*/t_j)p(t_j)} \qquad \text{where} \quad i = 1,\ldots 9 \tag{4.4}$$

with

$$p(z_k^*/t_i) = \frac{1}{(2\pi)^{\frac{k}{2}}|C(i)|^{\frac{1}{2}}} \exp\left[-\frac{1}{2\left(z_k^* C(i)^{-1} z_k^*\right)}\right] \tag{4.5}$$

where $C(i)$ is each filter's running covariance.

The Multiple Model concept is attractive as it directly assigns each fault to a corresponding KF model from the bank of filters for all the possible failure modes. However, if the faults are not precisely known, in terms of both direction and magnitude, convergence to the correct fault model becomes challenging, and it depends on the nature of the underlying process, which may exacerbate the problem. This is completely analogous to the convergence problems encountered in the Multiple Model (Adaptive) Kalman Filters of the 1970s, in cases where none of the chosen filters exactly matched the actual (unknown) parameter value of the estimated plant. A remedy to this can be a dramatic increase in the number of models used to capture all possible eventualities and fine gradations fault magnitudes. This could be a very costly and computationally intensive alternative. Unfortunately, after extensive testing, even this did not yield any improvement on the algorithm's ability to identify partial failures.

4.2.2 Outline of the Euresis Filter Operation

In Figure 4.4, each Kalman Filter of the Euresis technique has been designed with precisely, one thruster fault and its corresponding matrix reflects this failure. The conditional probabilities $p(t_i/z_i^*)$ correspond to the probability that thruster t_i has failed given the current measurement z_i^*. Eventually, the scheme will converge to the model that represents the true plant failure, as its corresponding conditional probability will have attained the highest value of all eight, (i.e. it will converge to the most probable model). In classical Multiple Model (Adaptive) Kalman Filtering, the estimate is synthesized as the weighed sum of the individual estimates in the bank of KFs where the conditional probabilities serve as the weights. Using this architecture for FDI and keeping track of how the conditional probabilities evolve, constitutes our primary concern, as this will be the indicator of which model matches the failed plant.

The outline of the necessary steps for calculating these conditional probabilities, which are continually updated with each new measurement, is presented in section 4.2.2. All the steps are initialized at the same value (1/8), or we can reflect any available prior in setting these initial conditions. From classical theory, the optimal estimate $\hat{x}_k$, at instant k, is the conditional mean, calculated as:

$$\hat{x}_k = \int x p(x/z_k^*) dx \tag{4.6}$$

Using marginal probability density functions, we derive that:

$$\hat{x}_k = \int x \int p(x,t/z_k^*)dtdx \tag{4.7}$$

where t stands for failed thruster and denotes the time instants when measurements are taken. Using Bayes rule:

$$p(x,t/z_k^*) = p(x/t,z_k^*)p(t/z_k^*) \tag{4.8}$$

substituting equation (4.8) into equation (4.7), while taking sums, since the probability density functions are discrete:

$$\hat{x}_k = \sum_{i=1}^{8} \hat{x}_k(t_i)p(t_i/z_k^*) \tag{4.9}$$

Equation (4.9) suggests a bank of Kalman Filters, parameterized by t, the thrusters' faults, as presented in Figure 4.4. Therefore:

$$p(t_i/z_k^*) = \frac{p(z_k^*/t_i)p(t_i)}{p(z_k^*)} \tag{4.10}$$

but:

$$p(z_k^*) = \sum_{j=1}^{8} p(z_k^*,t_j) = \sum_{j=1}^{8} p(z_k^*/t_j)p(t_j) \tag{4.11}$$

and, therefore, equation (4.11) can be modified to:

$$p(t_i/z_k^*) = \frac{p(z_k^*/t_i)p(t_i)}{\sum_{j=1}^{8} p(z_k^*/t_j)p(t_j)}, \qquad i = 1,\dots 8 \tag{4.12}$$

$p(t_j)$ is presumed as known and all the other processes to be Gaussian, in which case:

$$p(z_k^*/t_i) = \frac{1}{(2\pi)^{\frac{k}{2}}|C(i)|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\left(z_k^* C(i)^{-1} z_k^*\right)\right] \tag{4.13}$$

where $C(i)$ is the covariance matrix of z_{ki}^*, conditioned on t_i and $z_k^* = z_k - \hat{z}_k$, including all measurements up to time k. Next, $C(i)$ is expressed in terms of quantities nor-

mally computed on-line in the Euresis Filter architecture by the individual Kalman Filters.

The term $p(z_k^*/t_i)$ is initially expressed as the product of the conditional density functions; for simplicity, t_i is temporarily omitted.

$$\begin{aligned} p(z_k^*) &= p(z_k, z_{k-1}, \dots z_1) \\ &= p(z_k, z_{k-1}, \dots / z_1) p(z_1) \\ &= p(z_k, z_{k-1}, \dots / z_1, z_2) p(z_2/z_1) p(z_1) \\ &\vdots \\ &= p(z_k, z_{k-1}, \dots z_1) p(z_{k-1}/z_{k-2}, \dots / z_1) p(z_2/z_1) p(z_1) \end{aligned} \tag{4.14}$$

Note that each factor above is the probability density of a measurement given all the previous measurements; each must be normal. In addition, the measurement equation has the form:

$$z_j = c_j x_j + \theta_j \tag{4.15}$$

Therefore, the covariance matrix associated with the product form is:

$$C'(i) = \begin{bmatrix} \left[C_k \Sigma_{k/k-1} C_k^T + \Theta_k\right]_i & & & \\ & \left[C_{k-1} \Sigma_{k-1/k-2} C_{k-1}^T + \Theta_{k-1}\right]_i & & \\ & & \ddots & \\ & & & \left[C_1 \Sigma_{1/0} C_1^T + \Theta_1\right]_i \end{bmatrix} \tag{4.16}$$

The above diagonal terms are used in the gain computations of the Kalman filters. Each KF keeps updating its own product of terms $[C_k \Sigma_{k/k-1} C_k^T + \Theta_k]$ in each step of the process. These terms will be different for each Kalman Filter, due to the t_i and they are pre-computable. When considering the term $z_k^{*T} C(i)^{-1} z_k^*$, where $C(i)^{-1}$ is diagonal, each variate is a biased Gaussian random variable with the bias being the *α priori* estimate of the random variable:

$$z_k^{*T} C(i)^{-1} z_k^* = \frac{(z_k - \hat{z}_k)^2}{C_k \Sigma_{k/k-1} C_k^T + \Theta_k} + \frac{(z_{k-1} - \hat{z}_{k-1})^2}{C_{k-1} \Sigma_{k-1/k-2} C_{k-1}^T + \Theta_{k-1}} + \cdots + \frac{(z_1 - \hat{z}_1)^2}{C_1 \Sigma_{1/0} C_1^T + \Theta_1} \tag{4.17}$$

The numerators above are measurement residuals of the KFs weighted by $C\Sigma C^T + \Theta$. Based on the previous analysis, the steps leading to the computation of each filter's probability density function, $p(t_i/z_k^*)$ which is the quantity of interest in an FDI setting are listed below:

- Step 1:
Each filter computes its own $p(z_k^*/t_i)$ using the running product of $C\Sigma C^T + \Theta$.
- Step 2:

$$\text{Get:} \qquad p(z_k^*/t_i)p(t_i) \tag{4.18}$$

$$\text{Form:} \qquad \sum_{j=1}^{8} p(z_k^*/t_j)p(t_j) \tag{4.19}$$

- Step 3:

$$\text{Form:} \qquad \frac{p(z_k^*/t_i)p(t_i)}{\sum_{j=1}^{8} p(z_k^*/t_j)p(t_j)} \tag{4.20}$$

which is the desired quantity for the fault isolation.

4.3 Design Method 3: The Euphoria Filter

4.3.1 Theoretical Foundation

The Euphoria-Filter design methodology involves the selection of an appropriate open loop representation $[A, B, C, D]$ of the plant. The subsystems under consideration include the DF and SUS loops as those used in the FDI process [15, 29]. The columns of matrix B provide sufficient vectors for all the actuators to be monitored. The generic design framework, for both H_2 and H_∞ designs for filtering as well as control, for a plant with output feedback is outlined in Figure 4.5

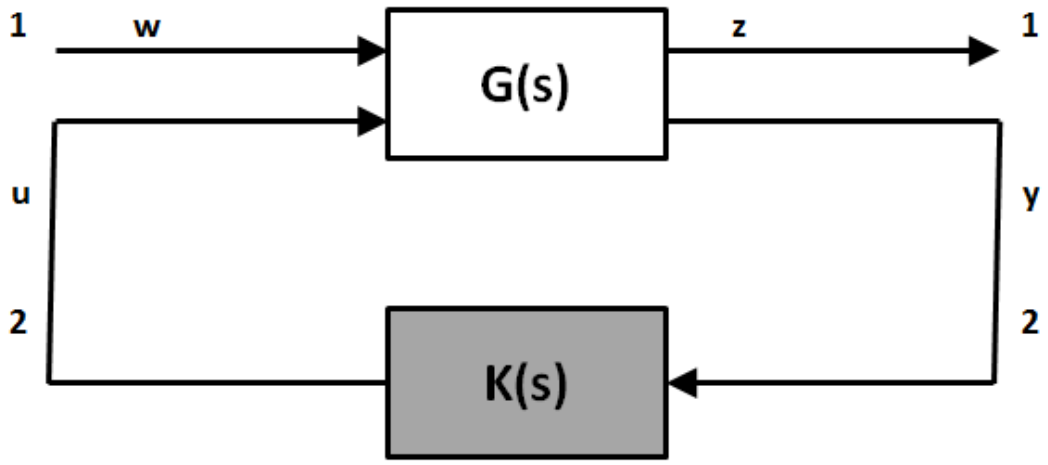


Fig. 4.5 The generic design framework for H_2, H_∞

where $G(s)$ is the plant, $K(s)$ a (dynamic) filter or compensator, depending on whether we are interested in filtering or control, while G and K are assumed to be real, rational and strictly proper. For both the H_2 and H_∞ problems, K must provide internal stability for the overall system; this is by the mathematical construction of the solution. Such a K is considered admissible and is synthesized from the corresponding solutions of the filtering or control algebraic Riccati equations, as the case may be, respectively. There, u is the input to the plant, y its measured output, and z represents the performance variables of interest, usually a subset of or the entire plant state, or linear combinations of those. In our case, z represents the best estimate of y and w denotes the plant disturbances and noise. Thus, we are interested in minimizing $\|Tzw_2\|$ and $\|Tzw_\infty\|$ respectively, depending on the

problem specifications. Other transfer functions of interest are $\|Tyu\|$ and $\|Tzu\|$, which are more relevant to the control problem, rather than filtering, and are not considered here.

The indices 1, 2 are used for denoting the four resulting TFs, i.e.:

$$G_{22}(s) = G_{yu}(s) = C_2(sI - A)^{-1}B_2 \tag{4.21}$$

In light of the above, the transfer function G (in a Linear Fractional Transformation (LFT) setting) is assumed to be of the form:

$$G(s) = \left(\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & D_{21} & 0 \end{array}\right) \tag{4.22}$$

which is the standard representation of the overall system TF in this mathematical framework. In the above, the following must hold true:

$$(A, B_1) \text{ stabilizable and } (C_1, A) \text{ detectable} \tag{4.23}$$

$$(A, B_2) \text{ stabilizable and } (C_2, A) \text{ detectable} \tag{4.24}$$

$$D_{12}^T [C_1 \;\; D_{12}] = [0 \;\; I] \tag{4.25}$$

$$\left[\begin{array}{c} B_1 \\ D_{21} \end{array}\right] D_{21}^T = \left[\begin{array}{c} 0 \\ I \end{array}\right] \tag{4.26}$$

Assumption (4.23) is required because if the (C_1, A) is detectable then the fact that $H_2 \in dom(Ric)$ is guaranteed, meaning that the corresponding Riccati equation has a semi definite stabilizing solution (i.e. which stabilizes the filter dynamics in our case). The fact that (A, B_1) is stabilizable will produce $J_2 \in dom(Ric)$. The same applies by analogy for assumption (4.24). Assumption (4.25) implies that $C_1 x$ and $D_{12} u$ are orthogonal which, in terms of the control, suggests that there is no cross term $x^T u$ in the performance index. Assumption (4.26), which is the dual of assumption (4.25), implies that the plant disturbance and sensor noise are orthogonal, i.e. uncorrelated. Here, the subscript 2 in H_2 and J_2 signifies that these are the appropriate matrices

under the $2-norm$ formulation. These are Hamiltonian matrices and are defined as follows:

$$H_2 = \begin{pmatrix} A & B_2B_2^T \\ -C_1^TC_1 & -A^T \end{pmatrix} \quad J_2 = \begin{pmatrix} A^T & -C_2C_2^T \\ -B_1B_1^T & -A \end{pmatrix} \tag{4.27}$$

where the first Hamiltonian, H_2, corresponds to the Control Algebraic Riccati Equation (CARE) and the second, J_2, to the Filter Algebraic Riccati Equation (FARE). We also note that J_2 is the dual of H_2. In terms of the above notation, the plant and filter equations become:

Plant

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1w + B_2u(t) + b^in_i(t) \\ y(t) &= C_2x(t) + D_{21}w \end{aligned} \tag{4.28}$$

Filter

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + B_2u(t) + G[z(t) - y(t)] \\ z(t) &= C\hat{x}(t) \end{aligned} \tag{4.29}$$

where:

$$G = Y_2C_2^T \tag{4.30}$$

Y_2 is the solution to the following equation:

$$AY_2 + Y_2A^T + B_1B_1^T - Y_2C_2C_2^TY \qquad \text{(FARE)} \tag{4.31}$$

and, in LFT form, the transfer function matrix of interest, $G_f(s)$, whose 2-norm is minimized, is:

$$G(s) = \begin{pmatrix} A + GC_2 & B_1 + GD_{21} \\ -B_2^T & 0 \end{pmatrix} \tag{4.32}$$

Definition 1: The 2-norm of a transfer matrix $G_{j\omega}$ is defined as (and calculated from) the following:

$$\begin{aligned} \|G(j\omega)\|_2^2 &= \frac{1}{2\pi}\int_{\infty}^{\infty} \mathrm{tr}\,(G)(j\omega) \times G(j\omega) d\omega = \\ &= \mathrm{tr}\,(CL_cC^T) = \\ &= \mathrm{tr}\,(B^T L_0 B) \end{aligned} \tag{4.33}$$

where L_c is the controllability Grammian of (A, B) and L_o the observability Grammian of (C, A), A is stable and G is strictly proper. L_c and L_o are the solutions to the Lyapunov equations:

$$\begin{aligned} AL_c + L_cA^T &= BB^T \\ A^TL_o + L_oA &= C^TC \end{aligned} \tag{4.34}$$

Specifically, the 2-norm of $G_f(s)$ is also computed from:

$$\|G_f(j\omega)\|_2^2 = \mathrm{tr}\,(B_2^T Y_2 B_2) \tag{4.35}$$

Definition 2: The infinity norm of a transfer matrix $G(j\omega)$ is defined as:

$$\|G(j\omega)\|_\infty = \sup_{\omega}[\sigma_{max} G(j\omega)] \tag{4.36}$$

where

$$\sigma_{max} = \sqrt{\lambda_{max}[G(j\omega)G(j\omega)]} \tag{4.37}$$

Remark: *In the interest of completeness, and in order to highlight the similarity in structure of the H_2 and H_∞ problems, we give below the corresponding Hamiltonians for CARE and FARE, and the Filter Algebraic Riccati Equation only, since our main focus is on Filtering.*

Then,

$$\begin{aligned} H_\infty &= \begin{pmatrix} A & \gamma^{-2}B_1B_1^TB_2B_2^T \\ -C_1^TC_1 & -A^T \end{pmatrix} \\ J_\infty &= \begin{pmatrix} A^T & \gamma^{-2}C_1^TC_1C_2C_2^T \\ -B_1B_1^T & -A \end{pmatrix} \end{aligned} \tag{4.38}$$

and the corresponding FARE is:

$$AY_\infty + Y_\infty A^T + B_1 B_1^T + \gamma^{-2} Y_\infty C_1^T C_1 Y_\infty - Y_\infty C_2 C_2^T Y_\infty = 0 \tag{4.39}$$

Fact 1: The H_2 Filter is indeed the steady state Kalman Filter.

Fact 2: The steady state Kalman Filter satisfies the KFDE given below:

$$[I + G_{KF}(s)]\,[I + G_{KF}(s)]^H = I + \frac{1}{\mu}\left[G_{FOL}(s) G_{FOL}(s)^H\right] \tag{4.40}$$

where $G_{KF}(s)$ is the KF transfer function matrix and $G_{FOL}(s)$ is the plant transfer function matrix, from disturbances to output, as given below:

$$\begin{aligned} G_{KF}(s) &= C(sI - A)^{-1} G \\ G_{FOL}(s) &= C(sI - A)^{-1} L \end{aligned} \tag{4.41}$$

with C, A, L, G corresponding to the matrices of the open loop dynamics, and μ the sensor noise intensity, according to $\Theta = \mu I$. For the purposes of using KFDE as a design framework/tool, both L and μ can be treated as design parameters. For example, in the limit as $\mu \to 0$:

$$G_{KF}(s) \approx \frac{1}{\sqrt{\mu}} G_{FOL}(s) \tag{4.42}$$

As seen in the limit of equation (4.42) the designer can further influence the filter convergence time. This is achieved by shaping the singular values of the Right-Hand Side (RHS) of equation (4.39) by the simultaneous choice of an appropriate value for the parameter μ and a structure for L, so that the RHS of equation (4.39) have the desired crossover, which directly translates into filter convergence time.

The synthesis of the filter gain G is then affected through the specific value of μ and L in the steady state Kalman Filter Riccati equation for the covariance given below:

$$A\Sigma + \Sigma A^T + L \Xi L^T - \Sigma C^T \Theta^{-1} C \Sigma = 0 \tag{4.43}$$

where A, C, L correspond to the matrices in equation (4.22) and Ξ and Θ are the corresponding noise intensities in equation (4.22). For design purposes, as implied from equations (4.40) and (4.42), the algebraic Riccati equation (4.43) can be written

as:

$$A\Sigma + \Sigma A^T + L\Xi L^T - \Sigma C^T \frac{1}{\mu} C\Sigma = 0 \tag{4.44}$$

Equations (4.43) and (4.44) are much more transparent and flexible to use in order to influence the design parameters of the H_2 Filter given its equivalence with the steady state KF. These design parameters are 'cloaked' in the Linear Fractional Transformation (LFT) representations in equations (4.31) and (4.32).

4.3.2 Outline of the Euphoria-Filter Operation

The following FDI architecture was developed with the key feature of separating geometry and dynamics: involving an H_2 Filter in conjunction with a residuals failure direction recovery and a subsequent projection algorithm onto the specific thruster failure directions (Figure 4.6). The residuals direction recovery process here is the key and a unique feature of the present algorithm (indicated in the box Residuals Direction Recover in Figure 4.6), which allows for the residuals direction but not the magnitude, to be used for fault diagnosis. The new design synthesis combines the desirable FDI robustness features of the Diagnosis Filter along with those that naturally arise in an H_2 design framework: the stability of the overall residual dynamics and the optimal disturbance and sensor noise rejection are inherent in the design. This latter feature allows for a considerable simplification of the verification and validation (V&V) process.

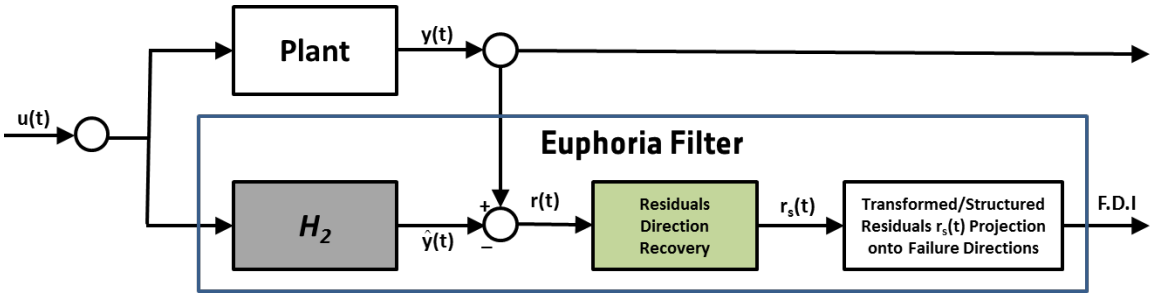


Fig. 4.6 Euphoria Filter Design Architecture

The plant dynamics are described by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + b^i n_i(t) + L\xi(t) \\ y(t) &= Cx(t) + \theta(t) \end{aligned} \tag{4.45}$$

where b^i are the thruster failure directions, corresponding to the columns of matrix B, and $n_i(t)$ their magnitudes, which are unknown. It should be noted here that, unlike in the case of the Diagnosis Filter, the plant dynamics include the presence of (stochastic) disturbances $\xi(t)$, while the output contains a measurement noise $\theta(t)$. These are explicitly considered in the filter design, and this constitutes a fundamental difference in the design procedure from the Diagnosis Filter, which was carried out in a deterministic setting.

Next, a UIO type observer, or filter, is set up, of the following form:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + G\left[\hat{y}(t) - y(t)\right] \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{4.46}$$

Note that the filter is designed according to the unfailed plant model dynamics.

Given the above mathematical descriptions, the residual dynamics are governed by the following equations, obtained by subtracting equation (4.46) from equation (4.45):

$$\begin{aligned} \dot{\tilde{x}}(t) &= \left[A - CG\right]\tilde{x}(t) + b^i n_i(t) + L\xi(t) + G\theta(t) \\ r(t) &= y(t) - \hat{y}(t) = C\tilde{x}(t) \end{aligned} \tag{4.47}$$

where $\tilde{x}(t) = x(t) - \hat{x}(t)$ is the estimation error.

The residual dynamics have essentially the same structure as in the case of the Diagnosis Filter, except from the two additional terms, $L\xi(t) + G\theta(t)$ for disturbances and sensor noise. Also, the mathematical derivation of G is different, according to the H_2 norm. Directionality information of the residuals is recovered by an adjoint observability operation and a projection onto the controllable subspaces of the failure directions, in order for FDI to be achieved. Thus, this new method, uses the direction rather than the magnitude of the appropriately transformed residuals as an FDI diagnostic. It must be emphasised that the inner product/angle criterion is much less sensitive to non-idealities, including parametric uncertainty, disturbances, and sensor noise, even though the H_2 filter, by construction, optimally rejects sensor noise and disturbances. However, because of optimality, it may be sensitive to off-design/off-nominal levels of noise. The direction/angle remains close to invariant in these circumstances and, at any rate, still provides an infallible comparison, because of the dominance of the failure direction both in the plant dynamics and in the residual. This is not the case with the magnitude, which directly reflects

substantively the effect of all such factors and is, therefore, much more vulnerable to them.

The principles of the H_2 filter design were outlined in the previous section.

Fact 1: If $[A, C]$ is observable, then $[A - GC]$ is always stable, resulting in asymptotically stable residual dynamics.

Proof: It follows directly from the optimality-based derivation of the solution for G.

Fact 2: The residual $r(t)$ consists of the transformed failure direction(s) via the output transition matrix:

Proof: From equation (4.47), the parameter r(t) is written, in terms of the solution for (t) as:

$$\begin{aligned} r(t) &= C\tilde{x}(t) \\ &= C\exp\{A - CG)(t - t_0)\}\,\tilde{x}(t_0) + \int_{t_0}^{t} \exp\{A - CG)(t - T)\}\,b^i(T)n(T)dT \end{aligned} \tag{4.48}$$

and applying the Cayley-Hamilton theorem, the equation (4.48) can be rewritten as:

$$\begin{aligned} r(t) = C\tilde{x}(t) = C\Big\{ &\exp\{(A - CG)(t - t_0)\}\,\tilde{x}(t_0) \\ &+ \int_{t_0}^{t} \left[I + \bar{A}(t - T) + \frac{1}{2!}\bar{A}^2(t - T)^2 + \cdots + \frac{1}{k!}\bar{A}^k(t - T)^k \right] b^i(T)n(T)dT \Big\} \end{aligned} \tag{4.49}$$

where $\bar{A} = A - GC$ and $k \in \mathbb{N}$.

In the above equation, the first term on the RHS converges quickly to zero, because of the guaranteed asymptotic stability of the matrix. As for the second term, if we consider $n(T)$ to be an impulse at the instant q when the failure occurs, the integral above reduces to:

$$C\left[I + (A - GC) + \frac{1}{2!}(A - GC)^2 + \cdots + \frac{1}{k!}(A - GC)^k \right] b^i(q)n(q) \tag{4.50}$$

Equation (4.50) is recognized as the output (impulse response) transition matrix of the pair $[A - CG, C]$. It is evident that the residual is dominantly the transformed failure direction b^i via the transition matrix (observable impulse response). Hence, in order to recover the original failure direction contained in the residual, it needs to be processed by the inverse of the output transition matrix, an adjoint operation that

reverts the residual to the failure onset. Consequently, the steps for realising the H_2 based FDI design are the following:

- *Step 1*: Design the H_2 Filter according to equation (FARE) repeated here for convenience:

$$AY_2 + Y_2A^T + B_1B_1^T - Y_2C_2C_2^TY_2 = 0 \tag{4.51}$$

where A is the combined DF and SUS open loop matrix, B_1 is the diagonal matrix of disturbance intensities entering the system, and $C_2 = C\Theta^{-1}$, where C is the output matrix (as given in Appendix B), and Θ is the diagonal matrix of the sensor noise intensities. Y_2 corresponds to the filter covariance matrix. Thus, $G = Y_2C_2^T$ with Y_2, C_2 as defined above.

- *Step 2*: Calculate and form the observability Grammian M_0 of $[\bar{A}, C]$, according to $M_0 = \left[C, C\bar{A}, C\bar{A}^2, \ldots C\bar{A}^k\right]$ where $k = 1,2,\ldots$ where r is the number that allows for as many independent rows as the number of failure directions (in this case 8). The variable r is a small number, usually 2-3, for a strongly observable system, such as the present one.

- *Step 3*: Transform the residuals via the observability Grammian adjoint to recover the failure direction contained in the system. From ***Fact 2***, since, $r(t) \sim M_0 b^i$ it follows that $b^i \sim adj(M_0)r$, or, simply, in this case:

$$b^i \sim \left(M_0M_0^T\right)^{-1} M_0^T \sim \left(M_0^TM_0\right)^{-1} M_0 r \triangleq r_s(t) \tag{4.52}$$

Equation (4.52) is the defining equation of the structured residual $r_s(t)$, via essentially an appropriately calculated pseudo inverse of M_0. This operation recovers the failure directions b^i in the filter residuals.

- *Step 4*: Calculate the angles θ_i subtended by the transformed residuals with failure directions (projection /inner product type operation focusing on direction/angle and not magnitude).

$$\theta_i = \arccos\left(\frac{b^{iT}\left(M_0^TM_0\right)^{-1} M_0 r}{\|b^i\|\left\|\left(M_0^TM_0\right)^{-1} M_0 r\right\|}\right) \quad \text{where} \quad i = 1,2,\ldots 8 \tag{4.53}$$

- *Step 5*: Identify the failed thruster as the one whose associated direction subtends the minimum of the angle with the transformed (structured) residual. The failed thruster direction is $\theta_k = \min_i(\theta_i) \Rightarrow b^k$ that corresponds to thruster k, where $i = 1,2,3,\ldots 8$ and $k = 1,2,3,\ldots 8$.

4.4 The Failure Severity Algorithm

Once a failure has been identified, the severity identification γ, represents an additional step in the FDI algorithm. Right at that point, a parallel identification algorithm runs with the objective of identifying the value of parameter δ as described in Section 2.3. The parameter γ in the interval $[0,1]$, represents the entire range of a thruster's failure. Equivalently, parameter δ, representing the thruster effectiveness, is complementary to the severity parameter γ; thus, for $\delta = 1$, γ is equal to zero, in the absence of failure. Values of $\delta < 1$ represent reduced thruster effectiveness, indicating some level of failure; similarly, a value of γ equal to one indicates total failure and zero thruster effectiveness, according to the defining equation below:

$$1 - \delta = \gamma, \quad \text{where} \quad \delta, \gamma \in [0,1]$$

$$\gamma = \% \quad \text{failure or failure severity}$$

As a result, any level of failure severity can be represented with the above model, thus covering a wide range of thruster failure possibilities and profiles. As an example, based on the mathematical model in equation (4.55), for $\delta = 0.3$ which means a 30% thruster effectiveness, the failure severity is 70%. If $\delta = 0$, there is total thruster loss and, according to equation (4.55), the i_{th} thruster is indeed absent from the system dynamics. The parameters γ or δ need to be non constant. Leakages can also be represented by appropriate functional representations of the parameters in the form:

$$\delta = \frac{1}{1 + \alpha t} \tag{4.54}$$

where α signifies the leakage rate, which can be set to a very low value. For example, for $\alpha = 0.0001$ the thruster effectiveness is almost 100% at the start, and is reduced to 99% at $t = 100\,\text{s}$, suggesting the presence of a very slow leakage in the thruster. Other functional representations are also possible, such as intermittent failures.

The specific functional representation of γ and δ does not affect the theoretical developments for severity identification. As the algorithm works extremely fast, it is able to keep track of the specific values of the parameter, as it evolves. At any rate, FDI and severity identification occur well below the 100^{th} second, which means that, for the particular choice of α above, the value of δ in the first 100 s is approximately 0.99. Thus, the algorithm can identify an extremely low and slow leakage failure.

For the theoretical development of the process, the plant with an actuator degradation/failure is described by the equations:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) - \gamma b^i u_i(t) \Rightarrow \\ \dot{x}(t) &= Ax(t) + Bu(t) - (1-\delta)\, b^i u_i(t) \\ y(t) &= Cx(t) \end{aligned} \tag{4.55}$$

where the exact column b^i has been identified by the FDI techniques; however, the failure severity and δ remain unknown. In the simulation environment, an adaptive identifier is therefore set, in exactly, the same form as the plant:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) - \hat{\gamma}(t) b^i u_i(t) \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{4.56}$$

The value of the parameter $\hat{\gamma}(t)$ is unknown but it can be adaptively adjusted. Objective for $\hat{\gamma}(t)$ is to converge to the true value γ, representing the failure severity. Let the error between plant and (adaptive) observer state to be defined as:

$$e(t) = \hat{x}(t) - x(t) \tag{4.57}$$

Then, from equations (4.55) and (4.56), the error dynamics are:

$$\begin{aligned} \dot{e}(t) &= Ae(t) + [\hat{\gamma}(t) - \gamma]\, b^i u_i(t) \\ &= Ae(t) + \phi(t) b^i u_i(t) \end{aligned} \tag{4.58}$$

where $\phi(t)$ is the actuator effectiveness parameter error $[\hat{\gamma}(t) - \gamma]$. Thus,

$$r(t) = Ce(t) \tag{4.59}$$

Setting up a Lyapunov function of the form:

$$V(e, \phi) = \frac{1}{2}\left(e^T P e + \phi^2\right) \tag{4.60}$$

for any: $P = P^T > 0$, then:

$$\dot{V} = \frac{1}{2} e^T \left(A^T P + PA \right) e + u_i (b^i)^T P \phi e + \dot{\phi} \phi$$

$$\text{where } Q = Q^T > 0 \text{ for } A \text{ stable} \tag{4.61}$$

$$\dot{V} = -\frac{1}{2} e^T Q e + u_i (b^i)^T e \phi + \dot{\phi} \phi$$

Since ϕ is unknown, but adjustable, it can be adjusted according to:

$$\begin{aligned} \dot{\phi} &= -e^T P b^i u_i(t) \\ \text{hence,} \qquad \dot{\hat{\gamma}}(t) &= -e^T P b^i u_i(t) \end{aligned} \tag{4.62}$$

Then, the sign indefinite terms in $\dot{V}$ disappear and

$$\dot{V} = -\frac{1}{2} e^T Q e \leq 0 \tag{4.63}$$

This means that the adaptation law, under strong signal conditions, will lead to parameter error convergence to zero, in this case requiring nonzero u and e. Indeed, for persistent $u(t)$, $e(t)$ will converge to zero only when ϕ converges to zero, thereby identifying the severity parameter value.

Since the signals $e(t)$ and $u(t)$ are available, the above parameter identification algorithm is very simple to implement from existing information on the FDI filter. Employing this additional procedure on the existing FDI designs as a simple plug-in does not interfere with the main FDI process whose main focus is to identify any failed actuator. The severity calculation flow diagram is illustrated in Figure 4.7.

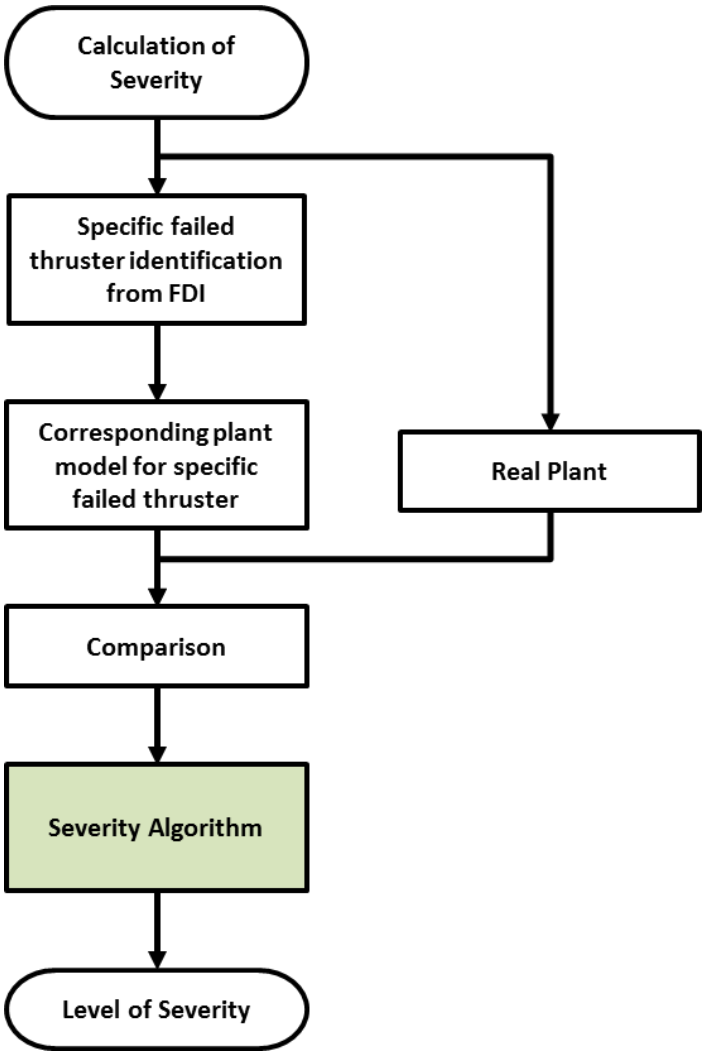


Fig. 4.7 Severity Calculation Flow Diagram

Chapter 5

Evaluation of the FDI Techniques

5.1 Validation Plan

The validation plan for the three design methods described earlier consists of two major parts:

- The performance evaluation of the designed algorithms on the true plant in a simulation environment.
- The performance evaluation of Sensitivity versus Robustness of both designs on the true plant in a simulation environment.

5.1.1 Performance Evaluation

This part of the validation plan mainly involves evaluating and checking the two proposed methods, under representative but also extreme conditions, in terms of noises, disturbances, and severity and parameter values, against the evaluation criteria described in section 2.4 and listed again below:

- False Alarm Rate $r_{f\alpha}$
- Missed Detection Rate r_{md}
- True Detection Rate r_{td}
- True Isolation Rate r_{ti}
- Wrong Isolation Rate r_{iw}
- Average detection Time t_{da}

- Average Isolation Time t_{ia}

All the above are being evaluated via extensive Monte Carlo simulations. The types of faults and their analytical models are as described in Sections 2.3 and 4.4, and include:

- Abrupt (total) thruster faults of varying constant magnitudes.
- Gradual degradation of thruster efficiency/leakage.

Both methods consider all the eight thruster faults. The primary focus is for one thruster fault at a time to be introduced at different time intervals during system operation. However, some tests will also be run on multiple thruster faults (two or three), occurring simultaneously or at different time intervals. Intermittent thruster faults are also considered. In both these cases, the testing serves more as a means of checking the robustness of the FDI schemes, rather than comprising a main thruster fault event. This means that some of the above evaluation criteria may not be fully explored in these cases.

5.1.2 Sensitivity versus Robustness

A good FDI scheme has to be adequately/efficiently sensitive to faults, yet insensitive/robust to uncertainty, whether this arises from parametric/model inaccuracies or from disturbances affecting the process, and ubiquitous measurement noises.

For example, a system parameter drift should not confuse the process of fault detection and identification and at the same time it should not be interpreted by the FDI scheme as a component failure (false alarm). All proposed designs are extensively tested for sensitivity/robustness to varying disturbance characteristics and noise statistics in simulation, as well as analytically, where possibly.

For LTI systems, a wealth of parameter sensitivity methods exists [30] that readily yield the system state sensitivity with respect to a particular parameter, in real time, as a signal obtained at specific points from a replica sensitivity model. Another important sensitivity test will also concern the parameters in the individual failure directions.

In addition to simulation or analytical sensitivity methods, the individual failure directions can also be addressed using a variety of machinery from linear algebra, including SVD decomposition. Based on the previous findings, the proposed designs

are fine-tuned to ensure a maximum robustness to uncertainty, while maintaining maximum sensitivity to parametric variation and maximal accuracy of fault detection, fault identification, and severity magnitude identification.

Figure 5.1, Figure 5.2, and Figure 5.3 schematically represent the fine-tuning approach employed in each method.

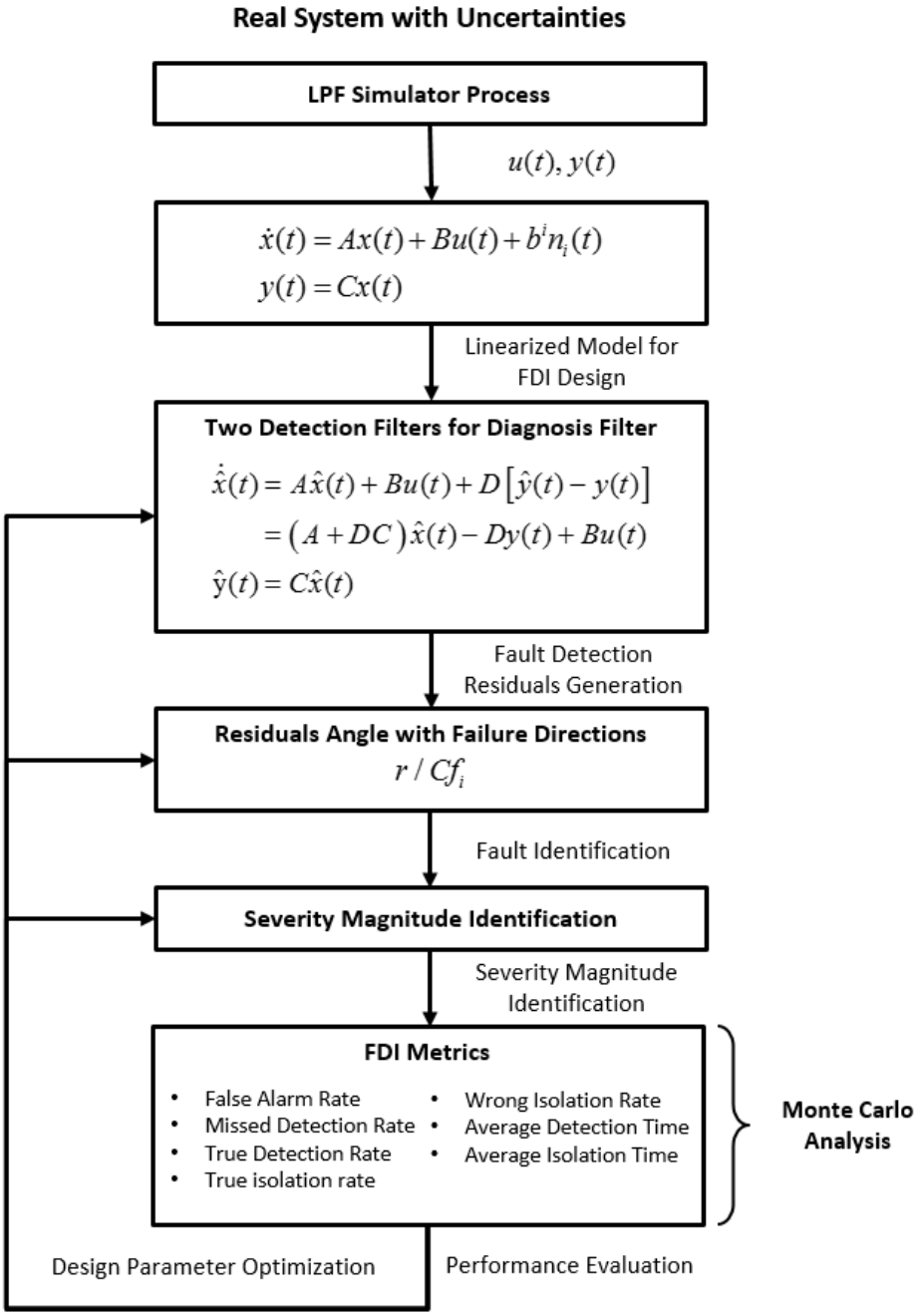


Fig. 5.1 Diagnosis Filter - Tuning Methodology

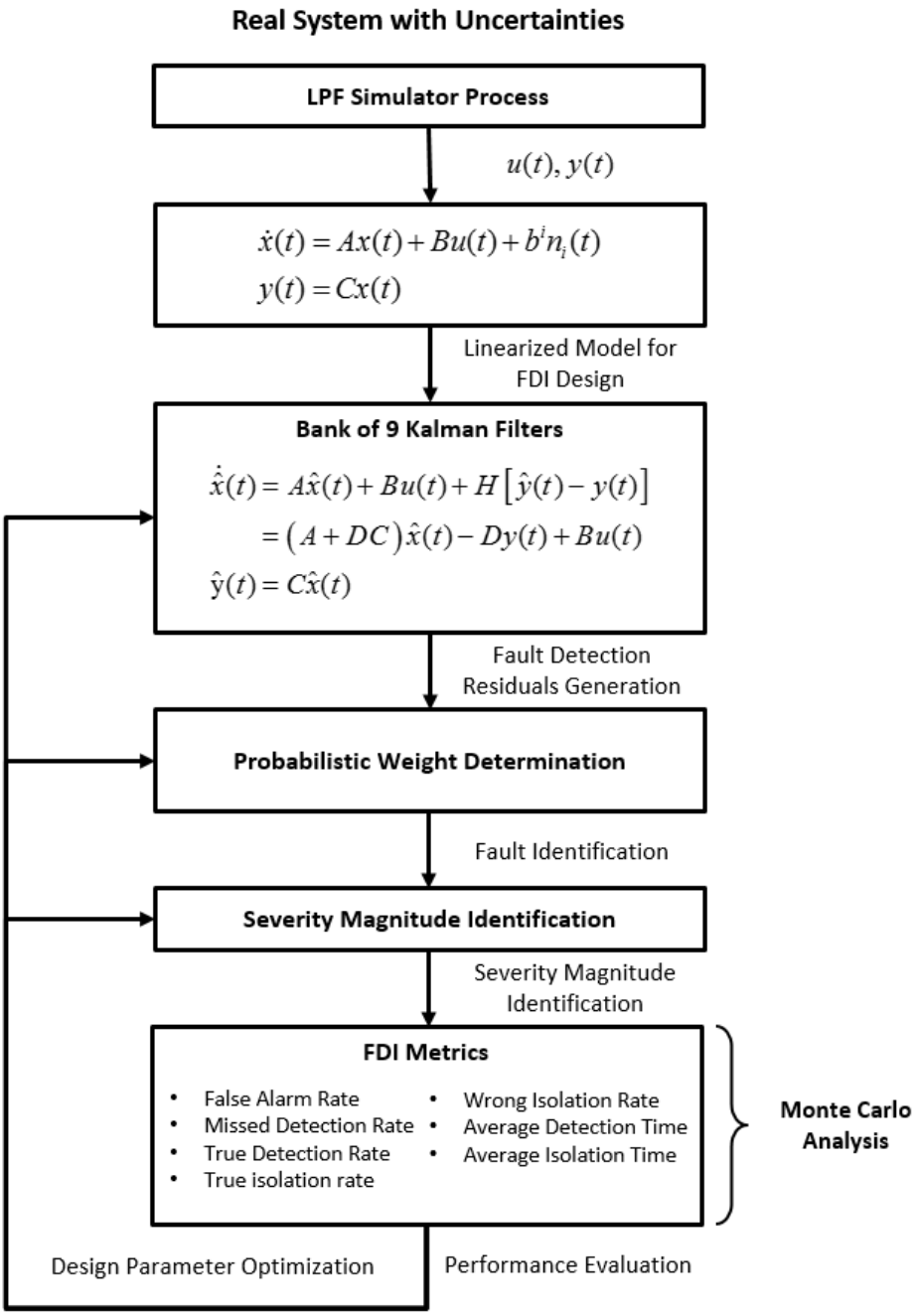


Fig. 5.2 Euresis Filter - Tuning Methodology

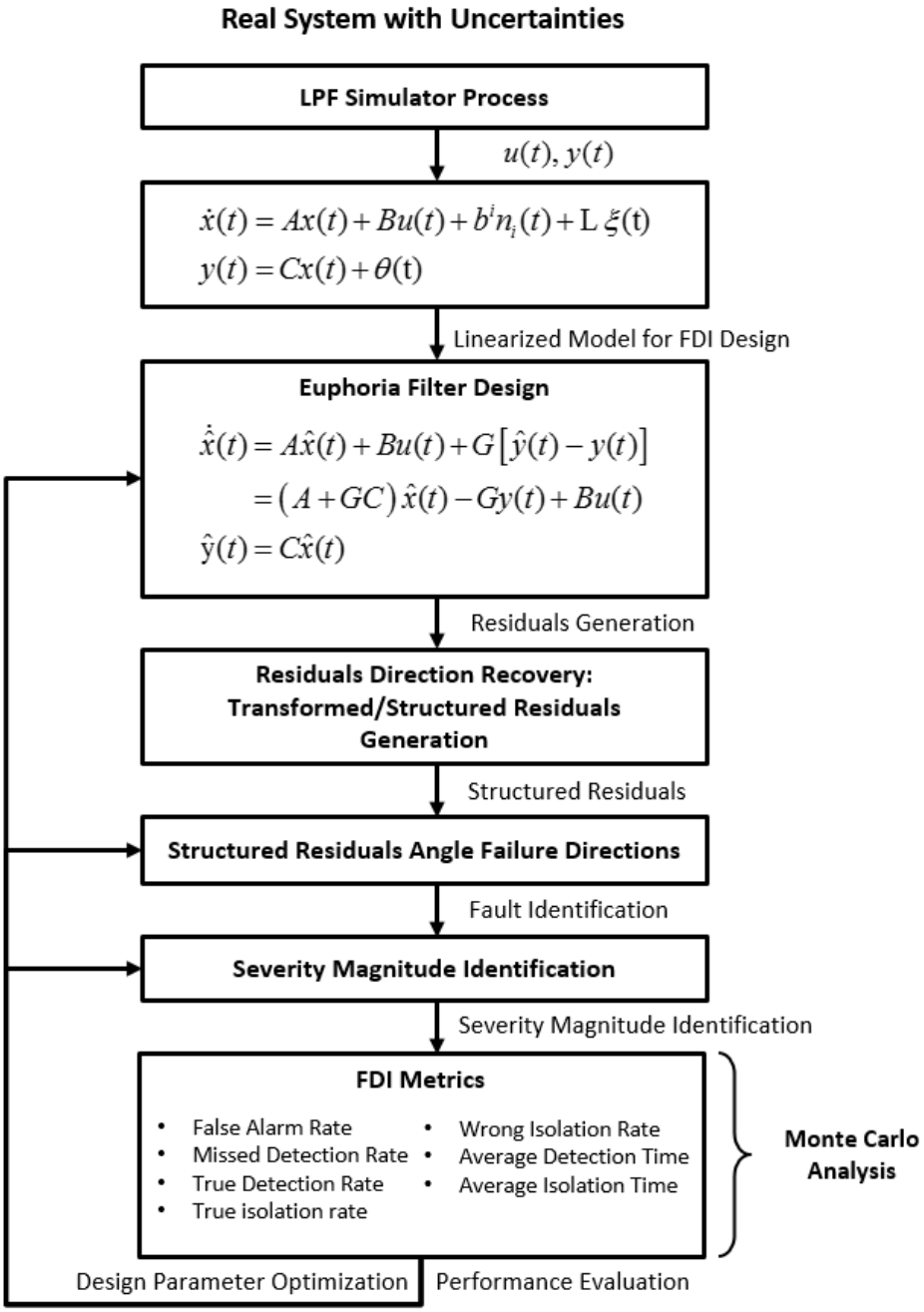


Fig. 5.3 Euphoria Filter - Tuning Methodology

5.2 Design Method 1: Diagnosis Filter

Individual trial runs of the Diagnosis Filter in the simulation environment, complete with all the disturbances present in the LPF simulator, are presented in this section.

All these runs are evaluated for a specific trimming point, assuming that a linear model is available, as described in the state space descriptions in Appendices A and B. Once the plant has reached its steady state (trimming) as implemented in the simulation environment, a total failure is introduced after 10 s, for each individual thruster, one at a time. The results and the colour designated for each thruster plot, are as follows:

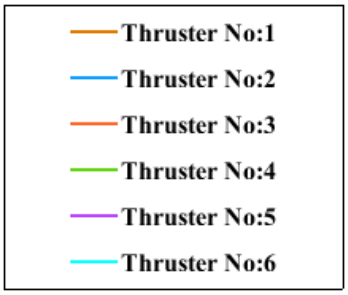


Fig. 5.4 Diagnosis Filter - Thrusters Colour Code

The results for each failure occurrence are illustrated by the plots of the time evolution and of the angles the residual makes with each pre-specified failure direction, as represented by the six output separable vectors in Section 4.1.2.

It should be noted that although the faults are identified correctly, the angles between the Diagnosis Filter residual and the corresponding fault direction are not zero. This is attributable to the presence of disturbances and measurement noise, as well as due to plant modelling inaccuracies (cross couplings that have been neglected, implementation induced delays, discretization effects, etc.). However, the distance of the angle corresponding to the failed directions from the others is safely/adequately large, which is very important from a robustness perspective.

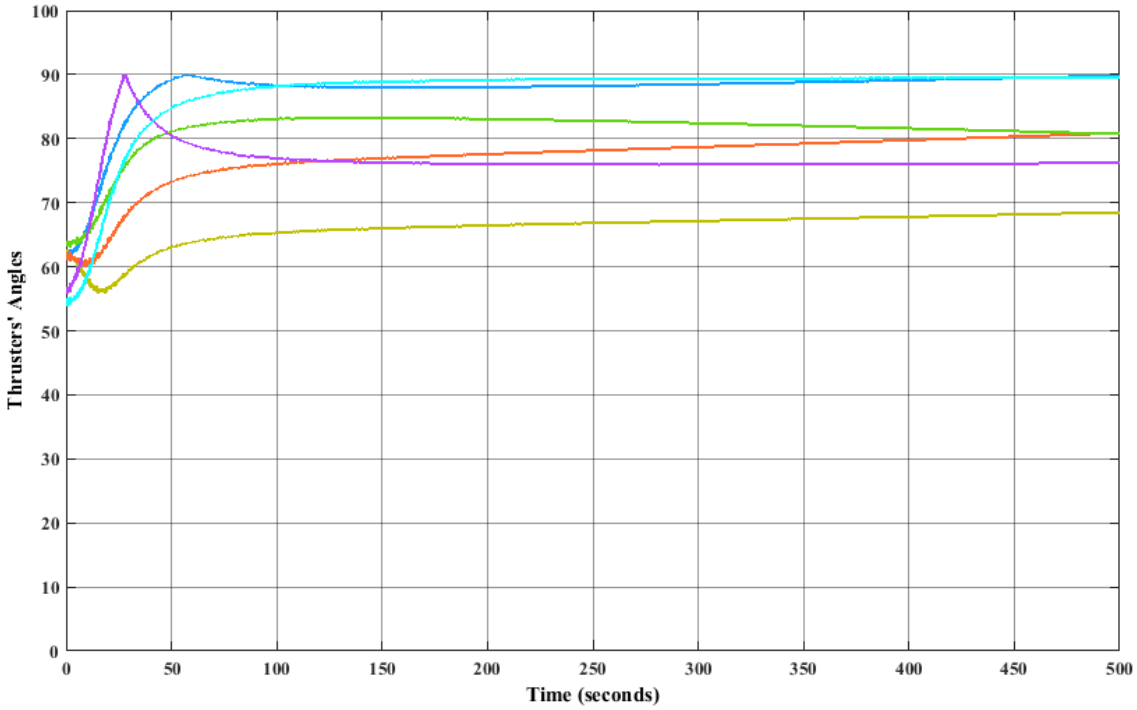


Fig. 5.5 Diagnosis Filter - Thrusters without Failure

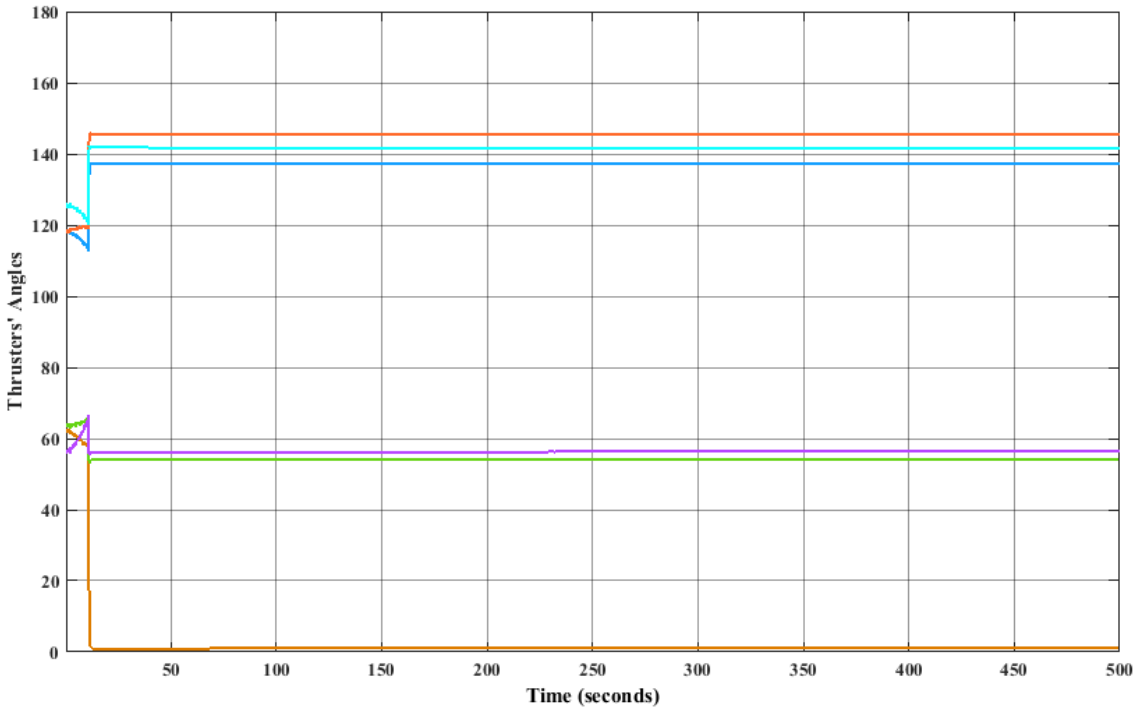


Fig. 5.6 Diagnosis Filter - 1^{st} Thruster Fault (Total Failure)

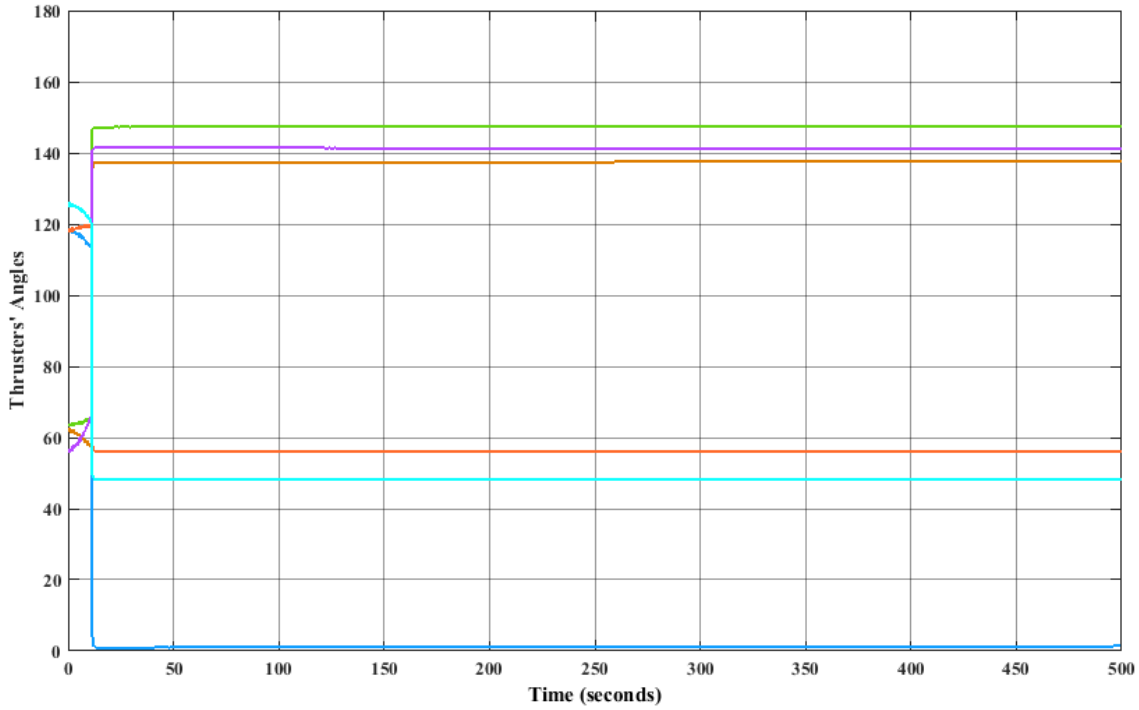


Fig. 5.7 Diagnosis Filter - 2^{nd} Thruster Fault (Total Failure)

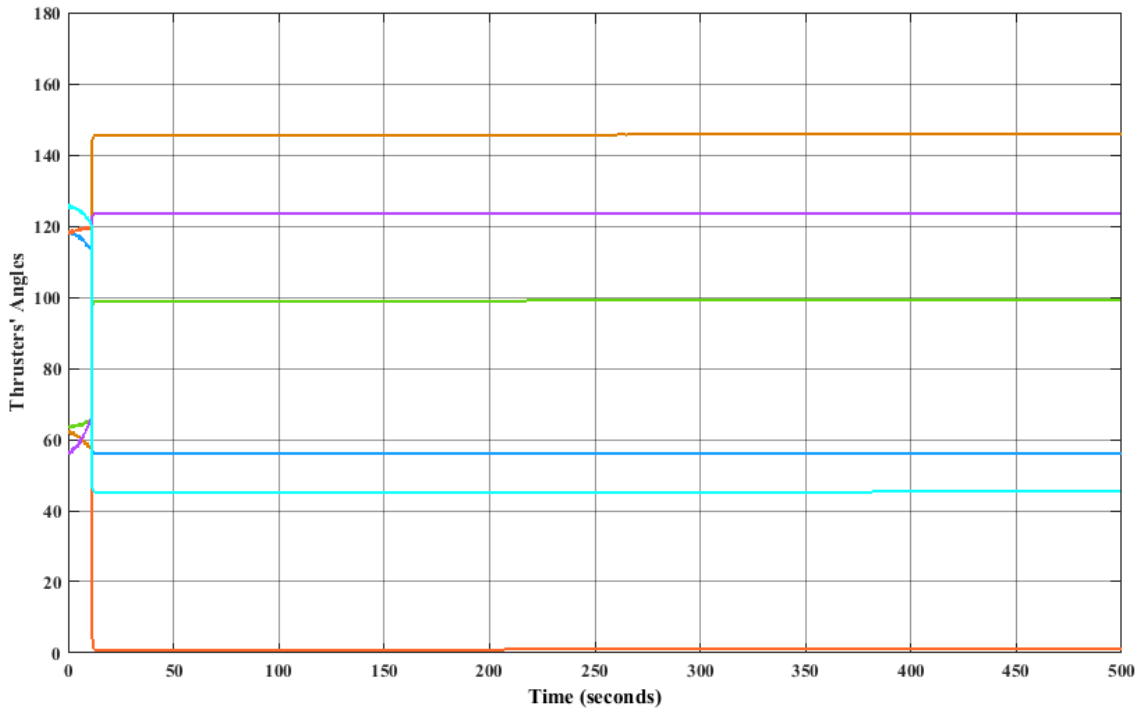


Fig. 5.8 Diagnosis Filter - 3^{rd} Thruster Fault (Total Failure)

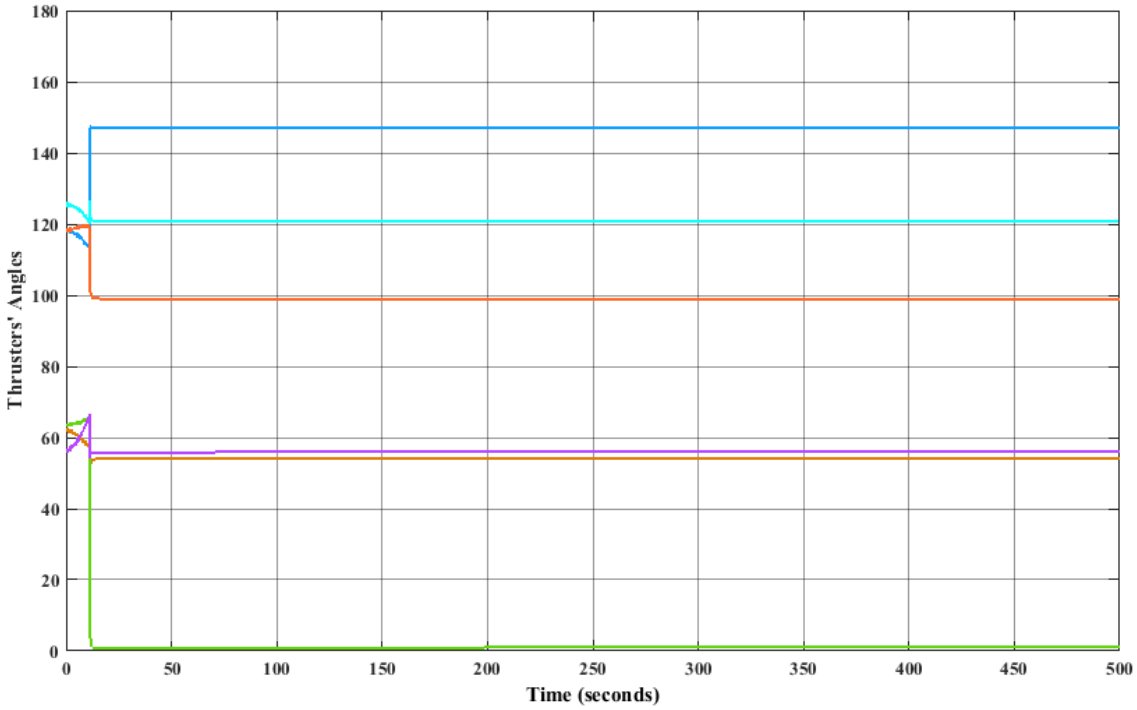


Fig. 5.9 Diagnosis Filter - 4^{th} Thruster Fault (Total Failure)

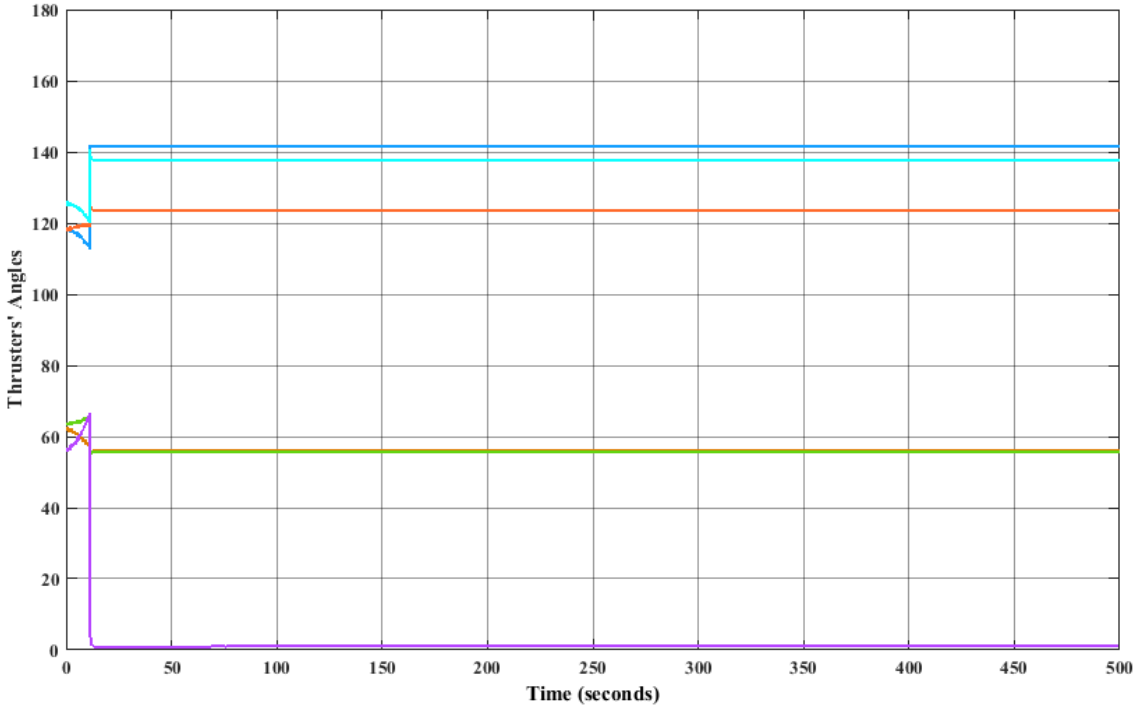


Fig. 5.10 Diagnosis Filter - 5^{th} Thruster Fault (Total Failure)

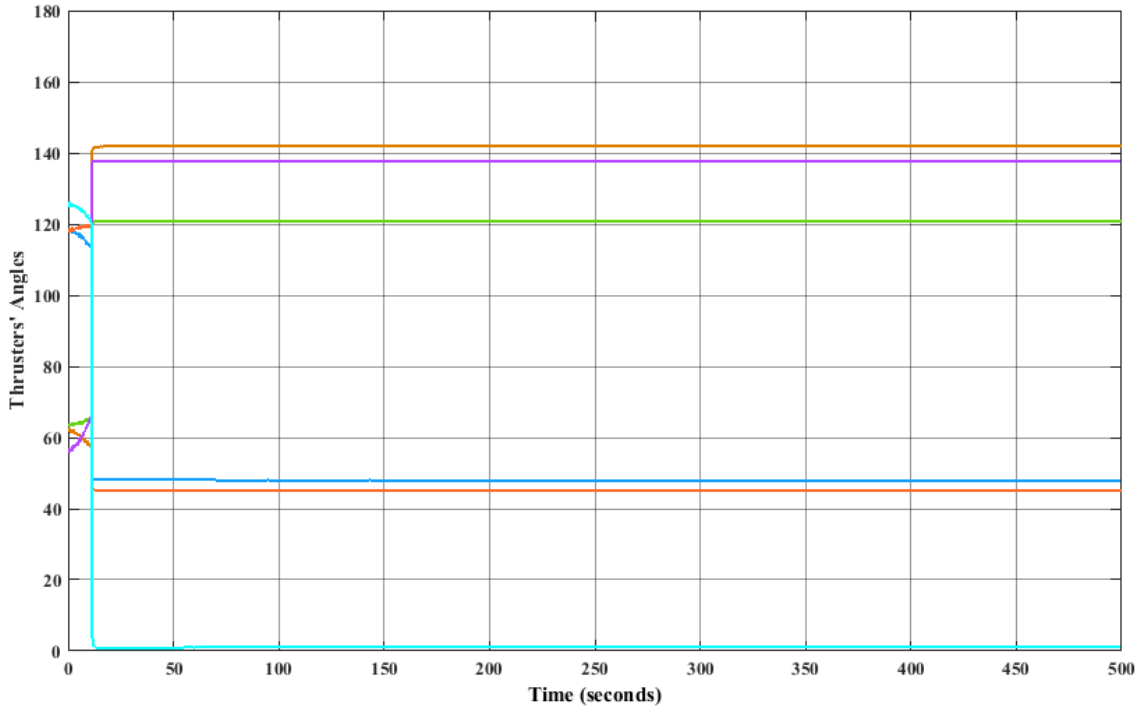


Fig. 5.11 Diagnosis Filter - 6^{th} Thruster Fault (Total Failure)

In all the preceding runs only one thruster total failure at a time was considered. The failure detection and fault identification time was significantly fast, around 10 s after fault inception.

In the following runs, two thrusters failed simultaneously at 1 s past the system reaching its proper trimming point. Again, the detection occurs very fast, at about 10 s. Although a two thruster simultaneous failure is very rare, it was nevertheless included in our testing of the method as a preliminary indicator of the filter's robustness, before the actual validation process. A three simultaneous thruster failures scenario was also considered.

Although the performance outcome was good for the dual and triple simultaneous thruster failures tested, these are not all exhaustive. The combinations that are not shown did not perform well, suggesting that some robustness issues may emerge during the validation of robustness and sensitivity to various types of uncertainty. This warrants the further fine-tuning of our existing designs.

It should also be noted here that only the nominal results for the first six failure directions relative to thrusters # 1, 2, 3, 6, 7, 8 are presented, as these nominal runs were carried out before the incorporation of the second Diagnosis Filter. However, the validity and completeness of the presented results is in no way compromised,

as the behaviour observed in subsequent runs with the complete design was of the same nature. The complete design (all eight thrusters' failure detection capability) was run extensively to establish performance bounds.

The strength of the Diagnosis Filter methodology lies in the fact that the stationarity of the failure's directionality is preserved by the specific construction of the detection filter. The filter is not expected to be sensitive to disturbances, as their directionality is not stationary, especially when they are stochastic. Nor can it be expected that these disturbances will coincide with a specific event vector; such a scenario would be extremely unlikely in any case. In fact, as subsequent extensive sensitivity and robustness tests established, the Diagnosis Filter was remarkably robust to parametric variation. Although an analytical proof is very difficult to obtain when it comes to directionality issues (as has been the case with all modern control design methods where directionality could not be analytically incorporated in the design methods), further investigation of the results obtained in the Monte Carlo runs showed precisely this.

For a 10% random variation in each element of the thruster vector directions, the angle between the nominal and the perturbed direction ranged from 1 to 5 degrees for all columns in matrix B. However, the most impressive findings concern the angles between the columns of the Diagnosis Filter matrix and their corresponding columns of matrix B, whether nominal or perturbed, all were very close to 90 degrees, with minimal deviations, further corroborating the inherent robustness of the specific design.

In conclusion, the filter ability to robustly identify failures is attributable to its strong preservation of the directionality of the failure, which cannot be easily confused by disturbances or parametric variation. As a consequence, no thresholds need to be accurately defined, other than boosting the overall FDI system disturbance rejection via additional filtering (if necessary). This additional filter decision is not based on the magnitude of a residual, which can be very sensitive to contamination but, rather, to its directionality, which is a much more robust property.

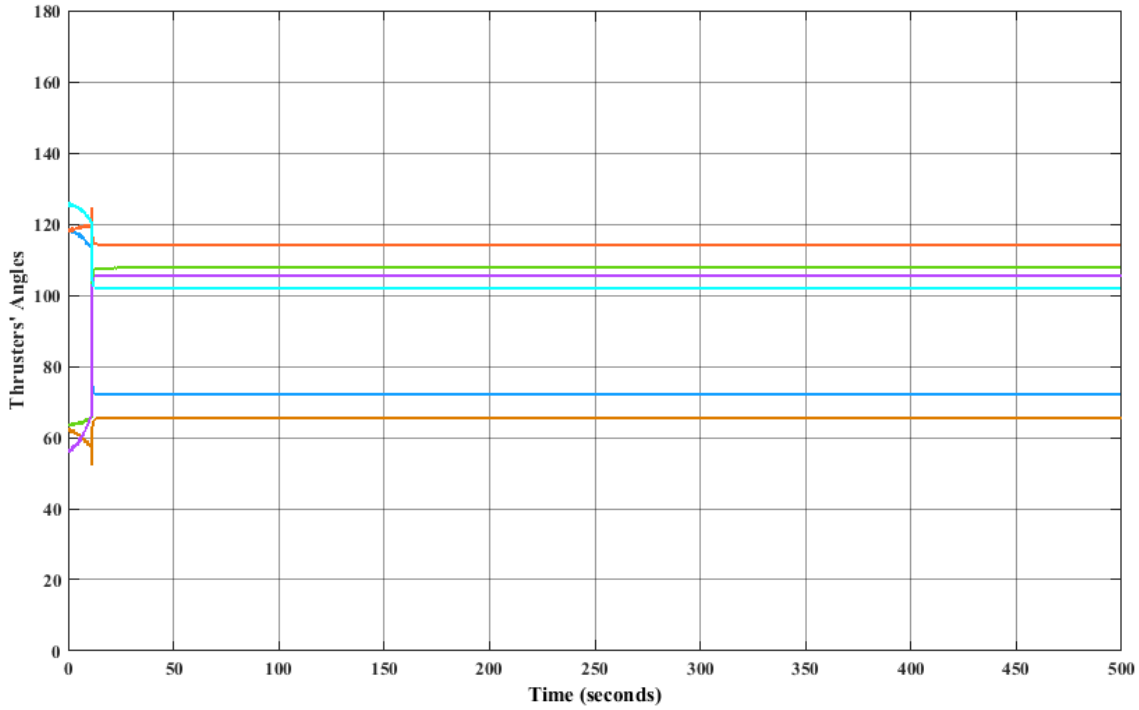


Fig. 5.12 Diagnosis Filter - 1^{st} and 2^{nd} Thruster Fault (Total Failure)

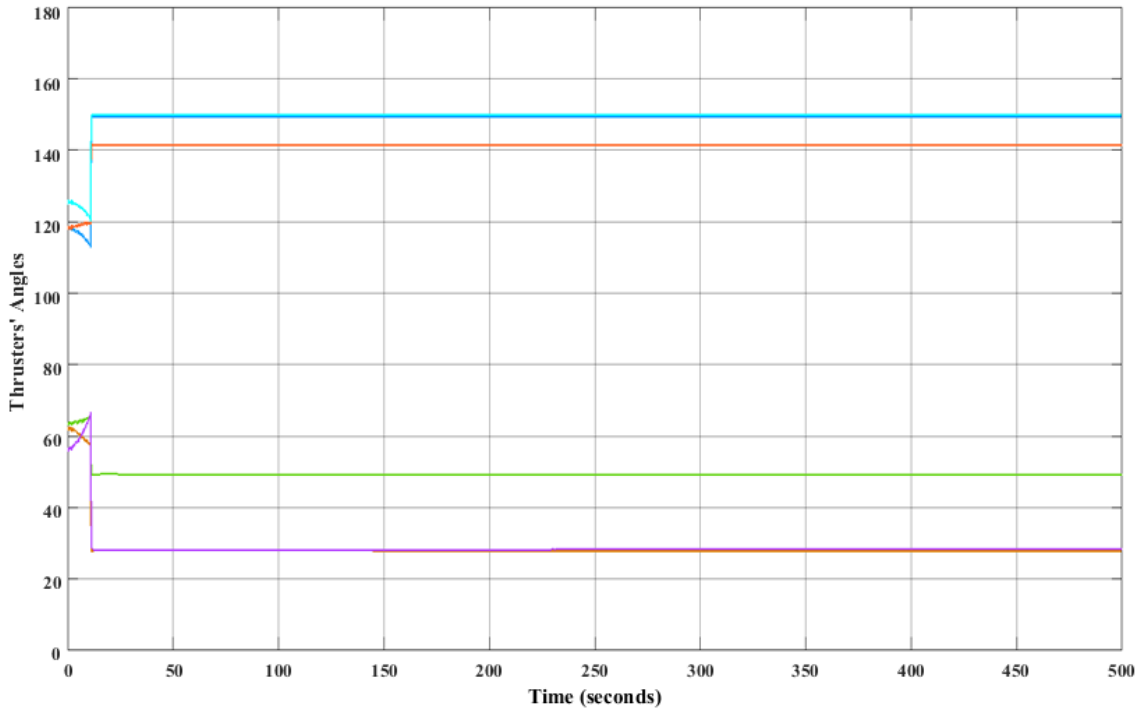


Fig. 5.13 Diagnosis Filter - 1^{st} and 5^{th} Thruster Fault (Total Failure)

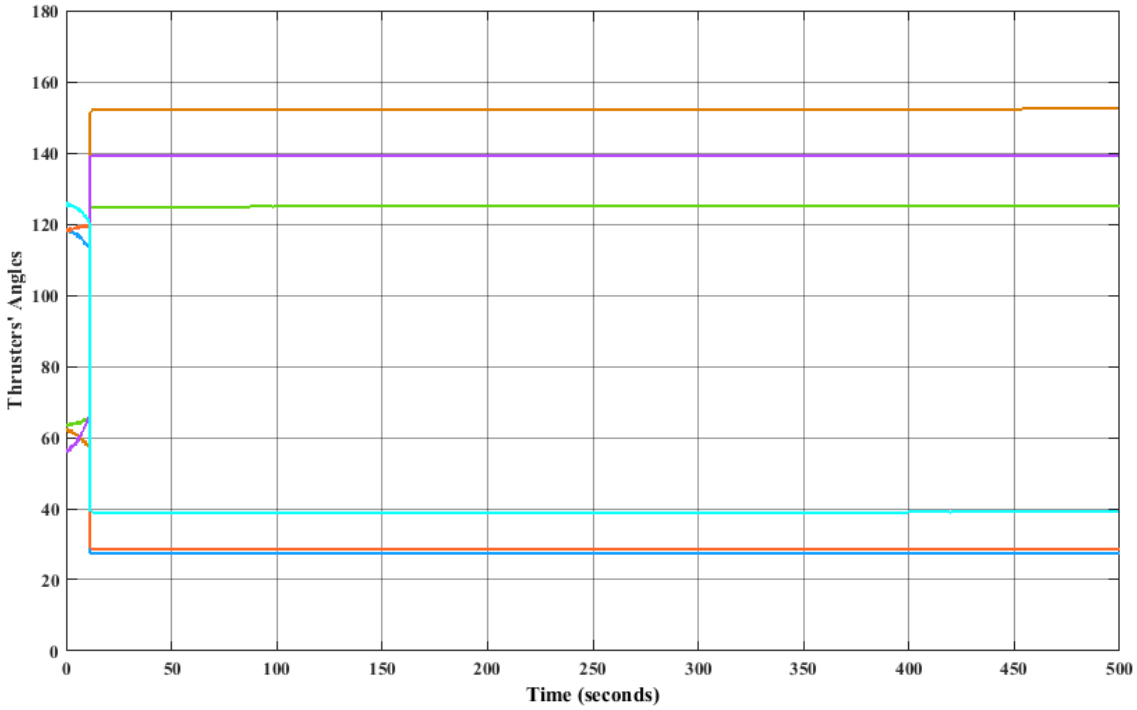


Fig. 5.14 Diagnosis Filter - 2^{nd} and 3^{rd} Thruster Fault (Total Failure)

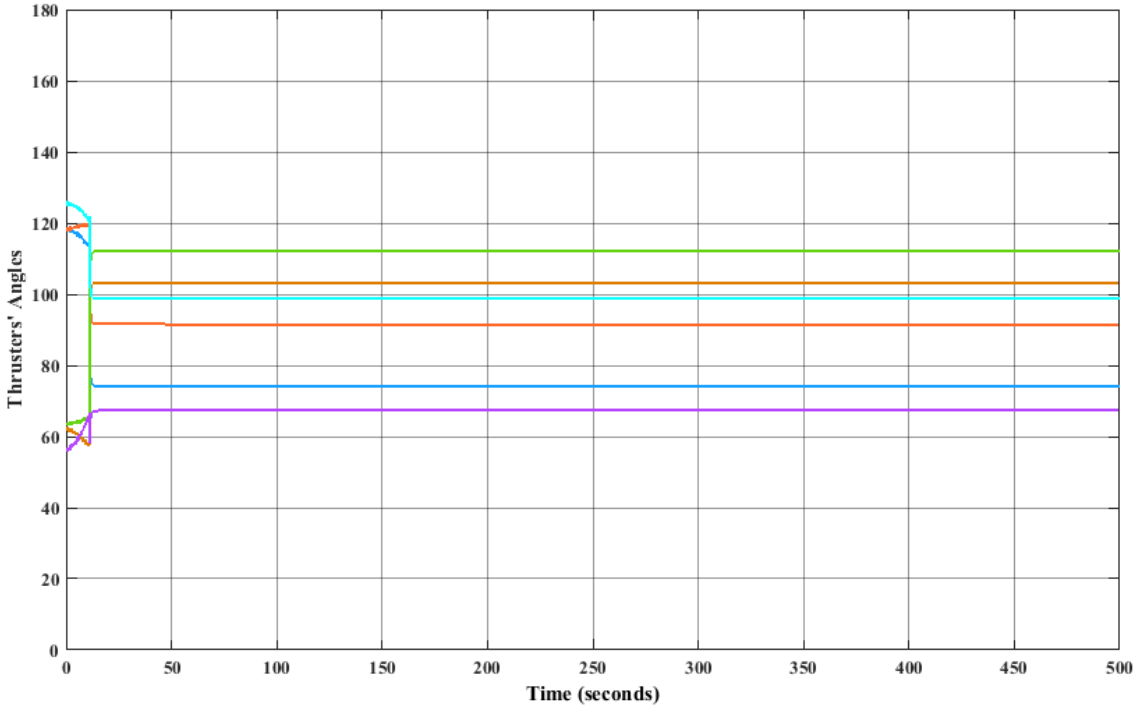


Fig. 5.15 Diagnosis Filter - 2^{nd} and 5^{th} Thruster Fault (Total Failure)

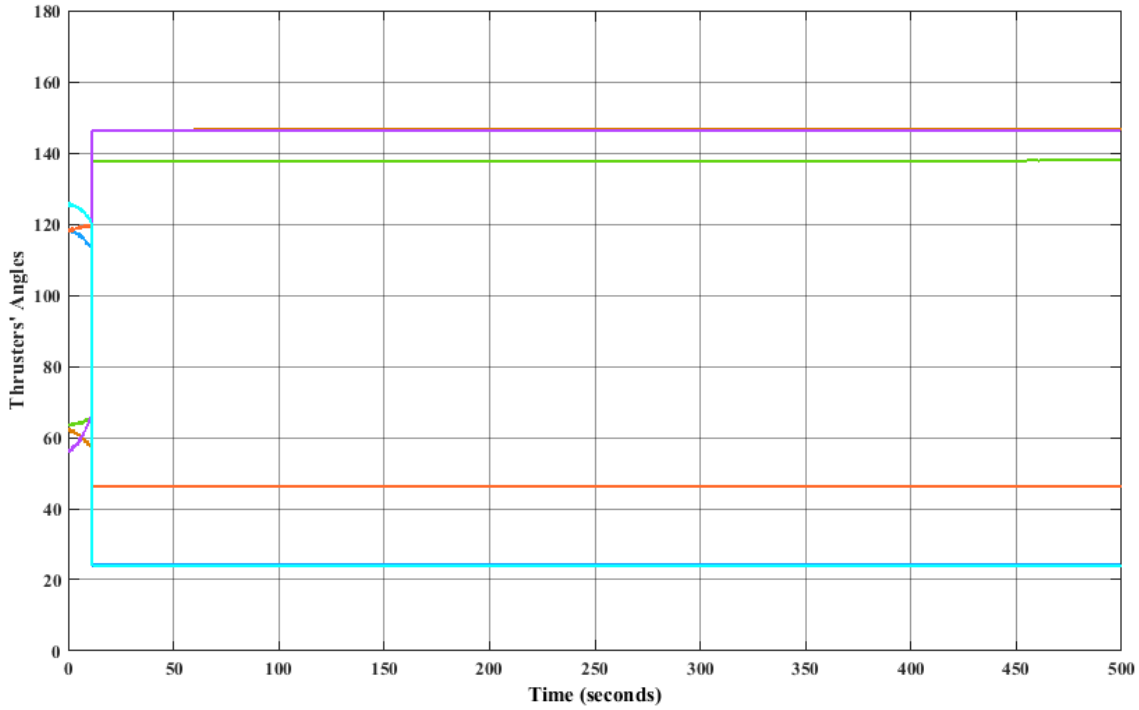


Fig. 5.16 Diagnosis Filter - 2^{nd} and 6^{th} Thruster Fault (Total Failure)

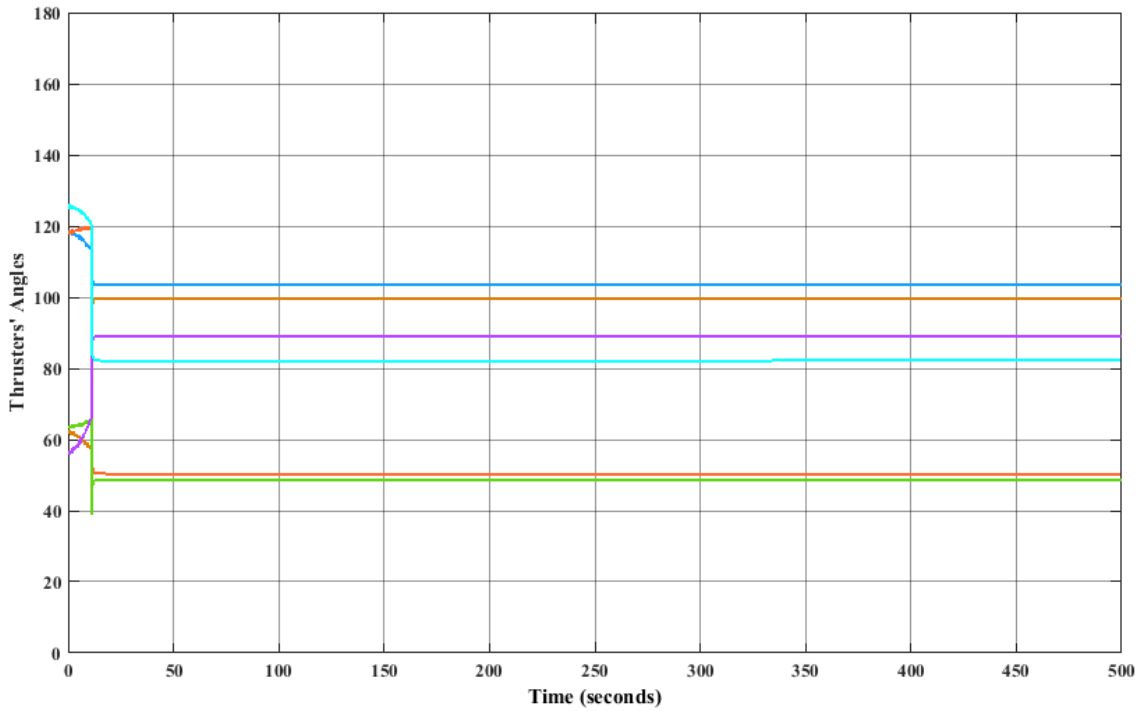


Fig. 5.17 Diagnosis Filter - 3^{rd} and 4^{th} Thruster Fault (Total Failure)

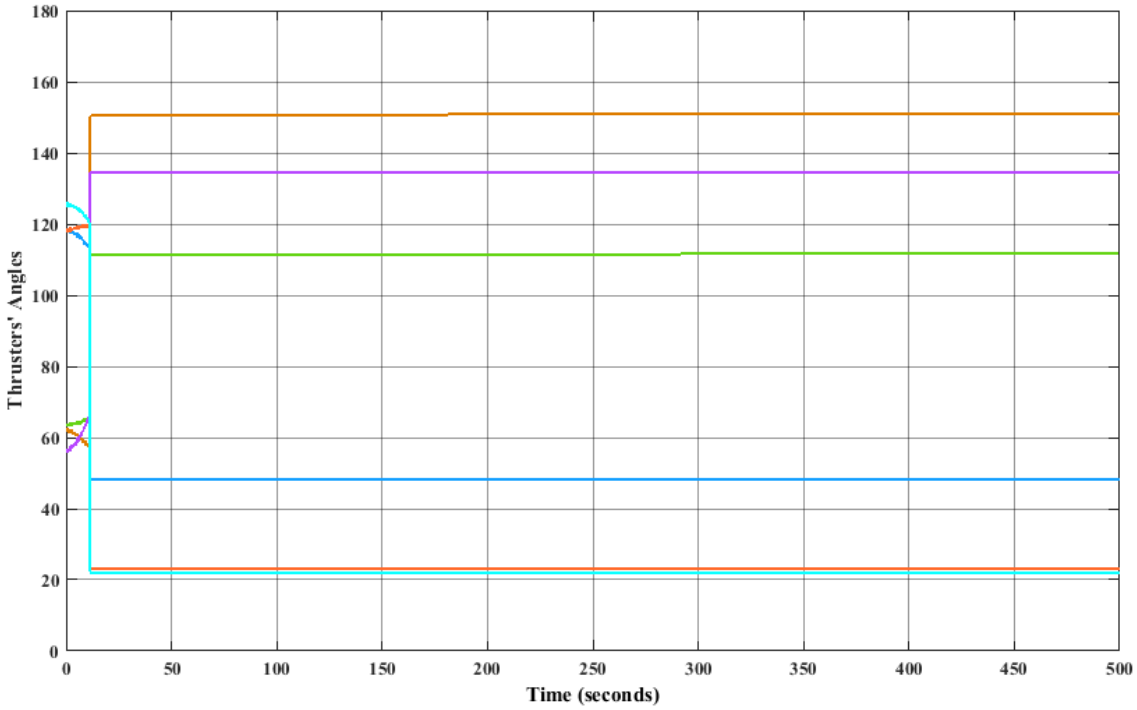


Fig. 5.18 Diagnosis Filter - 3^{rd} and 6^{th} Thruster Fault (Total Failure)

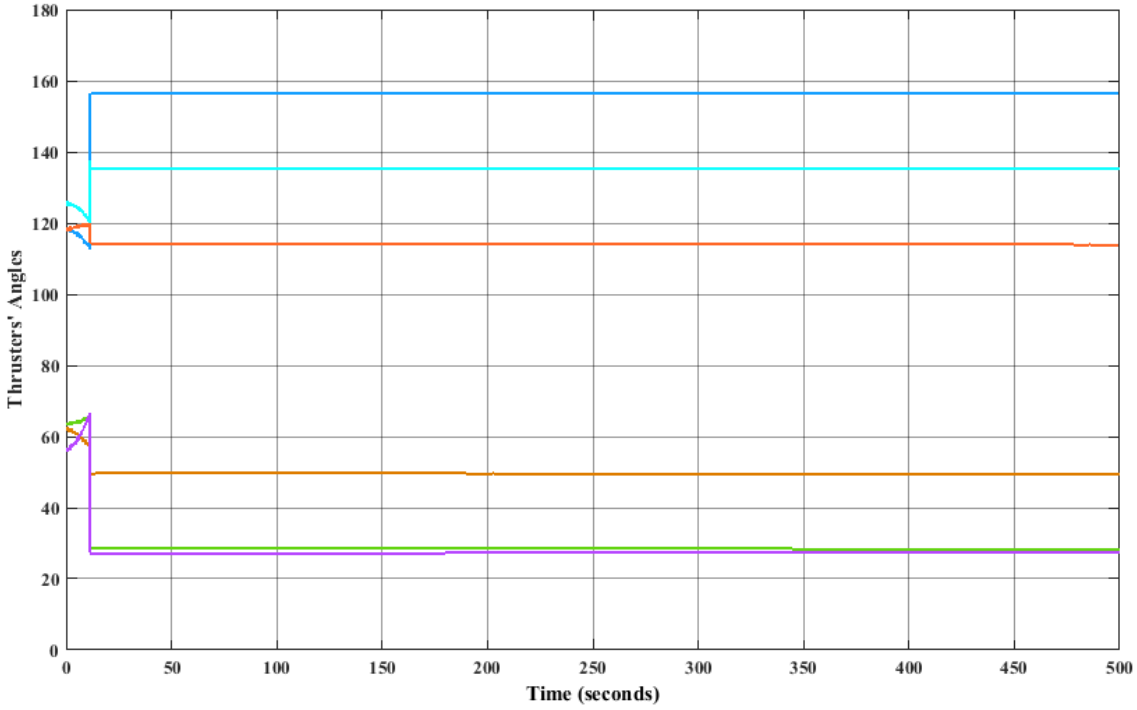


Fig. 5.19 Diagnosis Filter - 4^{th} and 5^{th} Thruster Fault (Total Failure)

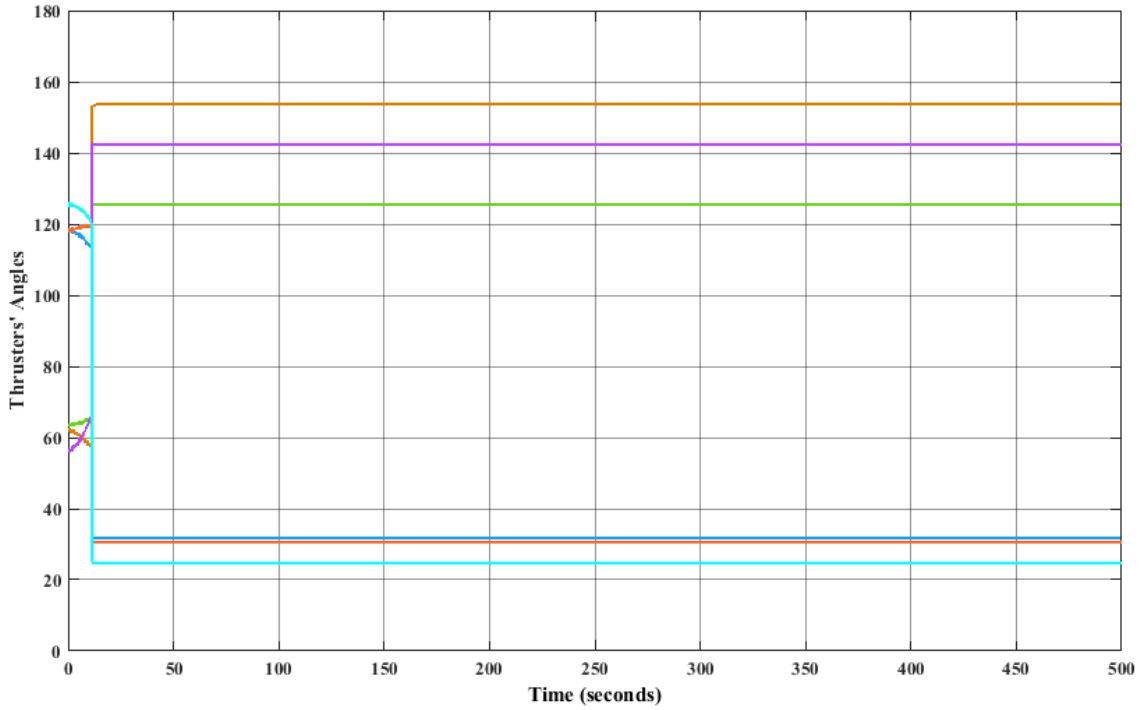


Fig. 5.20 Diagnosis Filter - 2^{nd}, 3^{rd} and 6^{th} Thruster Fault (Total Failure)

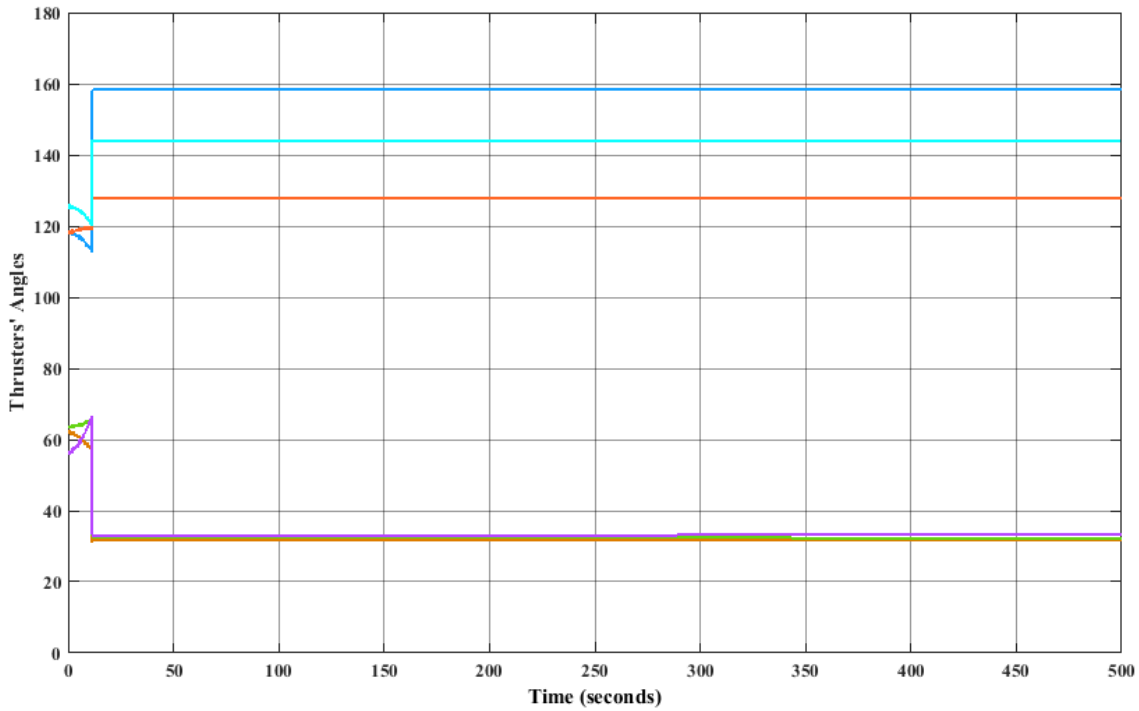


Fig. 5.21 Diagnosis Filter - 1^{st}, 4^{th} and 5^{th} Thruster Fault (Total Failure)

Establishing Algorithm Performance Limits in Off-Nominal Conditions

The complete Diagnosis Filter algorithm was run extensively to first establish maximum total noise levels and minimum severity magnitudes that can be correctly identified. The results are shown in Table 5.1.

Table 5.1 Diagnosis Filter - Severity Magnitudes and Noise levels of Validation Plan

Min Constant Failure	Max Noise Factor
1(total failure)	10000
0.1	1000
0.01	100
0.001	10
0.0001	1

Combinations of minimum leakage (slope) versus maximum noise were also extensively tested. In out leakage model the parameter α ranged from 0.0001 and up, with 0.0001 representing a leakage of 0.01 after 100 s. The results are shown in Table 5.2.

Table 5.2 Diagnosis Filter - Leakage Magnitudes and Noise Levels of Validation Plan

Min Leakage Rate	Max Noise Factor
0.001	1
0.01	10
0.1	100

In Tables 5.1 and 5.2, the max noise factor (NF) is simply a scaling factor from the nominal value of one (Astrium level) and upwards and includes noise from all sources, i.e. both from disturbances and sensor noise. It is also worth noting that given that FDI occurs under reasonable noise factor conditions (NF=10), in less than 50 s, the Diagnosis Filter is able to identify a very slow leakage, reaching a magnitude of less than 0.005 at the time of identification. Intermittent failures of very small magnitude were also tested with equal success. This is attributable to the fact that FDI is achieved within just a few seconds, a time interval which is normally much shorter than the timing cycle of intermittent failures. Indeed, as the

extensive simulations show, the Diagnosis Filter exhibits excellent performance and robustness to dramatically enhanced noise and minimal severity and leakage rate magnitudes.

Sensitivity and Robustness Results – Monte Carlo Simulations

As also reported in [50], the most important contributors to enhanced sensitivity and reduced robustness are:

- the thrusters' misalignment that is directly translated into the influence matrix of the thrusters on the spacecraft acceleration.
- the center of mass of the spacecraft.
- the suspension and drag-free stiffness.

The Monte Carlo Simulation Framework

The Monte Carlo Simulation framework was essentially the entire simulation environment realization of the FDI algorithms, complete with the severity identification module, where the parameters were chosen to change randomly and simultaneously, and where the noise factors were varied. For the errors listed above, four sets of 10000 Monte Carlo simulations were executed:

i. random up to $\pm 10\%$ misalignment, which corresponds to up to 3.6 degrees, and $\pm 10\%$ variation in centre of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 10.

ii. random up to $\pm 10\%$ misalignment, which corresponds to up to 3.6 degrees, and $\pm 10\%$ variation in centre of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 100.

iii. random up to $\pm 20\%$ misalignment, which corresponds to up to 7.2 degrees, and $\pm 20\%$ variation in centre of mass (nominal value $d^* = 0.376$) and suspension

and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 100.

iv. random up to $\pm 20\%$ misalignment, which corresponds to up to 7.2 degrees, and $\pm 20\%$ variation in center of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 1000.

Severity also varied randomly from 0.01 to 1, which corresponds to failures from 1% - 100%. All the eight thrusters were included.

Monte Carlo Results

i. In this case, which nonetheless represents a significant departure from nominal conditions, the results were perfect; absolutely 100% true detection and identification as well as accurate severity identification.

ii. In this case, 99.85% fault detection was achieved, with a 0.15% missed fault rate, 99.85% true severity identification and zero false alarms.

iii. In this case, there was 99.89% true detection and true identification, with 0.11% missed fault rate, 99.89% true severity identification and zero false alarms.

iv. This case, with 1000 noise factor and 20% parameter variation , there was 95.64% true detection rate, 4.31% missed fault rate and zero false alarms. The true isolation rate was at 95.35% and true severity magnitude identification at 95.35%, to within 0.1 difference in magnitude of severity.

Indicative corresponding graphs for failed thruster frequency and severity profiles for the 4^{th} case are given below.

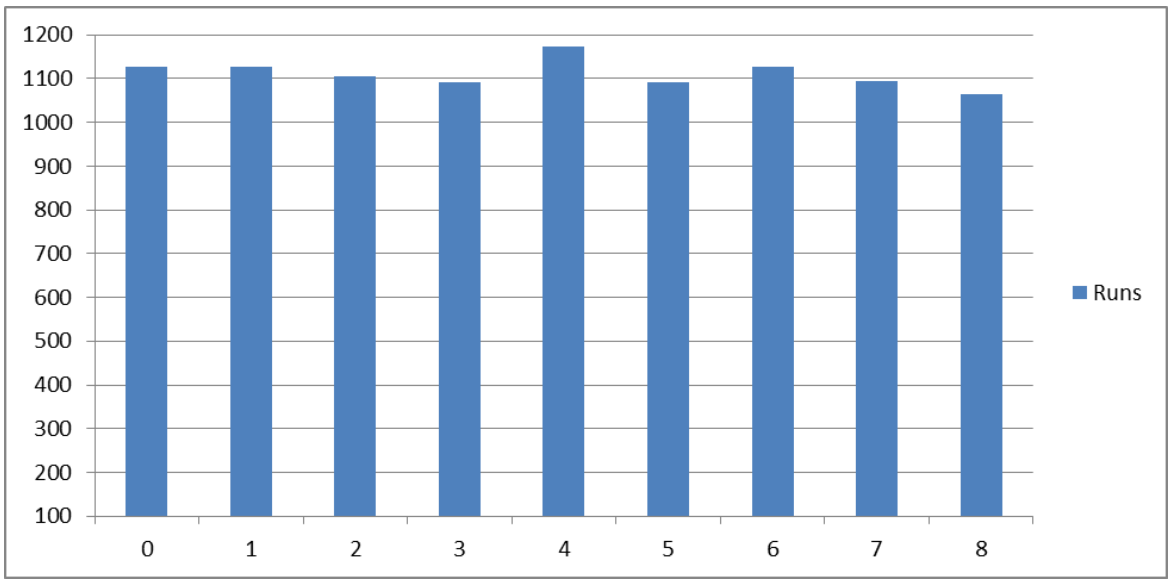


Fig. 5.22 Diagnosis Filter - Distribution of Failed Thrusters for 4th Monte Carlo Set

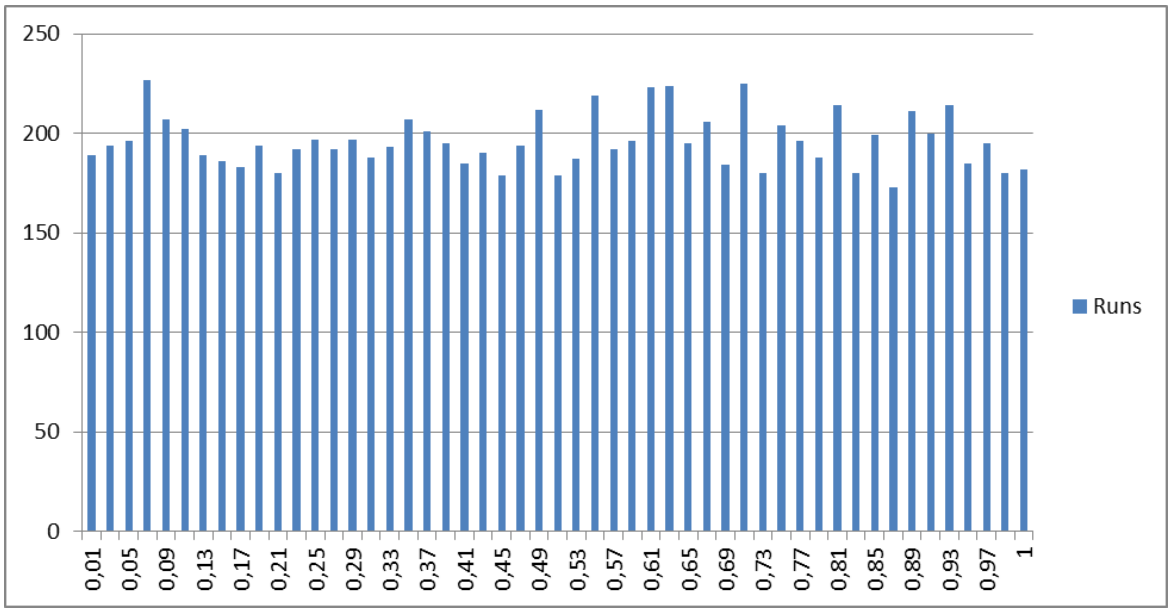


Fig. 5.23 Diagnosis Filter - Distribution of Severity Magnitudes for 4th Monte Carlo Set

Time to Detection and Identification

Detection and identification occurred almost simultaneously, with severity identification following immediately after. Therefore, there is no real insight or value to consider them separately. The histogram (Figure 5.24) is typical of the timing profiles encountered in all Monte Carlo sets; in fact, it may be slanted toward higher values, as this was the set for 20% variation and NF=1000. The significant majority of cases lie in the region below 47 s, with 122 at 0.5 s. The cases that took longer time are invariably those with very low severity and highest levels of parametric variation, as expected. In fact, with such high noise values the min angle subtended between residual and actual failed direction is in the vicinity of 27 degrees; this represents the upper bound on the min angle encountered in the extensive FDI runs. Under reasonable circumstances, represented by a 10% parameter variation and NF=10, the min angles range around 4 to 5 degrees, while under nominal circumstances, NF=1 and nominal parameter values, they are well below 1 degree.

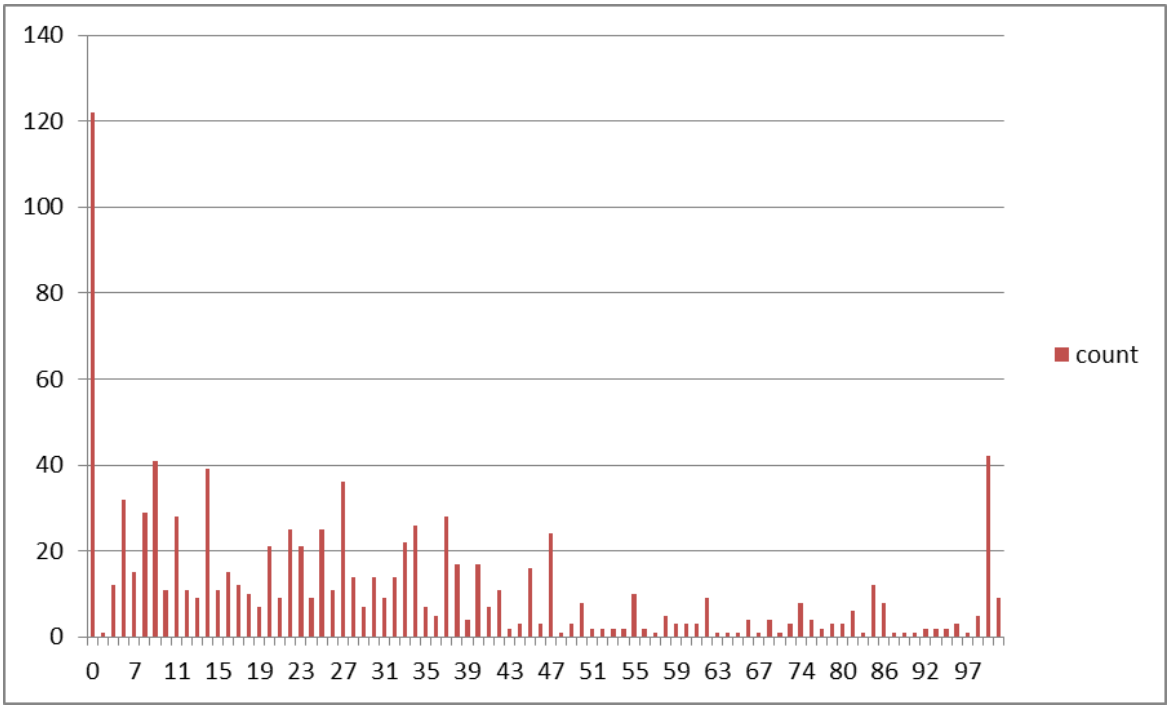


Fig. 5.24 FDI Detection Timing Profiles (Rounded Off)

The timing to FDI is very similar for all four runs; this was indeed the finding throughout the entire Monte Carlo simulation testing. Table 5.3 summarizes the results obtained for all four sets of the metrics considered.

Table 5.3 Diagnosis Filter – Summarized Results of Four Sets of Monte Carlo Analysis

Evaluation Criteria	1^{st} SET	2^{nd} SET	3^{rd} SET	4^{th} SET
False Alarm Rate r_{fa}	0%	0%	0%	0%
Missed Detection Rate r_{md}	0%	0.15%	0.11%	4.36%
True Detection Rate r_{td}	100%	99.85%	99.89%	95.64%
True Isolation Rate r_{ti}	100%	99.85%	99.89%	95.35%
Wrong Isolation Rate r_{iw}	0%	0.15%	0.11%	4.65%
Average Detection Time t_{da}	< 10 s	< 10 s	< 10 s	< 10 s
Average Isolation Time t_{ia}	< 10 s	< 10 s	< 10 s	< 10 s

Plant Instability and FDI

As observed during the tests, after FDI had occurred, with minimum angle subtended by the residual in a specific failure direction corresponding to a failed thruster identified, the angle started increasing after a period of time during which it lingered around its minimum. Beyond a certain time point, one could no longer trace the thruster that had been identified as failed. This is not surprising, as the underlying system contains instabilities, which after becoming pronounced, obstruct the FDI process. Although, at first glance, this might appear as a problem, or a 'tug-of-war' between FDI and plant instability, it is of no concern since FDI occurs extremely fast even for extreme cases, well before the onset of instability.

The min angle evolution for two representative and challenging cases is given below:

i. 10% misalignment, -20% d variation and NF= 100.

ii. 20% misalignment, -20% d variation and NF= 100.

There are three figures for each case: Figure 5.25 and Figure 5.28 entail the entire time interval run, while Figure 5.26 and Figure 5.29 focus around the min angle region. Figure 5.27 and Figure 5.30 depict an even smaller time scale, to show the signal characteristics. The effect of noise is evident in both cases. The horizontal axis represents the simulation time intervals, which have their correspondence in real seconds; these are recorded in the actual data files.

i. Up to 10% misalignment, -20% d variation, NF=100.

Table 5.4 Diagnosis Filter - min Angle Evolution for 1^{st} case

below 10 degrees in	within 1 degree of minimum angle in	min angle attained in
4.3 s	7.9 s	34.4 s

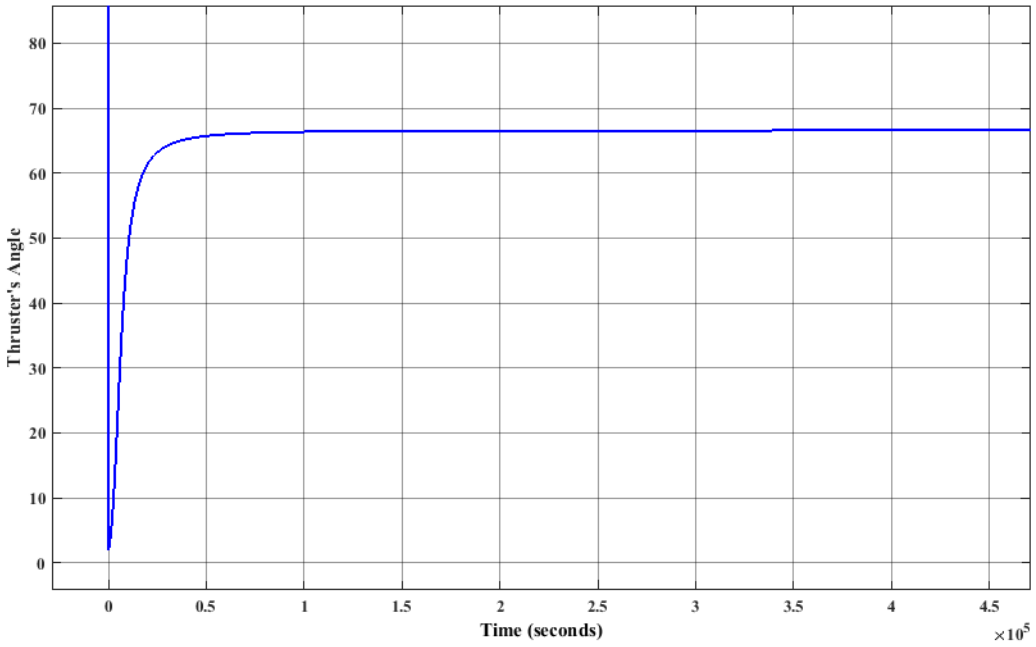


Fig. 5.25 Diagnosis Filter - Timing Profile showing Instability Inception Point Beyond which FDI is Jeopardised for 1^{st} case

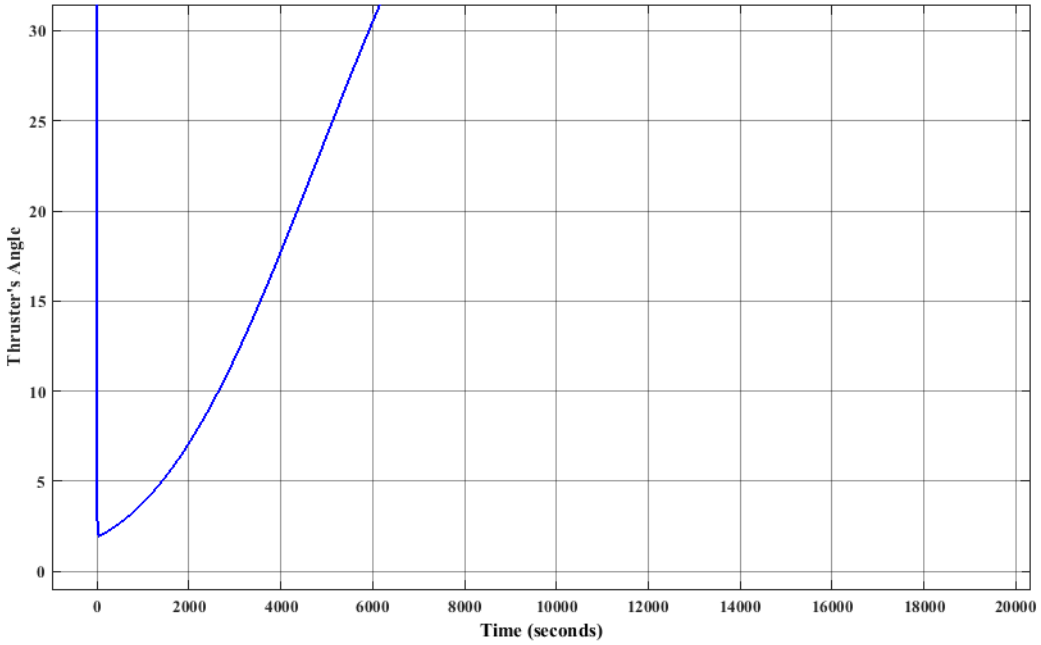


Fig. 5.26 Diagnosis Filter - Zooming in to Detection & Identification Time for 1^{st} case

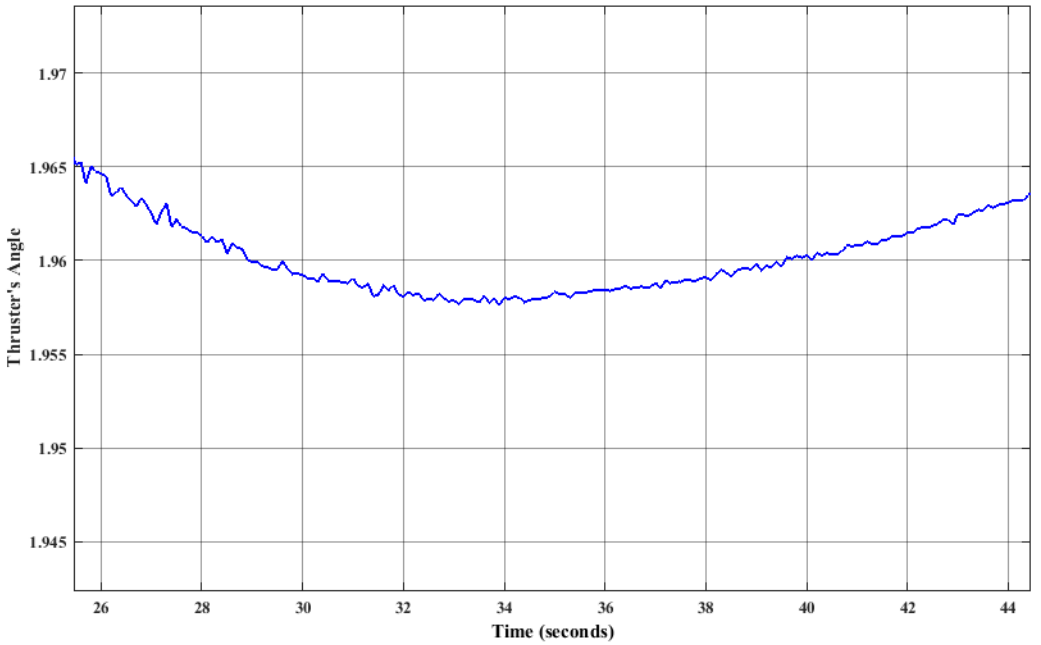


Fig. 5.27 Diagnosis Filter - Further Zooming in to Detection & Identification Time Showing the Effects of Noise for 1^{st} case

ii. Up to 20% Misalignment, -20% d variation, NF=100.

Table 5.5 Diagnosis Filter - min Angle Evolution for 2^{nd} case

below 30 degrees in	within 1 degree of minimum angle in	min angle attained in
6.4 s	10.3 s	29.8 s

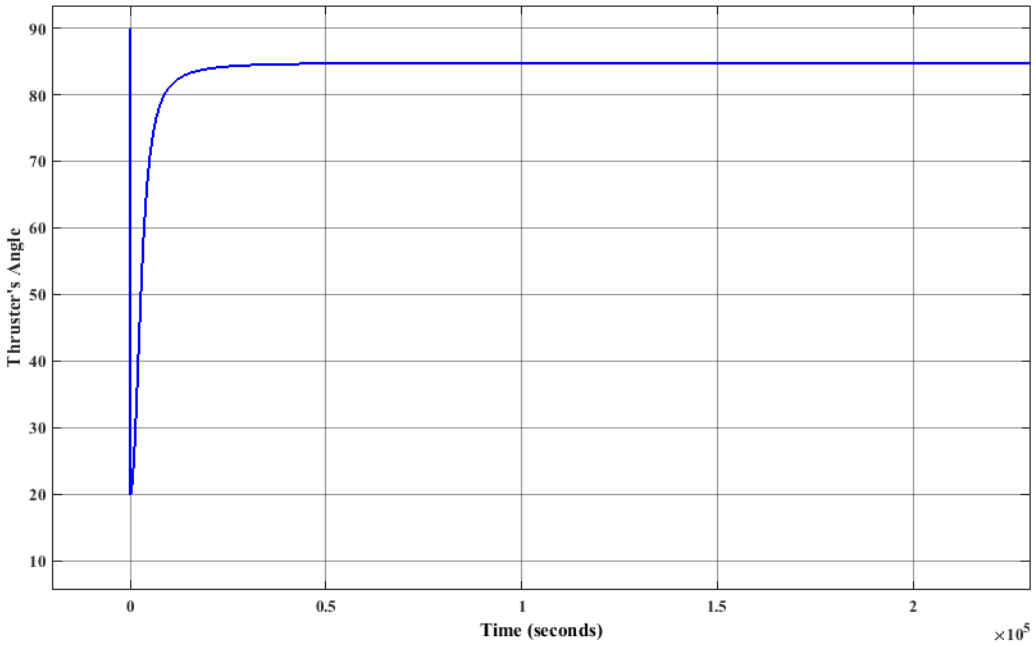


Fig. 5.28 Diagnosis Filter - Timing Profile showing Instability Inception Point Beyond which FDI is Jeopardised for 2^{nd} case

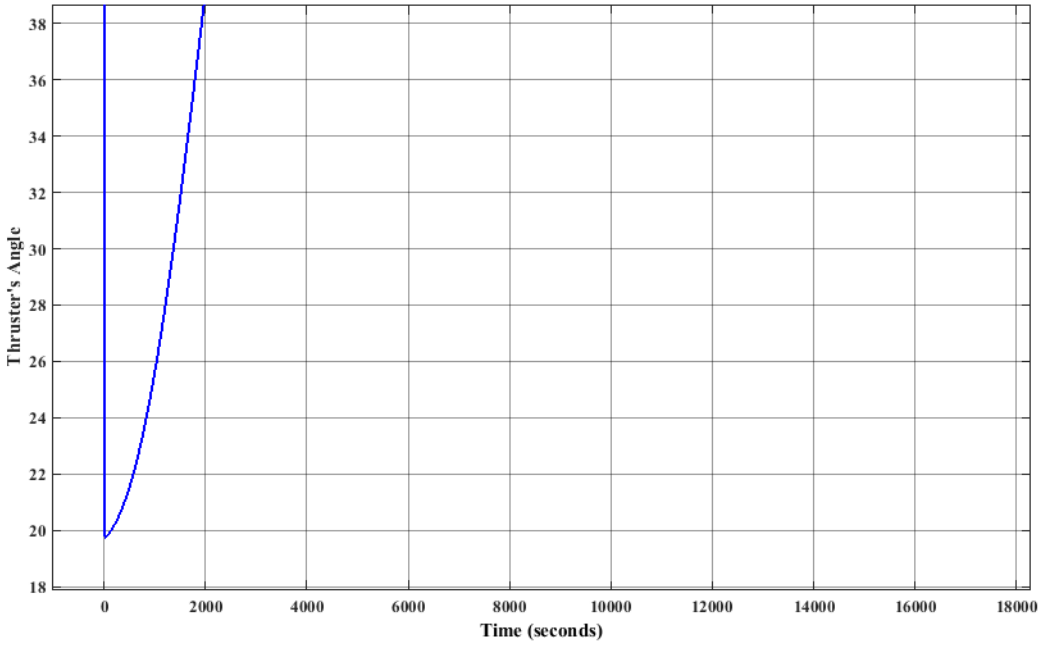


Fig. 5.29 Diagnosis Filter - Zooming in to Detection & Identification Time for 2^{nd} case

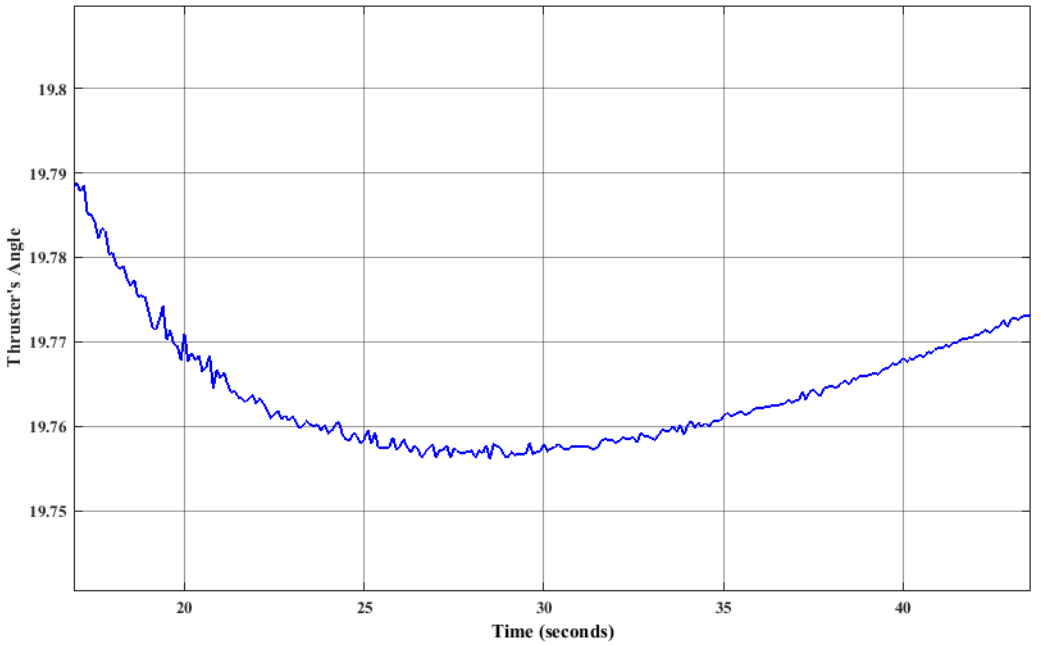


Fig. 5.30 Diagnosis Filter - Further Zooming in to Detection & Identification Time Showing the Effects of Noise for 2^{nd} case

5.3 Design Method 2: The Euresis Filter

The following figures were obtained on a simulation environment with full disturbance and measurement noise presence as pertains to the LPF environment. In the particular scenario of interest for the LPF application, where only actuator (thruster) failures are considered, the Euresis Filter performs very well, though not as fast as the Detection Filter, and is very easy to implement. The fact that the uncertain parameters (failures) entered only in matrix B, results in a significantly better (more robust) and faster convergence, relative to this scheme's inherent convergence capabilities, than if the uncertainty was in the system matrix A. The Euresis Filter uses a higher number of models - nine vs two in the case of the other two filters for this specific application. However, its convergence properties and accuracy are very good.

Figure 5.31 shows the eight KFs residuals under no failure; they are all in the order of 10^{-6}. Subsequently, one thruster was totally failed at a time, after 10 s of trim. FDI was considerably slower than in the case of the other two filters, occurring in about 70 s. The residual magnitude corresponding to the failed thruster's KF is consistently an order of magnitude smaller in comparison with the other thrusters, in the order of 10^3 as compared to 10^4 for the other thrusters, in all cases. When a failure is injected for a specific thruster, the simulation environment mechanization is such that the rest of the thruster magnitude signals reach the pre-specified upper bounds, as a way of compensating for the failed thruster in the plant closed loop; this does not occur, and is not a requirement, in the MMKFs employed in the FDI process. This explains the discrepancy in residual magnitudes between the unfailed and failure cases.

In the Euresis Filter, all the eight thrusters' failures could be identified with one bank of filters, consisting of nine models in total (one model represents a thruster without failure). Similarly, the same bank can identify two and three simultaneous thruster failures in any combination. In this respect, this approach seems promising for some degree of robustness.

However, since the MMKF FDI decision-making process is predicated on magnitude rather than direction, the overall robustness may not be so certain, when it comes to sensitivity to parameters, including those of the stochastic disturbances. On the other hand, the fact that directionality is imposed by the construction of each model, could serve as a mitigating factor. The sensitivity of the method is, of course, within the scope of the validation process.

Note: Because there is an order of magnitude difference between the level of noises and residuals, in the plots drawn on the residuals magnitude scale the noises are not discernible. Therefore, the plots (Figure 5.34) for the 3rd thruster failure were created in two scales, one at the noise level and the other at the residual level, to better show the noise effects on the actual residuals.

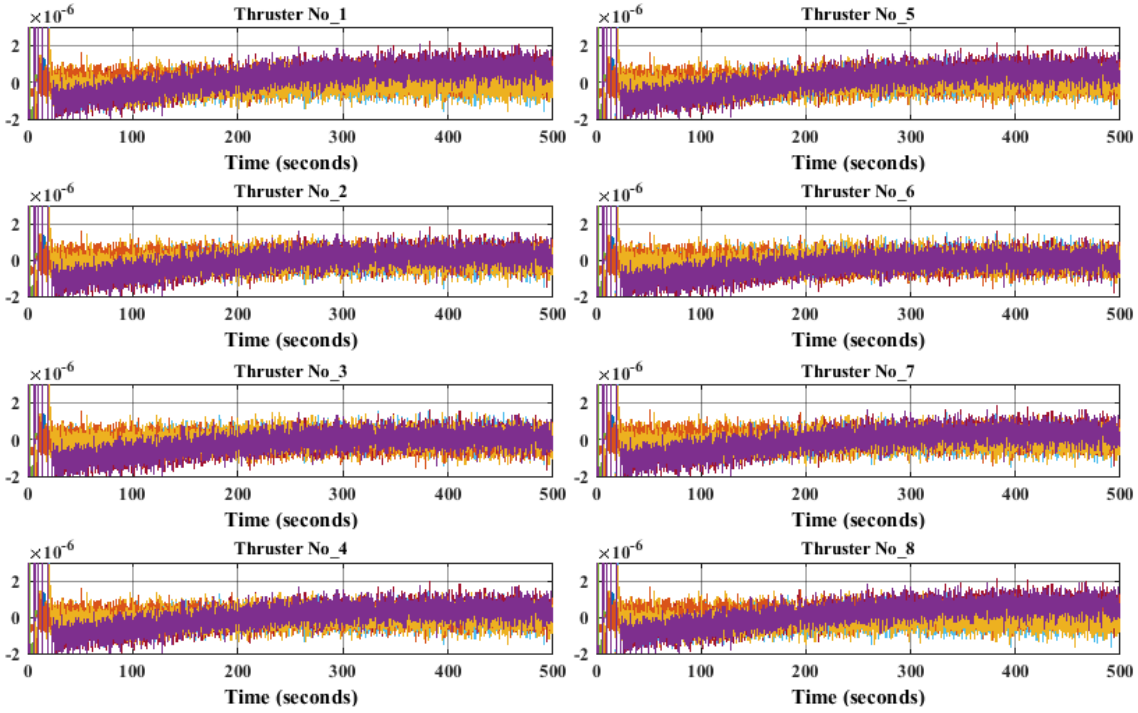


Fig. 5.31 Euresis Filter - Thrusters Without Failure

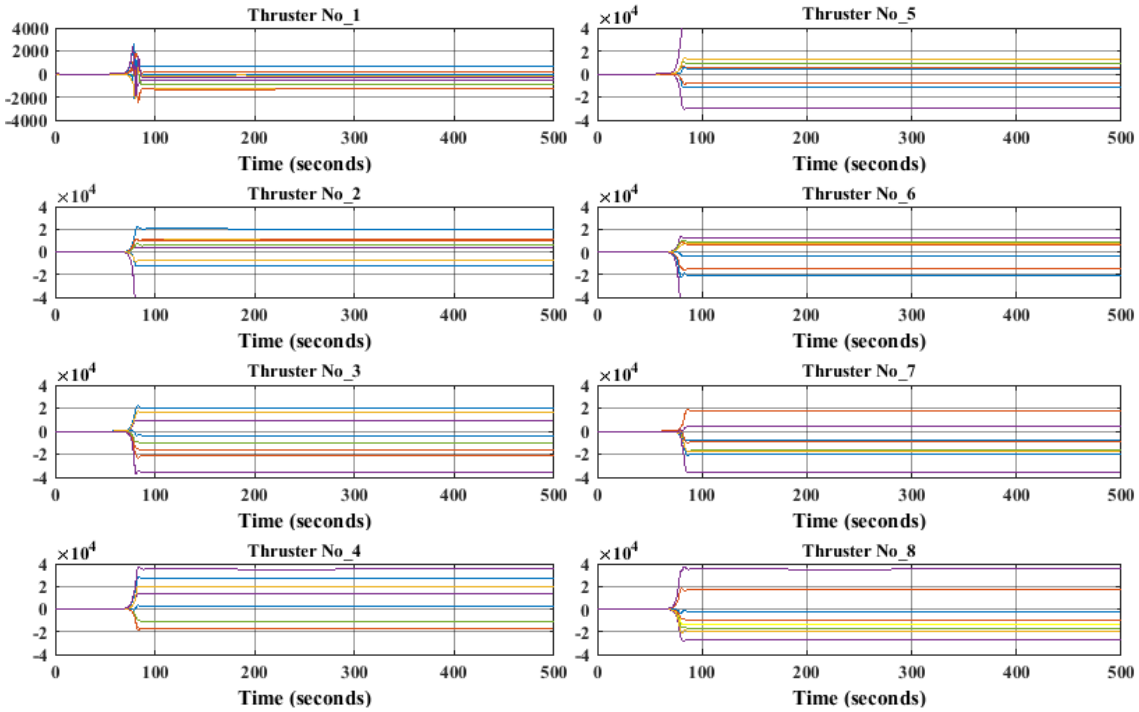


Fig. 5.32 Euresis Filter – 1^{st} Thruster (Total Failure)

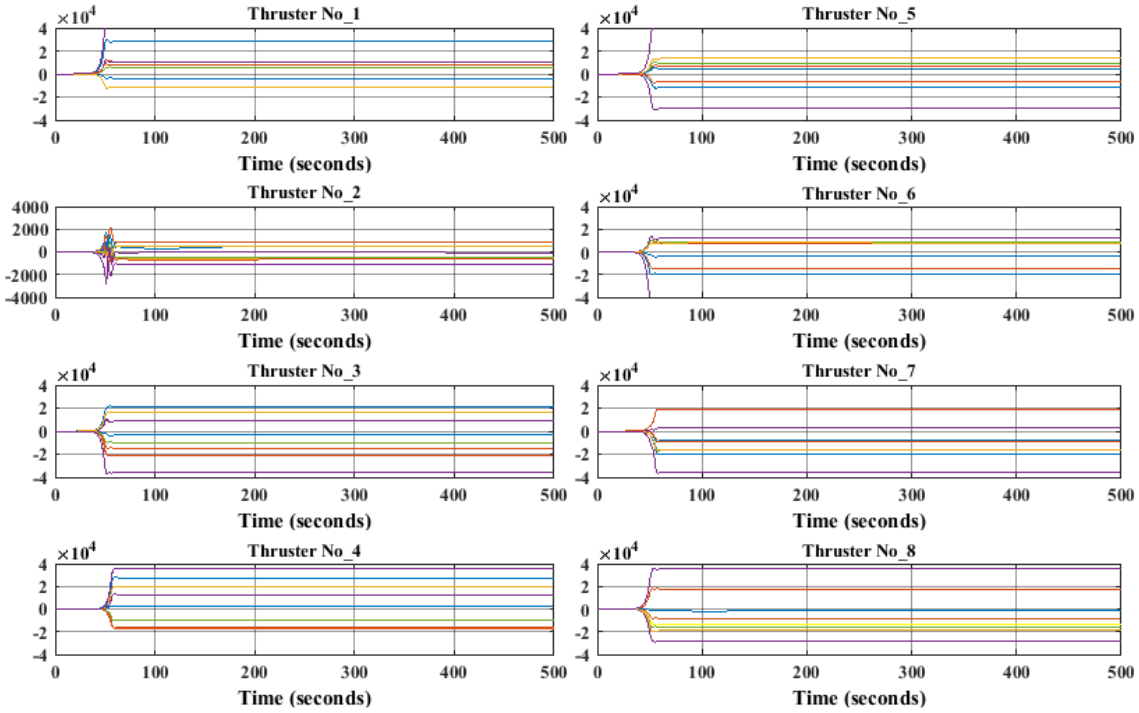


Fig. 5.33 Euresis Filter – 2^{nd} Thruster (Total Failure)

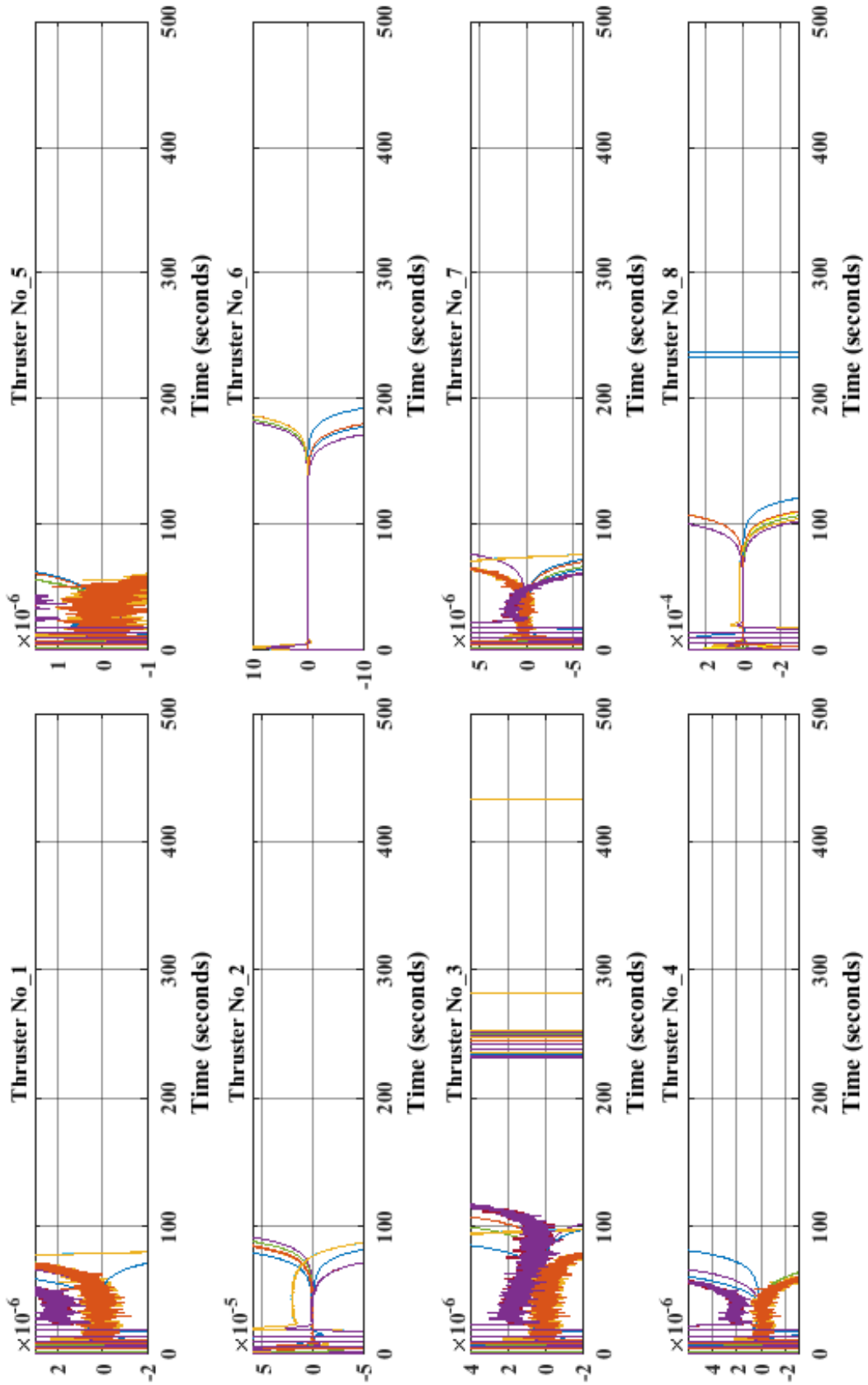


Fig. 5.34 Euresis Filter – 3^{rd} Thruster (Total Failure) - Different Scale to Show Presence of Disturbances in Residual

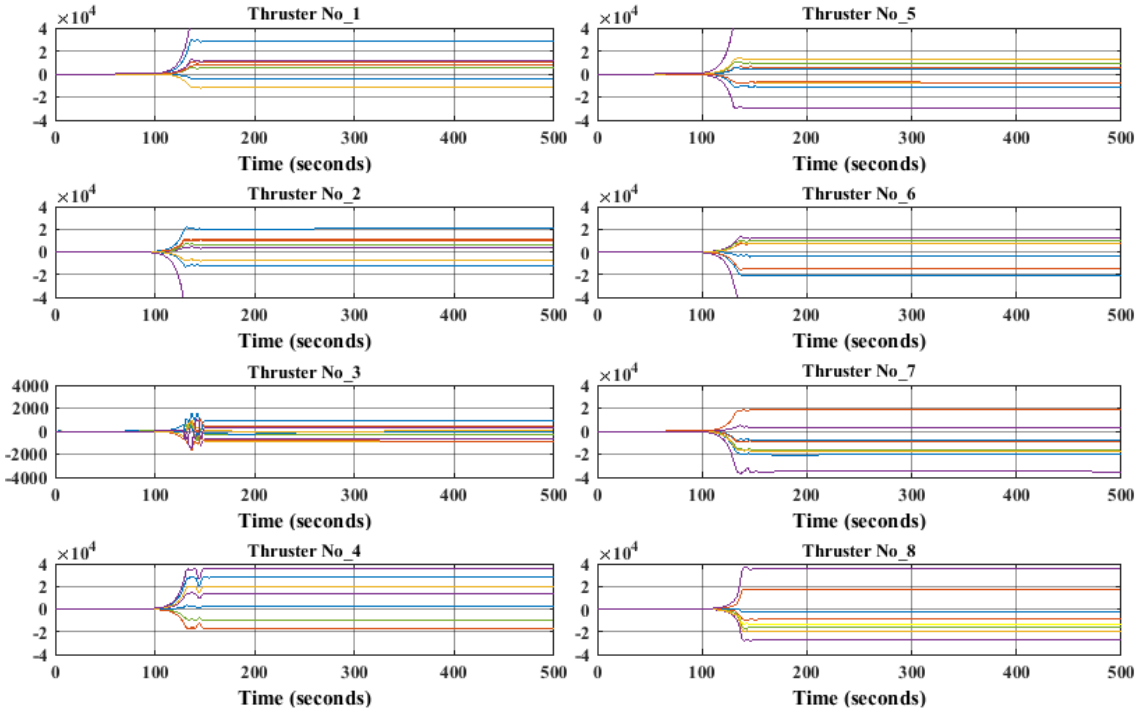


Fig. 5.35 Euresis Filter – 3^{rd} Thruster (Total Failure)

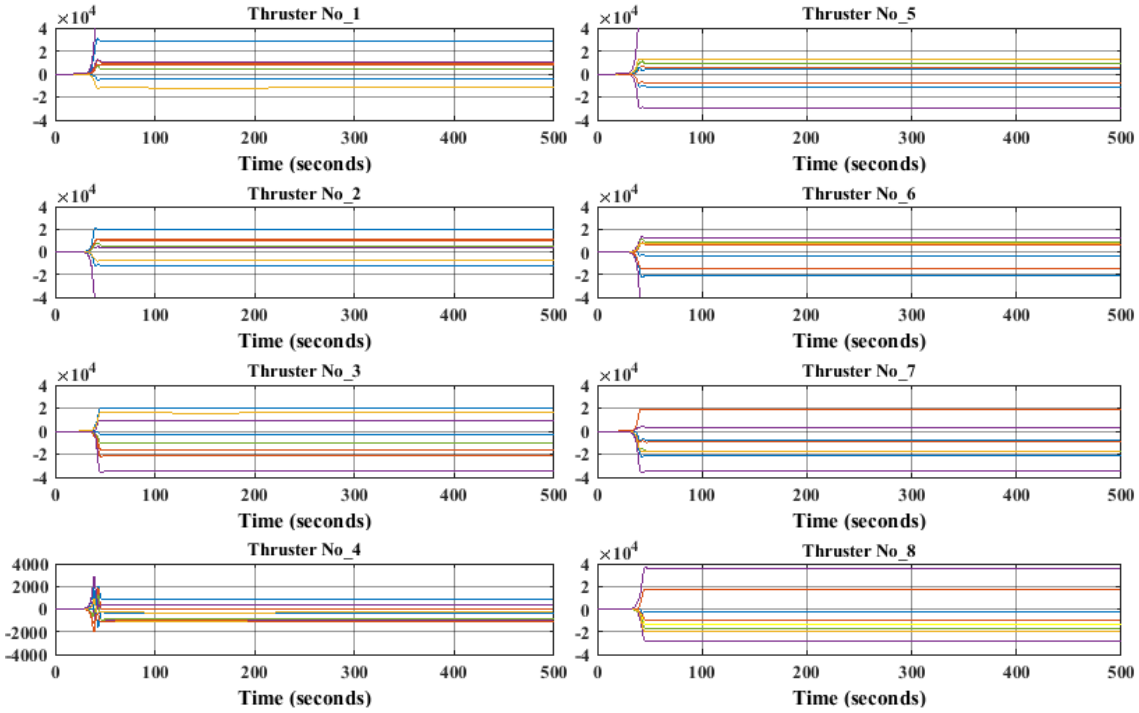


Fig. 5.36 Euresis Filter – 4^{th} Thruster (Total Failure)

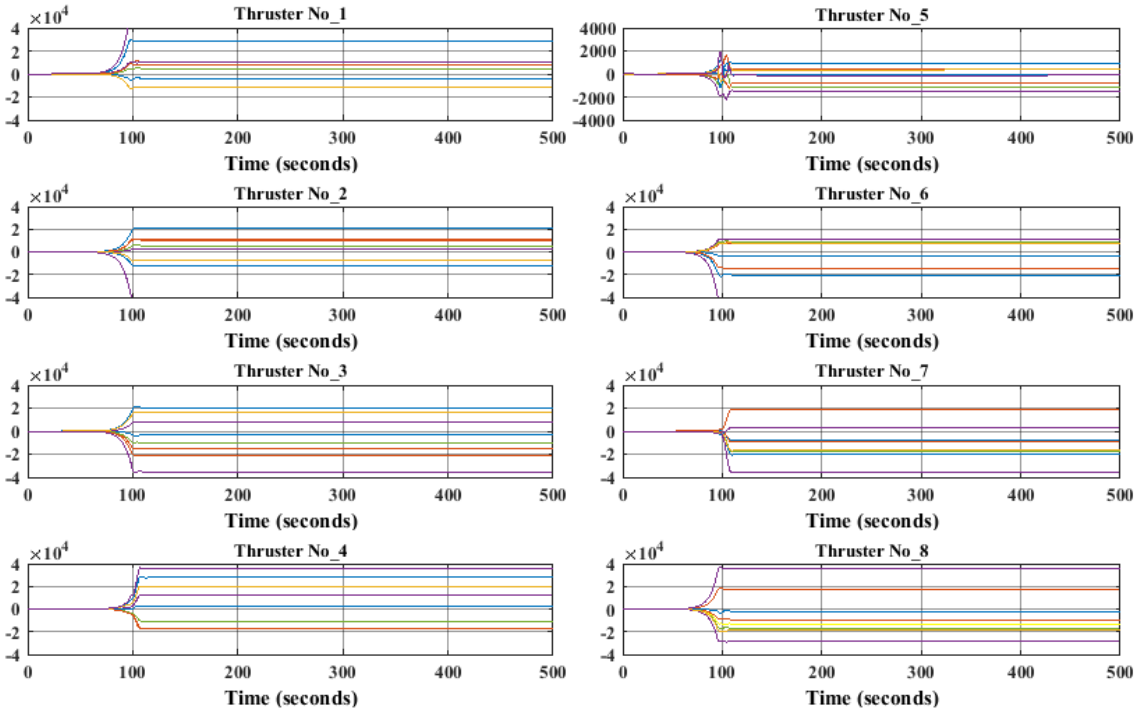


Fig. 5.37 Euresis Filter – 5^{th} Thruster (Total Failure)

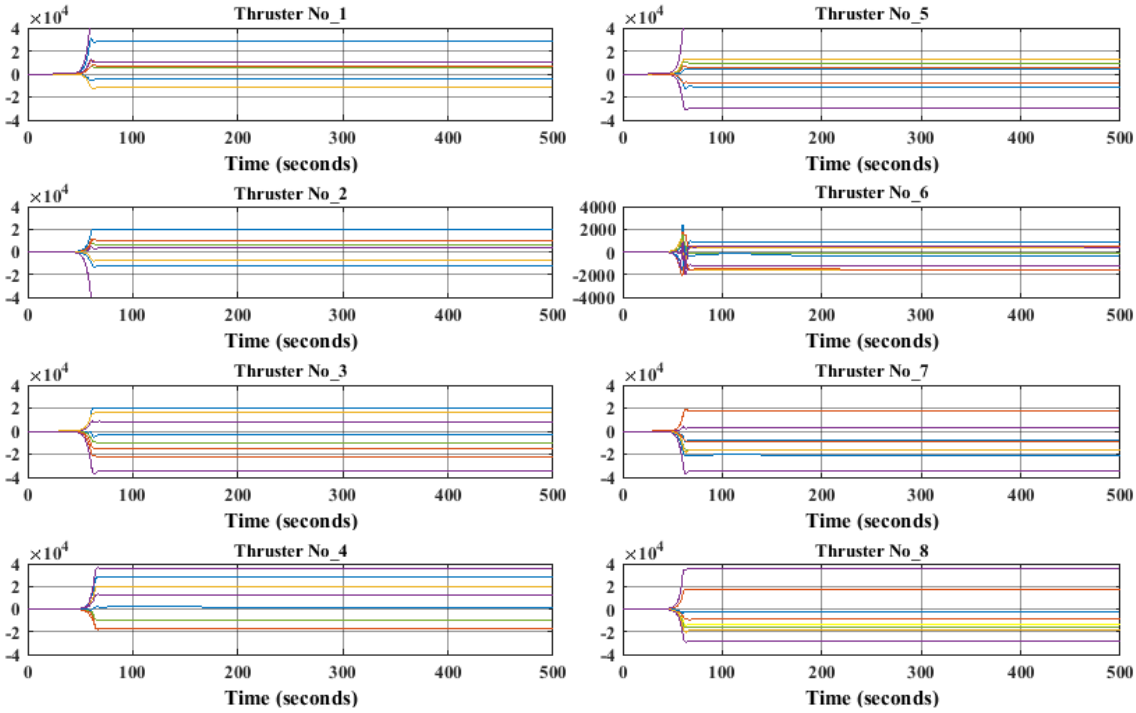


Fig. 5.38 Euresis Filter – 6^{th} Thruster (Total Failure)

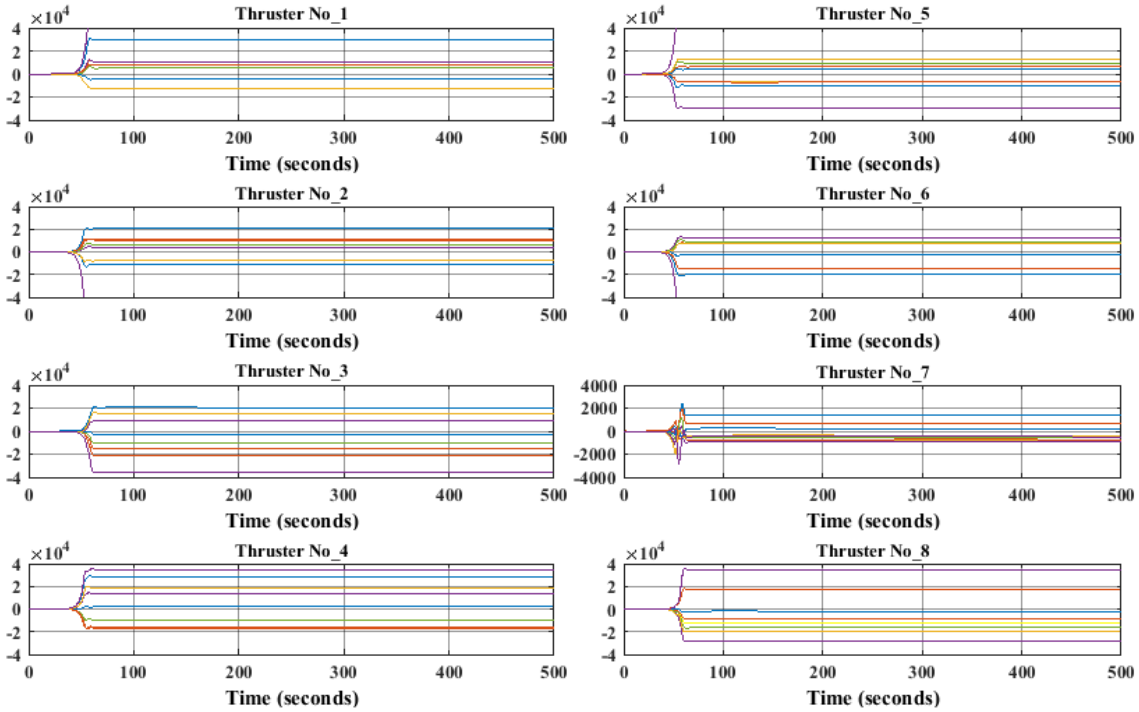


Fig. 5.39 Euresis Filter – 7^{th} Thruster (Total Failure)

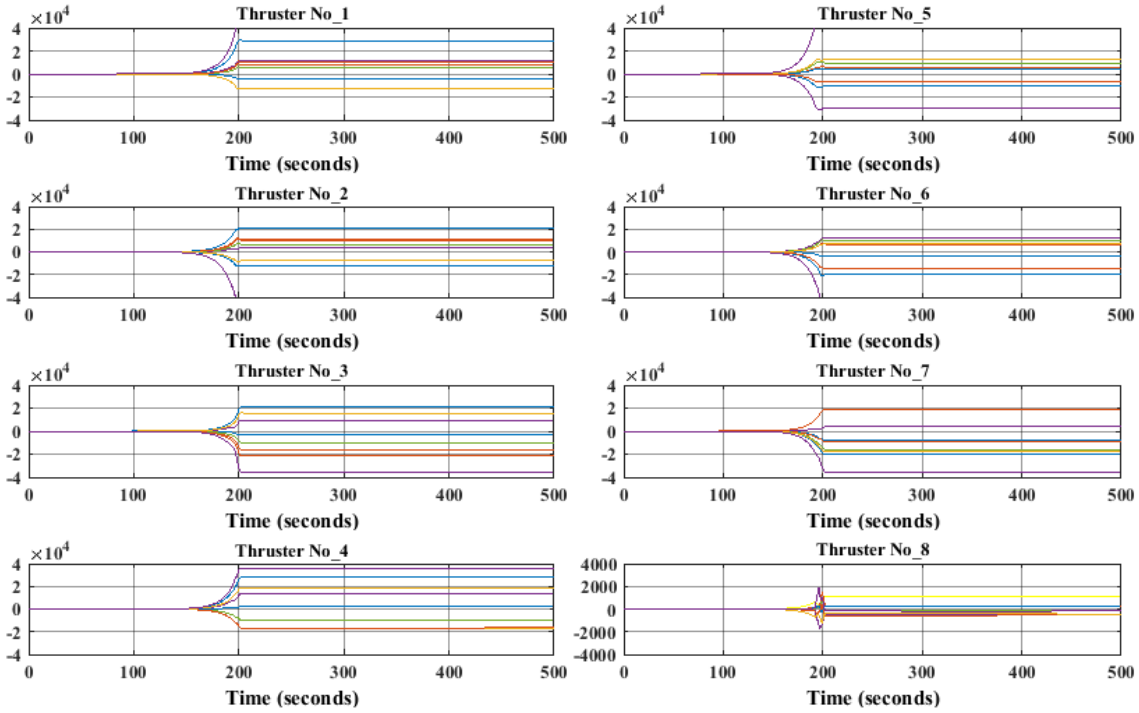


Fig. 5.40 Euresis Filter – 8^{th} Thruster (Total Failure)

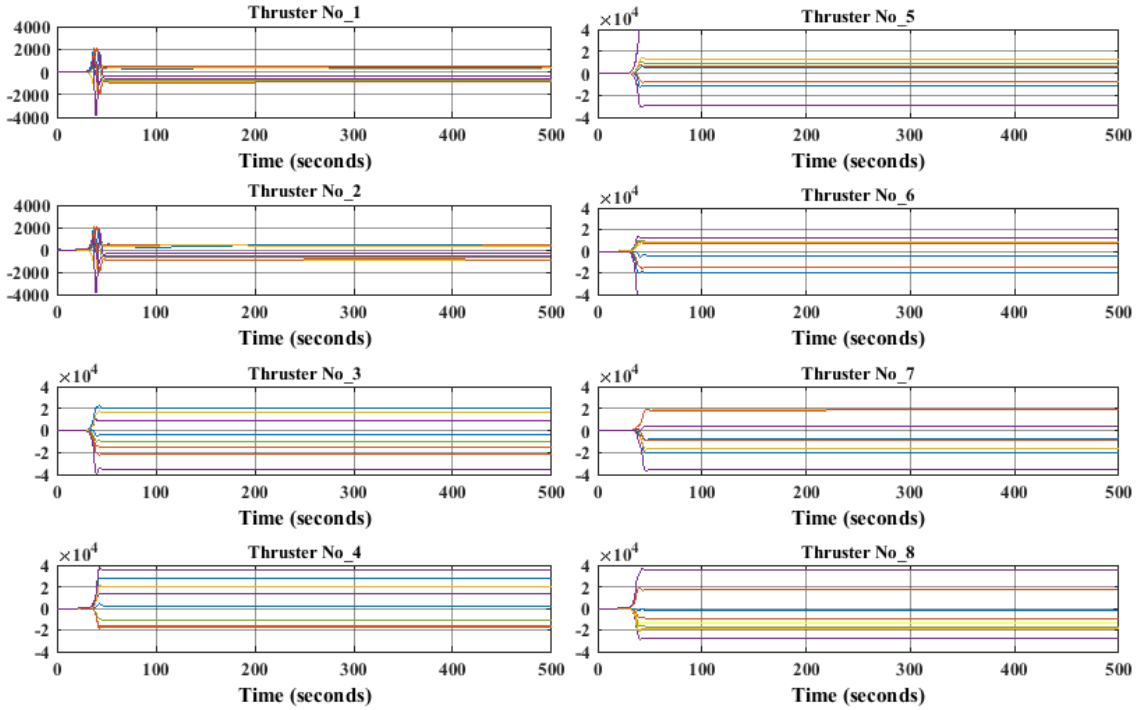


Fig. 5.41 Euresis Filter – 1^{st} and 2^{nd} Thruster (Total Failure)

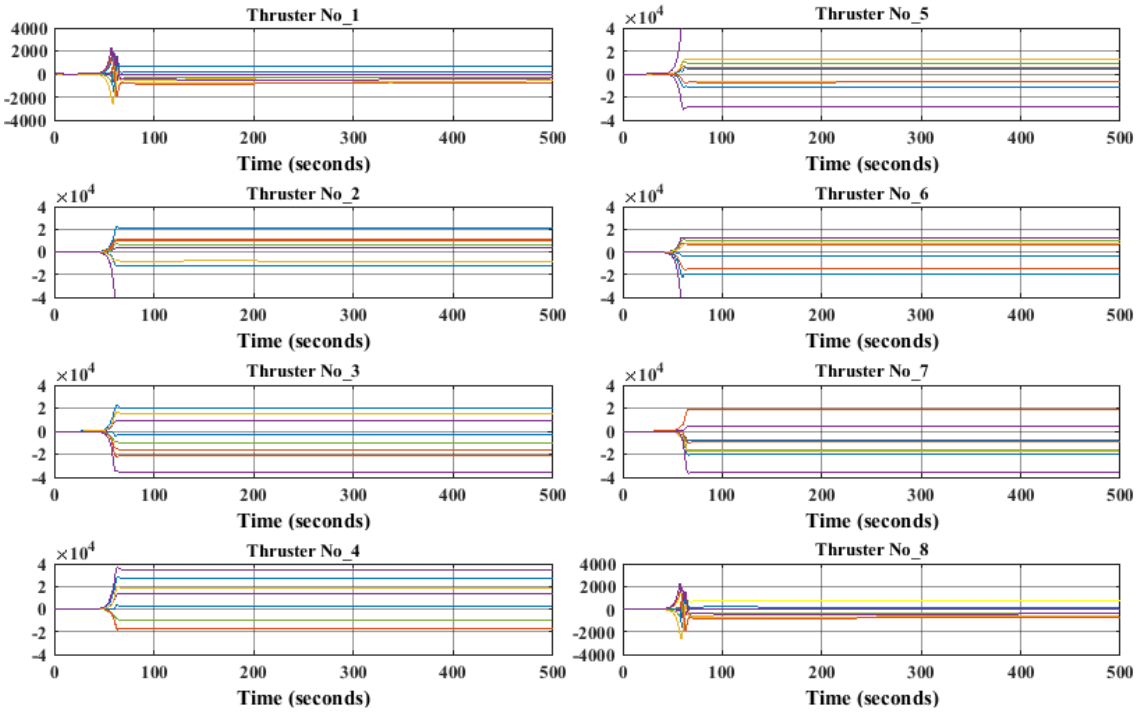


Fig. 5.42 Euresis Filter – 1^{st} and 8^{th} Thruster (Total Failure)

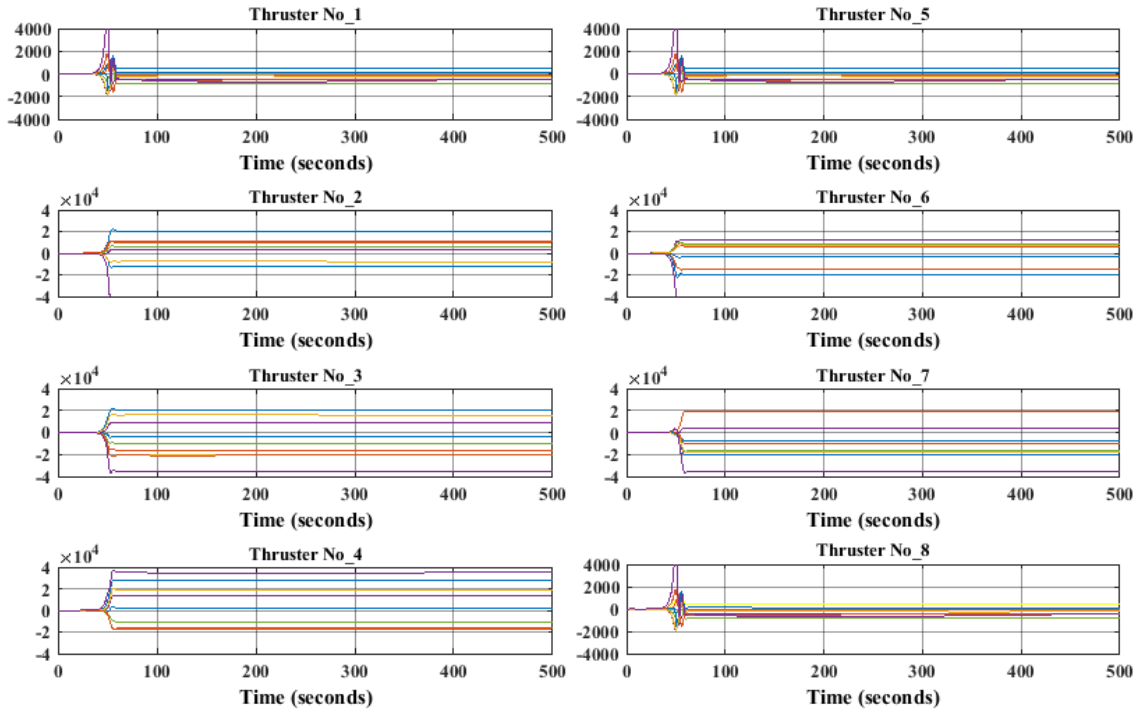


Fig. 5.43 Euresis Filter – 1^{st} , 5^{th} and 8^{th} Thruster (Total Failure)

Testing under off-Nominal Conditions: Noise, Sensitivity to Parameter Change, Robustness

The Euresis Filter design method exhibited a remarkable performance under enhanced noise conditions, higher than the other two filters, and several orders of magnitude higher than the nominal values on a simulation environment. In fact, this design was tested on the full simulator as it was directly designed in the simulation environment. However, the Euresis Filter was not able to accurately detect or identify anything other than total failures, nor did it exhibit the striking robustness characteristics of the other two filters. The marked difference between the three design methods could be attributed to the specific directionality of the residual, which the other methods hinge on, a quantity that remains invariant under scaling modifications such as severity. The Euresis Filter method, on the other hand, is based on relative residual magnitudes, which are directly influenced by scaling modifications and whose FDI performance, can be seriously degraded. Moreover, the Euresis Filter takes longer to FDI even under total failure, lets alone a partial

one, in which case the inherent plant instability may kick-in, affecting the residual magnitudes more significantly than the fault itself, particularly if it is small, thus confusing the FDI process. On the other hand, the impressive performance under severe disturbance and measurement noise conditions can be easily understood, since each of the (fault) models in this Euresis Filter is a KF, which optimally rejects noise.

A New Approach to MMKF

During the extensive testing of the Euresis Filter, we observed that it could not detector identify severity for any less than a total (100%) failure (not even a 99% failure) with the existing set of eight plus one models. For this reason, the Euresis Filter was enhanced with the necessary set of models to exactly reflect the failure severity magnitudes, taken in increments of 0.05, with an ultimate number of 160 plus one models, for the entire range of severity in the range of $[0,1]$, for all eight thrusters. The first test was executed in a small scale with a limited subset of the entire set of models. The results were at best confusing, concluding that even though this method performs extremely well, exhibiting an almost infinite tolerance to noise (NF=100000), albeit for total failures only, it breaks down as soon as the failure severity is anything less than 99% to 100%. On the other hand, the method was robust to up to 20% parameter variations, in the scenario of total failure and under extreme noise factor conditions.

The performance of this method is very puzzling and at odds with the theoretical basis. Two of the cases observed in the testing, which are indicative of the problems encountered, are presented below.

In the first case, there was a total failure in one thruster in the plant, and a corresponding model in the Euresis Filter setup with a 50% failure, i.e. a severity of 0.5. This case was detected and simultaneously *de facto* identified, even though the model did not exactly match the true plant. In the second case, there was a 0.5 failure severity in one thruster in the plant and a model reflecting a total failure in the corresponding thruster in the Euresis Filter setup. This case was not detected nor identified.

An alternative explanation that can be given is that the plant is coupled, so that, if a thruster is partially absent, this information is cloaked in the output magnitude alone, without any additional directionality or spectral information, due to the coupling. Additionally, some of the effects of the partially failed thruster may

be compensated for in the coupled plant. The models do not entirely reflect this coupling, as is the case in the designs of previous studies [15, 29]. In other words, not entirely including the overall system cross coupling in the models may be a severe handicap for the method, as it relies on exact model matching. Thus, it seems that choosing the number of models in the bank of MMKFs solely based on failure severities of the thrusters does not adequately represent the entire model set for this method.

An extremely large and complex set of models would appear to be required rendering the method cumbersome, costly and, ultimately, inefficient. Its running time would also be dramatically increased, which makes it unsuitable for real time implementation.

A Monte Carlo set of 1000 simulations was also run with varying severity levels, as shown below:

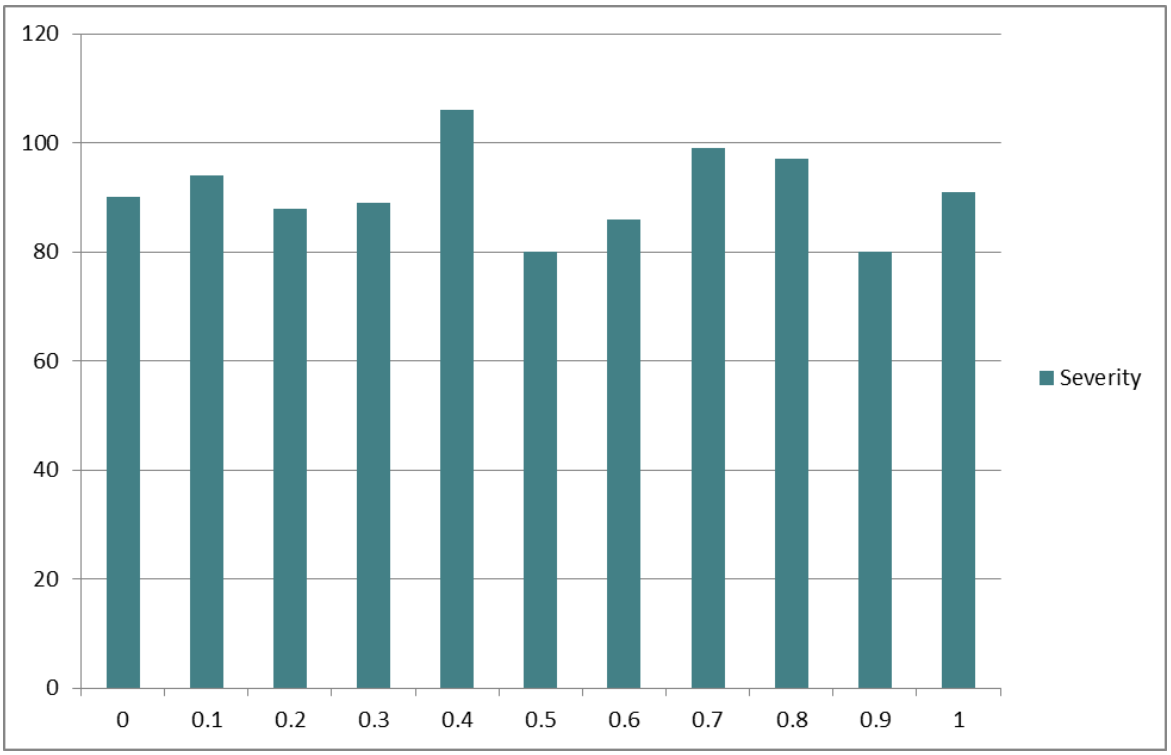


Fig. 5.44 Euresis Filter - Distribution of Severity Profiles

The frequency of failed thrusters is represented by the pie chart on Figure 5.45. Out of the 1000 runs, only 288 failed thrusters were correctly detected and identified, with the corresponding severity profiles as shown in figure 5.46

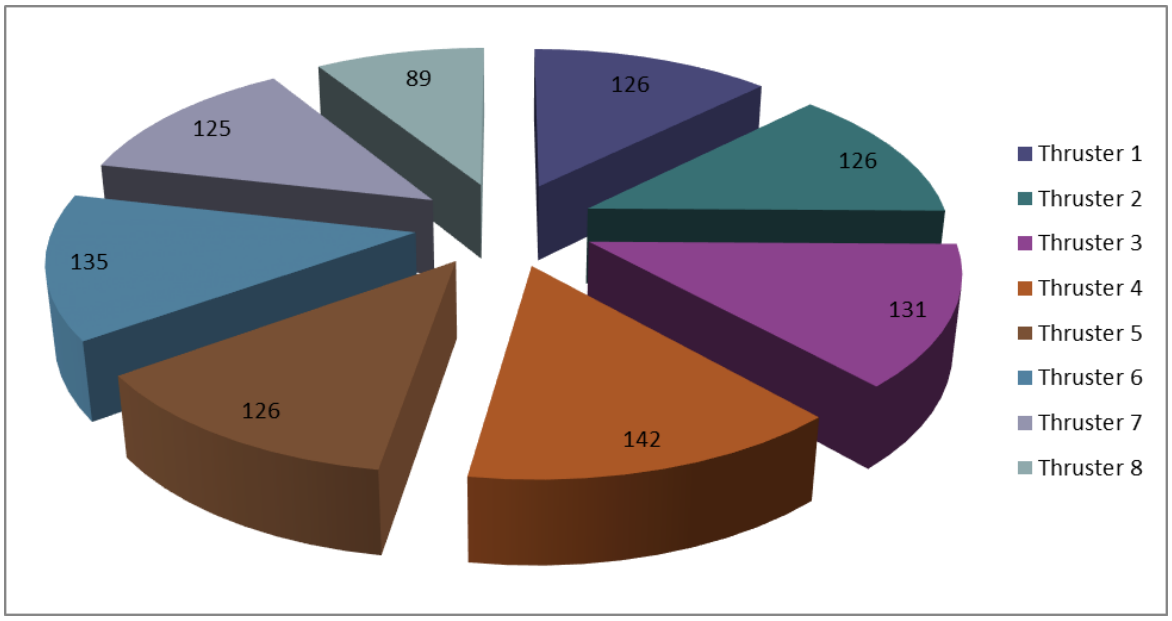


Fig. 5.45 Euresis Filter - Distribution of Thrusters' Failures

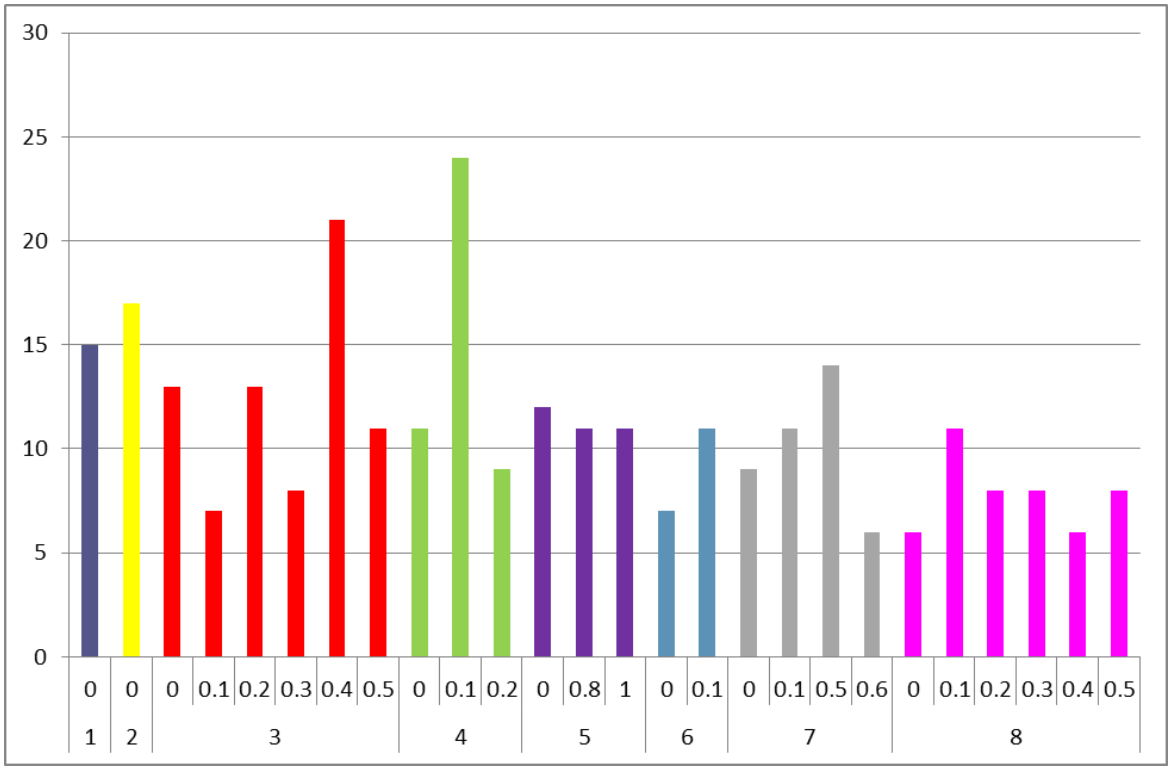


Fig. 5.46 Euresis Filter - Distribution of Correctly Identified Failed Thrusters

In Figure 5.46, the value zero corresponds to total failure, 0.1 to 90% failure, and so on. Except for the few cases (11) with only 20% failure for the 5^{th} thruster and 6 cases of 40% failure for 7^{th} thruster, all other failed thrusters with their failure severities values (that could be identified correctly) ranging from 50% and higher, with most of them in the range of 90% to 100% failure. Despite its poor performance, this setup provides an excellent estimate of the plant state, even under extreme noise conditions. However, one might not need to resort to such a costly apparatus; a well-tuned Kalman Filter alone could be sufficient.

5.4 Design Method 3: Euphoria Filter

The Euphoria Filter was first run extensively to establish maximum total noise levels and minimum severity magnitudes that can be correctly identified. The results are shown in Table 5.6.

Table 5.6 Euphoria Filter - Severity Magnitudes and Noise levels of Validation Plan

Min Constant Failure	Max Noise Factor
1(total failure)	1000
0.1	1000
0.01	1000

Note: Here the term noise factor refers to a scaling factor, from nominal, simultaneously, for all disturbances and sensor noise(s) in the system; e.g. a noise factor of 10 means that all the various nominal noise levels in the system are amplified by a factor of 10.

The filter was tested extensively for combinations of minimum leakage (slope) versus the maximum noise that could be identified. In fact, the parameter α in our leakage model ranged from 0.0001 and up, with the value of 0.0001 representing a leakage of 0,01 after 100 s. The results are shown in 5.7.

Table 5.7 Euphoria Filter - Leakage Magnitudes and Noise Levels of Validation Plan

Min Leakage Rate	Max Noise Factor
0.001	100
0.01	100
0.1	100

It is also worth noting that given that FDI occurs under reasonable noise factor conditions (NF=10) in less than 6 s (based on the results and the findings). This means that, indeed, the Euphoria Filter is able to identify a very slow leakage, reaching a magnitude of less than 0.005 at the time of identification. Intermittent failures of very small magnitude were also tested with the same success. This is attributable to the fact that FDI is achieved within just a few seconds, a time interval which is much shorter normally than the timing cycle of intermittent failures. Indeed, as the extensive simulations have shown, the Euphoria Filter exhibits excellent

performance and robustness to dramatically enhanced noise and minimal severity and leakage rate magnitudes. The Monte Carlo analysis of this filter was based on the same framework as for the Diagnosis Filter (section 5.2).

Six sets of 10000 Monte Carlo simulations were executed:

i. nominal parameter settings, to establish a baseline performance with -10% variation in centre of mass (nominal value $d^* = 0.376$) and noise factor of 100.

ii. random up to $\pm$10% misalignment, which corresponds to up to 3.6 degrees, -10% variation in centre of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 10.

iii. random up to $\pm$10% misalignment, which corresponds to up to 3.6 degrees, -10% variation in centre of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 100.

iv. random up to $\pm$20% misalignment, which corresponds to up to 7.2 degrees, +20% variation in center of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 10.

v. random up to $\pm$20% misalignment, which corresponds to up to 3.6 degrees, +20% variation in center of mass (nominal value $d^* = 0.376$) and suspension and drag-free stiffness (for each stiffness and on each axis) with a noise factor of 100.

vi. nominal parameter settings, to establish a performance perimeter bounds with -10% variation in centre of mass (nominal value $d^* = 0.376$) and noise factor of 1000.

Severity also varied randomly from 0.01 to 1, which corresponds to failures from 1% - 100%. All the eight thrusters were included.

Monte Carlo Results

i. In this case, which has nominal parameter values (except for d) but which nonetheless represents a significant departure from nominal conditions, because of the dramatically enhanced noise factor and the 10% variation in d, the

results were as follows: absolutely 100% true detection with 17 cases of missed identification, which makes the true identification percentage equal to 99.83%. True severity identification was also achieved at 99.83%. The figures below, give a succinct picture of the results obtained in this case. In addition, there were no false alarms and no missed detection.

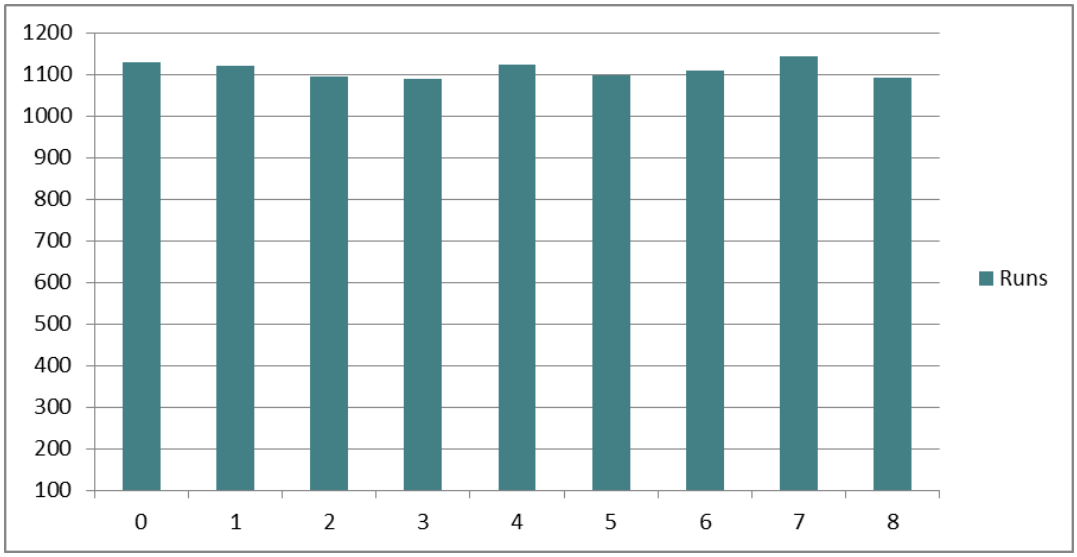


Fig. 5.47 Euphoria Filter - Distribution of Failed Thrusters for the 1^{st} case

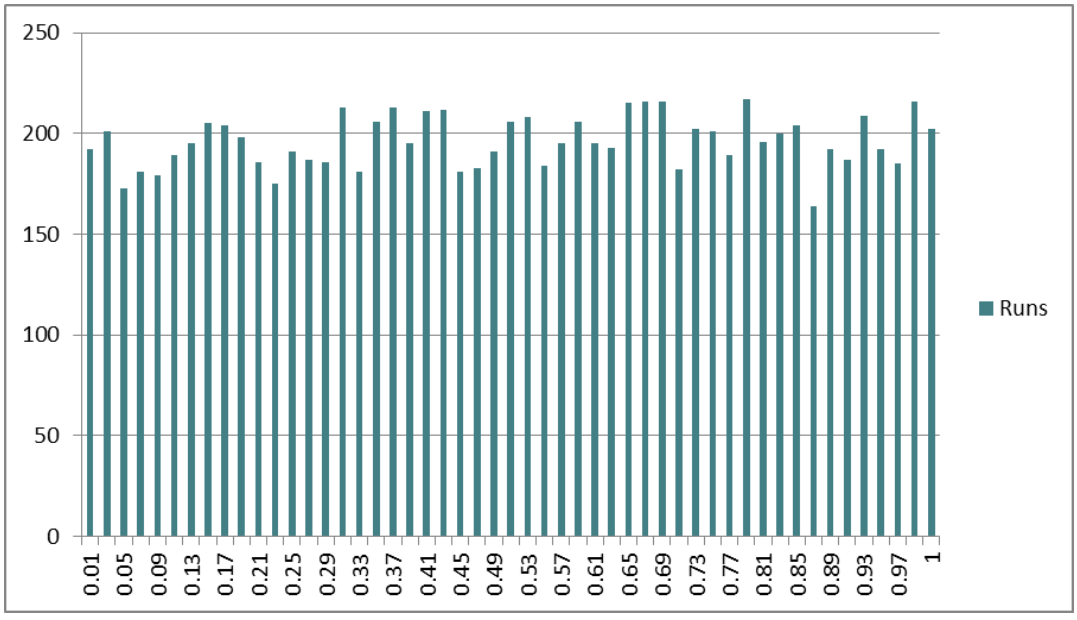


Fig. 5.48 Euphoria Filter - Distribution of Severity Magnitudes for the 1^{st} case

Table 5.8 Euphoria Filter - Missed Identification Cases for the 1^{st} case

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
1	0.01	6	-0.008815061	-0.01
1	0.01	6	-0.008339247	-0.01
1	0.01	6	-0.008014028	-0.01
1	0.01	6	-0.007846304	-0.01
6	0.01	1	-0.009638742	-0.01
1	0.01	6	-0.007824139	-0.01
6	0.01	1	-0.009537004	-0.01
1	0.01	6	-0.008144948	-0.01
6	0.01	1	-0.010671032	-0.01
6	0.01	1	-0.009537004	-0.01
1	0.01	6	-0.008838032	-0.01
6	0.01	1	-0.010075855	-0.01
6	0.01	1	-0.009638742	-0.01
6	0.01	1	-0.00999651	-0.01
1	0.01	6	-0.008144948	-0.01
6	0.01	1	-0.009756842	-0.01
6	0.01	1	-0.009979775	-0.01

As derived from Table 5.8, there are 17 missed identification cases in the 10000 runs, all of them associated with a severity magnitude of 0.01. It is remarkable that the algorithm consistently identified correctly the absolute value of the severity magnitude. The negative sign in this case can be considered as flagging a wrong identification. The fact that the 1^{st} and 6^{th} thrusters were consistently mistakenly identified can probably be attributed to the fact that their directions appear close in the presence of the dramatically increased noise factor. The filter, by design, optimally rejects disturbances and sensor noise, at the levels included in this design.

However, the optimal designs are sensitive to variations of the optimization parameters; here an increase by two orders of magnitude of the noise above the levels used in the filter design is probably the cause of this sensitivity.

ii. In this case, there was a 100% true detection, 100% true identification and a 100% true severity identification; no false alarms; no missed detection. The results of this set are shown below.

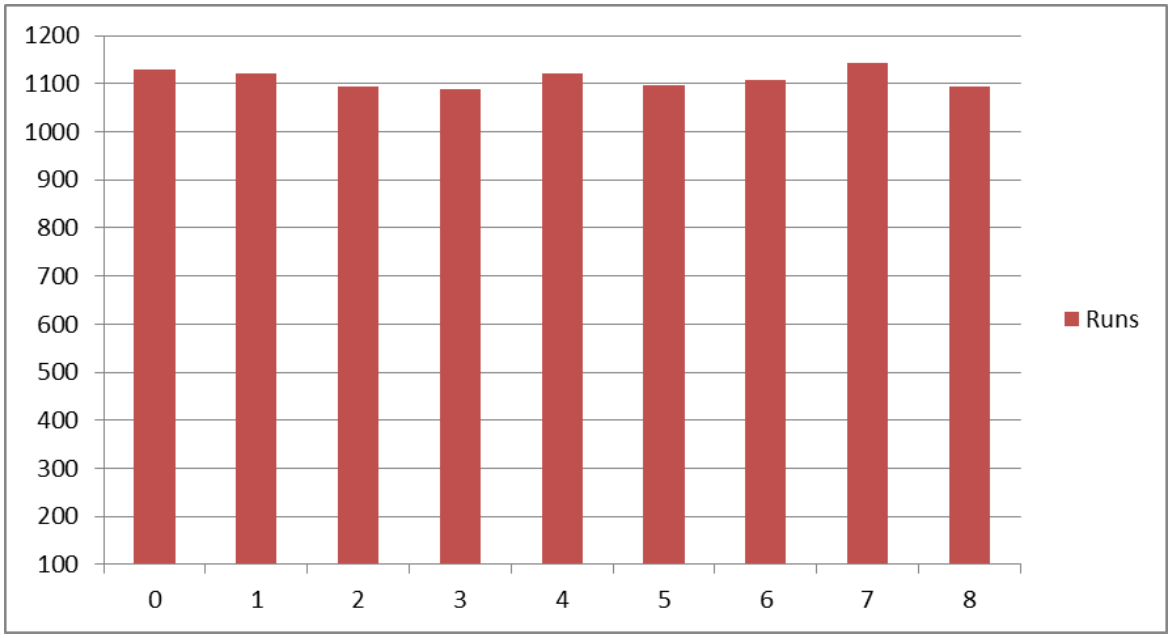


Fig. 5.49 Euphoria Filter - Distribution of Failed Thrusters for the 2^{nd} case

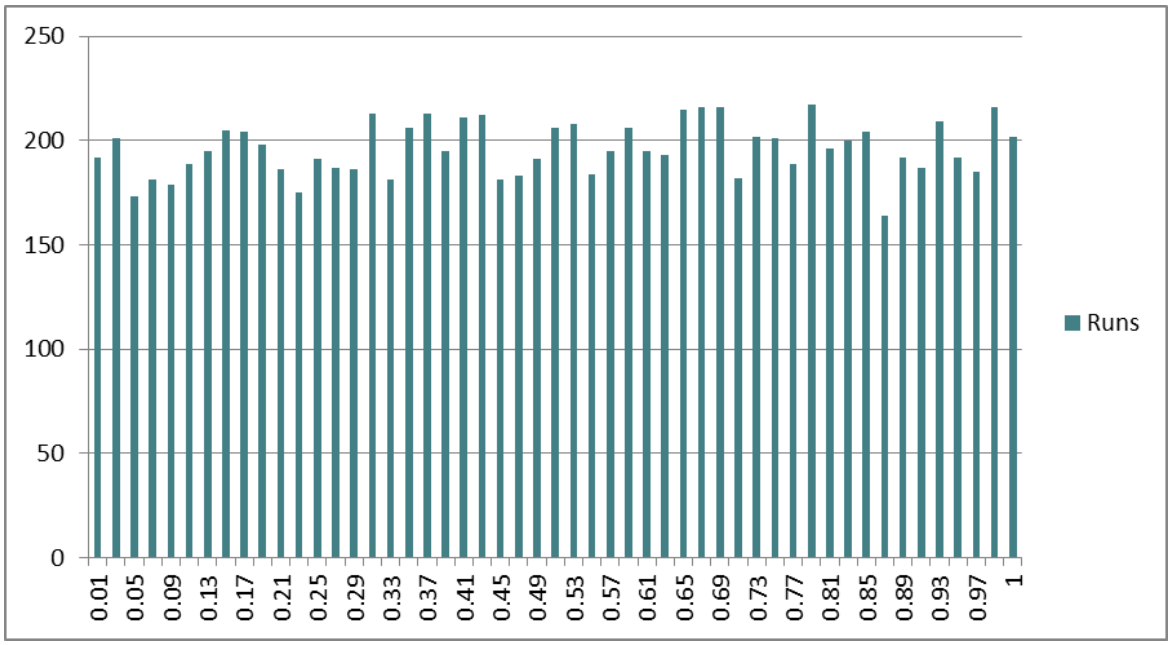


Fig. 5.50 Euphoria Filter - Distribution of Severity Magnitudes for the 2^{nd} case

iii. In this case, there was a 100% true detection and 99.83% true identification; true severity identification was also 99.83%. As the Table 5.9 shows, there were 17 cases of missed identification, all of which with severity 0.01. Again, the absolute value of severity was correctly identified, with negative severity flagging missed identification. There were no false alarms and no missed detection. The results are given below.

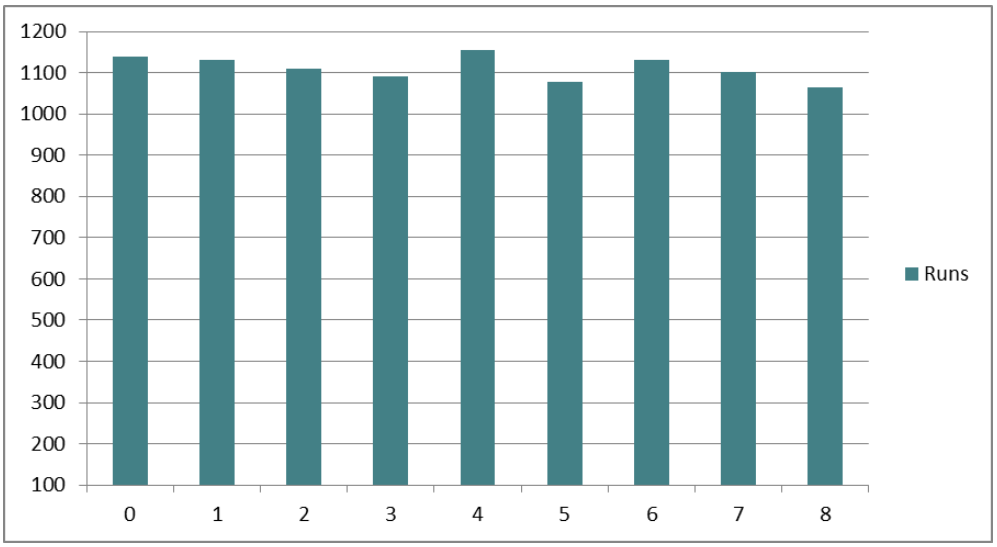


Fig. 5.51 Euphoria Filter - Distribution of Failed Thrusters for the 3^{rd} case

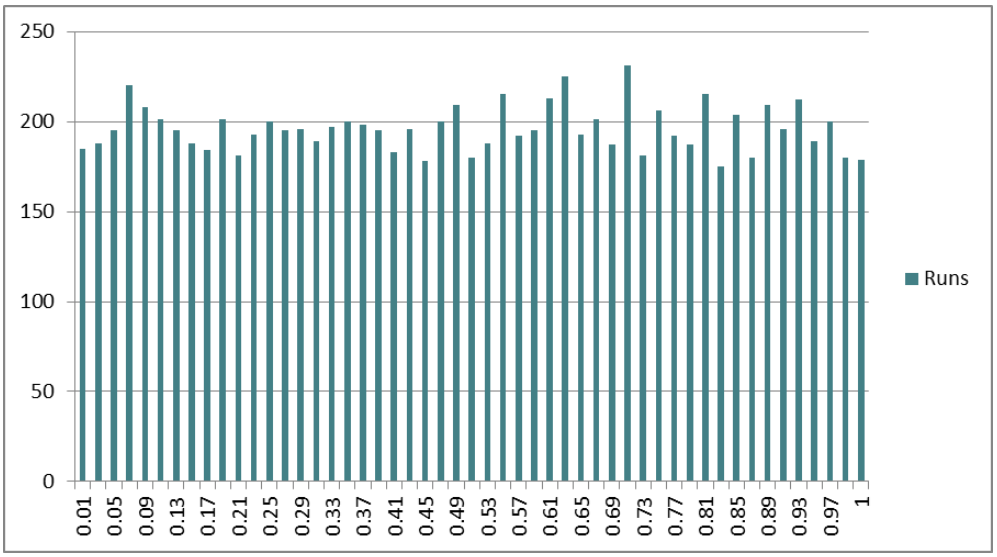


Fig. 5.52 Euphoria Filter - Distribution of Severity Magnitudes for the 3^{rd} case

Table 5.9 Euphoria Filter - Missed Identification Cases for the 3^{rd} case

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.01	1	-0.009676715	-0.01
2	0.01	1	-0.008158162	-0.01
1	0.01	2	-0.008104851	-0.01
1	0.01	2	-0.009513478	-0.01
1	0.01	6	-0.006638354	-0.01
2	0.01	1	-0.008020664	-0.01
1	0.01	2	-0.006523245	-0.01
1	0.01	6	-0.005691955	-0.01
1	0.01	6	-0.008245931	-0.01
2	0.01	1	-0.008653869	-0.01
6	0.01	1	-0.009638742	-0.01
1	0.01	6	-0.00734796	-0.01
6	0.01	1	-0.009555384	-0.01
2	0.01	1	-0.010141942	-0.01
6	0.01	1	-0.009700117	-0.01
6	0.01	1	-0.009638742	-0.01
6	0.01	1	-0.008794409	-0.01

iv. In this case, there was a 100% true detection and a 99.99% true identification and true severity magnitude identification. No false alarms; no missed detection. There was only one case of missed identification, as shown in Table 5.10, with the negative severity flagging it.

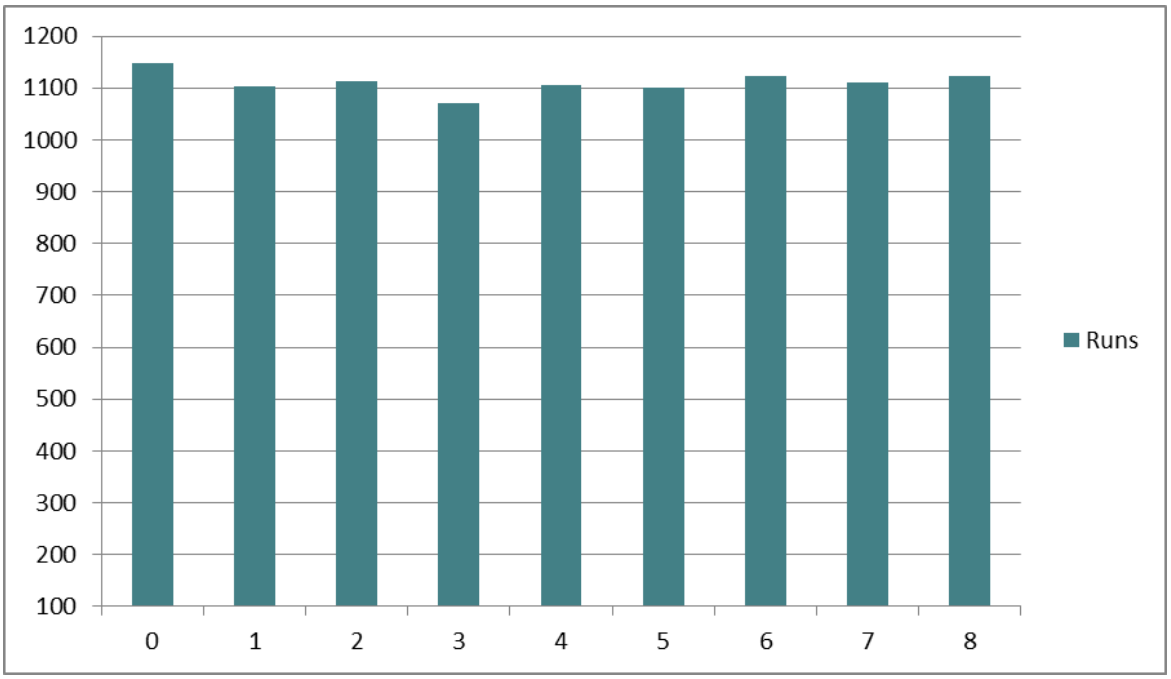


Fig. 5.53 Euphoria Filter - Distribution of Failed Thrusters for the 4^{th} case

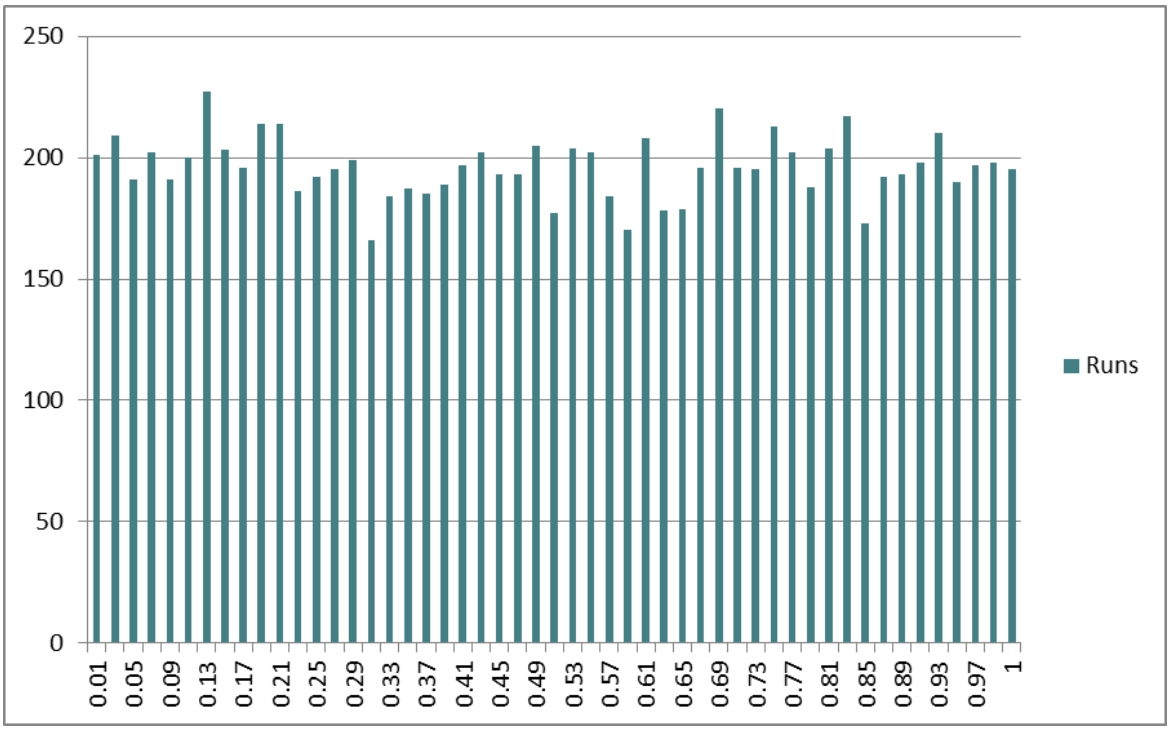


Fig. 5.54 Euphoria Filter - Distribution of Severity Magnitudes for the 4^{th} case

Table 5.10 Euphoria Filter - Missed Identification Cases for the 4^{th} case

Random Failed Thruster	Min Angle	FDI	Identified Severity	Round-Off Severity
4	27.73904386	2	-0.787899938	-0.79

v. In this case, there was a 100% true detection and a 99.81% true identification and true severity magnitude identification. No false alarms; no missed detection. The 19 cases of missed identification were again with severity magnitude of 0.01, identified with negative severity. The absolute value of the severity magnitude was correctly identified in all cases. Another interpretation of this would be to say that we have 19 cases of missed identification of a failed thruster, failing with a severity magnitude of 0.01.

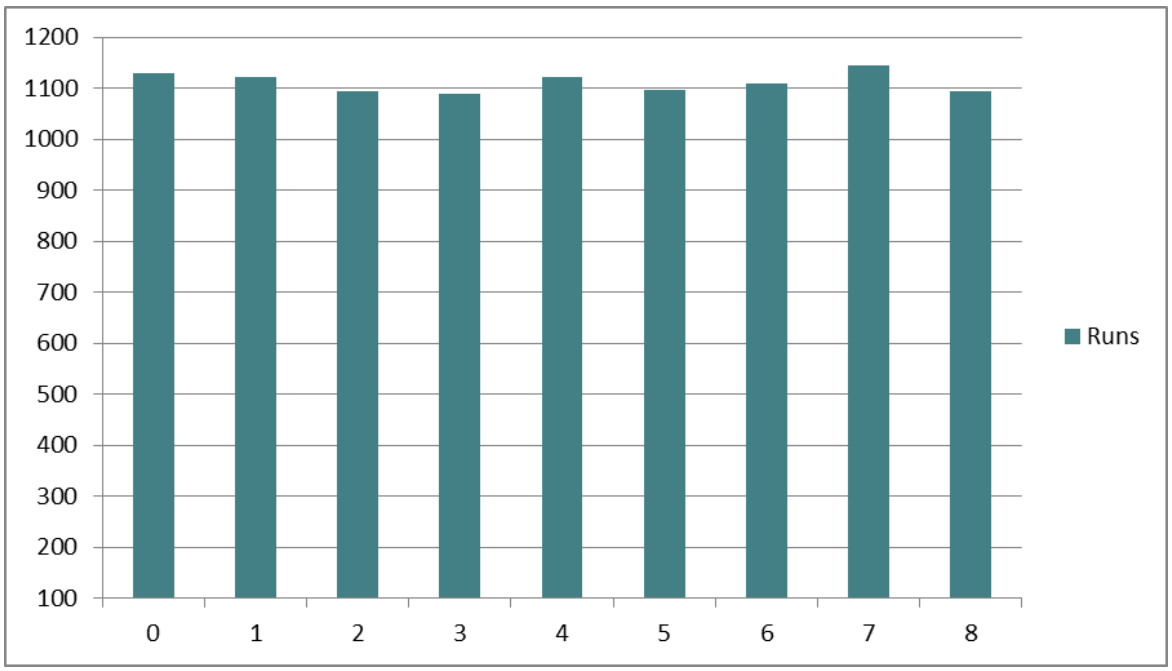


Fig. 5.55 Euphoria Filter - Distribution of Failed Thrusters for the 5^{th} case

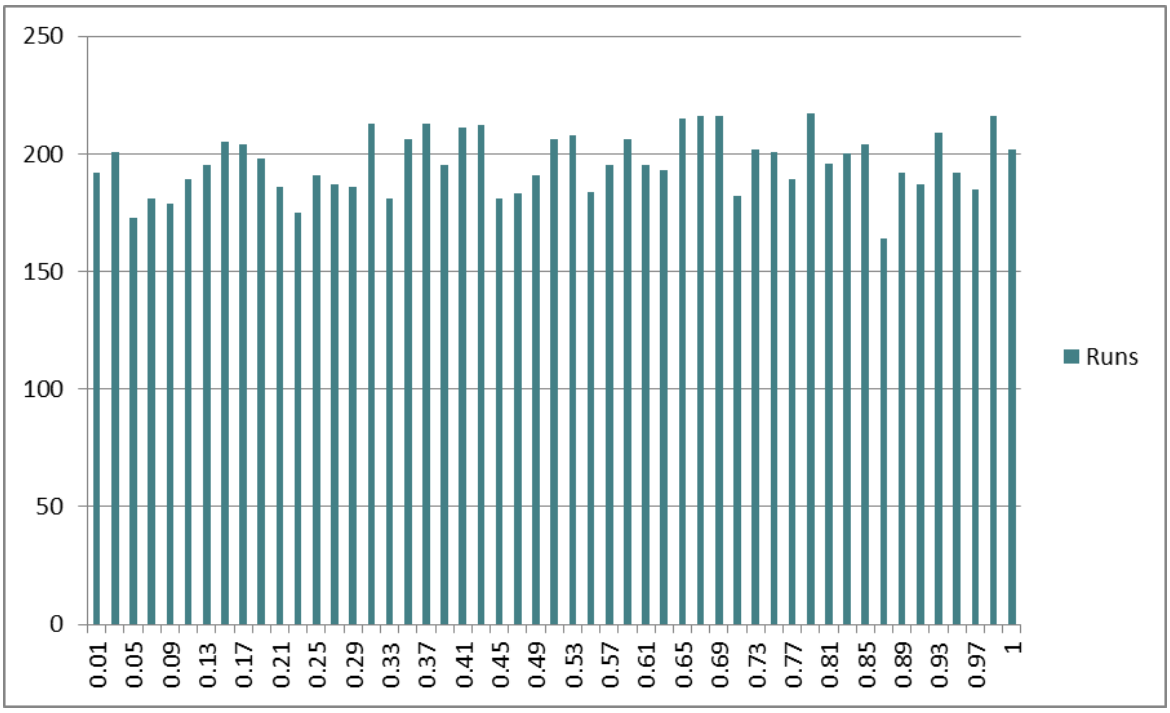


Fig. 5.56 Euphoria Filter - Distribution of Severity Magnitudes for the 5^{th} case

Table 5.11 Euphoria Filter - Missed Identification Cases for the 5^{th} case

Random Failed Thruster	Min Angle	FDI	Identified Severity	Round-Off Severity
6	0.01	1	-0.007798946	-0.01
2	0.01	1	-0.00857866	-0.01
1	0.01	2	-0.012109786	-0.01
1	0.01	6	-0.009328639	-0.01
1	0.01	6	-0.0076964	-0.01
6	0.01	1	-0.009113337	-0.01
1	0.01	6	-0.005702571	-0.01
6	0.01	1	-0.00972061	-0.01
1	0.01	6	-0.00695489	-0.01
1	0.01	6	-0.010594522	-0.01
1	0.01	2	-0.008028652	-0.01
6	0.01	1	-0.011257563	-0.01
2	0.01	1	-0.008185142	-0.01
2	0.01	1	-0.009771367	-0.01
6	0.01	1	-0.009325342	-0.01
6	0.01	1	-0.009373204	-0.01
4	0.01	2	-0.008892797	-0.01
6	0.01	1	-0.00972061	-0.01
6	0.01	1	-0.009113337	-0.01

vi. In this final set of runs, a 100% fault detection was achieved; also true identification of the failed thruster was missed in 430 cases (Appendix C), of which 302 with negative severity. However, both the negative severity as well as the positive severity were associated with missed identification; there was no particular flagging significance to negative severity. Here, the percentage of missed identification is 4.3%, which is also the percentage of missed severity identification. Therefore, in this case, we have a 95.7% true identification and a 95.7% correct severity magnitude identification. All these missed cases were associated with very low severity magnitudes, considering the level of noise factor. Severity was in the range 0.01 to 0.09, with only one case of 0.11. Here, the algorithm was unable to identify even the absolute magnitude of the severity correctly. This is not surprising, given the dramatic increase of noise factor. The algorithm sensitivity to (very significant) deviation of optimization parameters from design values is particularly pronounced here. Furthermore, even under these exacerbated circumstances, the FDI gives a 100% true detection and no false alarms or missed detection.

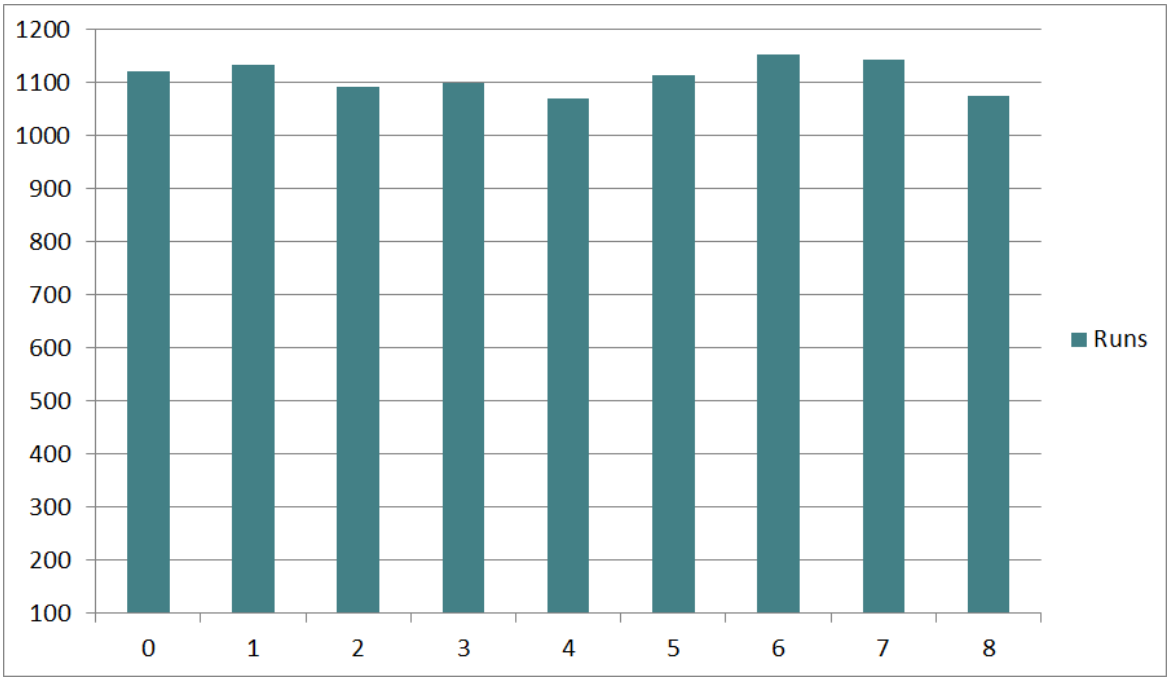


Fig. 5.57 Euphoria Filter - Distribution of Failed Thrusters for the 6^{th} case

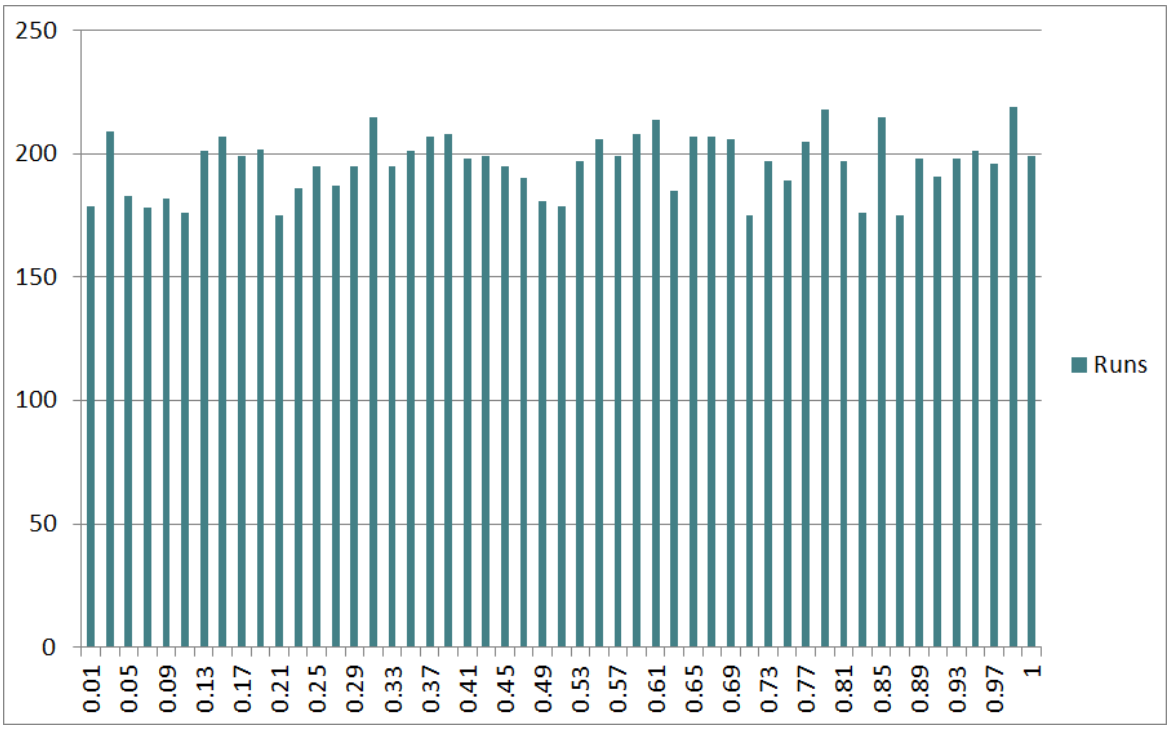


Fig. 5.58 Euphoria Filter - Distribution of Severity Magnitudes for the 6^{th} case

The 1^{st} and 6^{th} sets of the Monte Carlo runs, which are close to nominal except for d and the noise factor, clearly delineate the algorithm performance bounds with respect to noise increase from nominal, which cannot exceed two orders of magnitude, if true detection alone is not the only objective. If, however, there are such circumstances of dramatically increased disturbances and sensor noise, the algorithm still detects 100% of any thruster failure. The timing to FDI is very similar for all six runs; this was indeed the finding throughout the entire Monte Carlo simulation testing. Table 5.12 summarizes the results obtained for all six sets, both in terms of percentages as well as numerical values of the metrics considered.

Table 5.12 Euphoria Filter – Summarized Results of Six Sets of Monte Carlo Analysis

Evaluation Criteria	1^{st} SET	2^{nd} SET	3^{rd} SET	4^{th} SET	5^{th} SET	6^{th} SET
False Alarm Rate r_{fa}	0%	0%	0%	0%	0%	0%
Missed Detection Rate r_{md}	0%	0%	0%	0%	0%	4,3%
True Detection Rate r_{td}	100%	100%	100%	100%	100%	95.7%
True Isolation Rate r_{ti}	99.83%	100%	99.83%	99.99%	99.81%	95.7%
Wrong Isolation Rate r_{iw}	0.17%	0%	0.17%	0.01%	0.19%	4.3%
Average Detection Time t_{da}	< 2 s	< 2 s	< 2 s	< 2 s	< 2 s	< 2 s
Average Isolation Time t_{ia}	< 2 s	< 2 s	< 2 s	< 2 s	< 2 s	< 2 s

Figure 5.59 illustrates the FDI timing profile graphs for the case with 10% parameter variation and noise factor 10 (2^{nd} set).

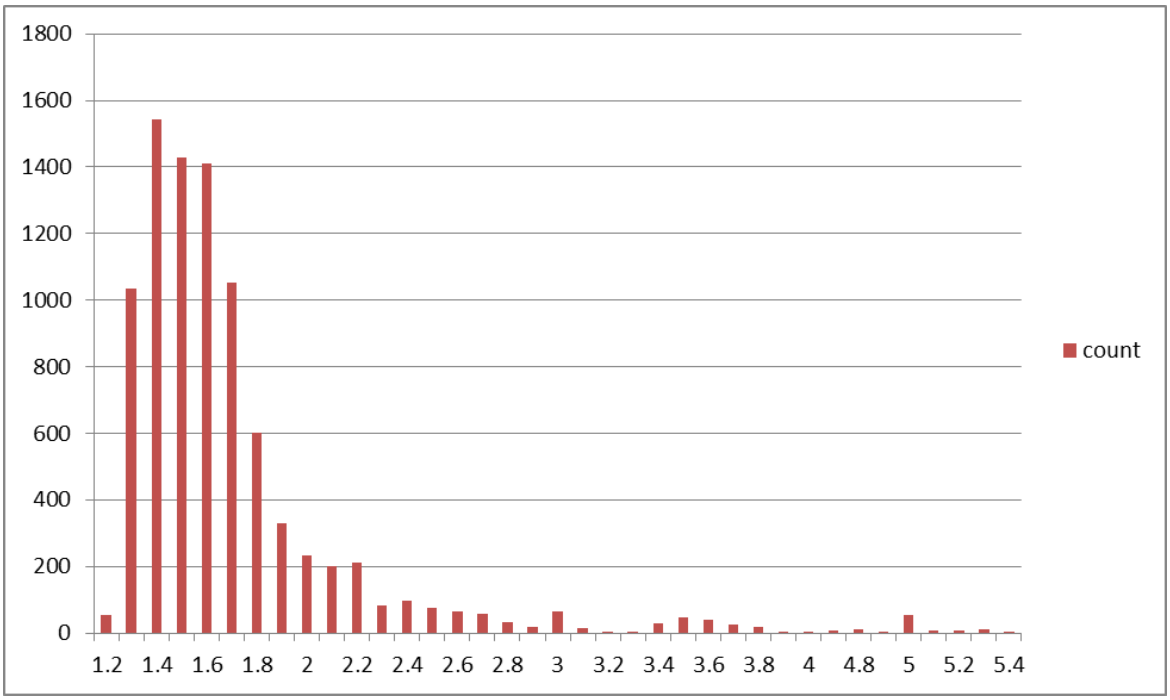


Fig. 5.59 Euphoria Filter - FDI Timing Profile for 2^{nd} Case

The same FDI timing was also observed in the more difficult cases, with an FDI convergence time less than 6 s. For comparison, the case with the 20% parameter variation and noise factor 100 in the Diagnosis Filter had an FDI convergence time of less than 23 s.

Indicative Test Runs – Leakage

In this subsection, individual runs are presented, where the thruster's failure appears as leakage of various rates. The parameter in the leakage model ranged from 0.0001 (very slow) (with a failure severity of 0.01 reached after 100 s of leakage), up to 0.01 (fast), with a failure severity of 0.1 reached after 10 s of leakage, and with parametric variations up to 20%, and noise factor up to 100. More specifically, the following test cases are shown individually:

i. Nominal, NF=1000, $d = d^* - 10\%$, $\alpha = 0{,}0001$.

ii. 10% parameter variation, NF=10, $d = d^* - 10\%$, $\alpha = 0.001$.

iii. 10% parameter variation, NF=10, $d = d^* - 10\%$, $\alpha = 0.01$.

iv. 10% parameter variation, NF=100, $d = d^* - 10\%$, $\alpha = 0.001$.

v. 10% parameter variation, NF=100, $d = d^* - 10\%$, $\alpha = 0.01$.

vi. 20% parameter variation, NF=10, $d = d^* + 20\%$, $\alpha = 0.001$.

vii. 20% parameter variation, NF=10, $d = d^* + 20\%$, $\alpha = 0.01$.

viii. 20% parameter variation, NF=100, $d = d^* + 20\%$, $\alpha = 0.001$.

ix. 20% parameter variation, NF=100, $d = d^* + 20\%$, $\alpha = 0.01$.

These individual test figures provide a bit more of an insight into the nature of the FDI process and its time profile. The time to FDI is longer for the slow leakages, but still very fast considering how subtle the failure remains for a long time. To put it differently, FDI catches up with the failure long before the failure reaches a level of 0.01 severity, which is the smallest constant severity tried in all our experiments.

The leakage of the faulty thruster (in each set of graphs the corresponded number of the failed thruster is inside the parenthesis) is according to the model $\delta = \dfrac{1}{1+\alpha t}$, where t is time and α is the leakage parameter.

The following sets of graphs corresponding to each of the cases above are shown in the order listed below:

- Leakage δ.
- Severity $\gamma = (1 - \delta)$.

- Residuals.
- Angle evolution.

The following colour code pertains to all angle evolution graphs.

Fig. 5.60 Euphoria Filter – Thrusters' Colour Code for Leakage Test Runs

i. Nominal, NF=1000, $d = d^* - 10\%$, $\alpha = 0.0001$ (Thruster No 1).

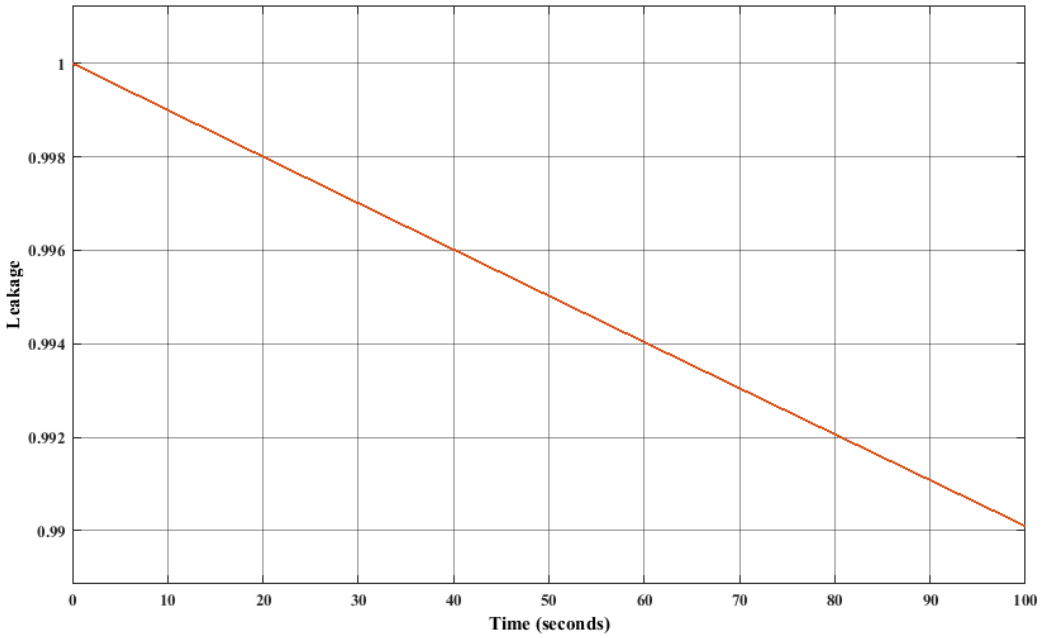


Fig. 5.61 Euphoria Filter - Graph of Leakage for 1^{st} Set

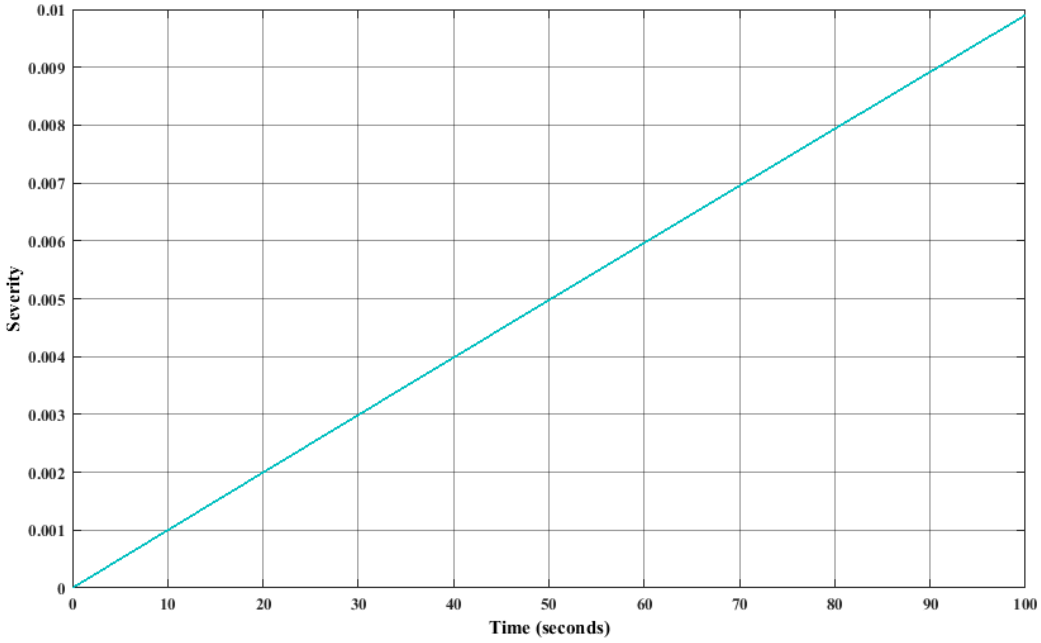


Fig. 5.62 Euphoria Filter - Graph of Severity for 1^{st} Set

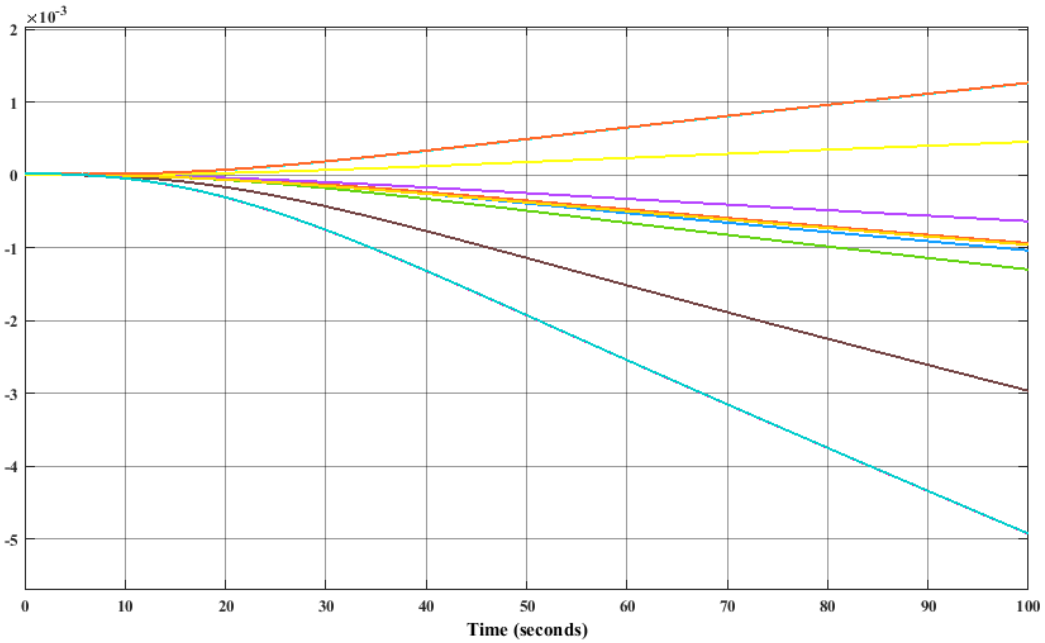


Fig. 5.63 Euphoria Filter - Graph of Residuals for 1^{st} Set

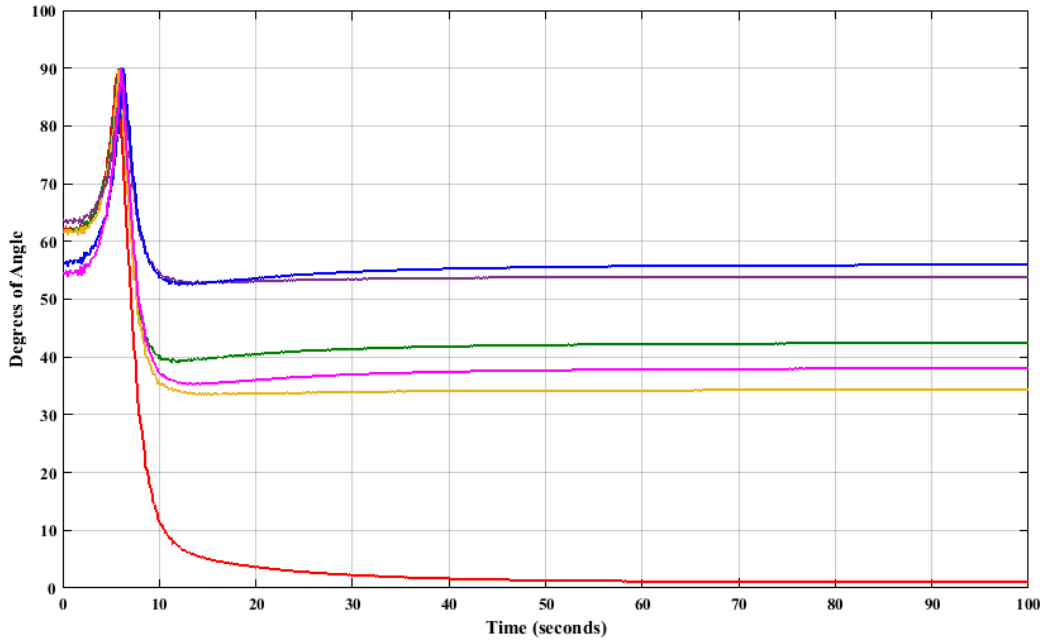


Fig. 5.64 Euphoria Filter – Thrusters' Angles for 1^{st} Set

ii. 10% parameter variation, NF=10, $d = d^* - 10\%$, $\alpha = 0.001$ (Thruster No 3).

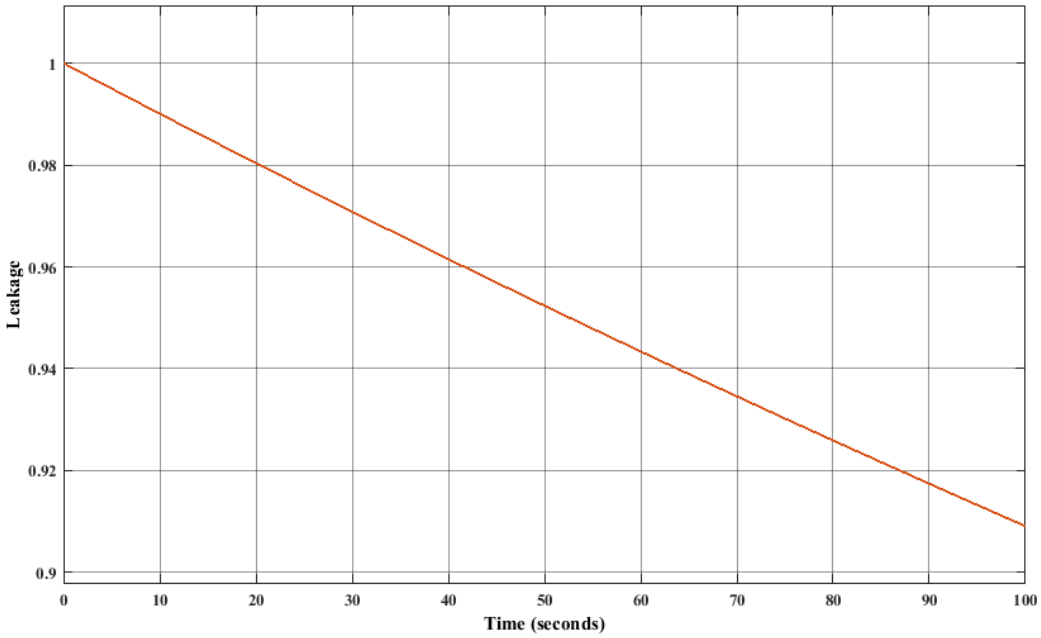


Fig. 5.65 Euphoria Filter - Graph of Leakage for 2^{nd} Set

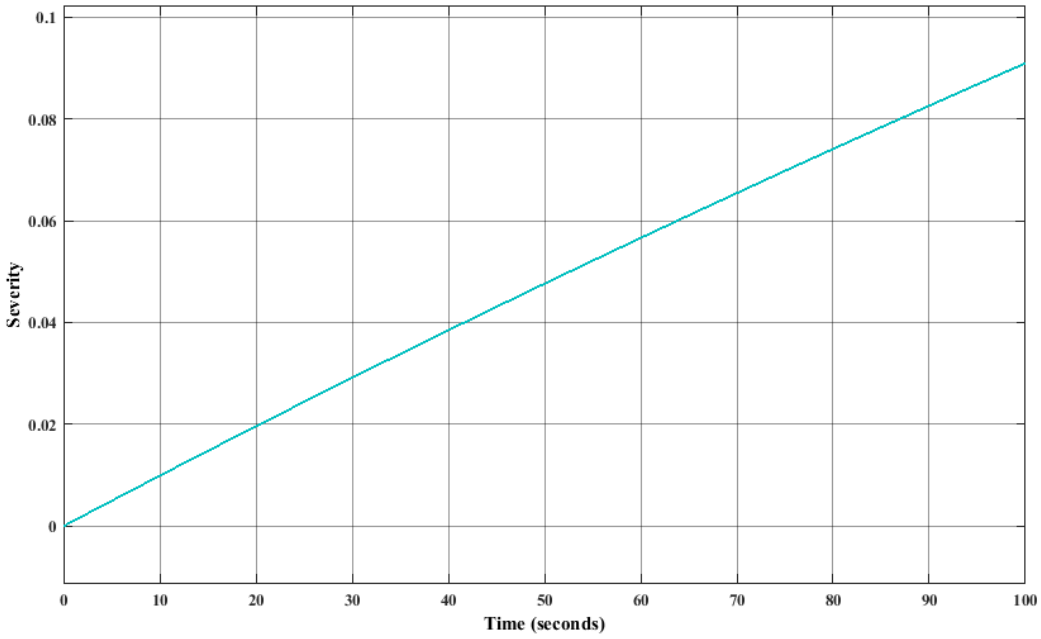


Fig. 5.66 Euphoria Filter - Graph of Severity for 2^{nd} Set

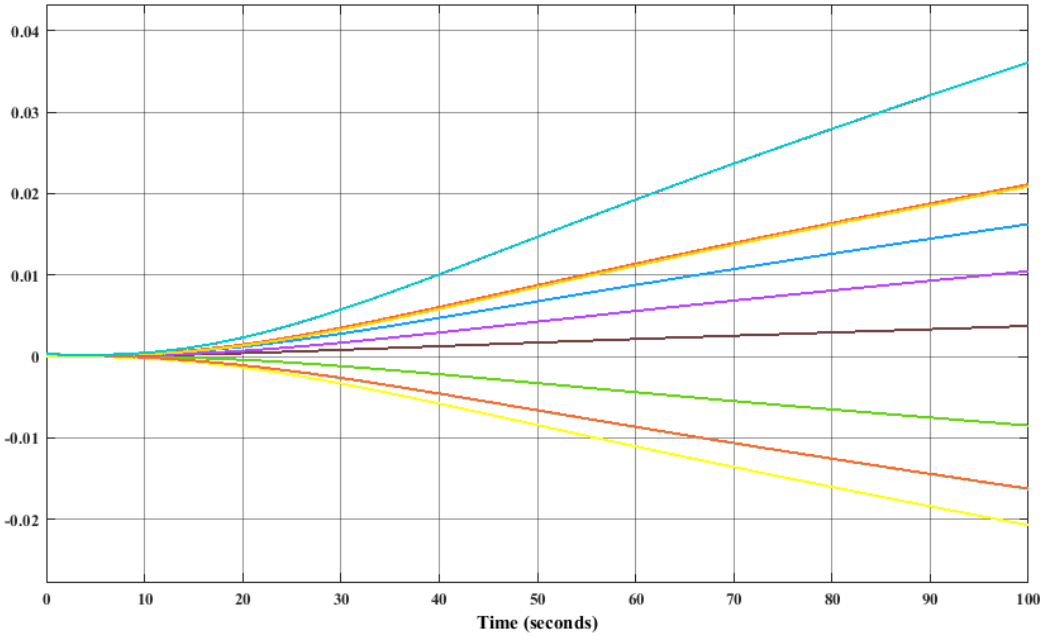


Fig. 5.67 Euphoria Filter - Graph of Residuals for 2nd Set

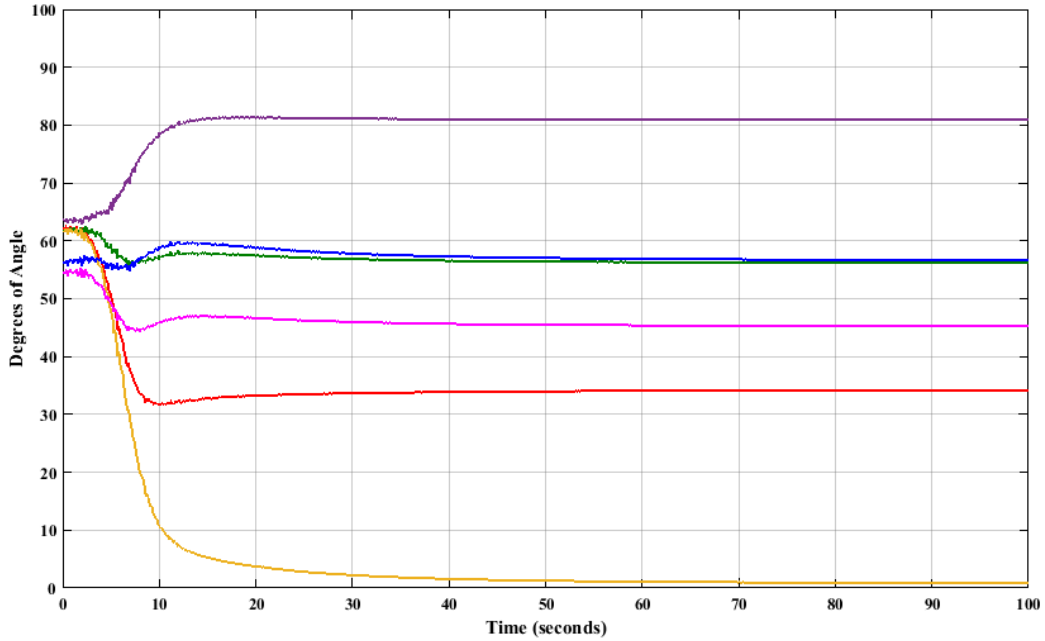


Fig. 5.68 Euphoria Filter – Thrusters' Angles for 2nd Set

iii. 10% parameter variation, NF=10, $d = d^* - 10\%$, $\alpha = 0.01$ (Thruster No 6).

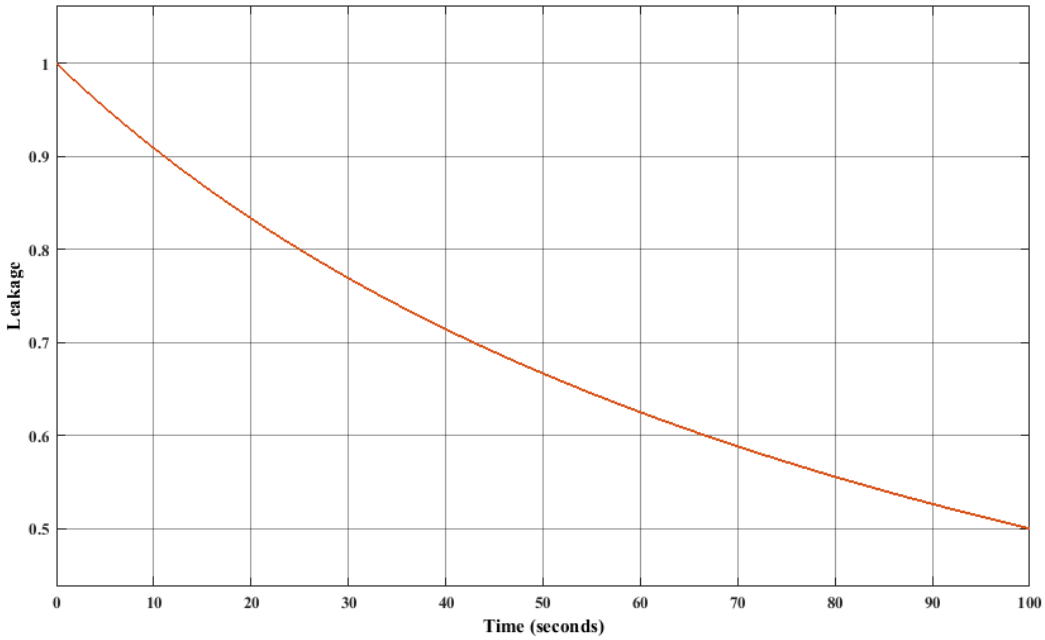


Fig. 5.69 Euphoria Filter - Graph of Leakage for 3^{rd} Set

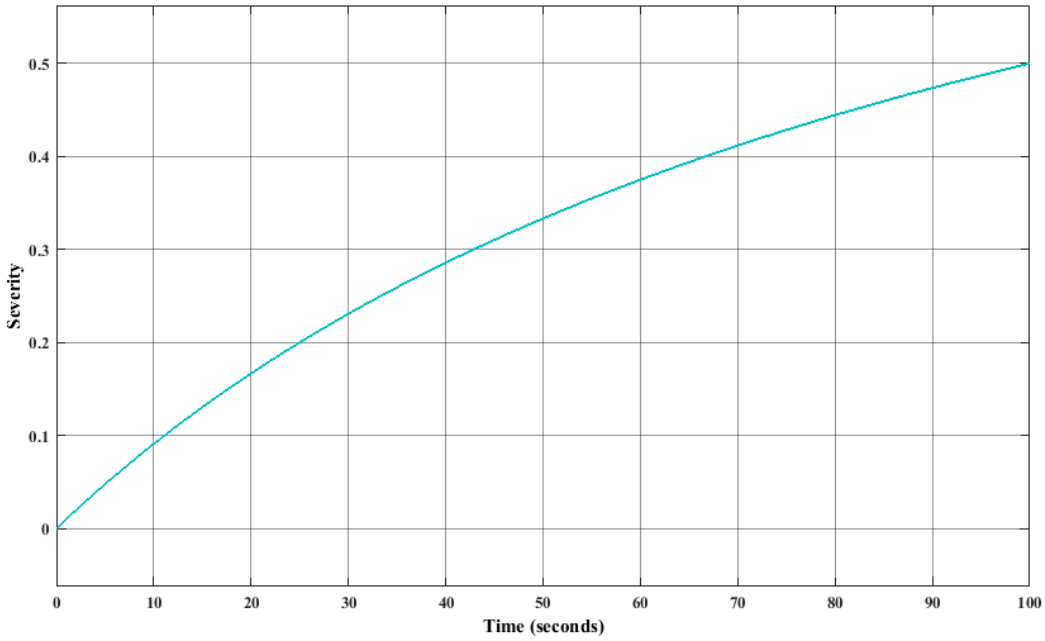


Fig. 5.70 Euphoria Filter - Graph of Severity for 3^{rd} Set

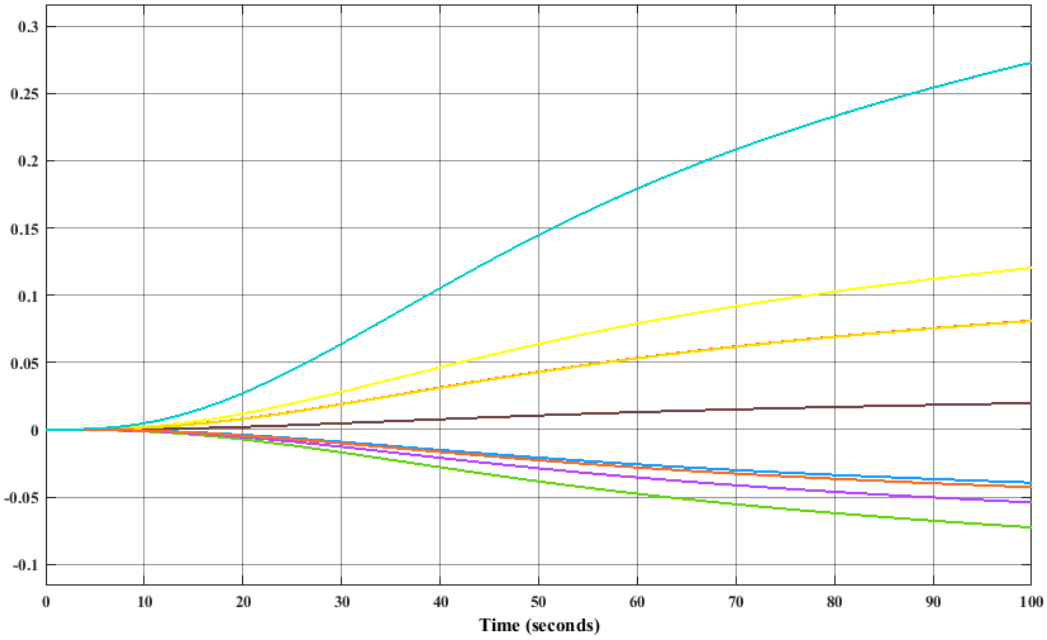


Fig. 5.71 Euphoria Filter - Graph of Residuals for 3^{rd} Set

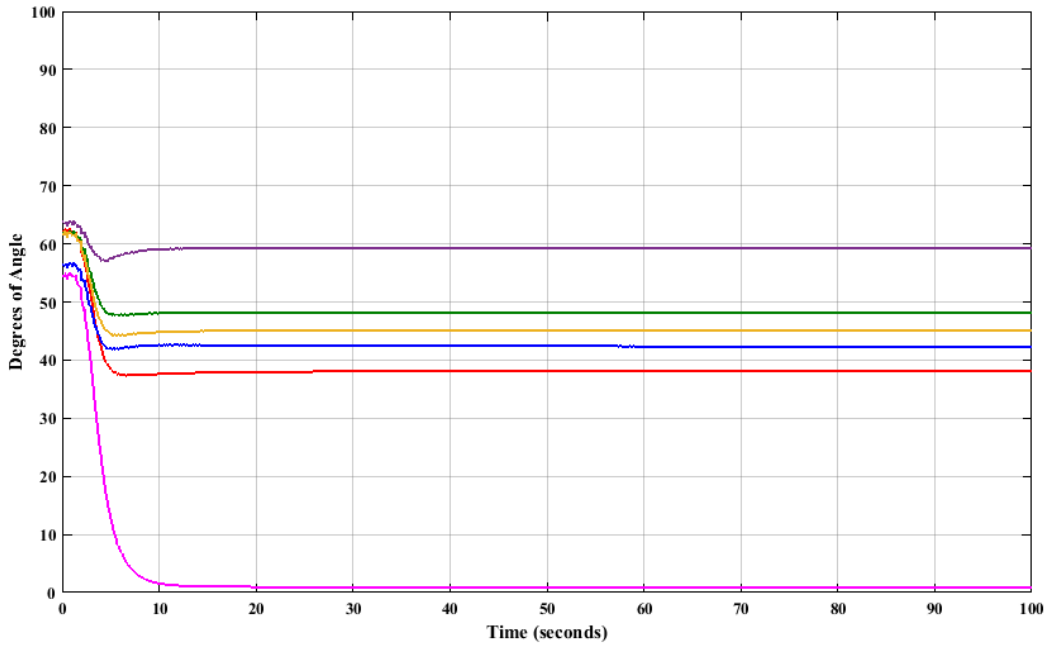


Fig. 5.72 Euphoria Filter – Thrusters' Angles for 3^{rd} Set

iv. 10% parameter variation, NF=100, $d = d^* - 10\%$, $\alpha = 0.001$ (Thruster No 5).

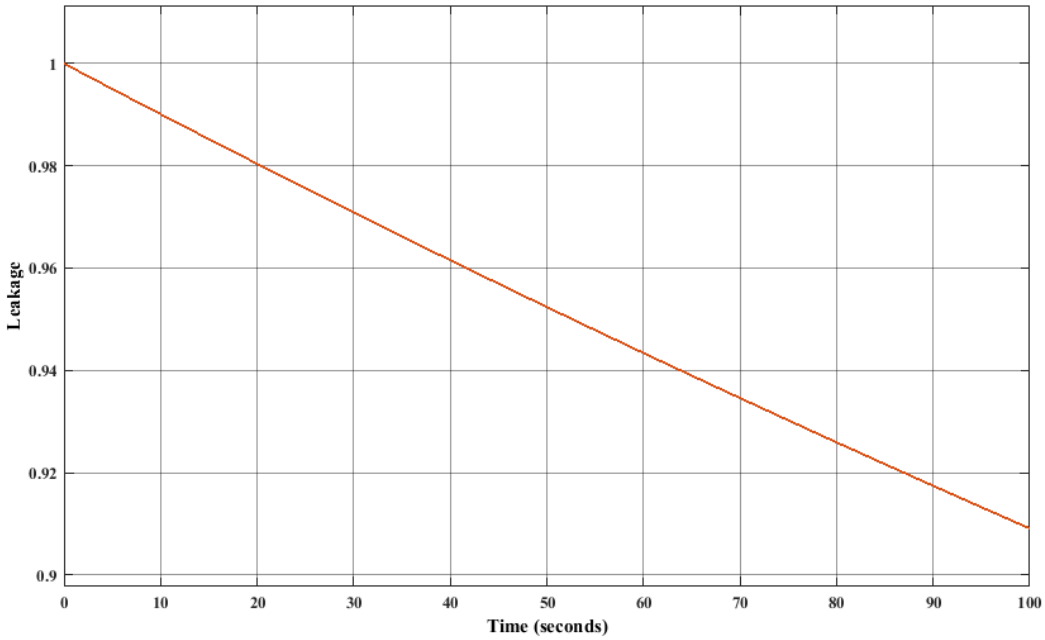


Fig. 5.73 Euphoria Filter - Graph of Leakage for 4^{th} Set

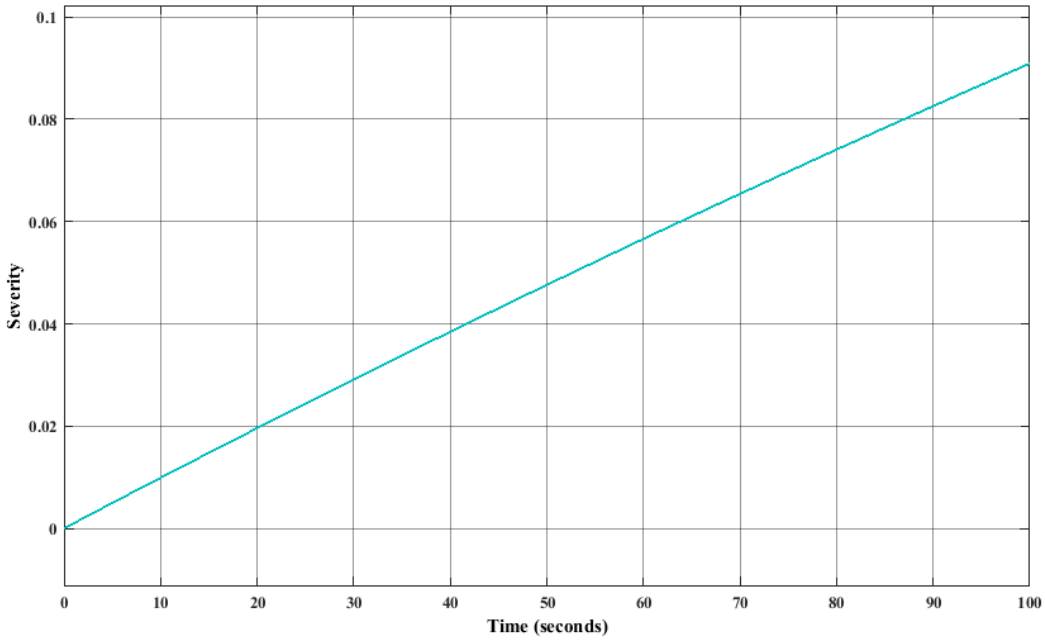


Fig. 5.74 Euphoria Filter - Graph of Severity for 4^{th} Set

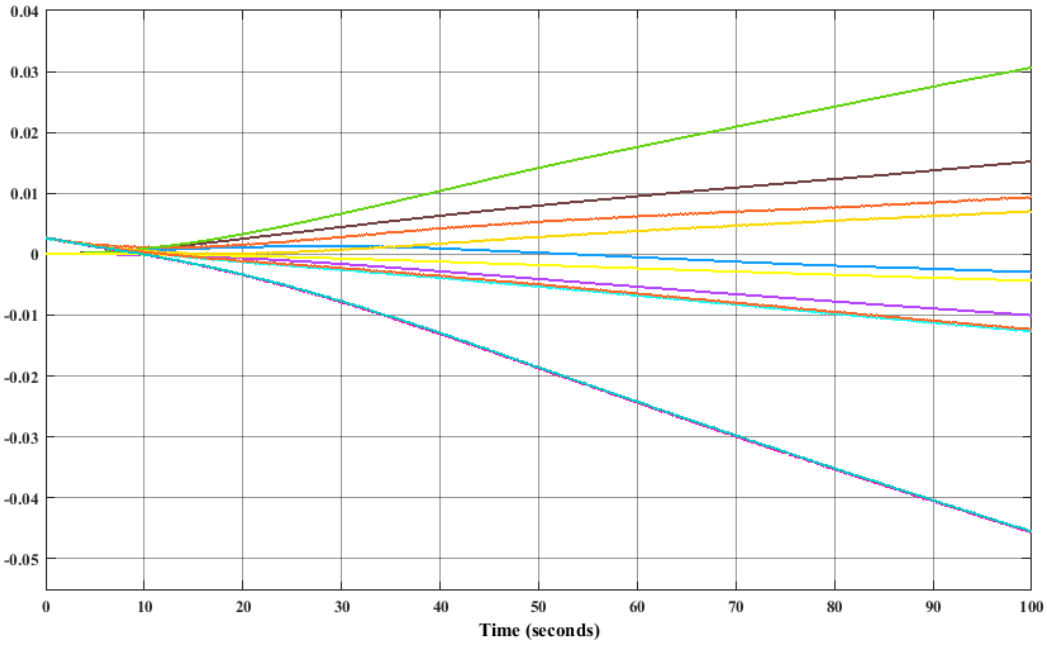


Fig. 5.75 Euphoria Filter - Graph of Residuals for 4^{th} Set

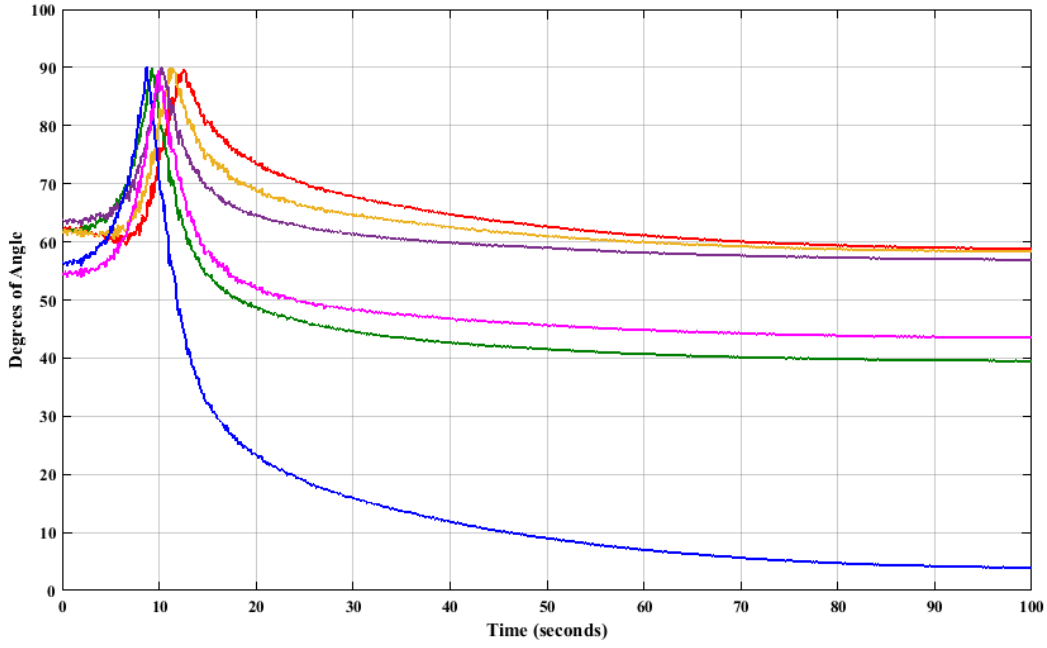


Fig. 5.76 Euphoria Filter – Thrusters' Angles for 4^{th} Set

v. 10% parameter variation, NF=100, $d = d^* - 10\%$, $\alpha = 0.01$ (Thruster No 4).

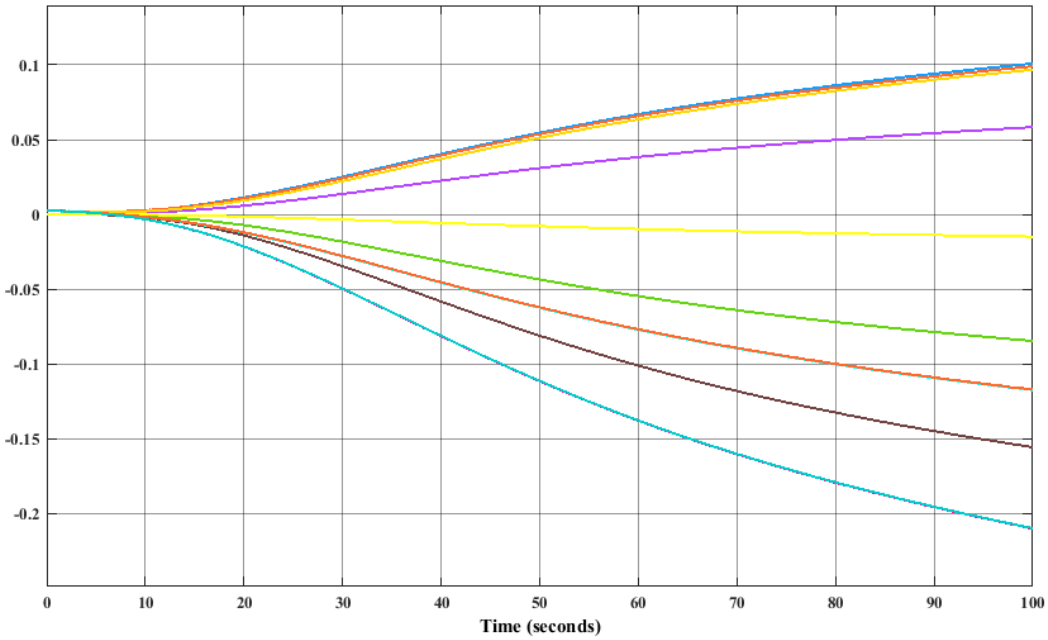


Fig. 5.77 Euphoria Filter - Graph of Leakage for 5^{th} Set

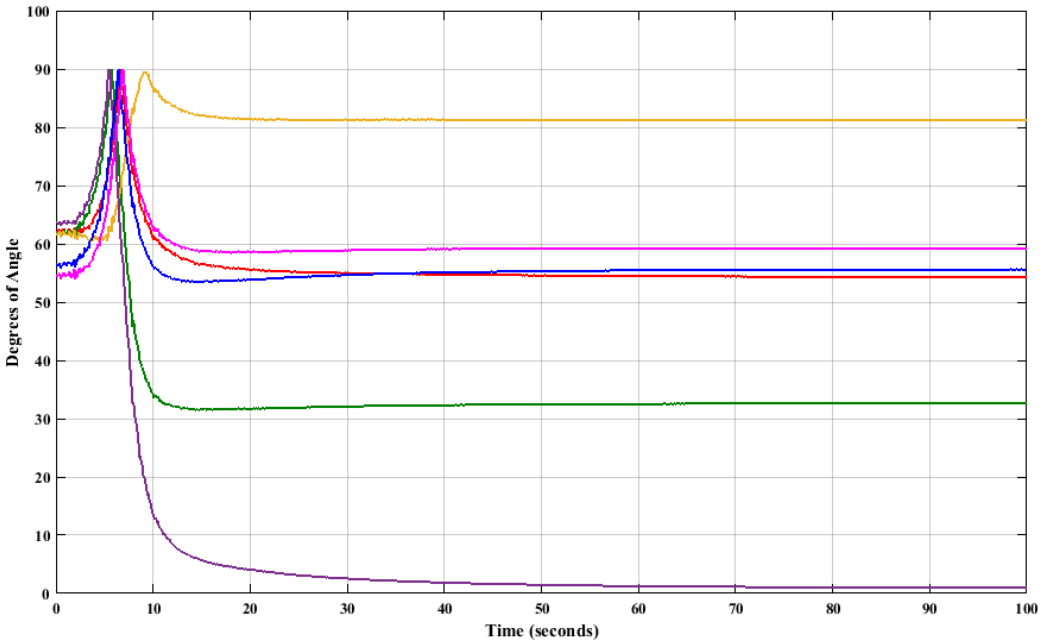


Fig. 5.78 Euphoria Filter - Graph of Severity for 5^{th} Set

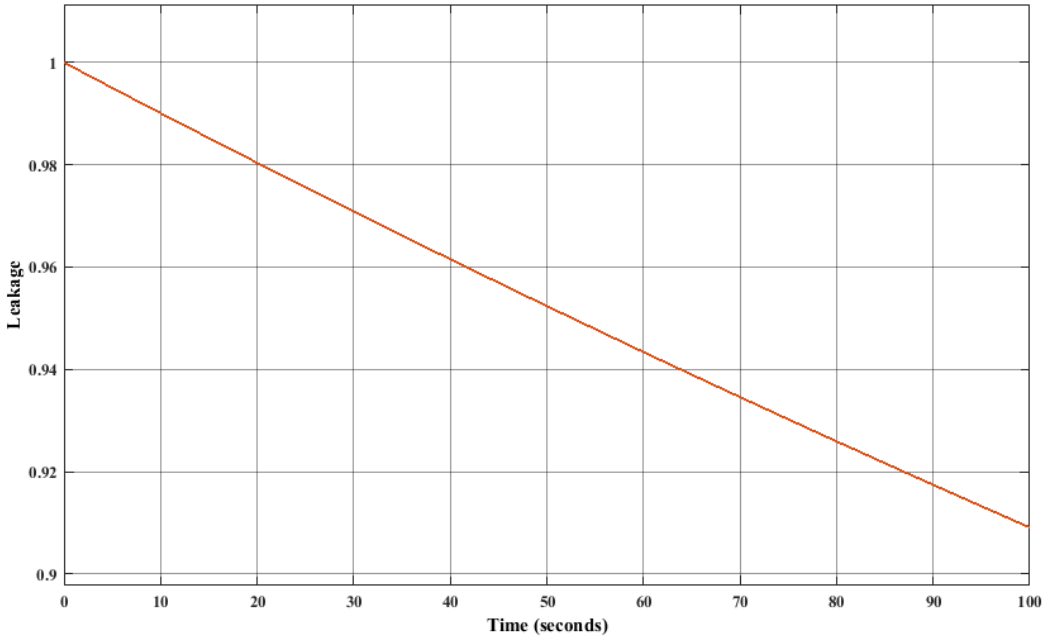


Fig. 5.79 Euphoria Filter - Graph of Residuals for 5^{th} Set

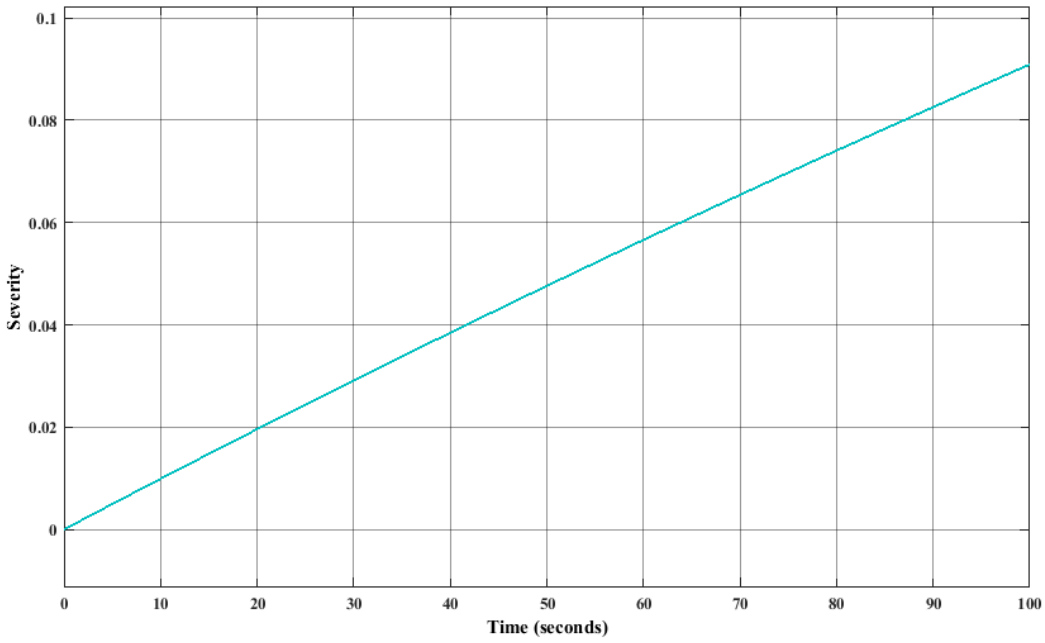


Fig. 5.80 Euphoria Filter – Thrusters' Angles for 5^{th} Set

vi. 20% parameter variation, NF=10, $d = d^* + 20\%$, $\alpha = 0.001$ (Thruster No 2).

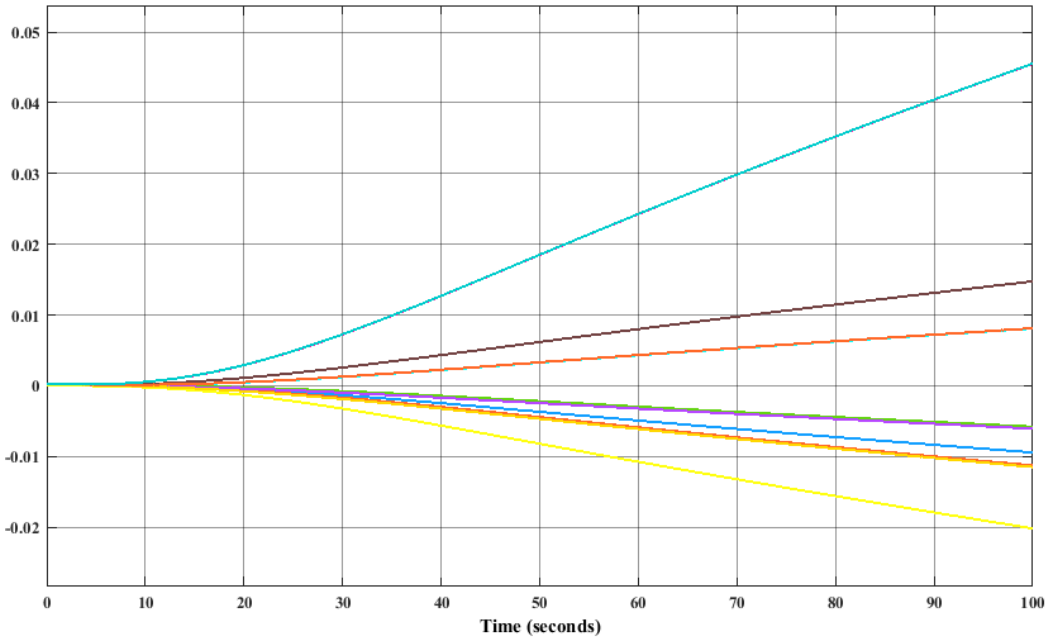


Fig. 5.81 Euphoria Filter - Graph of Leakage for 6^{th} Set

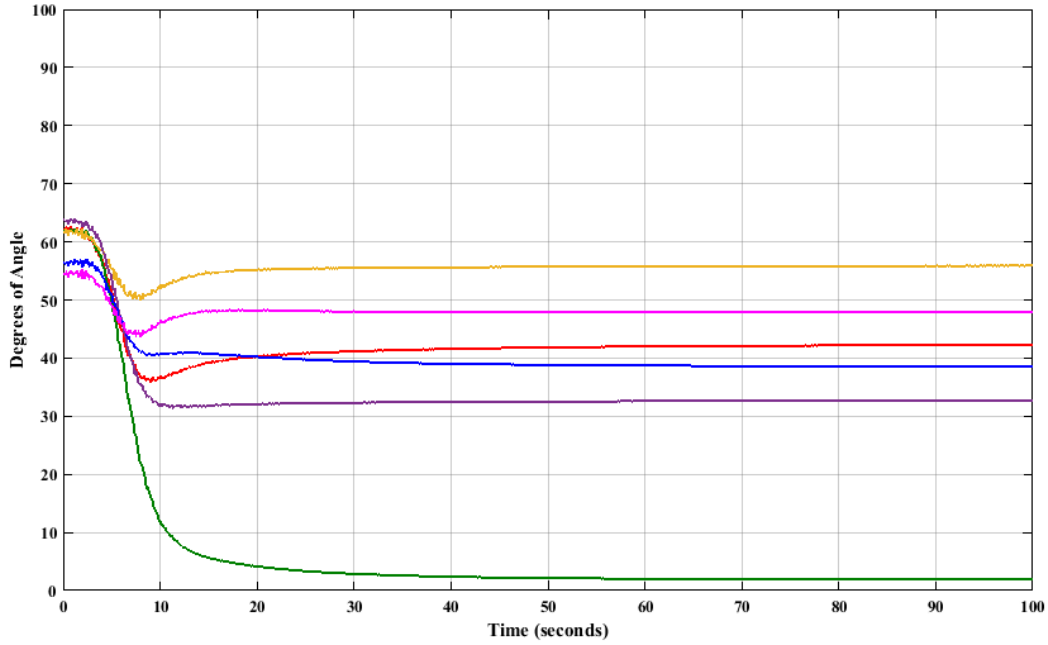


Fig. 5.82 Euphoria Filter - Graph of Severity for 6^{th} Set

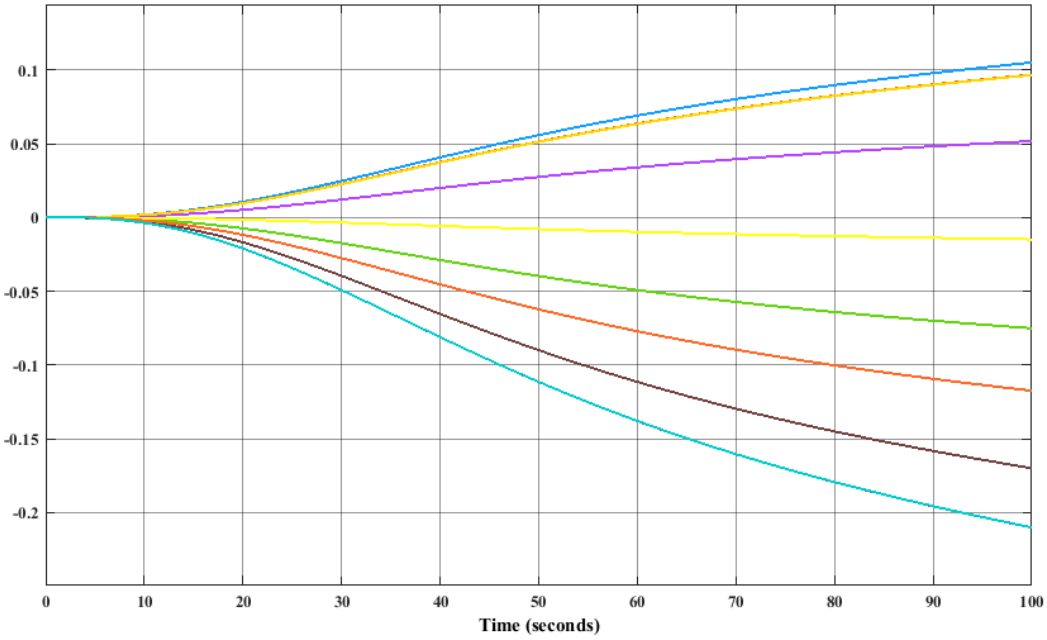


Fig. 5.83 Euphoria Filter - Graph of Residuals for 6^{th} Set

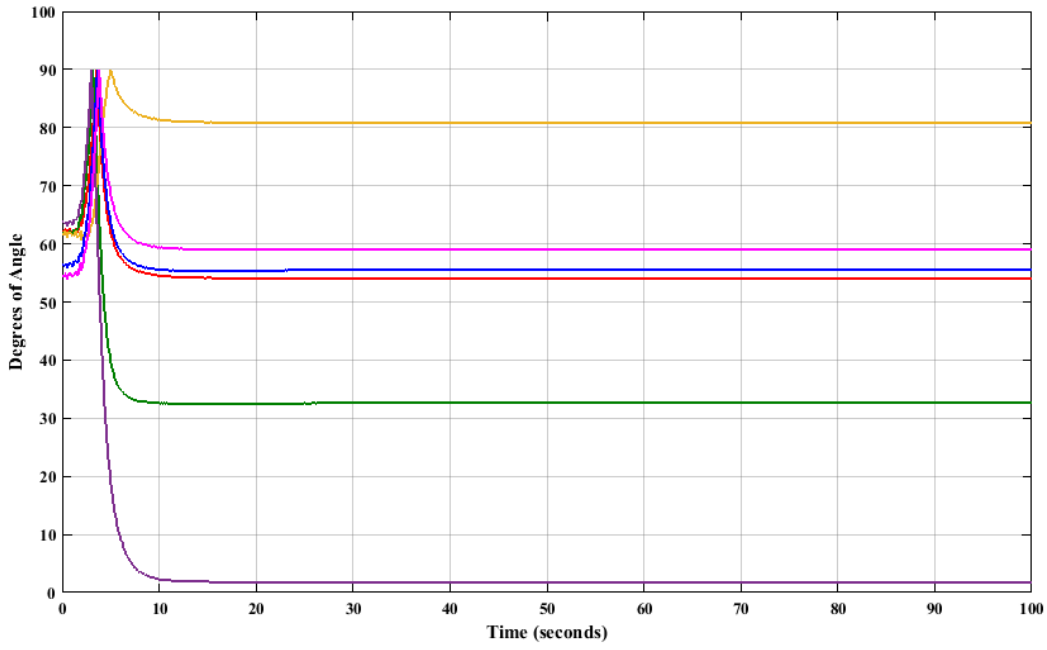


Fig. 5.84 Euphoria Filter – Thrusters' Angles for 6^{th} Set

vii. 20% parameter variation, NF=10, $d = d^* + 20\%$, $\alpha = 0.01$ (Thruster No 4).

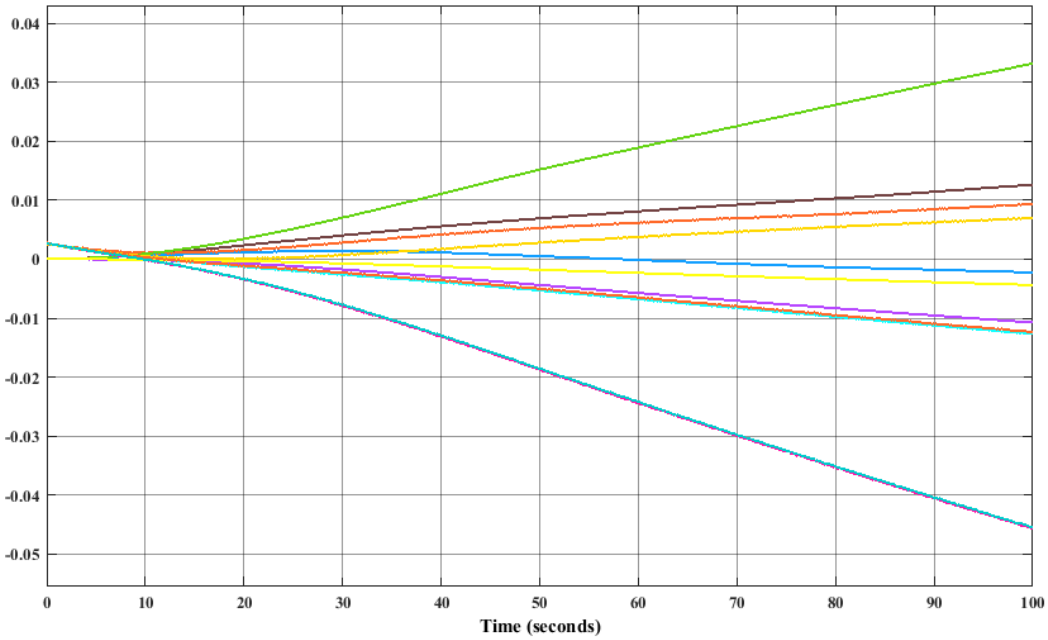


Fig. 5.85 Euphoria Filter - Graph of Leakage for 7^{th} Set

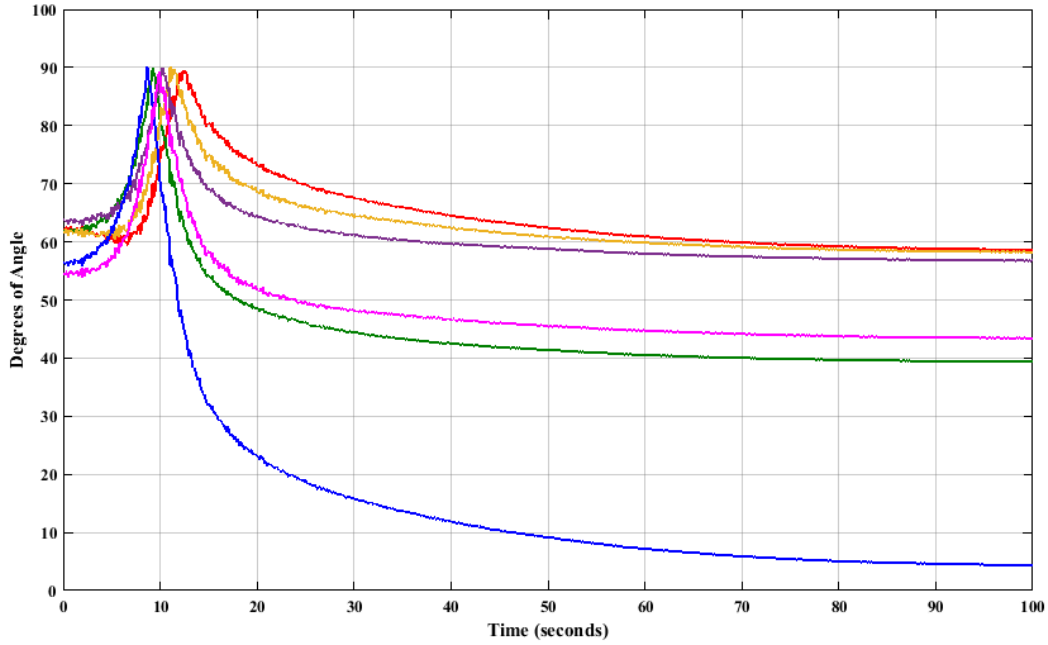


Fig. 5.86 Euphoria Filter - Graph of Severity for 7^{th} Set

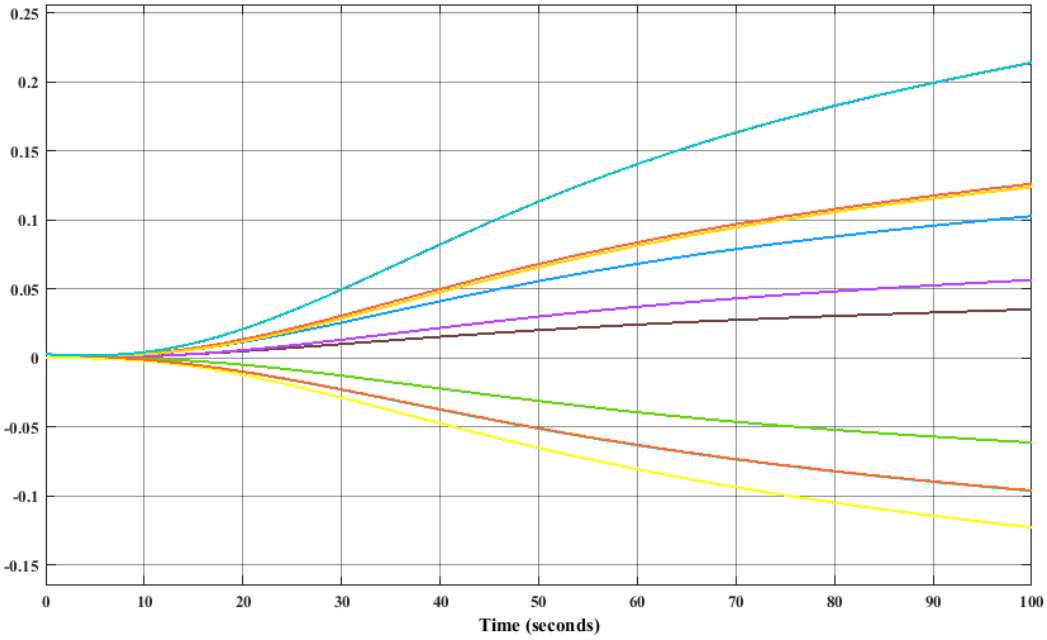


Fig. 5.87 Euphoria Filter - Graph of Residuals for 7^{th} Set

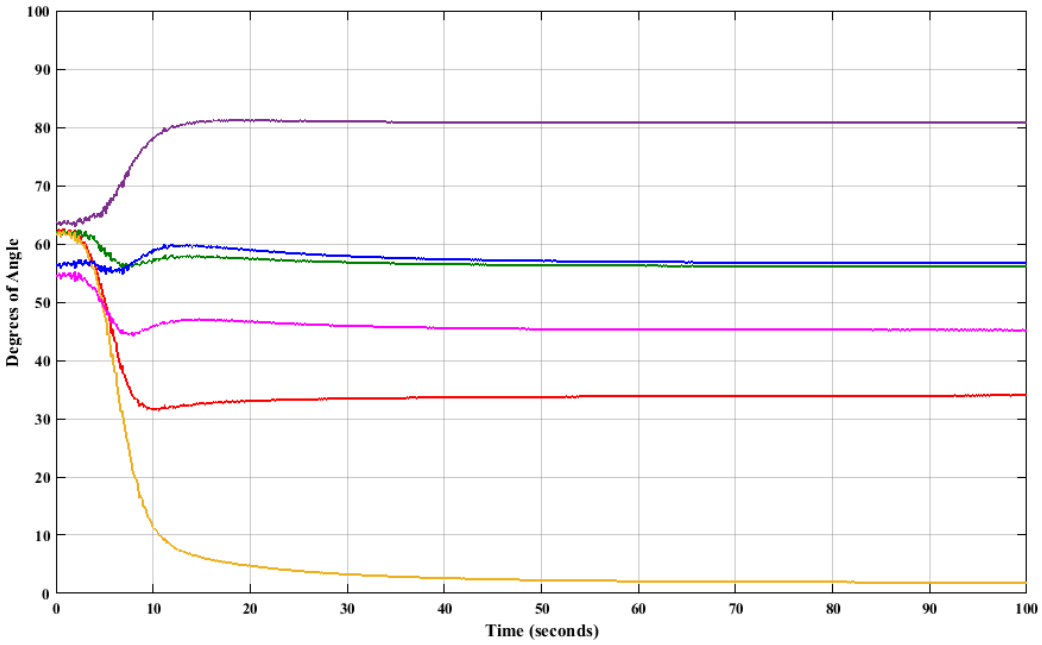


Fig. 5.88 Euphoria Filter – Thrusters' Angles for 7^{th} Set

viii. 20% parameter variation, NF=100, $d = d^* + 20\%$, $\alpha = 0.001$ (Thruster No 5).

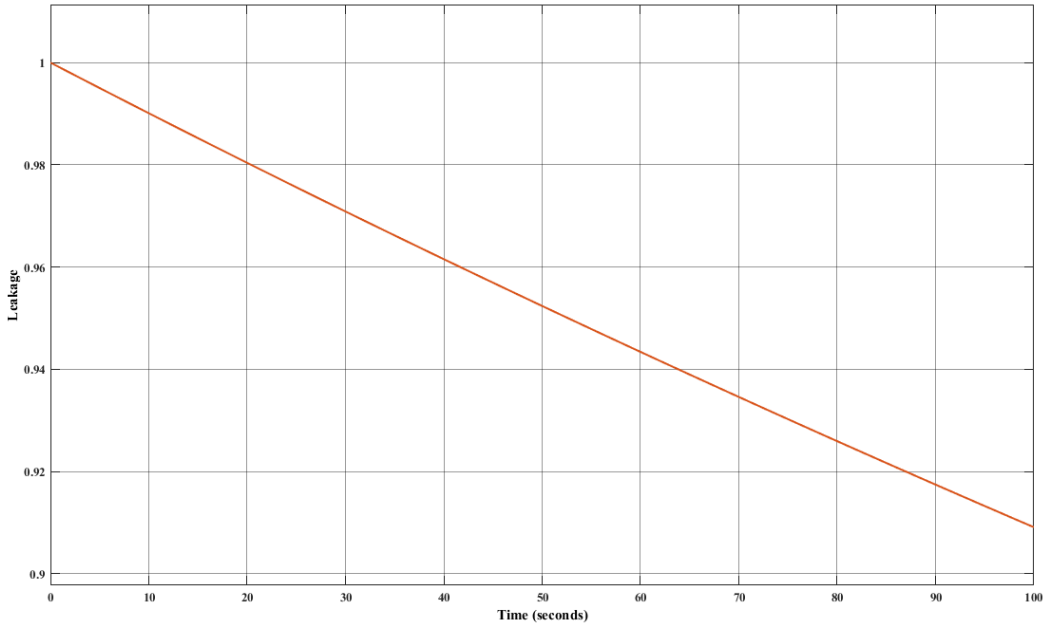


Fig. 5.89 Euphoria Filter - Graph of Leakage for 8^{th} Set

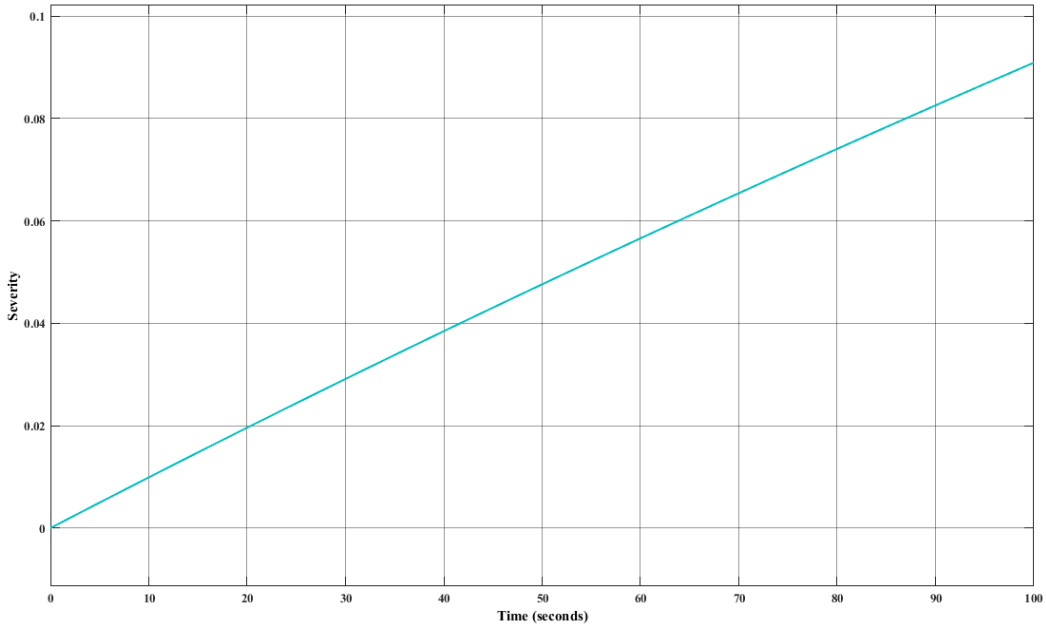


Fig. 5.90 Euphoria Filter - Graph of Severity for 8^{th} Set

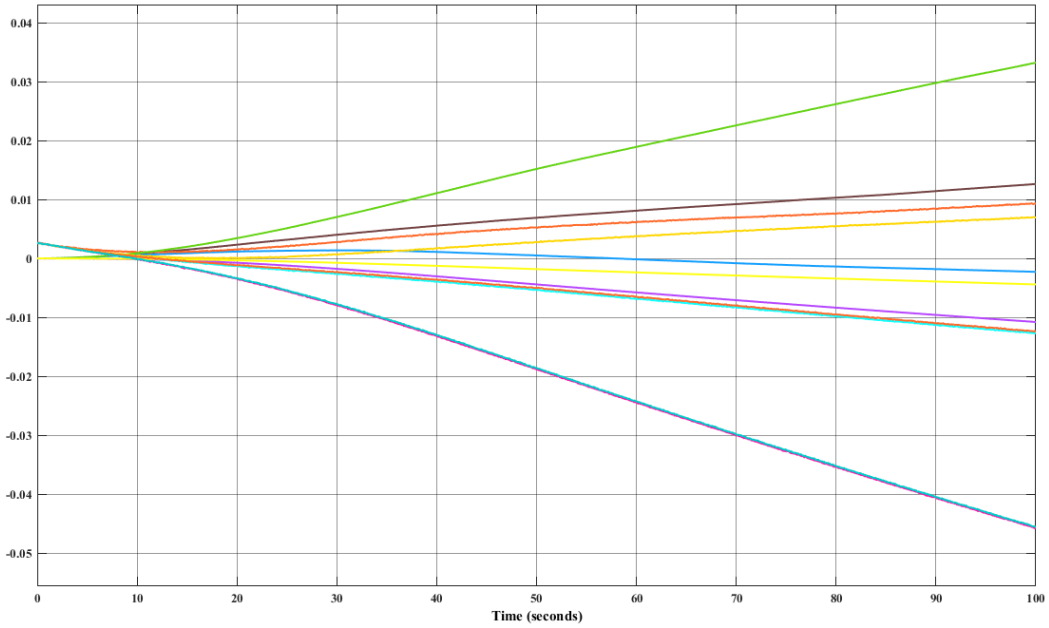


Fig. 5.91 Euphoria Filter - Graph of Residuals for 8^{th} Set

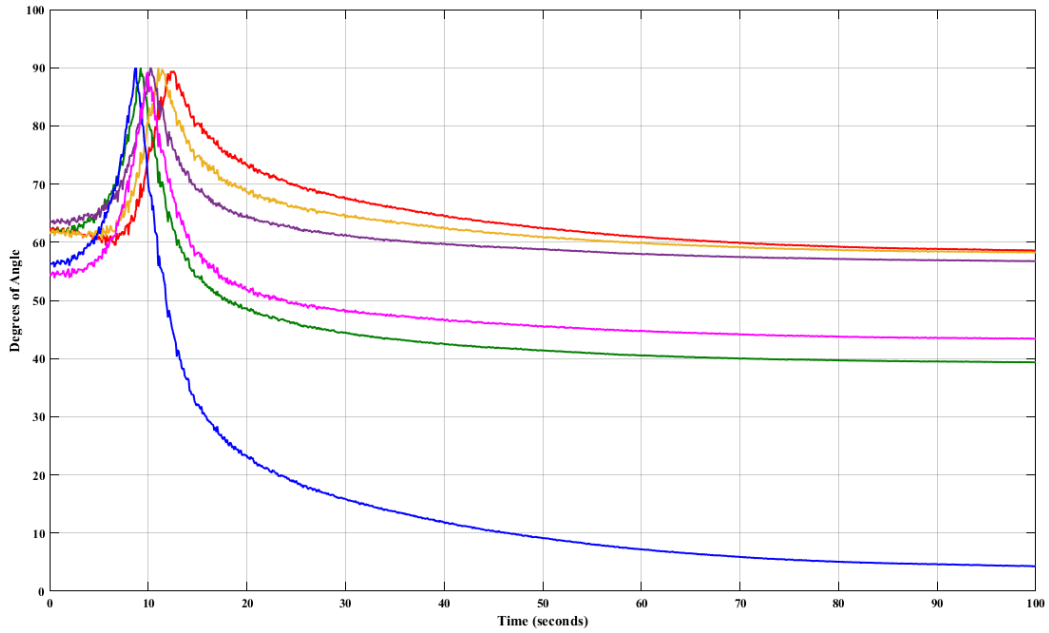


Fig. 5.92 Euphoria Filter – Thrusters' Angles for 8^{th} Set

ix. 20% parameter variation, NF=100, $d = d^* + 20\%$, $\alpha = 0.01$ (Thruster No 3).

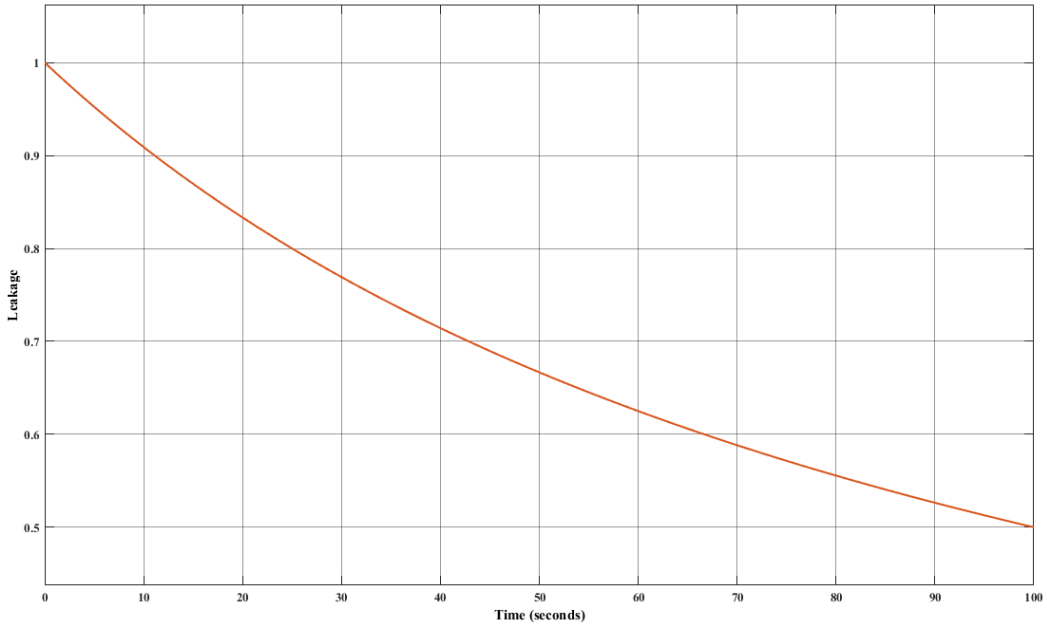


Fig. 5.93 Euphoria Filter - Graph of Leakage for 9^{th} Set

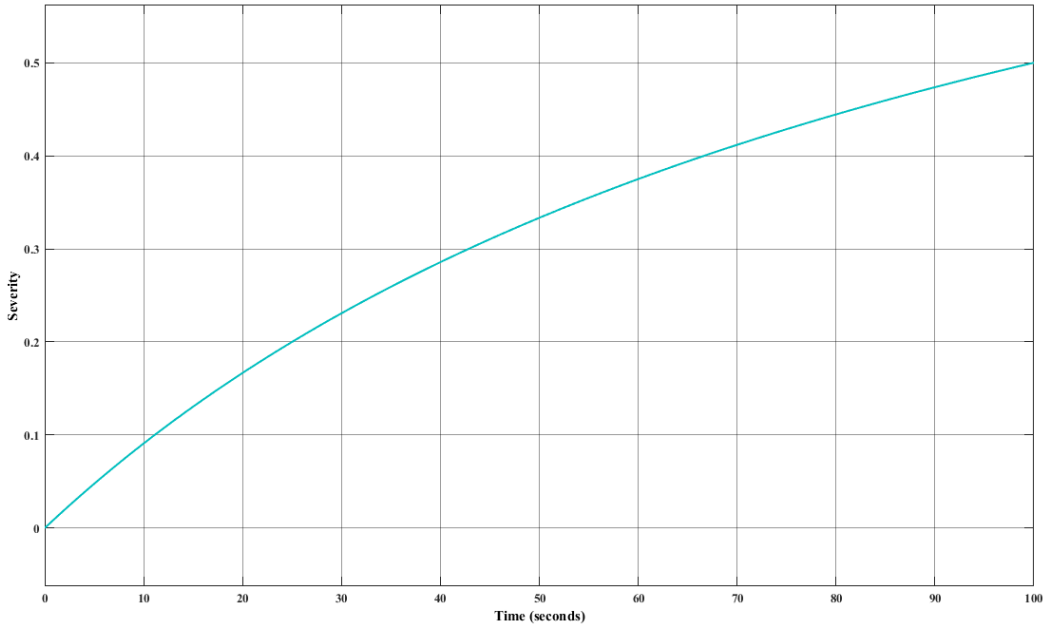


Fig. 5.94 Euphoria Filter - Graph of Severity for 9^{th} Set

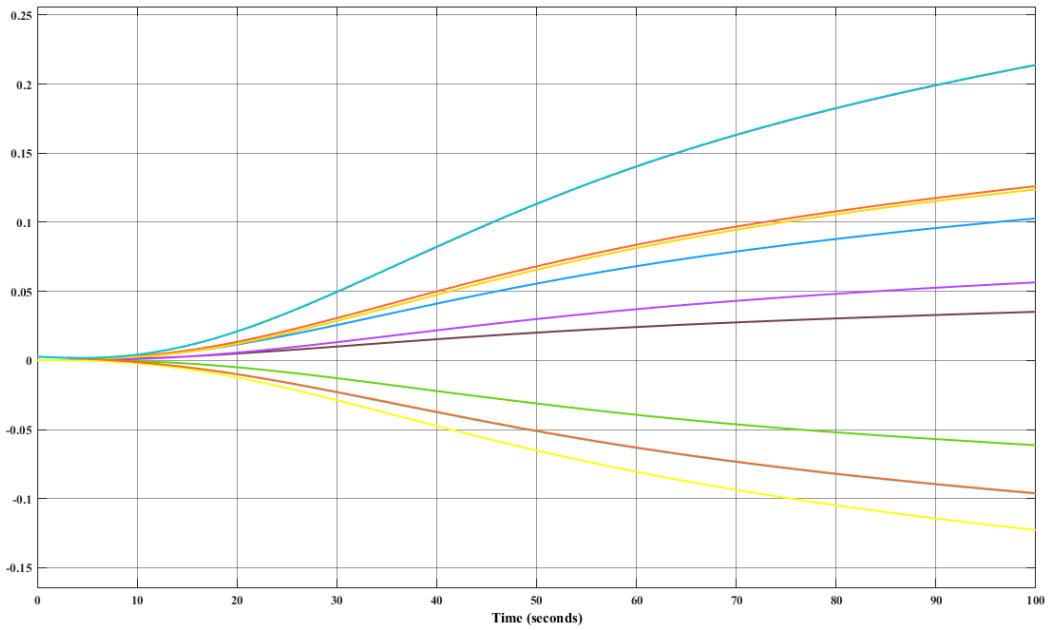


Fig. 5.95 Euphoria Filter - Graph of Residuals for 9^{th} Set

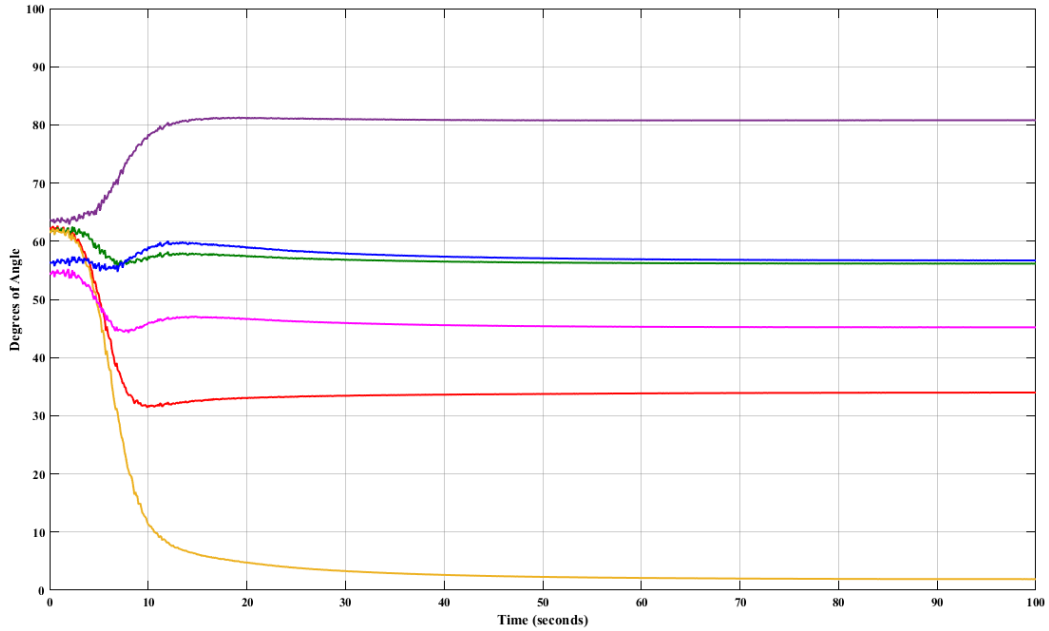


Fig. 5.96 Euphoria Filter – Thrusters' Angles for 9^{th} Set

Table 5.13 summarizes the findings from Monte Carlo analysis based on the detection t_d, for the pre-mentioned scenarios.

Table 5.13 Euphoria Filter – Leakage Cases - Time to reach Severity

Case Number	t_d	Time to reach Severity 0.01	Time to reach Severity 0.1
1	< 8 s	100 s	1000 s
2	< 6 s	10 s	100 s
3	< 3 s	0 s	10 s
4	< 25 s	10 s	100 s
5	< 8 s	0 s	10 s
6	< 8 s	10 s	100 s
7	< 4 s	0 s	10 s
8	< 28 s	10 s	100 s
9	< 6 s	0 s	10 s

Based on the above time profiles, it is evident that leakage is detected and identified very fast; in cases #2 and #8, for example, the failure was identified well before its severity had reached 0.01, with a NF=10. Even with a NF=100, it only took 18 seconds to identify the failure, after reaching a severity of 0.01.

Further Analysis for the Euphoria Filter, the KFDE and FDI Timing

The Euphoria Filter has, by construction, a significant advantage, in that it satisfies a frequency domain condition known as KFDE. This condition is a relationship between the plant open loop dynamics, from disturbances to output and the H_2 Filter open loop dynamics, from residuals input to output. Specifically:

$$KDFE: \quad [I + G_{KF}(s)]\,[I + G_{KF}(s)]^H = I + \frac{1}{\mu}\left[G_{FOL}(s)G_{FOL}(s)^H\right] \tag{5.1}$$

where:

$$G_{KF}(s) = C\,(sI - A)^{-1}\,G \tag{5.2}$$

with A, C the plant dynamics and output matrix respectively, and G the filter gain matrix-residuals input matrix:

$$G_{FOL}(s) = C\,(sI - A)^{-1}\,L \tag{5.3}$$

with A, C the plant dynamics and output matrix respectively, and L the disturbance input matrix to the plant and μ is the measurement noise intensity according to: $\Theta = \mu I$, with all sensor noises considered equal for the KFDE design option. In the limit, as $\mu \to 0$, where there is no virtually sensor noise,the KFDE yields:

$$G_{KF}(s) \sim G_{FOL}(s) \tag{5.4}$$

i.e. the filter loop characteristics are approximately equal to those of the open loop. This approximate relationship also holds for the singular values of the above transfer function matrices and, of course, the corresponding frequency crossovers as well.

The above result gives the designer some freedom to shape the filter crossover, and thus affect its convergence time, according to some pre-specified desired criterion, which is reflected in the judicious choice of matrix L, which serves as the design parameter. Thus, once the open loop desirable characteristics have been set, then by gradually reducing the sensor noise level μ, the filter assumes the pre-specified characteristics.

Furthermore, the filter noise and disturbance rejection capabilities are not significantly compromised. As seen in the preceding section of Monte Carlo runs, even this optimal design has considerable tolerance/robustness to an increase of noise factor up to 100. In addition, designating L as a parameter to be chosen by the designer is allowable, since, it cannot be argued that the way disturbances enter the system is accurately modelled or known; besides, the design has tolerance to a two orders of magnitude NF increase. In the specific context of the Euphoria Filter, the diagrams of Figure 5.97 and Figure 5.98 show the singular values of $G_{FOL}(s)$ and $G_{KF}(s)$, respectively.

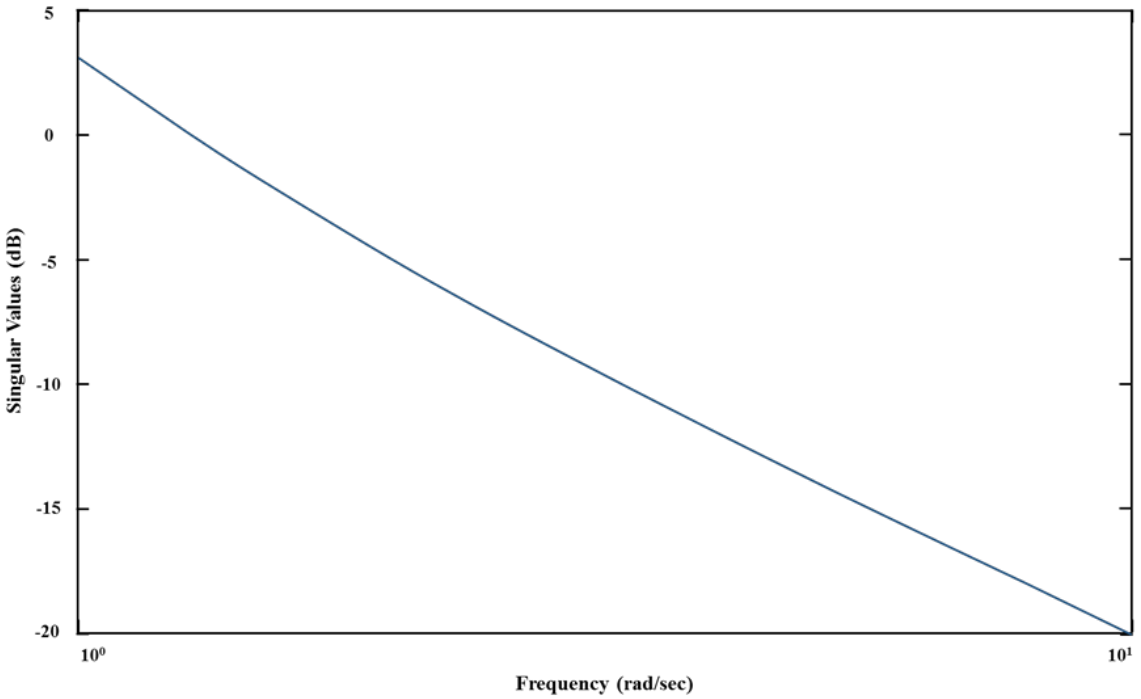


Fig. 5.97 Euphoria Filter - Singular Values of G_{FOL}

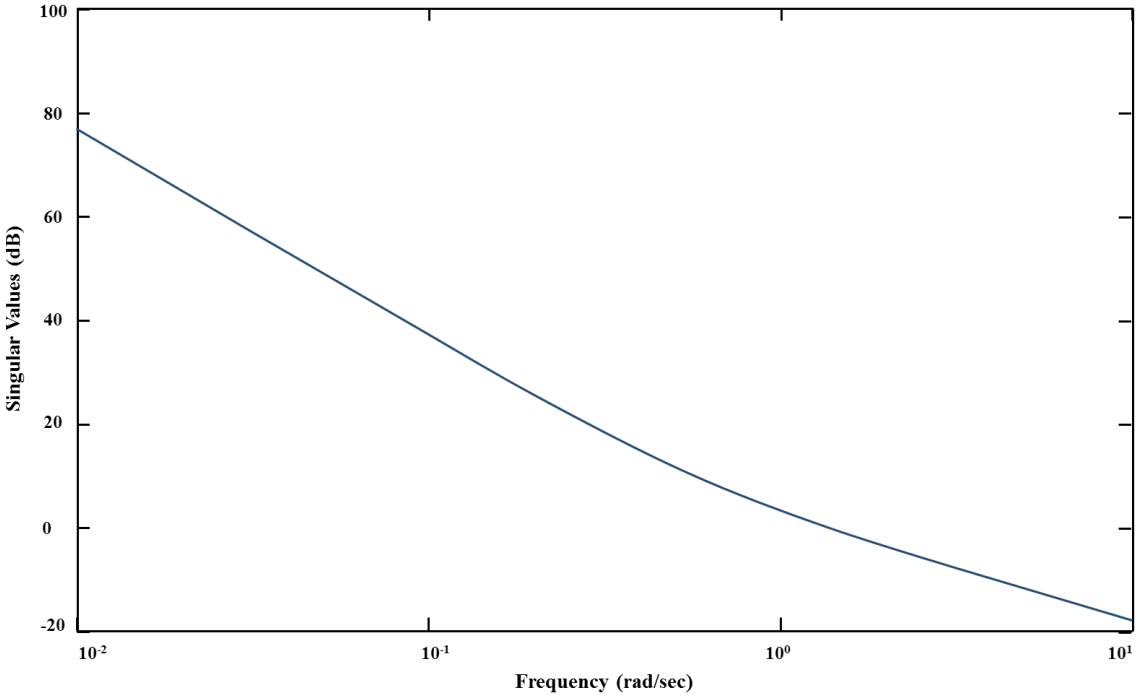


Fig. 5.98 Euphoria Filter - Singular Values of G_{KF}

with H_2 Filter gain matrix:

H2_matrix <22x11 double>

	1	2	3	4	5	6	7	8	9	10	11
1	0.1189	0	0	0	0	0	0	0	0	0	0
2	0.0071	0	0	0	0	0	0	0	0	0	0
3	0	0.1189	0	0	0	0	0	0	0	0	0
4	0	0.0071	0	0	0	0	0	0	0	0	0
5	0	0	0.1189	0	0	0	0	0	0	0	0
6	0	0	0.0071	0	0	0	0	0	0	0	0
7	0	0	0	0.1189	0	0	0	0	0	0	0
8	0	0	0	0.0071	0	0	0	0	0	0	0
9	0	0	0	0	0.1189	0	0	0	0	0	0
10	0	0	0	0	0.0071	0	0	0	0	0	0
11	0	0	0	0	0	0.1189	0	0	0	0	0
12	0	0	0	0	0	0.0071	0	0	0	0	0
13	0	0	0	0	0	0	0.1189	0	0	0	0
14	0	0	0	0	0	0	0.0071	0	0	0	0
15	0	0	0	0	0	0	0	0.1189	0	0	0
16	0	0	0	0	0	0	0	0.0071	0	0	0
17	0	0	0	0	0	0	0	0	0.1189	0	0
18	0	0	0	0	0	0	0	0	0.0071	0	0
19	0	0	0	0	0	0	0	0	0	0.1189	0
20	0	0	0	0	0	0	0	0	0	0.0071	0
21	0	0	0	0	0	0	0	0	0	0	0.1189
22	0	0	0	0	0	0	0	0	0	0	0.0071

Fig. 5.99 Euphoria Filter - H_2 matrix

The two singular value plots show that crossover occurs for both at approximately 1 rad / s, which translates to a time constant of 6 s. This was achieved by no decrease in the μ parameter as it was shown in the plots above. Since our FDI timing was excellent, significantly improving that of the Diagnosis Filter, there seems to exist no reason for a further reduction of μ. Making the FDI timing much faster than that might not allow the filter to properly process the measurement information, and could result in a compromised performance.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Many space exploration missions require a critical autonomous proximity operation. Mission safety is normally guaranteed through a hierarchical implementation of the fault diagnosis and fault tolerance with several levels of faults containments ranging from the local components up to the global system through various instruments (e.g. sensors like Inertial Measurement Unit (IMU), thrusters, etc.), redundancy paths and ground intervention. The main purpose of the FDI function is to maintain the plant operational or safe as required by the mission. The typical FDI hierarchical implementation approach, does not always sufficiently address dynamics deviations in critical aerospace operations. This is especially the case for thruster faults during rendezvous and docking/capture proximity operations, which could lead to mission loss. The robustness of an on-board FDI afforded by the three methods outlined in this dissertation could therefore be crucial for the safety of future missions, but it can also have implications extending in other domains such as Internet of Things (IoT), Cyber-Physical systems, and Bioengineering.

The three proposed techniques described here have been extensively tested with reference to all the thrusters under considerable parametric and noise variations, applying the ESA's LPF simulation model.

Comprehensive Monte Carlo tests have shown remarkable robustness and performance of the Diagnosis Filter. It can handle both fixed and leakage types of failures of minimal severity of 0.01, with remarkable accuracy and speed in terms of both fault detection and fault identification as well as the corresponding severity magnitude identification. Therefore, the Diagnosis Filter comprises a very promising design according to the defined FDI metrics for reliable mission performance.

The Euresis Filter also exhibited robustness to the same type of parametric uncertainty as the Diagnosis Filter, and with a noise factor of at least two orders of magnitude greater, albeit for total failures only. Further research is required into this method development in order to gain a better understanding of some of its puzzling outcomes regarding its total failure identification under extreme noise.

The Euphoria Filter (H_2 Based FDI method) proved to be an optimal state estimator while possessed some superior characteristics such as guaranteed convergence and control over time to FDI. Some of the most promising non-linear estimation techniques use the 2-norm in their development, which can have a significant impact on practically all the FDI scenarios, linear or non-linear. We have extensively studied the linear case for the LPF model. It will be very instructive and useful to investigate whether the process performance and robustness characteristics remain invariant for all types of plant dynamic structures, and to apply the method in the non-linear scenario.

The extensive tests performed substantiate the undeniable suitability of all the presented techniques for the LPF thruster failure FDI and, given the generic nature of their theoretical development, render them applicable to other spacecraft mission applications. Table 6.1, Table 6.2 and Table 6.3 summarize the design, evaluation, and application characteristics of the three proposed methods.

6.2 Future Work

Despite this work's direct applicability within the framework of LPF as established by the use of the ESA supplied simulation model, the theoretical basis of the presented algorithms can also be translated in other research fields. The reason for this is that a robust FDI is essential not only for spacecrafts' and UAVs' autonomy but also for any type of cyber-physical systems or machines (e.g. IoT, wearable exoskeletons and robots), incorporating controllers, sensors and actuators [70, 71].

Based on the findings presented in this thesis, the following issues should be further investigated for a complete FDI technique for small satellites and other spacecraft for autonomous missions:

- Develop a hybrid filter to simultaneously identify sensor and actuator failures. Although the detection of sensor failures is handled via hardware tests, accurate gyro isolation remains an open issue.

Table 6.1 Comparison of the Three FDI Techniques based on Design Characteristics

	Diagnosis Filter	**Euresis Filter**	**Euphoria Filter**
Design Approach	Deterministic	Stochastic	Optimal Stochastic
Design Synthesis Complexity	Simple and systematic design; SVD application results easily incorporated for robustness	Straightforward and well defined for thrusters faults. SVD application results easily incorporated for robustness	Very systematic design process
Directionality *vis a vis* Filter Design	Filter design based on directionality constraint *a priori*- structured residuals directly obtained	N/A	Filter designed as H_2 ; residuals directionality recovered *a posteriori*
Directionality as FDI diagnostic	Projection of filter residuals on transformed failure directions	N/A	Projection of direction recovered filter residuals on failure directions
Transparency for Easy Troubleshooting	Excellent due to its simplicity	Very good due to its simplicity	Very good
Implementation Reqs./Complexity	Very simple, low complexity	Simple set up but requires as many design models as the number of expected failures	Simple to set up. Requires as many design models as the number of expected failures
Algorithm dependency on subsystem/unit characteristics	Model based, hence dependent	Model based, hence dependent	Model based, hence dependent
Observability and Information Processing Requirements	Excellent for specific context with available measurements	Excellent for specific context with available measurements	Very good; depending on available measurements
Simultaneously State Estimator	No	Yes	Yes
Flexibility/Multiple Usage (i.e. in Control Loop)	No	Yes	Yes

Table 6.2 Comparison of the Three FDI Techniques based on Evaluation Characteristics

	Diagnosis Filter	**Euresis Filter**	**Euphoria Filter**
Time to FDI	Excellent	Excellent	Excellent; up to 4 times faster than Diagnosis Filter
Fault diagnosis criteria	Excellent due to enhanced robustness	Very good due to enhanced robustness	Very good
Real-time performance and software validation	Excellent, due to the simplest possible algorithm design; software validation straightforward	Very good, due to simplicity of architecture and associated FDI algorithm; Software validation eased	Very good. Software validation straightforward
Robustness to Parametric Variation	Excellent: up to ± 20%	No	Excellent: up to ± 20%
Robustness to Disturbances and Sensor Noise	Excellent: up to NF=10000	Excellent: up to NF=100000	Excellent: up to NF=1000
Performance with Reference to FDI Metrics	Excellent	Excellent on Total Failures Only	Excellent; better than Diagnosis Filter in timing
Guaranteed Stability	No	Yes	Yes
Verification and Validation	Requires Monte Carlo	Requires Monte Carlo	Simplified due to design assumptions

Table 6.3 Comparison of the Three FDI Techniques based on Application Characteristics

	Diagnosis Filter	**Euresis Filter**	**Euphoria Filter**
Compliance with LPF system engineering requirements	Perfectly matched	Very compatible	Very compatible
Applicability to LPF Thrusters FDI	Excellent by specific design	Excellent by specific design	Very good
Multiple Faults FDI	Excellent, due to enhanced robustness	Very good due to enhanced robustness; requires one model for each failure	Suitable, but requiring more than one design models, one for each failure
Suitable for Sensor/Actuator Failure Detection	Very suitable and effective with the SVD based modification	Very suitable for actuator faults; less effective for sensor faults	Very suitable in a multiple model setting
Suitability for other space applications	Generic and model based. Applicable where dynamic models exist	Model based; applicable where dynamic models exist	Model based; applicable where dynamic models exist
Admits to Nonlinear Model	No	No	Yes

- Explore post FDI reconfiguration issues within the present controller framework for the LPF.
- Take steps toward an overall multivariable controller design and the corresponding FDI development for the LPF.

Conceptual Modelling of FDI Techniques

Modelling plays an important role in managing the complexity of modern systems that consist of various types of actuators, sensors and transducers. The next challenge is the adaptation of FDI techniques in human's daily life. Examples which signify the importance of FDI techniques in our daily life activities are smartphones, smart band, smart IoT devices, etc. or even implantable medical devices [72, 73], where the early and accurate identification of a fault is of paramount importance for the survival of a human being. Another example is the various Cyber-Physical Systems (CPS) such as autonomous vehicle systems, robotics, automatic pilot avionics, smart grids, etc., where a fault in a sensor is able to cause tragic accidents [74]. Based on the previous examples, it is clear that model-based techniques may advance the operations of smart systems. The next challenge is how the complexity of these systems could be reduced. This could be achieved with conceptual modelling techniques. Models not only facilitate coping with the increasing complexity of CPS and IoT-enabled technologies by providing structuring, analysis and further processing qualities for human beings (i.e., conceptual modelling [75] but they can also be used for the development and management of software systems or CPS. Modelling methods guide the creation of valid models. A modelling method referring to the definition by Karagiannis and Kühn [3] is composed of three building blocks (Figure 6.1):

- A modelling language defines the elements of the modelling method and the rules for combining them (syntax). A comprehensive specification of a modelling language is built up of three parts: notation, syntax, and semantics. For every element of the modelling language's syntax, the semantics defining the meaning and the notation defining the visual representation need to be specified.
- The modelling procedure then uses the modelling language and defines the steps a modeler should perform while applying the modelling language in order to create valid models. The combination of modelling language and

modelling procedure is referred to as the modelling technique of a modelling method.

- Mechanisms and algorithms provide the functionality used by the modelling procedure to process the models, e.g. simulation algorithms, model transformation algorithms, model queries, and model validation.

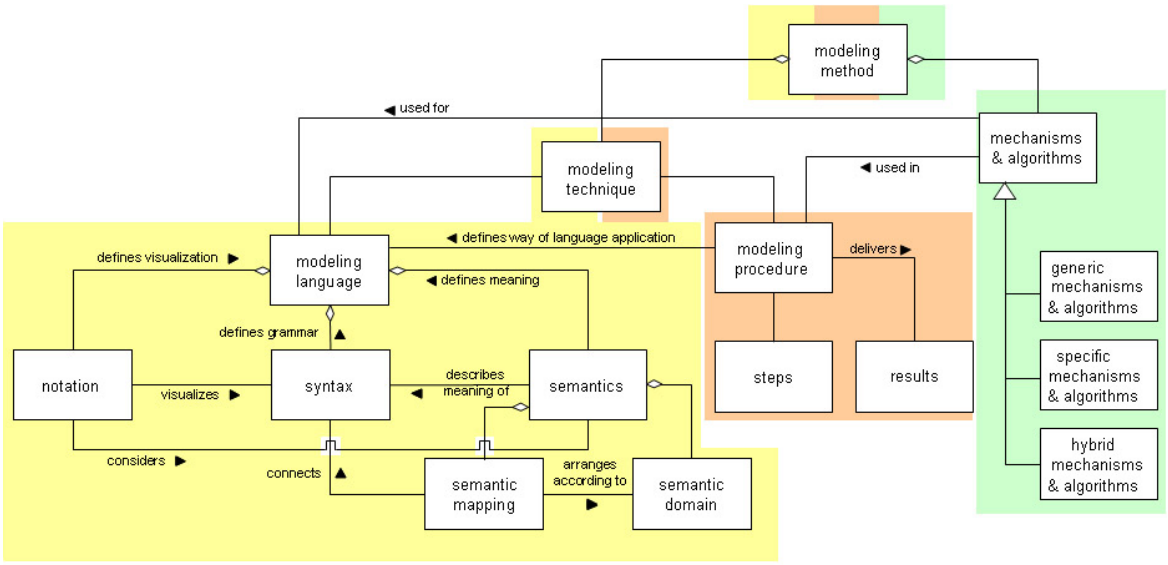


Fig. 6.1 Components of Modelling Methods (Karagiannis & Kühn [3])

The third building block could initiate the modelling of FDI mechanisms in a less complex method but with the same strong mathematical background. We can examine different ways to define semantically the information of the sensors, actuators, and controllers of the system and how the noise that influences the sensors, actuators and the real system from the useful data can be separated semantically. Another issue that could be answered through the creation of a new modelling method is how the systems' failures can be modelled with time dimension and FDI techniques in discrete and continuous systems. A successful modelling method which will describe dynamic systems with FDI mechanisms has to take into account the implication of time in the model and how time influences the stability and the interoperability of the system.

Another aspect that has to be realised during the creation of the modelling language is the different constituents (a dynamic model consists of different sub-models that participate actively in; such as sensors, controllers, actuators etc.). In parallel,

a modelling method has to serve multiple purposes depending on the various designers' and users' perspectives [76, 77]. Also, the importance of the formalization aspect which will enable intersubjective understanding and interoperability machine processing [78] has to be considered. One of the ulterior targets of this modelling method through the FDI mechanisms would be the prediction of faults, and the identification and the isolation of the system's incongruent behaviour.

A real example that encloses the pre-mentioned challenges is the Smart City [79]. A Smart City integrates information from different assets such as transportation systems, hospitals, power plants, water supply networks, waste management, etc. The IoT devices and Cyber-Physical systems have a significant role for the seamless and efficient operation of various Smart City services. The operational complexity of a Smart City could raise questions such as what happens if a component fails and how can we manage the disturbance noise more efficiently or what actions have to be taken if a fault occurs in an actuator of a smart water supply system. An example of how sensors could influence the services of a Smart City can be found in [80].

In the conclusion, it would be novel but at the same time challenging (according to all of the above mentioned reasons) to create a modelling language which will be oriented to model dynamical linear or non-linear systems in a more abstract and more comprehensive way.

List of Publications

[1]. N. Tantouris, D. Polemi and K. Dellios, "Euresis-Filter: A robust multiple-model FDI technique for small satellites thruster faults," forthcoming in Int. J. Model. Identif. Control, 2017.

[2]. N. Tantouris, D. Polemi and K. Dellios, "Diagnosis-Filter: A Guaranteed Robust FDI Technique for Lisa-Pathfinder Thruster Failures," Control Engineering Practice, A Journal of IFAC, International Federation of Automatic Control (under review) 2016.

[3]. N. Tantouris, D. Polemi and K. Dellios, "On the path to Cyber-Peace: A Robust 'diagnosis-filter' for safeguarding future systems and machines," presented at the 2^{nd} International Conference on Internet of Things, Data and Cloud Computing, Cambridge, United Kingdom, 22-23 March, 2017.

[4]. N. Tantouris and K. Dellios, "Euphoria-filter: a robust H_2/H_∞ technique amenable to the Lisa-Pathfinder thruster faults," Int. J. Sp. Sci. Eng., vol. 4, no. 1, pp. 45–63, 2016.

[5]. L. Valavani and N. Tantouris, "Guaranteed robust fault detection and isolation techniques for small satellites," in Progress in Flight Dynamics, Guidance, Navigation, Control, Fault Detection, and Avionics, C. Vallet, D. Choukroun, C. Philippe, G. Balas, A. Nebylov, and O. Yanova, Eds., vol. 6. Les Ulis, France: EDP Sciences, December 2013, pp. 407–422.

[6]. L. Valavani and N. Tantouris, "Model Based FDI for the LPF," Poster for the GNC 2011, 8^{th} International ESA Conference on Guidance and Navigation Control Systems, Carlsbad, Czech Republic, 5-10 June, 2011.

[7]. D. Bork, HG. Fill, D. Karagiannis, ET. Miron, N. Tantouris and M. Walch, "Conceptual Modelling for Smart Cities: A Teaching Case". Interaction Design & Architecture(s) Journal (IxD&A), Special Issue on Smart City Learning: Opportunities and Challenges, No. 27, pp. 10-27, 2015.

[8]. D. Bork, R. Buchmann, I. Hawryszkiewycz, D. Karagiannis, N. Tantouris and M. Walch, "Using Conceptual Modeling to Support Innovation Challenges in Smart Cities," 2016 IEEE 18^{th} International Conference on High Performance Computing and Communications; IEEE 14^{th} International Conference

on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Sydney, NSW, 2016, pp. 1317-1324.

References

[1] ESA. (2016) A perfectly still laboratory in space. [Online]. Available: http://sci.esa.int/lisa-pathfinder/57559-a-perfectly-still-laboratory-in-space/

[2] W. Fichter, A. Schleicher, and N. Brandt, "Control Tasks and Functional Architecture of the LISA Pathfinder Drag-Free System," in *Guidance, Navigation and Control Systems*, vol. 606, 2006.

[3] D. Karagiannis and H. Kühn, "Metamodelling Platforms," *Proceedings of the Third International Conference EC-Web*, vol. 2455, p. 182, 2002.

[4] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques-part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3757–3767, June 2015.

[5] P. Frank, S. Ding, and T. Marcu, "Model-based fault diagnosis in technical processes," *Transactions of the Institute of Measurement and Control*, vol. 22, no. 1, pp. 57–101, March 2000.

[6] A. Falcoz, F. Boquet, M. Dinh, B. Polle, G. G. Flandin, and E. Bornschlegl, "Robust fault diagnosis strategies for spacecraft application to LISA Pathfinder experiment," in *IFAC Proceedings Volumes*, vol. 43, no. 15, 2010, pp. 404–409.

[7] J. Borde, F. Teston, S. Santandrea, and S. Boulade, "Feasibility of the PROBA 3 formation flying demonstration mission as a pair of microsats in GTO," in *Small Satellites, Systems and Services*, vol. 571, 2004, pp. 103–114.

[8] R. Chen, J. Speyer, and L. Guntur, "Periodic Fault Detection Filters with Application to Satellite Systems," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, p. 5734, 2003.

[9] S. Nolet, E. Kong, and D. W. Miller, "Design of an algorithm for autonomous docking with a freely tumbling target," *Modeling, Simulation, and Verification of Space-based Systems II*, vol. 5799, pp. 123–134, 2005.

[10] A. M. Salkham, "Fault Detection, Isolation and Recovery (FDIR) in On-Board Software," Doctoral dissertation, Chalmers University of Technology, 2005.

[11] E. Wilson, D. Sutter, and R. Mah, "Motion-Based Thruster Fault Detection and Isolation," in *Infotech@Aerospace*, 2005, p. 7182.

[12] F. H. Bauer, K. Hartman, J. P. How, J. Bristow, D. Weidow, and F. D. Busse, "Enabling Spacecraft Formation Flying through Spaceborne GPS and Enhanced Automation Technologies," in *ION-GPS Conference*, vol. 1, 1999, pp. 369–384.

[13] F. D. Busse, "Precise Formation-state Estimation in Low Earth Orbit Using Carrier Differential GPS," Doctoral dissertation, Stanford University, 2003.

[14] C.-W. Park, "Precise Relative Navigation Using Using Augmented CDGPS," Doctoral dissertation, Stanford University, 2001.

[15] R. J. Patton, F. J. Uppal, S. Simani, and B. Polle, "Robust FDI applied to thruster faults of a satellite system," *Control Engineering Practice*, vol. 18, no. 9, pp. 1093–1109, 2010.

[16] V. Hruby, M. Gamero-Castano, D. Spence, C. Gasdaska, N. Demmons, R. McCormick, P. Falkos, J. Young, and W. Connolly, "Colloid thrusters for the new millennium, ST7 DRS mission," in *Aerospace Conference*, vol. 1. IEEE, 2004, pp. 202–213.

[17] N. Tudoroiu, E. Sobhani-Tehrani, and K. Khorasani, "Interactive bank of unscented Kalman filters for fault detection and isolation in reaction wheel actuators of satellite attitude control system," in *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*. Παρισ: IEEE, 2006, pp. 244–269.

[18] N. Tudoroiu and K. Khorasani, "Fault detection and diagnosis for satellite's attitude control system (ACS) using an interactive multiple model (IMM) approach," in *Proceedings of 2005 IEEE Conference on Control Applications, 2005. CCA 2005.* Toronto: IEEE, 2005, pp. 1287–1292.

[19] S. Jayaram and Sanjay, "Fault tolerant autonomous rendezvous and docking architecture for spacecraft in presence of control actuator failures," *International Journal of Intelligent Unmanned Systems*, vol. 1, no. 1, pp. 5–20, February 2013.

[20] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis part I: Quantitative model-based methods," *Computers and Chemical Engineering*, vol. 27, no. 3, pp. 293–311, 2003.

[21] V. Venkatasubramanian, R. Rengaswamy, S. N. Ka, S. N. Kavuri, and S. N. Ka, "A review of process fault detection and diagnosis Part II : Qualitative models and search strategies," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 313–326, 2003.

[22] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, "A review of process fault detection and diagnosis part III: Process history based methods," *Computers and Chemical Engineering*, vol. 27, no. 3, pp. 327–346, 2003.

[23] L. Valavani and N. Tantouris, "Guaranteed robust fault detection and isolation techniques for small satellites," in *Progress in Flight Dynamics, Guidance, Navigation, Control, Fault Detection, and Avionics*, C. Vallet, D. Choukroun, C. Philippe, G. Balas, A. Nebylov, and O. Yanova, Eds., vol. 6. Les Ulis, France: EDP Sciences, December 2013, pp. 407–422.

[24] Z. Gao, "Fault Estimation and Fault Tolerant Control for Discrete-Time Dynamic Systems," *Industrial Electronics, IEEE Transactions on*, vol. 62, no. 6, pp. 3874–3884, 2015.

[25] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 3, pp. 636–653, May 2010.

[26] L. Valavani, "Model Based FDI Techniques: Literature Review, Most Promising Concepts, Architecture and Algorithms," EUropean Space Agency, Athens, Tech. Rep., July 2009.

[27] D. Fertin, "Feasibility analysis of DFACS with US colloidal thrusters," S2-ESTTN, Tech. Rep., 2007.

[28] L. Valavani and P. Voulgaris, "High performance linear-quadratic and H_∞ designs for a'supermaneuverable'aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 1, pp. 157–165, 1991.

[29] M. Dinh and B. Polle, "Application of Method II to LPF Thruster Failure Detection - GNC T.TCN.712587.ASTR. Iss. 2." European Space Agency (E.S.A), Tech. Rep., June 2007.

[30] L. Valavani, "Adaptive control," Boston, 1994.

[31] L. Valavani and N. Tantouris, "Candidate Model Based FDI Concepts: Design Synthesis Procedures-Step Wise Design, Analysis and Validation Approach - HSS TN3-2," European Space Agency (E.S.A), Tech. Rep., June 2010.

[32] L. Valavani, "Candidate Model Based FDI Concepts: Detailed Design, Algorithm Specifications and Tuning Methodology - HSS TNP2-2," European Space Agency (E.S.A), Tech. Rep., August 2010.

[33] J. Deckert and J. Deyst, "Reliable dual-redundant sensor failure detection and identification for the NASA F-8 DFBW aircraft," Draper (Charles Stark) Lab., Inc.; Cambridge, MA, United States, Tech. Rep., 1978.

[34] M. N. Desai, J. C. Deckert, and J. J. Deyst Jr, "Dual-Sensor Failure Identification Using Analytic Redundancy," *Journal of Guidance, Control, and Dynamics*, vol. 2, no. 3, pp. 213–220, 1979.

[35] E. Y. Chow and A. S. Willsky, "Analytical Redundancy and the Design of Robust Failure Detection Systems," *Ieee Transactions on Automatic Control*, vol. 29, no. 7, pp. 603–614, 1984.

[36] J. S. Meserole, "Detection Filters for Fault-tolerant Control of Turbofan Engines," Doctoral dissertation, Massachusetts Institute of Technology (M.I.T), 1981.

[37] M. A. Massoumnia, "A geometric approach to failure detection and identification in linear systems," Massachusetts Institute of Technology (M.I.T), Tech. Rep., 1986.

[38] M. Basseville, "Detecting Changes in Signals and Systems - A Survey," *Automatica*, vol. 24, no. 3, pp. 309–326, 1988.

[39] R. J. Patton, S. W. Willcox, and J. Winter, "Parameter-insensitive technique for aircraft sensor fault analysis," *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 4, pp. 359–367, 1987.

[40] R. J. Patton, P. M. Frank, and R. N. Clarke, *Fault diagnosis in dynamic systems: theory and applications*. Prentice-Hall, Inc., 1989.

[41] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.

[42] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, 1993.

[43] R. J. Patton and J. Chen, "Review of parity space approaches to fault diagnosis for aerospace systems," *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 2, pp. 278–285, 1994.

[44] J. Chen, R. J. Patton, and H.-Y. Zhang, "Design of unknown input observers and robust fault detection filters," *International Journal of Control*, vol. 63, no. 1, pp. 85–105, 1996.

[45] P. P. Frank and X. Ding, "Survey of robust residual generation and evaluation methods in observer-based fault detection systems," pp. 403–424, 1997.

[46] R. Isermann and P. Ballé, "Trends in the application of model based fault detection and diagnosis of technical processes," *Control Engineering Practice*, vol. 5, no. 5, pp. 709–719, 1997.

[47] M. Nyberg and L. Nielsen, "Parity Functions as Universal Residual Glenerators and Tool for Fault Detectability Analysis," in *Decision and Control, 1997., Proceedings of the 36th IEEE Conference*, vol. 5, no. December. San Diego, CA: IEEE, 1997, pp. 4483–4489.

[48] Space Systems Finland Ltd., "Aurora FDIR - Final Report, ETP-SSF-RP-002," European Space Agency (E.S.A), Finland, Tech. Rep., 2002.

[49] D. G. Luenberger and Y. Yinyu, *Linear and Nonlinear Programming*. Springer, 2008.

[50] Dinh M., B. Polle, J. Morand, and G. Griseri, "LPF FDI Benchmark Case Definition - TN4 Robust estimation for failure detection," EADS Astrium, Tech. Rep., 2006.

[51] F. Uppal, S. Simani, and R. Patton, "Selection of Robust Estimation Methods for Failure Detection," European Space Agency (E.S.A), Tech. Rep., 2004.

[52] EADS Astrium, "Robust Estimation for Failure Detection - ESA contract 17945 CCNI," European Space Agency (E.S.A), Tech. Rep., 2007.

[53] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition*. International Society for Optics and Photonics, 1997, pp. 182–193.

[54] J. Uhlmann, S. Julier, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on automatic control*, vol. 45, no. 3, pp. 477–482, 2000.

[55] S. Julier and J. Uhlmann, "Unscented Filtering and Non Linear Estimation," in *Proceedings of the IEEE*, vol. 92, no. 3. IEEE, 2004, pp. 401–422.

[56] D. G. Valavani, L., Anastassiou, G., "An Improved Nonlinear Filter for Robust Distributed Estimation," ESA/ESTEC, Tech. Rep., 2007.

[57] L. Perea, J. How, L. Breger, and P. Elosegui, "Nonlinearity in Sensor Fusion: Divergence Issues in EKF, modified truncated SOF, and UKF," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, 2007, p. 6514.

[58] L. Cork and R. Walker, "Sensor Fault Detection for UAVs using a Nonlinear Dynamic Model and the IMM-UKF Algorithm," in *Information, Decision and Control*. Adelaide, Qld: IEEE, 2007, pp. 230–235.

[59] J. Ru and X. R. Li, "Variable-Structure Multiple-Model Approach to Fault Detection, Identification, and Estimation," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 5, pp. 1029–1038, 2008.

[60] X. R. Li, V. P. Jilkov, J. Ru, and A. Bashi, "Expected-mode augmentation algorithms for variable-structure multiple-model estimation," in *IFAC Proceedings Volumes*, vol. 35, no. 1, 2002, pp. 175–180.

[61] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, 1989.

[62] P. P. Khargonekar and K. M. Nagpal, "Filtering and smoothing in an H_∞ setting," in *IEEE Transactions on Automatic Control*, vol. 36, no. 2. IEEE, 1991, pp. 152–166.

[63] B. D. Appleby, "Robust Estimator Design using the H_∞ Norm and μ Synthesis," Doctoral dissertation, Massachusetts Institute of Technology (M.I.T), 1991.

[64] B. D. Appleby, J. R. Dowdle, and W. V. Velde, "Robust estimator design using μ synthesis," in *Proceedings of the 30th IEEE Conference on Decision and Control*. Brighton: IEEE, 1991, pp. 640–645.

[65] L. Xie, C. E. De Souza, and M. Fu, "H_∞ estimation for discrete-time linear uncertain systems," *International Journal of Robust and Nonlinear Control*, vol. 1, no. 2, pp. 111–123, 1991.

[66] R. Mangoubi, B. D. Appleby, and J. R. Farrell, "Robust estimation in fault detection," in *Proceedings of the 31st IEEE Conference on Decision and Control*. ucson, AZ: IEEE, 1992, pp. 2317–2322.

[67] P. Voulgaris, "On optimal ℓ^∞ to ℓ^∞ filtering," *Automatica*, vol. 31, no. 3, pp. 489–495, 1995.

[68] S. Grenaille, D. Henry, and A. Zolghadri, "Fault diagnosis in satellites using H_∞ estimators," in *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, vol. 6. The Hague, Netherlands: IEEE, 2004, pp. 5195–5200.

[69] D. Henry, "Robust fault diagnosis of the microscope satellite micro-thrusters," in *IFAC Proceedings Volumes*, vol. 39, no. 13, 2006, pp. 342–347.

[70] M. Carpenter and K. Duda, "A new spin on space suits," *IEEE Spectrum*, vol. 52, no. 10, pp. 30–35, October 2015.

[71] R. R. Murphy and D. D. Woods, "Beyond Asimov : The Three Laws of Responsible Robotics," *IEEE Intelligent Systems*, vol. 24, no. 4, pp. 14–20, 2009.

[72] Zhihao Jiang, M. Pajic, and R. Mangharam, "Cyber–Physical Modeling of Implantable Cardiac Medical Devices," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 122–137, January 2012.

[73] G.-Z. Yang and M. Yacoub, *Body Sensor Networks*, G.-Z. Yang, Ed. London: Springer London, 2006.

[74] B. d'Enquêtes et d'Analyses pour la sécurité de l'aviation Civile, "Final report on the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France Flight AF447 Rio de Janeiro – Paris," BEA, Paris, Tech. Rep., 2012.

[75] J. Mylopoulos, *Conceptual modelling and Telos*. New York: Wiley, 1992.

[76] D. Bork, E. J. Sinz, and D. Karagiannis, "A Development Method for the Conceptual Design of Multi-View Modeling Tools with an Emphasis on Consistency Requirements," Doctoral dissertation, University of Bamberg, 2015.

[77] D. Karagiannis, R. Buchmann, and D. Bork, "Managing Consistency in Multi-View Enterprise Models: An Approach based on Semantic Queries," in *Twenty-Fourth European Conference on Information Systems (ECIS 2016)*, June 2016.

[78] D. Bork and H.-G. Fill, "Formal Aspects of Enterprise Modeling Methods: A Comparison Framework," in *2014 47th Hawaii International Conference on System Sciences*. Waikoloa, HI: IEEE, 2014, pp. 3400–3409.

[79] D. Bork, R. Buchmann, and D. Karagiannis, "Preserving Multi-view Consistency in Diagrammatic Knowledge Representation," in *Knowledge Science, Engineering and Management*, Z. S., W. M., and Z. Z., Eds. Lecture Notes in Computer Science, vol 9403. Springer, Cham, 2015, pp. 177–182.

[80] D. Bork, R. Buchmann, I. Hawryszkiewycz, D. Karagiannis, N. Tantouris, and M. Walch, "Using Conceptual Modeling to Support Innovation Challenges in Smart Cities," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Sydney, NSW, 2016, pp. 1317–1324.

Appendix A

State Space Descriptions of Open Loop Dynamics

Fig. A.1 Attitude Open Loop - Matrix A of Astrium's TF

BATT_AST <6x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	0	0																		
2	0	0	0																		
3	0	1	0																		
4	0	0	0																		
5	0	0	1																		
6	0	0	0																		
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.2 Attitude Open Loop - Matrix B of Astrium's TF

Fig. A.3 Attitude Open Loop - Matrix C of Astrium's TF

DATT_AST <3x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0																		
2	0	0	0																		
3	0	0	0																		

Fig. A.4 Attitude Open Loop - Matrix D of Astrium's TF

AATT_C_AST <18x18 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	-0.0078	-0.0039	-0.0012	-1.1681e-04	-3.4360e-06	0	0	0	0	0	0	0	0	0	0	0	0	0			
2	0.0078	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
3	0	0.0078	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
4	0	0	0.0156	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
5	0	0	0	0.0313	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
6	0	0	0	0	0.0625	0	0	0	0	0	0	0	0	0	0	0	0	0			
7	0	0	0	0	0	0	-0.0056	-0.0040	-0.0017	-4.9241e-04	-4.2134e-05	0	0	0	0	0	0	0			
8	0	0	0	0	0	0	0.0039	0	0	0	0	0	0	0	0	0	0	0			
9	0	0	0	0	0	0	0	0.0039	0	0	0	0	0	0	0	0	0	0			
10	0	0	0	0	0	0	0	0	0.0039	0	0	0	0	0	0	0	0	0			
11	0	0	0	0	0	0	0	0	0	0.0078	0	0	0	0	0	0	0	0			
12	0	0	0	0	0	0	0	0	0	0	0.0156	0	0	0	0	0	0	0			
13	0	0	0	0	0	0	0	0	0	0	0	0	-0.0057	-0.0041	-0.0018	-5.2043e-04	-4.3036e-05	0			
14	0	0	0	0	0	0	0	0	0	0	0	0	0.0039	0	0	0	0	0			
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0039	0	0	0	0			
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0039	0	0	0			
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0078	0	0			
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0156	0			

Fig. A.5 Attitude Open Loop - Matrix A of Astrium Compensator's TF

BATT_C_AST <18x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	7.6294e-06	0	0																		
2	0	0	0																		
3	0	0	0																		
4	0	0	0																		
5	0	0	0																		
6	0	0	0																		
7	0	7.6294e-06	0																		
8	0	0	0																		
9	0	0	0																		
10	0	0	0																		
11	0	0	0																		
12	0	0	0																		
13	0	0	7.6294e-06																		
14	0	0	0																		
15	0	0	0																		
16	0	0	0																		
17	0	0	0																		
18	0	0	0																		
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.6 Attitude Open Loop - Matrix B of Astrium Compensator's TF

CATT_C_AST <3x18 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	-1.2854e-08	3.5098e-08	-4.2305e-07	-5.8040e-06	-3.3425e-08	-3.6374e-11	0	0	0	0	0	0	0	0	0	0	0	0			
2	0	0	0	0	0	0	-6.1787e-11	-5.1271e-10	-1.9070e-07	-1.3289e-05	-2.3374e-07	-7.5516e-10	0	0	0	0	0	0			
3	0	0	0	0	0	0	0	0	0	0	0	0	-2.8639e-10	4.9862e-10	-1.7171e-07	-1.1261e-05	-1.5506e-07	-3.8425e-10			

Fig. A.7 Attitude Open Loop - Matrix C of Astrium Compensator's TF

DATT_C_AST <3x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0																		
2	0	0	0																		
3	0	0	0																		

Fig. A.8 Attitude Open Loop - Matrix D of Astrium Compensator's TF

Fig. A.9 Attitude Open Loop - Matrix A of Simulation Environment

BATT_LINMOD <6x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0																		
2	1	0	0																		
3	0	0	0																		
4	0	1	0																		
5	0	0	0																		
6	0	0	1																		
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.10 Attitude Open Loop - Matrix B of Simulation Environment

Fig. A.11 Attitude Open Loop - Matrix C of Simulation Environment

DATT_LINMOD <3x3 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0																		
2	0	0	0																		
3	0	0	0																		

Fig. A.12 Attitude Open Loop - Matrix D of Simulation Environment

AGDF_AST <24x24 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	-20	-8.3333	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
2	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
5	0	0	0	0	-20	-8.3333	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
6	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
7	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
9	0	0	0	0	0	0	0	0	-20	-8.3333	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
10	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
12	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0					
13	0	0	0	0	0	0	0	0	0	0	0	0	-20	-8.3333	0	0	0	0	0	0	0	0	0	0					
14	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0					
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0					
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0					
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-20	-8.3333	0	0	0	0	0	0					
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0					
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0					
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0					
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-20	-8.3333	0	0					
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0					
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0					
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0					

Fig. A.13 Drag Free Open Loop - Matrix A of Astrium's TF

BGDF_AST <24x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	4	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	4	0	0	0	0															
6	0	0	0	0	0	0															
7	0	0	0	0	0	0															
8	0	0	0	0	0	0															
9	0	0	4	0	0	0															
10	0	0	0	0	0	0															
11	0	0	0	0	0	0															
12	0	0	0	0	0	0															
13	0	0	0	4	0	0															
14	0	0	0	0	0	0															
15	0	0	0	0	0	0															
16	0	0	0	0	0	0															
17	0	0	0	0	4	0															
18	0	0	0	0	0	0															
19	0	0	0	0	0	0															
20	0	0	0	0	0	0															
21	0	0	0	0	0	4															
22	0	0	0	0	0	0															
23	0	0	0	0	0	0															
24	0	0	0	0	0	0															

Fig. A.14 Drag Free Open Loop - Matrix B of Astrium's TF

CGDF_AST <6x24 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	0	0.0156	-0.3125	2.0833	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	0	0	0	0	0	0.0156	-0.3125	2.0833	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
3	0	0	0	0	0	0	0	0	0	0.0156	-0.3125	2.0833	0	0	0	0	0	0	0	0	0	0	0	0		
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0156	-0.3125	2.0833	0	0	0	0	0	0	0	0		
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0156	-0.3125	2.0833	0	0	0	0		
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0156	-0.3125	2.0833		

Fig. A.15 Drag Free Open Loop - Matrix C of Astrium's TF

DGDF_AST <6x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.16 Drag Free Open Loop - Matrix D of Astrium's TF

AKDF_AST <36x36 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
1	-17.6558	-7.7380	-3.3708	-1.4607	-1.0081	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
2	16	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
3	0	8	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
4	0	0	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	0
5	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
6	0	0	0	0	0.0625	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
7	0	0	0	0	0	0	-15.1425	-5.6210	-4.1103	-1.4848	-0.8497	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
8	0	0	0	0	0	0	16	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
9	0	0	0	0	0	0	0	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
10	0	0	0	0	0	0	0	0	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
11	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
12	0	0	0	0	0	0	0	0	0	0	0.2500	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
13	0	0	0	0	0	0	0	0	0	0	0	0	-13.1319	-8.2811	-5.0776	-3.0496	-1.4421	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	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
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-3.5060	-1.1810	-0.7739	-0.2484	-0.0314	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	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
22	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
23	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
24	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
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-15.2681	-5.7118	-2.1020	-1.5261	-0.8761	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0
27	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	8	0	0	0	0	0	0	0	0	0	0
28	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	2	0	0	0	0	0	0	0	0	0
29	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
30	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	0.2500	0	0	0	0	0	0	0
31	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	-6.0319	-3.4837	-1.9529	-1.0711	-0.2311	0
32	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	4	0	0	0	0	0
33	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	2	0	0	0	0
34	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
35	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
36	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

Fig. A.17 Drag Free Open Loop - Matrix A of Astrium Compensator's TF

BKDF_AST <36x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0.5000	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7	0	0.5000	0	0	0	0															
8	0	0	0	0	0	0															
9	0	0	0	0	0	0															
10	0	0	0	0	0	0															
11	0	0	0	0	0	0															
12	0	0	0	0	0	0															
13	0	0	0.5000	0	0	0															
14	0	0	0	0	0	0															
15	0	0	0	0	0	0															
16	0	0	0	0	0	0															
17	0	0	0	0	0	0															
18	0	0	0	0	0	0															
19	0	0	0	0.0313	0	0															
20	0	0	0	0	0	0															
21	0	0	0	0	0	0															
22	0	0	0	0	0	0															
23	0	0	0	0	0	0															
24	0	0	0	0	0	0															
25	0	0	0	0	0.5000	0															
26	0	0	0	0	0	0															
27	0	0	0	0	0	0															
28	0	0	0	0	0	0															
29	0	0	0	0	0	0															
30	0	0	0	0	0	0															
31	0	0	0	0	0	0.1250															
32	0	0	0	0	0	0															
33	0	0	0	0	0	0															
34	0	0	0	0	0	0															
35	0	0	0	0	0	0															
36	0	0	0	0	0	0															
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.18 Drag Free Open Loop - Matrix B of Astrium Compensator's TF

CKDF_AST <6x36 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
1	0	0	0	0.6810	0.1326	0.0129	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
2	0	0	0	0	0	0	0	0	0	0.5357	0.0799	0.0024	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.6940	0.0916	0.0011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0576	0.0023	4.7739e-06	0	0	0	0	0	0	0	0	0	0	0	0
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	0.5178	0.0699	0.0025	0	0	0	0	0	0
6	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	0.1499	0.0099	2.9580e-05

Fig. A.19 Drag Free Open Loop - Matrix C of Astrium Compensator's TF

DKDF_AST <6x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					

Fig. A.20 Drag Free Open Loop - Matrix D of Astrium Compensator's TF

ADF_LINMOD <12x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	0	0	0	0	0	0	0	0	0	0
2	1.3025e-06	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	1
12	0	0	0	0	0	0	0	0	0	0	0	0

Fig. A.21 Drag Free Open Loop - Matrix A of Simulation Environment

BDF_LINMOD <12x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	1	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	1	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	1	0	0	0															
7	0	0	0	0	0	0															
8	0	0	0	1	0	0															
9	0	0	0	0	0	0															
10	0	0	0	0	1	0															
11	0	0	0	0	0	0															
12	0	0	0	0	0	1															
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.22 Drag Free Open Loop - Matrix B of Simulation Environment

CDF_LINMOD <6x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	0	0	0	0	0	0	0	0	0	0	0									
2	0	0	1	0	0	0	0	0	0	0	0	0									
3	0	0	0	0	1	0	0	0	0	0	0	0									
4	0	0	0	0	0	0	1	0	0	0	0	0									
5	0	0	0	0	0	0	0	0	1	0	0	0									
6	0	0	0	0	0	0	0	0	0	0	1	0									

Fig. A.23 Drag Free Open Loop - Matrix C of Simulation Environment

DDF_LINMOD <6x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.24 Drag Free Open Loop - Matrix D of Simulation Environment

ASUS_AST <12x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0.0020	0	0	0	0	0	0	0	0	0	0									
2	0.0039	0	0	0	0	0	0	0	0	0	0	0									
3	0	0	0	0.0031	0	0	0	0	0	0	0	0									
4	0	0	0.0020	0	0	0	0	0	0	0	0	0									
5	0	0	0	0	0	0.0010	0	0	0	0	0	0									
6	0	0	0	0	0.0020	0	0	0	0	0	0	0									
7	0	0	0	0	0	0	0	0.0020	0	0	0	0									
8	0	0	0	0	0	0	0.0039	0	0	0	0	0									
9	0	0	0	0	0	0	0	0	0	0.0020	0	0									
10	0	0	0	0	0	0	0	0	0.0039	0	0	0									
11	0	0	0	0	0	0	0	0	0	0	0	0.0031									
12	0	0	0	0	0	0	0	0	0	0	0.0020	0									

Fig. A.25 Suspension Open Loop - Matrix A of Astrium's TF

BSUS_AST <12x6 double>

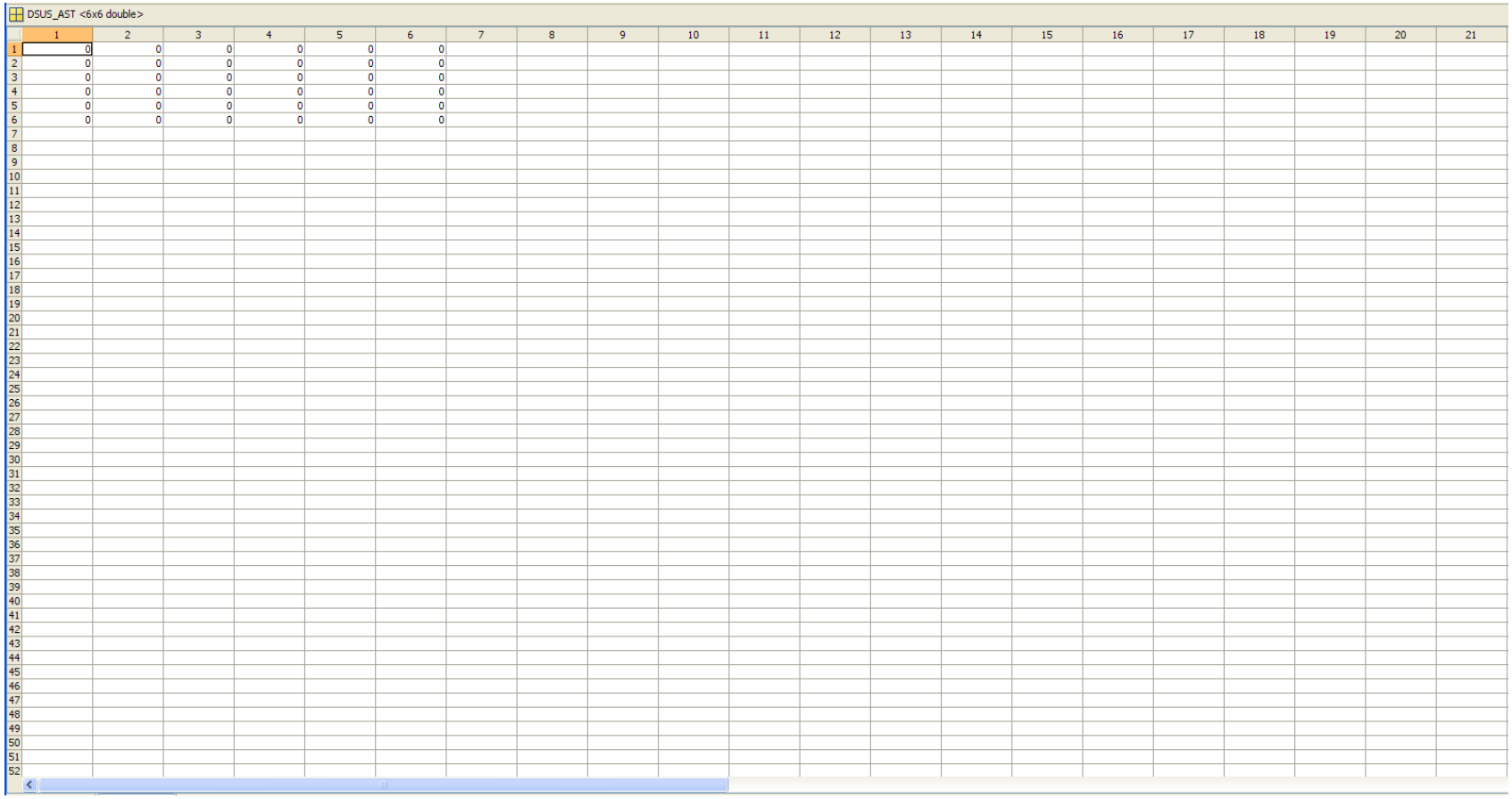
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	16	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	32	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	32	0	0	0															
6	0	0	0	0	0	0															
7	0	0	0	16	0	0															
8	0	0	0	0	0	0															
9	0	0	0	0	16	0															
10	0	0	0	0	0	0															
11	0	0	0	0	0	32															
12	0	0	0	0	0	0															
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.26 Suspension Open Loop - Matrix B of Astrium's TF

CSUS_AST <6x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	16	0	0	0	0	0	0	0	0	0	0									
2	0	0	0	16	0	0	0	0	0	0	0	0									
3	0	0	0	0	0	16	0	0	0	0	0	0									
4	0	0	0	0	0	0	0	16	0	0	0	0									
5	0	0	0	0	0	0	0	0	0	16	0	0									
6	0	0	0	0	0	0	0	0	0	0	0	16									

Fig. A.27 Suspension Open Loop - Matrix C of Astrium's TF



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					

Fig. A.28 Suspension Open Loop - Matrix D of Astrium's TF

ASUS_C_AST <24x24 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	-0.6645	-0.0943	-0.0161	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0.2500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0.1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0.2500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	-0.9639	-0.5660	-0.0162	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0.5000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0.5000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	-0.2185	-0.1530	-0.0496	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0.1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0.1250	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0.1250	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	-6.5534	-0.8512	-0.0298	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5000	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0.6645	-0.0943	-0.0161	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.2500	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1250	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.2500	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-0.9639	-0.5660	-0.0162	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5000	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5000	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Fig. A.29 Suspension Open Loop - Matrix A of Astrium Compensator's TF

BSUS_C_AST <24x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0.0039	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0.0078	0	0	0	0															
6	0	0	0	0	0	0															
7	0	0	0	0	0	0															
8	0	0	0	0	0	0															
9	0	0	0.0156	0	0	0															
10	0	0	0	0	0	0															
11	0	0	0	0	0	0															
12	0	0	0	0	0	0															
13	0	0	0	0.0156	0	0															
14	0	0	0	0	0	0															
15	0	0	0	0	0	0															
16	0	0	0	0	0	0															
17	0	0	0	0	0.0039	0															
18	0	0	0	0	0	0															
19	0	0	0	0	0	0															
20	0	0	0	0	0	0															
21	0	0	0	0	0	0.0078															
22	0	0	0	0	0	0															
23	0	0	0	0	0	0															
24	0	0	0	0	0	0															

Fig. A.30 Suspension Open Loop - Matrix B of Astrium Compensator's TF

CSUS_C_AST <6x24 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	0.0056	1.5597e-04	1.7818e-07	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0.0064	4.7213e-05	3.2835e-08	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0.0085	5.3067e-04	2.7328e-05	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0172	1.6026e-04	3.8101e-08	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0056	1.5597e-04	1.7818e-07	0	0	0	0	
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0064	4.7213e-05	3.2835e-08	

Fig. A.31 Suspension Open Loop - Matrix C of Astrium Compensator's TF

DSUS_C_AST <6x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															
7																					
8																					
9																					
10																					
11																					
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.32 Suspension Open Loop - Matrix D of Astrium Compensator's TF

ASUS_LINMOD <12x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	1	0	0	0	0	0	0	0	0	0	0									
2	4.3704e-06	0	0	0	0	0	0	0	0	0	0	0									
3	0	0	0	1	0	0	0	0	0	0	0	0									
4	0	0	0	0	0	0	0	0	0	0	0	0									
5	0	0	0	0	0	1	0	0	0	0	0	0									
6	0	0	0	0	0	0	0	0	0	0	0	0									
7	0	0	0	0	0	0	0	1	0	0	0	0									
8	0	0	0	0	0	0	0	0	0	0	0	0									
9	0	0	0	0	0	0	0	0	0	1	0	0									
10	0	0	0	0	0	0	0	0	0	0	0	0									
11	0	0	0	0	0	0	0	0	0	0	0	1									
12	0	0	0	0	0	0	0	0	0	0	0	0									

Fig. A.33 Suspension Open Loop - Matrix A of Simulation Environment

BSUS_LINMOD <12x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	1	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	1	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	1	0	0	0															
7	0	0	0	0	0	0															
8	0	0	0	1	0	0															
9	0	0	0	0	0	0															
10	0	0	0	0	1	0															
11	0	0	0	0	0	0															
12	0	0	0	0	0	1															
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.34 Suspension Open Loop - Matrix B of Simulation Environment

Fig. A.35 Suspension Open Loop - Matrix C of Simulation Environment

DSUS_LINMOD <6x6 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0															
2	0	0	0	0	0	0															
3	0	0	0	0	0	0															
4	0	0	0	0	0	0															
5	0	0	0	0	0	0															
6	0	0	0	0	0	0															

Fig. A.36 Suspension Open Loop - Matrix D of Simulation Environment

ADFSUS <22x22 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig. A.37 Combination of Drag Free and Suspension Loops - Matrix A of Simulation Environment

BDFSUS <22x19 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	0.0026	-0.0012	-3.8155e-04	0.0025	-0.0011	-3.8289e-04	-8.3473e-04	-0.0013	1	0	0	0	0	0	0	0	0	0	0		
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
4	9.3022e-04	8.6648e-04	-0.0015	-0.0016	5.0722e-04	6.0182e-04	-9.2207e-04	-8.8465e-04	0	1	0	0	0	0	0	0	0	0	0		
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
6	8.2202e-04	0.0011	-0.0019	-0.0015	-6.2814e-04	-0.0013	0.0018	0.0017	0	0	1	0	0	0	0	0	0	0	0		
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
8	0.0011	3.9920e-04	8.6727e-04	0.0013	-0.0027	0.0012	3.8859e-04	-0.0025	0	0	0	1	0	0	0	0	0	0	0		
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
10	5.2265e-04	5.8639e-04	-9.3410e-04	-8.7262e-04	9.4565e-04	8.5105e-04	-0.0015	-0.0015	0	0	0	0	1	0	0	0	0	0	0		
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
12	-0.0011	-7.4492e-04	0.0015	0.0018	0.0012	6.6284e-04	-0.0016	-0.0018	0	0	0	0	0	-1	0	0	0	0	0		
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
14	0.0042	-0.0042	-0.0033	0.0033	0.0042	-0.0042	-0.0033	0.0033	0	0	0	0	0	0	-1	0	0	0	0		
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
16	-3.9166e-04	0.0019	0.0019	2.3030e-04	4.0240e-04	-0.0019	-0.0019	-2.2192e-04	0	0	0	0	0	0	0	-1	0	0	0		
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
18	8.2202e-04	0.0011	-0.0019	-0.0015	-6.2814e-04	-0.0013	0.0018	0.0017	0	0	0	0	0	0	0	0	-1	0	0		
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
20	-0.0011	-7.4492e-04	0.0015	0.0018	0.0012	6.6284e-04	-0.0016	-0.0018	0	0	0	0	0	0	0	0	0	-1	0		
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
22	0.0042	-0.0042	-0.0033	0.0033	0.0042	-0.0042	-0.0033	0.0033	0	0	0	0	0	0	0	0	0	0	-1		

Fig. A.38 Combination of Drag Free and Suspension Loops - Matrix B of Simulation Environment

CDFSUS <11x22 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Fig. A.39 Combination of Drag Free and Suspension Loops - Matrix C of Simulation Environment

DDFSUS <11x19 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. A.40 Combination of Drag Free and Suspension Loops - Matrix D of Simulation Environment

Appendix B

Drag Free and Suspension Loop Dynamics (Combined) for the Detection Filter - Modified B Matrix

Fig. B.1 Combined Drag Free and Suspension Loop - Matrix A of FDI Techniques

BDFSUS_DET_F <22x8 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0	0	0													
2	0.0026	-0.0012	-3.8155e-04	0.0025	-0.0011	-3.8289e-04	-8.3473e-04	-0.0013													
3	0	0	0	0	0	0	0	0													
4	9.3022e-04	8.6648e-04	-0.0015	-0.0016	5.0722e-04	6.0182e-04	-9.2207e-04	-8.8465e-04													
5	0	0	0	0	0	0	0	0													
6	8.2202e-04	0.0011	-0.0019	-0.0015	-6.2814e-04	-0.0013	0.0018	0.0017													
7	0	0	0	0	0	0	0	0													
8	0.0011	3.9920e-04	8.6727e-04	0.0013	-0.0027	0.0012	3.8859e-04	-0.0025													
9	0	0	0	0	0	0	0	0													
10	5.2265e-04	5.8639e-04	-9.3410e-04	-8.7262e-04	9.4565e-04	8.5105e-04	-0.0015	-0.0015													
11	0	0	0	0	0	0	0	0													
12	-0.0011	-7.4492e-04	0.0015	0.0018	0.0012	6.6284e-04	-0.0016	-0.0018													
13	0	0	0	0	0	0	0	0													
14	0.0042	-0.0042	-0.0033	0.0033	0.0042	-0.0042	-0.0033	0.0033													
15	0	0	0	0	0	0	0	0													
16	-3.9166e-04	0.0019	0.0019	2.3030e-04	4.0240e-04	-0.0019	-0.0019	-2.2192e-04													
17	0	0	0	0	0	0	0	0													
18	8.2202e-04	0.0011	-0.0019	-0.0015	-6.2814e-04	-0.0013	0.0018	0.0017													
19	0	0	0	0	0	0	0	0													
20	-0.0011	-7.4492e-04	0.0015	0.0018	0.0012	6.6284e-04	-0.0016	-0.0018													
21	0	0	0	0	0	0	0	0													
22	0.0042	-0.0042	-0.0033	0.0033	0.0042	-0.0042	-0.0033	0.0033													
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					

Fig. B.2 Combined Drag Free and Suspension Loop - Matrix B of FDI Techniques

Fig. B.3 Combined Drag Free and Suspension Loop - Matrix C of FDI Techniques

DDFSUS_DET_F <11x8 double>

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	0	0	0	0	0	0	0													
2	0	0	0	0	0	0	0	0													
3	0	0	0	0	0	0	0	0													
4	0	0	0	0	0	0	0	0													
5	0	0	0	0	0	0	0	0													
6	0	0	0	0	0	0	0	0													
7	0	0	0	0	0	0	0	0													
8	0	0	0	0	0	0	0	0													
9	0	0	0	0	0	0	0	0													
10	0	0	0	0	0	0	0	0													
11	0	0	0	0	0	0	0	0													
12																					
13																					
14																					
15																					
16																					
17																					
18																					
19																					
20																					
21																					
22																					
23																					
24																					
25																					
26																					
27																					
28																					
29																					
30																					
31																					
32																					
33																					
34																					
35																					
36																					
37																					
38																					
39																					
40																					
41																					
42																					
43																					
44																					
45																					
46																					
47																					
48																					
49																					
50																					
51																					
52																					
53																					

Fig. B.4 Combined Drag Free and Suspension Loop - Matrix D of FDI Techniques

Appendix C

Diagnosis Filter (The true identification of the Failed Thruster was missed in 430 Cases, of which 302 with Negative Severity)

Table C.1 Euphoria Filter - Missed Identification Cases for the 6^{th} case

Begin of Table

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.05	1	−0.061310591	−0.06
6	0.05	1	−0.054844044	−0.05
1	0.07	2	−0.054136608	−0.05
4	0.05	2	−0.046701643	−0.05
4	0.03	5	0.028336245	0.03
6	0.09	1	−0.08889456	−0.09
2	0.03	1	−0.04118333	−0.04
1	0.05	5	0.042491209	0.04
2	0.01	1	−0.02884842	−0.03
4	0.03	5	0.029565063	0.03
2	0.05	1	−0.05409358	−0.05
1	0.07	2	−0.055479359	−0.06
2	0.09	1	−0.084154537	−0.08
1	0.05	5	0.043112787	0.04
4	0.05	2	−0.04090668	−0.04
1	0.05	5	0.03872375	0.04
7	0.01	8	0.010528629	0.01
2	0.05	1	−0.052619075	−0.05
1	0.03	5	0.029920285	0.03
1	0.09	6	−0.072529175	−0.07
4	0.03	5	0.025596409	0.03
2	0.01	1	−0.026704585	−0.03
2	0.01	1	−0.026384788	−0.03
3	0.05	1	−0.057779853	−0.06
4	0.01	5	0.015891293	0.02
2	0.03	1	−0.040769215	−0.04
3	0.03	1	−0.044024093	−0.04
2	0.05	1	−0.055252829	−0.06

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.03	1	−0.041526168	−0.04
1	0.07	6	−0.051370871	−0.05
3	0.05	1	−0.06061371	−0.06
1	0.01	5	0.017637747	0.02
2	0.09	1	−0.080967507	−0.08
7	0.01	8	0.010284915	0.01
2	0.03	1	−0.040411548	−0.04
6	0.07	1	−0.073568186	−0.07
3	0.03	1	−0.042953722	−0.04
6	0.01	1	−0.028316978	−0.03
6	0.03	1	−0.042955118	−0.04
1	0.09	6	−0.07923341	−0.08
2	0.05	1	−0.057902014	−0.06
2	0.01	1	−0.027161431	−0.03
1	0.09	6	−0.077375676	−0.08
2	0.01	1	−0.027899539	−0.03
6	0.03	1	−0.044327536	−0.04
2	0.09	1	−0.093004768	−0.09
1	0.07	2	−0.055525058	−0.06
6	0.07	1	−0.073568186	−0.07
4	0.03	5	0.026974204	0.03
2	0.01	1	−0.026367131	−0.03
3	0.01	1	−0.027911007	−0.03
1	0.09	6	−0.070795714	−0.07
4	0.01	5	0.018084546	0.02
2	0.01	1	−0.025142098	−0.03
1	0.09	6	−0.073584194	−0.07
2	0.09	1	−0.085326283	−0.09
3	0.03	1	−0.045014193	−0.05
6	0.03	1	−0.042010821	−0.04
1	0.01	5	0.018963634	0.02

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
3	0.03	1	−0.042313308	−0.04
2	0.03	1	−0.042313867	−0.04
6	0.07	1	−0.074028675	−0.07
4	0.03	5	0.027656552	0.03
7	0.01	8	0.010119408	0.01
2	0.07	1	−0.072678687	−0.07
2	0.07	1	−0.075145125	−0.08
4	0.01	5	0.016965479	0.02
6	0.07	1	−0.076827088	−0.08
3	0.01	1	−0.028434088	−0.03
6	0.09	1	−0.085874299	−0.09
3	0.03	1	−0.045622145	−0.05
1	0.07	2	−0.059205841	−0.06
3	0.05	1	−0.058203482	−0.06
6	0.01	1	−0.028200683	−0.03
7	0.01	8	0.010245393	0.01
6	0.07	1	−0.071468364	−0.07
3	0.03	1	−0.042757311	−0.04
1	0.09	2	−0.073496841	−0.07
6	0.07	1	−0.073568186	−0.07
6	0.03	1	−0.044327536	−0.04
2	0.01	1	−0.029371296	−0.03
4	0.01	5	0.017681018	0.02
2	0.03	1	−0.04098656	−0.04
6	0.07	1	−0.073568186	−0.07
1	0.01	5	0.018494714	0.02
4	0.03	5	0.02860585	0.03
2	0.05	1	−0.055526663	−0.06
6	0.01	1	−0.027336849	−0.03
1	0.05	5	0.043319006	0.04
3	0.01	1	−0.027808595	−0.03

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
1	0.09	6	−0.073432541	−0.07
1	0.01	5	0.017688687	0.02
6	0.03	1	−0.043002984	−0.04
1	0.01	5	0.017688687	0.02
3	0.05	1	−0.058562449	−0.06
2	0.03	1	−0.041925418	−0.04
4	0.03	5	0.02961401	0.03
6	0.09	1	−0.085907219	−0.09
4	0.01	5	0.017681605	0.02
2	0.01	1	−0.028420772	−0.03
6	0.05	1	−0.060972165	−0.06
4	0.03	5	0.02939649	0.03
4	0.01	5	0.017207353	0.02
4	0.01	5	0.016226604	0.02
3	0.03	1	−0.045910283	−0.05
2	0.05	1	−0.055074609	−0.06
6	0.07	1	−0.073264607	−0.07
3	0.03	1	−0.042953722	−0.04
4	0.03	5	0.028741007	0.03
3	0.01	1	−0.027388524	−0.03
2	0.01	1	−0.028132479	−0.03
2	0.03	1	−0.043062696	−0.04
3	0.03	1	−0.044602309	−0.04
6	0.07	1	−0.071380605	−0.07
2	0.03	1	−0.04118333	−0.04
7	0.01	8	0.008537949	0.01
6	0.09	1	−0.08707992	−0.09
2	0.03	1	−0.04154244	−0.04
4	0.01	5	0.018078374	0.02
2	0.03	1	−0.044362707	−0.04
4	0.03	5	0.02437181	0.02

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
1	0.05	5	0.0405775	0.04
6	0.03	1	−0.047692369	−0.05
1	0.03	5	0.028879555	0.03
6	0.01	1	−0.02814415	−0.03
2	0.07	1	−0.067765369	−0.07
6	0.07	1	−0.072549278	−0.07
6	0.09	1	−0.090679576	−0.09
3	0.01	1	−0.031636472	−0.03
4	0.03	5	0.027128264	0.03
4	0.03	5	0.025118212	0.03
6	0.03	1	−0.04316405	−0.04
1	0.07	2	−0.056198792	−0.06
1	0.09	6	−0.071909122	−0.07
2	0.05	1	−0.055481311	−0.06
1	0.07	2	−0.057317906	−0.06
3	0.01	1	−0.027854746	−0.03
6	0.09	1	−0.082448219	−0.08
6	0.03	1	−0.039989628	−0.04
2	0.05	1	−0.055252829	−0.06
7	0.01	8	0.011147497	0.01
6	0.03	1	−0.041506473	−0.04
3	0.05	1	−0.058239533	−0.06
3	0.03	1	−0.041727245	−0.04
4	0.01	5	0.016868027	0.02
1	0.01	5	0.016838688	0.02
3	0.01	1	−0.02825094	−0.03
2	0.01	1	−0.027594489	−0.03
6	0.07	1	−0.0715637	−0.07
1	0.03	5	0.028879555	0.03
4	0.01	5	0.018505206	0.02
2	0.01	1	−0.025924722	−0.03

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
4	0.01	1	−0.01391228	−0.01
1	0.01	5	0.018874851	0.02
1	0.03	5	0.03063809	0.03
6	0.03	1	−0.046143774	−0.05
1	0.07	2	−0.051884012	−0.05
4	0.01	5	0.015939219	0.02
6	0.09	1	−0.092572313	−0.09
1	0.09	6	−0.083771175	−0.08
2	0.07	1	−0.069490063	−0.07
6	0.09	1	−0.089354056	−0.09
6	0.03	1	−0.041263008	−0.04
6	0.03	1	−0.042955118	−0.04
3	0.05	1	−0.058349533	−0.06
6	0.09	1	−0.089005168	−0.09
2	0.05	1	−0.057747797	−0.06
2	0.09	1	−0.084648312	−0.08
3	0.03	1	−0.041337687	−0.04
4	0.03	5	0.029395571	0.03
1	0.09	6	−0.07519877	−0.08
3	0.01	1	−0.02703127	−0.03
4	0.03	5	0.02772244	0.03
2	0.05	1	−0.055440672	−0.06
2	0.07	1	−0.067294626	−0.07
1	0.09	6	−0.069819619	−0.07
7	0.01	8	0.009097634	0.01
4	0.03	5	0.02998005	0.03
1	0.09	6	−0.070255427	−0.07
2	0.01	1	−0.026845808	−0.03
4	0.05	2	−0.043977312	−0.04
2	0.05	1	−0.060078902	−0.06
6	0.07	1	−0.074320872	−0.07

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
1	0.05	5	0.038434158	0.04
1	0.09	6	−0.07361438	−0.07
3	0.05	1	−0.06332375	−0.06
1	0.03	5	0.030668913	0.03
2	0.05	1	−0.055252829	−0.06
1	0.03	5	0.02908108	0.03
2	0.03	1	−0.041603393	−0.04
4	0.03	5	0.026590662	0.03
6	0.01	1	−0.027720383	−0.03
7	0.01	8	0.009836185	0.01
2	0.05	1	−0.053206599	−0.05
2	0.05	1	−0.05492879	−0.05
1	0.01	5	0.016326392	0.02
2	0.09	1	−0.083391827	−0.08
3	0.03	1	−0.04458742	−0.04
6	0.07	1	−0.068560513	−0.07
1	0.01	5	0.015368935	0.02
1	0.03	5	0.025331479	0.03
1	0.05	5	0.040232276	0.04
3	0.05	1	−0.058726645	−0.06
6	0.01	1	−0.02850147	−0.03
4	0.03	5	0.029381965	0.03
3	0.03	1	−0.04326762	−0.04
3	0.03	1	−0.042797773	−0.04
3	0.03	1	−0.041828495	−0.04
2	0.09	1	−0.083391827	−0.08
7	0.01	8	0.007447366	0.01
3	0.01	1	−0.027775755	−0.03
6	0.07	1	−0.07377282	−0.07
4	0.03	5	0.028336245	0.03
6	0.03	1	−0.041336877	−0.04

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.07	1	−0.070406136	−0.07
6	0.07	1	−0.076662226	−0.08
3	0.03	1	−0.04326762	−0.04
4	0.01	5	0.018034714	0.02
3	0.01	1	−0.027560243	−0.03
7	0.01	8	0.008537949	0.01
6	0.05	1	−0.063645975	−0.06
2	0.03	1	−0.04475937	−0.04
1	0.11	6	−0.089711088	−0.09
7	0.01	8	0.010666416	0.01
2	0.09	1	−0.084709243	−0.08
6	0.07	1	−0.071380605	−0.07
3	0.05	1	−0.057724453	−0.06
6	0.05	1	−0.06102715	−0.06
1	0.07	2	−0.059869503	−0.06
4	0.01	5	0.015819147	0.02
4	0.03	5	0.027846423	0.03
6	0.09	1	−0.094530473	−0.09
4	0.01	5	0.01917824	0.02
1	0.01	5	0.015265724	0.02
2	0.05	1	−0.054924879	−0.05
4	0.03	5	0.028674042	0.03
1	0.01	5	0.017913697	0.02
2	0.03	1	−0.042406462	−0.04
2	0.09	1	−0.083235695	−0.08
7	0.01	8	0.010119408	0.01
2	0.09	1	−0.08514362	−0.09
6	0.07	1	−0.075656387	−0.08
6	0.05	1	−0.057730854	−0.06
6	0.03	1	−0.040849877	−0.04
6	0.07	1	−0.076274953	−0.08

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.05	1	−0.058431752	−0.06
2	0.03	1	−0.04111772	−0.04
6	0.05	1	−0.058359576	−0.06
1	0.05	5	0.043636127	0.04
2	0.09	1	−0.076427553	−0.08
7	0.01	8	0.011548816	0.01
1	0.01	5	0.016803528	0.02
3	0.01	1	−0.026988767	−0.03
2	0.01	1	−0.027161431	−0.03
3	0.05	1	−0.062940238	−0.06
6	0.01	1	−0.030720909	−0.03
6	0.07	1	−0.073568186	−0.07
4	0.01	5	0.017873983	0.02
2	0.09	1	−0.087707145	−0.09
6	0.03	1	−0.047624261	−0.05
4	0.03	5	0.028334484	0.03
4	0.03	5	0.02959249	0.03
1	0.09	6	−0.071950992	−0.07
6	0.07	1	−0.076662226	−0.08
3	0.03	1	−0.042128824	−0.04
7	0.01	8	0.008773694	0.01
4	0.01	5	0.016518662	0.02
2	0.03	1	−0.043440014	−0.04
2	0.01	1	−0.027593311	−0.03
2	0.05	1	−0.055806166	−0.06
1	0.01	5	0.017095678	0.02
2	0.07	1	−0.068500382	−0.07
7	0.01	8	0.010609975	0.01
1	0.07	2	−0.056198792	−0.06
2	0.07	1	−0.067436746	−0.07
2	0.07	1	−0.069095276	−0.07

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
3	0.05	1	−0.058562449	−0.06
6	0.05	1	−0.060353099	−0.06
1	0.09	6	−0.073584194	−0.07
2	0.05	1	−0.053905985	−0.05
6	0.01	1	−0.028117936	−0.03
2	0.01	1	−0.028406443	−0.03
3	0.05	1	−0.05838351	−0.06
2	0.09	1	−0.08514362	−0.09
1	0.09	6	−0.071577302	−0.07
7	0.01	8	0.008773694	0.01
3	0.03	1	−0.043013218	−0.04
1	0.05	5	0.040496131	0.04
2	0.01	1	−0.026057776	−0.03
1	0.07	6	−0.056368186	−0.06
1	0.07	2	−0.055532566	−0.06
4	0.01	5	0.017681605	0.02
1	0.09	6	−0.073862054	−0.07
4	0.01	1	−0.014073114	−0.01
6	0.01	1	−0.027057481	−0.03
6	0.01	1	−0.02850147	−0.03
1	0.03	5	0.02924173	0.03
1	0.03	5	0.029398668	0.03
2	0.01	1	−0.026373226	−0.03
7	0.01	8	0.008923648	0.01
4	0.01	1	−0.014744989	−0.01
1	0.05	5	0.04076159	0.04
1	0.05	5	0.039606572	0.04
4	0.03	5	0.028597338	0.03
6	0.01	1	−0.025931512	−0.03
2	0.03	1	−0.039996889	−0.04
7	0.01	8	0.010088324	0.01

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
1	0.01	5	0.016769182	0.02
2	0.09	1	−0.083391827	−0.08
2	0.09	1	−0.08183401	−0.08
1	0.01	5	0.018455154	0.02
3	0.03	1	−0.044087055	−0.04
4	0.01	5	0.017681605	0.02
2	0.05	1	−0.050358718	−0.05
1	0.05	5	0.040232276	0.04
3	0.01	1	−0.029096284	−0.03
6	0.01	1	−0.028157496	−0.03
2	0.03	1	−0.041725527	−0.04
6	0.01	1	−0.027720383	−0.03
2	0.09	1	−0.090636404	−0.09
2	0.01	1	−0.028970771	−0.03
4	0.03	5	0.027446144	0.03
2	0.07	1	−0.074008555	−0.07
1	0.07	2	−0.053675556	−0.05
2	0.01	1	−0.02800502	−0.03
3	0.03	1	−0.044024093	−0.04
2	0.07	1	−0.067126186	−0.07
2	0.03	1	−0.04118333	−0.04
3	0.01	1	−0.028186596	−0.03
3	0.05	1	−0.055126484	−0.06
6	0.09	1	−0.089134674	−0.09
3	0.03	1	−0.041239543	−0.04
2	0.07	1	−0.069567954	−0.07
3	0.03	1	−0.044693026	−0.04
2	0.09	1	−0.080437103	−0.08
1	0.07	2	−0.056198792	−0.06
2	0.03	1	−0.04118333	−0.04
4	0.01	5	0.019078911	0.02

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
2	0.03	1	−0.042945499	−0.04
4	0.01	5	0.018524348	0.02
1	0.03	5	0.027645073	0.03
1	0.07	2	−0.056198792	−0.06
6	0.05	1	−0.056006966	−0.06
6	0.05	1	−0.063354208	−0.06
6	0.01	1	−0.029592044	−0.03
2	0.05	1	−0.05423303	−0.05
6	0.05	1	−0.058205809	−0.06
2	0.01	1	−0.028102419	−0.03
6	0.05	1	−0.058205809	−0.06
3	0.05	1	−0.057355295	−0.06
2	0.07	1	−0.067410459	−0.07
1	0.05	5	0.039890134	0.04
1	0.03	5	0.029465782	0.03
7	0.01	8	0.008818845	0.01
1	0.03	5	0.026645169	0.03
6	0.09	1	−0.085393709	−0.09
2	0.07	1	−0.067716868	−0.07
1	0.05	5	0.039966721	0.04
1	0.01	5	0.0191124	0.02
1	0.07	2	−0.059869503	−0.06
2	0.03	1	−0.042021474	−0.04
1	0.07	2	−0.053254796	−0.05
1	0.11	6	−0.095714044	−0.1
1	0.09	6	−0.073584194	−0.07
2	0.09	1	−0.085628634	−0.09
1	0.03	5	0.028797845	0.03
3	0.01	1	−0.029818931	−0.03
7	0.01	8	0.010264852	0.01
2	0.09	1	−0.085394877	−0.09

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.07	1	−0.072728905	−0.07
4	0.03	5	0.028868192	0.03
4	0.03	5	0.02576499	0.03
1	0.07	6	−0.056128466	−0.06
2	0.03	1	−0.041154847	−0.04
6	0.05	1	−0.059230856	−0.06
2	0.05	1	−0.053337883	−0.05
1	0.01	5	0.018094653	0.02
4	0.03	5	0.026380147	0.03
1	0.05	5	0.038374471	0.04
3	0.05	1	−0.06031712	−0.06
2	0.01	1	−0.027001919	−0.03
6	0.05	1	−0.055957842	−0.06
6	0.03	1	−0.044607809	−0.04
6	0.01	1	−0.028237376	−0.03
2	0.05	1	−0.055252829	−0.06
2	0.05	1	−0.053206599	−0.05
4	0.03	5	0.0278583	0.03
6	0.07	1	−0.073665867	−0.07
6	0.01	1	−0.02606291	−0.03
6	0.05	1	−0.05630525	−0.06
3	0.01	1	−0.027560243	−0.03
1	0.09	6	−0.07324416	−0.07
4	0.01	5	0.018124099	0.02
6	0.07	1	−0.074460599	−0.07
3	0.01	1	−0.029086448	−0.03
1	0.05	5	0.037133117	0.04
6	0.03	1	−0.046725414	−0.05
1	0.09	6	−0.069646406	−0.07
2	0.05	1	−0.054563978	−0.05
6	0.03	1	−0.04399869	−0.04

Continuation of Table C.1

Random Failed Thruster	Actual Random Severity	FDI	Identified Severity	Round-Off Severity
6	0.01	1	−0.028454089	−0.03
6	0.05	1	−0.058597533	−0.06
1	0.03	5	0.031472233	0.03
6	0.03	1	−0.042101724	−0.04
3	0.05	1	−0.055803097	−0.06
1	0.07	2	−0.056618066	−0.06
3	0.03	1	−0.04326762	−0.04
1	0.05	5	0.038074169	0.04
4	0.03	5	0.028786952	0.03
1	0.09	6	−0.073432541	−0.07
7	0.01	8	0.008965189	0.01
6	0.05	1	−0.058285585	−0.06
2	0.09	1	−0.082801516	−0.08
2	0.09	1	−0.081266226	−0.08
1	0.01	5	0.016724229	0.02
1	0.01	5	0.017929883	0.02
4	0.03	5	0.025886268	0.03
1	0.07	2	−0.054136608	−0.05
4	0.03	5	0.029622378	0.03
2	0.01	1	−0.027113831	−0.03
1	0.03	5	0.029991387	0.03
1	0.05	5	0.037790238	0.04
6	0.03	1	−0.040358491	−0.04
1	0.01	5	0.017868975	0.02
1	0.03	5	0.027685445	0.03
6	0.07	1	−0.07377282	−0.07
4	0.03	5	0.02775212	0.03
3	0.03	1	−0.040982218	−0.04
2	0.05	1	−0.054441604	−0.05
2	0.03	1	−0.04118333	−0.04

End of Table