

# FAULT DIAGNOSIS AND PROGNOSIS OF SATELLITE ATTITUDE CONTROL SYSTEM WITH REACTION WHEELS AND CONTROL MOMENT GYROS

by

Afshin Rahimi

Bachelor of Science, K.N. Toosi University of Technology (2010)

Master of Applied Science, Ryerson University (2012)

A dissertation  
presented to Ryerson University  
in partial fulfillment of the  
requirements for the degree of  
Doctor of Philosophy  
in the Program of  
Aerospace Engineering

Toronto, Ontario, Canada, 2017

© Afshin Rahimi 2017

## Author's Declaration

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

---

Afshin Rahimi

# Abstract

## **Fault Diagnosis and Prognosis of Satellite Attitude Control System with Reaction Wheels and Control Moment Gyros**

Afshin Rahimi, Doctor of Philosophy, Aerospace Engineering  
Ryerson University, Toronto, August 2017

Condition-based maintenance (CBM) and prognostics and health management (PHM), as consisting parts of diagnosis, prognosis, and health monitoring (DPHM) framework, have developed over the past decades to remedy the limitations of the traditional maintenance practices for complex systems. In space, where mass and power budget are restricted, application of CBM and PHM has become more vital to the success of a mission.

Reaction wheels (RW) and Control Moment Gyros (CMG), as the most commonly used actuators onboard satellites, are prone to faults and failures. The ability to detect faults, isolate their location and severity, and estimate the remaining useful life (RUL) of the faulty unit can enhance mission success rate and reduce maintenance and damage costs extensively.

Therefore, in this thesis, a model-based DPHM framework is developed and evaluated. Firstly, a novel fault detection algorithm is proposed, using Unscented Kalman filters (UKF) in conjunction with residual and innovation sequences, for detecting agile faults in RW/CMG onboard satellites. Secondly, a novel fault isolation algorithm is proposed, using UKF, Bayes' probability and interacting multiple models (IMM), to isolate the location of the fault and its severity. Finally, a new fault prognosis approach is proposed, using UKF and particle filters (PF) to estimate the RUL of a faulty unit. Extensive simulations were conducted for each phase of the DPHM to verify advantages of the proposed techniques over the available methods in the literature.

Extensive simulations were conducted to evaluate the performance of the proposed

methods in each module of the framework. Regarding the proposed fault detection scheme, results showed superior performance of the proposed adaptation technique compared to the original UKF and a previously developed AUKF. The proposed fault isolation scheme was able to successfully isolate the faulty unit at multiple levels of isolation including formation level, system level, and actuator level with over 99% success rate for formation level, over 99% success rate for the RW assembly and for up to 90% success rate for the CMG assembly in the system level. For the CMG assembly, due to direct estimation of the fault parameters, it was possible to determine the severity of the faults as well as their location. Finally, the proposed fault prognosis approach provided RUL estimates with errors as low as 1.5% compared to the actual remaining useful life.

Overall, the proposed framework can be regarded as a promising tool for fault detection, isolation and identification, and prognosis of the complex nonlinear systems. Furthermore, the proposed framework can be extended to other complex systems in space including multi-agent formation systems and other areas where the model of the system under study is available.

# Acknowledgements

After four years and a half and many uncertain and turbulent times, I have finally completed my Ph.D. thesis. I am grateful for my supervisors Dr. Krishna Dev Kumar and Dr. Hekmat Alighanbari and my external examiner Dr. Khashayar Khorasani for thoroughly reviewing my work and providing me with immense amount of constructive feedback. I appreciate the challenges they threw my way and their confidence in me.

Over these past years, I came to know other knowledgeable engineers such as members of the SSDC lab. I appreciate your friendship and camaraderie, and during this period I learned a lot from each one of you. I want to thank you Bahram, Michael, Alex, Sobhan, and Venkatesh individually and let you know that all those memories will never be forgotten. Amirhossein Ahadi and Sogol Bandehali were nothing but my brother and sister during hardship and peace. I am grateful to have them in my life and am hopeful to remain as close with them, regardless of what the future holds.

Industrial opportunities at Bombardier Transportation through MITACS program and Pratt & Whitney Canada through RIADI program were invaluable. I gained an immense amount of knowledge during these fellowships and could not be more exposed to technical implementations of my research any other way.

It was an honor to receive the Vanier Canada Graduate Scholarship for my research and contributions in the field of aerospace focusing on fault diagnosis and prognosis. This scholarship helped me with the peace of mind to focus on my research and perform better in the absence of financial concerns.

Last but not the least my parents Majid and Farideh, are the ones whom their unconditional love made them the most selfless individuals I have ever come to know in my life. I will always be indebted to them and feel blessed and privileged to have them as my parents. This thesis is dedicated to them, in the hope that it would make them content and proud of the son they raised to this point.

To my loving parents

*Majid Rahimi* and *Farideh Farhang*,

without whose support and endless love, I would not be here.

# Table of Contents

List of Tables .....	x
List of Figures .....	xiii
Nomenclature .....	xvii
Introduction .....	1
1.1 Motivation .....	2
1.2 Satellite Attitude Control System.....	5
1.2.1 ACS Units .....	6
1.2.2 Actuators.....	7
1.3 Fault Detection, Isolation, and Prognosis .....	8
1.3.1 Fault Classification.....	8
1.3.2 Fault Diagnosis and Prognosis Cycle.....	10
1.4 Literature Review.....	11
1.4.1 Fault Detection.....	11
1.4.2 Fault Isolation .....	12
1.4.3 Fault Prognosis.....	15
1.5 Problem Statement .....	17
1.6 Research Objectives .....	19
1.7 Main Contributions.....	21
1.8 Thesis Outline.....	22
Preliminaries .....	23
2.1 Satellite Attitude Model.....	23
2.1.1 Satellite Dynamics .....	24
2.1.2 Attitude Kinematics .....	25
2.2 Formation Flying.....	27
2.2.1 Decentralized FDI architecture.....	28
2.2.2 Semi-decentralized FDI architecture.....	28
2.2.3 Centralized FDI architecture .....	29
2.3 Actuator Model.....	30
2.3.1 Reaction Wheel Model.....	30
2.3.2 Reaction Wheel Assembly .....	34
2.3.3 Control Moment Gyro.....	34
2.4 Kalman Filters.....	38
2.4.1 Unscented Kalman Filter.....	38

2.4.2	Adaptive Unscented Kalman Filter .....	41
2.5	Particle Filter.....	41
2.5.1	Algorithm .....	42
2.5.2	Degeneracy .....	43
2.5.3	Importance Density Choice.....	44
2.5.4	Resampling.....	44
2.5.5	Alternatives to Sub-optimality .....	45
2.5.6	Variations of Particle Filter.....	45
2.6	Confusion Matrix .....	46
2.7	Conclusions .....	47
Fault Detection .....		48
3.1	Proposed Covariance Adaptive UKF (CAUKF) .....	49
3.1.1	Fault Annunciation Metric.....	49
3.1.2	Filter Adaptation Algorithm .....	50
3.2	Stability of the CAUKF.....	51
3.2.1	Instrumental Diagonal Matrix and Extra Positive Definite Matrix.....	51
3.2.2	Stochastic Boundedness of Estimation Error .....	52
3.3	Error Performance of the CAUKF .....	53
3.3.1	Cramér–Rao lower bound .....	53
3.3.2	The Mean Square Error of the UKF.....	54
3.4	Computational Complexity of the CAUKF.....	54
3.4.1	Fault Annunciation Complexity .....	55
3.4.2	Fault Adaptation Complexity.....	56
3.4.3	Overall Complexity.....	56
3.5	Noise Considerations.....	57
3.6	Results and Discussion.....	58
3.6.1	Fault Types .....	58
3.6.2	Monte Carlo Simulations .....	59
3.6.3	Reaction Wheels .....	60
3.6.4	Control Moment Gyros.....	71
3.7	Conclusions .....	76
Fault Isolation.....		77
4.1	Proposed Methodology.....	77
4.1.1	Fault Detection.....	79
4.1.2	Likelihood-based Isolation .....	80
4.1.3	Adaptive Windowing.....	83



4.2	Results and Discussion.....	84
4.2.1	Formation Level .....	84
4.2.2	RW Assembly.....	88
4.2.3	CMG Assembly .....	100
4.3	Conclusions .....	104
	Fault Prognosis .....	105
5.1	Model-based Prognostics.....	106
5.2	Model Definition .....	106
5.3	Particle Filter Process.....	108
5.3.1	Step 1 – Initialization .....	108
5.3.2	Step 2 – Update.....	108
5.3.3	Step 3 – Resampling .....	109
5.4	Predicting the Damage State and RUL .....	110
5.5	Results and Discussion.....	113
5.5.1	Exponential Decline.....	113
5.5.2	Normal Distribution .....	117
5.5.3	Lognormal Distribution .....	121
5.6	Conclusions .....	123
	Conclusions and Future Work.....	124
6.1	Highlights.....	124
6.2	Summary of Contributions.....	126
6.3	Future Work.....	127
	References .....	128

# List of Tables

Table 1.1 Satellite program costs by mass [3], [4].....	2
Table 1.2 Summary of on-orbit failure of MW/RW/CMG [12].....	4
Table 2.1 ITHACO Type ‘A’ Reaction Wheel Parameters [76].....	33
Table 2.2 Sample confusion matrix.....	46
Table 2.3 Terminology for confusion matrix [157] .....	46
Table 2.4 Sample Monte Carlo confusion matrix.....	47
Table 3.1 Proposed CAUKF covariance adaptation algorithm.....	50
Table 3.2 Matrix square root algorithm computational requirement [164] .....	55
Table 3.3 Computational complexity of different estimation methods .....	56
Table 3.4 Sample Monte Carlo confusion matrix.....	59
Table 3.5 Fault detection simulation parameters for RW .....	61
Table 3.6 Abrupt Bus Voltage fault profile .....	61
Table 3.7 Abrupt Bus Voltage fault performance measures .....	63
Table 3.8 MC for LOE 5%, $\phi=10$ .....	64
Table 3.9 MC for LOE 5%, $\phi=20$ .....	64
Table 3.10 MC for LOE 5%, $\phi=30$ .....	64
Table 3.11 MC for LOE 5%, $\phi=40$ .....	64
Table 3.12 MC for LOE 5%, $\phi=50$ .....	64
Table 3.13 MC for LOE 5%, $\phi=100$ .....	64
Table 3.14 MC for LOE 5%, $\phi=150$ .....	64
Table 3.15 MC for LOE 5%, $\phi=200$ .....	64
Table 3.16 MC for LOE 10%, $\phi=10$ .....	65
Table 3.17 MC for LOE 10%, $\phi=20$ .....	65
Table 3.18 MC for LOE 10%, $\phi=30$ .....	65
Table 3.19 MC for LOE 10%, $\phi=40$ .....	65
Table 3.20 MC for LOE 10%, $\phi=50$ .....	65
Table 3.21 MC for LOE 10%, $\phi=100$ .....	65
Table 3.22 MC for LOE 10%, $\phi=150$ .....	65
Table 3.23 MC for LOE 10%, $\phi=200$ .....	65
Table 3.24 MC for LOE 15%, $\phi=10$ .....	66
Table 3.25 MC for LOE 15%, $\phi=20$ .....	66
Table 3.26 MC for LOE 15%, $\phi=30$ .....	66
Table 3.27 MC for LOE 15%, $\phi=40$ .....	66
Table 3.28 MC for LOE 15%, $\phi=50$ .....	66

Table 3.29 MC for LOE 15%, $\phi=100$ .....	66
Table 3.30 MC for LOE 15%, $\phi=150$ .....	66
Table 3.31 MC for LOE 15%, $\phi=200$ .....	66
Table 3.32 MC for LOE 20%, $\phi=10$ .....	67
Table 3.33 MC for LOE 20%, $\phi=20$ .....	67
Table 3.34 MC for LOE 20%, $\phi=30$ .....	67
Table 3.35 MC for LOE 20%, $\phi=40$ .....	67
Table 3.36 MC for LOE 20%, $\phi=50$ .....	67
Table 3.37 MC for LOE 20%, $\phi=100$ .....	67
Table 3.38 MC for LOE 20%, $\phi=150$ .....	67
Table 3.39 MC for LOE 20%, $\phi=200$ .....	67
Table 3.40 Transient Bus Voltage fault profile.....	67
Table 3.41 Transient Bus Voltage fault performance measures.....	68
Table 3.42 Intermittent Bus Voltage and motor torque fault profile.....	69
Table 3.43 Intermittent Bus Voltage and motor torque performance measures .....	69
Table 3.44 Incipient Bus Voltage fault profile.....	70
Table 3.45 Incipient Bus Voltage performance measures.....	70
Table 3.46 Fault detection simulation parameters CMG.....	74
Table 3.47 CMG fault parameters out-of-phase abrupt fault profile .....	76
Table 3.48 CMG out-of-phase abrupt fault scenario performance measures.....	76
Table 4.1 Fault scenarios for RW/CMG assemblies and satellite formation .....	81
Table 4.2 Logic for determining output of the isolation scheme.....	84
Table 4.3 Single BEMF constant and Bus Voltage fault .....	85
Table 4.4 Confusion matrix for 4 satellites in formation .....	87
Table 4.5 Simulation Parameters [5], [7].....	89
Table 4.6 Single BEMF constant and Bus Voltage fault .....	90
Table 4.7 Confusion matrix for three orthogonal assembly .....	97
Table 4.8 Confusion matrix for pyramid assembly .....	97
Table 4.9 Multiple BEMF constant and Bus Voltage fault profile – Scenario 1.....	99
Table 4.10 Multiple BEMF constant and Bus Voltage fault profile – Scenario 2 .....	99
Table 4.11 CMG fault parameters in-phase abrupt fault profile.....	101
Table 4.12 CMG in-phase abrupt fault scenario performance measures .....	101
Table 4.13 Confusion matrix for CMG assembly.....	102
Table 5.1 Exponential decline in Motor Torque Constant fault profile .....	114
Table 5.2 Exponential decline in Motor Torque Constant performance measures.....	114
Table 5.3 Fault prognosis simulation parameters .....	117
Table 5.4 Fault prognosis for a normally distributed sample set with 500 particles.....	120
Table 5.5 Fault prognosis for a normally distributed sample set with 100 particles.....	120

Table 5.6 Fault prognosis for a normally distributed sample set with 1000 particles ...	120
Table 5.7 Fault prognosis for a log-normally distributed sample set with 500 particles .....	122
Table 5.8 Fault prognosis for a log-normally distributed sample set with 100 particles .....	123
Table 5.9 Fault prognosis for a log-normally distributed sample set with 1000 particles .....	123

# List of Figures

Figure 1.1 SpaceWorks small satellites launch historical, current, and future projections data .....	3
Figure 1.2 SpaceWorks nano/microsatellites trends by sector .....	3
Figure 1.3 SpaceWorks nano/microsatellite trends by purpose: (Left) 2009-2016 (Right) 2017-2019.....	3
Figure 1.4 Schematic of the fault tolerant satellite attitude control system .....	5
Figure 1.5 Satellite attitude control system units.....	6
Figure 1.6 Types of faults based on addition to the system .....	8
Figure 1.7 Types of faults based on time characteristics .....	9
Figure 1.8 The CBM/PHM cycle .....	10
Figure 1.9 Major fault detection methods [46].....	12
Figure 1.10 Major fault diagnosis methods [46] .....	13
Figure 1.11 Major fault prognosis methods [2].....	16
Figure 1.12 Different stage of a model-based CBM/PHM framework .....	22
Figure 2.1 Geometry of orbit motion for rigid body spacecraft .....	24
Figure 2.2 Decentralized FDI architecture [134] .....	28
Figure 2.3 Semi-decentralized FDI architecture [134] .....	29
Figure 2.4 Centralized FDI architecture [134] .....	29
Figure 2.5 Reaction wheel model [76] .....	31
Figure 2.6 Reaction wheel unit developed in the SSDC lab at Ryerson University .....	33
Figure 2.7 Torque decomposition for RW assembly (a) pyramid, (b) standard four-wheel .....	34
Figure 2.8 RW assembly (a) pyramid configuration, (b) standard four-wheel configuration .....	36
Figure 2.9 Pyramid assembly of four SGCMGs (a) schematics (b) free body diagram [135].....	37
Figure 2.10 Schematics of the particle filter algorithm.....	44
Figure 3.1 Placement of fault detection module in a model-based CBM/PHM flow .....	48
Figure 3.2 Comparison of $\beta$ and its threshold for transient fault scenario (a) and (b) between (c) fixed and (d) adaptive approaches .....	58
Figure 3.3 Simulation setup for performance evaluation of the proposed filter.....	60
Figure 3.4 Case 1 – state and parameters estimates for abrupt fault with $\phi=10$ .....	62
Figure 3.5 Case 1 - state and parameters residuals for abrupt fault with $\phi=10$ .....	62
Figure 3.6 Case 1 – $\beta$ and $\beta_0$ for abrupt fault with $\phi=10$ .....	62

Figure 3.7 Case 1 – comparison of <b><i>Kt</i></b> estimates for abrupt fault with $\phi=10$ .....	63
Figure 3.8 Case 1 – comparison of <b><i>Vbus</i></b> estimates for abrupt fault with $\phi=10$ .....	63
Figure 3.9 Case 2 – comparison of <b><i>Kt</i></b> estimates for transient fault with $\phi=10$ .....	68
Figure 3.10 Case 2 – comparison of <b><i>Vbus</i></b> estimates for transient fault with $\phi=10$ .....	68
Figure 3.11 Case 3 – comparison of <b><i>kt</i></b> estimates for intermittent fault with $\phi=10$ .....	69
Figure 3.12 Case 3 – comparison of <b><i>Vbus</i></b> estimates for intermittent fault with $\phi=10$ ...	70
Figure 3.13 Case 4 – comparison of <b><i>Kt</i></b> estimates for incipient fault with $\phi=10$ .....	71
Figure 3.14 Case 4 – comparison of <b><i>Vbus</i></b> estimates for incipient fault with $\phi=10$ .....	71
Figure 3.15 Simulation results for CMG fault isolation out-of-phase abrupt CMG faults with $\phi=10$ (a) Satellite attitude (b) Satellite angular velocities (c) Scaler $\beta$ trend (d) CMGs gimbal angles.....	74
Figure 3.16 Comparison of fault parameters estimates for out-of-phase abrupt fault with $\phi=10$ (a) CMG1 (b) CMG2 (c) CMG3 (d) CMG4.....	75
Figure 4.1 Placement of fault isolation module in a model-based CBM/PHM flow .....	77
Figure 4.2 Proposed hierarchical approach towards fault isolation.....	78
Figure 4.3 Proposed fault isolation scheme based on likelihood and probability distribution .....	82
Figure 4.4 Proposed fault isolation scheme for formation level.....	85
Figure 4.5 Attitude response of the system under fault scenario 15 with pyramid RW assembly: (a) satellite #1 quaternions (b) satellite #2 quaternions (c) satellite #3 quaternions (d) satellite #4 quaternions .....	86
Figure 4.6 Angular velocity of the system under fault scenario 15 with pyramid RW assembly: (a) satellite #1 angular velocities (b) satellite #2 angular velocities (c) satellite #3 angular velocities (d) satellite #4 angular velocities.....	86
Figure 4.7 Angular velocity of the RWs for all satellites under fault scenario 15.....	87
Figure 4.8 Proposed fault isolation simulation setup.....	90
Figure 4.9 Attitude response of the system under fault scenario 1 with pyramid RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes ..	91
Figure 4.10 Attitude response of the system under fault scenario 1 with standard four- wheel RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes .....	94
Figure 4.11 Fault isolation performance, scenario 1, pyramid assembly, uniform initial distribution: first iteration (a) prior probability (b) likelihood (c) posterior probability; final iteration (d) prior probability (e) likelihood (f) posterior probability.....	95
Figure 4.12 Fault isolation performance, scenario 1, pyramid assembly, normal initial distribution: first iteration (a) prior probability (b) likelihood (c) posterior probability;	

final iteration (d) prior probability (e) likelihood (f) posterior probability.....	95
Figure 4.13 Fault isolation performance, scenario 1, standard four-wheel assembly: first iteration (a) prior probability (b) likelihood (c) posterior probability; final iteration (d) prior probability (e) likelihood (f) posterior probability .....	96
Figure 4.14 Fault isolation performance, scenario 6 and 9, pyramid assembly: final iteration for scenario 6(a) prior probability (b) likelihood (c) posterior probability; final iteration for scenario 9 (d) prior probability (e) likelihood (f) posterior probability .....	96
Figure 4.15 Attitude response of the system to multiple faults scenario 1 with pyramid RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes ..	98
Figure 4.16 Attitude response of the system to multiple faults scenario 2 with pyramid RW assembly: (a),(e) satellite quaternions (b),(d) satellite angular velocities (c) reaction wheels torques .....	99
Figure 4.17 Fault isolation performance, multiple fault scenario, pyramid assembly: final iteration of scenario 1 (a) prior probability distribution (b) likelihood (c) posterior probability distribution; final iteration of scenario 2 (d) prior probability distribution (e) likelihood (f) posterior probability distribution .....	100
Figure 4.18 Simulation results for CMG fault isolation in-phase abrupt CMG faults with $\phi=10$ (a) Satellite attitude (b) Satellite angular velocities (c) Scaler $\beta$ trend (d) CMGs gimbal angles.....	102
Figure 4.19 Comparison of fault parameters estimates for in-phase abrupt fault with $\phi=10$ (a) CMG1 (b) CMG2 (c) CMG3 (d) CMG4.....	103
Figure 5.1 Placement of fault prognosis module in a model-based CBM/PHM flow ....	105
Figure 5.2 Different stages of model-based prognostics .....	106
Figure 5.3 Fault location statistics for a group of reaction wheels.....	107
Figure 5.4 Particle filtering process visual representation.....	110
Figure 5.5 Inverse CDF resampling method .....	110
Figure 5.6 RUL prediction process schematics .....	111
Figure 5.7 State and parameters estimates for incipient fault with $\phi=10$ .....	115
Figure 5.8 $\beta$ and $\beta_0$ for incipient fault with $\phi=10$ .....	115
Figure 5.9 State and parameters residuals for incipient fault with $\phi=10$ .....	115
Figure 5.10 Comparison of $Kt$ estimates for incipient fault with $\phi=10$ .....	116
Figure 5.11 Comparison of $Vbus$ estimates for incipient fault with $\phi=10$ .....	116
Figure 5.12 Flow of the fault detection leading to fault prognosis.....	116
Figure 5.13 Fault prognosis for $kt$ under normal noise.....	118
Figure 5.14 RUL distribution based on $kt$ damage model for normal noise .....	118
Figure 5.15 Fault prognosis for $b$ under normal noise .....	119
Figure 5.16 Fault prognosis for $\sigma$ under normal noise .....	119

Figure 5.17 Fault prognosis for <b><math>kt</math></b> under lognormal noise.....	121
Figure 5.18 RUL distribution based on <b><math>kt</math></b> damage model for lognormal noise.....	122
Figure 5.19 Fault prognosis for <b><math>b</math></b> under lognormal noise.....	122
Figure 5.20 Fault prognosis for <b><math>\sigma</math></b> under lognormal noise .....	123



# Nomenclature

## Alphabetical

$I$	identity matrix
$k_m$	back electromotive force constant
$X_I, Y_I, Z_I$	x, y, z-axes for the Earth-centered inertial frame
$X_O, Y_O, Z_O$	x, y, z-axes for the orbital frame
$X_B, Y_B, Z_B$	x, y, z-axes for the body-fixed frame
$\bar{e}$	principal axis from Euler's theorem
$q_v \in \mathbb{R}^{3 \times 1}$	Euler parameters representing the spacecraft body frame orientation
$C_L^B$	orientation transformation matrix from body frame to orbital frame
$R_c$	spacecraft distance from the center of the Earth
$R_E$	average Earth radius
$h$	angular momentum generated by a momentum exchange device
$h_s$	altitude of the spacecraft from the surface of the Earth
$I_{bus}$	bus current
$V_{bus}$	bus voltage
$I_m$	motor current
$K_e$	back electromotive force voltage gain
$T$	temperature
$f_1$	motor disturbance term
$f_2$	motor disturbance term
$f_3$	the electromotive torque limiting block
$f_4$	approximation of the sign function in the Coulomb friction block
$f_5$	speed limiter block
$F_p$	fault parameters matrix
$f_p$	fault parameter
$V_{Comm}$	torque command voltage
B	ripple torque
C	cogging torque
$k_t$	motor torque constant
$R_B$	bridge resistance

$G_d$	driver gain
$N$	number of motor poles
$R_{IN}$	input filter resistance
$P_q$	quiescent bus power
$k_f$	voltage feedback gain
$k_s$	over-speed circuit gain
$X_k \in \mathbb{R}^n$	state vector at time step $k$
$Z_k \in \mathbb{R}^m$	measurement vector at time step $k$
$w_k \in \mathbb{R}^n$	additive process noise at time step $k$
$v_k \in \mathbb{R}^m$	additive measurement noise at time step $k$
$Q_k$	process noise covariance matrix
$R_k$	measurement noise covariance matrix
$W^{(c)}$	component weight matrix for covariance calculation
$W^{(m)}$	component weight matrix for mean calculation
$n$	dimension of the states
$y$	output of the measurement model
$P_{yy}$	covariance of the measurements matrix
$x$	$n$ -state random variable
$x_p$	sigma points
$dt = \Delta t = T_s$	discretization step size
$P_{xx}$	posteriori estimates covariance matrix
$Y_i$	the result of instantiated sigma points through measurement model
$P_{xy}$	cross covariance matrix
$K$	Kalman filter gain
$C_r$	residual sequence covariance matrix
$N$	window size
$N_s$	number of samples
$N_{eff}$	effective sample size
$w_k^*$	true weight
$PP_k$	temporary posteriori estimate covariance matrix
$r$	residual error
$V_{comm}$	input voltage
$V_{comm_0}$	initial input voltage

$Q_0$	initial process noise covariance matrix
$R_0$	initial measurement noise covariance matrices
$I_m$	reaction wheel motor current
$n_\beta$	adaptive threshold factor for the
$i$	each faulty case scenario
$j$	index in the residual window
$r_{i,0}$	initial value for the residual
$i_{out}$	index of the case with maximum probability
$\mathcal{C}_{out}$	threshold for confidence in the output of the logic
$A \in \mathbb{R}^{3 \times 4}$	reaction wheel assembly mapping matrix
$J \in \mathbb{R}^{3 \times 3}$	satellite moment of inertia including the actuators
$J_s \in \mathbb{R}^{3 \times 3}$	satellite moment of inertia
$J_w \in \mathbb{R}^{4 \times 4}$	RWs moment of inertia
$k_{ts}$	failure threshold

## Greek

$\mathcal{L}$	orbital frame
$\mathcal{B}$	body-fixed coordinate frame
$\mathcal{J}$	earth-centered inertial (ECI) reference frame
$\omega_{BI}^B$	angular velocity relative to inertial frame expressed in body frame
$\tau_e \in \mathbb{R}^{3 \times 1}$	external torque
$\Omega \in \mathbb{R}^{4 \times 1}$	axial angular velocity of the reaction wheels
$\tau_{RW}$	torque generated by RWs
$\Phi$	principal angle
$\omega_0$	magnitude of the orbital angular velocity of LVLH frame
$\dot{\theta}$	orbital frame angular velocity
$\mu$	gravitational constant parameter
$\tau_c$	control torque applied by the momentum exchange device
$\tau_{gg}$	gravity gradient torque
$\tau_{srp}$	solar radiation pressure
$\tau_{aero}$	aerodynamic torque
$\tau_d$	other disturbances
$\omega_m$	reaction wheel angular speed

$\tau_v$	viscous torque
$\tau_{noise}$	torque noise disturbance
$\theta_a$	Jitter Angle
$\omega_d$	Driver Bandwidth
$\omega_s$	Maximum Wheel Speed
$\alpha$	in-plane angle for RW and CMG
$\beta$	out-of-plane angle for RW and CMG
$\delta$	control moment gyro gimbal angle
$\delta_{kj}$	Kronecker- $\delta$ function
$\beta_0$	scalar value $\beta$ threshold for CAUKF
$\mu_\beta$	mean scalar value $\beta$ for CAUKF
$\sigma_\beta$	standard deviation of scalar value $\beta$ for CAUKF
$\alpha, \kappa, \lambda$	Kalman filter parameters
$\sigma_Q$	process noise standard deviation
$\sigma_R$	measurement noise standard deviation
$\sigma_n$	motor current noise standard deviation
$\zeta$	adaptive fading factor for Kalman filter

## Notations

$(\dot{x})$	$d(x)/dt$
$(\ddot{x})$	$d^2(x)/dt^2$
$[x]^\times$	skew-symmetric matrix for matrix $x$
$[x]^T$	transpose operation of matrix $x$
$\ x\ $	Euclidian norm of $x$
$ x $	absolute value of $x$
$\hat{x}$	estimate of $x$
$\bar{x}$	mean of $x$
$x_0$	value of $x$ at time 0 (initial value)
$x^+$	post-process value of $x$
$x^-$	pre-process value of $x$
$v_{LI}^B$	a vector of frame $\mathcal{L}$ relative to frame $\mathfrak{I}$ , expressed in frame $\mathcal{B}$

## Functions

$f(\cdot)$	nonlinear process model
$g(\cdot)$	nonlinear measurement model
$Var(\cdot)$	variance of a set
$Pr(\cdot)$	probability of the condition inside brackets
$\sin(\cdot) = s(\cdot)$	sine
$\cos(\cdot) = c(\cdot)$	cosine
$trace(\cdot)$	sum of the diagonal elements of a matrix
$\delta_{X_k^i}(\cdot)$	Kronecker- $\delta$ function, resulting in 1 if $X_k^i$ is in the set and 0 otherwise
$q(\cdot)$	importance density
$p(\cdot)$	probability density function
$\pi(\cdot)$	importance density or density of a distribution
$E[\cdot]$	expected value

## Acronyms

AAD	Attitude Anomaly Detection
ACC	Accuracy
ACS	Attitude Control System
AR	Autoregressive
ARIMA	Autoregressive Integrated Moving Average
ASIR	Auxiliary Sampling Importance Resampling
AUKF	Adaptive Unscented Kalman Filter
BEMF	Back Electromotive Force
BLDC	Brushless Direct Current
BM	Bayesian Methods
CAUKF	Covariance Matching Adaptive Unscented Kalman Filter
CBM	Condition-Based Maintenance
CDF	Cumulative Distribution Function
CI	Confidence Interval
CMG	Control Moment Gyro
CRLB	Cramér–Rao lower bound
CSS	Coarse Sun Sensor
DGCMG	Double-gimbal Control Moment Gyro

DPHM	Diagnosis, Prognosis, and Health Monitoring
ECI	Earth Centered Inertial
EKF	Extended Kalman Filter
FDI	Fault Detection, Isolation, and Identification
FLOPS	Floating Point Operations Per Second
FN	False Negative
FP	False Positive
FSS	Fine Sun Sensor
FUSE	Far Ultraviolet Spectroscopic Explorer
GYR	Gyroscope
HI	Health Index
HOF	Hard-Over Failure
IID	Independent and Identically Distributed
ILO	Iterative Learning Observer
IMM	Interacting Multiple Models
INPID	Iterative Neuron Proportional, Integral, and Derivative
JAXA	Japan Aerospace Exploration Agency
JWST	James Webb Space Telescope
KF	Kalman Filter
LEO	Low Earth Orbit
LIP	Lock in Place
LOE	Loss of Effectiveness
LVLH	Local Vertical Local Horizontal
MC	Monte Carlo
MGM	Magnetometer
MGT	Magnetorquer
MIT	Massachusetts Institute of Technology
MM	Multiple Model
MSE	Mean Squared Error
MTTF	Mean Time to Failure
MW	Momentum Wheel
ND-AR	Nonlinear Degradation Auto Regression
PCA	Principal Component Analysis

PF	Particle Filter
PHM	Prognostics and Health Management
PPV	Positive Predictive Value
PR	Prediction Interval
RK4	Runge-Kutta
RMS	Root Mean Squared
RPF	Regularized Particle Filter
RRG	Robotics Research Group
RUL	Remaining Useful Life
RVM	Relevance Vector Machine
RW	Reaction Wheel
RWA	RW Assembly
SDA	Stacked Denoising Autoencoder
SGCMG	Single-gimbal Control Moment Gyro
SHM	Structural Health Monitoring
SIR	Sampling Importance Resampling
SIS	Sequential Importance Sampling
SMC	Sequential MC
SOH	State of Health
STR	Star Sensor
SVM	Support Vector Machine
SVR	Support Vector Regression
TESS	Transiting Exoplanet Survey Satellite
TIMED	Thermosphere, Ionosphere, Mesosphere Energetics and Dynamics
TN	True Negative
TP	True Positive
UKF	Unscented Kalman Filter
WAFIT	Weight Application to Exponential Parameters

## CHAPTER 1

# Introduction

---

In the recent years, the topic of intelligent systems and autonomous vehicles has attracted a wide range of scientists and industries. The need for more intelligent systems comes from the fact that technological systems are vulnerable to faults [1]. With the paradigm shift in systems complexity and advances in technological processes[2], it is imperative to formulate and employ autonomous systems capable of self-monitoring and maintenance. Over the past years, condition-based maintenance (CBM), prognostics, and health management (PHM) have emerged as powerful and significant technologies that have positively influenced the military and commercial sectors.

Reaction wheels (RW) and control moment gyros (CMG) are the most commonly used actuators in spacecraft attitude control system; they are prone to malfunction, which could lead to catastrophic failures. Such malfunctions can be detected and addressed in time if proper analytical redundancy algorithms such as parameter estimation and control reconfiguration are employed. Major challenges in parameter estimation include speed and accuracy of the employed algorithm. Tracking changes in system parameters for mechanical systems has been of constant interest for system monitoring and control [1–4]. Analytical redundancy, as one of the major subsidies in this field, has evolved to remedy the limitations of hardware redundancy for complex systems. Such limitations include major budget constraints, space limitations for design and manufacturing, concerns with safety and reliability, etc. One of the major challenges in system monitoring and fault detection is to achieve the ability for tracking sudden changes in non-measurable system parameters; this becomes more challenging when the system under study is nonlinear and complex. Reaction wheels and control moment gyros can be considered as such and are prone to hardware failures [5]. Sudden changes in non-measurable RW/CMG parameters can occur while operating in space. If such changes are not tracked precisely, and with a reasonable delay, catastrophic failures could occur. Therefore, a framework that could detect faults, isolate and identify their location and severity onboard satellite and forecast the remaining useful life (RUL) of the faulty components can help develop fail-safe satellites where hardware redundancy is not possible. This can be accomplished through providing alternative options for the system to maintain its partial operability in order to accomplish its mission or fail without catastrophic consequences through reprogrammed control logic for less risky failures. This thesis, aims to address this need in space system actuators by proposing a CBM/PHM framework.



## 1.1 Motivation

Heavier satellites need larger launchers, whereas, smaller satellites only require lower end launch vehicles and can be launched in multiples. Therefore, they bring about financial benefits making it possible to save costs for design and mass production. They also bring about the opportunity to enable missions, otherwise impossible by larger satellites such as arrangements for low data rate communications, formations flying to gather data from multiple points, in-orbit inspection, data collection, and monitoring of larger satellites, and finally university-related research projects.

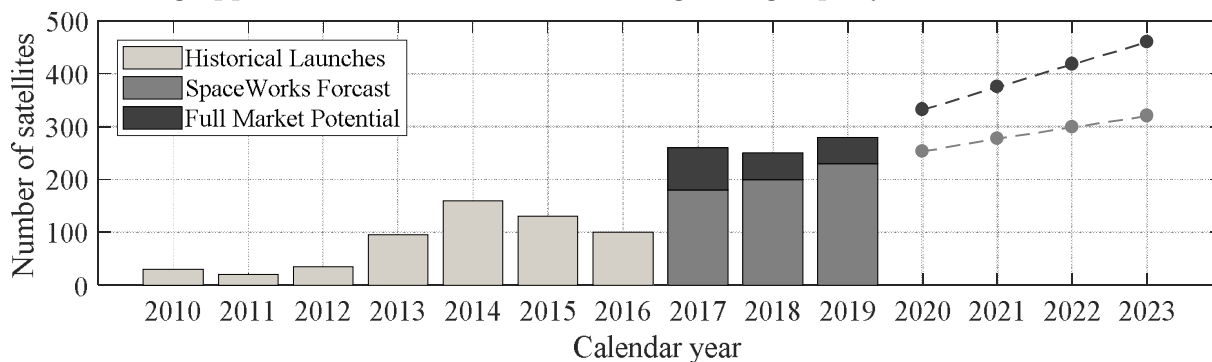
Satellites, in terms of mass, can be categorized into eight major categories [3], [4]. Table 1.1 shows each class of satellites with their corresponding mass and production costs. As can be observed from Table 1.1, production costs immensely decrease with the decrease in mass and size of the satellite. Therefore, smaller satellites like RyeSat [5], BeeSat [6], and BeeSat-2 [7], individually or in a formation are preferred compared to their larger costly counterparts.

**Table 1.1 Satellite program costs by mass [3], [4]**

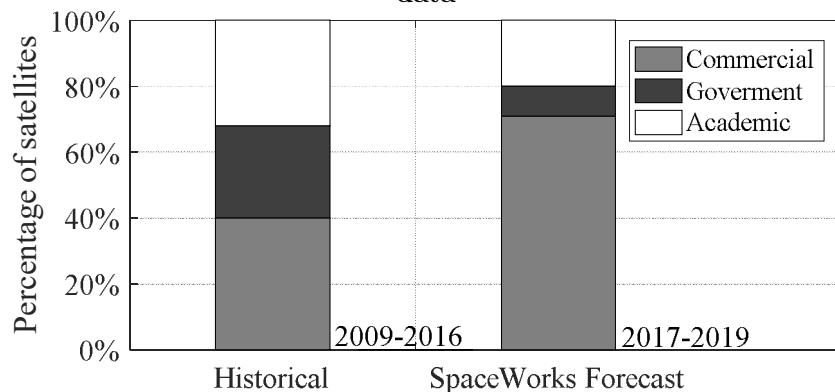
Class	Mass (kg)	Cost (M US\$)	Class	Mass (kg)	Cost (M US\$)
Large	>1000	>100	Nano	1-10	0.5-1
Medium	500-1000	50-100	Pico	0.1-1	<0.08
Mini	100-500	5-20	Femto	0.01-0.1	<0.0001
Micro	10-100	2-3	Atto	0.001-0.01	~100 US\$

This preference has started a shift in the way satellites are being designed, developed, and manufactured. Tafazoli [8] has provided an extensive review of 156 on-orbit spacecraft failures conducted on more than 130 different spacecraft up to 2009 where he provides some insights on how to avoid most common failures in the most vulnerable satellite subsystems. SpaceWorks Enterprises publishes an annual report on small satellites marketing assessments. Their 2014 report [9] analyses the developments to date and provides a potential market for future developments until 2020. SpaceWorks had estimated that 93 nano/microsatellites would launch globally in 2013 where 92 were actually launched and showed a 269% increase over 2012. In SpaceWorks' 2017 report [10] it is mentioned that they had estimated 210 nano/microsatellites global launches across all sectors in 2016; however, only 101 nano/microsatellites were actually launched which represents a 23% decrease compared to 2015. The major factor contributing to this shortfall was launch delays. However, SpaceWorks' 2017 forecast reflects that despite technical challenges and a continuing backlog of satellites awaiting launch (including SpaceX's Falcon family and Spaceflight Industries' SHERPA), the market potential for the industry remains high. According to Figure 1.1,

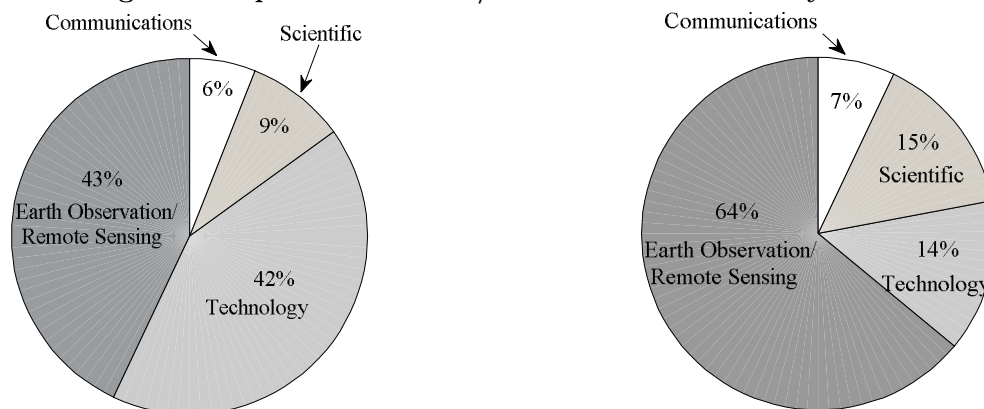
about 2400 nano/microsatellites will require a launch during 2017-2023. Sector-wise, referring to Figure 1.2, commercial satellites will be the majority of the future small satellites deployed in the space as the government projects shrink in the upcoming years. In addition, academic small satellites deployment will be a record in 2017 as industrious are working through their current launch backlog. Figure 1.3 shows that the Earth observation and remote sensing applications of small satellites are growing rapidly.



**Figure 1.1** SpaceWorks small satellites launch historical, current, and future projections data



**Figure 1.2** SpaceWorks nano/microsatellites trends by sector



**Figure 1.3** SpaceWorks nano/microsatellite trends by purpose: (Left) 2009-2016 (Right) 2017-2019

This growth requires advanced monitoring systems that can compensate for the lack of redundancy in hardware due to smaller design and mass restrictions. Table 1.2 summarizes

some of the recent momentum/reaction wheel failures in space programs emphasizing on the importance of proper monitoring systems in place to avoid such failures. In 2005, two of the Japan Aerospace Exploration Agency (JAXA) Hayabusa's RWs failed, causing the mission plan to be altered. The failed reaction wheels were manufactured by ITHACO Space Systems, later acquired by Goodrich. On 12 July 2007, NASA's Far Ultraviolet Spectroscopic Explorer (FUSE) final RW failed and caused mission termination. One of the NASA's Thermosphere, Ionosphere, Mesosphere Energetics and Dynamics (TIMED) RWs failed in 2007. Another case was observed in NASA's Dawn spacecraft where it suffered two RW failures in 2010 and 2012. Kepler's RWs failed on 14 July 2012 and 11 May 2013, stopping its primary mission. Another telescope that will rely on the functionality of the RWs is the Transiting Exoplanet Survey Satellite (TESS) [11], which is set to launch in 2017 searching for Earth-like planets. TESS is part of NASA's Explorer program led by a team from Massachusetts Institute of Technology (MIT). Reaction wheels are essential parts of a telescope as they aim the telescope with pinpoint accuracy and allow it to focus on a small desired target for long periods. Larger satellites can afford to carry redundant wheels in case of faulty behavior or failure of main RW units. For example, Swift Gamma-Ray Burst Explorer (Swift) employs six RWs for rapid slewing, and the James Webb Space Telescope (JWST) used a similar RW configuration. However, for smaller satellites including nano/microsatellites, hardware redundancy cannot be accommodated due to limited space and power budget.

**Table 1.2 Summary of on-orbit failure of MW/RW/CMG [12]**

Spacecraft	Cause of anomaly	Year
Radarsat-1	2 pitch MWs failed	1999, 2002
ISS	1 CMG failed on June 8	2002
Hayabusa	X and Y axis RW failed on July 31 and October 2, respectively	2005
FUSE	Final RW required for accurate pointing failed	2007
TIMED	Single RW failure	2007
Dawn	Two RW failures due to excessive friction development	2010, 2012
Kepler	Two RW failures disabled accurate positioning/data collection	2012, 2013

This is where analytical redundancy comes in. CBM and PHM can provide means to detect, isolate, and forecast future failures based on available historical data, a mathematical model of the system, and the combination of the two. Therefore, in this study, we are interested in exploiting the possibility of providing a framework for prognostics and health monitoring of small satellites. This seems feasible when hardware redundancy is not an option, but rather analytical redundancy is available to compensate for the lack of physical space and reliability improvements. In this thesis, we investigate three main components of a CBM and PHM framework, namely fault detection, fault isolation, and fault prognosis. The goal

is to develop, implement, and assess the performance of fault detection, fault isolation, and fault prognosis algorithms for individual RWs/CMGs, assembly of three or four RWs/CMGs, and RWs/CMGs as part of satellite attitude control system (ACS). As discussed earlier, early detection, isolation, and identification of fault onboard satellite can help eliminate early mission abortions, unexpected extra costs, and finally the development of fail-safe satellites.

## 1.2 Satellite Attitude Control System

The ACS is one of the satellite subsystems with main functionality of controlling its attitude, which processes telecommands and telemetry data. It also obtains measurements from a set of sensors on a continuous basis and uses them to calculate control signals needed for a set of actuators. The schematic for a closed-loop controlled satellite attitude and orbit system is shown in Figure 1.4. In the following sections, each block of this diagram will be defined and discussed as needed.

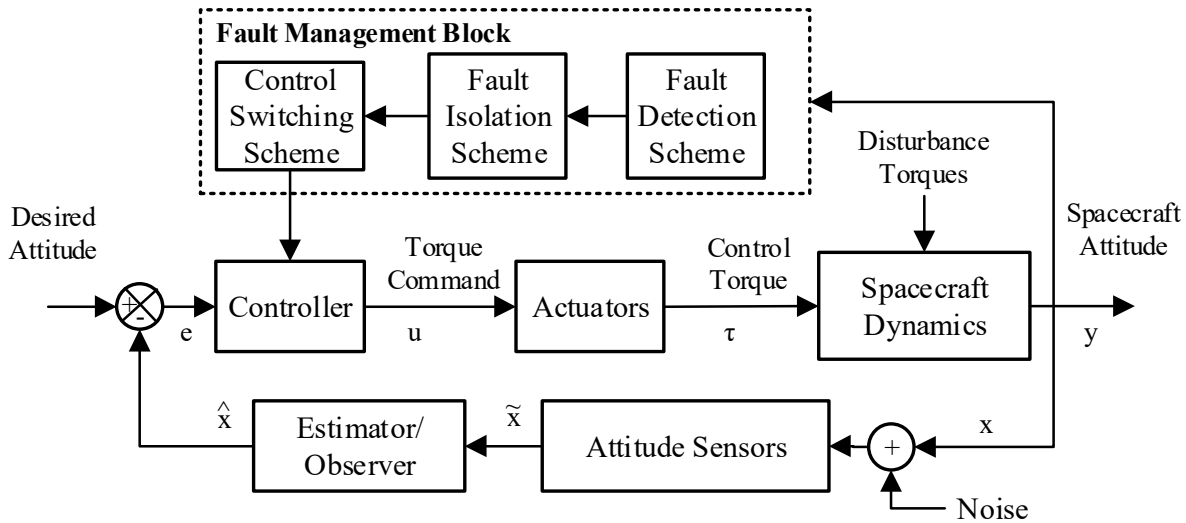


Figure 1.4 Schematic of the fault tolerant satellite attitude control system

The ACS is usually responsible for the following functions [13]:

**Attitude Control Function:** for any satellite, a nominal attitude is defined, which it should ideally maintain. It may be either calculated on the ground and sent to the satellite or computed by ACS onboard. Actuators help maintain the nominal attitude under a closed loop and autonomous control system.

**Telecommand Processing:** in order to process information received from the ground, telecommands are sent to the ACS in an asynchronous manner.

**Telemetry Processing:** the ACS generates telemetry and sends it to the central computer.

**Failure Detection and Isolation:** the ACS is responsible for detecting anomalies

and isolating their cause. Typical checks include:

1. ***Attitude Anomaly Detection (AAD)***, which ensures that the nominal attitude is maintained.
2. ***Single Sensor Consistency Check***, which verifies that the sensor outputs are feasible and within operational range.
3. ***Multi-Sensor Consistency Check***, which cross-checks multiple sensors outputs for the same measurement to detect and, if possible, isolate failures.
4. ***Watchdog Alarm***, which is triggered if a ‘watchdog event’ is not captured by its dedicated hardware (the ‘watchdog’) within a certain period.

**Failure Recovery:** recovery actions include ACS software reset, notifying the ground and take no further actions, changing the operational mode, reconfiguring a sensor or an actuator.

**Reconfigurations:** due to redundancy requirements, the ACS only uses a subset of its available units at any given time. When the ACS detects a failure, attempting to recover by performing a reconfiguration is necessary to exclude the faulty unit. The types of reconfigurations include:

1. ***Unit Reconfiguration:*** if the faulty unit is identified, a switchover to the redundant copy of the faulty unit is performed.
2. ***Subsystem Reconfiguration:*** if fault isolation fails, all subsystem units are switched over to their redundant copies followed by a re-initialization of the subsystem.

**Manoeuvre Execution:** in order to achieve a pre-defined goal, ACS is responsible for performing a sequence of actions at the certain time.

### 1.2.1 ACS Units

The main ACS units consist of its *sensors* and *actuators*. The sensors are used to collect measurements while the actuators deploy torques and forces. There are two types of attitude sensors: one with no internal processors (*Passive*) and the other one with an internal processor (*Active*). The most common types of passive sensors are shown in Figure 1.5 [13]:

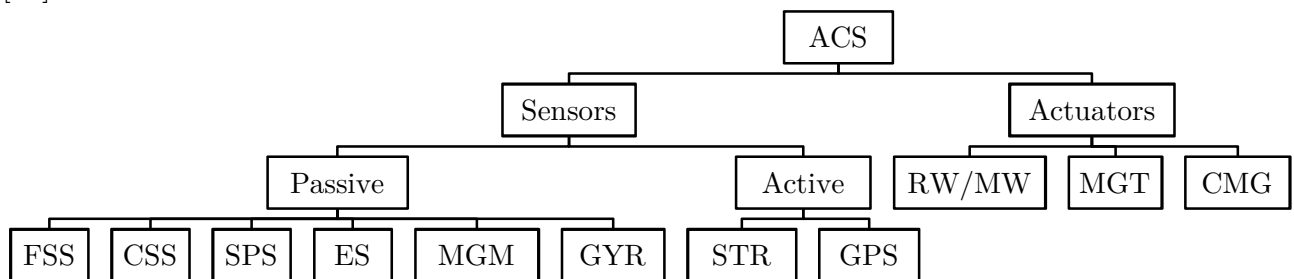


Figure 1.5 Satellite attitude control system units

where *Fine Sun Sensor (FSS)* and *Coarse Sun Sensor (CSS)* measure the direction of the sun line with high and low accuracy, accuracy. *Sun Presence Sensor (SPS)* outputs “sun present” and “sun not present”. *Earth Sensor (ES)* measures the earth direction. *Magnetometers (MGM)* measure the direction of the earth’s magnetic field. *Gyroscope (GYR)* measures satellite’s inertial angular rate. *Star Sensor (STR)* measures positions of several stars. *GPS Receiver* provides position determination. *Reaction wheel (RW)* provides a torque by accelerating or decelerating its flywheel. *Momentum wheel (MW)* provides a constant torque using its constant speed rotating wheel. *Magnetorquers (MGT)* generate control torques by interact with the Earth’s magnetic field. And finally, *Control Moment Gyros (CMG)* are gimballed RWs.

### 1.2.2 Actuators

The most common actuator technologies for small satellites are RWs/MWs, thrusters, and magnetorquers, while CMGs are found in larger satellites. An RW consists of a flywheel attached to an electric motor. Minimum of three RWs are required for fully actuated attitude control.

RWs can be used to perform maneuvers or to keep the system stable under disturbances by accelerating or decelerating its wheel. An MW, on the other hand, is always spinning at a very high speed helping stabilization of the spacecraft, making it resistant to change its attitude. An MGT usually consists of a wire coil with a ferrite core. When a current passes through the coil, a magnetic field is created. Multiple MGTs can form a magnetic dipole that counteracts the Earth's magnetic field and provides two-axis pointing of a spacecraft. MGTs are usually employed along with RWs for full three-axis attitude control. Finally, CMG is a momentum exchange device similar to an RW and MW. A CMG unit consists of a flywheel that is gimballed about one, two, or three of its axes. Gyroscopic torques are generated as the angular momentum vector is rotated about axes perpendicular to the flywheel spin axis. International Space Station (ISS) uses CMGs, Hubble has MWs and Kepler employs RWs.

Depending on mission requirements one type of actuator may deem more appropriate. In particular, the required degree of pointing accuracy is a primary factor for selecting actuators. MGTs provide the lowest pointing accuracy because of the time-varying nature of the Earth's magnetic field and their inability to provide control about more than two axes. They are usually used in conjunction with RWs and CMGs for momentum dumping. For a small satellite, MGTs are small enough to satisfy the mass/power/volume constraints; hence, making them a popular choice. RWs provide substantially improved pointing accuracy and agility relative to MGTs. However, power consumption and mass tend to be larger. CMGs can provide even more accurate and agile pointing capabilities due to their torque amplification characteristics and gyroscopic

stabilization at a price of larger units, more complex dynamics, and power consumption.

Based on a study by [8], more than half of the failures (54%) in ACS are mechanical in nature, while a small portion is electrical and although there is a large portion of unknown component responsible for these failures, gyroscopes, momentum wheels and thrusters attribute to about 50% of ACS failures.

### 1.3 Fault Detection, Isolation, and Prognosis

Faults and failures are inevitable when dealing with mechanical parts and if occurred, can add to the costs of a mission or lead it to termination. The most probable area on spacecraft for a fault to occur is in its ACS, more particularly its actuators. Most actuators consist of moving mechanical parts. These components are subject to unanticipated faults/failures like cold solder joint, minute particles, or massive temperature fluctuations [14]. Traditionally, faults and failures were dealt with hardware redundancy in larger satellites. However, recently due to budget constraints, hardware redundancy is limited. Therefore, in the case of a fault occurrence, numerical methods can be utilized to diagnose and correct the problem(s). If the satellite can detect, isolate, and identify faults, the operators can take remedial actions remotely, or the satellite could autonomously correct itself. Eventually, the aim is to achieve at least partial completion of a mission in the presence of faults/failures. As a result, fault detection, isolation, and identification (FDI) techniques have been developed to monitor/estimate some of these parameters. Furthermore, prognostics techniques are developed to for estimating such parameter's remaining useful life (RUL) and mean time to failure (MTTF).

#### 1.3.1 Fault Classification

A “fault” is used to denote an unpredicted change of system behavior that results in a degradation of performance or prevents any semblance of normal operation of the system. Actuator faults can be categorized as (1) additive, and (2) multiplicative, based on their modeling and addition to the system (Figure 1.6). Additive faults are typically observed in ACS actuators RWs. Changes in friction between stator and rotor resulted from aging or temperature can lead to discrepancies between the generated and commanded torques. When malfunctioning of the actuator leads to continuous addition or lack of torque or force, such faults can be modeled as an abrupt change in the nominal control action and categorized as multiplicative faults.

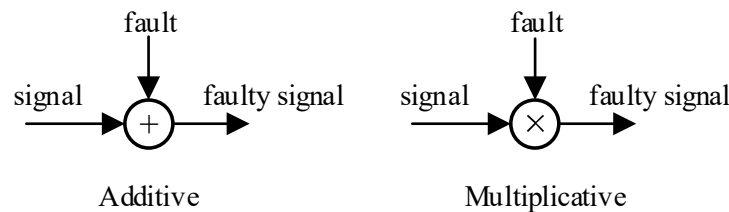
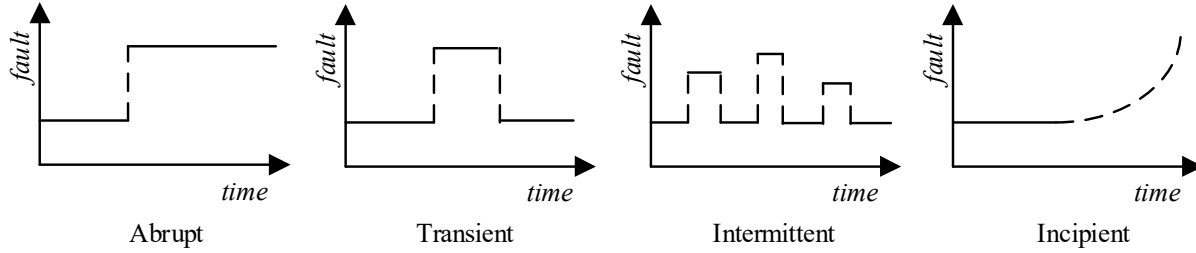


Figure 1.6 Types of faults based on addition to the system

Faults can also be categorized based on their form or timely characteristics as (1) abrupt, (2) transient, (3) intermittent, and (4) incipient (Figure 1.7). Abrupt faults occur instantaneously due to hardware damage. This can lead to a complete failure of the unit unless the unit is repaired or replaced. Transient faults represent a temporary malfunction in the unit, and if repeated they are categorized as intermittent. Such faults may or may not be crucial to the performance of the unit depending on the time of occurrence. Finally, incipient faults are gradual variations in the actuator performance and typically appear with the aging of the unit. Enduring incipient faults is crucial in ACS as their propagation over time can lead to complete system failure. Fault prognostics is a useful asset in forecasting fault propagation and estimating unit's RUL and MTTF.



**Figure 1.7** Types of faults based on time characteristics

The nature of failures in RW and their consequential effects on the system performance can be categorized under four major fault scenarios [15]:

1. *Failure to respond to control signal*: due to faulty drive electronics drive motor, and power supply, the wheel fails to respond to any control signals and decelerates slowly or holds its position. This can cause zero or undesired torque outputs from the wheel.
2. *Decreased reaction torque*: caused by the friction between rotor and stator, marginal failure of the bearings, or decreased motor torque and current drive can lead to less than desired/commanded reaction torque from the wheel.
3. *Increased bias torque*: incipient faults, caused by friction due to aging, or time-varying temperature, etc. can lead to undesired acceleration or deceleration of the wheel. Consequently, the wheel would generate a bias torque even in the event of zero commanded torque.
4. *Continuous generation of reaction torque*: intermittent time-varying faults in the motor current or fault in the bus voltage can lead to continuous increase/decrease in the wheel speed. This can lead to undesired reaction torques generated by the wheel regardless of the commanded torque by the controller.

Furthermore, CMGs can face particular types of problems including singularities, saturation, anti-parallel alignment, and hitting the gimbal stops.



### 1.3.2 Fault Diagnosis and Prognosis Cycle

A comprehensive health management system integrates readings from the monitoring sensor through the reasoning software, which offers optimal use of maintenance assets. This strategy relies intensely on the ability to (1) predict the inception of forthcoming faults/failures or RUL of critical components accurately and (2) isolate the root cause of such anomalies efficiently. This can lead to the main objective of the health management system, which is maximizing system availability and minimizing its downtime through more effective troubleshooting.

The two main steps of CBM/PHM frameworks are (1) extraction of trends from measurements, (2) interpretation of the extracted trends [16]. Extensive work has been done on trend extraction and representation [17]. Primary components of the interpretation schemes include (1) detection (2) isolation and identification (3) prognostics (Figure 1.8). In FDI, fault detection refers to determining fault presence in a system, fault isolation determines the type, location, and time of detection of a fault, and comes after fault detection. Fault identification determines size and behavior of a fault proceeding fault occurrence and comes after fault isolation. Fault diagnosis refers to fault isolation, and identification tasks joined. Finally, fault prognosis refers to determining the RUL and MTTF of a component in faulty conditions, which can help with scheduling maintenance actions taken onward.

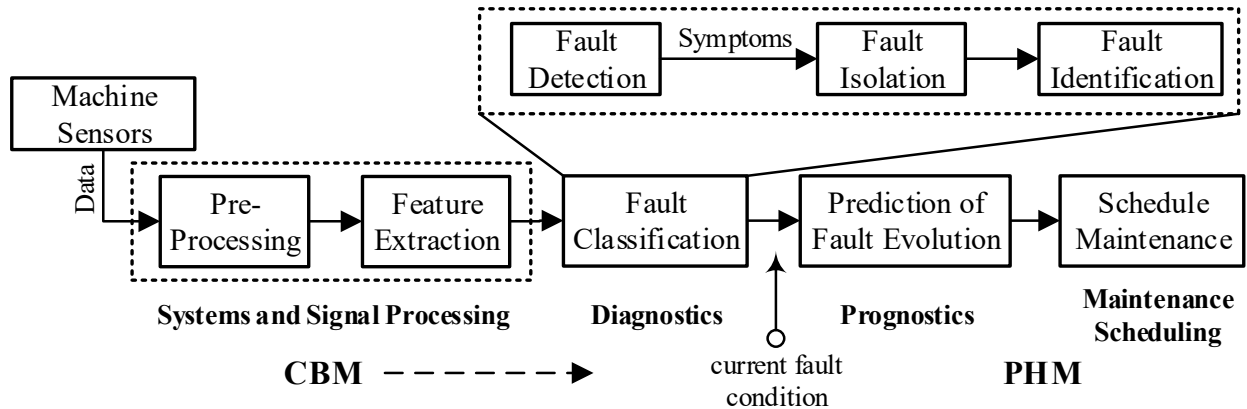


Figure 1.8 The CBM/PHM cycle

The focus of this thesis is on the interpretation of the measurement aspect of the CBM/PHM process. Particularly, we will look at fault detection, fault isolation, and fault prognosis modules of the process. In each module, based on the literature reviewed, a new approach is proposed to handle the problem at hand. Next, the proposed methodology is implemented in a simulation for a high-fidelity system to evaluate its performance and compare its results with some of the existing methods. In the following section, a literature review on the above-mentioned modules is provided to outline the problems addressed in this thesis.

## 1.4 Literature Review

Recent developments in micro and nanotechnology have changed the focus of design and manufacturing in the aerospace industry toward smaller, lower-cost spacecraft [18]. The smaller the satellites become, there is less space to accommodate larger high accuracy or redundant hardware. Therefore, ACS actuator failure is still one of the most common causes of spacecraft operational deficiencies. Therefore, the need for analytical redundancy measures to compensate for the lack of hardware redundancy and lower precision is paramount. This section places the objectives of this thesis in perspective with the state-of-the-art fault detection, isolation, and prognosis techniques available in the literature for various purposes including spacecraft attitude control using reaction wheels.

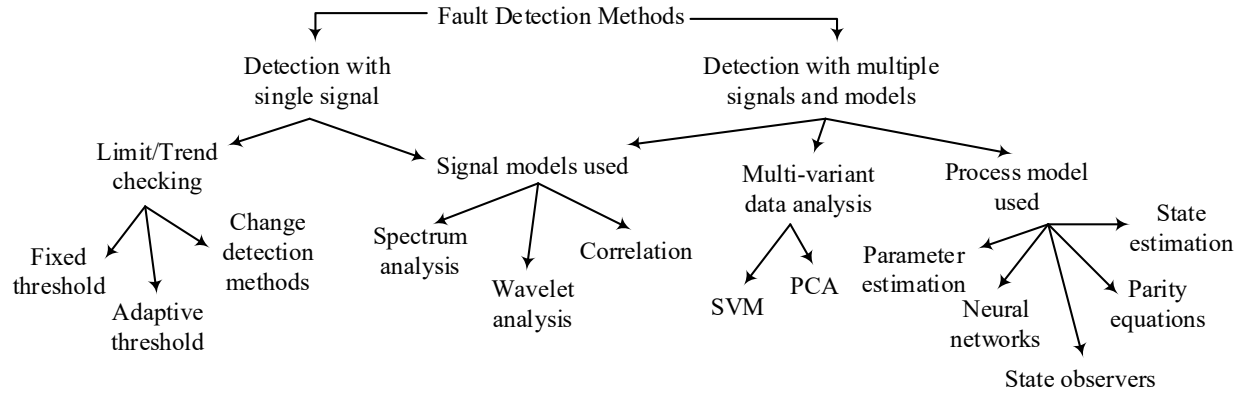
### 1.4.1 Fault Detection

Several researchers have examined the problem of fault detection, isolation, and identification [19]–[22]. Gertler [19] has surveyed model-based methods for fault detection and concludes that major quality issues for failure detection algorithms are isolability, sensitivity, and robustness. Marzat et al. [20] have reviewed model-based fault diagnosis approaches for aerospace systems; these approaches include expert systems [23], neural networks [24]–[26], support vector machine (SVM) [27]–[29], principal component analysis (PCA) [30]–[33], parameter estimation [34], [35], Kalman filters (KF) [36]–[38], and unscented Kalman filters (UKF) [39]–[43] and Cubature Kalman Filter (CKF) [44]. More recently, Gao et al. [21], [22] have comprehensively reviewed fault diagnosis approaches and their applications from the model and signal-based perspectives. Tidriri et al. [45] have also investigated features of different model-based and data-driven fault diagnosis and health monitoring individually as well as hybrid approaches that incorporate advantages of each. Figure 1.9 illustrates major fault detection methods as discussed in [46].

The reviewed literature suggests advantages of using Kalman filters as small false alarm rate, short detection delay, robustness to model uncertainty, and isolation of simultaneous faults with the only limitation being restrictive Gaussian noise assumption. In addition, parameter estimation using Kalman filters is considered as an effective approach for structural damage detection with the limitation of non-applicability for on-line identification due to time delays. The following categories can be mentioned in the field of fault detection and diagnosis:

- 1) Mathematical model-based: which is the main approach employed in this study, consists of using a mathematical model of the system with ideal behavior in a fault-free mode. The outputs of the model are compared to the measured states and when noticeable deviations are observed a fault is annunciated. Boskovic et al. [47] designed thruster FDI based on Multiple Models while Wu and Saif [30] developed a nonlinear iterative neuron proportional, integral, and derivative (INPID) observer-based FDI. Tudoroiu and Khorasani

[48] developed FDI for RW using multiple interacting models while Azarnoush and Khorasani [49] applied linear and nonlinear observers for the same purpose. The main deficiency of adaptive-observer-based FDI methods is their inability to detect time-varying faults. An iterative learning observer (ILO) was proposed by Chen and Saif [50] to overcome this limitation.



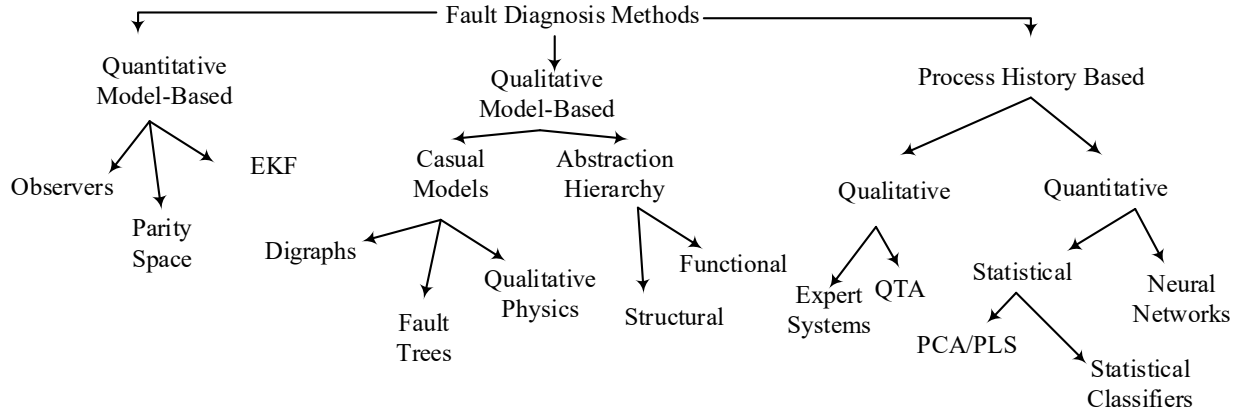
**Figure 1.9 Major fault detection methods [46]**

2) Learning-Based: use time-histories of the system input/output data to learn its mathematical model and generate residuals. Li and Khorasani [51] used a dynamic neural network to learn the non-linear RW dynamics and generate residuals. Later, a fault identification function was developed using a mathematical model of the system along with self-learning computational intelligence techniques [52]. The algorithm was only able to identify one fault at a time, and its application to different systems required redesign and re-training of the algorithm's intelligent portion.

3) Expert System-Based (fuzzy rule-based): employs a set of “*if the*” rules that are based on expert knowledge of the system or otherwise neural networks. Neural networks can well perform trend analysis and/or feature extraction to determine the operating state of a system [53], [54]. Clustering algorithms based on Bayesian classification rules have also been developed to classify data into clusters representing a particular operating condition of the system [55].

### 1.4.2 Fault Isolation

There are two main categories of fault isolation approaches [56], [57]: model-based and data-driven. When the precise mathematical model of the system under study is not available, data-driven approaches are more suitable. Figure 1.10 illustrates major fault diagnosis methods available in the literature.



**Figure 1.10 Major fault diagnosis methods [46]**

Various approaches including neural networks [58], fuzzy logic [59], [60], and clustering algorithms [61], [62], as well as hierarchical approaches, have been studied to enhance isolation scheme [59], [63], [64]. However, the main issue with data-driven approaches is the requirement for historical data as well as computational complications. On the other hand, in model-based approaches, residual is generated based on the continuous comparison of the nominal model with current system outputs [19], [20]. An extensive review is published in a three-part series by Venkatasubramanian et al. [56], [57]. Part I, II, and III of this series provide a review on quantitative model-based, qualitative model-based, and history-based approaches, respectively. Parameter estimation methods [34], [35], [41], [65], provide fault detection and isolation simultaneously since they provide time, source, and size of fault by estimating parameters of the system. The limitations of parameter estimation methods are extensive computational resources as well as noise distribution for Kalman-based filters (KF) [66], and speed of convergence for particle-based filters (PF) [67], [68]. Multiple model methods (MM) [36], [48], [69]–[73] use multiple models of the system, one with nominal performance, and others covering a range of faulty conditions. At any given time, the current state of the system is a combination of the output provided by all models. Cheng et al. [74] have combined a model-based approach with an intelligent method to detect and isolate small faults in actuators. Tudoroiu and Khorasani [48] use a bank of interacting KFs for satellite fault diagnosis where all filters run in parallel at all times and a pre-known probability model provides the overall estimate for the satellite states by combining outputs from all individual filters. The limitations in [48] can be listed as (1) Running parallel filters at all times can consume tremendous computational resources. (2) All states are assumed measurable including RW angular velocity and motor current. (3) Only one RW assembly (standard four-wheel) is considered. (4) No disturbance is added to the satellite model, and external disturbances are ignored. (6) RW model dynamics are simplified. (7) Linear KFs are used, and system equations are linearized at various operating points, which limits applicability range for the proposed approach. Jiang and Khorasani [75] have investigated

FDI for satellite attitude control subsystem with redundant reaction wheels using sliding mode observer to detect faults and incorporating residuals post-processing to isolate the fault. The limitations of this work are (1) extensive *a priori* knowledge required to reconstruct each fault scenario for a particular RW assembly. (2) RW model is not detailed, and it is not clear if it is simplified from its original model by [76]. (3) Only one reaction wheel assembly is considered, and particular scenarios are built based on specifics of that configuration. Meskin and Khorasani [77] have worked on a similar problem to [75] where they used a modified standard geometric FDI approach in conjunction with a set of detection filters for fault isolation. The limitations of their work include: (1) RW dynamics is not considered; hence, explicit fault isolation is proposed in their work. (2) Satellite dynamics is simplified and external disturbances are considered as constant values. (3) Particular scenarios are considered based on *a priori* knowledge of the system where these scenarios may not always hold. In a more recent effort, Meskin et al. [78] have explored an MM approach towards fault isolation of jet engines where they employ multiple linearized models of the system at various operating points to build the overall model piecewise and establish an isolation scheme. The limitation of this approach is the requirement for linearization and consequently its applicability range. In an extension to [78], Pourbabaei et al. [36] have exploited a novel sensor FDI for gas turbine engines using MM approach. Their scheme is based on multiple hybrid KFs representing nonlinear mathematical system model with piecewise linear models. The limitations of their work include (1) The assumption on a minimum delay between “concurrent faults” to allow model reconfiguration, which in the proposed method in this thesis also holds. (2) The assumption on constant severity for sensor bias fault. (3) Lack of feedback from fault identification module to the FDI scheme, which cannot be used for reconfiguring the control logic.

In addition to the single satellite and actuator fault diagnosis problem, there exists multiple satellites in formation flying and multi-agent cooperative systems where only very few research results are available in the literature. Barua and Khorasani [59], [63] have developed a decomposition hierarchical framework through a Bayesian network-based model called component dependency model. Their model specifies the network parameters using domain experts’ beliefs and node fault diagnosis performance data. In another work by Azizi and Khorasani [79] a constrained-state distributed Kalman filter is proposed to estimate the states of the model for formation, which is developed from an overlapping block-diagonal state-space representation of a hierarchical large-scale system transformed into a constrained-state block-diagonal state-space model. The cooperative fault diagnosis in formation flying of spacecraft is still an open problem. Exchange of information among different units and different levels of the diagnosis system leads to new research problem where Meskin and Khorasani [80] have developed three FDI architectures including centralised, decentralised, and semi-decentralised, for a network of unmanned vehicles with relative

state measurements and compared the performance of these architectures. In another attempt, Azizi and Khorasani [81] have considered three levels for fault estimation and recovery in formation flying of satellites, namely low-level, high-level, and formation-level. The satellites are considered as one integrated module, and fault severities are estimated cooperatively by using distributed estimation filters in the formation-level and the high-level acts as a mediator between the formation-level and low-level (component) diagnosis to infuse data and isolate faults using sub-observers.

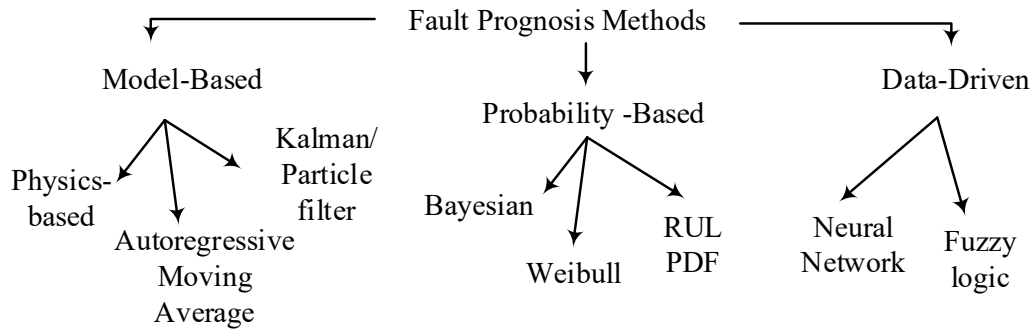
As pointed in [82], the literature on FDI of formation flying of spacecraft is limited to the studies mentioned above. However, the research on FDI of multi-agent systems can be of relevance among which, Meng, Dimarogonas, and Johansson [82] have proposed a decentralised FDI framework for two different cooperative multi-agent system models. And Shames, Teixeira, Sandberg, and Johansson [83] have constructed a bank of unknown input observers for networks of interconnected second-order linear time-invariant systems. More recently Zhang et al. [84] have tackled the problem of unknown input observer for a class of linear multi-agent systems and its application to fault estimation for a leader-follower system. Chadli et al. [85] have proposed a new approach for the design of distributed state estimation and FDI filters for a class of linear parameter-varying multi-agent systems with non-identical state-space representation of the agents, among other research work.

### 1.4.3 Fault Prognosis

The core element of a condition monitoring and assessment block in CBM and PHM is a reliable and real-time fault diagnosis and failure prognosis system [2]. Many researchers have investigated the feasibility of fault prognosis algorithms [86], [87]. Figure 1.11 illustrates major fault prognosis methods found in the literature. An et al. [86] have reviewed data-driven and physics-based prognostic methods and examined advantages and limitations of each method on a particular fatigue crack growth problem. They aimed to provide researchers with practical options for selecting proper methods based on the problem in hand. Liao and Köttig [87] have categorized the developed prognostics models in the literature into experience-based, data-driven, and physics-based and further categorized hybrid approaches into possible combinations of the other two. They argue that there is no universally accepted best model to estimate remaining useful life (RUL) due to system complexity, data availability and application constraints. Further, hybrid approaches in the literature are systematically classified in their study, and a hybrid prognostics method is proposed and applied to a battery degradation [88].

Most researchers have studied the application of fault prognosis and RUL estimation on batteries and specifically lithium-ion type [89]–[95]. Saha et al. [93] have reviewed various approaches toward estimating RUL for lithium-ion batteries including autoregressive integrated moving average (ARIMA), extended Kalman filter (EKF), relevance vector machine

(RVM), and particle filter (PF) framework [94]. Bolde et al. [89] have proposed a novel method for the adaptation of parameters in an electrochemical model of a lithium-ion battery using an unscented Kalman filter (UKF) with battery current and voltage over randomized discharge profiles. Liu et al. proposed a Gaussian process model [90] and an improved autoregressive (AR) time series model named Nonlinear Degradation Auto Regression (ND-AR) model [91] for RUL estimation of lithium-ion batteries. Olivares and Muñoz [92] have proposed a PF-based prognostic framework for estimating the state of health (SOH) and predicting the RUL of lithium-ion batteries. Zhou et al. [95] have proposed an on-line data-driven RUL estimation approach using support vector regression (SVR).



**Figure 1.11 Major fault prognosis methods [2]**

In the field of aerospace RUL prediction, and particularly satellite actuators, including RWs, there are not many studies available in the literature. Tung and Yang [96] have reviewed, summarized and classified published techniques up to 2009 in diagnosis and prognosis of rotating machinery. Ciandrini et al. [97] have reviewed various fault detection and failure prediction methods for rotating electrical machines. Their study focuses on data collection from accelerometers to acquire vibration measurements on a laboratory-scale experimental system. Jin et al. [98] have proposed a novel reliability estimation framework in terms of physics of failure, which is based on the relationship between the MW's physics performance and its failure mechanisms. Rocchi et al. [99] have proposed a fault prognostics algorithm for rotating electrical machines using recursive least square. Their experimental setup is the same as [97] using vibration data to develop a failure propagation model. Chen et al. [100] have investigated stacked denoising autoencoder (SDA) deep learning method, which is suitable for health state identifications under noisy signals. Wathiq et al. [101] propose an approach for identifying bearing defects in brushless direct current (BLDC) motors under non-stationary operating conditions. They use a recurrent neural network to detect and classify presence of bearing faults with implemented and tested simulations on data collected from an experimental setup. Ahmad et al. [102] propose a bearing degradation model from smoothed historical data using a monotonically increasing function that is globally non-linear and locally linearized. A neural network uses this model for learning and estimates the RUL of a bearing by predicting future levels of vibration acceleration. More

recently, Jingfu et al. [103] have investigated a data-driven approach combined with similitude method [104], [105] to predict the RUL of an MW. The similitude method is based on similarity; it uses available data on the life of other similar systems to predict the RUL of the current system employing weighted average. In Jingfu et al. study, a neural network method is used to model the degradation trend of the telemetry data for a momentum wheel that had failed combined with a Weight Application to Exponential Parameters (WAFT) method for the similitude forecast to estimate the RUL of the wheel under study. Other literature pertaining to the RUL prediction of MWs are mentioned in [103]. However, the author verification of the following literature was a burden as they were not available online or in English. Li [106] have used a Bayesian network to fuse failure data and performance data from ground tests of particular MWs in order to form a model for RUL prediction. Hao [107] has proposed an RUL prediction method based on RVM, which utilizes temperature trends to extrapolate MWs lifespan. Liu [107] has also used the Bayes method to fuse momentum wheel ground test information for predicting the RUL of MWs. Fan et al. [108] have analyzed mental wear, cage wear, lack of lubricant and other failure modes. They propose a testing program based on failure physics and test analysis method, which present a range of RUL prediction models for MWs.

## 1.5 Problem Statement

Drawing on the literature reviewed in the previous section. The following problems can be formulated in each module of the CBM/PHM scheme in this thesis. In each of the formulated problems, references are given to the studies where particular issues were observed.

[PROBLEM 1] When sudden changes are expected, off-line tuning of the estimators is not acceptable. Therefore, an approach that can track non-measurable system parameters is required. In a previous study [41], a parameter tracking approach based on adaptive unscented Kalman filters was proposed. However, that approach suffers from limitations to be addressed in the present work as follows: (1) The system model used in [41], neglects some of the stochastic components that could in fact adversely affect simulation results. (2) The adaptation system in [41] considers only system and measurement noise covariances, and the optimization process is done off-line. This approach suffers from a lack of agility when abrupt faults occur, even after algorithm parameters are optimized. (3) The fault scenario in [41], [109] only includes abrupt fault while in practical situations, transient, intermittent, and incipient faults also occur. (4) The tracking speed of the estimator is very slow for sudden changes in the system. (5) The proposed work in [109] considers a constant threshold for the fault annunciation with limitation of requirement for all experimental data before designing the filter parameters.



[PROBLEM 2] In the published literature, there has not been much emphasis on the integrated approach for the purpose of FDI for actuator system onboard satellites wherein the following factors are considered: (1) parallel filters do not run at all times and do not interact unlike [48]. This is in particular important to mention that the proposed method in this thesis uses a 2-step approach (detection+isolation) unlike other MM approaches, which run parallel filters at all times. (2) All system states are not assumed measurable unlike [48]. (3) More than one RW assembly is considered unlike [48], [75] where moving from orthogonal to pyramid configuration can cause remapping, dynamics difficulties, and symmetrical isolation issues. (4) External disturbances are considered in simulations unlike [48]. (5) Full RW model dynamics are considered unlike [75], [77]. (6) Complicated statistical equations are not incorporated unlike [75], [77]. (7) Extensive *a priori* knowledge about system and its faults are not required unlike [36], [75], [77], [78].

[PROBLEM 3] To the author’s knowledge, not many studies have focused on fault prognosis for rotary machines specifically in space application including RW. There is almost no published work on RW fault prognosis in space application based on system model, and there was no comparative study found on the topic as well. It should be noted that knowing the RUL of a system, which is not physically accessible in space, does not limit its usefulness in other areas of maintenance and configurability. It may be true that the cost associated with accessing a satellite in space and performing replacements can be so high that cannot justify its feasibility. However, one can argue other maintenance/remedial actions can be taken into consideration knowing how the system is deteriorating and when it is expected to fail. For example, one can change the course of orbit/attitude considering the current and estimated future behavior of certain components of the system. Furthermore, software updates can be uploaded to the C&C unit of the satellite to compensate for what is expected to be less effective or fail altogether in near future. One example of that can be going from fully actuated system to an under-actuated configuration to compensate for the lack of additional actuator while maintaining required mission specifications.

In addition, NASA and other big players in the space field, have invested and explored the feasibility of such actions. Some publications on such efforts can be found in [110]–[112]. Moreover, contracts available on certain projects dealing with this matter including Payload Integrated Health Management Systems<sup>1</sup> and others where it is explicitly mentioned: “[Sest] will have predictive prognostic models combining the individual component/subsystem (such as electro optics, refrigeration and heat

---

<sup>1</sup> <https://www.sbir.gov/sbirsearch/detail/10163>

rejection system, gimbal, electronics, *reaction wheels*, etc.) and their interactions, *identify degradation modes and possible failure mechanisms*, and impact of uncertainties in the variables controlling the payload performance and reliability.” And another example in the book *System health management: with aerospace applications* [113] where it is mentioned, “Adding *intelligent, decision making technology* into commercial and military jets. Identifies the equipment that will fail prematurely on *satellites* and launch vehicles for *replacement before launch, stops premature equipment failures that cause catastrophic failures*.” Both of which emphasize on the importance of such frameworks in place for space systems. Therefore, investigating RUL prediction for RW onboard satellite seemed an interesting researching area.

[PROBLEM 4] To the author’s knowledge, there are not many studies published on the fault detection, isolation, and identification of the CMGs and there is almost no literature on the fault prognosis of such devices. Minor studies on CMGs are mostly concerned with the fault-tolerant control and design of a controller [114]–[121] rather than establishing an FDI framework. The few minor cases where FDI framework is proposed for CMGs are Beck et al. [122] where they have used a sliding mode observer for detecting secular and oscillating disturbances. Oh and Kim [123] where a continuous second order sliding mode observer is suggested and applied for the detection of single disturbances/faults on CMGs. Choi et al. [60] have used fuzzy logic and Q-learning for FDI of a small CMG-actuated satellite. The challenge from extending application from RW to CMG lies in the complexity of modeling and certainty of the simulations. It should be noted that CMGs have more degrees of freedom (DOF) (2 or 3 DOF) compared to RWs (1 DOF) and this can affect the performance of the FDI algorithm through dynamics, symmetry, and other possible issues pertaining only to CMG as singularities, saturation, anti-parallel alignment, and hitting the gimbal stops.

## 1.6 Research Objectives

To address the problems identified above, the following objectives were set for the studies resulting in the thesis in hand:

[OBJECTIVE 1] To address the limitations in PROBLEM 1, in this work a new 2-step fault detection approach is proposed that (1) considers the missing components in the RW system model. (2) Considers states error covariance matrix when abrupt changes occur to ensure agile tracking of non-measurable system parameters. (3) Is also applicable to all fault scenarios including transient, intermittent, and incipient faults also occur. (4) proposes an adaptive threshold for the proposed modification of UKF in [109] where a constant threshold for the fault annunciation is considered.

the proposed modification in this thesis is expected to resolve the limitation of requirement for all experimental data before designing the filter parameters in [109].

[OBJECTIVE 2] To address the limitations in PROBLEM 2, a new fault 2-step isolation approach is proposed in this thesis having (1) a new hierarchical MM isolation scheme wherein multiple models are used only when a fault is detected until it is isolated unlike [48]. The novelty of the approach lies in its 2-step procedure. In step 1, fault is detected using the proposed fault detection method in this thesis, and in step 2, fault is isolated using temporary MM approach. In other MM approaches, all models run at all times and the final estimate of system states/parameters is the weighted average of all filters. In the proposed approach in this thesis, however, only one model of the system runs in parallel with the hardware, and MM are only used from inception of the fault to its isolation. This is achieved under full external disturbances unlike [48] and the assumption of partial measurements availability at each level of the isolation unlike [48]. Multiple unscented Kalman filters (UKF) are incorporated to generate estimates, one for the nominal system and others for possible faulty scenarios. The fault is detected using residual sequence followed by the first level of isolation using Bayes' probability theorem to locate the faulty unit and the second level of isolation using parameter estimation to locate the faulty parameter. (2) Incorporating two reaction wheel assemblies, namely standard four-wheel and pyramid, for simulations unlike [48], [75] that use only one configuration. (3) Incorporating full RW dynamics including all disturbance terms unlike [75], [77]. (4) Using only limited *a priori* information for initializing the simulations and running isolation scheme unlike [36], [75], [77], [78].

[OBJECTIVE 3] To address PROBLEM 3, the feasibility of implementing a hybrid UKF+PF for fault prognosis of RW onboard satellite is studied using a mathematical model of the system, its degradation and synthetic data. The proposed approach shall be capable of handling Gaussian and non-Gaussian measurement noise while providing satisfactory RUL estimations with low error.

[OBJECTIVE 4] To investigate the applicability of the proposed methods in each section, a second space system actuator, namely CMG is considered. The choice for this system was made based on the fact that there are very limited number of publications pertaining to the problem of fault detection, isolation, and identification as well as fault prognosis for such CMGs. In addition, CMGs are the next level of complexity for RWs where additional degree(s) of freedom are added to the actuator using gimbal(s).

## 1.7 Main Contributions

- [CONTRIBUTION 1] A new 2-step fault detection approach is proposed that (1) considers the missing components in the RW system model. (2) Considers states error covariance matrix when abrupt changes occur to ensure agile tracking of non-measurable system parameters. (3) Is also applicable to all fault scenarios including transient, intermittent, and incipient faults also occur. (4) Proposes an adaptive threshold for the proposed modification of UKF in [109] where a constant threshold for the fault annunciation is considered. the proposed modification in this thesis resolves the *limitation of requirement for all experimental data before designing the filter parameters* in [109] as it is adapting the threshold to generated residuals online and based on residuals statistical properties, namely mean and standard deviations.
- [CONTRIBUTION 2] A new 2-step fault isolation approach is proposed based on a hierarchical MM scheme. In step 1 of the proposed method, the fault is detected using residual sequence as detailed in CONTRIBUTION 1, followed by the first level of isolation at multi-agent/formation level to isolate the faulty spacecraft, a second level of isolation using Bayes' probability theorem with limited *a priori* information to locate the faulty actuator unit with limited measurements to the satellite attitude only and no actuator measurements; and a third level of isolation using parameter estimation to track the non-measurable faulty parameters in the faulty actuator to isolate the parameter and the root cause of the faulty behaviour in the complex formation system. It should be noted that the main advantage of the proposed method in this thesis lies in its ability to isolate the fault without having to run parallel models at all times. However, the limitation of this proposal is the inability of the proposed method in isolating a new fault occurrence "concurrent fault" while the isolation process is in progress. Therefore, if a second or third fault occurs in the system before the current fault is isolated and the model running in parallel with the hardware is updated, the isolation cannot be accomplished.
- [CONTRIBUTION 3] The proposed fault detection scheme, is implemented and numerically analyzed through computer simulations for a small CMG-actuated satellite for fault detection of faulty units under multiple in-phase and out-of-phase faults. In addition, a variation of the proposed fault isolation scheme is implemented and numerically analyzed through computer simulations for a small CMG-actuated satellite with multiple in-phase faults occurring at the same time with different severities and inception times. Since there are not many literature published on the fault detection, isolation, and identification of CMGs onboard satellites, this study can provide more insight into evaluating performance of the proposed techniques on the CMG system. Moreover, it can provide guidelines as how to extend the applicability of the proposed methods to more complex systems (from RW to CMG)

using a set of modifications in the proposed methods.

[CONTRIBUTION 4] A new hybrid UKF+PF-based approach is proposed for fault prognosis of RW onboard satellite using a mathematical model of the system and its mathematical degradation model using synthetic data to investigate the feasibility of employing UKF and PF for such applications. UKF is used to estimate non-measurable parameter of the RW, namely  $k_t$ . Next, Gaussian and non-Gaussian measurement noises are added to the estimates from the UKF to generate measurements fed into the PF to identify parameters of the degradation model for  $k_t$ . Once the degradation model parameters are identified, the model is extrapolated and its intersection with the  $k_t$  threshold is used as the end of life for that particular agent's estimates. RUL is defined as end of life minus current time and a distribution is created using PF agents to find the distribution of RUL for the faulty unit.

## 1.8 Thesis Outline

In Chapter 2 preliminaries are explained to lay out a foundation for the reader to understand the proceeding contents better. Contents of this chapter include single satellite attitude model, formation of multi-agent systems, actuator models including RW, CMG and assembly of four actuators in different configurations, namely standard four-wheel and pyramid, Kalman filters, and Particle filters. In Chapter 3, the first stage of the CBM/PHM process (Figure 1.12), namely fault detection is explained, and the proposed method is outlined, followed by performance evaluations and comparative studies. In Chapter 4, the second stage of the CBM/PHM process (Figure 1.12), namely fault isolation is explained, and the proposed method is outlined, followed by performance evaluation and discussions. In Chapter 5, the third stage of the CBM/PHM process (Figure 1.12), namely fault prognosis is described, and the proposed method is outlined, followed by performance evaluation and discussions. Finally, in Chapter 6, conclusions of the study are provided followed by potential future work and suggestions for further investigations on the topic.

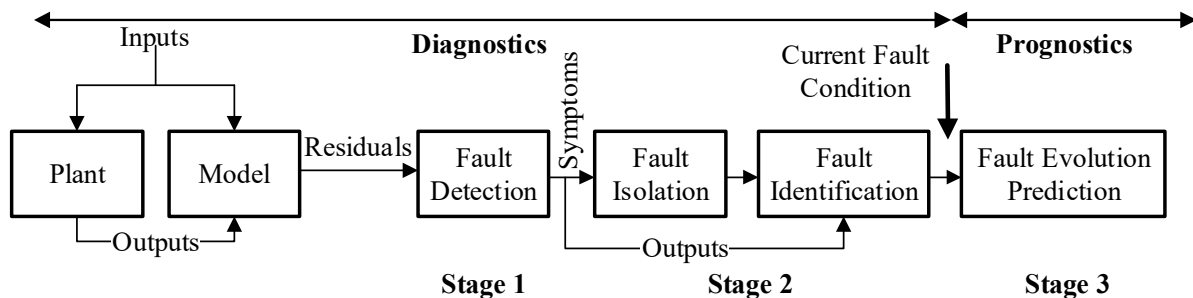


Figure 1.12 Different stage of a model-based CBM/PHM framework

## CHAPTER 2

# Preliminaries

---

In order to lay out a foundation for better understanding the contents of the following chapters, in this chapter, essential preliminaries are explained to ensure the reader can easily follow the forthcoming derivations and proposed methodologies. The contents of this chapter are organized as follows: first, the satellite attitude model, including dynamics and kinematics, is explained in this section, satellite dynamics differs based on the actuator used onboard. Therefore, dynamics equations are provided separately for satellite with RW and CMG onboard. Next, the actuator model including a high fidelity RW's mathematical model and a RW assembly (RWA) along with its mapping methods are explained followed by CMG dynamics model and its integration with satellite dynamics when used onboard satellite. Next, a brief review on Kalman filters, unscented Kalman filters, and adaptive unscented Kalman filters is provided with focus on UKF and adaptation schemes. Finally, a brief review on particle filters and various aspects of their implementation strategies along with possibility of implementation of such methods into a fault prognosis problem is presented.

### 2.1 Satellite Attitude Model

The satellite is modeled as a rigid body. The coordinate frames used for the satellite modeling are shown in Figure 2.1. An Earth-centered inertial (ECI) reference frame denoted by  $\mathfrak{I} - X_I Y_I Z_I$  has its origin at the center of the Earth, where  $Z_I$  axis passes through celestial North Pole,  $X_I$  axis points towards the vernal equinox, and  $Y_I$  axis completes the triad with right-handed rule. The *orbital frame* also known as local vertical local horizontal (LVLH) is denoted by  $\mathcal{L} - X_O Y_O Z_O$  and is fixed at the center of the spacecraft. In LVLH,  $X_O$  axis is in the direction of motion,  $Y_O$  axis in the opposite direction of the angular velocity, and  $Z_O$  axis is pointing at the Earth. True anomaly (for eccentric orbits) or reference angle  $\theta$  (for circular orbits) is measured from perigee. The body-fixed coordinate frame denoted by  $\mathcal{B} - X_B Y_B Z_B$  has its origin at the spacecraft center of mass [124]. A vector of frame  $\mathcal{L}$  relative to frame  $\mathfrak{I}$ , expressed in frame  $\mathcal{B}$  is denoted by  $v_{LI}^B$  [125].

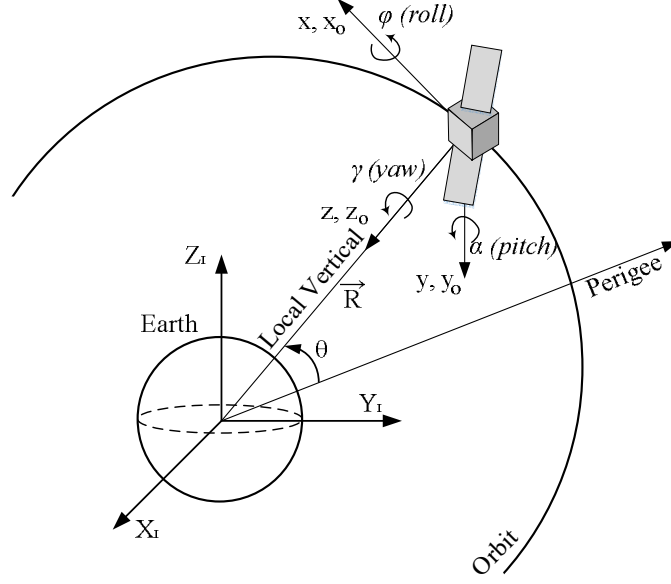


Figure 2.1 Geometry of orbit motion for rigid body spacecraft

### 2.1.1 Satellite Dynamics

The rotational equations of motion for a spacecraft actuation by momentum exchange devices such as RWs or CMGs can be described as [126]

$$\dot{H}_{BI}^B + \omega_{BI}^B \times H_{BI}^B = \tau_e \quad (2.1)$$

where  $\omega_{BI}^B$  is the angular velocity of the spacecraft relative to the inertial frame expressed in the body frame.  $\tau_e \in \mathbb{R}^{3 \times 1}$  is the external torque.  $H_{BI}^B$  is the total angular momentum of the spacecraft relative to the inertial frame expressed in the body frame given by [126]

$$H_{BI}^B = J\omega_{BI}^B + h \quad (2.2)$$

where  $J$  is defined as  $J = J_s - AJ_w A^T$  where  $J_s \in \mathbb{R}^{3 \times 3}$  is the moment of inertia of the spacecraft including the actuators.  $J_w \in \mathbb{R}^{4 \times 4} = \text{diag}([J_{w1}, J_{w2}, J_{w3}, J_{w4}])$  denotes the axial moment of inertia of each momentum exchange device.  $h$  is the total momentum provided by the momentum exchange device (RW) expressed in the spacecraft body-fixed frame [18].

$$H_{BI}^B = J\omega_{BI}^B + AH \quad (2.3)$$

where  $A \in \mathbb{R}^{3 \times 4}$  maps the influence of the actuators to the principal axes of the spacecraft.  $H$  is the momentum generated by momentum exchange devices in their respective axes. Combining Eqs. (2.1)-(2.3), we can obtain [126]

$$(J\dot{\omega}_{BI}^B + A\dot{H} + \dot{A}H) + \omega_{BI}^B \times (J\omega_{BI}^B + AH) = \tau_e \quad (2.4)$$

If the control torque applied by the momentum exchange device is denoted by  $\tau_c$  we have [126]

$$J\dot{\omega}_{BI}^B + \omega_{BI}^B \times J\omega_{BI}^B = -\tau_c + \tau_e \quad (2.5)$$

$$A\dot{H} + \dot{A}H + \omega_{BI}^B \times AH = \tau_c \quad (2.6)$$

The terms  $\dot{A}$  and  $\dot{H}$  appear because depending on system and variation of the elements, each can contribute to the overall momentum. For example, if RWs are used,  $\dot{A} = \mathbf{0}$  because there are no varying terms in  $A$ . On the other hand, if CMGs are employed,  $\dot{H} = \mathbf{0}$  because flywheel speed is constant and there are no variations with time wheel speed and only gimbaling causes the momentum exchange. If RWs are used as actuators,  $H_w$  is used instead of  $H$ , which is expressed as [126]

$$H_w = J_w(\Omega + A_{RW}^T \omega_{BI}^B) \quad (2.7)$$

where  $\Omega \in \mathbb{R}^{4 \times 1}$  denotes the axial angular velocity of the reaction wheels. Substituting  $H_w$  in Eq. (2.3) gives [126]

$$H_{BI}^B = J\omega_{BI}^B + A_{RW}J_w(\Omega + A_{RW}^T \omega_{BI}^B) \quad (2.8)$$

Putting this back into Eq. (2.1) gives [126]

$$\begin{aligned} J\dot{\omega}_{BI}^B + A_{RW}J_w(\dot{\Omega} + A_{RW}^T \dot{\omega}_{BI}^B) + \omega_{BI}^B \times (J\omega_{BI}^B + A_{RW}J_w(\Omega + A_{RW}^T \omega_{BI}^B)) \\ = \tau_e \end{aligned} \quad (2.9)$$

Expanding the terms and using the definitions provided earlier, a fully actuated rigid body spacecraft controlled by reaction wheels with internal and external torques acting on it can be formulated as [127]

$$J\dot{\omega}_{BI}^B = -\omega_{BI}^B \times (J_s\omega_{BI}^B + A_{RW}J_w\Omega) - A_{RW}\tau_{RW} + \tau_e \quad (2.10)$$

where  $\tau_{RW}$  is the torque generated by RWs and can be formulated as [127]

$$\tau_{RW} = \dot{H}_w = J_w(\dot{\Omega} + A_{RW}^T \dot{\omega}_{BI}^B) \quad (2.11)$$

In this study, reaction wheels angular velocities ( $\Omega$ ) are calculated from RW dynamics using Eq. (2.31) in Section 2.3.1 and are fed into satellite dynamics and kinematics. If CMGs are used as actuators, from Eq. (2.1) and Eq. (2.2) we have [120]

$$J\dot{\omega}_{BI}^B = -\omega_{BI}^B \times (J_s\omega_{BI}^B + h_{CMG}) - \dot{h}_{CMG} + \tau_e \quad (2.12)$$

where  $h_{CMG}$  is defined in Eq. (2.38)  $\dot{h}_{cmg}$  is defined in Eq. (2.39).

### 2.1.2 Attitude Kinematics

The kinematic equations for the spacecraft using quaternions can be formulated as [120]

$$\begin{bmatrix} \dot{q}_v \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_4 I + q_v^\times \\ -q_v^T \end{bmatrix} \omega_{BL}^B \quad (2.13)$$

where  $\bar{q} = [\bar{e} \sin(\frac{\Phi}{2}), \cos(\frac{\Phi}{2})]^T = \begin{bmatrix} q_v \\ q_4 \end{bmatrix}$  is unit quaternion,  $\Phi$  denotes the principal angle,  $\bar{e} = [e_1, e_2, e_3]^T$  is the principal axis from Euler's theorem ( $e_1^2 + e_2^2 + e_3^2 = 1$ ).  $q_4 \in \mathbb{R}$  and  $q_v \in \mathbb{R}^{3 \times 1} = [q_1, q_2, q_3]^T$  denote the Euler parameters representing the spacecraft body frame orientation with respect to the orbital frame where  $q_v^T q_v + q_4^2 = 1$ .  $I \in \mathbb{R}^{3 \times 3}$  is the identity



matrix and  $q_v^\times$  is the skew-symmetric matrix of the quaternion vector given by [120]

$$q_v^\times = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \quad (2.14)$$

The angular velocity of the body frame with respect to the inertial frame can be written as [120]

$$\omega_{BI}^B = \omega_{BL}^B + \omega_{LI}^B \quad (2.15)$$

The transformation matrix  $C_L^B$  is used to describe the orientation of the spacecraft body frame  $\mathcal{B}$  with respect to the orbital frame  $\mathcal{L}$ . It can be formulated in terms of Euler parameters as follows [120]

$$C_L^B = (q_4^2 - q_v^T q_v)I + 2q_v q_v^T - 2q_4 q_v^\times \quad (2.16)$$

The orbital angular velocity expressed in the body frame  $\omega_{OI}^B$  (i.e. the speed the orbit is rotating with the Earth) can be defined as [120]

$$\omega_{LI}^B = C_L^B [0 \quad -\omega_0 \quad 0]^T \quad (2.17)$$

where  $\omega_0 = \dot{\theta}$  is the orbital frame angular velocity. This value is equal to  $\dot{\theta} = \sqrt{\mu_e/R_c^3}$  for circular orbits where  $\mu_e$  is the gravitational constant parameter of the Earth equal to  $398600 \text{ km}^3/\text{s}^2$  and  $R_c$  is the spacecraft distance from the center of the Earth. This will be equal to  $R_c = R_E + h_s$  where  $R_E$  is the average Earth radius equal to  $6378 \text{ km}$  and  $h$  is the altitude of the spacecraft from the surface of the Earth. In this study  $h_s = 500 \text{ km}$ .

Equation (2.10) is with respect to the inertial frame reference  $\mathfrak{I}$  because the Euler' equation holds with respect to the inertial frame. However, in orbital satellite motion, it is desired to work with the spacecraft body angular velocity relative to the orbital frame  $\mathcal{L}$  expressed in the body frame  $\dot{\omega}_{BL}^B$ . From theory we have [128]

$$\dot{C}_L^B = -[\omega_{BL}^B]^\times C_L^B \quad (2.18)$$

Taking a time derivative of Eq. (2.15) and applying it to Eq. (2.18) we have [128]

$$\dot{\omega}_{BI}^B = \dot{\omega}_{BL}^B - [\omega_{BL}^B]^\times \omega_{LI}^B \quad (2.19)$$

Since we are interested in  $\dot{\omega}_{BL}^B$  we have [128]

$$\dot{\omega}_{BL}^B = \dot{\omega}_{BI}^B + [\omega_{BL}^B]^\times \omega_{LI}^B \quad (2.20)$$

The external torque is assumed to include gravity gradient torque ( $\tau_{gg}$ ), solar radiation pressure ( $\tau_{srp}$ ), aerodynamic torque ( $\tau_{aero}$ ), and other disturbances ( $\tau_d$ ).  $\tau_e = \tau_{gg} + \tau_{srp} + \tau_{aero} + \tau_d$ . These disturbances can be defined as [5], [18, p. 112], [129], [130]

$$\tau_{gg} = 3\omega_0^2 c_3^\times J_s c_3; c_3 = C_O^B [0 \quad 0 \quad 1]^T \quad (2.21)$$

$$\tau_{srp} = \begin{bmatrix} T_{sx} \\ T_{sy} \\ T_{sz} \end{bmatrix} = S_f \begin{bmatrix} (2 \times 10^{-5})[1 - 2\sin(\omega_0 t)] \\ (1 \times 10^{-3})\cos(\omega_0 t) \\ -(5 \times 10^{-5})\cos(\omega_0 t) \end{bmatrix} \quad (2.22)$$

$$\tau_{aero} = \begin{bmatrix} T_{ax} \\ T_{ay} \\ T_{az} \end{bmatrix} = 1.36A_f \begin{bmatrix} 1 + \sin(\omega_0 t) + \frac{1}{2}\sin(2\omega_0 t) \\ -(5 \times 10^2)[4 + 2\sin(\omega_0 t) + \frac{1}{2}\sin(2\omega_0 t)] \\ -[1 + \sin(\omega_0 t) + \frac{1}{2}\sin(2\omega_0 t)] \end{bmatrix} \quad (2.23)$$

$$\tau_d = \left( \frac{1}{2} + \|\omega_{BO}^B\|^2 \right) \begin{bmatrix} \sin(\omega_0 t) \\ \cos(\omega_0 t) \\ \cos(\omega_0 t) \end{bmatrix} \quad (2.24)$$

where positive scaling factors  $S_f$  and  $A_f$  are calculated based on worst expected disturbance torques in low earth orbit (LEO) and are equal to  $S_f = 1.70 \times 10^{-6}$ ,  $A_f = 1.0 \times 10^{-11}$  for RyeSat (pico-satellite) [5].

## 2.2 Formation Flying

There are several advantages in employing a group of satellites in formation flying to accomplish a mission compared to employing a single-satellite platform. The ability to avoid a single point of failure and achieving a more robust mission is one of the most important advantages of the formation flying systems. As discussed earlier, the multiple spacecraft approach will also cost less due to smaller launch vehicles and development costs associated with it. There are other advantages to this approach including higher reliability and redundancy, higher resolution, simpler design, faster build times [131], [132].

It is well known that efficiency and reliability of the formation mission can be jeopardized due to occurrence of faults in the actuators of the satellites. Therefore, autonomous real-time and online fault detection and isolation schemes can help alleviate this problem. Diagnosing faults in individual satellite's subsystems before they lead to severe damage and catastrophic failures can save a lot of costs and save missions and the entire networked formation flying system. There are three major FDI architectures defined in [82], namely decentralized, semi-decentralized, and centralized, which will be discussed briefly as follows. However, only centralized architecture is employed in this thesis as an extension from single to multiple satellite systems.

There are five architectures for formation flying control proposed in the literature. These include Multiple Input-Multiple Output, Leader/Follower, Virtual Structure, Cyclic, and Behavioral based [133]. In this thesis, formation flying of spacecraft is considered to be accomplished by a pre-designed set of inputs. Therefore, the controller module in this formation is not detailed further in this thesis. Furthermore, there are various types of network

topologies including Ring Mesh, Star, Line, Fully Connected, Tree, and Bus. In this thesis, since it is an extension of the system level (single satellite) to multi-system level (formation) and the objective is to merely provide a proof of concept for the future studies on more detailed investigations of fault tolerant formation flying, a bidirectional fully connected topology is employed. This can be argued as a restrictive and unrealistic choice, however, the proof-of-concept was meant to be evaluated in this thesis at this level and further, more detailed investigations are left for future studies.

### 2.2.1 Decentralized FDI architecture

In the decentralized architecture [134] (Figure 2.2), there is a dedicated FDI unit for each spacecraft with no communications among spacecraft for the purpose of accomplishing the FDI goal. Therefore, the  $i$ th spacecraft would detect faults in its actuators solely based on the internal information it has of its input, output and control signals.

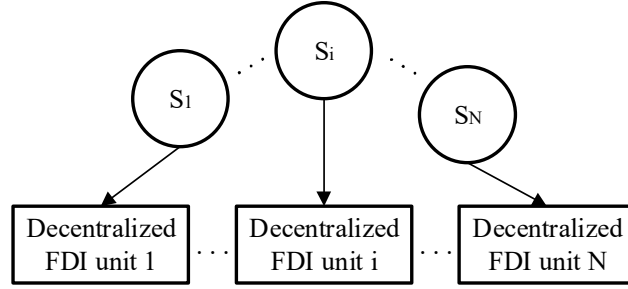


Figure 2.2 Decentralized FDI architecture [134]

The mathematical model of the  $i$ th spacecraft for this structure can be formulated as [134]

$$\begin{aligned}\dot{x}_i &= f(x_i(t), \theta_i(t), u_i(t)) + w_i(t) \\ y_i(t) &= g(x_i(t)) + v_i(t)\end{aligned}\tag{2.25}$$

where  $x_i$  denotes states,  $\theta_i$  denotes system parameters,  $u_i$  denotes control signals,  $y_i$  denotes the outputs or measurements,  $f(\cdot)$  denotes the nonlinear process model,  $g(\cdot)$  denotes the nonlinear measurement model, and  $w_i$  and  $v_i$  denote process and measurement noise, respectively.

### 2.2.2 Semi-decentralized FDI architecture

In the semi-decentralized architecture as show in Figure 2.3, the FDI unit for each spacecraft has access to the control input and information from its neighbouring spacecraft. The communication of information between spacecraft is modelled using graph theory with a vertices and edges.

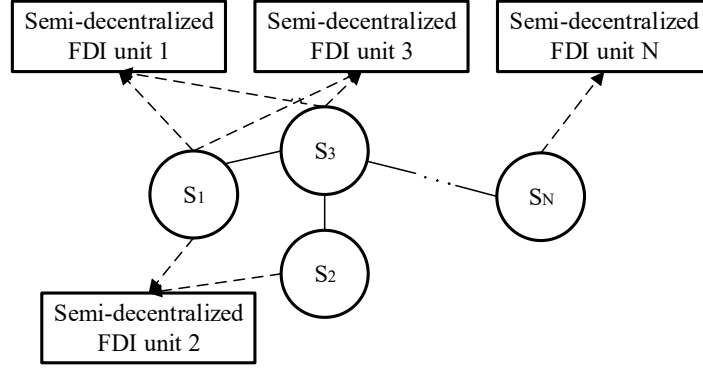


Figure 2.3 Semi-decentralized FDI architecture [134]

The information communicated between spacecraft gives the FDI unit of any spacecraft the ability to detect and isolate actuator faults of the given spacecraft as well as of its neighbouring spacecraft. The mathematical model of this architecture is not discussed in this section, since applications of the proposed methods to this architecture are suggested as future work for this research.

### 2.2.3 Centralized FDI architecture

In the centralized architecture as shown in Figure 2.4, a single FDI module is designed and in charge of the entire formation flying mission. All spacecraft send their states and control signal information to the central FDI module for processing.

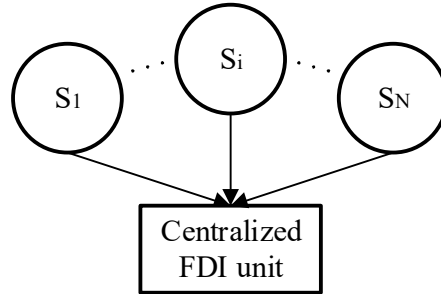


Figure 2.4 Centralized FDI architecture [134]

The mathematical model for this structure can be formulated as [134]

$$\begin{aligned} \dot{x} &= [\dot{x}_1, \dot{x}_2, \dots, \dot{x}_N]; \quad \dot{x}_i = f(x_i(t), \theta_i(t), u_i(t)) + w_i(t) \\ y &= [y_1, y_2, \dots, y_N]; \quad y_i(t) = g(x_i(t)) + v_i(t) \end{aligned} \tag{2.26}$$

where  $x_i$  denotes states,  $\theta_i$  denotes system parameters,  $u_i$  denotes control signals,  $y_i$  denotes the outputs or measurements,  $f(\cdot)$  denotes the nonlinear process model,  $g(\cdot)$  denotes the nonlinear measurement model, and  $w_i$  and  $v_i$  denote process and measurement noise, respectively. The main difference between this model and decentralized model is in the processing of information where in the decentralized architecture, each spacecraft individually processes information (hence,  $\dot{x}_i, y_i$ ), while in this architecture all information is processed

by one central unit (hence,  $\dot{x}, y$ ).

In this thesis, at the formation level, the states and measurements required for the isolation are satellite angular velocity and quaternions for each individual satellite. However, it was observed that the only satellite angular velocity or quaternion would suffice isolation requirement under the numerical simulations conducted in this study. Nevertheless, it should be considered in the future studies the feasibility of using partial information availability as well as other topologies and control schemes applicable to type of formation.

$$\begin{aligned}
 \dot{x} &= [\dot{x}_1, \dot{x}_2, \dots, \dot{x}_N]; \\
 \dot{x}_i &= \begin{cases} \dot{\omega}_{BI_i}^B = J_i^{-1}(-\omega_{BI_i}^B \times (J_{s_i} \omega_{BI_i}^B + A_{RW_i} J_w \Omega_i) - A_{RW_i} \tau_{RW_i} + \tau_{e_i}) + w_i \\ \begin{bmatrix} \dot{q}_{v_i} \\ \dot{q}_{4_i} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_{4_i} I + q_{v_i}^\times \\ -q_{v_i}^T \end{bmatrix} \omega_{BL_i}^B \\ \dot{I}_m = G_d \omega_d [f_3(\omega, I_m) - f_5(\omega)] - \omega_d I_m + G_d \omega_d V_{Comm} \\ \dot{\omega}_m = \frac{1}{J_w} \{f_1(\omega) + k_t I_m [f_2(\omega) + 1] - \tau_v \omega - \tau_c f_4(\omega) + \tau_{noise}\} \end{cases} \quad (2.27) \\
 y &= [y_1, y_2, \dots, y_N]; \\
 y_i &= x_i(t) + v_i
 \end{aligned}$$

where the governing satellite equations are discussed in the previous section in eqs. (2.10) and (2.13); and the governing actuator (RW) equations are discussed in the next section in eq. (2.31).

## 2.3 Actuator Model

To date, RWs are the most common actuators that can fit within small satellites mass, power, and volume constraints and provide adequate control torque [14]. These constraints vary depending on the payload and mission requirements. This section provides details on RW model, structure, and possible fault/failure scenarios.

### 2.3.1 Reaction Wheel Model

Reaction wheels have been among the most common actuators to provide full 3-axis attitude control for small satellites. An RW is a flywheel mounted to an electric motor. The RW model includes motor disturbances such as cogging and ripple torque, nonlinearities, and back electromotive force (BEMF) torque limiting (Figure 2.5). The RW model parameters in this study are based on ITHACO 'type A' by Goodrich where system parameters can be found in [76]. There are several essential loops incorporated in each RW model to allow for precision.

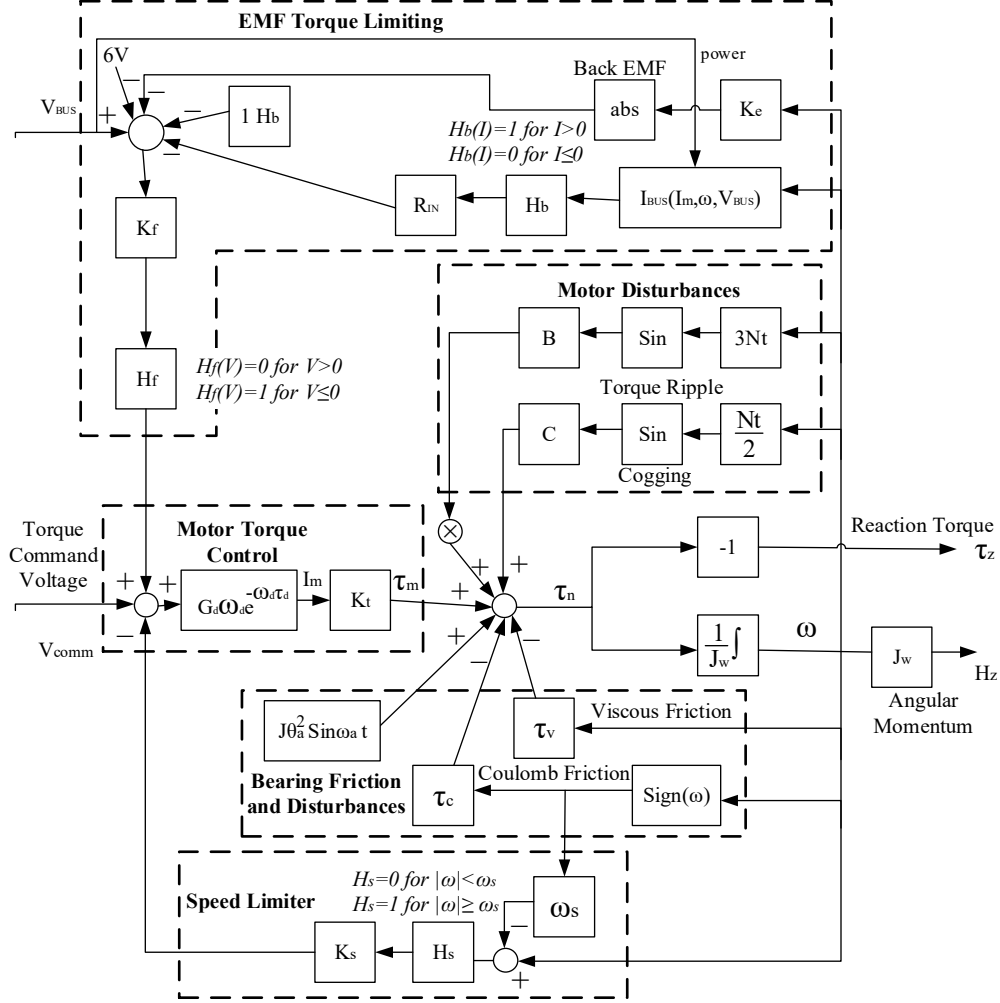


Figure 2.5 Reaction wheel model [76]

These loops are shown with dashed lines in Figure 2.5 and can be formulated as follows [16]:

- a) The negative feedback EMF torque limiting loop due to low bus voltage condition that can limit the motor torque at high speed due to increasing back-EMF voltage gain of the motor [16]:

$$I_{bus} = \left( \frac{1}{V_{bus} - 1} \right) (I_m^2 R_B + 0.04 |I_m| V_{bus} + P_q + \omega I_m K_e) \quad (2.28)$$

where  $I_{bus}$  and  $V_{bus}$  are the bus current and voltage, respectively;  $I_m$  is the motor current,  $K_e$  is the BEMF voltage gain, and  $\omega$  is the wheel's angular speed.

- b) The negative feedback viscous friction in addition to Coulomb friction, which is generated in the bearings [16]:

$$\tau_v = (0.049 - 0.0002(T + 30)) \quad (2.29)$$

where  $\tau_v$  is the viscous torque, and  $T$  is the wheel's temperature in degrees Celsius. It is

important to note the unit for  $\tau_v$  is mN.m/rad/sec when incorporating this parameter in simulations.

- c) The Coulomb friction is caused by the friction in bearings.
- d) The negative feedback speed limiter loop to prevent wheel speed from saturation.
- e) The motor torque control. The motor has a torque gain  $k_t$ , which delivers a torque proportional to the current driver. Therefore, to inject faults in the motor current it would be sufficient to change the torque gain  $k_t$ .
- f) The torque noise disturbance  $\tau_{noise}$  is a very low frequency torque variation from bearings due to lubricant dynamics and is formulated as follows [16].

$$\tau_{noise} = J_w \theta_a \omega_a^2 \sin(\omega_a t) \quad (2.30)$$

The nonlinear model of the RW including discontinuous functions approximated with sigmoidal functions can be expressed as follows [16]:

$$\begin{aligned} \dot{I}_m &= G_d \omega_d [f_3(\omega, I_m) - f_5(\omega)] - \omega_d I_m + G_d \omega_d V_{Comm} \\ \dot{\omega} &= \frac{1}{J_w} \{f_1(\omega) + k_t I_m [f_2(\omega) + 1] - \tau_v \omega - \tau_c f_4(\omega) + \tau_{noise}\} \end{aligned} \quad (2.31)$$

in which [16]

$$\begin{aligned} f_1(\omega) &= C \sin\left(\frac{Nt}{2}\omega\right) \\ f_2(\omega) &= B \sin(3Nt\omega) \\ f_3(\omega, I_m, V_{bus}) &= \frac{\exp[-aV(\omega, I_m, V_{bus})]}{1 + \exp[-aV(\omega, I_m, V_{bus})]} V(\omega, I_m, V_{bus}) \\ f_4(\omega) &= \frac{1 - \exp(-a\omega)}{1 + \exp(-a\omega)} \\ f_5(\omega) &= \frac{k_s[\omega - \omega_s f_4(\omega)]}{2} \left\{ \frac{1}{1 + \exp[-a(\omega - \omega_s)]} + \frac{1}{1 + \exp[a(\omega + \omega_s)]} \right\} \\ V(\omega, I_m, V_{bus}) &= k_f \left[ V_{bus} - 6 - \frac{1}{1 + \exp(-aI_{bus})} (1 + R_{in} I_{bus}) \right. \\ &\quad \left. - \frac{1 - \exp(-ak_e \omega)}{1 + \exp(-ak_e \omega)} k_e \omega \right] \end{aligned} \quad (2.32)$$

where  $f_1$  and  $f_2$  account for motor disturbances,  $f_3$  accounts for the EMF torque limiting block,  $f_4$  accounts for analytical approximation of the sign function in the Coulomb friction block,  $f_5$  represents the speed limiter block, and  $V_{Comm}$  is the torque command voltage.  $I_m$  (motor current) and  $\omega$  (motor angular velocity) are RW system states. Finally, the sigmoidal parameter  $a = 10$  is used [16].



Figure 2.6 Reaction wheel unit developed in the SSDC lab at Ryerson University

Table 2.1 ITHACO Type ‘A’ Reaction Wheel Parameters [76]

Parameter	Value
Coulomb Friction ( $\tau_c$ )	0.002 Nm
Viscous Friction ( $\tau_v$ )	$3.84 \times 10^{-4}$ Nm/rad/s
Drive gain time constant ( $\tau_d$ )	0.245
Ripple Torque (B)	0.22
Temperature ( $T$ )	23° C
Cogging Torque (C)	0
Torque Noise Frequency ( $\omega_a$ )	0.2 rad/sec
Jitter Angle ( $\theta_a$ )	0.05 rad = 3 degrees
BEMF ( $K_e$ ) Nominal	0.029 V/rad/s
Motor Torque Constant ( $k_t$ )	0.029 Nm/A
Bus Voltage ( $V_{bus}$ ) Nominal	8 V
Bridge Resistance ( $R_B$ )	2 $\Omega$
Driver Gain ( $G_d$ )	0.19 A/V
Number of Motor Poles (N)	36
Input Filter Resistance ( $R_{IN}$ )	2 $\Omega$
Quiescent Bus Power ( $P_q$ )	3 W
Driver Bandwidth ( $\omega_d$ )	9 rad/s
Voltage Feedback Gain ( $k_f$ )	0.5 V/V
Over-speed Circuit Gain ( $k_s$ )	95
Maximum Wheel Speed ( $\omega_s$ )	680 rad/s

The analytical nonlinear model was discretized using Euler’s backward difference method with sampling time of  $T_s = 10 \text{ ms}$ . Based on a discussion provided in [16], this value for sampling time ensures the validity of Nyquist-Shannon sampling theorem. In addition, extensive numerical simulations on continuous model illustrated in Figure 2.5 and discrete model from Eq. (2.31) demonstrated consistency between the two [16]. All reaction wheel parameters used for the simulations in this study are listed in Table 2.1 with their values for ITHACO Type ‘A’ RW.



### 2.3.2 Reaction Wheel Assembly

Reaction wheel assemblies can be in many forms and configurations depending on the requirements for specific applications. In this study, we have examined performance of the proposed methodology on most popular configurations: (1) standard four-wheel configuration with three orthogonal RWs and one redundant oblique RW, (2) four RWs in pyramid configuration (see Figure 2.8). The mapping of each actuator's torque contribution to each principal axis of the spacecraft body frame (see Figure 2.7) is as follows:

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = A_{RW} \begin{bmatrix} \tau_{w1} \\ \tau_{w2} \\ \tau_{w3} \\ \tau_{w4} \end{bmatrix} \quad (2.33)$$

where  $\tau_x, \tau_y, \tau_z$  are applied torques to satellite in  $x, y, z$  directions, respectively and  $\tau_{wi}$  is torque generated by each RW on its own axis of rotation. The mapping matrix ( $A$ ) used for configuration 1 and 2 are as follows:

$$A_{RW_1} = \begin{bmatrix} 1 & 0 & 0 & -c\beta s\alpha \\ 0 & 1 & 0 & -c\beta s\alpha \\ 0 & 0 & 1 & s\beta \end{bmatrix} \quad (2.34)$$

$$A_{RW_2} = \begin{bmatrix} c\beta s\alpha & -c\beta s\alpha & -c\beta s\alpha & c\beta s\alpha \\ -c\beta c\alpha & -c\beta c\alpha & c\beta c\alpha & c\beta c\alpha \\ s\beta & s\beta & s\beta & s\beta \end{bmatrix}$$

where  $A_{RW_i}$  is mapping matrix for configuration  $i$ . Math functions  $\sin(\cdot)$  and  $\cos(\cdot)$  are abbreviated as  $c(\cdot)$  and  $s(\cdot)$ , respectively with  $\alpha$  as the in-plane angle and  $\beta$  as the out-of-plane angle. For this study  $\alpha$  and  $\beta$  angles are considered as  $45^\circ$  each. The 3D models in Figure 2.8 were drafted using Autodesk Inventor professional.

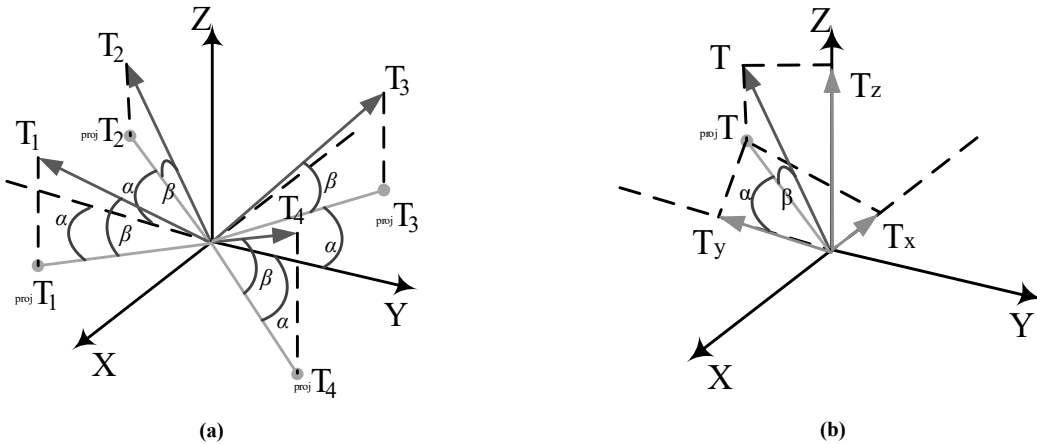


Figure 2.7 Torque decomposition for RW assembly (a) pyramid, (b) standard four-wheel

### 2.3.3 Control Moment Gyro

Reaction wheels have a fixed spin axis, and the torque generated by these actuators

is through a change in their flywheel angular speed. They are the simplest and least expensive of all momentum exchange devices; however, the control torque of most reaction wheels with less than 1 N.m maximum torque is much smaller than their counterpart CMGs with 100-500 N.m maximum torque.

A CMG is a RW or MW, whose angular momentum vector direction can be changed by gimbaling the spinning rotor. The spinning rotor is mounted on a gimbal if it is a single gimbal design or multiple gimbals for more degrees of freedom and precision. Spinning the rotor causes a precise, gyroscopic reaction, which is orthogonal to the rotor spinning and gimbal axis. Therefore, CMG is a torque amplification device that can apply large control outputs on the satellite with a small gimbal torque input. This capability makes CMGs favorable for precision pointing and tracking control of agile spacecraft in LEO and momentum management of large space vehicles.

In general, the categorization of CMGs is based on the gimbaling arrangements and the configuration used for redundancy management and failure accommodation. The two basic types of CMGs are (1) single-gimbal control moment gyros (SGCMGs) and (2) double-gimbal control moment gyros (DGCMGs). For SGCMGs, the spinning rotor is constrained to rotate in a circle in a plane normal to the gimbal axis. The SGCMGs are much simpler than DGCMGs in terms of hardware and offer significant cost, power, weight, and reliability advantages over DGCMGs.

For the purpose of optimal redundancy management and failure accommodation, many different arrangements of CMGs have been developed in the past including four SGCMGs in a pyramid configuration, six parallel-mounted SGCMGs used in the Mir space station, three orthogonally mounted DGCMGs used in NASA's Skylab, and four parallel-mounted DGCMGs used in ISS.

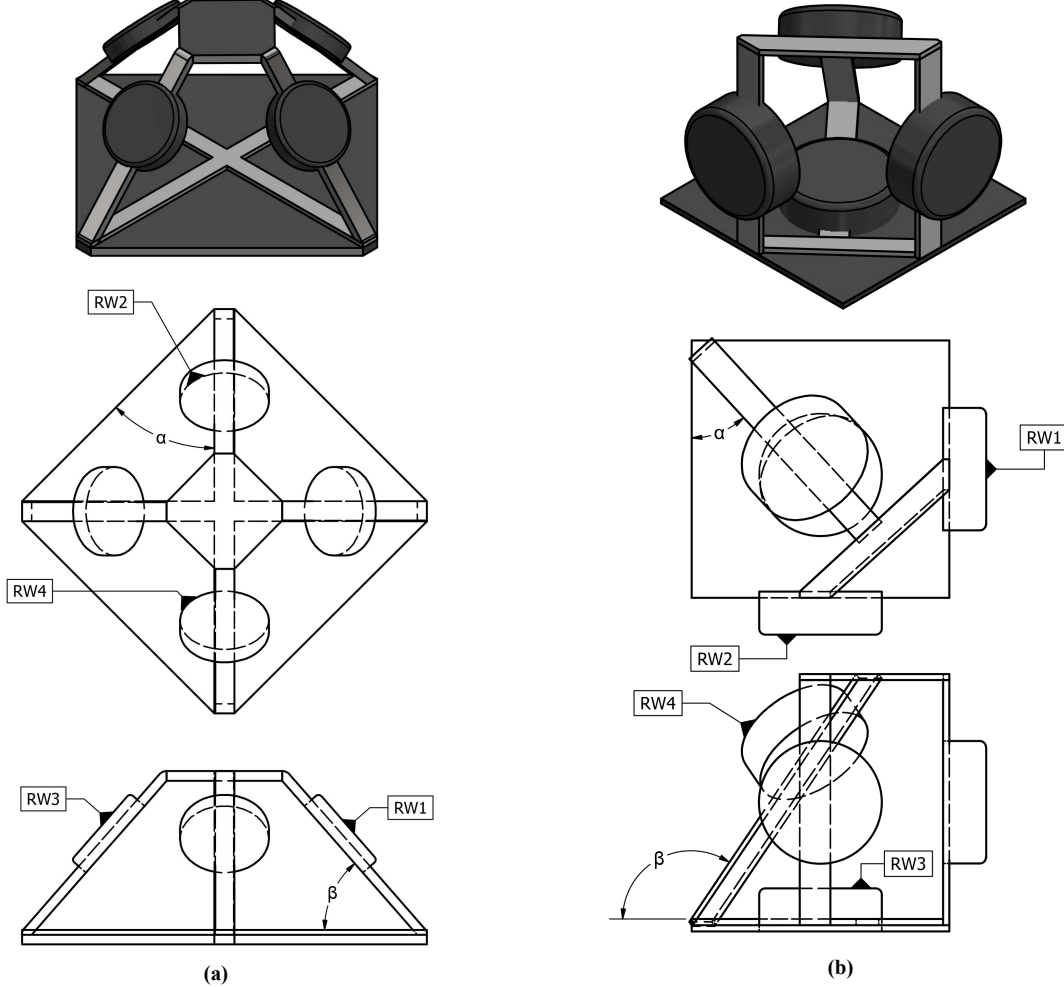
When using CMGs, a steering logic needs to be developed, which generated the required gimbal rate commands based on the commanded spacecraft control torques. The optimal steering logic is the one that generates exactly the commanded spacecraft control torques. One of the main limitations with CMGs is the geometric singularity problem when they are used for the spacecraft attitude control. When singularity occurs, no control torque is generated for the commanded gimbal rates, and most valuable type among others are SGCMGs due to reduced degrees of freedom.

Referring back to Eq. (2.4), when CMGs are used as actuators instead of RWs, the  $\dot{A}$  term in Eq. (2.6) will be non-zero, while  $\dot{H}$  will be zero due to no change in CMG's flywheel speed and consequently momentum. The CMG angular momentum is in general a function of CMG gimbal angles  $\delta = (\delta_1, \dots, \delta_n)$  and flywheels angular speed  $\Omega = (\Omega_1, \dots, \Omega_n)$

$$H_{CMG} = H(\delta, \Omega) \quad (2.35)$$

One approach to the CMG steering logic design is simply to find the inverse of Eq.

(2.35) where the task is to determine optimal gimbal angles trajectories that can generate the commanded momentum  $\mathbf{h}$  while meeting hardware constraints such as gimbal rate limits and gimbal stops, while avoiding singularities.



**Figure 2.8** RW assembly (a) pyramid configuration, (b) standard four-wheel configuration

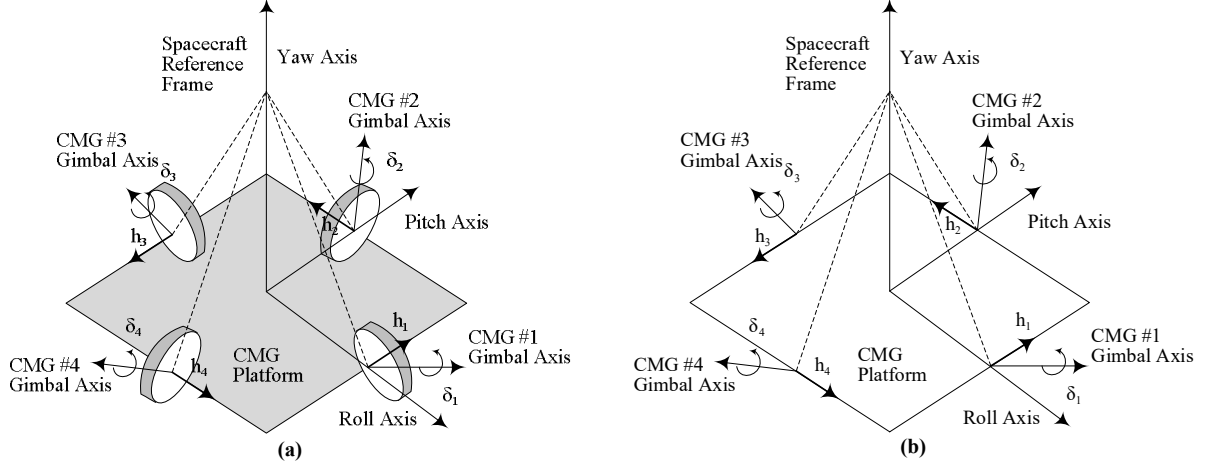
The second approach is to use the differential relationship between gimbal angles and the CMG momentum vector. For such method, the derivation of  $\mathbf{h}$  is obtained as [135]

$$\dot{\mathbf{h}}_{CMG} = \mathbf{A}_{CMG} \dot{\boldsymbol{\delta}} \quad (2.36)$$

where  $\mathbf{A}_{CMG} = \mathbf{A}_{CMG}(\boldsymbol{\delta}) \in \mathbb{R}^{3 \times n}$  is the Jacobin matrix defined as [135]

$$\mathbf{A}_{CMG} = \frac{\partial \mathbf{h}}{\partial \boldsymbol{\delta}} = \left[ \frac{\partial h_i}{\partial \delta_i} \right] \quad (2.37)$$

The CMG steering logic then becomes the inverse of  $\mathbf{A}_{CMG} \dot{\boldsymbol{\delta}} = \dot{\mathbf{h}}_{CMG}$ . If we consider a typical pyramid mounting configuration for four SGCMGs as shown in Figure 2.9 where each face is inclined with an out-of-plane angle of  $\beta = 54.73 \text{ deg}$ , which makes the momentum envelope nearly spherical [135].



**Figure 2.9** Pyramid assembly of four SGCMGs (a) schematics (b) free body diagram [135]

For the conventional pyramid mount (Figure 2.9), the total CMG angular momentum can be expressed in the spacecraft reference frame, considering that the moment of inertia of the gimbals along the rotation axis is considered while the rotational inertia about other axes are assumed to be negligible, as [135]

$$\begin{aligned}
 h_{CMG} &= \sum_{i=1}^4 h_i(\delta_i, \Omega_i) \\
 &= \begin{bmatrix} -c\beta s\delta_1 & -c\delta_2 & c\beta s\delta_3 & c\delta_4 \\ c\delta_1 & -c\beta s\delta_2 & -c\delta_3 & c\beta s\delta_4 \\ s\beta s\delta_1 & s\beta s\delta_2 & s\beta s\delta_3 & s\beta s\delta_4 \end{bmatrix} [h_{0_1}(\Omega_1) \ h_{0_2}(\Omega_2) \ h_{0_3}(\Omega_3) \ h_{0_4}(\Omega_4)]^T
 \end{aligned} \quad (2.38)$$

where  $h_i$  is the angular momentum of each CMG expressed in the spacecraft reference frame.  $\delta_i$  are the gimbal angles,  $\Omega_i$  are flywheel angular speed, and  $h_{0_i}$  is the momentum magnitude for the  $i$ th CMG. The time derivative of the CMG angular momentum can be obtained as [135]

$$\dot{h}_{CMG} = \sum_{i=1}^4 \dot{h}_i(\delta_i, \Omega_i) = [h_{0_1}(\Omega_1) \ h_{0_2}(\Omega_2) \ h_{0_3}(\Omega_3) \ h_{0_4}(\Omega_4)] A_{CMG} \dot{\delta} \quad (2.39)$$

where  $\delta$  is the gimbal angle vector and [135]

$$A_{CMG} = \begin{bmatrix} -c\beta c\delta_1 & -s\delta_2 & c\beta c\delta_3 & -s\delta_4 \\ -s\delta_1 & -c\beta c\delta_2 & -s\delta_3 & c\beta c\delta_4 \\ s\beta c\delta_1 & s\beta c\delta_2 & s\beta c\delta_3 & s\beta c\delta_4 \end{bmatrix} \quad (2.40)$$

For a known control torque  $\tau_c$ , the CMG torque command  $\dot{h}$  is chosen as [135]

$$\dot{h}_{CMG} = u = -\tau_c - \omega_{BI}^B \times h_{CMG} \quad (2.41)$$

And the gimbal rate command  $\dot{\delta}$ , given  $h_0 = h_{0_1} = h_{0_2} = h_{0_3} = h_{0_4}$  is obtained as [135]

$$\dot{\delta} = \left(\frac{1}{h_0}\right) A_{CMG}^+ \dot{h}_{CMG} \quad (2.42)$$

where  $A_{CMG}^+ = A_{CMG}^T (A_{CMG} A_{CMG}^T)^{-1}$ , often referred to as the pseudoinverse steering logic and

most CMG steering laws determine the gimbal rate commands with its variations. In the following chapters, dynamic equations for satellite are employed based on the actuator used onboard. Where CBM of RW is of interest, Eq. (2.10) is used, and where CMGs are employed, Eq. (2.12) is used.

## 2.4 Kalman Filters

The Kalman Filter (KF) is a model-based observer that produces estimates based on the stochastic properties of a system. This feature of the KF makes it robust to measurement and process noise, hence, make it practical to implement. The most common application for KFs is in state and parameter estimation, and therefore, a great candidate for fault diagnosis and identification (FDI), which has been studied extensively in the literature. Three of the common KF variants applied to the FDI problem are the EKF, AKF, UKF, and adaptive unscented Kalman filter (AUKF). The EKF uses linearization to estimate the true mean and covariance of the random variable while the UKF uses sigma points or particles. These sigma points when spread through a non-linear system, capture the posterior mean and covariance accurate to the third order, whereas the standard EKF is only accurate to the first order [62]. This means that the UKF is a better option for highly non-linear systems but has no advantage for weakly non-linear systems. Whether the EKF or UKF structures are used the adaptive mechanism, which could be used in both, remains the same. These filters can be implemented for either state/parameter estimation or joint state and parameter estimation. State estimation uses the standard KF equations without any modifications and hence, is the most straightforward approach to Kalman filtering. The goal of this approach is to estimate the system states based on the mathematical model of the system. The FDI approach could be as simple as just comparing the measured system states from the sensors and with the predicted model states, if the residuals exceed a threshold, then a fault has been detected. On the other hand, isolation and identification are not as direct and generally require a good choice of residuals in addition to post-processing of the raw data. Parameter estimation, however, is a form of system identification as it implicates estimating the physical parameters of a system. In order to accommodate the parameters as the state-vector for this purpose, modifications need to be made to the KF equations. The FDI problems are then directly resolved when the parameters of a system are estimated; the reason for that is that the change in parameter(s) identifies where the faults have occurred and the level of severity for each. Therefore, the parameter estimation approach is better suited for FDI problems than the state-estimation is.

### 2.4.1 Unscented Kalman Filter

The Robotics Research Group (RRG) in Oxford UK proposed a "New Filter" in 1994, which was named the UKF. Then, in 1997 the first paper was published describing the UKF as a new extension of the KF to nonlinear systems. The UKF is a variant of the KF with

the capability of estimating the mean and covariance of a random variable to the third order while the EKF only approximates them to the first order. As a result, higher order terms in the dynamics are not ignored in UKF. This filter is built on the belief that "it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function." [63]. Simply put, it means that the linear approximations are not used to approximate non-linear functions in here; instead, the statistical moment of the state is approximated. The UKF is, in fact, a form of particle filter applied to a random variable with a Gaussian distribution. Particle filters can be applied to systems with sampling densities that are non-Gaussian [64] where the posterior distribution of the state is approximated using a large number of "well chosen" *particles* or *sigma points* that change randomly in time according to the model dynamics and system measurements [65].

The parameter estimation with the assumption that the full state measurement is available is first presented based on the standard UKF.

Consider the following nonlinear discrete stochastic system

$$\begin{cases} X_k = f(X_{k-1}, \theta_k, u_k) + w_k \\ Z_k = g(X_k) + v_k \end{cases} \quad (2.43)$$

where  $X_k \in \mathbb{R}^n$  is the state vector at time step  $k$ ,  $\theta_k \in \mathbb{R}^n$  is the model parameter vector,  $Z_k \in \mathbb{R}^m$  is the measurement vector,  $w_k \in \mathbb{R}^n$  is the additive process noise,  $v_k \in \mathbb{R}^m$  is the additive measurement noise,  $f(\cdot)$  is a nonlinear process model, and  $g(\cdot)$  is nonlinear measurement model. The process and measurement noises are assumed to be uncorrelated zero-mean Gaussian white noises with covariances  $E[w_k w_j^T] = Q_k \delta_{kj}$ ,  $E[v_k v_j^T] = R_k \delta_{kj}$ , respectively. The process noise covariance  $Q_k$  is non-negative definite, the measurement noise covariance  $R_k$  is positive definite, and  $\delta_{kj}$  is the Kronecker- $\delta$  function. When the system equation is in continuous form as

$$\dot{x} = f(x(t), \theta(t), u(t)) + w(t) \quad (2.44)$$

It can be transformed to the discrete domain using

$$X_k = X_{k-1} + \int_{(k-1)\Delta t}^{k\Delta t} f(x(t), \theta(t), u(t)) dt + w_k \quad (2.45)$$

where  $\Delta t$  or  $T_s$  is discretization step size or sampling time. The integration can be carried out using the fourth order Runge-Kutta (RK4) method. The formulation for parameter estimation of a nonlinear discrete stochastic system using UKF [136] with minor modifications [137] can be described in the following steps:

**1) Compute weights** [137]

$$\mathbf{W}_0^{(m)} = \frac{\kappa}{n + \kappa}; \quad \mathbf{W}_0^{(c)} = \frac{\kappa}{n + \kappa} + (1 - \alpha^2 + \lambda); \quad \mathbf{W}_i^{(m)} = \mathbf{W}_i^{(c)} = \frac{1}{2(n + \kappa)} \quad (2.46)$$

2) *Establish symmetric sigma points about the state estimate* [137]

$$\hat{\chi}_0 = \hat{\mathbf{x}}_{p(k+1)}^- \quad (2.47)$$

$$\hat{\chi}_i = \hat{\mathbf{x}}_{p(k+1)}^- \pm \sqrt{(n + \kappa) \mathbf{P}_{\mathbf{x}\mathbf{x}}^-}_{(k+1)i}}, \quad \forall i = 1, 2, \dots, n \quad (2.48)$$

3) *Predict mean and covariance of states* [137]

$$\mathbf{P}_{\mathbf{x}\mathbf{x}}^-_{k+1} = \mathbf{P}_{\mathbf{x}\mathbf{x}}^-_k + \mathbf{Q}_{k+1} \quad (2.49)$$

$$\mathbf{x}_{p(k+1)}^- = \mathbf{x}_{pk}^+ \quad (2.50)$$

4) *Instantiate sigma points through measurement model* [137]

$$\mathbf{Y}_i = g(\hat{\chi}_i) \quad (2.51)$$

5) *Predict mean and covariance of measurements* [137]

$$\bar{\mathbf{y}} = \sum_{i=0}^{2n} W_i \mathbf{Y}_i \quad (2.52)$$

$$\mathbf{P}_{yy} = \sum_{i=0}^{2n} W_i [\mathbf{Y}_i - \bar{\mathbf{y}}][\mathbf{Y}_i - \bar{\mathbf{y}}]^T + \mathbf{R}_{k+1} \quad (2.53)$$

6) *Predict cross covariance* [137]

$$\mathbf{P}_{xy} = \sum_{i=0}^{2n} W_i [\hat{\chi}_i - \hat{\mathbf{x}}_{p(k+1)}^-][\mathbf{Y}_i - \bar{\mathbf{y}}]^T \quad (2.54)$$

7) *Calculate gain and update* [137]

$$\mathbf{K}_{k+1} = \mathbf{P}_{xy} \mathbf{P}_{yy}^{-1} \quad (2.55)$$

$$\hat{\mathbf{x}}_{p(k+1)}^+ = \hat{\mathbf{x}}_{p(k+1)}^- + \mathbf{K}_{k+1}(\mathbf{y} - \bar{\mathbf{y}}) \quad (2.56)$$

$$\mathbf{P}_{\mathbf{x}\mathbf{x}}^+_{k+1} = \mathbf{P}_{\mathbf{x}\mathbf{x}}^-_{k+1} - \mathbf{K}_{k+1} \mathbf{P}_{yy} \mathbf{K}_{k+1}^T \quad (2.57)$$

where the  $\sqrt{(n + \kappa) \mathbf{P}_{\mathbf{x}\mathbf{x}}^-}_{i}$  terms denote the scaled  $i^{th}$  rows or columns of  $\sqrt{\mathbf{P}_{\mathbf{x}\mathbf{x}}^-}$ ,  $\mathbf{W}^{(m)}$  is the component weight for mean calculation,  $\mathbf{W}^{(c)}$  is the component weight for covariance calculation,  $n$  is the dimension of the states,  $\kappa \geq 0$ ,  $0 \leq \alpha < 1$ , and  $\lambda \geq 0$  are the control factors for the spread of sigma points,  $\bar{\mathbf{y}}$  is the mean of variable  $\mathbf{y}$ ,  $\mathbf{P}_{yy}$  is the covariance of measurements matrix,  $\mathbf{x}$  is the n-state random variable,  $\hat{\mathbf{x}}$  is the initial sigma point,  $\mathbf{P}_{xx}$  is the posteriori estimates covariance matrix,  $\mathbf{Y}_i$  is the result of instantiated sigma points through measurement model,  $\mathbf{P}_{xy}$  is the cross covariance matrix,  $\mathbf{K}$  is the Kalman filter gain, and + and - superscripts show the pre-process and post-process values at each iteration, respectively. The primary difference between state and parameter estimation with UKF is most notable in steps 3. Since no dynamics information are available for parameters, the sigma

points from step 2 are not propagated; instead they are used in subsequent steps as is [14]. For state estimation with UKF step 3 is broken into two steps as follows

**3a) *Instantiate sigma points through process model [14]***

$$\tilde{\chi}_i = f(\hat{\chi}_i) \quad (2.58)$$

**3b) *Predict mean and covariance of states [14]***

$$\bar{x} = \sum_{i=0}^{2n} W_i \tilde{\chi}_i \quad (2.59)$$

$$P_{xx} = \sum_{i=0}^{2n} W_i [\tilde{\chi}_i - \bar{x}][\tilde{\chi}_i - \bar{x}]^T + Q_{k+1} \quad (2.60)$$

### 2.4.2 Adaptive Unscented Kalman Filter

The proposed adaptive scheme consists of three major parts: (1) adapting process and measurement noises, (2) detecting fault occurrence, (3) adapting states/parameters covariance matrix. The following adaptive mechanism is used to adjust process and measurement noise covariance matrices on-line [138]. The approach is adapted from [41] where the algorithm parameters are optimized off-line for specific abrupt fault scenarios. First Eq. (2.49) is re-written as [41]

$$P_{xx}^-_{k+1} = P_{xx}^-_k \varepsilon + Q_{k+1} \quad (2.61)$$

where the adaptive fading factor  $\varepsilon$  is defined as [41]

$$\varepsilon = 1 + \frac{\text{trace}(\mathbf{R}_k)}{\zeta} \quad (2.62)$$

where  $\text{trace}(\cdot)$  is the sum of the diagonal elements of the matrix  $\mathbf{R}_k$ , and  $\zeta$  is the controlling factor. Next, the process and measurement matrices are adjusted as follows [41]:

$$\hat{\mathbf{C}}_r = \frac{1}{N} \sum_{j=k-N+1}^k r_k^T r_k \quad (2.63)$$

$$\hat{\mathbf{R}}_{k+1} = \hat{\mathbf{C}}_r + P_{yy} \quad (2.64)$$

$$\hat{\mathbf{Q}}_{k+1} = \mathbf{K}_{k+1} \hat{\mathbf{C}}_r \mathbf{K}_{k+1}^T \quad (2.65)$$

where  $r_k = (z - \bar{y})$  is the residual,  $z$  is the measurement,  $\bar{y}$  is the estimated measurement,  $\hat{\mathbf{C}}_r$  is the estimated residual sequence covariance matrix,  $\hat{\mathbf{R}}$  is the estimated measurement noise covariance matrix,  $\hat{\mathbf{Q}}$  is the estimated process noise covariance matrix, and  $N$  is the window size.

## 2.5 Particle Filter

The most common forms of particle filters developed so far are based on a sequential



importance sampling (SIS) algorithm. It is a Monte Carlo (MC) method that forms the basis for the majority of the sequential MC (SMC) filters developed [139], [140]. This SMC approach is widely known as bootstrap filtering [141], the condensation algorithm [142], particle filtering [143], interactive particle approximations [144], and survival of the fittest [145]. In this technique, a recursive Bayesian filter by MC simulations is implemented to represent the posterior probability density function (PDF) through a set of random samples (particles) with associated weights and estimations of the two. As the number of samples grows, the estimations become closer to optimal Bayesian estimates, and the MC characterization becomes an equivalent representation to the actual PDF description.

### 2.5.1 Algorithm

A discrete weighted approximation of the true posterior  $p(x_{0:k}|z_{1:k})$  can be formulated as [146]

$$p(x_{0:k}|z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k} - x_{0:k}^i) \quad (2.66)$$

where  $N_s$  is number of samples,  $\{x_{0:k}^i, i = 1, \dots, N_s\}$  is a set of support points with associated weights  $\{w_k^i, i = 1, \dots, N_s\}$ , and  $x_{0:k} = \{x_j, j = 0, \dots, k\}$  is the set of all states up to time  $k$ . The function  $\delta_{x_k^i}(\cdot)$  is a function that if you integrate it over a set  $A$ , the result is 1 if  $x_k^i$  is in  $A$  and 0 otherwise:  $\int_A \delta_{x_k^i}(x) dx = 1$  if and only if  $x_k^i \in A$  and 0 otherwise. The weights are normalized to ensure  $\sum w_k^i = 1$ . The weights are chosen using the principle of importance sampling [147]. In this principle,  $p(x) \propto \pi(x)$  is a PDF that is difficult to generate samples from but  $\pi(x)$  can be evaluated for it. If  $x^i \sim q(x), i = 1, \dots, N_s$  are samples generated from a proposal importance density  $q(\cdot)$ , a weighted approximation to the density function  $p(\cdot)$  can be formulated as [146]

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i) \quad (2.67)$$

where [146]

$$w^i = \frac{\pi(x^i)}{q(x^i)} \quad (2.68)$$

Is the normalized weight of the  $i$ th particle. Hence, if samples  $x_{0:k}^i$  were drawn from an importance density  $q(x_{0:k}|z_{1:k})$ , the weights in Eq. (2.66) can be defined by Eq. (2.68) as [146]

$$w_k^i = \frac{p(x_{0:k}^i|z_{1:k})}{q(x_{0:k}^i|z_{1:k})} \quad (2.69)$$

In the sequential case, having samples that form an approximation to  $p(x_{0:k}|z_{1:k-1})$  the

goal is to approximate  $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$  with new set of samples. If the importance density is chosen as [146]

$$q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})q(\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k-1}) \quad (2.70)$$

Samples of  $\mathbf{x}_{0:k}^i \sim q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$  can be obtained by augmenting the existing samples  $\mathbf{x}_{0:k-1}^i \sim q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$  with the new state  $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})$ . The weight update equation can be formulated as [146]

$$w_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i, \mathbf{z}_{1:k})} \quad (2.71)$$

Furthermore, if  $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$ , the importance density would only be reliant on  $\mathbf{x}_{k-1}$  and  $\mathbf{z}_k$ . This is useful when only filtered estimate of  $p(\mathbf{x}_k|\mathbf{z}_{1:k})$  is required at each time step. In this case, only  $\mathbf{x}_k^i$  needs to be stored and the path  $\mathbf{x}_{0:k-1}^i$  and history of observations  $\mathbf{z}_{1:k-1}$  can be discarded. The modified weight can then be formulated as [146]

$$w_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)} \quad (2.72)$$

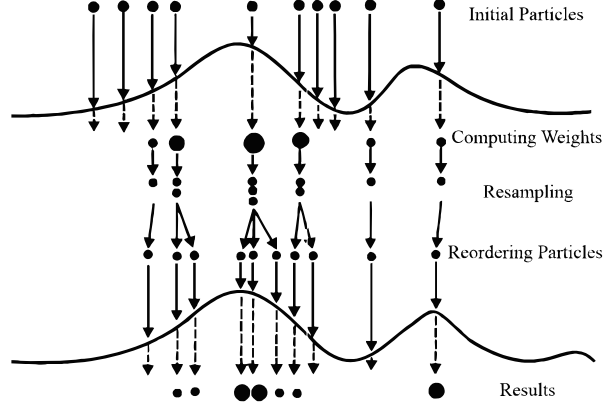
And the posterior filtered density can be approximated as [146]

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \quad (2.73)$$

where the weights are given in Eq. (2.72) and as  $N_s \rightarrow \infty$  the approximation in Eq. (2.73) approaches the true posterior density  $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ . It can therefore, be seen that the SIS algorithm sequentially propagates weights and support points as each measurement is received. A general schematic of the particle filter algorithm is shown in Figure 2.10 where particles are first initialized using a particular method. After initialization, a weight is computed for each particle using a distribution similarity function to measure how close that particle is to its true measured value. After all particles are weighted and a distribution is formed, resampling is done to ensure diversity of the particles. The process is repeated until satisfactory criteria are met.

### 2.5.2 Degeneracy

A common limitation with the SIS particle filter is that after a few iterations, all particles except for one will have negligible weights. This phenomenon is known as degeneracy. It is shown in [140] that it is impossible to avoid degeneracy due to the fact that the variance of the importance weights can only increase over time. Degeneracy means that a lot of computational resources are dedicated to updating particles that have almost no contribution to the approximation of the posterior density  $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ .



**Figure 2.10** Schematics of the particle filter algorithm

A suitable measure of degeneracy of the algorithm is the effective sample size  $N_{eff}$  [147]

$$N_{eff} = \frac{N_s}{1 + \text{Var}(w_k^{*i})} \quad (2.74)$$

where  $w_k^{*i} = p(x_k^i | z_{1:k}) / q(x_k^i | x_{k-1}^i, z_k)$  is known as *true weight*, and  $\text{Var}(\cdot)$  is the variance of a set. Since true weight cannot be calculate precisely, an estimate of  $N_{eff}$  can be obtained by [147]

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_s} (w_k^i)^2} \quad (2.75)$$

where  $w_k^i$  is the normalized weight obtained from Eq. (2.71). It is important to note  $N_{eff} \leq N_s$  and small values for  $N_{eff}$  points to acute degeneracy. Degeneracy is an undesirable effect in particle filters and a simple solution would be to use a large number of particles for  $N_s \rightarrow \infty$ , however, this is often impractical. Therefore, other solutions have been proposed in the literature including (1) good choice of importance density and (2) resampling.

### 2.5.3 Importance Density Choice

A good choice of importance density means choosing a density  $p(x_k | x_{k-1}^i, z_{1:k})$  that minimizes  $\text{Var}(w_k^{*i})$  so that  $N_{eff}$  is maximized. This optimal density is shown to be [140]

$$q(x_k | x_{k-1}^i, z_k)_{opt} = p(x_k | x_{k-1}^i, z_k) = \frac{p(z_k | x_k, x_{k-1}^i) p(x_k | x_{k-1}^i)}{p(z_k | x_{k-1}^i)} \quad (2.76)$$

However, this choice of importance density suffers from two major drawbacks (1) it requires to sample from  $p(x_k | x_{k-1}^i, z_k)$  and (2) it needs to evaluate the integral over the new state. In general, it is not straightforward to do either of these things, although there are two cases where use of optimal importance density is possible (See [146]).

### 2.5.4 Resampling

The basic idea of resampling is to eliminate those particles with negligible weights

and concentrate on those particles with predominant weights. The resampling involves generating a new set  $\{\mathbf{x}_k^{i*}\}_{i=1}^{N_s}$  by resampling  $N_s$  times from an approximate discrete representation of  $p(\mathbf{x}_k|\mathbf{z}_{1:k})$  using Eq. (2.73) so that  $\Pr(\mathbf{x}_k^{i*} = \mathbf{x}_k^j) = w_k^j$ . The resulting sample is in fact independent and identically distributed (IID) sample from the discrete density in Eq. (2.73). As a result, the weights are now reset to  $w_k^i = 1/N_s$ . Various methods of resampling include stratified sampling and residual sampling [148] and systematic resampling [149]. Although resampling process reduces degeneracy effect, it introduces other practical problems:

1. It limits the opportunity to parallelize since all particles need to be combined.
2. Sample impoverishment: particles with large weights are statistically selected and reselected again and again. This causes loss of diversity among particles as the regenerated sample set would include a lot of repeated particles. This is particularly problematic in the case of small process noise where all particles would collapse to a single point within a few iterations. Some methods to solve this issue can be named as *resample-move* algorithm [150] and *regularization* algorithm [151].
3. Since the diversity of the particles' paths is reduced, any smoothed estimates based on these paths can degenerate.

### 2.5.5 Alternatives to Sub-optimality

A good importance density is not always available. For example, if the prior  $p(\mathbf{x}_k|\mathbf{x}_{k-1})$  is used as the importance density and has a much broader distribution than the likelihood  $p(\mathbf{z}_k|\mathbf{x}_k)$ , only a few particles will have a large weight. In order to remedy this problem, some methods exist that introduce intermediate distributions between the prior and likelihood such as bridging densities [152] and progressive correction [153]. If the likelihood is very peaked but can be factorized into number of broader distributions, the partitioned sampling method [142] can be used.

### 2.5.6 Variations of Particle Filter

The SIS algorithm forms the basis for most particle filters. Other variations of particle filter proposed in the literature can be regarded as special cases of the SIS algorithm modifying the choice of importance sampling and/or resampling method. Among these variations stand *sampling importance resampling* (SIR) filter [141], *auxiliary sampling importance resampling* (ASIR) filter [154] and *regularized particle filter* (RPF) [155]. More details on variations and advantages of each variation can be found in their respective published work or in [146].

## 2.6 Confusion Matrix

In the field of machine learning and more specifically the problem of statistical classification, a confusion matrix, which is also known as an error matrix [156], is referred to as a specific table layout that allows for visualization of an algorithm's performance. Each row represents the instances in an actual class while each column represents the instances in a predicted class, or vice versa [156]. The name originates from the fact that this table layout can make it easy to see if the system is confusing two classes or mislabelling one as another. It can be considered as a special kind of contingency table, with two dimensions ("actual" and "predicted") where identical sets of "classes" are shown in both dimensions. Table 2.2 shows a sample of a confusion matrix. In this example, there were 10 trials to identify given samples as Class 1 or Class 2. It can be seen that the algorithm has performed poorly by classifying only 30% (3 out of 10) of the given samples correctly.

**Table 2.2 Sample confusion matrix**

Identified → Actual ↓	Class 1	Class 2
Class 1	3	7
Class 2	7	3

There are many metrics associated with a confusion matrix to quantitatively evaluate the performance of an algorithm or method. These metrics come with a series of terminologies where important factors from these series are listed below:

**Table 2.3 Terminology for confusion matrix [157]**

Terminology	Definition
condition positive (P)	the number of real positive cases in the data
condition negatives (N)	the number of real negative cases in the data
true positive (TP)	equivalent with hit
true negative (TN)	equivalent with correct rejection
false positive (FP)	equivalent with false alarm
false negative (FN)	equivalent with miss

The important metrics derived from these values that are more common in the literature are precision and accuracy. Precision or positive predictive value (PPV) and accuracy (ACC) are defined as [157]

$$PPV = \frac{TP}{TP + FP} \quad (2.77)$$

$$ACC = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.78)$$

These metrics are used in the following sections to evaluate performance of the proposed algorithms on various cases for particular systems through Monte Carlo simulations. In the following chapters, when dealing with fault detection, the matrix shown in Table 2.4 is used as a template to evaluate performance of the proposed algorithm for a particular fault scenario under different simulation conditions.

**Table 2.4 Sample Monte Carlo confusion matrix**

Identified $\rightarrow$ Actual $\downarrow$	Alarm	No Alarm
Fault	True Positive (TP)	False Negative (FN)
No Fault	False Positive (FP)	True Negative (TN)

Furthermore, based on the PPV and ACC metrics introduced above, precision and accuracy of the algorithm under different cases will be analyzed.

## 2.7 Conclusions

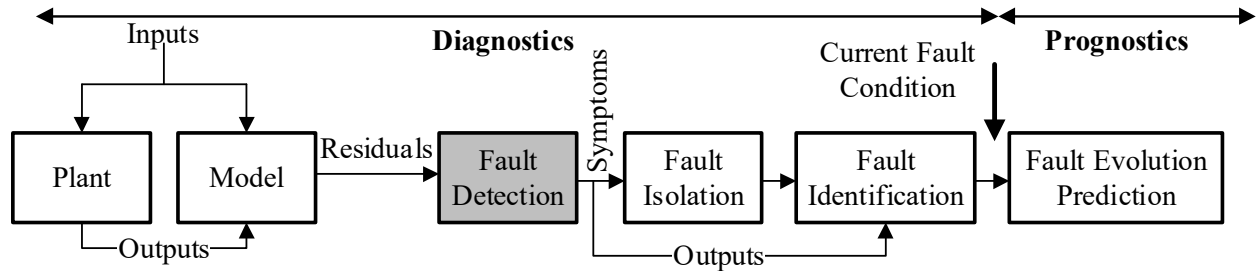
In this chapter, required preliminaries were presented to lay out a foundation for the better understanding of the reader. first, the satellite attitude model, including dynamics and kinematics, were explained based on the actuator used onboard. Next, the actuator model including a high fidelity RW's mathematical model and a RW assembly (RWA) along with its mapping methods were explained followed by CMG dynamics model and its integration with satellite dynamics when used onboard satellite. Next, a brief review on Kalman filters, unscented Kalman filters, and adaptive unscented Kalman filters were provided with focus on UKF and adaptation schemes. Finally, a brief review on particle filters and various aspects of their implementation strategies along with possibility of implementation of such methods into a fault prognosis problem were presented. In the following sections, proposed methodologies in fault detection, isolation, and prognosis are presented, respectively.

## CHAPTER 3

# Fault Detection

Reaction wheels, as one of the most commonly used actuators in satellite attitude control systems, are prone to malfunction which could lead to catastrophic failures.

Such malfunctions can be detected and addressed in time if proper analytical redundancy algorithms such as parameter estimation and control reconfiguration are employed. Major challenges in parameter estimation include speed and accuracy of the employed algorithm. Figure 3.1 illustrates a flow diagram for a model based CBM/PHM system. As can be seen, fault detection (the grayed box) is the first major module in this process where proper detection of anomalies in the system can play a major role in the success of the subsequent modules. Therefore, it is important to ensure the fault detection module of the framework is founded on solid ground.



**Figure 3.1 Placement of fault detection module in a model-based CBM/PHM flow**

This chapter presents a new 2-step approach for improving parameter estimation with adaptive unscented Kalman filter. The enhancement in tracking speed of the unscented Kalman filter is achieved by systematically adapting the covariance matrix to the faulty estimates using innovation and residual sequences combined with an adaptive fault annunciation scheme. The proposed approach provides the filter with the advantage of tracking sudden changes in the system non-measurable parameters accurately. Results showed successful detection of reaction wheel malfunctions without requiring *a priori* knowledge about system performance in the presence of abrupt, transient, intermittent, and incipient faults. Furthermore, the proposed approach resulted in superior filter performance with less mean squared errors for residuals compared to generic and adaptive unscented Kalman filters. The contents of this chapter are organized as follows: first the proposed covariance adaptive UKF is formulated. Next stability, convergence and boundedness of the proposed algorithm are analyzed. Next, computational complexity of the proposed algorithm is analyzed and compared with other relevant methods discussed in the literature. And finally, results and

discussions are provided for the considered scenarios through extensive Monte Carlo simulations, confusion matrices, and their metrics.

### 3.1 Proposed Covariance Adaptive UKF (CAUKF)

In this section, the proposed 2-step detection and matching technique is detailed. The 2-steps comprising this algorithm are fault annunciation and filter adaptation. In the fault annunciation step, a scalar is calculated to amplify cumulative effect of residuals from the measurement outputs. In the adaption step, after a fault is annunciated, filters estimated covariance matrix is readjusted to the system's performance in such a way to accommodate a sudden change in the tracking as quickly as possible. Details of these two steps and the rationale behind their mechanisms are provided as follows.

#### 3.1.1 Fault Annunciation Metric

The posteriori estimates covariance matrix  $\mathbf{P}_{xx}$  elements show available confidence in the estimated values for the system under study. The larger the values of the elements, the less confidence in the estimations. Hence, when initiating the filter, it is common to have large values for the diagonal elements of this matrix, however, as the filter converges to the system parameters, the diagonal elements automatically decrease to show more confidence in the estimation. The challenge is when a sudden change in system parameters occurs; convergence deems slow and may not lead to fast estimation.

Based on a discussion provided in [109], that a manual adjustment of the diagonal elements can remove such effect in the filter's performance, the following approach is proposed to enhance filter's tracking speed and accuracy. The proposed method consists of two steps: Annunciation and adjustment. In the Annunciation step, an anomaly in the filter estimations is detected; followed by adjustment where state covariance matrix elements are adjusted for better performance. Scalar  $\beta$  is introduced as [109]

$$\beta = r_k^T P_{yy}^{-1} r_k \quad (3.1)$$

Bisht and Singh [109] propose a constant threshold for anomaly detection where  $\beta_0$  is the threshold based on the available data and a fault is annunciated when  $\beta > \beta_0$ . Since this can cause false alarms, an outlier detection can also be incorporated to remedy the problem. Although the results shown in their work seems promising, using a constant-valued threshold has two major limitations: (1) it requires *a priori* knowledge about the dataset since  $\beta_0$  is determined based on statistical properties of the whole dataset. (2) It can lead to divergence or poor performance when it is not pre-defined properly. In order to address these limitations, we propose an adaptation mechanism for  $\beta_0$  that does not require *a priori* knowledge; considering that for a normally distributed dataset, 99.73% of the data falls within three standard deviations ( $3\sigma$ ) from the mean [158]. At time step  $k$ , mean ( $\mu_\beta$ ) and standard deviation ( $\sigma_\beta$ ) are computed considering a memory window of size  $N$  as



$$\sigma_\beta = \sqrt{\frac{1}{N} \sum_{i=1}^N (\beta_i - \mu_\beta)^2} \quad (3.2)$$

$$\mu_\beta = \frac{1}{N} \sum_{i=1}^N \beta_i \quad (3.3)$$

which leads to calculating the adaptive threshold as

$$\beta_0 = n_\beta \sigma_\beta \quad (3.4)$$

where  $n_\beta$  denotes the adaptive threshold factor and is selected based on the statistical information about the noise in the system as discussed earlier.

### 3.1.2 Filter Adaptation Algorithm

When a fault is annunciated,  $P_{xx}$  elements need to be adjusted to compensate for the filter latency. In order to adjust  $P_{xx}$  elements, the algorithm in Table 3.1 is considered. This algorithm is a modified version of the algorithm presented in [109].

**Table 3.1 Proposed CAUKF covariance adaptation algorithm**

1:	$PP_k = P_{k-1}$ and $r_{old} = \infty$
2:	For $i = 1$ to $\text{rows}(P_{k-1})$
3:	For $j = 0.1$ to $1$ step $0.1$
3a:	Set $PP_{kii} = j$
3b:	Calculate $x_k$ using $PP_k$ using CAUKF
3c:	Calculate $r_{new} = (z_k - x_k)^T (z_k - x_k)$ with measurement $z_k$
3d:	If $r_{new} < r_{old}$ set $P_{kii} = j$ ; $r_{old} = r_{new}$
4:	Go to 3
5:	Go to 2

where  $PP_k$  is the temporary posteriori estimate covariance matrix used for finding diagonal elements of the actual  $P_k$  matrix. The following modifications compared to the algorithm in [109] are applied: (1) Residual error  $r$ , unlike in [109], is calculated without  $(P_{yy})^{-1}$  in Eq. (3.1) to reduce simulation computational time. It was found that inclusion or exclusion of such term does not affect the performance of the filter to an extent that results become unsatisfactory. (2) A grid search is employed instead of a threshold check compared to [109], since  $\beta_0$  is not known in advance and adaptively changes on-line based on the incoming data. The grid search limits for each  $PP_k$  diagonal elements are based on numerical simulations where lower bound cannot be 0 due to numerical stability and upper bound needs to be large enough to speed up the tracking speed. It was found that the upper bound value

of 1 is sufficient for the purpose of the cases studied in this thesis. For other systems and scenarios, proper value of the upper bound can be determined via simulations.

### 3.2 Stability of the CAUKF

In order to establish required stability, convergence, and boundedness of the modified CAUKF, the summary of the proof provided by Xiong et al. [159] for a modified version of the UKF is presented here, followed by the discussion on how the proposed modification in this thesis affect the so-called proof and concludes that the proof holds for the proposed modification in this thesis, given the structure of the filter is not fundamentally modified. The proof in [159] consists of two major sections as follows

#### 3.2.1 Instrumental Diagonal Matrix and Extra Positive Definite Matrix

to give a formulation for the Unscented Transform (UT) technique, instrumental time-varying matrices are introduced. Defining the estimation and prediction error as [159]

$$\tilde{x}_k = x_k - \hat{x}_k \quad (3.5)$$

$$\tilde{x}_k|_{k-1} = x_k - \hat{x}_k|_{k-1} \quad (3.6)$$

Expanding  $x_k$  by a Taylor series about  $\hat{x}_{k-1}$  gives an approximate equality [159]

$$\tilde{x}_k|_{k-1} \approx F_k \hat{x}_{k-1} + w_k, \quad (3.7)$$

where [159]

$$F_k = \left( \frac{\partial f(x)}{\partial x} \Big|_{\hat{x}_{k-1}} \right) \quad (3.8)$$

An unknown instrumental diagonal matrix  $\beta_k = \text{diag}(\beta_{1,k}, \beta_{2,k}, \dots, \beta_{M,k})$  is introduced to take these residuals into account and obtain an exact equality so that [159]

$$\tilde{x}_k|_{k-1} = \beta_k F_k \tilde{x}_{k-1} + w_k, \quad (3.9)$$

The residual of the measurement is defined as [159]

$$\tilde{y}_k = y_k - \hat{y}_k \quad (3.10)$$

Substituting noise, their covariances, and measurement update into (3.10) provides, [159]

$$\tilde{y}_k = H_k \tilde{x}_k|_{k-1} + v_k \quad (3.11)$$

In the modified UKF algorithm [159], the predicted covariance matrix is calculated as [159]

$$\hat{P}_k|_{k-1} = \sum_{i=0}^{2L} w_i (X_{i,k}|_{k-1} - \hat{x}_k|_{k-1}) (X_{i,k}|_{k-1} - \hat{x}_k|_{k-1})^T + Q_k + \Delta Q_k, \quad (3.12)$$

where  $\Delta Q_k$  is an extra positive definite matrix. The real prediction error covariance matrix is [159]

$$\begin{aligned}
P_k|_{k-1} &= E[\tilde{x}_k|_{k-1} \tilde{x}_k^T|_{k-1}] \\
&= E[(\beta_k F_k \tilde{x}_{k-1} + w_k)(\beta_k F_k \tilde{x}_{k-1} + w_k)^T] \\
&= \beta_k F_k \hat{P}_{k-1} F_k^T \beta_k + \Delta P_k|_{k-1} + Q_k,
\end{aligned} \tag{3.13}$$

where  $P_k|_{k-1}$  is the difference between  $\beta_k F_k \hat{P}_{k-1} F_k^T \beta_k$  and  $E(\beta_k F_k \tilde{x}_{k-1} \tilde{x}_{k-1}^T F_k^T \beta_k)$ . Let  $\delta P_k|_{k-1}$  be the difference between the real covariance matrix  $P_k|_{k-1}$  and the sample one  $\sum_{i=0}^{2L} w_i (X_{i,k}|_{k-1} - \hat{x}_k|_{k-1})(X_{i,k}|_{k-1} - \hat{x}_k|_{k-1})^T + Q_k$ , the calculated covariance matrix shown in (3.12) becomes [159]

$$\hat{P}_k|_{k-1} = P_k|_{k-1} + \delta P_k|_{k-1} + \Delta Q_k = \beta_k F_k \hat{P}_{k-1} F_k^T \beta_k + \Delta P_k|_{k-1} + \hat{Q}_k \tag{3.14}$$

where [159]

$$\hat{Q}_k = \Delta P_k|_{k-1} + Q_k + \delta P_k|_{k-1} + \Delta Q_k, \tag{3.15}$$

### 3.2.2 Stochastic Boundedness of Estimation Error

For analysis of the error dynamics residual, standard results regarding boundedness of stochastic processes the proof provided in [159] are recalled with the following additional remarks (1)  $\beta_k$  is an unknown instrumental diagonal matrix introduced to evaluate the error introduced by the UT. Therefore, the stability of the algorithm does not depend on the magnitude of  $\beta_k$ . Although different  $\beta_k$  may change the value of  $\lambda_{min}$ , the require inequality will still hold and its sign will not change so long as the matrix  $\hat{Q}_k$  is positive definite. In other words, if  $\hat{Q}_k \geq \hat{q}_{min} I$  is fulfilled, an upper bound on  $E[V(\xi_k)|\xi_{k-1}]$  can be obtained and the estimation error will remain bounded even for bad approximation to the nonlinear model. Since in the adaption proposed in this thesis, the matrix  $\hat{Q}_k$  is not manipulated manually, the proof provided here holds for the proposed modification in this thesis. Therefore, it can be concluded that the error remains bounded for the proposed CAUKF. (2) To ensure the stability of UKF, the matrices  $\hat{Q}_k$  need to be positive definite. From (23), as  $\Delta P_k|_{k-1}$  and  $\delta P_k|_{k-1}$  may not be positive definite matrices, extra additive matrix  $\Delta Q_t$  should be introduced as a modification to the UKF so that  $\hat{Q}_k \geq \hat{q}_{min} I$  is satisfied. Clearly, if  $\hat{Q}_k$  is sufficiently large, the required condition can always be fulfilled. This means that the modified UKF can tolerate high order error introduced during the UT by enlarging the noise covariance matrix. Furthermore, the precision of the algorithm also relates to the value of  $\hat{Q}_k$  and its is closely related to observability property of the linearized system [160]. It should be added that in the proposed CAUKF in this thesis, since the structure of the  $\hat{Q}_k$  is not manually manipulated, the proof and discussion provided herein still holds. Therefore, it can be concluded that the matrix  $\hat{Q}_k$  remains positive definite and the proposed CAUKF is stable.

### 3.3 Error Performance of the CAUKF

Cramér–Rao lower bound (CRLB), which provides a lower bound on the MSE is widely used to assess the performance of an estimator. In this section, CRLB is applied to evaluate the effectiveness of the modified UKF in [159], and further extended to the CAUKF proposed in this thesis.

#### 3.3.1 Cramér–Rao lower bound

Let  $x_0^k = \{x_0, x_1, \dots, x_k\}$  represent state vector series,  $Y_0^k = \{y_0, y_1, \dots, y_k\}$  noisy observations,  $p(Y_0^k | X_0^k)$  the joint probability density of the pair  $(Y_0^k, X_0^k)$  and  $\hat{x}_k$  a estimate of  $x_k$ .  $\hat{x}_k$  is a function of  $Y_0^k$ . The CRLB on the estimation error has the form [160]

$$P_k \triangleq E\{[x_k - \hat{x}_k][x_k - \hat{x}_k]^T\} \geq J_k^{-1}, \quad (3.16)$$

where  $J_k$  is the Fisher information matrix [159]

$$J_k = E \left[ -\frac{\partial^2 \ln p(Y_0^k, X_0^k)}{\partial x_k^2} \right]. \quad (3.17)$$

An efficient method for computing  $J_k$  recursively is given in [161].

**Proposition 1.** *The sequence  $\{J_k\}$  of posterior information matrices for estimating state vectors  $\{x_k\}$  obeys the recursion [159]*

$$J_k = D_k^{22} - D_k^{21}(J_{k-1} + D_k^{11})^{-1}D_k^{12}, \quad (3.18)$$

where

$$\begin{aligned} D_k^{11} &= E \left\{ -\frac{\partial^2 \ln p(x_k | x_{k-1})}{\partial x_{k-1}^2} \right\}, \\ D_k^{12} &= E \left\{ -\frac{\partial^2 \ln p(x_k | x_{k-1})}{\partial x_k \partial x_{k-1}} \right\}, \\ D_k^{21} &= E \left\{ -\frac{\partial^2 \ln p(x_k | x_{k-1})}{\partial x_{k-1} \partial x_k} \right\}, \\ D_k^{22} &= E \left\{ -\frac{\partial^2 \ln p(x_k | x_{k-1})}{\partial x_k^2} \right\} + E \left\{ -\frac{\partial^2 \ln p(y_k | x_k)}{\partial x_k^2} \right\}. \end{aligned} \quad (3.19)$$

From Proposition 1, it follows that

**Proposition 2.** *Assume that the nonlinear filtering is applied to system (2.43), the CRLB is given by [159]*

$$P_k \geq J_k^{-1}, \quad (3.20)$$

where [159]

$$J_k = Q_k^{-1} + H_k^T R_k^{-1} H_k - Q_k^{-1} F_k (J_{k-1} + F_k^T Q_k^{-1} F_k)^{-1} F_k^T Q_k^{-1}. \quad (3.21)$$

The proof of Proposition 2 can be found in Appendix B of [159].

### 3.3.2 The Mean Square Error of the UKF

In this subsection, the MSE for the UKF is derived and compared to the CRLB.

**Theorem 2.** *Assuming that the UKF for (2.43) is expressed as Section 2.4.1, the MSE of the UKF-based estimation method is given by [159]*

$$P_k = [\hat{Q}_k^{-1} + H_k^T R_k^{-1} H_k - \hat{Q}_k^{-1} \beta_k F_k (\hat{P}_{k-1}^{-1} + F_k^T \beta_k \hat{Q}_k^{-1} \beta_k F_k)^{-1} F_k^T \beta_k \hat{Q}_k^{-1}]^{-1} + \Delta P_k, \quad (3.22)$$

where

$$\Delta P_k = (I - K_k H_k) \left( P_k|_{k-1} - \hat{P}_k|_{k-1} \right) (I - K_k H_k)^T. \quad (3.23)$$

The proof of Theorem 2 can be found in Appendix C of [159].

**Remark 2.** Compared to the CRLB shown in (3.20), it is evident that the MSE of the UKF (3.22) depends on the instrumental matrix  $\beta_k$  and the calculated covariance matrix  $\hat{Q}_k$ , while the CRLB is related to  $Q_k$ . If the error introduced by the UT is negligible, and  $\hat{Q}_k = Q_k$ , or equivalently,  $\beta_k = I, \Delta P_k = 0$ , and (3.14) can be reduced to [159]

$$\tilde{x}_k|_{k-1} = F_k \tilde{x}_{k-1} + w_k, \quad (3.24)$$

$$\hat{P}_k|_{k-1} = P_k|_{k-1}. \quad (3.25)$$

Then the MSE of the UKF is equal to the CRLB. Thus, the UKF reaches the optimal performance if high order error is negligible and the difference between  $\hat{Q}_k$  and  $Q_k$  is small enough. Revisiting the argument provided in the previous section, since the structure of  $Q_k$  and its estimate is not manually manipulated in the proposed CAUKF in this thesis, the analysis for the error performance provided here holds and is applicable to the proposed CAUKF error performance analysis. It should be noted that the main difference in the proposed CAUKF and UKF discussed in [159] is the change in computation process from one seamless procedure to a 2-step approach, i.e. detect and match proposed in this thesis. The matching process also deals only with  $P_k$ , which is the state/parameter estimate covariance and does not affect the proofs or analyses provided in this section. Therefore, it can be concluded that the stability, convergence, and boundedness analyses provided in this section and the previous section are applicable to and hold for the proposed CAUKF in this thesis.

## 3.4 Computational Complexity of the CAUKF

In this section, the computational complexity of the proposed CAUKF is analyzed and compared to AUKF and UKF to evaluate how intense the computations would be with

the modifications proposed in this thesis.

Computational complexity theory is a division of the theory of computation in theoretical computer science focusing on classifying computational problems according to their intrinsic difficulty. A computational problem is understood to be a task that is to be solved by a computer, which also means the problem may be solved by mechanical application of mathematical steps, such as an algorithm.

A problem is regarded as inherently difficult if its solution requires significant resources, whatever the algorithm used. One of the roles of computational complexity theory is to determine the practical limits on what computers can and cannot do. However, in recent years, other fields employ this notion to evaluate performance of the algorithms developed regardless of the machines they will be run on. Therefore, in this section a brief analysis of the computational complexity for the methods discussed in this chapter is provided.

In a study done by Rhudy [162], the computational complexity of the UKF for different types of matrix square root employed in the algorithm are calculated as shown in Table 3.2 where FLOPS refers to floating point operations per second.

**Table 3.2 Matrix square root algorithm computational requirement [162]**

Algorithm	FLOPS
Diagonalization	$49n^3/3 - n^2/2 + 43n/6$
Schur	$85n^3/3$
Newton's iteration	$8n^3/3$
Denman-Beavers (DB) iteration	$4n^3$
Product DB iteration	$4n^3$
Cyclic Reduction (CR) iteration	$14n^3/3$
Cholesky	$n^3/3$

In this thesis, the Cholesky matrix square root is considered for the original UKF employed in the thesis. Therefore, the rest of the analysis and computation provided in this section is based on the fact that the original UKF with Cholesky matrix square root method is equal to  $n^3/3$  FLOPS.

In order to compute the computational complexity for the AUKF and proposed CAUKF, it is important to note that both of these methods consist of 2 parts: annunciation, and adaptation. The main difference between the two methods is in the annunciation component since both employ similar covariance matching algorithm. In the following section, the computational complexity of each component will be analyzed.

### 3.4.1 Fault Annunciation Complexity

Referring to eq. (3.1), this component is shared between the two methods AUKF

and CAUKF. Considering that each FLOPS addresses a floating operation, calculating eq. (3.1), which is in the matrix multiplication form, can be calculated as follows. Since multiplication of two  $n \times n$  matrices requires  $2n^3$  FLOPS [163], it can be seen that eq. (3.1), which is multiplication of three  $n \times n$  matrices requires  $3n^3$  FLOPS<sup>1</sup>.

Moving to the modification proposed in this thesis, there are extra terms added to the annunciation, which include Eqs. (3.2), (3.3), and (3.4). calculating each term's computational complexity yields

- (1) Eq. (3.2) is a square root of a mean of a summation. Since summation is a loop with constant increment in the step size<sup>2</sup>, the complexity of this term is  $O(n)$ .
- (2) Eq. (3.3) is a mean of a summation. Since summation is a loop with constant increment in the step size, the complexity of this term is also  $O(n)$ .
- (3) Eq. (3.4) is a multiplication. It is considered as complexity  $O(1)$ .

Since these extra steps are added together in the modification proposed in this thesis, the overall complexity would be the summation of these terms. This leads to  $n + n + 1 = 2n + 1$ . Therefore, the overall complexity term for the fault annunciation can be considered as  $3n^3 + 2n + 1$ .

### 3.4.2 Fault Adaptation Complexity

Referring to Table 3.1, the proposed algorithm consists of two inner loops with constants increments in step size. This leads to the computational complexity of  $O(n^2)$ . This is without considering the new estimates calculated in the most inner loop. That is to avoid extra calculations for the sake of comparison in this study. Therefore, the computational complexity of the adaptation algorithm, which is similarly shared between AUKF and CAUKF is  $O(n^2)$ .

### 3.4.3 Overall Complexity

Referring to the discussions provided above, it can be seen that the overall computational complexity of each algorithm is the summation of its components. That means, if individual components/blocks of an algorithm are put together to form its totality, the overall computational complexity of the final algorithm, is equal to summation of all individual consisting blocks. Therefore, the computational complexity of the algorithm discussed in this chapter can be presented as shown in Table 3.3.

**Table 3.3 Computational complexity of different estimation methods**

Algorithm	FLOPS

<sup>1</sup> <https://stackoverflow.com/questions/13900538/theoretical-and-practical-matrix-multiplication-flop>

<sup>2</sup> <http://www.geeksforgeeks.org/analysis-of-algorithms-set-4-analysis-of-loops/>

---

UKF	$n^3/3$
AUKF	$n^3/3 + 3n^3 + n^2$
CAUKF	$n^3/3 + 3n^3 + n^2 + 2n + 1$

---

It can be seen that the order of complexity, which is the order of the term with the highest power, for all three approaches is equal to 3. In other words, the order of complexity for all three approaches is  $O(n^3)$ . This means that the computation of each algorithm should roughly take the same amount of time mostly driven by the dominant term.

### 3.5 Noise Considerations

In the simulations conducted in this study, measurement and process noise are considered for the systems under simulation. In fault detection and isolation chapters, for the measurement and process noise, Gaussian zero-mean noise with a variance provided in each section is considered based on the discussions provided in [164]. In fault prognosis chapter, for the process noise, Gaussian zero-mean noise with a variance provided in each section is considered based on the discussions provided in [164] and for the measurement noise, to evaluate performance of PF under non-Gaussian noise distribution, other types of additive noise are considered as described in respective sections. The noise levels considered for each chapter under various sections of scenarios and simulations, are considered as white noise with standard deviation ( $\sigma$ ) of  $10^{-4}$  and  $10^{-5}$  for motor current and flywheel's angular velocity, respectively. These values are consistent with the values provided in [165]. In the context of this study, it is assumed that the measurement sensors for the RWs onboard satellite can provide such level of accuracy. Furthermore, it is assumed that an attitude control system (ACS) exists on the satellite and it can handle this level of accuracy for quaternion and angular velocity measurements of the satellite using available sensor systems such as magnetometers, and rate gyros [164]. The Kalman filter, which processes the incoming sensors measurements and provides a state/parameter estimate of the system states and non-measurable parameters also requires process and measurement noise matrices where these values are selected in such a way to accommodate the noise variance requirements set in [164] as listed in the tables for each simulation provided under results and discussion section of each of the Chapters 3, 4, and 5.

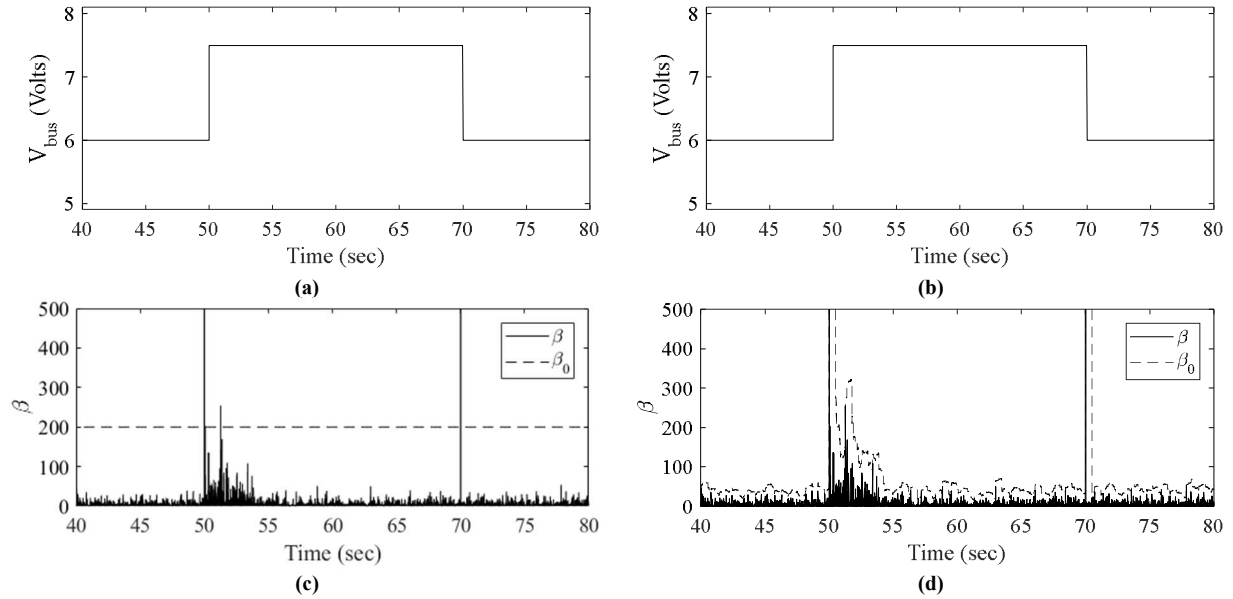
It should be noted that to further investigate validity and stability of the approaches proposed in this thesis, multiple levels of fault of introduced and investigated through extensive Monte Carlo simulations. The noise level factor is introduced as  $\varphi$  where it represents the multiplication of the proposed measurements noise variance by a factor. Furthermore, it should be noted that since motor current and flywheel's angular velocity for the RW are not of the same order of magnitude, the additive noise should be adjusted accordingly for each measurement.



### 3.6 Results and Discussion

In order to evaluate the performance of the proposed algorithm presented, the RW model was examined for abrupt, transient, intermittent, and incipient fault scenarios. And CMG unit was examined for abrupt faults. All simulations were conducted on a Dell PC XPS8700 running on an Intel® Core™ i7-4790 CPU with 3.60 GHz processing power, 8 MB cache and 16 GB of RAM. Simulations were set up as shown in Figure 3.3.

First, we examine the constant threshold defined in [109] and the proposed adaptive threshold. Figure 3.2 illustrates changes of  $\beta$  and  $\beta_0$  for a sample scenario of the RW model in Eq. (2.31) Using formulation in [109], the threshold is constant as shown in Figure 3.2(c) while in Figure 3.2(d) the threshold changes its value with changes in  $\beta$ , based on Eq. (3.4).



**Figure 3.2** Comparison of  $\beta$  and its threshold for transient fault scenario (a) and (b) between (c) fixed and (d) adaptive approaches

#### 3.6.1 Fault Types

In this section, different fault types that can be introduced to the system actuator are formulated mathematically. In this thesis, the actuator faults of the spacecraft are investigated because actuators are the components that are among the most critical and vital parts of a spacecraft. Sensor redundancy can be a low-cost solution for handling sensor faults, but for actuators it is not possible to have always extra actuators to recover from the possibility of actuator fault occurrence.

The actuator faults can be classified into four types, namely: lock-in-place, hard-over failure, float, and loss-of-effectiveness [166]. These faults may be represented as follows:

$$B_i^f u = \begin{cases} B_i u_i & \text{No Fault} \\ (B_i + f_i) u_i = B_i (I - \Gamma_i) u_i & \text{Loss of Effectiveness (LOE)} \\ 0 & \text{Float} \\ B_i u_i^{Lock} & \text{Lock in Place (LIP)} \\ B_i u_i^{min} \text{ or } B_i u_i^{max} & \text{Hard – Over Failure (HOF)} \end{cases} \quad (3.26)$$

where  $B_i$  is the healthy actuator control matrix,  $u$  is the control input,  $B_i^f$  is the faulty actuator control matrix,  $f_i = -B_i \Gamma_i$  is a negative semi-definite matrix with diagonal matrix  $\Gamma_i = \text{diag}(\gamma_{ik})$  where  $\gamma_{ik} = [\gamma_{i1} \ \gamma_{i2} \ \gamma_{i3}]^T$  includes the LOE parameters  $0 < \gamma_{ij} < 1$ . The  $U_i^{max}$  and  $U_i^{min}$  are the maximum and minimum actuations, respectively. And  $U_i^{Lock}$  ( $U_i^{min} < U_i^{Lock} < U_i^{max}$ ) is a constant level of actuation [81]. In this thesis, loss of effectiveness actuator fault is considered, which is the most common fault in spacecraft actuators. For this kind of fault, the applied control signal becomes a percentage of the desired control signal. In order to introduce faults, for RW, LOE is presented in terms of change in a system dynamics parameter value (i.e.  $k_t$  or  $V_{bus}$ ). For CMG, LOE is presented in a form of a matrix ( $F_p$ ) where effectiveness itself is multiplied by output of each unit onboard satellite. Further details of these fault presentations are provided in the respective sections for both RW and CMG where results and discussions are provided.

### 3.6.2 Monte Carlo Simulations

To further substantiate the results in each section, a comprehensive Monte Carlo simulation is provided for each fault type under two different simulation condition changes: (1) noise level, (2) severity of the fault. For each case, a table similar to Table 3.4 is provided to quantify performance metrics including precision and accuracy.

**Table 3.4 Sample Monte Carlo confusion matrix**

Identified → Actual ↓	Alarm	No Alarm
Fault	True Positive (TP)	False Negative (FN)
No Fault	False Positive (FP)	True Negative (TN)
Precision	PPV%	
Accuracy	ACC%	

For each table, simulations are conducted 100 times under each scenario. Each scenario is explained in the caption of the table, and data are recorded as described in Table 3.4 for TP, FP, FN, TN. Furthermore, PPV and ACC are calculated based on the equations provided in eq. (2.77) and eq. (2.78), respectively. TP was considered if the trigger for alarm/detection of the fault was within  $\pm 0.5$  sec of the actual fault time.

### 3.6.3 Reaction Wheels

For the following simulations the process model in Eq. (2.43) is provided as Eq. (2.31) and the measurement model in Eq. (2.43) is provided as  $g(X_k) = X_k + v_k$ . Other parameters are presented in Table 3.5 where  $I_{m \times m}$  is the identity matrix of size  $m \times m$ . To produce measured states, normally distributed zero-mean white noise with standard deviation  $\sigma_n$  is added to the nominal states; where  $\phi$  is used for noise factor, which can also be considered as an indicator for signal to noise (SNR) ratio. The step size for grid search in Table 3.1 is selected to be 0.1 based on low computational demand for the grid search and high tracking performance requirements. Other parameters including  $N, \zeta, n$  were selected based on extensive simulations. Simulation period for all cases presented is from 0 sec to 100 sec. The residual vector is defined as

$$r_x = [\tilde{I}_m - \hat{I}_m, \tilde{\omega}_m - \hat{\omega}_m] \quad (3.27)$$

where  $\tilde{I}_m$  is the measured current from the RW,  $\hat{I}_m$  is the estimated current for the RW,  $\tilde{\omega}_m$  is the measured angular velocity of the RW, and  $\hat{\omega}_m$  is the estimated angular velocity for the RW. Furthermore, parameter residuals vector is defined as

$$r_\theta = [k_t - \hat{k}_t, V_{bus} - \hat{V}_{bus}] \quad (3.28)$$

where  $k_t$  and  $V_{bus}$  are the values defined in the fault scenario and  $\hat{k}_t$  and  $\hat{V}_{bus}$  are estimated values for these parameters. It is important to note that the estimator only uses  $r_x$  residuals for calculating estimates, while  $r_\theta$  values are only used for visualisation of the results.

For Monte Carlo simulations where loss of effectiveness or severity of the fault is discussed for a particular parameter, the percentage provided as the LOE or severity is calculated as follows

$$LOE\% = severity\ \% = \frac{faulty\ value - nominal\ value}{nominal\ value} \times 100 \quad (3.29)$$

where faulty value is the value of the parameter under which the simulation was conducted and the nominal value is the value defined as the nominal performance case for the system. For RW, non-measurable parameters  $k_t$  and  $V_{bus}$  are considered while for the CMG fault parameters are directly defined as an effectiveness factor.

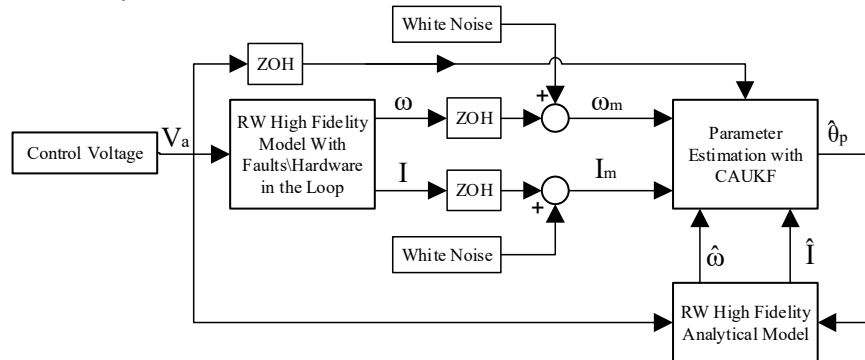


Figure 3.3 Simulation setup for performance evaluation of the proposed filter

**Table 3.5 Fault detection simulation parameters for RW**

Parameter	Value
Input Voltage ( $V_{comm}$ )	$5\sin(0.2t)$ V
Initial Input Voltage ( $V_{comm_0}$ )	0.1 V
Initial States ( $X_0 = [I, \omega]$ )	[0 A, 0 rpm]
Initial Parameters ( $\theta_0 = [k_t, V_{bus}]$ )	[0.1 Nm/A, 2 V]
Filter Parameters ( $[\alpha, \kappa, \lambda]$ )	[1, 2, 2]
Process Noise Standard Deviation ( $\sigma_Q$ )	$10^{-3}$
Measurement Noise Standard Deviation ( $\sigma_R$ )	$10^{-4}$
Process Noise Covariance Matrix ( $Q_0$ )	$\sigma_Q^2 I_{n \times n}$
Measurement Noise Covariance Matrices ( $R_0$ )	$\sigma_R^2 I_{m \times m}$
Motor Current $\sigma_n$	$[10^{-2} \sigma_R \phi]$ A
Motor Angular Velocity $\sigma_n$	$[10^{-1} \sigma_R \phi]$ rpm
Window Size ( $N$ )	50
Adaptive Fading Factor ( $\zeta$ )	60000
Adaptive Threshold Factor ( $n_\beta$ )	6
Step time ( $T_s$ )	0.01 sec

### 3.6.3.1 Case 1 – Abrupt Fault

The abrupt fault essentially occurs when a parameter in a system changes its value from one to another at a certain time. In this case, an abrupt fault scenario in the bus voltage ( $V_{bus}$ ) of the RW is considered. State and parameter estimates for this case are illustrated in Figure 3.4 while their residuals are shown in Figure 3.5. Scalar  $\beta$  and its adaptive threshold  $\beta_0$  are depicted in Figure 3.6. As can be observed from Figure 3.4, when a fault occurs, there is a jump in states and more specifically in motor current ( $I$ ) value. Consequently, a spike can be observed in respective residuals shown in Figure 3.5 at the instant of fault occurrence.

**Table 3.6 Abrupt Bus Voltage fault profile**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $K_t$ (s)	$K_t$ (Nm/A)
$t < 50$	6	$t < 50$	0.029
$t \geq 50$	7.5	$t \geq 50$	0.029

In addition, these remarks agree with the observations from Figure 3.6 where at the occurrence of the fault,  $\beta$  exceeds its adaptive threshold ( $\beta_0$ ) and  $\beta_0$  adapts itself accordingly afterwards to avoid consequent false fault annunciations.

Referring to Figure 3.7 and Figure 3.8, performance evaluations are presented for the fault scenario described in Table 3.6 for generic unscented Kalman filter (UKF), adaptive unscented Kalman filter (AUKF) proposed in [109], and covariance-matching adaptive unscented Kalman filter (CAUKF) proposed in this study. In addition, numerical results

are presented in Table 3.7, where ‘time’ refers to the total simulation time, to put comparisons in perspective. Results in Table 3.7 show that CAUKF takes nearly 4% more computation time but provides about 4% less mean squared error (MSE) for RW parameter residuals compared to AUKF for two noise levels.

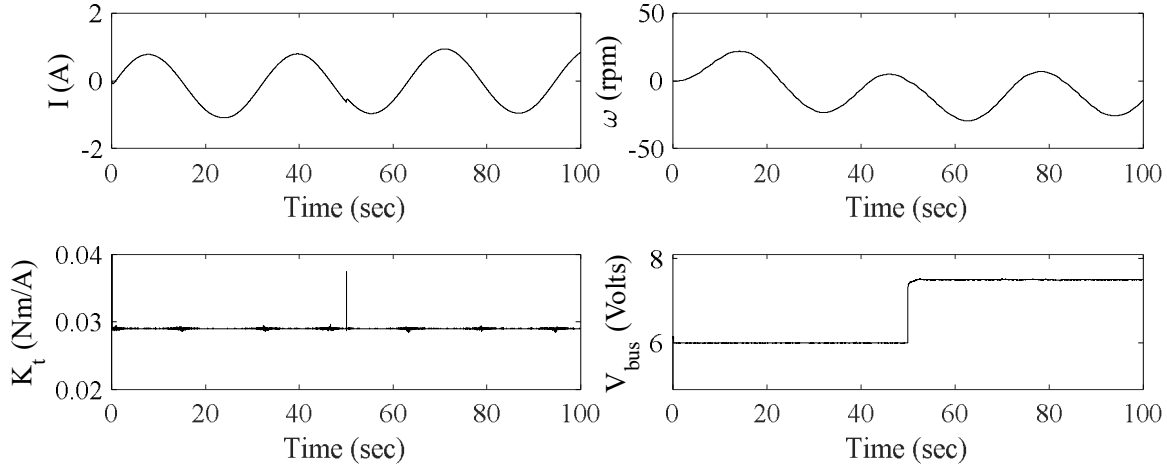


Figure 3.4 Case 1 – state and parameters estimates for abrupt fault with  $\phi=10$

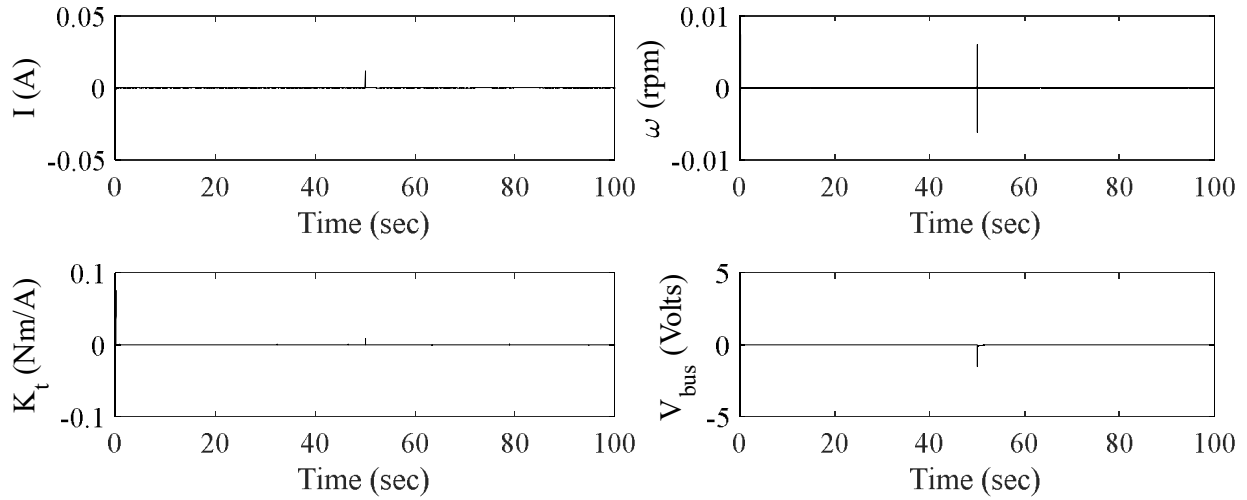


Figure 3.5 Case 1 - state and parameters residuals for abrupt fault with  $\phi=10$

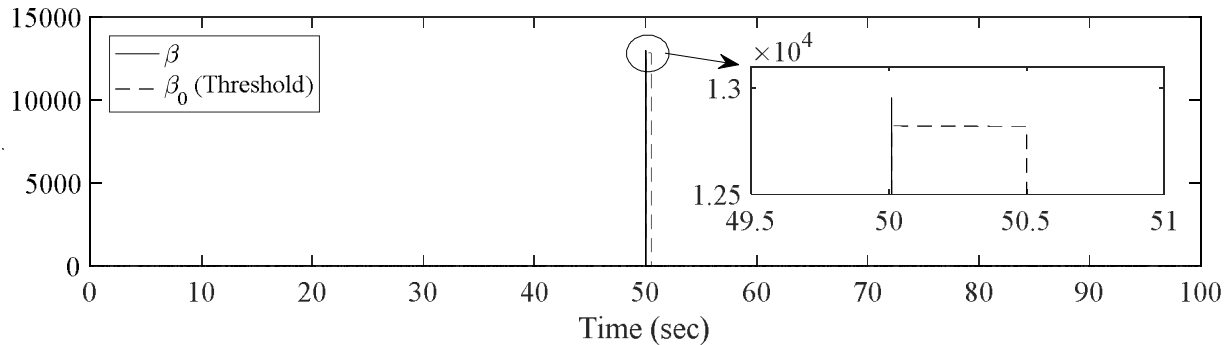
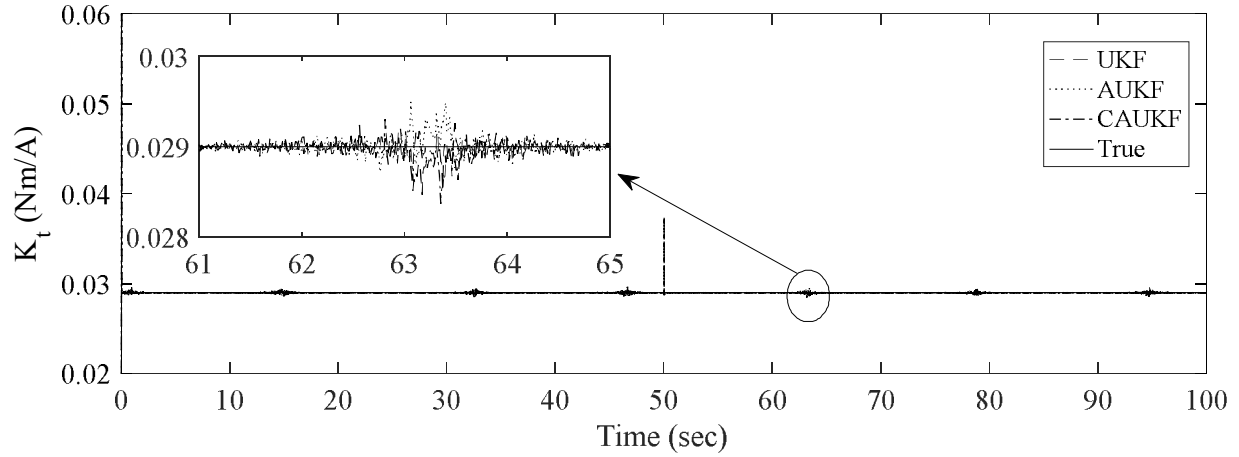
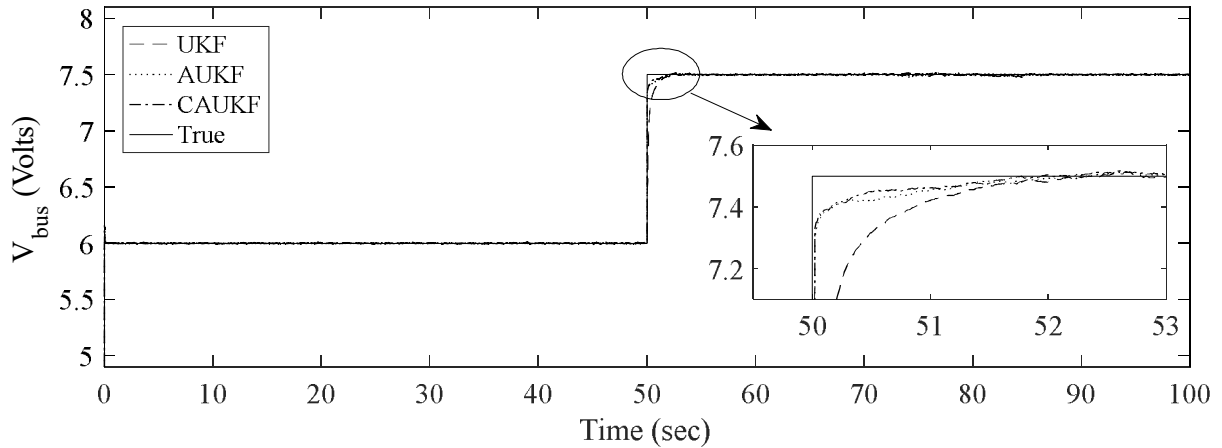


Figure 3.6 Case 1 –  $\beta$  and  $\beta_0$  for abrupt fault with  $\phi=10$

**Table 3.7 Abrupt Bus Voltage fault performance measures**

Algorithm	Noise factor $\phi = 1$			Noise factor $\phi = 10$		
	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )
UKF	2.22	$7.28 \times 10^{-7}$	$4.27 \times 10^{+2}$	2.17	$4.12 \times 10^{-7}$	$2.63 \times 10^{-3}$
AUKF	2.23	$1.01 \times 10^{-7}$	$2.53 \times 10^{-3}$	2.20	$1.01 \times 10^{-7}$	$2.53 \times 10^{-3}$
CAUKF	2.30	$1.01 \times 10^{-7}$	$2.53 \times 10^{-3}$	2.30	$1.01 \times 10^{-7}$	$2.53 \times 10^{-3}$

**Figure 3.7 Case 1 – comparison of  $K_t$  estimates for abrupt fault with  $\phi=10$** **Figure 3.8 Case 1 – comparison of  $V_{bus}$  estimates for abrupt fault with  $\phi=10$** 

For all the remaining case studies, similar responses were observed for the changes in state estimates and their residuals (see Figure 3.4 and Figure 3.5), and  $\beta$  (see Figure 3.6). This comes from the fact that input provided to all cases is similar and the only difference is in the time and severity of the faults introduced in the system. As the focus in this chapter is on parameter estimation performance of the proposed scheme compared to other methods available in the literature, only comparative figures (see Figure 3.7 and Figure 3.8) are

presented for the remaining case studies.

In the following tables, results of comprehensive Monte Carlo simulations are provided under various simulation conditions including noise level in accordance to practical experimental applications [167], [168] and fault severity. In the following MC simulation, the nominal value for  $V_{bus}$  is considered as 6 Volts and percentage of severity is calculated from eq. (3.29). furthermore, noise is reposted in the  $\phi$  form as discussed at the beginning of this section. The inception of the fault is generated randomly using the *rand*( $\cdot$ ) function in MATLAB and imposing limit between 5 sec to 50 sec.

Table 3.8 to Table 3.15 show results for Monte Carlo simulations for a 5% loss of effectiveness, which is equal to the  $V_{bus} = 6.3$  volts. Results are presented from  $\phi = 10$  to  $\phi = 200$  as an extreme case. The 5% LOE can be considered as a slight change in the  $V_{bus}$ . While the  $\phi = 200$  imposes an extreme signal to noise (SNR) ratio condition.

**Table 3.8 MC for LOE 5%,  $\phi=10$**

	Alarm	No Alarm
Fault	100	0
No Fault	4	0
Precision	96.15%	
Accuracy	96.15%	

**Table 3.10 MC for LOE 5%,  $\phi=30$**

	Alarm	No Alarm
Fault	97	3
No Fault	3	0
Precision	97.00%	
Accuracy	94.17%	

**Table 3.12 MC for LOE 5%,  $\phi=50$**

	Alarm	No Alarm
Fault	38	60
No Fault	2	0
Precision	95.00%	
Accuracy	38.00%	

**Table 3.14 MC for LOE 5%,  $\phi=150$**

	Alarm	No Alarm
Fault	0	97
No Fault	3	0
Precision	0.00%	
Accuracy	0.00%	

**Table 3.9 MC for LOE 5%,  $\phi=20$**

	Alarm	No Alarm
Fault	100	0
No Fault	10	0
Precision	90.91%	
Accuracy	90.91%	

**Table 3.11 MC for LOE 5%,  $\phi=40$**

	Alarm	No Alarm
Fault	73	26
No Fault	7	0
Precision	91.25%	
Accuracy	68.87%	

**Table 3.13 MC for LOE 5%,  $\phi=100$**

	Alarm	No Alarm
Fault	1	96
No Fault	3	0
Precision	25.00%	
Accuracy	1.00%	

**Table 3.15 MC for LOE 5%,  $\phi=200$**

	Alarm	No Alarm
Fault	2	97
No Fault	1	0
Precision	66.67%	
Accuracy	2.00%	

From Table 3.8 to Table 3.15 it can be clearly observed that the because the LOE is comparatively small, as the noise level is increased, precision and accuracy of the fault

detection algorithm decreases. It should be noted that the difference between precision and accuracy is that precision is how close the algorithm is outputting results to actual/desired results, while accuracy is how well the algorithm is performing over time. Therefore, effect of noise, as long as it does not completely overshadows the signal itself (e.g.  $\phi = 200$ ), should be more present in the precision and effect of noise overall should be more visible in the accuracy. Referring back to the results from Table 3.8 to Table 3.15, it can be observed that with lower noise levels ( $\phi = 10$  to  $\phi = 30$ ) accuracy remains almost constant, while with higher noise levels ( $\phi > 30$ ) accuracy start to degrade. Another factor to consider is the threshold adjustment for different levels of noise. In these MC simulations, threshold factor ( $n$ ) was kept constant. However, one can adjust this factor to obtain more accuracy.

**Table 3.16 MC for LOE 10%,  $\phi=10$** 

	Alarm	No Alarm
Fault	100	0
No Fault	6	0
Precision	94.34%	
Accuracy	94.34%	

**Table 3.18 MC for LOE 10%,  $\phi=30$** 

	Alarm	No Alarm
Fault	100	3
No Fault	5	0
Precision	95.24%	
Accuracy	95.24%	

**Table 3.20 MC for LOE 10%,  $\phi=50$** 

	Alarm	No Alarm
Fault	99	1
No Fault	2	0
Precision	98.02%	
Accuracy	97.06%	

**Table 3.22 MC for LOE 10%,  $\phi=150$** 

	Alarm	No Alarm
Fault	4	93
No Fault	3	0
Precision	57.14%	
Accuracy	4.00%	

**Table 3.17 MC for LOE 10%,  $\phi=20$** 

	Alarm	No Alarm
Fault	100	0
No Fault	3	0
Precision	97.09%	
Accuracy	97.09%	

**Table 3.19 MC for LOE 10%,  $\phi=40$** 

	Alarm	No Alarm
Fault	100	26
No Fault	3	0
Precision	97.09%	
Accuracy	97.09%	

**Table 3.21 MC for LOE 10%,  $\phi=100$** 

	Alarm	No Alarm
Fault	37	62
No Fault	3	0
Precision	92.50%	
Accuracy	36.27%	

**Table 3.23 MC for LOE 10%,  $\phi=200$** 

	Alarm	No Alarm
Fault	1	96
No Fault	3	0
Precision	25.00%	
Accuracy	1.00%	

Table 3.16 to Table 3.23 show results for the LOE of 10% for a range of noise levels as above. It can be observed from these tables that with 5% increase in the LOE, the precision and accuracy of the algorithm have increased slightly because now there is a bigger difference in the parameter value change.



**Table 3.24 MC for LOE 15%,  $\phi=10$** 

	Alarm	No Alarm
Fault	100	0
No Fault	5	0
Precision	95.24%	
Accuracy	95.24%	

**Table 3.26 MC for LOE 15%,  $\phi=30$** 

	Alarm	No Alarm
Fault	100	0
No Fault	2	0
Precision	98.04%	
Accuracy	98.04%	

**Table 3.28 MC for LOE 15%,  $\phi=50$** 

	Alarm	No Alarm
Fault	100	0
No Fault	2	0
Precision	98.04%	
Accuracy	98.04%	

**Table 3.30 MC for LOE 15%,  $\phi=150$** 

	Alarm	No Alarm
Fault	37	58
No Fault	5	0
Precision	88.10%	
Accuracy	37.00%	

**Table 3.25 MC for LOE 15%,  $\phi=20$** 

	Alarm	No Alarm
Fault	100	0
No Fault	5	0
Precision	95.24%	
Accuracy	95.24%	

**Table 3.27 MC for LOE 15%,  $\phi=40$** 

	Alarm	No Alarm
Fault	100	0
No Fault	3	0
Precision	97.09%	
Accuracy	97.09%	

**Table 3.29 MC for LOE 15%,  $\phi=100$** 

	Alarm	No Alarm
Fault	100	6
No Fault	5	0
Precision	94.95%	
Accuracy	89.52%	

**Table 3.31 MC for LOE 15%,  $\phi=200$** 

	Alarm	No Alarm
Fault	9	88
No Fault	3	0
Precision	75.00%	
Accuracy	9.00%	

This can be expected since the jump in the parameter change has a higher SNR compared to LEO of 5% and therefore, more observable by the algorithm. Table 3.24 to Table 3.31 show results for the LOE of 15% for a range of noise levels as above. It can be observed from these tables that with 5% increase in the LOE, the precision and accuracy of the algorithm have again increased slightly because now there is a bigger difference in the parameter value change. This can be expected since the jump in the parameter change has a higher SNR compared to LEO of 5% and therefore, more observable by the algorithm. This phenomenon can be further observed in the results shown in Table 3.32 to Table 3.39 for the LOE of 20% for a range of noise levels as above. Result in Table 3.32 to Table 3.39 further verify that the increase in LEO improves the performance of the algorithm under higher levels of noise by increasing the accuracy compared to the previous lower LOE cases. Since other types of fault including transient, intermittent, and incipient can be arguably constructed with blocks of abrupt fault, it can be argued that the same performance is expected for the following cases provided as transient, intermittent, and incipient faults.

Therefore, Monte Carlo simulations are not provided for the remaining cases of this chapter and a single case with comparison plots and tables among UKF, AUKF, and CAUKF results are provided.

**Table 3.32 MC for LOE 20%,  $\phi=10$** 

	Alarm	No Alarm
Fault	100	0
No Fault	6	0
Precision	94.34%	
Accuracy	94.34%	

**Table 3.34 MC for LOE 20%,  $\phi=30$** 

	Alarm	No Alarm
Fault	100	0
No Fault	2	0
Precision	98.04%	
Accuracy	98.04%	

**Table 3.36 MC for LOE 20%,  $\phi=50$** 

	Alarm	No Alarm
Fault	100	0
No Fault	0	0
Precision	100%	
Accuracy	100%	

**Table 3.38 MC for LOE 20%,  $\phi=150$** 

	Alarm	No Alarm
Fault	83	17
No Fault	1	0
Precision	98.81%	
Accuracy	82.18%	

**Table 3.33 MC for LOE 20%,  $\phi=20$** 

	Alarm	No Alarm
Fault	100	0
No Fault	2	0
Precision	98.04%	
Accuracy	98.04%	

**Table 3.35 MC for LOE 20%,  $\phi=40$** 

	Alarm	No Alarm
Fault	100	0
No Fault	5	0
Precision	95.24%	
Accuracy	95.24%	

**Table 3.37 MC for LOE 20%,  $\phi=100$** 

	Alarm	No Alarm
Fault	100	0
No Fault	1	0
Precision	99.01%	
Accuracy	99.01%	

**Table 3.39 MC for LOE 20%,  $\phi=200$** 

	Alarm	No Alarm
Fault	43	55
No Fault	2	0
Precision	95.56%	
Accuracy	43.00%	

### 3.6.3.2 Case 2 – Transient Fault

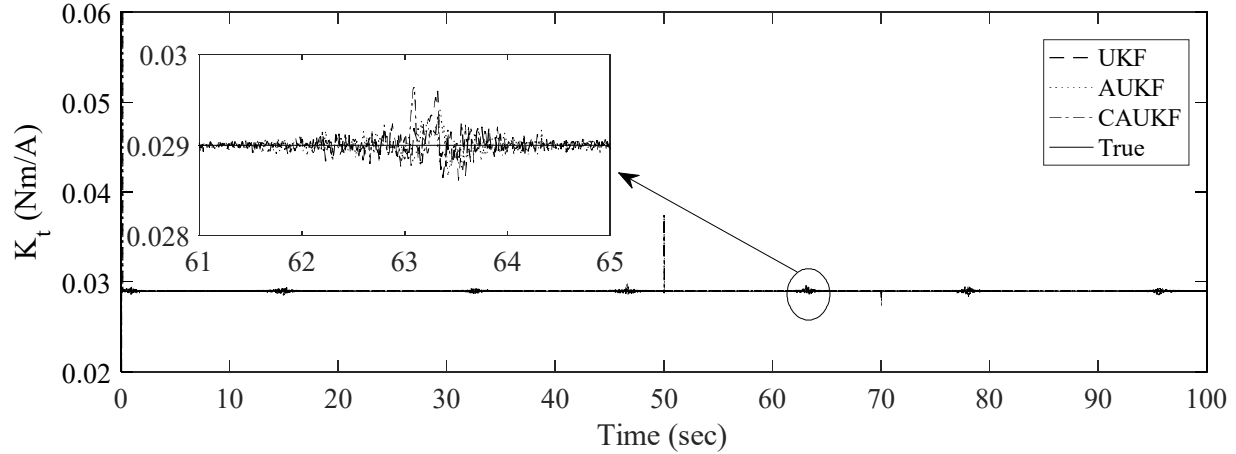
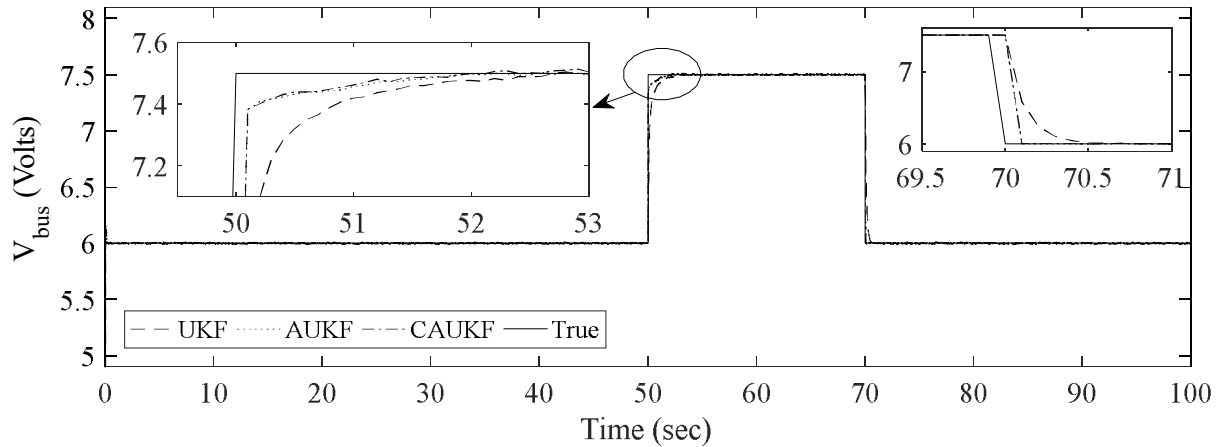
The transient fault essentially occurs when a parameter in a system changes its value from one to another for a certain period and then back to its nominal value afterward. In this case, a transient fault scenario in the bus voltage ( $V_{bus}$ ) of the RW is considered.

**Table 3.40 Transient Bus Voltage fault profile**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $K_t$ (s)	$K_t$ (Nm/A)
$t < 50$	6	$t < 50$	0.029
$50 \leq t < 70$	7.5	$50 \leq t < 70$	0.029
$t \geq 70$	6	$t \geq 70$	0.029

**Table 3.41 Transient Bus Voltage fault performance measures**

Algorithm	Noise factor $\phi = 1$			Noise factor $\phi = 10$		
	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )
UKF	2.24	$3.09 \times 10^{-4}$	$4.04 \times 10^{+1}$	2.24	$5.84 \times 10^{-8}$	$3.52 \times 10^{-3}$
AUKF	2.24	$1.01 \times 10^{-7}$	$2.99 \times 10^{-3}$	2.23	$1.20 \times 10^{-7}$	$3.87 \times 10^{-3}$
CAUKF	3.32	$1.56 \times 10^{-7}$	$2.99 \times 10^{-3}$	2.30	$1.44 \times 10^{-7}$	$3.31 \times 10^{-3}$

**Figure 3.9 Case 2 – comparison of  $K_t$  estimates for transient fault with  $\phi=10$** **Figure 3.10 Case 2 – comparison of  $V_{bus}$  estimates for transient fault with  $\phi=10$** 

Referring to Figure 3.9 and Figure 3.10, performance evaluations are presented for the fault scenario described in Table 3.40 with the same details described for Case 1. Results in Table 3.41 show that CAUKF takes nearly 4% more computation time but provides 14% less MSE for RW parameter residuals compared to AUKF.

### 3.6.3.3 Case 3 – Intermittent Fault

The intermittent fault occurs when a parameter in the system changes its value from time to time on an irregular basis. In this case intermittent in-phase and out-of-phase fault

scenarios in the bus voltage ( $V_{bus}$ ) and motor torque ( $k_t$ ) parameters of the RW are considered (see Table 3.42).

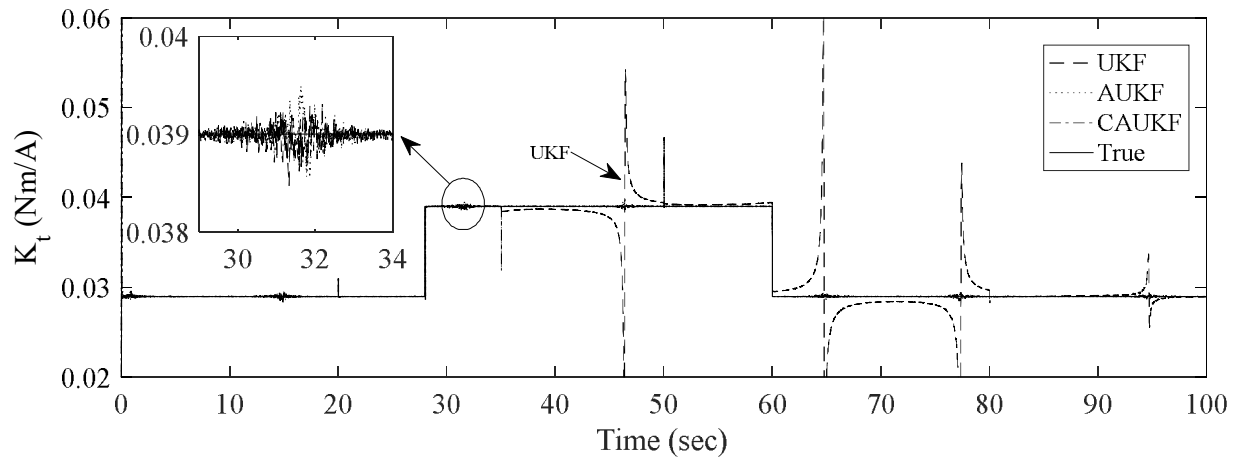
**Table 3.42 Intermittent Bus Voltage and motor torque fault profile**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $K_t$ (s)	$K_t$ (Nm/A)
$t < 20$	6	$t < 20$	0.029
$20 \leq t < 28$	7	$20 \leq t < 28$	0.029
$28 \leq t < 35$	8	$28 \leq t < 35$	0.039
$35 \leq t < 50$	6	$35 \leq t < 50$	0.039
$50 \leq t < 60$	6.5	$50 \leq t < 60$	0.039
$60 \leq t < 70$	5	$60 \leq t < 70$	0.029
$70 \leq t < 80$	8	$70 \leq t < 80$	0.029
$t \geq 80$	6	$t \geq 80$	0.029

Referring to Figure 3.11 and Figure 3.12, UKF fails to track the parameters properly during simulation (see the spikes in Figure 3.11), where AUKF and CAUKF successfully track the parameters. However, in Table 3.43, CAUKF takes nearly 5% more computation time but provides 42% less MSE for RW parameter residuals compared to AUKF.

**Table 3.43 Intermittent Bus Voltage and motor torque performance measures**

Algorithm	Noise factor $\phi = 1$			Noise factor $\phi = 10$		
	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )
UKF	2.87	$5.19 \times 10^{-5}$	$1.47 \times 10^{-2}$	2.17	$2.16 \times 10^{-4}$	$2.65 \times 10^{-1}$
AUKF	2.96	$2.11 \times 10^{-7}$	$5.98 \times 10^{-3}$	2.19	$2.42 \times 10^{-7}$	$8.81 \times 10^{-3}$
CAUKF	3.10	$2.05 \times 10^{-7}$	$5.95 \times 10^{-3}$	2.32	$2.18 \times 10^{-7}$	$6.17 \times 10^{-3}$



**Figure 3.11 Case 3 – comparison of  $k_t$  estimates for intermittent fault with  $\phi=10$**

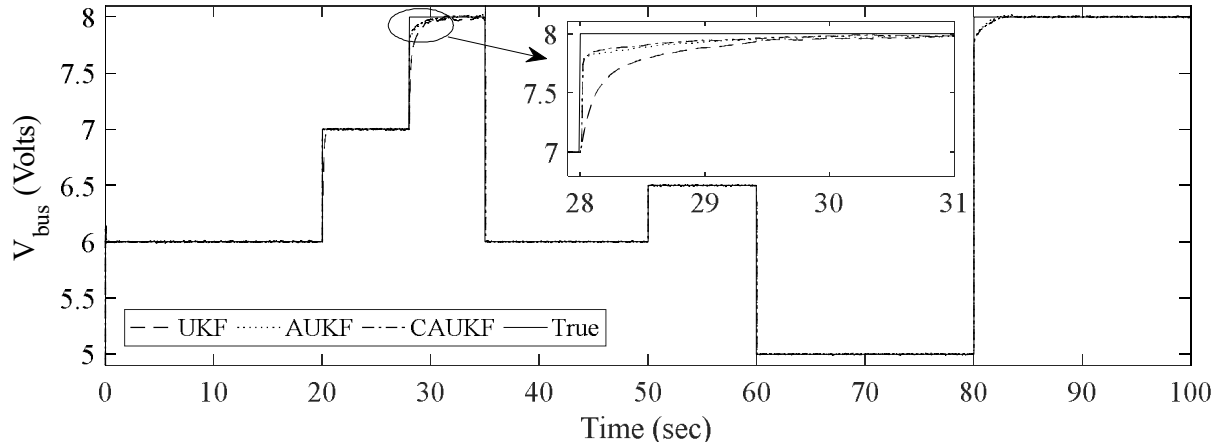


Figure 3.12 Case 3 – comparison of  $V_{bus}$  estimates for intermittent fault with  $\phi=10$

### 3.6.3.4 Case 4 – Incipient Fault

Incipient faults are fast decaying transients and very difficult to detect even for experts and may cause severe problems leading to faults and consequent outage. In this case an incipient fault in the bus voltage ( $V_{bus}$ ) of the RW is considered (see Table 3.44).

Table 3.44 Incipient Bus Voltage fault profile

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $K_t$ (s)	$K_t$ (Nm/A)
$t < 30$	6	$t < 30$	0.029
$30 \leq t < 40$	$6+0.1(t-30)$	$30 \leq t < 40$	0.029
$t \geq 40$	8	$t \geq 40$	0.029

Table 3.45 Incipient Bus Voltage performance measures

Algorithm	Noise factor $\phi = 1$			Noise factor $\phi = 10$		
	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )
UKF	2.39	$4.23 \times 10^{-8}$	$2.83 \times 10^{-3}$	2.15	$4.11 \times 10^{-8}$	$2.32 \times 10^{-2}$
AUKF	2.62	$4.00 \times 10^{-6}$	$2.94 \times 10^{-3}$	2.30	$3.53 \times 10^{-6}$	$3.82 \times 10^{-3}$
CAUKF	3.12	$3.82 \times 10^{-6}$	$2.27 \times 10^{-3}$	2.72	$4.10 \times 10^{-6}$	$2.24 \times 10^{-3}$

Figure 3.13 and Figure 3.14 illustrate parameter estimations for the fault scenario in Table 3.44. Results in Table 3.44 show that CAUKF outperforms both UKF and AUKF in terms of decreasing MSE for both RW parameters by more than ten times; however, it takes nearly 14% more computational time. It can be concluded that the proposed CAUKF could successfully perform parameter estimation for the abrupt, transient, intermittent, and incipient faults in the presented scenarios. This is substantiated further through Monte Carlo simulations provided for various levels of loss of effectiveness and various levels of noise for the abrupt scenario. As discussed earlier, abrupt MC results can be extended to other cases of transient, intermittent and incipient.

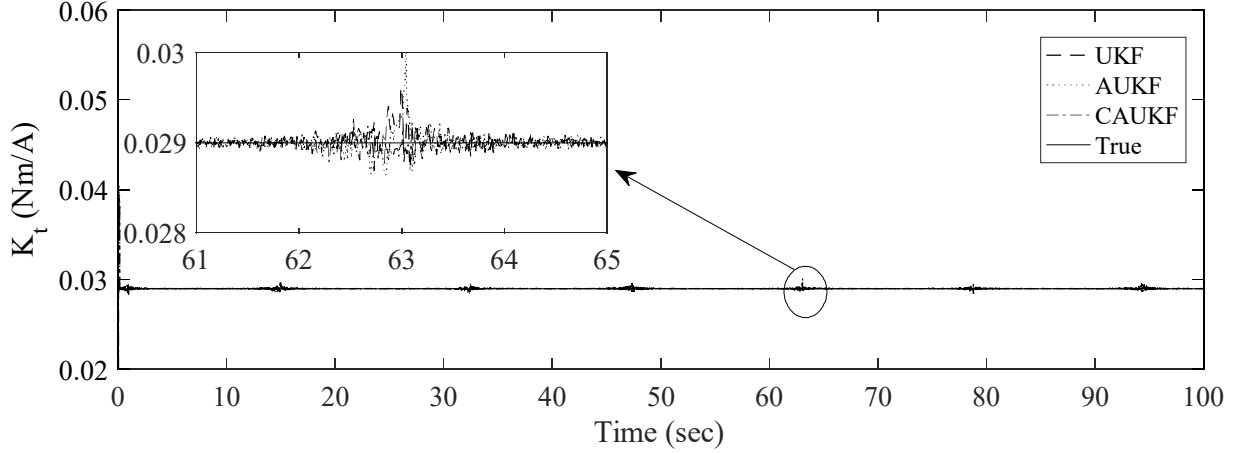


Figure 3.13 Case 4 – comparison of  $K_t$  estimates for incipient fault with  $\phi=10$

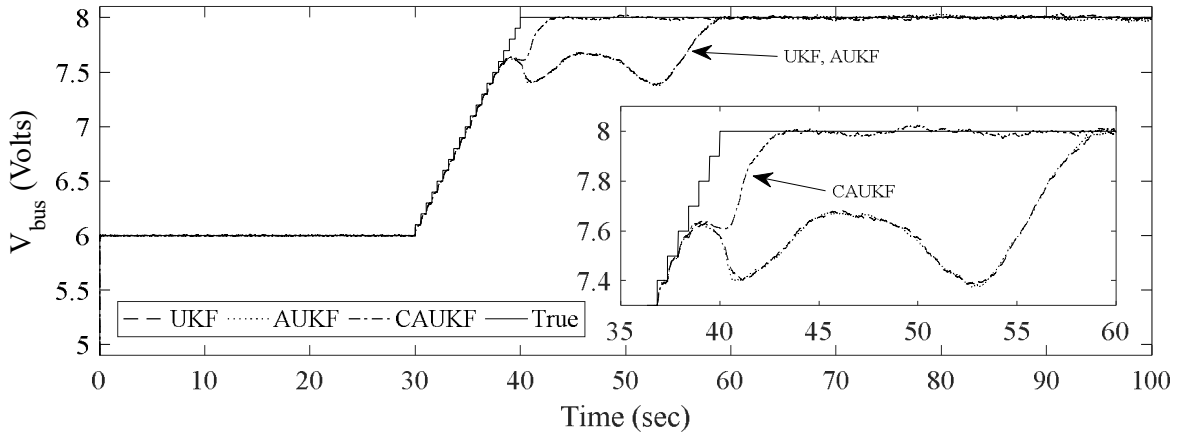


Figure 3.14 Case 4 – comparison of  $V_{bus}$  estimates for incipient fault with  $\phi=10$

### 3.6.4 Control Moment Gyros

In order to examine the performance of the proposed scheme on another actuator (as an extension of the applicability verification), fault detection for a CMG assembly mounted onboard an in-orbit satellite is presented in this section. The choice for extending the work to CMG as another actuator was made based on the fact that CMGs have more degrees of freedom due to gimbaling and can be a challenge in terms of dynamics simulations as well as effects on satellite attitude. For the CMGs, the input command is provided as the gimbal angle rates since flywheels on the CMG are of constant speed. The process model in Eq. (2.43) is provided as eqs. (2.12), (2.13) and the measurement model in Eq. (2.43) is provided as  $g(X_k) = X_k + v$ . Since there are no RWs in this setup, fault parameters are introduced in the mapping function of CMG ( $A_{CMG}$ ) in Eq. (2.40), which can be regarded as effectiveness parameters directly associated with LOE, as [14]

$$A_{CMG} = A_{CMG}F_p \quad (3.30)$$

where

$$F_p = \begin{bmatrix} f_{p_1} & 0 & 0 & 0 \\ 0 & f_{p_2} & 0 & 0 \\ 0 & 0 & f_{p_3} & 0 \\ 0 & 0 & 0 & f_{p_4} \end{bmatrix} \quad (3.31)$$

where  $F_p$  is the fault parameter matrix, and  $f_{p_i}$  corresponds to the efficiency of the  $i$ th CMG and ranges from 0 to 1. Multiplicative faults are applied because they are more likely to resemble actual gimbal motor faults. The reason that additive faults would not be realistic for this case is that they would amount to gimbal rotor constantly rotating in which case no control cannot be achieved. Therefore, each fault parameter can determine how well each CMG is performing where  $f_p = 1$  means ideal performance and  $f_p = 0$  denotes failure of the CMG and therefore, no control torque from that CMG. Since there are no control schemes incorporated in this setup, gimbal angle rates are fed to CMGs as listed in Table 3.46. The reason for different profiles for each input lies in the fact that if CMGs cancel out each others' generated torque, accumulative torque applied on the spacecraft remains zero or unchanged. If outputs of the system are not sensitive to the inputs (e.g. control command does not result in control torque), parameter identification and state/parameter estimation deems impossible. Hence, the choice for gimbal angle rates. Other simulation parameters for this section are provided in Table 3.46. It should be noted that in this system, there are 11 states and 4 parameters estimated, which is more complex compared to the RW system with 2 states and 2 parameters. In addition, this simulation considers CMGs in an assembly onboard satellite and not individually in isolation. The reason that RW simulations were conducted in isolation and not onboard a satellite, was merely caused by the direction of the studies. In the studies provided in this thesis, modeling was started with RW in isolation with extensive analysis, simulation tests, and hardware tests and data acquisition. Next, RW was considered in a RWA and was further mounted on a satellite in orbit. Next CMG was considered as a second actuator system, namely CMG because CMG was more complex than RW due to its ability to also gimbal its mounted fly wheel. Finally, formation of multiple satellites was considered to extend applicability of the proposed schemes to a more complex configuration. Therefore, the direction of the study imposed some restrictions as how to present the work in the order provided in this thesis.

The residual vector is defined as

$$r_x = [\tilde{\omega} - \hat{\omega}, \tilde{q} - \hat{q}] \quad (3.32)$$

where  $\tilde{\omega}$  is the measured angular velocity vector of the satellite,  $\hat{\omega}$  is the estimated angular velocity vector of the satellite,  $\tilde{q}$  is the measured quaternion vector of the satellite, and  $\hat{q}$  is the estimated quaternion vector of the satellite. Furthermore, parameter residuals vector is defined as

$$r_\theta = [F_p - \hat{F}_p] \quad (3.33)$$

where  $F_p$  is the defined value of fault parameter matrix diagonal elements and  $\hat{F}_p$  is the estimated value for this matrix and its diagonal elements. It is important to note that the estimator only uses  $r_x$  residuals and  $r_\theta$  values are only used for visualisation purposes.

The fault profile for each fault parameter is listed in Table 3.47 and results are illustrated in Figure 3.15-Figure 3.16, it can be observed that UKF has inferior performance compared to AUKF and CAUKF. The inferior performance of the UKF compared to others is most visible in  $fp_4$ . During simulations, it was found that including the 4th gimbal angle rate ( $\dot{\delta}_4$ ) had a major influence on the results of the estimations. Including a 4th input feed for the gimbal angle rates caused estimation performance to degrade while not including it resulted in much more accurate estimates. It can be concluded that the 4th input can cause influence on the other CMGs torques, which can result in confusion for the estimation algorithm as discussed earlier. Results in Table 3.48 also confirm the superiority of the proposed detection algorithm compared to the other methods.

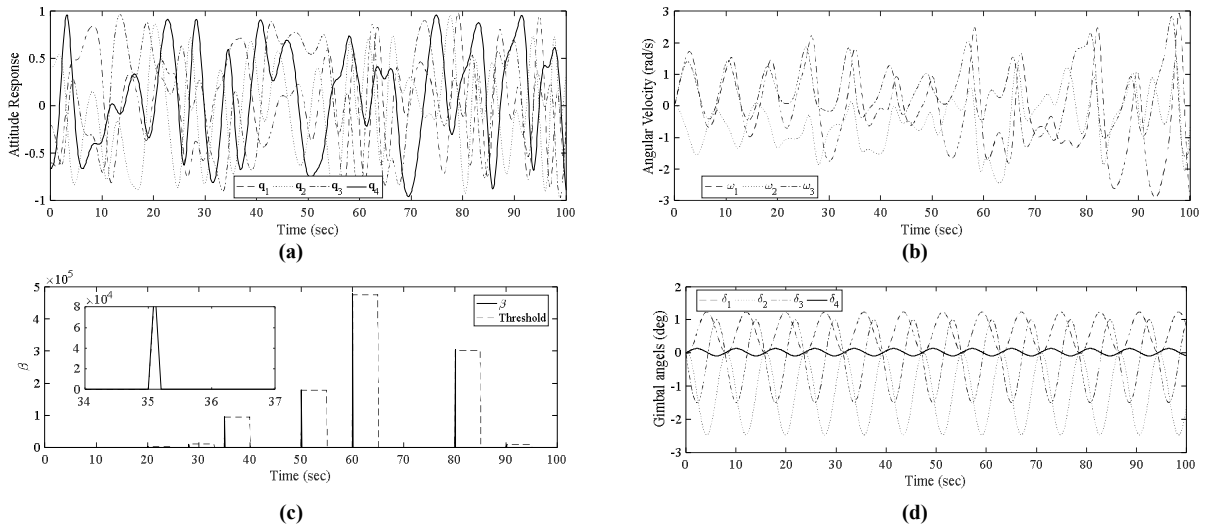
In Table 3.48, MSE is provided for each fault parameter and the RMS of the MSEs is also provided to compare the overall performance of all parameters. Another important note is the time required to adapt to the new fault parameters, which as can be observed from Figure 3.16 is agile. For the noise factor in CMG case, only  $\phi = 10$  was considered for the presentation of the results; however, other noise factors (up to 30) were also considered and results were similar to the ones presented. Overall, results show the proposed CAUKF outperforms AUKF and UKF. The limitation of the proposed method is its requirement for nearly 10% more computational time on average; however, the overall MSE was decreased by up to 10 times. It can be concluded that tracking the non-measurable system parameters in case of sudden changes was performed better when CAUKF was used compared to UKF and AUKF. Moreover, AUKF requires *a priori* knowledge about system performance from beginning to the end of simulation to determine  $\beta_0$  where unlike the proposed CAUKF.

One of the main factors to consider in the simulations for the CMGs, is the availability of less dynamics information compared to the RW dynamics. Since in the RW simulations, full dynamics of the internal motors is considered, estimation of the parameters is done more precisely and accurately. However, in the CMG simulation, the motor dynamics is not considered and the total momentum of the wheel is considered for each CMG. Furthermore, gimbal angles dynamics are considered for the simulations and they are related to the satellite attitude through dynamics equations and mapping matrices. This difference in the availability of the detailed mathematical equations for the system can play a major role where other measures are not available for compensation for un-modelled components and uncertainties in the system.



**Table 3.46 Fault detection simulation parameters CMG**

Parameter	Value
Input $\delta_1$	$0.5 \sin(0.8t) \times \pi/180$ rad/s
Input $\delta_2$	$-\sin(0.8t) \times \pi/180$ rad/s
Input $\delta_3$	$-\sin(0.8t) \times \pi/180$ rad/s
Input $\delta_4$	$0.1 \sin(\cos(0.8t)) \times \pi/180$ rad/s
$[\delta_{10} \delta_{20} \delta_{30} \delta_{40}]$	$[0 \ 0 \ 0 \ 0]$ deg
$[q_{10} q_{20} q_{30}]$	$[-0.6 \ 0.4 \ -0.2]$
$[\omega_{10} \omega_{20} \omega_{30}]$	$[0 \ 0 \ 0]$ rad/s
Initial Parameters ( $[f_{p1} f_{p2} f_{p3} f_{p4}]$ )	$[0.8 \ 0.8 \ 0.8 \ 0.8]$
$h_0 = h_{01} = h_{02} = h_{03} = h_{04}$	$0.3$ [kg. m <sup>2</sup> /s]
Filter Parameters ( $[\alpha, \kappa, \lambda]$ )	$[1, 2, 2]$
Process Noise Standard Deviation ( $\sigma_Q$ )	$10^{-3}$
Measurement Noise Standard Deviation ( $\sigma_R$ )	$10^{-4}$
Process Noise Covariance Matrix ( $Q_0$ )	$\sigma_Q^2 I_{n \times n}$
Measurement Noise Covariance Matrices ( $R_0$ )	$\sigma_R^2 I_{m \times m}$
Adaptive Fading Factor ( $\zeta$ )	60000
Adaptive Threshold Factor ( $n_\beta$ )	6
Satellite Moment of Inertia ( $J_s$ )	$\begin{bmatrix} 0.015 & 0 & 0 \\ 0 & 0.017 & 0 \\ 0 & 0 & 0.020 \end{bmatrix} \text{ [kg. m}^2\text{]}$
CMGs Moment of Inertia ( $J_w$ )	$1 \times 10^{-5} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ [kg. m}^2\text{]}$
Step time ( $T_s$ )	0.1 sec



**Figure 3.15 Simulation results for CMG fault isolation out-of-phase abrupt CMG faults with  $\phi=10$  (a) Satellite attitude (b) Satellite angular velocities (c) Scaler  $\beta$  trend (d) CMGs gimbal angles**

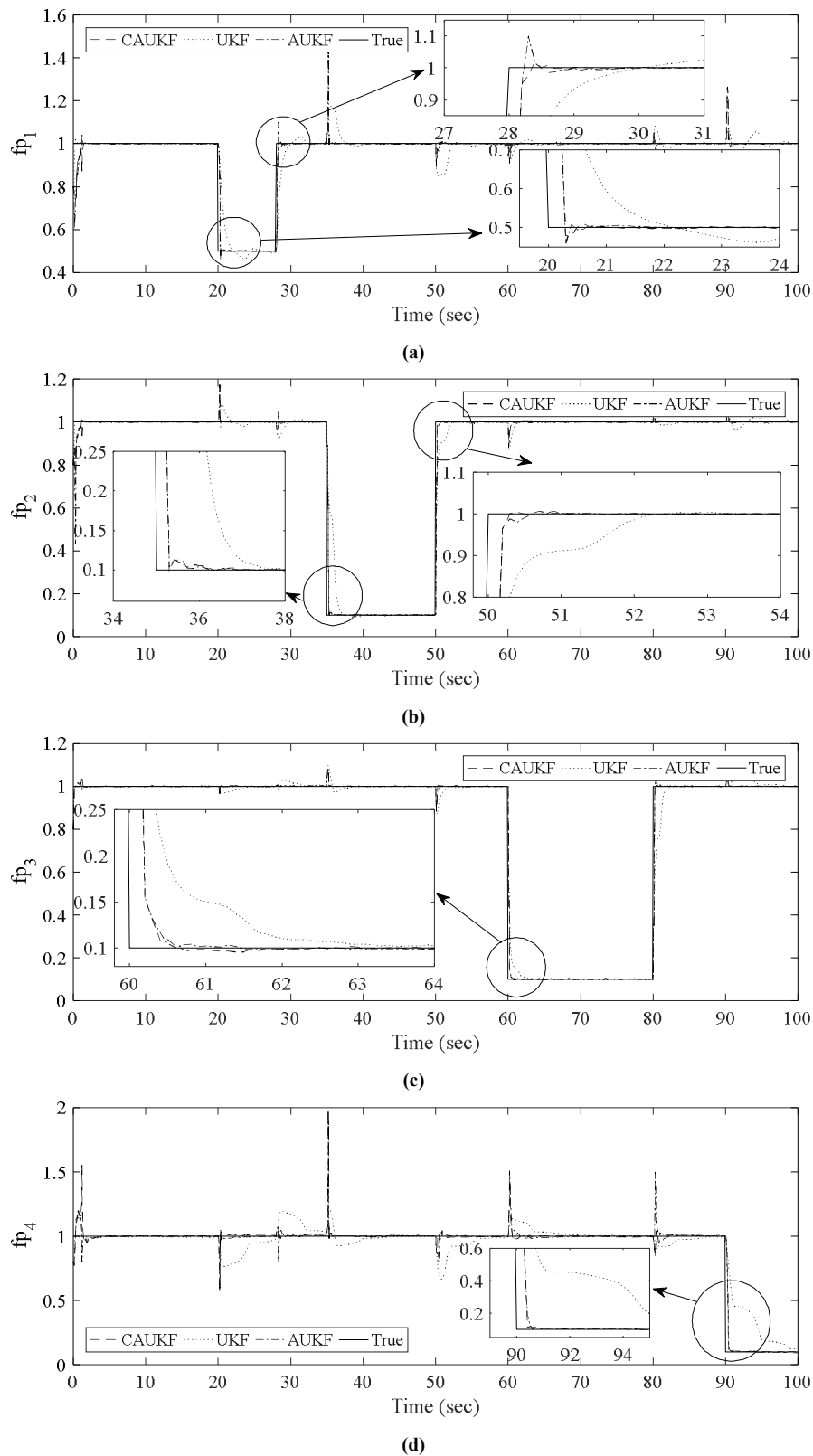


Figure 3.16 Comparison of fault parameters estimates for out-of-phase abrupt fault with  $\phi=10$  (a) CMG1 (b) CMG2 (c) CMG3 (d) CMG4

Table 3.47 CMG fault parameters out-of-phase abrupt fault profile

Time (s)	$f_{p_1}$	$f_{p_2}$	$f_{p_3}$	$f_{p_4}$
$t < 30$	1	1	1	1
$20 \leq t < 28$	0.5	1	1	1
$28 \leq t < 35$	1	1	1	1
$35 \leq t < 50$	1	0.1	1	1
$50 \leq t < 60$	1	1	1	1
$60 \leq t < 80$	1	1	0.1	1
$80 \leq t < 90$	1	1	1	1
$t \geq 90$	1	1	1	0.1

Table 3.48 CMG out-of-phase abrupt fault scenario performance measures

		Noise factor $\phi = 10$				
Algorithm	time (sec)	MSE ( $f_{p_1}$ )	MSE ( $f_{p_2}$ )	MSE ( $f_{p_3}$ )	MSE ( $f_{p_4}$ )	RMS
UKF	2.63	$2.98 \times 10^{-3}$	$4.77 \times 10^{-3}$	3.40e-03	$1.37 \times 10^{-2}$	$7.58 \times 10^{-3}$
AUKF	7.39	$1.64 \times 10^{-3}$	$2.56 \times 10^{-3}$	2.29e-03	$4.58 \times 10^{-3}$	$2.98 \times 10^{-3}$
CAUKF	7.69	$1.62 \times 10^{-3}$	$2.58 \times 10^{-3}$	2.31e-03	$4.44 \times 10^{-3}$	$2.93 \times 10^{-3}$

### 3.7 Conclusions

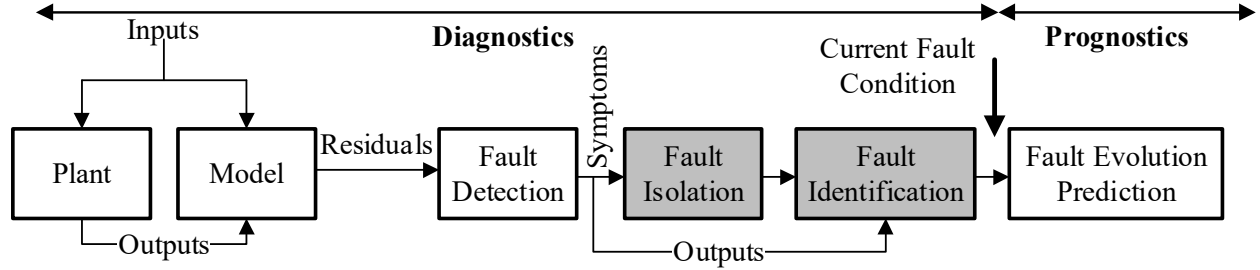
In this chapter, a new 2-step adaptive scheme was proposed to adapt states/parameters covariance matrix of the UKF in the case of sudden changes in non-measurable system parameters. The adaptive scheme was developed based on innovation and residual sequences of system states. The proposed scheme was tested on an RW system in isolation, as well as on a 4SGCMG configuration onboard satellite, to evaluate its performance. For the RW, numerical simulations comprised four fault scenarios including abrupt, transient, intermittent, and incipient. For each fault scenario, two noise levels were considered for detailed comparison as well as extensive MC simulations under multiple LOE % and  $\phi$ . Results showed superior performance compared to the UKF and AUKF. Parameter tracking for non-measurable system parameters was enhanced by up to 10 times based on MSE measures at the cost of 10% more computational time. For the CMG, simulations comprised only out-of-phase abrupt faults in all four CMGs, the in-phase fault detection is also possible using the proposed scheme, however, that scenario will be discussed in the fault isolation section where it fits more appropriately. The proposed scheme can be implemented where *a priori* knowledge about system performance is not available and computational resources can allow for the additional calculation requirements. Therefore, small satellites with could be a reasonable target to implement such method.

## CHAPTER 4

# Fault Isolation

The problem of promptly and accurately finding the root cause of a fault in a complex system has been the center of attention for engineers, dynamists, and control experts.

When a fault is detected, it is important to find the source of error and consequently, take remedial actions before a complete failure results in mission abortion. This constitutes the second major module in the CBM/PHM framework as can be seen in Figure 4.1, where the gray boxes show fault isolation and identification modules. Due to highly non-linear nature of satellite dynamics in space and effect of actuator faults on its attitude, it is vital to employ approaches that provide reasonable solutions using limited resources.



**Figure 4.1 Placement of fault isolation module in a model-based CBM/PHM flow**

RWs are prone to fault or failure mainly due to 1) Friction torque increase due to increase in bearing temperature or failure of lubrication system and 2) BEMF increase caused by a surge in the control circuit voltage [62]. An RW yields a highly non-linear dynamic system. The system becomes more complex when RWs are placed in an assembly (RWA), coupled with the dynamics and attitude of the satellite in orbit. If only limited measurements are available to the FDI scheme, it becomes a challenging task to detect faults and isolate their root cause.

In this chapter, a new 2-step methodology for fault isolation of RWs (standard four-wheel/pyramid configuration) onboard satellite is proposed based on a reduced variation of a hierarchical multiple model approach in conjunction with Bayes' probability distribution and adaptive windowing. Multiple models are considered for faulty variations of the nominal system using unscented Kalman filters for fault isolation. Results show over 99% successful isolation for various fault scenarios including single fault, multiple in-phase faults, and multiple faults in sequence.

### 4.1 Proposed Methodology

The proposed fault isolation methodology in this study uses hierarchical approach

towards fault isolation. The hierarchical approach ensures resources are allocated for computation only when required. The proposed scheme is shown in Figure 4.2 where at each level, only certain information is passed to the subsequent level to isolate the location of a fault. At the highest level of this hierarchy, there is a centralized FDI unit supervising a set of 4 spacecraft. It is important to note that the number of satellites can be expanded without the loss of generality. However, in this study 4 satellites were considered to be consistent and use the same terminologies as the 4 actuators discussed at system level. At this level, all spacecraft send their attitude information to the FDI unit including their quaternions and angular velocities. Using the nominal model and parallel UKFs running to generate residuals, when a fault is announced, a window of residuals is created to isolate the faulty spacecraft. After faulty spacecraft are identified, respective satellite attitude information is passed to the fault isolation scheme. It is worthy to mention that limited measurements are available to the isolation scheme including only satellite quaternions and angular velocities. There is no access to RW output measurements at the isolation level. Next, based on the RW assembly configuration and likelihood of each model to the current state of the system, the faulty RW unit is determined ( $RW_i, i = 1,2,3,4$ ). Finally, to further locate source of the fault in the actuator, parameter estimation is employed to determine the faulty parameter ( $k_t$  or  $V_{bus}$ ) for RW and fault parameter for CMG.

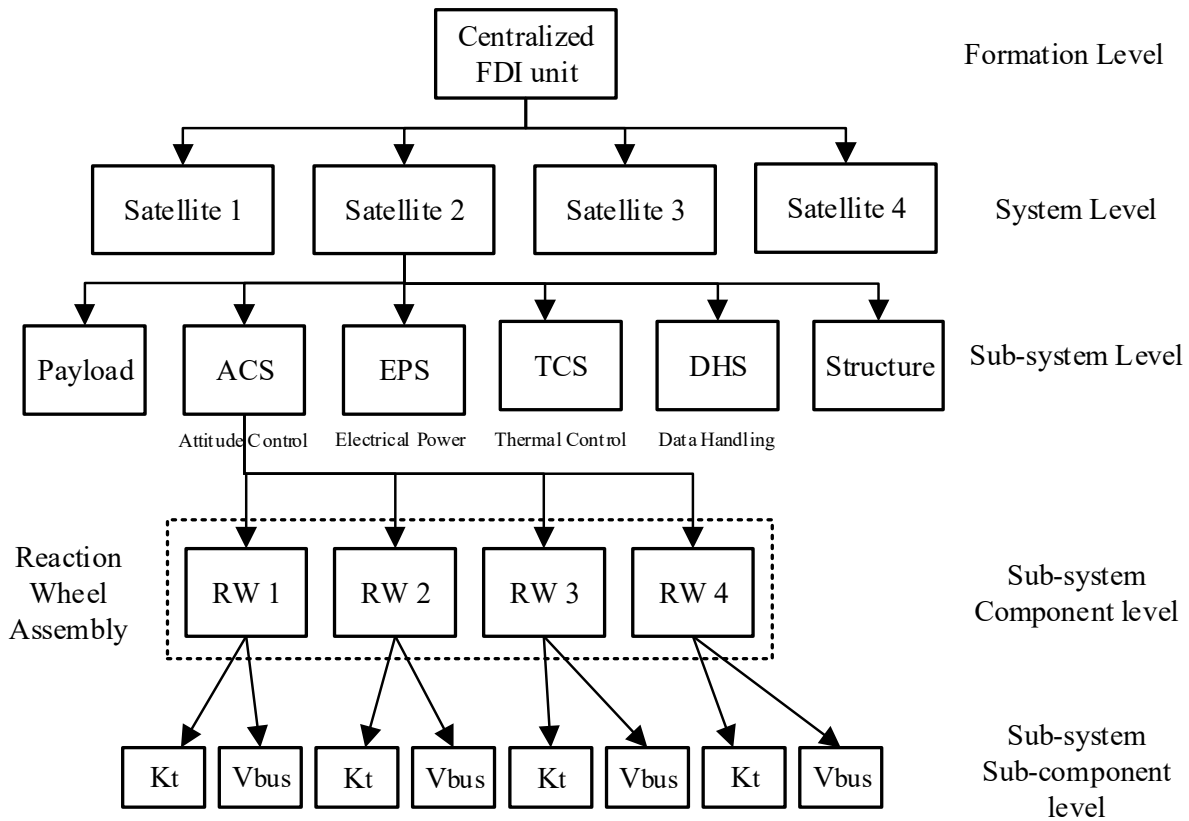


Figure 4.2 Proposed hierarchical approach towards fault isolation

#### 4.1.1 Fault Detection

The first step of the isolation is to detect an anomaly in the system's behavior. In order to detect anomalies in the system behavior, scalar  $\beta$  is defined as [169]

$$\beta = r_k^T r_k \quad (4.1)$$

where  $r_k$  is the residual at time step  $k$ . At the formation level, the residual vector is defined as

$$r_{x_i} = [\tilde{\omega}_i - \hat{\omega}_i, \tilde{q}_i - \hat{q}_i] \quad (4.2)$$

where  $\tilde{\omega}_i$  is the measured angular velocity vector of the  $i$ th satellite in the formation,  $\hat{\omega}_i$  is the estimated angular velocity vector of the  $i$ th satellite in the formation,  $\tilde{q}_i$  is the measured quaternion vector of the  $i$ th satellite in the formation, and  $\hat{q}_i$  is the estimated quaternion vector of the  $i$ th satellite in the formation. It was also observed that using only one set of the residual vector elements (i.e.  $\tilde{\omega}_i - \hat{\omega}_i$  or  $\tilde{q}_i - \hat{q}_i$ ) can also suffice for the purpose of formation level isolation.

At the system level, the residual vector is defined as

$$r_x = [\tilde{\omega} - \hat{\omega}, \tilde{q} - \hat{q}] \quad (4.3)$$

where  $\tilde{\omega}$  is the measured angular velocity vector of the satellite,  $\hat{\omega}$  is the estimated angular velocity vector of the satellite,  $\tilde{q}$  is the measured quaternion vector of the satellite, and  $\hat{q}$  is the estimated quaternion vector of the satellite.

And finally at the sub-system level (i.e. RW), the residual vector is defined as

$$r_x = [\tilde{I}_m - \hat{I}_m, \tilde{\omega}_m - \hat{\omega}_m] \quad (4.4)$$

where  $\tilde{I}_m$  is the measured current from the RW,  $\hat{I}_m$  is the estimated current for the RW,  $\tilde{\omega}_m$  is the measured angular velocity of the RW, and  $\hat{\omega}_m$  is the estimated angular velocity for the RW. Furthermore, parameter residuals vector is defined as

$$r_\theta = [k_t - \hat{k}_t, V_{bus} - \hat{V}_{bus}] \quad (4.5)$$

where  $k_t$  and  $V_{bus}$  are the values defined in the fault scenario and  $\hat{k}_t$  and  $\hat{V}_{bus}$  are estimated values for these parameters. It is important to note that the estimator only uses  $r_x$  residuals for calculating estimates, while  $r_\theta$  values are only used for visualisation of the results.

A constant threshold for anomaly detection ( $\beta_0$ ) is used based on the available data and, a fault is annunciated when  $\beta > \beta_0$ . This choice was made since the focus of this chapter was on isolation rather than detection. At the formation level, a UKF runs in parallel to each spacecraft to generate residuals and detect faults. When fault is detected, a window of cumulative residual is created to isolate faulty spacecraft, residuals are calculated until a certain threshold is reached and faulty spacecraft are identified based on the description in Table 4.1. Next, in the system level, referring to Figure 4.3, a UKF runs in parallel to the hardware model of the spacecraft at all times to generate residuals. Scalar  $\beta$

is calculated at each time step and fault isolation module is activated once  $\beta$  passes its threshold  $\beta_0$ .

#### 4.1.2 Likelihood-based Isolation

The main advantage of the proposed approach is its ability to generate a likelihood distribution of all possible scenarios for the faulty system and use it to generate posterior probability distribution of each case scenario. The term all possible scenarios means the number of possible cases to choose 1,2,3 or 4 actuators out of 4 actuators. This can be calculated for any number of satellite using the  $k$ -combination rule or the binomial coefficient as

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)\dots(n-k+1)}{k(k-1)\dots 1} \quad (4.6)$$

Using this for any arbitrary number of satellites summation of all possible combination is the maximum number of cases to be considered. For 4 options (e.g. satellites or actuators) this can be calculated as

$$\binom{4}{1} + \binom{4}{2} + \binom{4}{3} + \binom{4}{4} = 4 + 6 + 4 + 1 = 15 \quad (4.7)$$

where if adding the case of no faulty unit comes to 16 possible cases.

Posterior probability is the probability of the parameter  $\theta$  given the evidence  $X: p(\theta|X)$ , In contrary to likelihood function, which is the probability of the evidence given the parameters  $p(X|\theta)$ . If we have a prior belief that the probability distribution function is  $p(\theta)$  and we obtain observations  $x$  with the likelihood  $p(x|\theta)$ , the posterior probability function then becomes [170, pp. 21–24]

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} \quad (4.8)$$

which can be rephrased in a simpler language as

$$posterior \propto prior \times likelihood \quad (4.9)$$

Using this concept, we assume there is an initial probability distribution for all possible cases of fault in the system (see Table 4.1). The initial probability distribution can be of any form, however, in this study, we have only examined uniform and normal distributions. Further discussions for the effect of initial distribution on the performance of the algorithm are provided in the results and discussion section. To be able to refer to each fault case more conveniently, all possible cases are assigned with a number in Table 4.1 where the faulty wheel number is defined in Figure 2.8 for each RW assembly configuration.

**Table 4.1 Fault scenarios for RW/CMG assemblies and satellite formation**

Scenario No.	Faulty Units	Scenario No.	Faulty Units
0	None	8	2,3
1	1	9	2,4
2	2	10	3,4
3	3	11	1,2,3
4	4	12	1,2,4
5	1,2	13	1,3,4
6	1,3	14	2,3,4
7	1,4	15	1,2,3,4

The overall scheme for fault isolation is shown in Figure 4.3. At each time step, the control input is fed into two models. One is the satellite attitude hardware model, and the other one is the mathematical model running in parallel to detect faults in the system and subsequently conduct isolation. Since there was no access to the hardware in this study, we assume outputs of the hardware model are measurements from the hardware. State estimates are obtained using UKF from the mathematical model before generating residuals to ensure robustness of the algorithm to uncertainties and disturbances.

Comparison is made between outputs of mathematical model and hardware model to generate residuals as follows

$$r_k = \sum_{j=1}^7 |x_{k,j} - \hat{x}_{k,j}| \quad (4.10)$$

where  $\mathbf{x}$  is the state vector from the satellite models. This state vector includes  $\dot{\mathbf{x}} = [\dot{\mathbf{q}} \ (\dot{\boldsymbol{\omega}}_{BL}^B)]^T$  where  $\mathbf{q}$  is the quaternion for the satellite attitude and  $\boldsymbol{\omega}_{BL}^B$  is the satellite angular velocity described in Eq. (2.15).  $\mathbf{x}$  refers to hardware model states and  $\hat{\mathbf{x}}$  refers to mathematical model states, which runs in parallel to the actual model. It is important to note that  $\mathbf{x}$  is a vector with 7 states in it. Therefore, a summation of all differences is generated for  $r_k$ . At each time step Using Eq. (4.1) scalar  $\beta$  is calculated and when it is greater than a threshold  $\beta_0$ , fault isolation scheme considers the state of the system as faulty and activates the isolation module. This measurement/state set is used for both formation level and system level, however, it should be noted that at formation level due to direct availability of system measurements, it is possible to use only satellite angular velocity instead of full quaternion plus angular velocity set as discussed in [82]. This is important to note that one of the major differences between the proposed approach in this thesis and other MM based approaches (e.g. Nareri's [171] proposed MM UKF-based approach in his PhD dissertation under Chapter 3), is that the proposed approach in this thesis only runs parallel models when fault is announced while others run multiple models at all time. The limitation, however, is that while isolating the fault, this algorithm cannot isolate another fault.



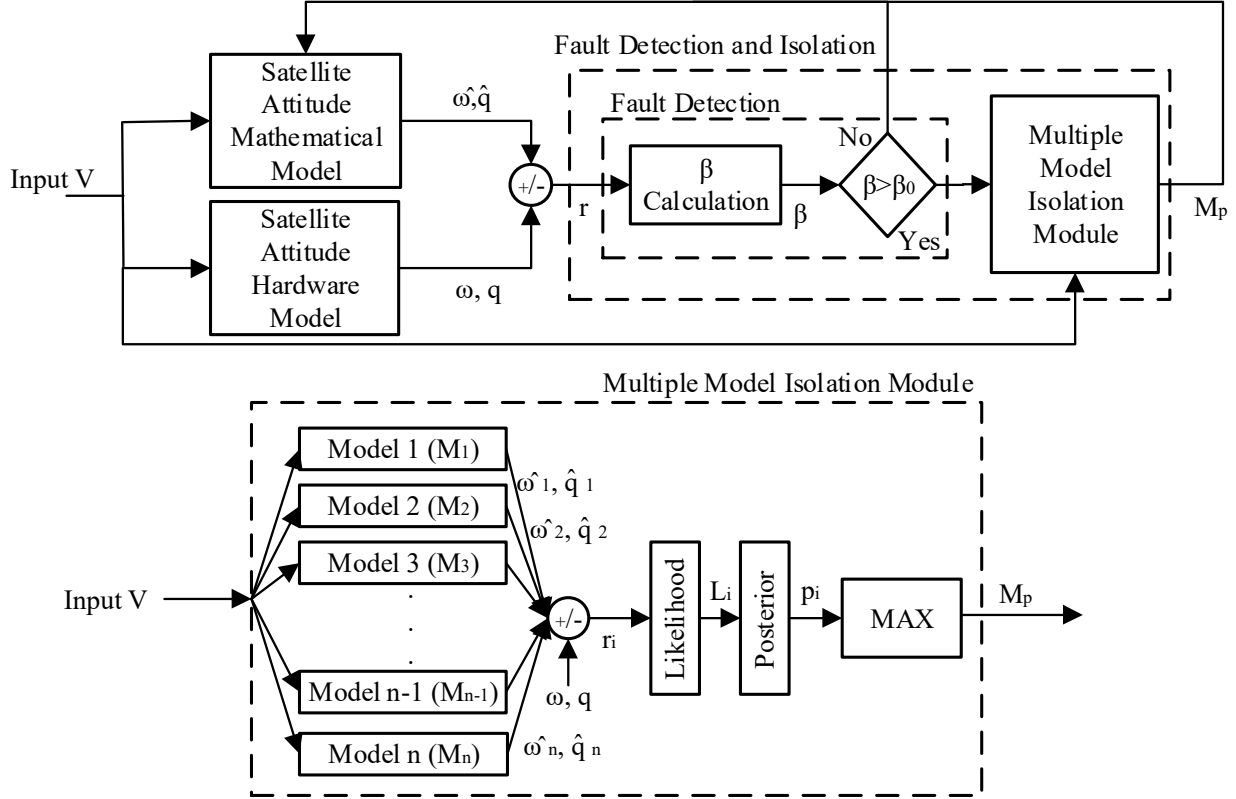


Figure 4.3 Proposed fault isolation scheme based on likelihood and probability distribution

In this study  $\beta_0 = 0.06$  was chosen based on extensive simulations and can vary for different applications. It should also be noted that adaptive threshold can also be implemented from definitions in the previous chapter, however, in this chapter since the focus is on fault isolation, adaptive threshold was not considered. It is important to note that in this approach only one UKF is running before a fault is annunciated unlike MM methods where multiple models run at all times. When the threshold is passed, multiple UKFs based on faulty models of the system where each model corresponds to a particular fault scenario (see Table 4.1) are activated and system input is fed into all UKFs in parallel. The latest nominal states before fault annunciation are used as initial states for multiple UKFs to generate new states. This means if fault is detected at time step  $k$  and the states of the system are  $\mathbf{x}_k$  at this time step, multiple models use  $\mathbf{x}_{k-1}$  for their initial states. The estimates of each UKF corresponding to a particular faulty model is then compared with the “actual model” outputs and mean square error (MSE) is calculated for each residual as follows

$$r_i = \sum_{j=1}^7 |x_{k_j} - \hat{x}_{k_j}|^2 \quad (4.11)$$

where  $i$  corresponds to each faulty case scenario listed in Table 4.1. The farther we get from the inception of the fault, for models that do not describe the current state of the system,

the residual becomes larger. In order to accommodate this accumulation of error in residuals, a window is defined for each model to store specific number of residuals over a period of time as follows

$$r_{i,j} = r_{i,j-1} + \sum_{k=j}^7 |x_{k_j} - \hat{x}_{k_j}|^2 \quad (4.12)$$

where  $j$  is the index in the residual window and  $r_{i,0}$ , which is the initial value for the residual, is equal to 0. Based on this MSE, a likelihood function is generated for the probability of each case being the current state of the system. Presumably, the smaller the MSE the closer the current state of the system to the corresponding model. In order to generate the likelihood and consequently the posterior probability function, the following steps are taken

**1) Find maximum MSE among cases**

$$M = \max(r) \quad (4.13)$$

**2) Recompute MSE compared to maximum**

$$r_i = M - r_i \quad (4.14)$$

**3) Compute summation of the residuals**

$$S = \sum r_i \quad (4.15)$$

**4) Compute likelihood**

$$L_i = \frac{r_i}{S} \quad (4.16)$$

**5) Compute posterior probability**

$$p_i = p_{i-1} \times L_i \quad (4.17)$$

These steps provide a likelihood and posterior distribution at each step. As mentioned, initial distribution can be of any form; however, in this study, we only show results for uniform and normal initial distributions.

### 4.1.3 Adaptive Windowing

Since it is computationally heavy to store a large window of residuals for all cases and not efficient to run parallel models of the system for a long period of time, adaptive windowing is employed to finalize the isolation procedure. There is maximum window size is considered  $N_{w_{max}}$  to avoid overflow of memory. At each iteration in the isolation scheme, the  $i$ th case corresponding to the maximum  $p_i$  is the case closest to the current state of the system. However, we need a confidence level to make sure the case with maximum  $p_i$  is actually the current state of the system. When calculating  $p_i$  at each step, the following logic is used to determine the result of isolation

**Table 4.2 Logic for determining output of the isolation scheme**


---



---

1:	$i_{out} = \text{index of } (Max(p_i))$
2:	if $j > N_{wmax}$ then output $i_{out}$ else
3:	if $Max(p_i) > C_{out}$ then output $i_{out}$
4:	end
5:	end

---

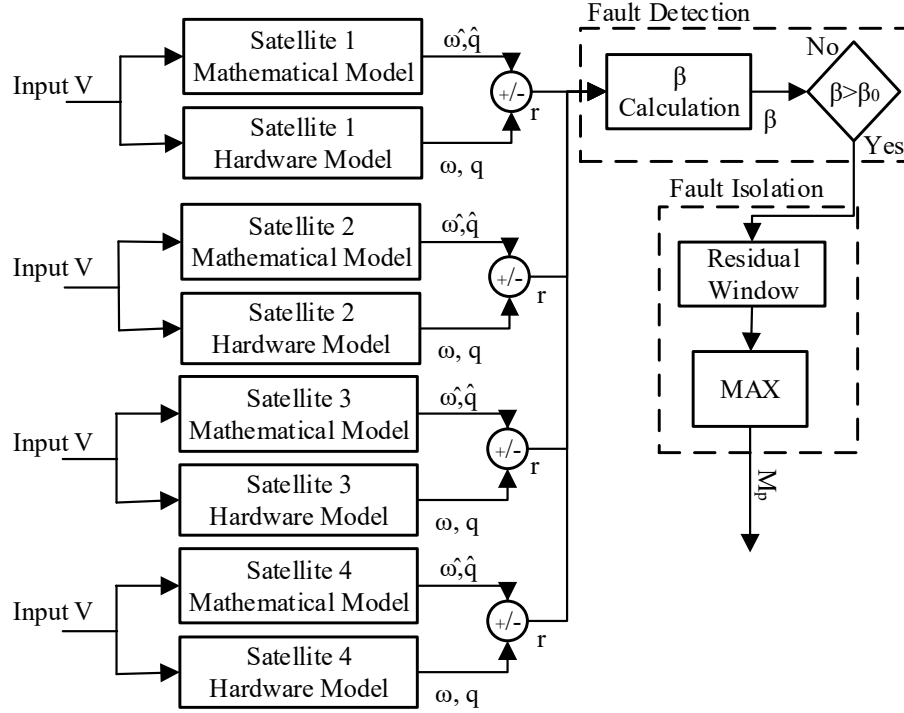
where  $i_{out}$  is the index of the case with maximum probability of being the current state of the system and  $C_{out}$  is a threshold for confidence in the output of the logic. In this study  $C_{out} = 90\%$  and  $N_{wmax} = 60$ . The 60 for window was chosen based on the worst case scenario results and a safety factor. For the worst case scenario, it took the algorithm 42 iterations to isolate the case properly, adding 12 more iterations as a safety net resulted in 60 iterations as the maximum threshold for number of iterations in the isolation process. Employing this logic in the adaptation of the isolation scheme, ensures that computation of the likelihood and posterior probability is conducted until satisfactory confidence in probability distributing is obtained or maximum number of attempts has reached. The latter is particularly useful when symmetry in the system may causes probability distribution to lean towards two or more cases at the same time.

## 4.2 Results and Discussion

Numerical simulations were conducted to evaluate the performance of the proposed isolation scheme on a RWA and a CMG assembly. Simulations were performed on an Intel® Core™ i7-4790 CPU with 3.60 GHz power PC and 16.00 GB RAM.

### 4.2.1 Formation Level

At formation level, control input to all satellites are the same as and each individual satellite is setup the same way as discussed in Section 0. The only difference is that at this level, instead of a single satellite, there is an array of four satellites operating in parallel. An actual model and an analytical model is considered for each satellite and residuals are generated as the difference between the outputs of the two. For each satellite, a UKF runs in parallel to estimate the states of the system and feed them to the residual generation. When a threshold is passed, a window of residuals is collected until a threshold is passed or window is out of memory based on the logic described in Table 4.2. The overall simulation setup can be considered as shown in Figure 4.4 where all satellites send their info to the central FDI unit and once a fault is annunciated, the isolation scheme identifies the faulty. All satellites are setup with Table 4.5



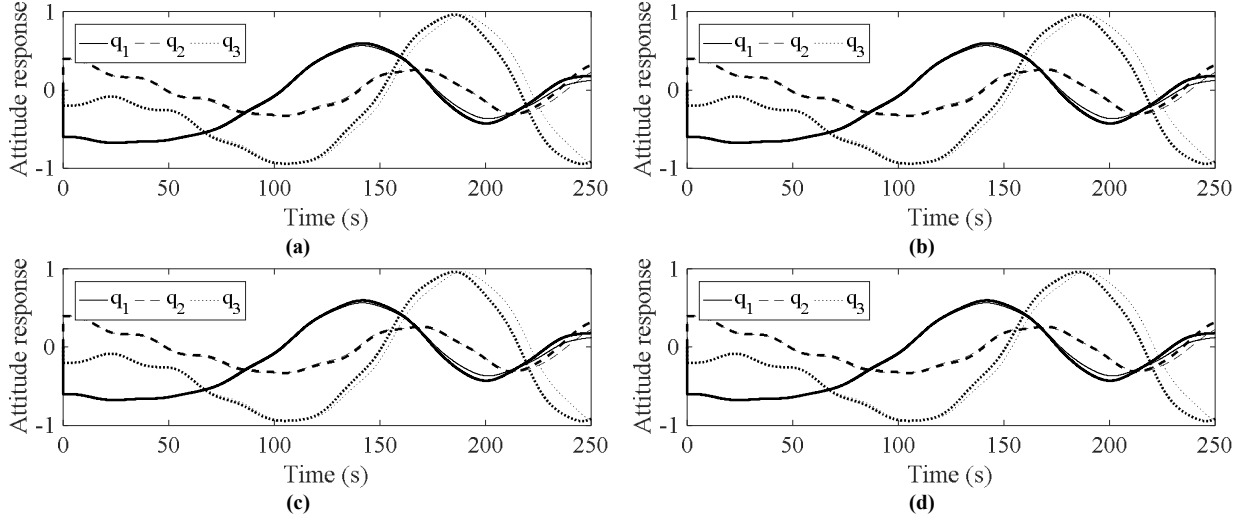
**Figure 4.4 Proposed fault isolation scheme for formation level**

The fault introduced in this scenario is described in Table 4.3 where a transient change in the satellite RW's bus voltage is considered. Faulty RW(s) and satellite(s) are defined in different scenarios as described in Table 4.1.

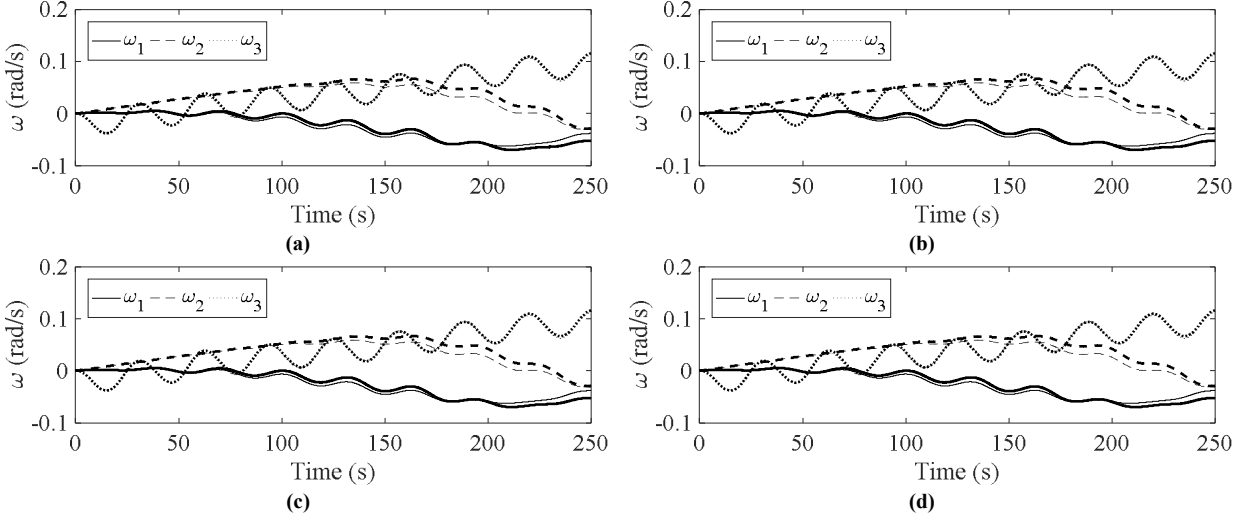
**Table 4.3 Single BEMF constant and Bus Voltage fault**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $k_t$ (s)	$k_t$ (Nm/A)
$t < 50$	8	$t < 50$	0.029
$50 \leq t < 60$	3	$50 \leq t < 60$	0.02
$t \geq 60$	8	$t \geq 60$	0.029

Results for fault isolation at formation level with a centralized FDI architecture are presented in Figure 4.5, Figure 4.6, and Figure 4.7 for fault case 15 with all satellites faulty and all RWs fault. Performance of isolation scheme for other scenarios is presented in Table 4.4. Figure 4.5 illustrates attitude response of all satellites where *thin lines represent nominal behaviour and bold lines represent faulty behaviour*. It can be observed that at the incept of fault, faulty attitude deviates from the nominal states and fault detection scheme announces the fault and starts the isolation process. It should be noted that the deviation of the fault is not observable with naked eyes. That is why human cannot as quickly detect and isolate source of the fault when such changes in non-measurable system parameters occur. Figure 4.6 illustrates angular velocity of each satellite for nominal and faulty estimates and it is clear that faulty estimates deviate at fault inception.



**Figure 4.5 Attitude response of the system under fault scenario 15 with pyramid RW assembly: (a) satellite #1 quaternions (b) satellite #2 quaternions (c) satellite #3 quaternions (d) satellite #4 quaternions**



**Figure 4.6 Angular velocity of the system under fault scenario 15 with pyramid RW assembly: (a) satellite #1 angular velocities (b) satellite #2 angular velocities (c) satellite #3 angular velocities (d) satellite #4 angular velocities**

Furthermore, many other approaches have investigated the feasibility of the MM models techniques to explore applicability of such methods in different systems under different operating conditions. However, in space, physical access to the system (e.g. satellite) is very limited or not at all possible. Therefore, employing analytical techniques to analyze the data from satellite sensors is of paramount importance to be further explored. Although human operators in the ground station constantly monitor the incoming data and performance of the critical system components, there is only so much human beings are capable of in terms of sensory and physical/analytical capabilities. Therefore, the argument can be made that employing analytical techniques is an inevitable reality.

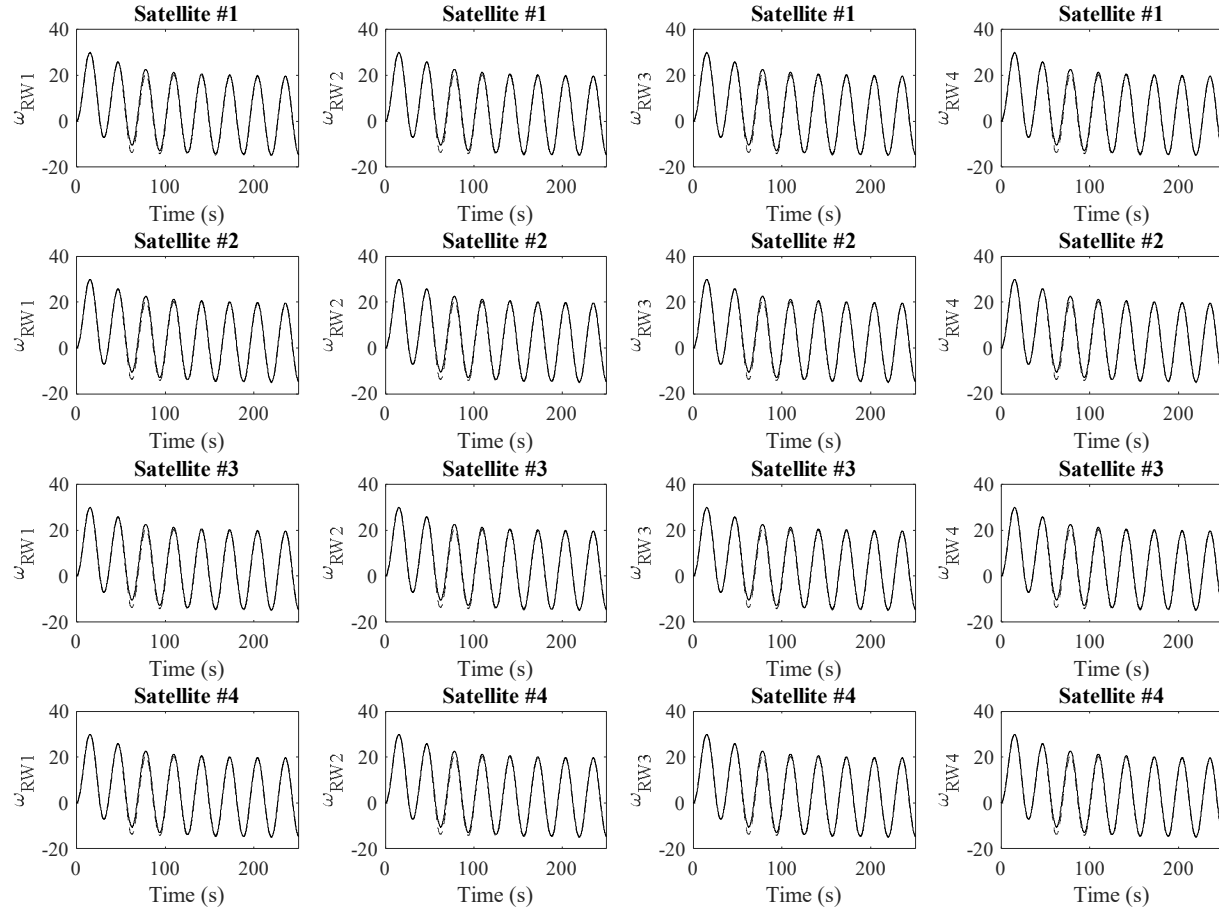


Figure 4.7 Angular velocity of the RWs for all satellites under fault scenario 15

Table 4.4 Confusion matrix for 4 satellites in formation

Identified → Actual ↓	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	99	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	99	0	0	0	0	0	1	0	0	0	0	0	0	0
3	0	0	99	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	99	0	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	99	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	99	0	0	0	0	0	0	1	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	99	0	0	1	0	0	0
10	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
12	0	0	0	0	0	0	0	0	0	1	0	99	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0
14	0	0	0	0	0	0	0	0	1	0	0	0	0	99	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100

Figure 4.7 shows nominal and faulty angular wheel speed for each RW on each satellite. It can be observed from this figure as well as the attitude response and angular velocity figures that faulty estimates deviate from the nominal at the inception of fault and continue to deviate further after fault is resolved in the RW. In the RW responses it can be seen that faulty estimates return back to nominal after fault is resolved, however, their effect on satellite attitude remains even after their nominal performance is recovered, due to the irreversible effects they leave on the satellite motion in the absence of a controller.

Results in Table 4.4 are presented in a form of a confusion matrix. Table 4.4 shows that the isolation scheme is capable of accurately identifying faulty satellites in all possible scenarios with more than 99% success rate. This is important as this is the first level of isolation and any false alarm or misidentification can lead to further wrong decision along the hierarchy. With this success rate, it can be concluded that the proposed scheme for fault isolation at the formation level is capable of handling identification of the faulty satellites with attitude measurements directly provided to it from each satellite in a centralized FDI architecture.

In the following sections, the next level of isolation at subsystem level is examined. Fault isolation for both RW and CMG units are investigated for different configurations of the actuators onboard satellite.

#### 4.2.2 RW Assembly

A control voltage of a sine form with an amplitude of 5V and a frequency of 0.2 Hz is applied to the RW assembly, which applies the same voltage to all RWs (Figure 4.8) enclosed within it. This voltage is fed into all RWs onboard satellite throughout the simulation. Hence, the system is open loop controlled. The output of the RW assembly is the torque generated by each wheel individually; these torques are then mapped to satellite principal axes using mapping matrix based on the RW assembly configuration specified in Eq. (2.34). Satellite dynamics in Eq. (2.10) and satellite kinematics in Eq. (2.13) are used to output angular velocities and quaternions for the process model in Eq. (2.43). The measurement model in Eq. (2.43) is provided as  $\mathbf{g}(\mathbf{X}_k) = \mathbf{X}_k + \mathbf{v}_k$ , which in practice is provided by the onboard sensors. Numerical integration of the states are done in MATLAB using classical Runge-Kutta method (RK4) [172] with a sampling period of  $T_s = 0.01$  s. White-noise is added to the integration outputs of the satellite dynamics model, including RWs, as measurement noise with 0 mean and standard deviation ( $\sigma$ ) of  $1 \times 10^{-5}$ . The total simulation time = 250 s. The overall simulation setup is illustrated in Figure 4.8 where control switching scheme block and attitude control block are not considered in this study. Further simulation parameters are listed in Table 4.5.

**Table 4.5 Simulation Parameters [5], [7]**

Parameter	Value
Input Voltage ( $V_{comm}$ )	$5\sin(0.2t)$ V
Initial Input Voltage ( $V_{comm_0}$ )	0.1 V
UKF Filters Parameters ( $[\alpha, \kappa, \lambda]$ )	[1,2,2]
Process Noise Standard Deviation ( $\sigma_Q$ )	$10^{-2}$
Measurement Noise Standard Deviation ( $\sigma_R$ )	$10^{-5}$
Process Noise Covariance Matrix ( $Q_0$ )	$\sigma_Q^2 I_{n \times n}$
Measurement Noise Covariance Matrices ( $R_0$ )	$\sigma_R^2 I_{m \times m}$
Additive States $\sigma_n$	$10^{-5}$
Window Size ( $N$ )	60
Step time ( $T_s$ )	0.01 sec
Satellite Moment of Inertia ( $J_s$ )	$\begin{bmatrix} 0.015 & 0 & 0 \\ 0 & 0.017 & 0 \\ 0 & 0 & 0.020 \end{bmatrix} [\text{kg} \cdot \text{m}^2]$
RWs Moment of Inertia ( $J_w$ )	$1 \times 10^{-5} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} [\text{kg} \cdot \text{m}^2]$
Satellite Initial Quaternions [ $q_1 q_2 q_3$ ]	[-0.6 0.4 -0.2]
Satellite Initial Angular Velocities [ $\omega_1 \omega_2 \omega_3$ ]	[0 0 0] rad/s

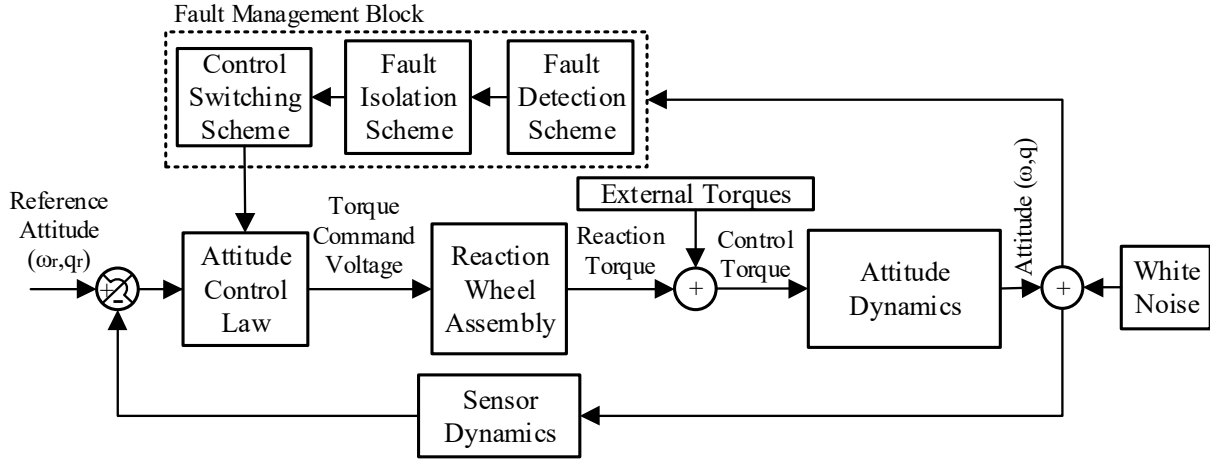
As discussed, the potential sources of failure for RWs are: (1) faults in the bus voltage and ( $V_{bus}$ ) (2) faults in the motor torque BEMF constant ( $k_t$ ). Hence, transient faults are introduced in RWs regarding these parameters. The objective is to identify where a fault has occurred at subsystem level having access only to the system level measurements. The faults injected into the system are BEMF constant and bus voltage faults occurring in phase (Table 4.6). The values for fault parameters used in isolation module are [ $k_t V_{bus}$ ]=[0.029 Nm/A 5 V]. This is to ensure the values of the faulty parameters do not affect the performance of the isolation module and only direction of change is of significance. However, if the values in isolation module are far from the values in the actual system, the isolation module performance can degrade and result in false isolation.

For different fault case scenarios, it is assumed that the fault injection scenario and values are the same for all cases (i.e. time of fault and value of parameter changes in the wheel), and the only difference is where the fault is injected in the assembly pertaining to which wheels are malfunctioning.

Figure 4.9 illustrates the response of the system under nominal and faulty conditions for a fault case where wheel 1 is defective in the pyramid configuration. In this figure, *thin line corresponds to the nominal response and bold line corresponds to the faulty response*. Figure 4.9 (a) and Figure 4.9 (b) show when wheel 1 becomes faulty, satellite angular velocities deviate from the nominals. This phenomenon is more visible in Figure 4.9 (c)



where wheel 1 torque deviates from its nominal response after wheel goes faulty. At the fault isolation module, we do not have access to wheel outputs including torque generated by each wheel and need to isolate the fault using only measurements from satellite attitude and angular velocity. Figure 4.9 (d) and Figure 4.9 (e) illustrate trends of satellite angular velocities and quaternions in 3D, respectively.



**Figure 4.8 Proposed fault isolation simulation setup**

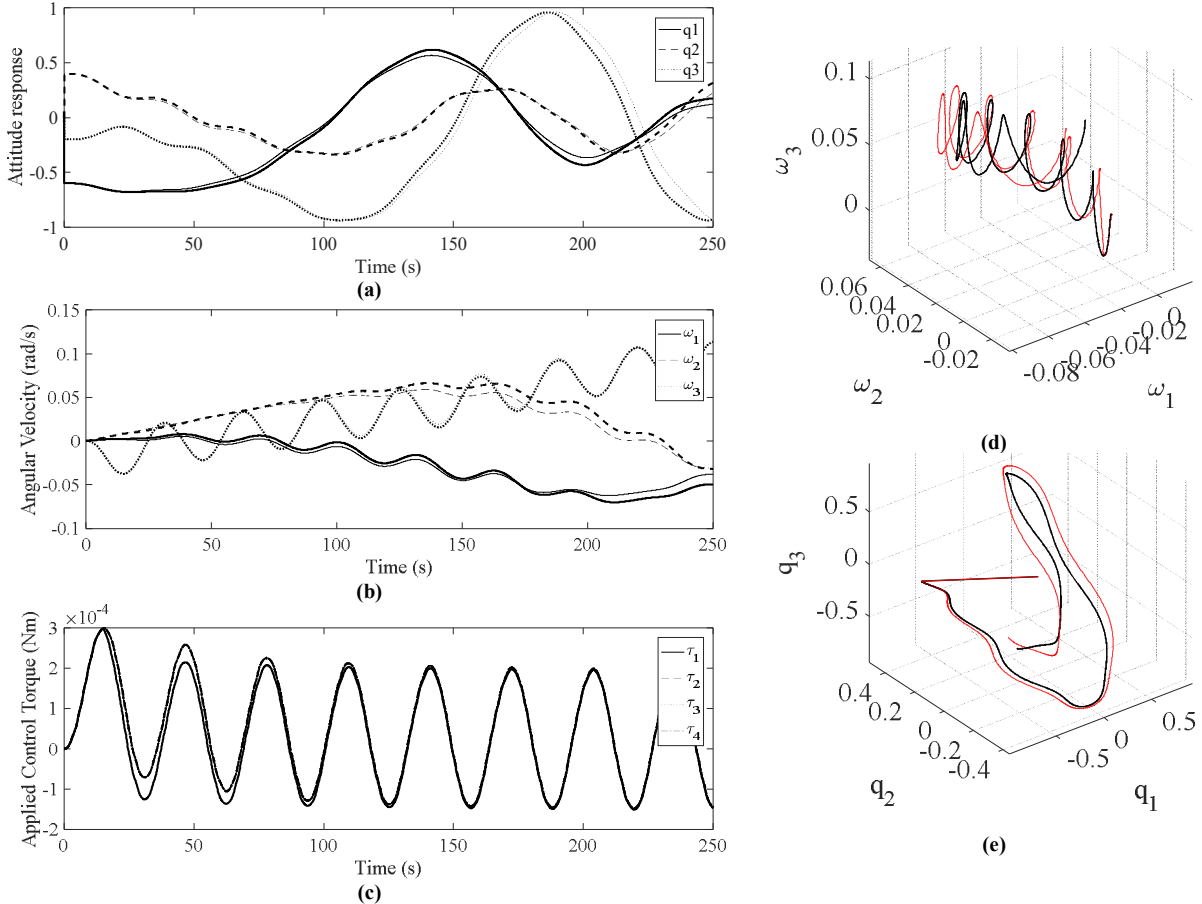
**Table 4.6 Single BEMF constant and Bus Voltage fault**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $k_t$ (s)	$k_t$ (Nm/A)
$t < 15$	8	$t < 15$	0.029
15 $t < 77$	3	15 $t < 77$	0.02
$t \geq 77$	8	$t \geq 77$	0.029

Figure 4.11 shows the performance of the fault isolation module. Figure 4.11 (a)-(c) illustrate prior distribution, likelihood, and posterior distribution of fault cases at the first iteration of isolation. Figure 4.11 (d)-(f) illustrate prior distribution, likelihood, and posterior distribution of fault cases at the final iteration of isolation. As can be seen in Figure 4.11 (a), at the beginning of isolation we assume a uniform distribution for all fault cases. It means that all cases have the same probability of being the current state of the system; however, as we proceed through iterations of isolation, likelihood function multiplied by prior distribution provides posterior distribution. In Figure 4.11 (f), case 1 has the highest probability of being the current state of the system among others. As mentioned earlier, the threshold is set in the probability to be more than 90% in the final posterior probability distribution. Hence, for different cases and under different conditions, final iteration differs. Figure 4.12 illustrates isolation performance for the same scenario (wheel 1 is faulty in the pyramid configuration) with normal initial distribution.

Referring to Figure 4.12 (a) the initial distribution is normal. However, Figure 4.12 (f) shows that after isolation scheme reaches a decision, case 1 is the most probable case.

Another difference that the initial distribution can make is the number of iterations required for the isolation scheme to reach the decision. This can be explained based on the distribution itself. Referring to Figure 4.11 and Figure 4.12, it can be seen that at the beginning case 1 has a higher probability in Figure 4.11 than it has in Figure 4.12. Therefore, if the same likelihood distribution is generated in both cases, it takes more iterations for case 1 under normal initial distribution to pass the threshold compared to case 1 under uniform initial distribution. The fault scenario under which wheel 1 is faulty on a standard four-wheel configuration is illustrated in Figure 4.10. As can be observed, due to change in the configuration of the wheels on the assembly, system response and consequently attitude deviation after fault inception is changed. However, isolation is accomplished promptly for both cases regardless of the assembly configuration. Referring to Figure 4.13, case 1 is the predominant case in the final iteration of isolation and therefore is chosen by the isolation module as the current faulty state of the system (satellite).



**Figure 4.9** Attitude response of the system under fault scenario 1 with pyramid RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes

A confusion matrix is provided for each RW assembly under different fault cases with uniform initial distribution to investigate the performance of the proposed methodology. The tables are filled with the data of 100 simulations for each fault case under the same initial conditions and fault start and end times. The only variation in these simulations is the external white noise added to the system and measurement units. Table 4.7 shows the confusion matrix for the orthogonal configuration. As can be observed from this table, the majority of the cases are identified as the actual case except for some cases where the noise has caused likelihood to shift from actual probabilities and result in wrong isolation. However, isolation results are accurate in more than 99% of the cases. Table 4.8 shows the confusion matrix for the pyramid configuration. In this table, it is noticeable that case 6 and 9 are isolated interchangeably. Looking at the configuration of the assembly, it is evident that case 6 corresponds to fault in wheel 1 and wheel 3 while case 9 corresponds to fault in wheel 2 and wheel 4. Due to symmetry in this configuration, it is expected that fault in any of the combinations for case 6 or 9 could have similar effects on satellite attitude response. Figure 4.14 shows final iteration of isolation for case 6 and 9 in the pyramid configuration. Referring to Figure 4.14 (c) and Figure 4.14 (f) both cases have a final probability of less than 90%; however, according to Table 4.2, another termination logic with a maximum number of 60 iterations was defined, which is imposed in this case.

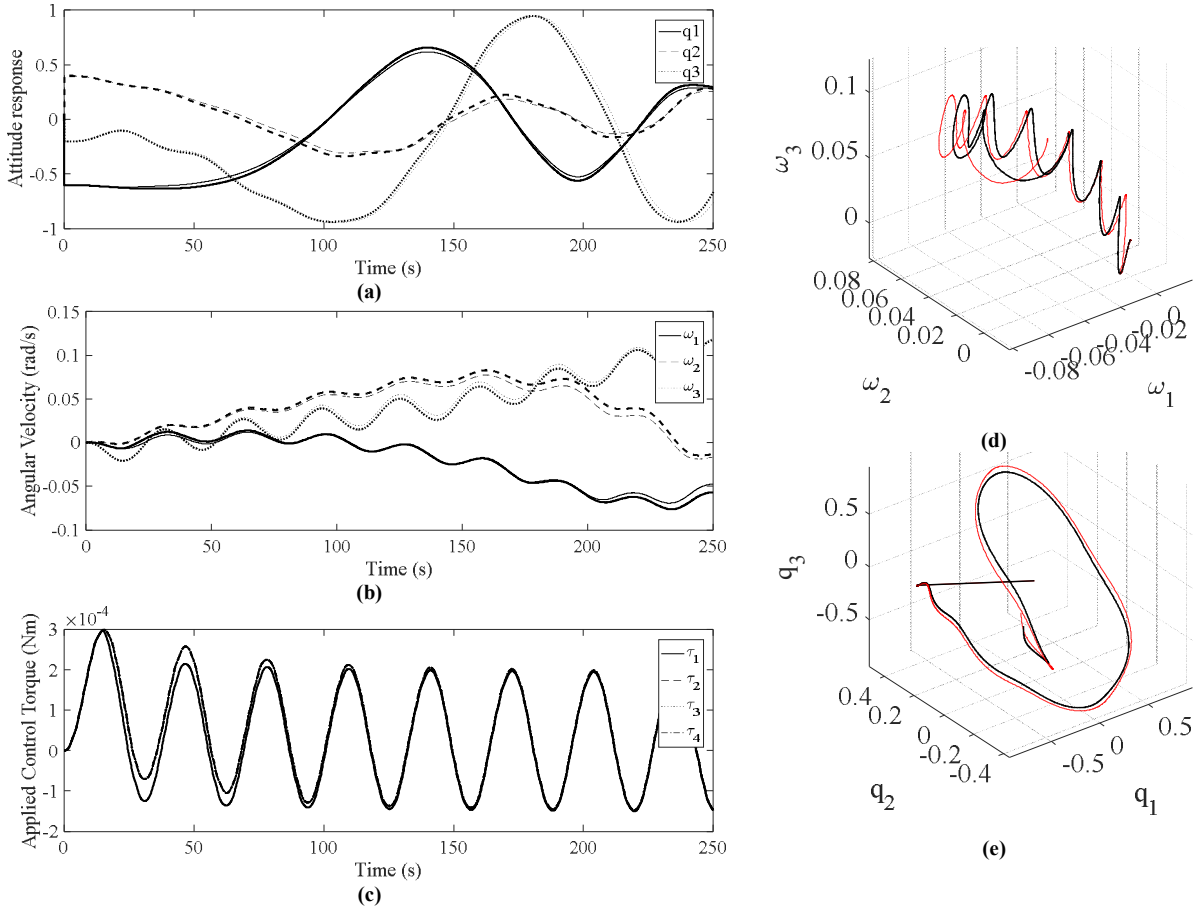
A new scenario was considered where wheels go faulty in sequence to examine the performance of the proposed methodology for multiple faults. Multiple faulty wheels are covered under the scenarios discussed in Table 4.1. However, a more probable case is when wheels go faulty in sequence, one wheel at a time or in-phase with another. For this scenario, we have two possibilities: (1) the first wheel is recovered before the second wheel goes faulty; (2) the first wheel is still faulty while the second wheel goes faulty. Scenario (1) is setup as listed in Table 4.9 and scenario (2) is setup as listed in Table 4.10. Corresponding system responses are shown in Figure 4.15 and Figure 4.16 for scenario 1 and 2, respectively. Referring to Figure 4.3, when a fault is isolated, the current state of the system is updated to the faulty state determined by the isolation module. Hence, isolating a new fault is achieved based on residuals being generated based on the updated system model. When faults occur in sequence, it is important that faults cannot be isolated simultaneously if the previous fault is not isolated. This means, for the sequence of faults to be isolated properly, it is required that fault isolation of the previous fault be completed and system model is updated with the current state of the system.

A change in system response compared to single fault case studies is expected as in these cases two wheels become faulty and consequently satellite motion is altered in two stages. It can be observed from Figure 4.15 (d)-(e) and Figure 4.16 (d)-(e) that satellite motion under these fault scenarios is more chaotic compared to single fault scenarios. Figure 4.15 (c) and Figure 4.16 (c) show wheel 1 and wheel 2 outputs degrading from nominal values. In Figure 4.16 (c) due to concurrent faulty conditions in wheel 1 and 2, degradation is more visible compared to Figure 4.15 (c). Figure 4.16 (b) shows coupled effect of a concurrent fault in wheels 1 and 2 compared to Figure 4.15 (b). The performance of the isolation algorithm is depicted in Figure 4.17 (a)-(c) for scenario 1 and Figure 4.17 (d)-(f) for scenario 2 at their final iteration of isolation. The main observation from these figures is the lower confidence in fault case for scenario 2 in Figure 4.17 (f) compared to that of Figure 4.17 (c) for scenario 1. It should be noted that for scenario 1 it took isolation algorithm 18 and 24 iterations for wheel 1 and wheel 2, respectively to reach confidence level greater than 90% for the isolated case. However, for scenario 2 it took isolation algorithm 17 and 30 iterations for wheel 1 and wheel 2, respectively to reach confidence level greater than 90% for the isolated case.

This is important since 60 is the maximum number of iterations defined by the isolation algorithm. This means the termination logic for isolation was a maximum number of faults rather than required confidence level in the isolated case of scenario 2. This can be explained by the behavior of the system under concurrent faults from both wheels. Since each wheel affects the response of the system, the isolation algorithm cannot differentiate fully between effects of each wheel on the overall response of the system. Hence, lower confidence in the final decision is acquired. Similar results were obtained for same multiple fault scenarios under standard four-wheel RW assembly configuration. For scenarios where more than 2 wheels are faulty, the same procedure can be implemented to isolate faults in sequence. Various simulations were conducted to examine the stability of the proposed method under different influencing parameters. These simulations were mainly concerned with: (1) starting time of the fault (2) initial conditions for the satellite attitude (3) change in parameters of the wheels and their values. Regarding starting time of the fault, it was observed that the earlier a fault occurs in the system, the more accurate the isolation scheme can provide proper results. This can be explained by the fact that at the beginning of the simulation, due to motor dynamics in the RWs, there is more sensitivity to input and hence, larger amplitudes of outputs. As the motor passes its transient period, amplitudes of change

become stable and smaller in size, which leaves MSE generation to work with smaller values and less sensitivity.

Regarding initial conditions for the satellite, it was observed that increasing satellite's initial values for angular velocities causes more chaotic satellite motions but does not affect the performance of the isolation module. Finally, the change of values in RW parameters influences the isolation results. The change in values cannot be smaller than 10% of their nominal values; otherwise, isolation module may provide wrong results. It should be noted that changes in these parameters are in reality larger than 10% because if the  $V_{bus}$  is surged or  $k_t$  is changed due to change in temperature, the changes are only visible at the system level when there is substantial change of momentum and consequently a major change in satellite attitude.



**Figure 4.10** Attitude response of the system under fault scenario 1 with standard four-wheel RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes

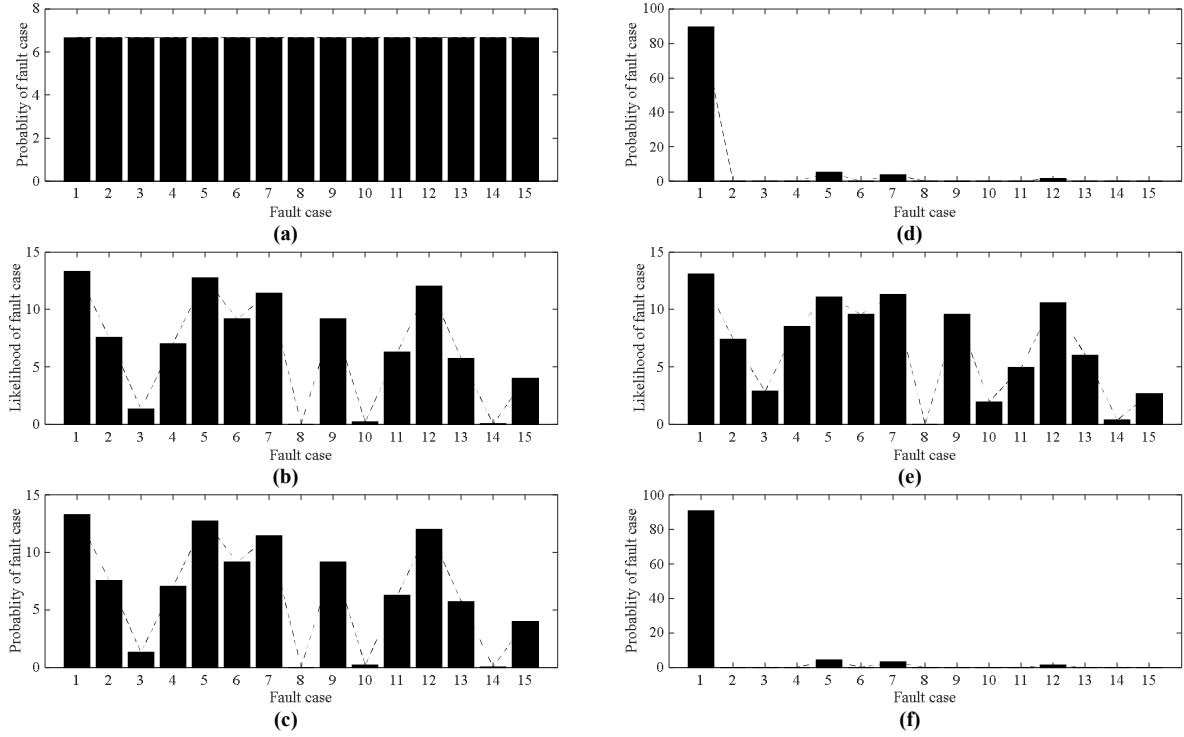


Figure 4.11 Fault isolation performance, scenario 1, pyramid assembly, uniform initial distribution: first iteration (a) prior probability (b) likelihood (c) posterior probability; final iteration (d) prior probability (e) likelihood (f) posterior probability

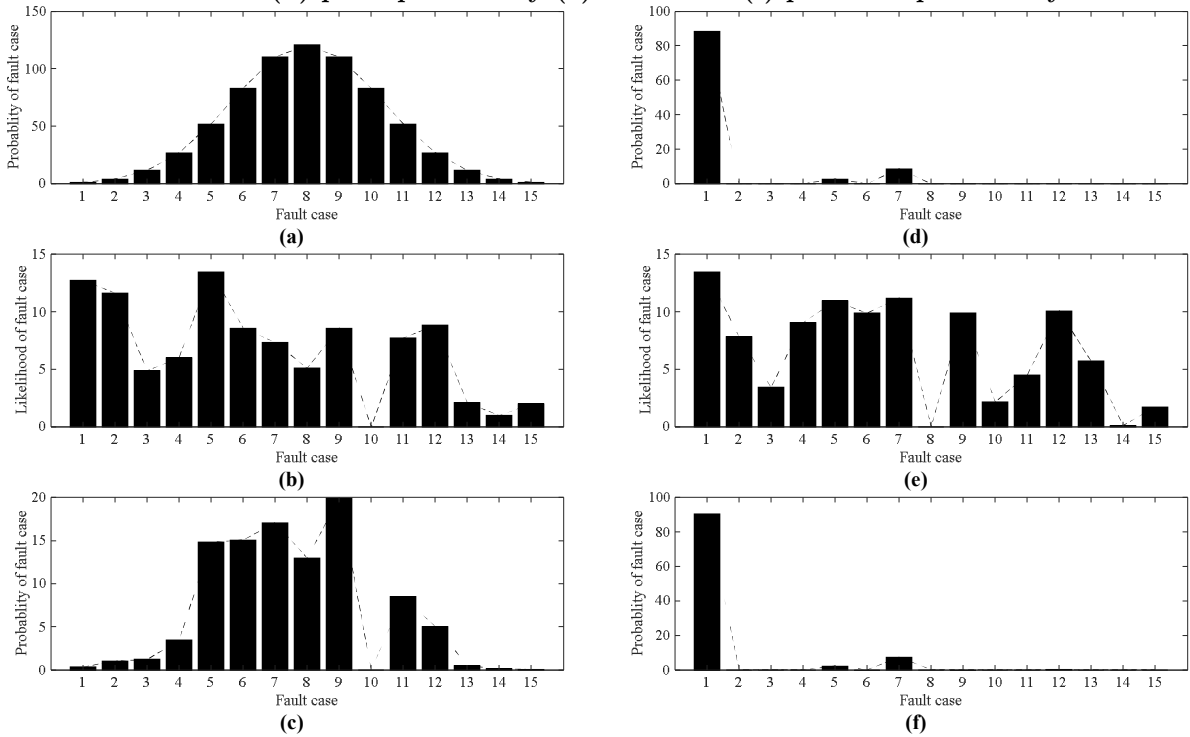
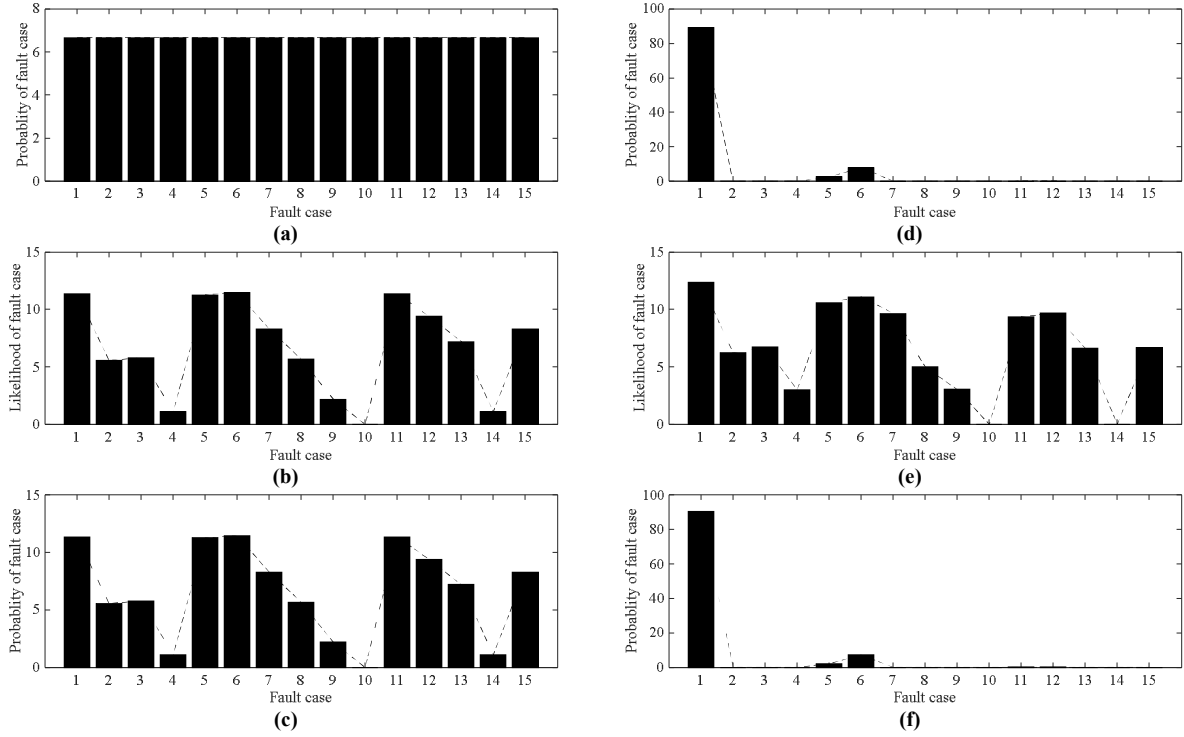
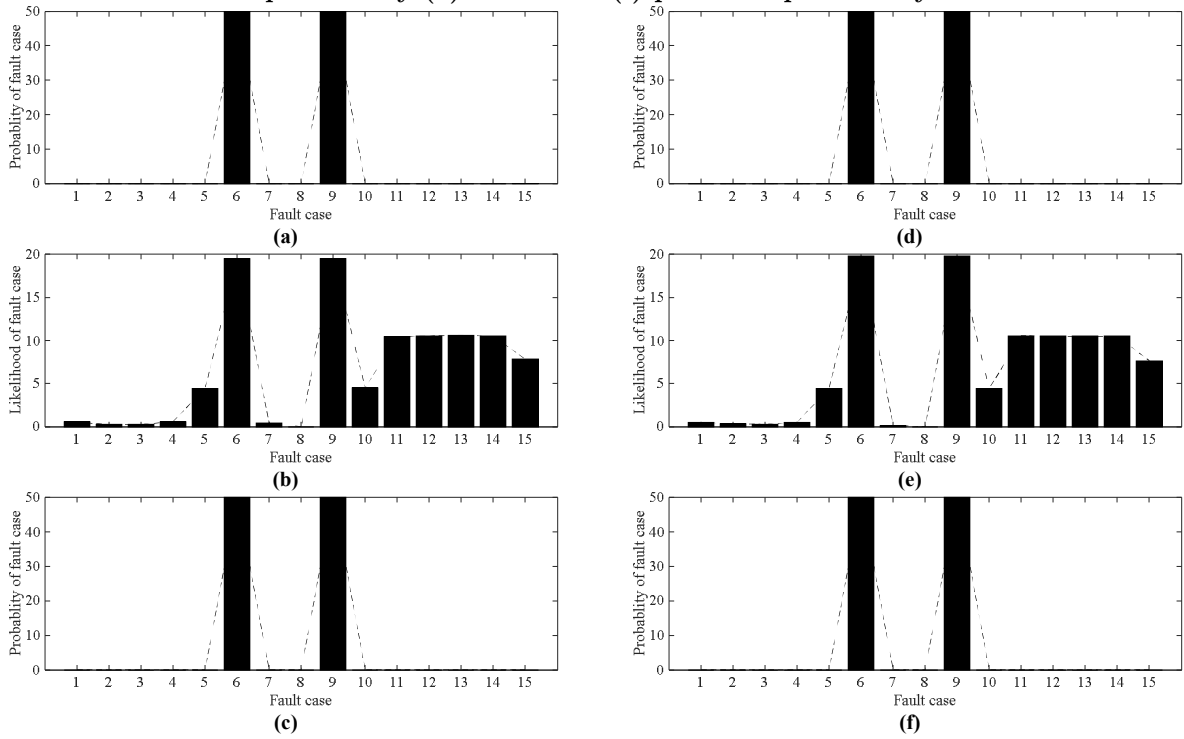


Figure 4.12 Fault isolation performance, scenario 1, pyramid assembly, normal initial distribution: first iteration (a) prior probability (b) likelihood (c) posterior probability; final iteration (d) prior probability (e) likelihood (f) posterior probability



**Figure 4.13** Fault isolation performance, scenario 1, standard four-wheel assembly: first iteration (a) prior probability (b) likelihood (c) posterior probability; final iteration (d) prior probability (e) likelihood (f) posterior probability



**Figure 4.14** Fault isolation performance, scenario 6 and 9, pyramid assembly: final iteration for scenario 6(a) prior probability (b) likelihood (c) posterior probability; final iteration for scenario 9 (d) prior probability (e) likelihood (f) posterior probability

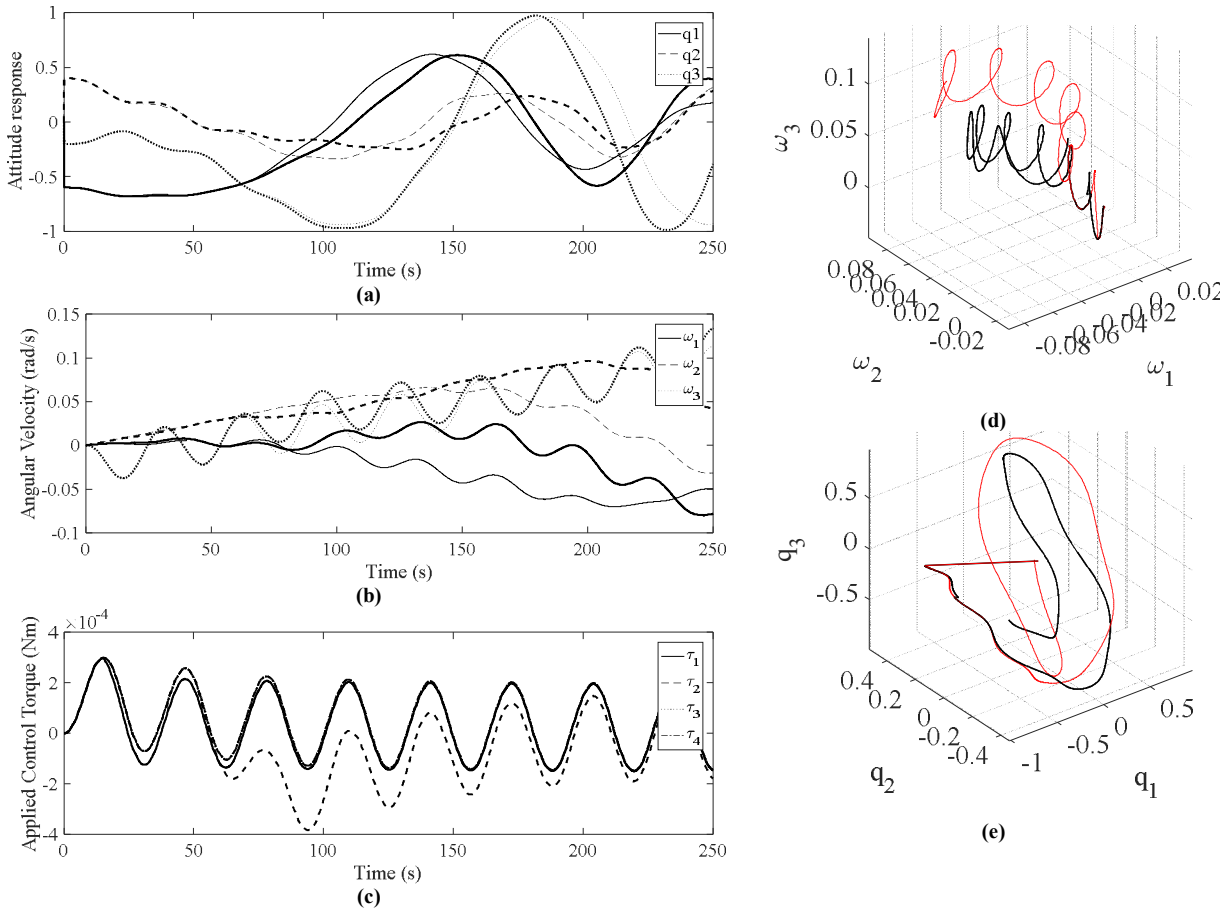
**Table 4.7 Confusion matrix for three orthogonal assembly**

Identified → Actual ↓	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	99	0	0	0	0	1	0	0	0	0	0	0	0	0	0
2	0	99	0	0	0	0	0	1	0	0	0	0	0	0	0
3	0	0	99	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	99	0	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	99	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	99	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	99	0	0	0	0	1	0
10	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	2	98	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0
14	0	0	0	0	0	0	0	0	1	1	0	0	0	98	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100

**Table 4.8 Confusion matrix for pyramid assembly**

Identified → Actual ↓	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	99	0	0	0	0	0	1	0	0	0	0	0	0	0	0
2	0	98	0	0	1	0	0	1	0	0	0	0	0	0	0
3	0	0	98	0	0	0	0	0	0	2	0	0	0	0	1
4	0	0	0	96	0	0	2	0	0	2	0	0	0	0	0
5	0	0	0	0	95	0	0	0	0	0	3	2	0	0	0
6	0	0	0	0	0	55	0	0	45	0	0	0	0	0	0
7	0	0	0	0	0	0	99	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	99	0	0	0	0	0	1	0
9	0	0	0	0	0	53	0	0	47	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	99	0	0	0	1	0
11	0	0	0	0	0	0	0	0	0	0	99	0	0	0	1
12	0	0	0	0	0	0	0	0	0	0	0	99	0	0	1
13	0	0	0	0	0	0	0	0	0	0	0	0	97	0	3
14	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0
15	0	0	0	0	0	0	0	0	0	0	1	0	0	0	99





**Figure 4.15 Attitude response of the system to multiple faults scenario 1 with pyramid RW assembly: (a) satellite quaternions (b) satellite angular velocities (c) reaction wheels torques (d) satellite angular velocities in 3D axes (e) satellite quaternions in 3D axes**

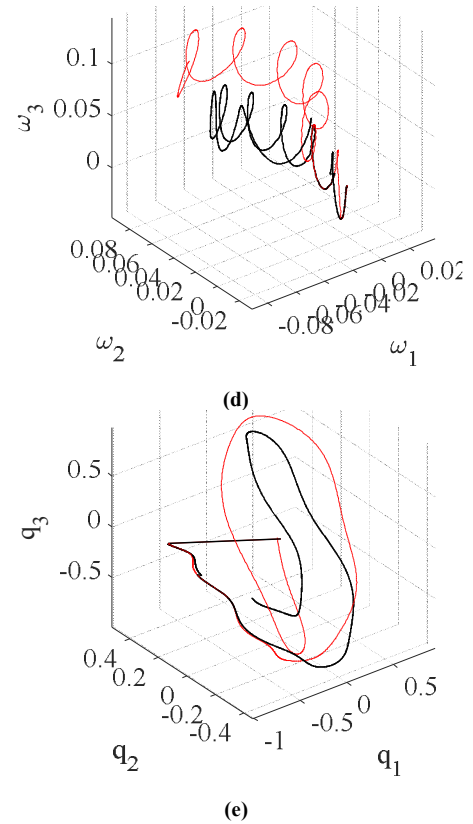
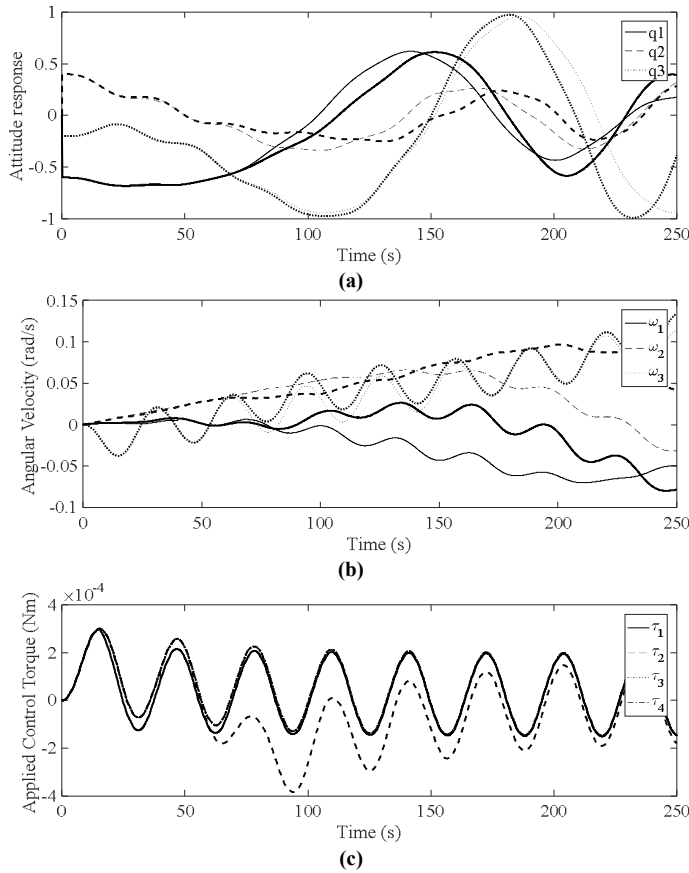
For the second level of isolation, parameter changes at the RW level can be examined [169]. Parameter estimation methods can be used to estimate RW parameters and provide further details on which parameters have changed and their severity as detailed in Chapter 3. This, however, can only be achieved if further information is available to the estimator. For example, RW current and angular velocities either directly or through other observers. For the control part of the satellite system, outputs of the isolation module can be fed into a control switching unit (Figure 4.8) and accommodate for faulty actuators to maintain mission objectives and satellite attitude in orbit. This part can be further investigated in future work to complete the package of fault tolerant satellites in orbit and under uncertainties.

**Table 4.9 Multiple BEMF constant and Bus Voltage fault profile – Scenario 1**

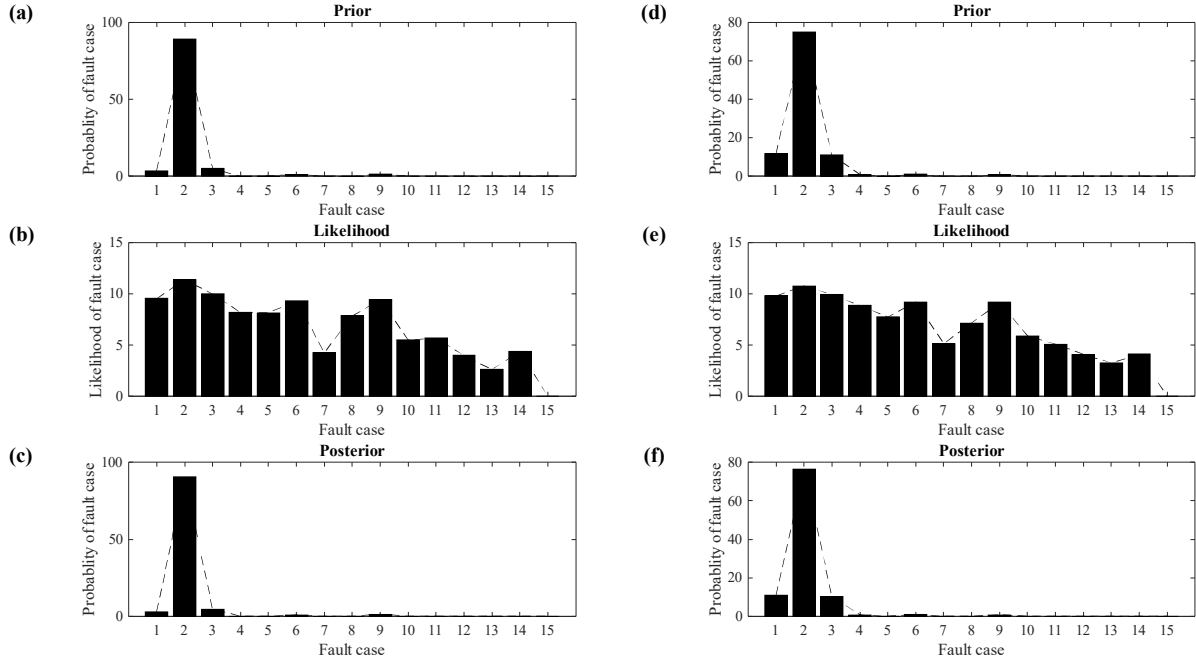
Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $k_t$ (s)	$k_t$ (Nm/A)	Faulty wheel
$t < 15$	8	$t < 15$	0.029	None
$15 \leq t < 25$	3	$15 \leq t < 25$	0.02	Wheel 1
$25 \leq t < 50$	8	$25 \leq t < 50$	0.029	None
$50 \leq t < 80$	3	$50 \leq t < 80$	0.02	Wheel 2
$t \geq 80$	8	$t \geq 80$	0.029	None

**Table 4.10 Multiple BEMF constant and Bus Voltage fault profile – Scenario 2**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $k_t$ (s)	$k_t$ (Nm/A)	Faulty wheel
$t < 15$	8	$t < 15$	0.029	None
$15 \leq t < 25$	3	$15 \leq t < 25$	0.02	Wheel 1
$20 \leq t < 50$	3	$20 \leq t < 50$	0.02	Wheel 2
$25 \leq t < 50$	8	$25 \leq t < 50$	0.029	Wheel 1
$t \geq 50$	8	$t \geq 50$	0.029	None



**Figure 4.16 Attitude response of the system to multiple faults scenario 2 with pyramid RW assembly: (a),(e) satellite quaternions (b),(d) satellite angular velocities (c) reaction wheels torques**



**Figure 4.17** Fault isolation performance, multiple fault scenario, pyramid assembly: final iteration of scenario 1 (a) prior probability distribution (b) likelihood (c) posterior probability distribution; final iteration of scenario 2 (d) prior probability distribution (e) likelihood (f) posterior probability distribution

### 4.2.3 CMG Assembly

Since it is possible to directly estimate the fault parameters for the system described in Eq. (3.30), it can be concluded that it is straightforward to isolate which CMG is faulty in the assembly along with the severity of the fault(s). In order to investigate the performance of the fault isolation for the CMG assembly, an in-phase fault scenario was considered with different fault severities for each CMG. The process model in Eq. (2.43) is provided as Eqs. (2.12), (2.13) and the measurement model in Eq. (2.43) is provided as  $g(X_k) = X_k + v_k$ . Other simulation parameters are the same as Table 3.46 and the fault scenario is described in Table 4.11 where at each fault instant, at least two of the CMGs are performing with lower than perfect performance ( $fp < 1$ ) and at some instances, all CMGs go back to nominal performance ( $fp = 1$ ). This scenario is considered to ensure the algorithm performs well when multiple CMGs go faulty with different fault severities. Results for this scenario are shown in Figure 4.18-Figure 4.19 and listed in Table 4.12. As can be seen, all parameters are properly estimated using AUKF and CAUKF, while UKF has failed to precisely estimate all parameters. In addition, Table 4.12 confirms that the CAUKF has outperformed both UKF and AUKF by having the least RMS.

**Table 4.11 CMG fault parameters in-phase abrupt fault profile**

Time (s)	$f_{p_1}$	$f_{p_2}$	$f_{p_3}$	$f_{p_4}$
$t < 30$	1	1	1	1
$20 \leq t < 28$	0.5	0.3	1	1
$28 \leq t < 35$	1	0.1	0.5	1
$35 \leq t < 50$	1	1	1	1
$50 \leq t < 60$	1	0.1	0.1	0.3
$60 \leq t < 80$	0.8	0.8	0.8	0.8
$80 \leq t < 90$	1	1	1	1
$t \geq 90$	1	1	1	0.1

**Table 4.12 CMG in-phase abrupt fault scenario performance measures**

		Noise factor $\phi = 10$				
Algorithm	time (sec)	MSE ( $f_{p_1}$ )	MSE ( $f_{p_2}$ )	MSE ( $f_{p_3}$ )	MSE ( $f_{p_4}$ )	RMS
UKF	2.57	$4.95 \times 10^{-3}$	$9.74 \times 10^{-3}$	$3.55 \times 10^{-3}$	$3.76 \times 10^{-2}$	$1.97 \times 10^{-2}$
AUKF	8.15	$2.27 \times 10^{-3}$	$5.96 \times 10^{-3}$	$2.52 \times 10^{-3}$	$4.99 \times 10^{-3}$	$4.24 \times 10^{-3}$
CAUKF	7.85	$1.44 \times 10^{-3}$	$5.81 \times 10^{-3}$	$2.52 \times 10^{-3}$	$3.49 \times 10^{-3}$	$3.69 \times 10^{-3}$

To investigate all possible cases of in-phase fault, a confusion matrix is presented in Table 4.13. To generate this table, fault cases from Table 4.1 are considered. The fault inception time and severity of each fault (i.e. value for  $f_{p_i}$ ) is generated randomly with MATLAB rand function, which produces a normally distributed random number. Next, when fault detection scheme captures a fault, it readjusts the covariance matrix and continues to estimate the fault parameters. After 1 second from the inception of the fault, parameters are considered to have settled at their new value and the isolation starts.

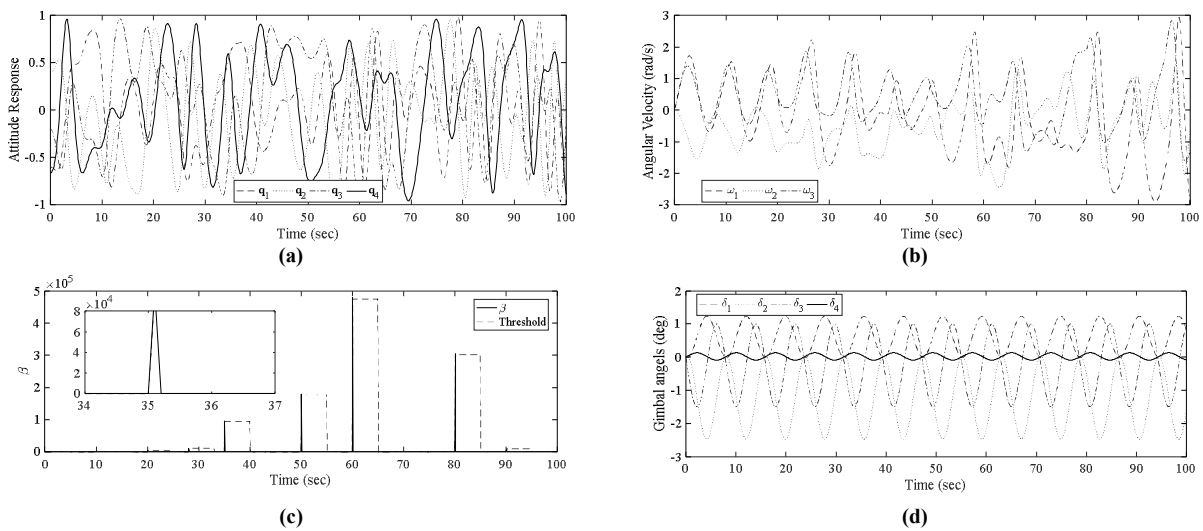
Isolation works by creating a vector that calculates each CMG's  $f_p$  by subtracting its value from 1 (nominal performance); and if the difference is more than 0.05, the CMG unit is considered to be faulty. Therefore, a vector of size  $1 \times 4$  is generated where each element can have a value of 1 (faulty) or 0 (non-faulty). This vector is then compared with the cases Table 4.1 to determine which case is the current state of the system. For all possible cases in Table 4.1, 100 simulations were conducted, each with a randomly generated fault inception time and fault severity, and results are presented in Table 4.13.

As can be seen, results show successful isolation in all cases with the average success rate of about 89.5%. The reason for less accuracy of this approach compared to the RW isolation scheme, is the fact that less information is available to isolation scheme and parallel models are not running to ensure the probability of the current state. Instead, direct estimation of the fault parameters is employed. Another consideration for these results is the

fact that fault inception time and fault severity is generated randomly for each fault. Therefore, if fault severity is very small, the fault isolation scheme will consider that as a non-faulty case; hence, the result of the isolation will not be accurate. This can be most visibly observed where fault case 0 is determined by the isolation scheme, while it was not introduced to the system at all. Other wrong determinations follow the same justification.

**Table 4.13 Confusion matrix for CMG assembly**

Identified → Actual ↓	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	98	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	97	0	0	0	0	0	0	1	0	0	0	0	0
4	0	0	0	98	0	0	0	0	0	0	0	0	0	0	0
5	4	6	0	0	89	0	0	0	0	0	0	1	0	0	0
6	8	0	7	0	0	82	0	0	0	0	0	0	1	0	0
7	6	0	0	3	0	0	91	0	0	0	0	0	0	0	0
8	0	3	4	0	0	0	0	93	0	0	0	0	0	0	0
9	0	6	0	5	0	0	0	0	88	0	0	0	0	0	0
10	0	0	5	5	0	0	0	0	0	87	0	0	0	0	0
11	0	0	0	0	2	9	0	5	0	0	83	0	0	0	0
12	0	1	0	0	7	0	2	0	7	0	0	83	0	0	0
13	0	0	1	0	0	7	2	0	0	4	0	0	86	0	0
14	0	0	1	1	0	0	0	3	4	7	0	0	0	84	0
15	0	0	0	0	2	0	0	0	0	1	4	1	5	1	86



**Figure 4.18 Simulation results for CMG fault isolation in-phase abrupt CMG faults with  $\phi=10$  (a) Satellite attitude (b) Satellite angular velocities (c) Scaler  $\beta$  trend (d) CMGs gimbal angles**

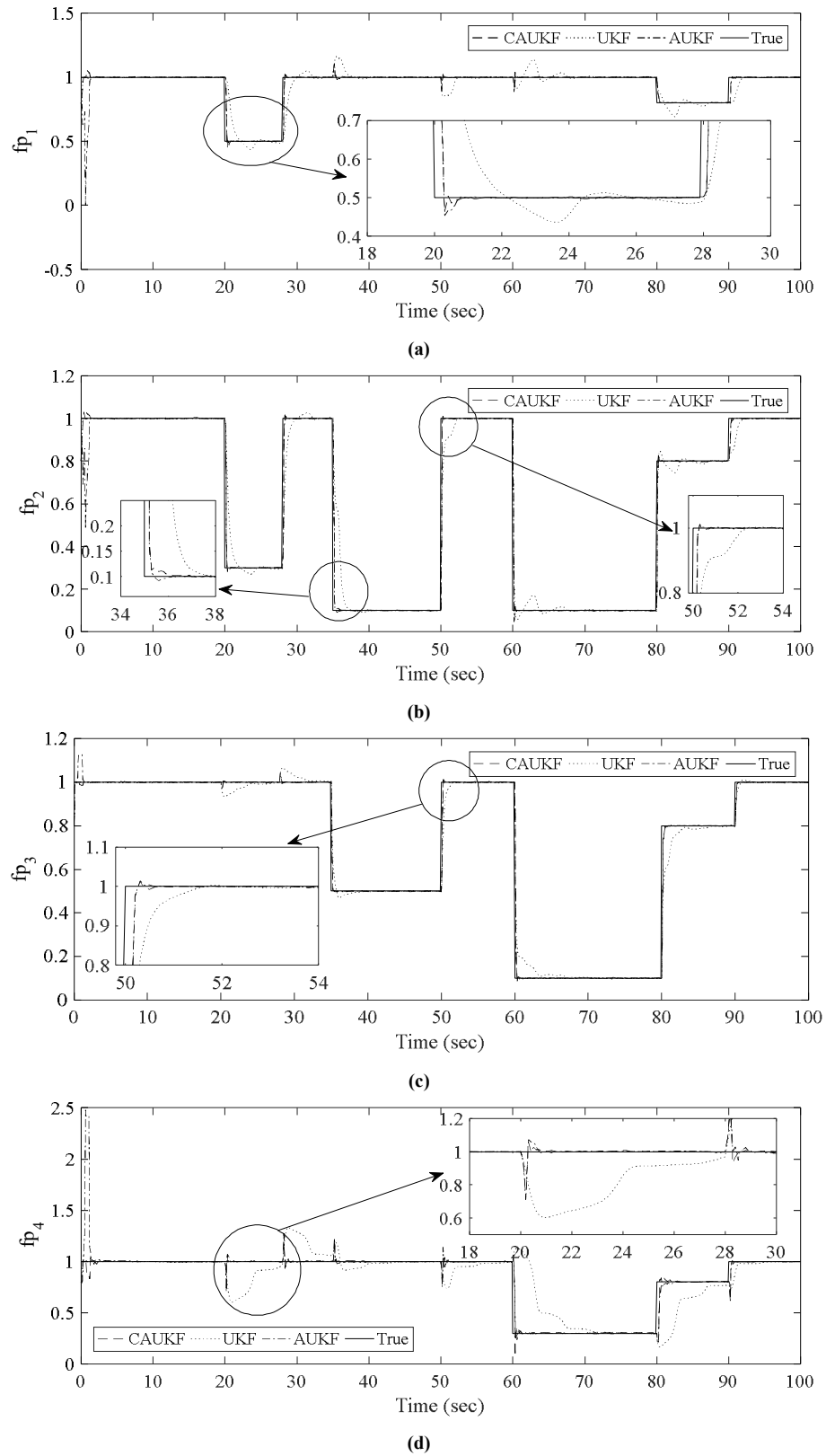


Figure 4.19 Comparison of fault parameters estimates for in-phase abrupt fault with  $\phi=10$   
 (a) CMG1 (b) CMG2 (c) CMG3 (d) CMG4

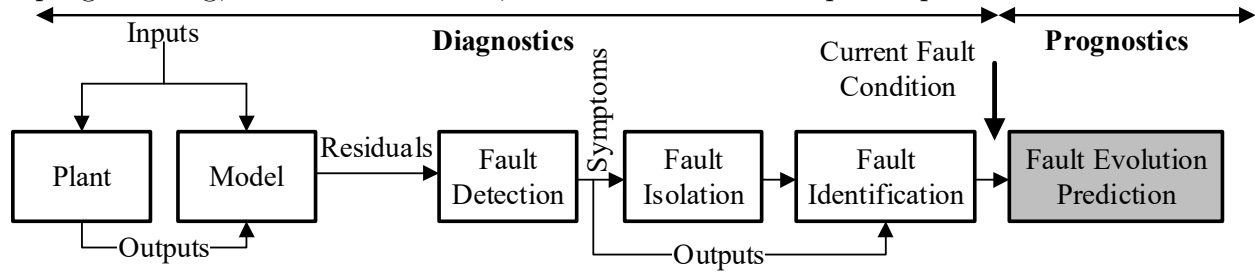
### 4.3 Conclusions

In this chapter, a new 2-step approach was proposed for improving the fault isolation scheme of fault detection, isolation, and identification of satellites in orbit and in formation using a hierarchical approach. The goal was to find the root cause of the fault while only having access to limited measurements at each level of isolation. At the first level of isolation in the formation level, performance of the proposed scheme was evaluated for possible scenarios of fault in the satellites and results showed more than 99% success rate in isolation attempts. Next, for the second of isolation, two different actuators were considered, namely RW (in two configurations) and CMG. For RW, the isolation was achieved using model-based probability distributions and online likelihood distribution generation with parallel model comparison and MSE calculations. The performance of the proposed methodology was investigated for different fault scenarios under two RW assembly configurations, namely pyramid and standard four-wheel. Single and multiple transient abrupt faults were investigated. Results were presented in a confusion matrix and proved that the proposed methodology could isolate faults in more than 99% of the cases. For CMG, the parameter estimation method was directly used to estimate fault parameters and determine the current state of the system using a Boolean logic. Faults were investigated with different inception times and different severity for each CMG, and a confusion matrix was generated. Results showed close to 90% success rate in isolation results. Future work on extending the proposed methodology was also discussed considering carrying out a complete fault-tolerant scheme for satellite attitude in orbit.

## CHAPTER 5

# Fault Prognosis

In order to effectively control the satellite, reaction wheels should be able to provide torque or momentum with adequate volume and precision. Reaction wheels are required to work stably and reliably for over 10 years [103] considering their performance is an important factor affecting lifetime and mission of a satellite [8]. Therefore, creating a model of the degradation for the reaction wheel in order to assess its lifespan is of paramount importance [173]. As can be seen in Figure 5.1, fault evolution prediction or RUL estimation is the last component of the CBM/PHM framework. The outcomes of this module can help operators or intelligence systems to perform remedial actions that include, satellite control reprogramming, mission alternation, etc. in case of catastrophic repercussions.



**Figure 5.1 Placement of fault prognosis module in a model-based CBM/PHM flow**

In general, prognostic methods can be categorized in three major subsidies: data-driven, model-based, and hybrid methods. The data-driven method does not require a physical model of the system and relies heavily on the measured and historical data collected over time. When data is available, it is possible to construct models that mimic the pattern observed in the collected data and further extrapolate that trend to predict future behavior [174]–[176]. On the other hand, a model-based approach requires a physical model of the damage or degradation behavior and combines the model with collected or measured data to identify model parameters. In order to improve the performance and reliability of the algorithm, hybrid methods use the two above-mentioned approaches. In this study, a model-based prognosis approach is considered since access to historical data on the system under study was not available. In this approach, parameters of the model which affect system's behavior are unknown and need to be identified before the prognostics process begins or as part of it. Various methods can be used to estimate these parameters including KFs that give exact analytical PDF for linear systems with Gaussian noise [177], PFs where the posterior distribution of the model parameters is expressed through a number of particles/agents and their respective weights [178], and Bayesian methods (BM) that estimate



the parameters using measured data and incorporates them into a single posterior distribution [179]–[181]. In this study, PF is employed since it is the most widely used approach in the field of prognostics and can be used for nonlinear systems with non-Gaussian noise.

## 5.1 Model-based Prognostics

The process for a model-based prognostic is illustrated in Figure 5.2 where the necessary stages can be named as (1) model identification, (2) state reconstruction, and (3) prognostics. In the model identification stage, unknown parameters of the model are estimated based on the measured data available to the system. The parameters estimated in the identification stage are then used in the state reconstruction stage along with a Markovian form of the model, the data, and input command to estimate the current state of the system. It should be noted that since the damage cannot be measured directly in most cases, a damage quantification process is required from sensor measurement data. This process is usually referred to as structural health monitoring (SHM) where health index (HI) parameters are identified and used for the next stage. In this study, a non-measurable parameter ( $k_t$ ) is used as a HI due to extensive discussions provided in [173] and the explanation provided for this parameter in [52], where it is mentioned to be the factor affecting output torque of the RW proportionally and therefore, faults can affect this parameter one way or the other. Finally, in the prognostics stage, the RUL of the system is estimated based on the reconstructed state and available failure thresholds for a particular parameter or state.

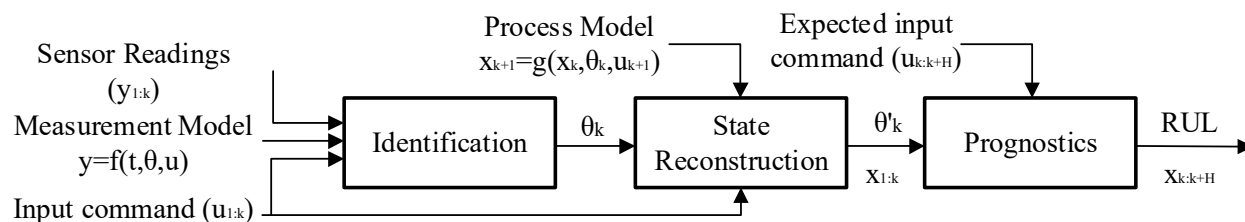
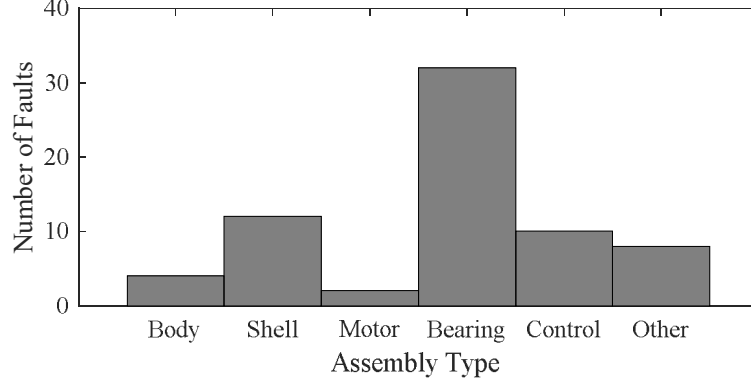


Figure 5.2 Different stages of model-based prognostics

## 5.2 Model Definition

The reaction wheel is a key component of the satellite orbit and attitude control system, which consists of the motor assembly, control circuit, bearing assembly, shell assembly, etc. In a study by Wu mentioned in [103], he calculated the test records for a set of reaction wheels as shown in Figure 5.3. It can be seen that the most number of faults occur in the bearing assembly, which also in the engineering design, life of the bearing assembly is the shortest of the components. Therefore, it can be concluded that health condition of the reaction wheel can be represented by the performance data from the bearing assembly in addition to the discussions provided above for choosing  $k_t$  as a HI. The failure causes of the bearing assembly include insufficient or excessive lubrication, poor bearing

preloading, cage guide problem, etc. where most of these difficulties can be detected from changes in the bearing assembly friction torque. Since the friction torque cannot be directly measured from the telemetry data, an indirect estimation or observation is required for monitoring purposes.



**Figure 5.3 Fault location statistics for a group of reaction wheels**

According to engineering practices, indicators such as wheel speed, wheel current, and bearing temperature can display variations in the bearing friction torque. Other sources of fault can be identified as unexpected changes in the bus voltage  $V_{bus}$  (i.e. bus voltage drop); and unexpected changes in the motor current, which can be represented (or modelled) by changes in the motor torque gain  $k_t$ . Therefore, in this study, faults in the motor current were modelled and injected intermittently as time-varying variations in the RW's motor torque gain [16]. Based on previous studies [97], [99], [176], [182] working on the electrical motors, vibration, friction, and degradation behaviour of such systems and their respective parameters, the degradation model for  $k_t$  (note that  $t$  here is part of the torque gain symbol and not time index) was considered to be of the exponential growth/decline model as [173]

$$k_t = ae^{-bt} \quad (5.1)$$

where  $a$  and  $b$  are model parameters to be estimated and  $t$  is the time of the evolution. The particle filter process was described earlier in Section 2.5. As discussed particle filter works on basis of Bayesian inference with Bayes' theorem explained in Eq. (4.8). In particle filter, the Bayesian update is processed in a sequential manner with particles carrying the probability distribution of the unknown parameters. When a new measurement is available, the *posterior* at the previous step is used as the *prior* at the current step, and the parameters are updated by multiplication of the *prior* with *likelihood* as explained in Eq. (4.9). Hence, the name sequential Monte Carlo method. The general process of the particle filter is based on the state transition function  $f(\cdot)$ , which is also known as the damage model in the prognosis field, and the measurement function  $g(\cdot)$  as described in Eq. (2.43).

The degradation model in Eq. (5.1) can be rewritten in discrete form as

$$k_{t_k} = k_{t_{k-1}} e^{-b_k \Delta t} \quad (5.2)$$

where  $t_k = t_{k-1} + \Delta t$ . In this case the process noise  $w_k$  is neglected due to the fact that it can be handled though uncertainty considered for the model parameters. For the measurement function, since  $k_t$  is not directly measurable, parameter estimation techniques introduced in Chapter 2 for the fault detection are used. Using RW dynamics from Eq. (2.31) with measurements as wheel speed ( $\omega$ ) and motor current ( $I_m$ ), system parameters  $k_t$  and  $V_{bus}$  are estimated using the proposed CAUKF in Chapter 2.

When  $k_t$  values are estimated, the fault trend is among the four possible scenarios for the faulty situation as described in Chapter 2, including abrupt, transient, intermittent, and incipient. Based on the discussion provided for each type of fault progression, it can be concluded that fault prognosis is mainly concerned with the gradual degradation of the system, which can be observed most visibly in incipient fault progression. Therefore, in this chapter, we only deal with the incipient fault progression.

### 5.3 Particle Filter Process

As explained earlier, the process of PF is based on Bayes' theorem. At the first step of the process ( $k = 1$ ),  $n$  samples of the parameter are drawn from the initial (*prior*) distribution. Next, the following steps are followed

#### 5.3.1 Step 1 – Initialization

The *posterior* distribution of the model parameters at the previous step ( $k - 1$ th) is used as the *prior* distribution for the current step ( $k$ th) in the form of samples. In addition, damage state at the current time is transmitted from the samples of the damage model at the previous step, based on the model parameters. The samples in step correspond to  $p(\theta)$  in Eq. (4.8).

#### 5.3.2 Step 2 – Update

Assuming a distribution model for the noise, the likelihood of the measurements is obtained. If the measurement noise  $v_k$  is assumed to be normally distributed, from the normal distribution function formulation,

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (5.3)$$

where  $\mu$  is the mean or expected value of the distribution for  $x$ ,  $\sigma$  is its standard deviation, and  $\sigma^2$  is its variance. The likelihood of the measurements can be obtained from

$$p(z_k|x_k^i, b_k^i, \sigma_k^i) = \frac{1}{\sqrt{2\pi(\sigma_k^i)^2}} \exp \left[ -\frac{1}{2} \left( \frac{z_k - x_k^i(b_k^i)}{\sigma_k^i} \right)^2 \right], i = 1, 2, \dots, n \quad (5.4)$$

The weight for the  $i$ th particle is assigned based on the likelihood calculated from Eq. (5.4). If the measurement noise  $v_k$  is assumed to be lognormally distributed, from the lognormal distribution function formulation,

$$f(x|\mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}} \quad (5.5)$$

where  $\mu$  is the mean or expected value of the distribution for  $x$ ,  $\sigma$  is its standard deviation, and  $\sigma^2$  is its variance. The likelihood of the measurements can be obtained from

$$p(z_k|x_k^i, b_k^i, \sigma_k^i) = \frac{1}{x_k^i \sigma_k^i \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{\ln z_k - x_k^i(b_k^i)}{\sigma_k^i} \right)^2 \right], i = 1, 2, \dots, n \quad (5.6)$$

The weight for the  $i$ th particle is assigned based on the likelihood calculated from Eq. (5.6). These weights are used for the next step of resampling.

For the lognormal distribution, mean, median, and mode are calculated differently from that of the normal distribution. The formulation for these terms can be written as follows

$$mean = \exp(\mu + \sigma^2/2) \quad (5.7)$$

$$median = \exp(\mu) \quad (5.8)$$

$$mode = \exp(\mu - \sigma^2) \quad (5.9)$$

It is important to note that  $\mu$  and  $\sigma$  are not location and scale parameters for the lognormally distributed variable, but rather location and scale parameters of its logarithm. If mean, standard deviation, and variance of the non-logarithmic set are denoted as  $m$ ,  $\sigma$ , and  $v$ , respectively, the two sets can be linked as

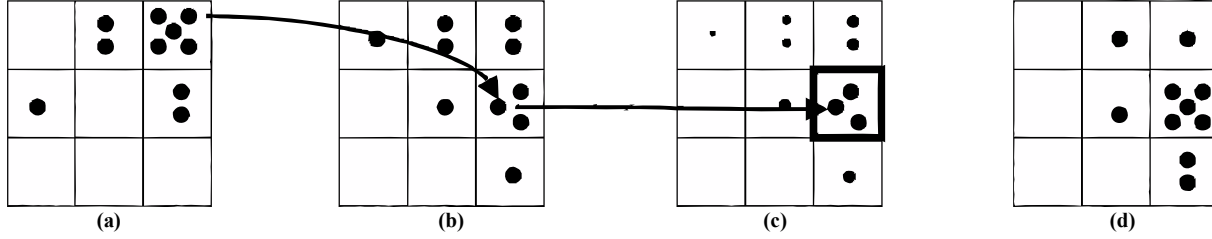
$$\mu = \ln \left( \frac{m}{\sqrt{1 + \frac{v}{m^2}}} \right) \quad (5.10)$$

$$\sigma = \sqrt{\ln \left( 1 + \frac{v}{m^2} \right)} \quad (5.11)$$

### 5.3.3 Step 3 – Resampling

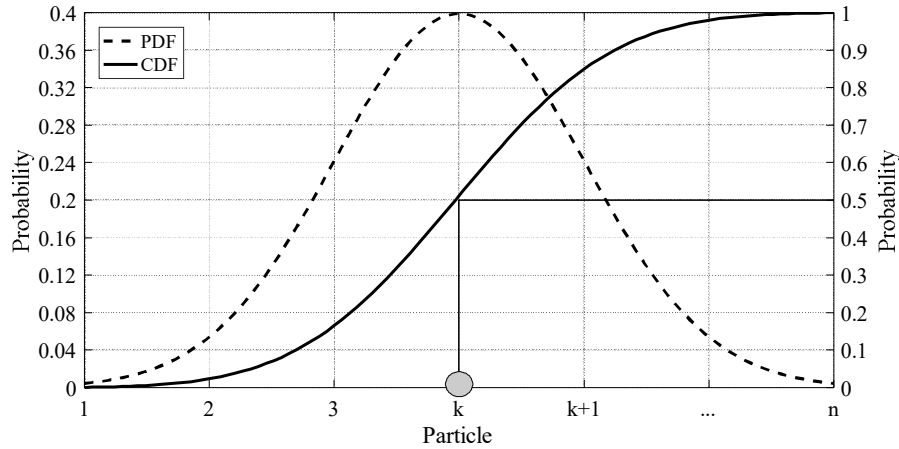
In the resampling step, based on the weights for each particle and the resampling method used, particles are eliminated or duplicated to form a new population from the previous population. The idea is to go back from weighted samples to unweighted samples to enforce diversity of the population and avoid degeneracy. As shown in Figure 5.4 after initiation process in Figure 5.4 (a), particles are propagated through the transition model to get the current states from the previous states as shown in Figure 5.4 (b). Next, all particles are weighted based on a probability distribution or fitness criterion as mentioned in step 2 of the particle filter process, which results in weighted particles as shown in Figure 5.4 (c). The weighted particles are prone to degeneracy, and potential solution spaces may disappear if weighted particles are continued with. Therefore, using a resampling technique, the new sample set is generated having unweighted particles, which essentially represents

the weights of the previously weighted particles with the number of particles rather than associated weights. This process can be observed visually in the transition from Figure 5.4 (c) to Figure 5.4 (d).



**Figure 5.4 Particle filtering process visual representation**

Among other methods, the inverse cumulative distribution function (CDF) resampling method is employed in this study. The CDF of a real-valued random variable  $X$  or its PDF, evaluated at  $x$ , is equal to the probability that  $X$  would take a value  $\leq x$ . Therefore, for a continuous distribution, it equals the area under the PDF from  $-\infty$  to  $x$ . Figure 5.5 illustrates the inverse CDF process where first, a CDF is constructed from the likelihood in Eq. (5.4). Next, a random value between 0 and 1 is generated (e.g. 0.5 in Figure 5.5) and used to find the corresponding sample that has the same CDF among other particles (shown as a circle on the  $x$  axis). It should be noted that since the CDF in this case is discrete, the closest sample to the corresponding CDF value is selected as the new sample. Repeating this process  $n$  times, generates  $n$  new samples from the previous weighted particles. The newly generated particles are the *posterior* distribution at the current ( $k$ th) step and the *prior* distribution at the following ( $k + 1$ th) step.



**Figure 5.5 Inverse CDF resampling method**

## 5.4 Predicting the Damage State and RUL

Once the estimated parameters are obtained from the particle filter, which can handle non-Gaussian additive noise unlike other methods, through the process detailed above, the future damage state and RUL can be predicted by propagating the damage state

through the damage model until it reaches a pre-set threshold. The remaining useful life of the unit is then calculated as the difference between the end of life (where damage model extrapolation intersects with failure threshold) and the current time (observation of faulty behaviour). This can be formulated in mathematical form as [183]

$$RUL = t_{end} - t \quad (5.12)$$

where  $t_{end}$  refers to end of life for the unit and  $t$  refers to the current time. Figure 5.6 illustrates the schematics of the RUL prediction process input/outputs. In Figure 5.6, the two dashed lines surrounding the true model's extrapolation, represent the prediction interval of the damage state. While the distribution curve at the failure threshold line represents the distribution of time when the damage state reaches the threshold.

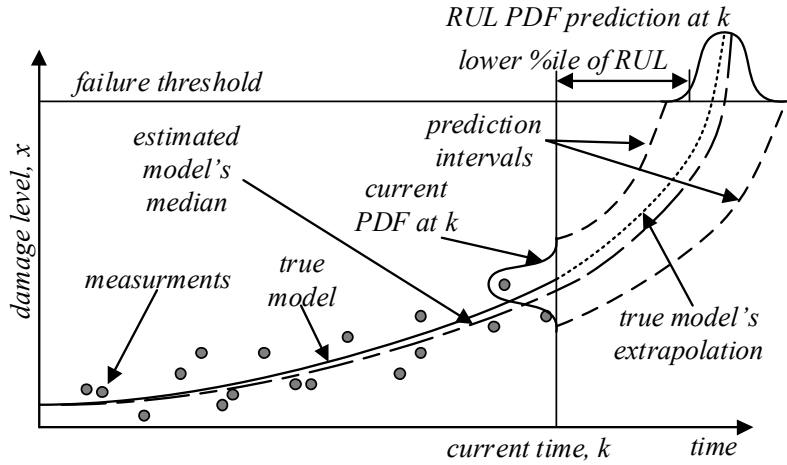


Figure 5.6 RUL prediction process schematics

In order to obtain the RUL distribution, the distribution of time needs to be subtracted from the threshold. It should be noted that in the prognosis step, only the damage state is transmitted and damage model parameters are no longer updated. In addition, during the prognosis step, the measurement error with the updated standard deviation is added to the damage state for accuracy.

The novelty in this approach lies in its 2-step process. In the literature, as discussed earlier, fault prognosis is not investigated for satellite systems specially actuators. Most recently Nguyen et al. [173] have examined the induction motors but for terrestrial applications. In addition, they use system measurements to feed data to the PF estimator for RUL predictions, however, in the approach proposed in this thesis, the proposed CAUKF in this thesis is used to estimate the non-measurable system parameter  $k_t$  as HI and feed it to the PF estimator for damage model reconstruction and RUL predication. Therefore, the mathematical formulation of the problem addressed in this chapter can be represented as follows:

**Step 1: Estimate  $k_t$** 

As discussed earlier, the first step is to estimate non-measurable parameter  $k_t$  from RW measurable states  $\omega$  and  $I$  as presented in Chapter 2 with

$$\begin{aligned}\dot{I}_m &= G_d \omega_d [f_3(\omega, I_m) - f_5(\omega)] - \omega_d I_m + G_d \omega_d V_{Comm} \\ \dot{\omega} &= \frac{1}{J_w} \{f_1(\omega) + k_t I_m [f_2(\omega) + 1] - \tau_v \omega - \tau_c f_4(\omega) + \tau_{noise}\}\end{aligned}\quad (5.13)$$

where

$$\begin{aligned}f_1(\omega) &= C \sin\left(\frac{Nt}{2}\omega\right) \\ f_2(\omega) &= B \sin(3Nt\omega) \\ f_3(\omega, I_m, V_{bus}) &= \frac{\exp[-aV(\omega, I_m, V_{bus})]}{1 + \exp[-aV(\omega, I_m, V_{bus})]} V(\omega, I_m, V_{bus}) \\ f_4(\omega) &= \frac{1 - \exp(-a\omega)}{1 + \exp(-a\omega)} \\ f_5(\omega) &= \frac{k_s[\omega - \omega_s f_4(\omega)]}{2} \left\{ \frac{1}{1 + \exp[-a(\omega - \omega_s)]} + \frac{1}{1 + \exp[a(\omega + \omega_s)]} \right\} \\ V(\omega, I_m, V_{bus}) &= k_f \left[ V_{bus} - 6 - \frac{1}{1 + \exp(-aI_{bus})} (1 + R_{in} I_{bus}) \right. \\ &\quad \left. - \frac{1 - \exp(-ak_e \omega)}{1 + \exp(-ak_e \omega)} k_e \omega \right]\end{aligned}\quad (5.14)$$

**Step 2: Add noise to estimated  $K_t$** 

In order to investigate performance of the estimated RUL from proposed PF approach, it is important to evaluate the performance under different types of additive noise. Particularly, non-Gaussian noise needs to be evaluated because unlike other KF family methods, PF does not rely on the Gaussian distribution for the added noise to the measurements.

$$\tilde{k}_t = \hat{k}_t + v_k \quad (5.15)$$

where  $\tilde{k}_t$  is the generated measured  $k_t$  fed to the PF for degradation model identification.  $\hat{k}_t$  is the estimated  $k_t$  from the CAUKF as discussed in Step 1. And  $v_k$  is the measurement noise added to the estimated  $k_t$  before feeding to PF. It is important that there is no restriction on the type of noise that  $v_k$  represents, since PF is capable of handling Gaussian and non-Gaussian noise. Detailed discussions on convergence theories and results for various types of PF are provided in a survey by Crisan and Doucet [184].

**Step 3: Identify degradation model parameters using PF**

Using the input measurements from Step 2, PF runs sequential Monte Carlo simulations as discussed earlier and identifies model parameters in eq. (5.1). Therefore, the system for which the PF identifies models can be presented as

$$\begin{cases} \dot{x} = f(k_t) = ae^{-bt} \\ y = x = k_t + v_k \end{cases} \quad (5.16)$$

#### Step 4: Extrapolate $k_t$ model and determine RUL

The last step of this process is to extrapolate the estimated degradation model in time and intersect it with the failure threshold defined for the HI, which in here is  $k_t$ . The time at which these two intersect is the end of life time and the remaining useful time is defined as the difference between the current time and end of life time.

Since PF provides a series of estimates for all of its agents/particles, the final RUL is presented as a distribution of these estimates. Furthermore, one can identify mean, median, or another property of this distribution to provide a value for the RUL. Further discussions on the choice for this property as well as case studies are presented in the following section.

### 5.5 Results and Discussion

Numerical simulations were conducted to evaluate the performance of the proposed prognosis scheme. Numerical integration of the states was done in MATLAB using classical Runge-Kutta method (RK4) [172] with a sampling period of  $T_s = 0.01$  s. White-noise was added to the integration outputs of the RW for measurement noise. The total simulation time = 100 s and simulations were performed on an Intel® Core™ i7-4790 CPU with 3.60 GHz power PC and 16.00 GB RAM. The overall simulation setup can be referred to the flow shown in Figure 5.12. Further simulation parameters are listed in Table 3.5. First, new fault scenarios are considered for the RW parameter changes that were not discussed in Chapter 3. Since the damage model in this chapter is considered to be of exponential form, parameter's behaviour from the inception of the fault also needs to be of exponential form, which was not considered in Chapter 2.

#### 5.5.1 Exponential Decline

In this section, new fault scenario wherein value of  $k_t$  degrades exponentially is introduced (Table 5.1). As can be seen in this scenario, unlike the other ones we introduced in Chapter 2, fault inception continues until the end of simulation time. This is to show the fault progression is gradual and lasts until the end of life for the component. This is the reason that incipient faults are particularly important when it comes to fault prognosis and RUL estimation.



**Table 5.1 Exponential decline in Motor Torque Constant fault profile**

Time $V_{bus}$ (s)	$V_{bus}$ (V)	Time $K_t$ (s)	$K_t$ (Nm/A)
$t < 30$	8	$t < 30$	0.029
$30 \leq t < 100$	8	$30 \leq t < 100$	$0.019 + 0.01 \exp(-0.064 * t)$

Figure 5.7 illustrates state and parameter estimations for this fault scenario. As can be seen, all states and parameters are accurately estimated. Figure 5.8 shows how  $\beta$  and  $\beta_0$  values change for this fault scenario. Figure 5.9 depicts states and parameters residuals, which further confirms accurate estimation of the state and parameters by providing bounded residuals. As can be seen, since there are no sudden jumps in the trend for parameters, there are no sudden spikes in the  $\beta$  trend either. This means that in this case the CAUKF behaves very much like AUKF because there are not many manual adjustments of the filter covariance matrices. Figure 5.10 shows the comparison between different methods for  $k_t$  estimates. This figure is important since the prognosis process starts after this trend is captured by the fault detection scheme. As discussed earlier, inputs to the fault prognosis scheme are provided by the estimates calculated from the parameter estimation algorithm. Finally, Figure 5.11 illustrates comparison between different methods for  $v_{bus}$  estimates. As  $v_{bus}$  remains unchanged in this fault scenario, this figure shows a straight line and different method perform relatively the same for this estimation.

**Table 5.2 Exponential decline in Motor Torque Constant performance measures**

Algorithm	Noise factor $\phi = 1$			Noise factor $\phi = 10$		
	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )	time (sec)	MSE ( $K_t$ )	MSE ( $V_{bus}$ )
UKF	2.31	$7.04 \times 10^{-8}$	$3.18 \times 10^{-3}$	2.25	$7.51 \times 10^{-8}$	$3.20 \times 10^{-3}$
AUKF	2.28	$4.58 \times 10^{-8}$	$3.18 \times 10^{-3}$	2.28	$5.28 \times 10^{-8}$	$3.19 \times 10^{-3}$
CAUKF	3.35	$4.55 \times 10^{-8}$	$3.18 \times 10^{-3}$	3.39	$5.13 \times 10^{-8}$	$3.19 \times 10^{-3}$

Summary of the performance measures for two different noise levels is provided in Table 5.2. Results in Table 5.2 show that CAUKF outperforms both UKF and AUKF in terms of decreasing MSE for both RW parameters.

The flow of the prognosis is illustrated in Figure 5.12 where the output of the UKF parameter estimation, which is  $k_t$  is fed to the particle filter fault prognosis scheme. An additional measurement noise is added to  $k_t$  before it is fed to damage model identification where it is  $v_k \sim N(0, \sigma)$  and  $\sigma$  is an unknown standard deviation to the identification scheme. Therefore, the unknown parameters for the damage model become  $\Theta = [k_t, \theta(=b), \sigma]$ ; where  $k_{t_k}$  is also included and obtained based on the model parameter  $b_k$ .

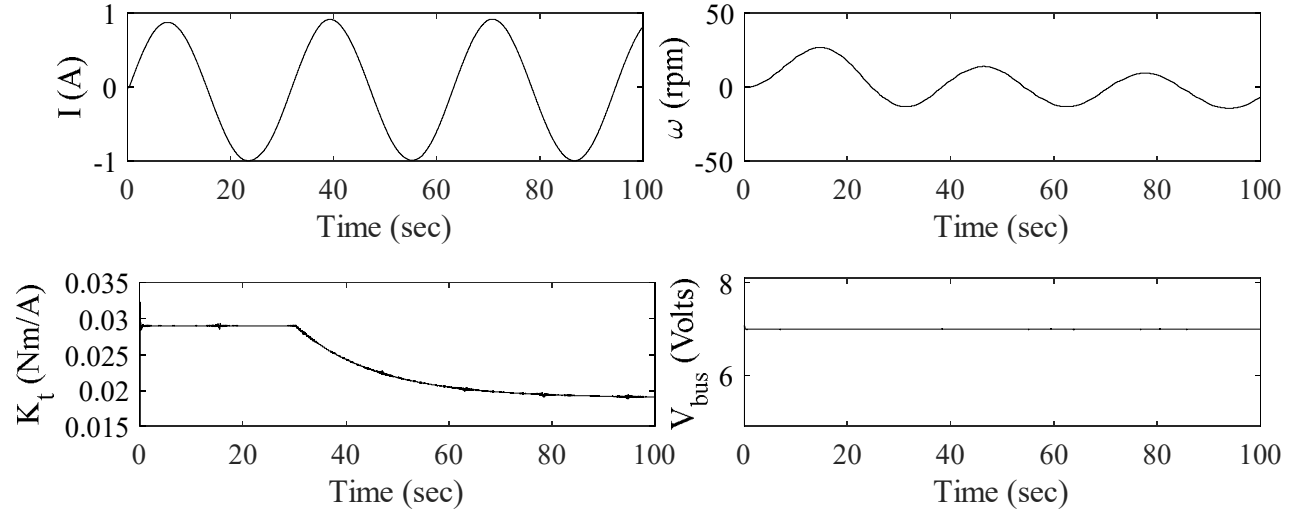


Figure 5.7 State and parameters estimates for incipient fault with  $\phi=10$

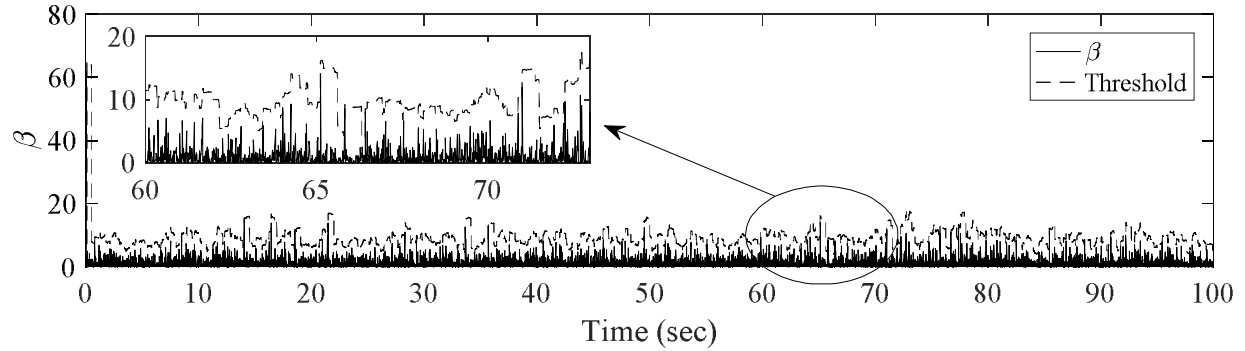


Figure 5.8  $\beta$  and  $\beta_0$  for incipient fault with  $\phi=10$

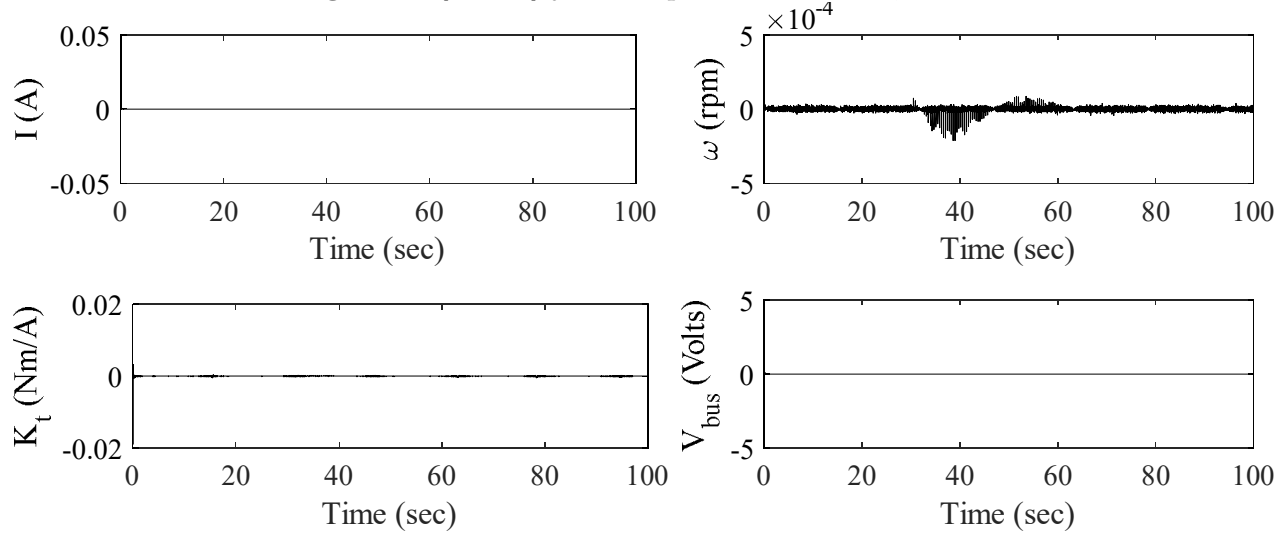


Figure 5.9 State and parameters residuals for incipient fault with  $\phi=10$

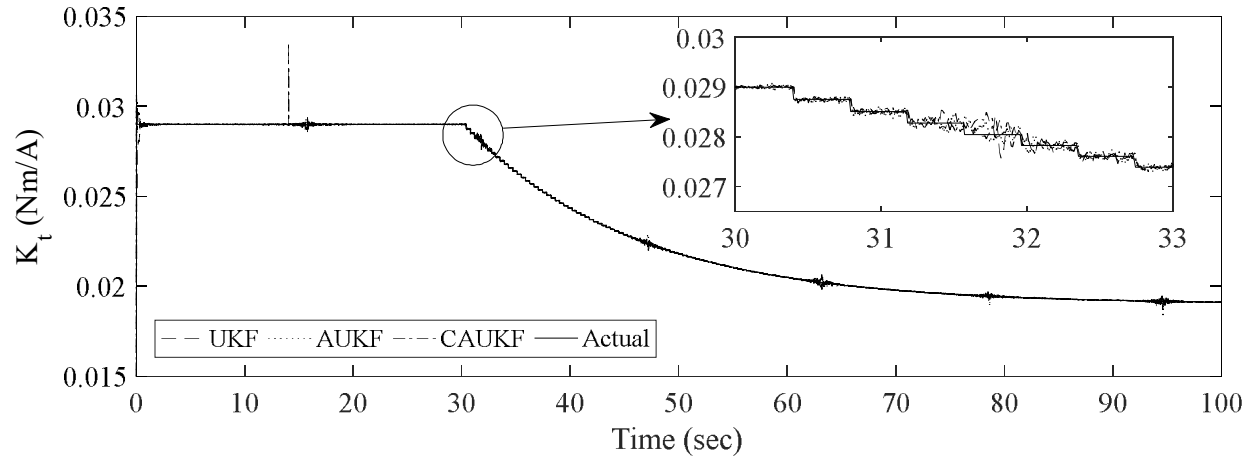
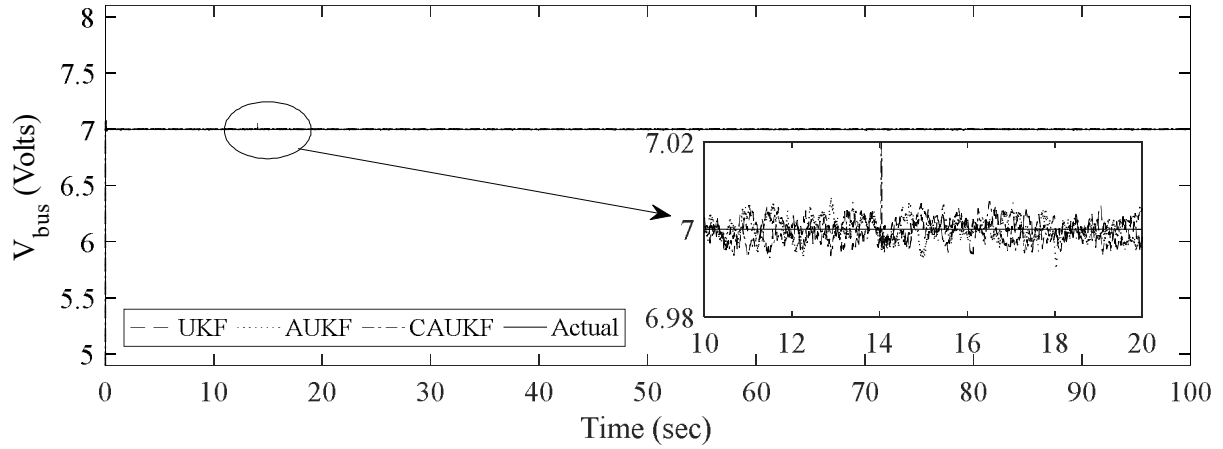
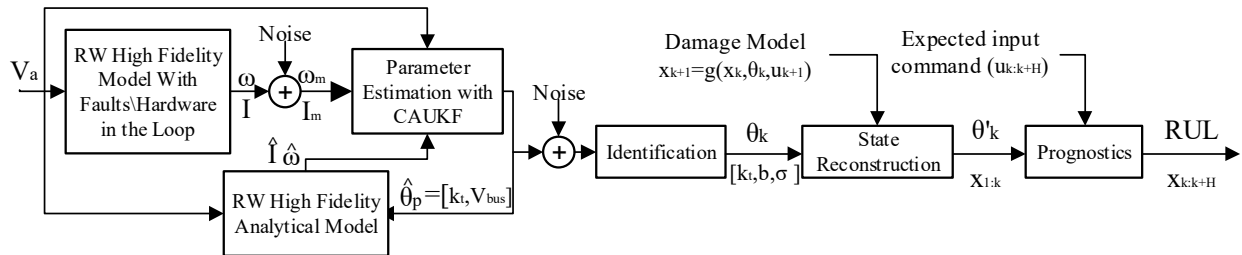

 Figure 5.10 Comparison of  $K_t$  estimates for incipient fault with  $\phi=10$ 

 Figure 5.11 Comparison of  $V_{bus}$  estimates for incipient fault with  $\phi=10$ 


Figure 5.12 Flow of the fault detection leading to fault prognosis

After initializing the fault scenario and estimating  $k_t$  and  $v_{bus}$ , where only  $k_t$  is further fed to PF for RUL estimation, the next step is to initialize the fault prognosis scheme. In this scenario, we assume the data coming from the estimator are available for a number of measurements after fault inception. This gives the prognosis algorithm *priori* information to start the prognosis. We assume that the data provided to the prognosis algorithm are measured every day after inception of the fault; therefore, the time unit for

the prognosis is considered to be *days*. Next based on the damage model provided in Eq. (5.1) we need values for  $a$  and  $b$  to setup the true model values (Table 5.3). Since there are no *priori* information about  $\Theta$  is not available, the initial distribution for all the parameters is assumed to be uniform with lower and upper bounds provided in Table 5.3. The number of particles can be chosen between 500 to 5000 particles depending on the complexity of the problem. Next, there is significant level ( $\alpha$ ) [185], [186] for calculating confidence interval (C.I.) and prediction interval (P.I.). Significant level can be chosen as 5, 2.5, or 0.5 for 90%, 95% or 99% intervals, respectively.

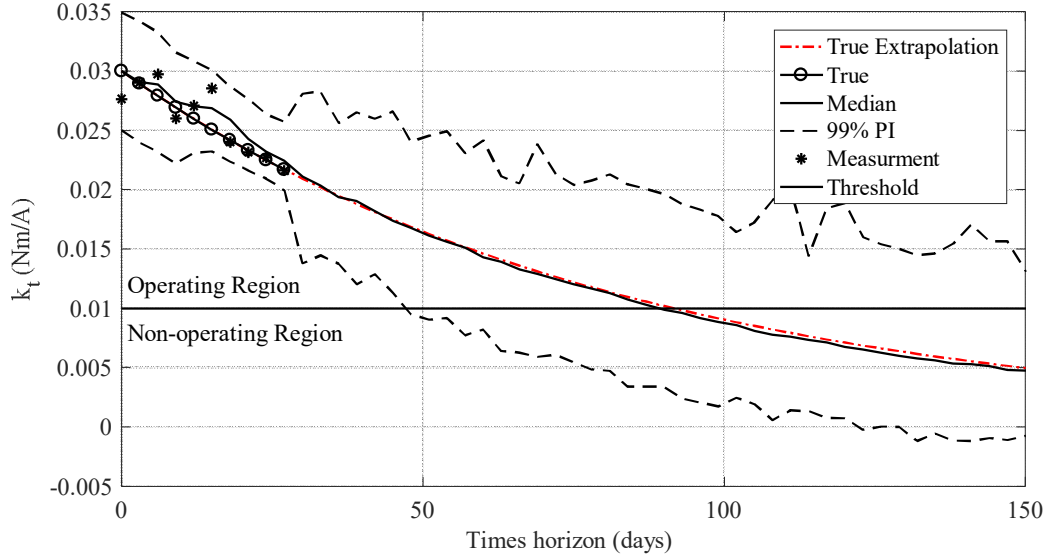
**Table 5.3 Fault prognosis simulation parameters**

Parameter	Value
Time unit	days
Data collection delay ( $dt$ )	3 [time unit]
$a_{true}$	0.03 [Nm/A]
$b_{true}$	0.012
Failure threshold ( $k_{ts}$ )	0.01 [Nm/A]
$k_{t0}$	$\sim U(0.025, 0.035)$
$b_0$	$\sim U(0, 0.02)$
$\sigma_0$	$\sim U(0, 0.01)$
$n$	500
$\alpha$	0.5 (99%)

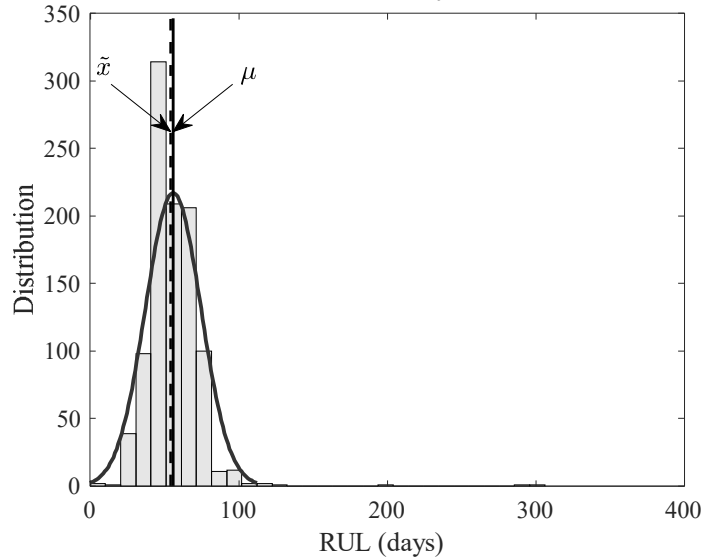
### 5.5.2 Normal Distribution

Results for a particular simulation under normal distribution are shown in Figure 5.13 to Figure 5.16. Figure 5.13 illustrates the available data and the forecasted data on the system combined. In this figure, True curve represents the degradation model with designed values provided in Table 5.3. As can be seen, the data coming from the estimator and were discussed in the previous section, are now treated as measurement data for the prognosis algorithm. Extra measurement noise is added to the data points to ensure completeness of the results. The threshold line is set at 0.01, which separates the operating and non-operating regions. It is also used for intersecting the true extrapolation of the model with the threshold to find the true RUL of the system. Next, 99% prediction intervals and median of the data show forecasted progression of the system parameter and the confidence in its value at each point in time. It can be observed that as soon as there are no more measurements available to the prognosis algorithm, the prediction interval widens to compensate for the loss of confidence in the forecast data. Figure 5.14 illustrates the RUL distribution for the system based on  $k_t$  reaching its threshold. In this figure,  $\mu$  is the mean of the data and  $\tilde{x}$  is the median of the data. Median can be a better representative of the estimated RUL for the component in this case [183]. In order to obtain the mean and median, a normal

distribution curve (normal distribution curve) was fitted to the estimated data spread (bars), and the mean was obtained from that fit. Median could be obtained from the estimated data spread (bars) directly or from the fitted distribution.



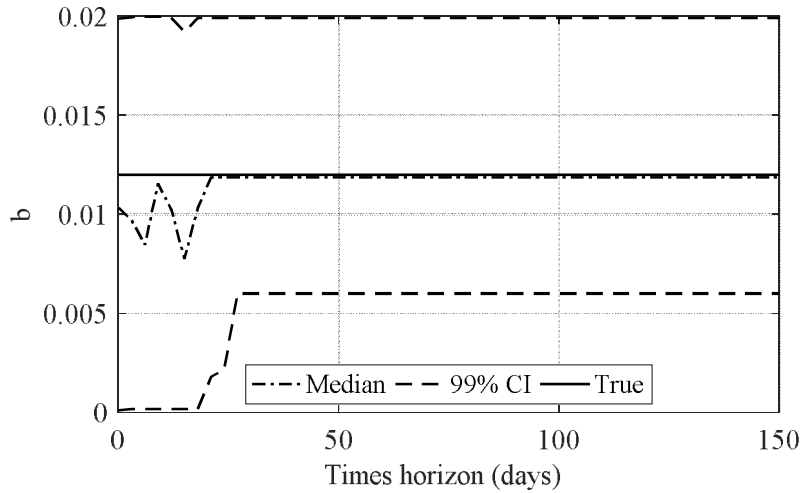
**Figure 5.13** Fault prognosis for  $k_t$  under normal noise



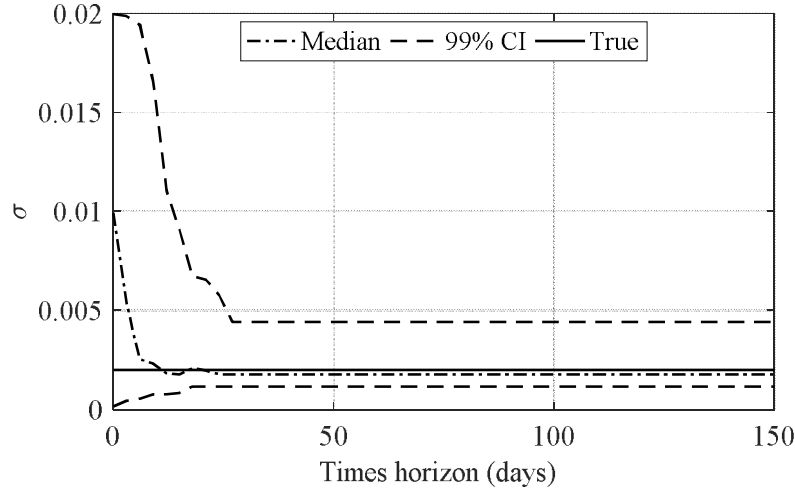
**Figure 5.14** RUL distribution based on  $k_t$  damage model for normal noise

Figure 5.15 and Figure 5.16 show the fault prognosis results obtained for the damage model parameter  $b$  and measurement noise  $\sigma$ . The median of the data and the true data seem to match closely after a few iterations (days) in the prognosis. This means that the model has matched accurately with damage model considered in generating measurement data after the fault inception. The confidence intervals seem to narrow down as the estimated value converges to the true value to provide more confidence in the estimations. This can be looked at from the UKF point of view where covariance matrix elements become smaller to

show higher confidence in the estimates.



**Figure 5.15 Fault prognosis for  $b$  under normal noise**



**Figure 5.16 Fault prognosis for  $\sigma$  under normal noise**

Detailed results for the prognosis of the parameter degradation are provided in Table 5.4 for simulation with 500 particles under a normally distributed sample set, using inverse CDF resampling method. Since the simulation setup is based on random data (noise/particle) generation, outcomes of the simulations can vary in each run. Therefore, a set of 100 simulations were conducted, and minimum, average, and the maximum value of the whole set were reported under each measure. It is also important to note that sometimes particles may not converge to a particular solution and system may keep running. For this reason, it was considered to use a computation time threshold above which the simulation would be terminated, and that particular simulation results were nullified and removed from simulation set. For the results provided in Table 5.4, out of 100 simulation attempts, 93 were successful, and the rest failed to converge within the specified threshold of 10 seconds. Total

computation time for this set was 101.93 sec and minimum, average, and maximum computation time for individual runs are reported in the table under  $t$  parameter.

**Table 5.4 Fault prognosis for a normally distributed sample set with 500 particles**

Metric	RUL (days)						
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	$t$ (sec)
Min	64.56	37.25	33	3	57	7.39	0.053
Mean	64.56	70.30	61.26	12.70	286.05	46.04	0.33
Max	64.56	256.30	264	57	2046	277.45	4.99

From the results provided in Table 5.4, it can be observed that median ( $\tilde{x}$ ) is a better representation of the RUL compared to mean ( $\mu$ ) of the data. It can also be observed that the estimated RUL is reasonably close to the actual RUL by 3.3 in value or  $\sim 5\%$  error. Table 5.5 presents results for the prognosis for same system in Table 5.4 with 100 particles instead of 500. This simulation took 8.17 sec to complete and all 100 simulations were successful. It can be observed that in this case, mean of the data provides a better estimated of the actual RUL compared to the median. However, if we keep our estimated value as the median of the estimations, it is clear that with smaller number of particles the accuracy has decreased to 7.54 in value or  $\sim 12\%$  error.

**Table 5.5 Fault prognosis for a normally distributed sample set with 100 particles**

Metric	RUL (days)						
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	$t$ (sec)
Min	64.56	19.05	18	3	48	4.26	0.021
Mean	64.56	65.05	57.02	17.85	211.72	36.99	0.079
Max	64.56	312.27	192	153	2328	494.00	0.973

Table 5.6 presents the results for the system in Table 5.4 and Table 5.5 with the same conditions except for the number of particles. In Table 5.6, 1000 particles were employed. The whole simulation took 182 sec to complete, and 94 attempts were successful out of the total 100. It can be clearly observed that increasing the number particles has improved the accuracy of the estimates by decreasing the error to 1.07 in value or  $\sim 1.6\%$ . However, the trade-off is the increased computation time.

**Table 5.6 Fault prognosis for a normally distributed sample set with 1000 particles**

Metric	RUL (days)						
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	$t$ (sec)
Min	64.56	35.56	33	3	63	7.41	0.09
Mean	64.56	75.23	63.49	12.11	302.71	56.08	1.16
Max	64.56	759.06	519.00	43.50	2005	657.57	8.44

### 5.5.3 Lognormal Distribution

In order to investigate other distributions, results for a simulation under lognormal distribution are shown in Figure 5.17-Figure 5.20. Figure 5.17 illustrates the available data and the forecasted data on the system. The descriptions on the Figure 5.17 are the same as Figure 5.13. Figure 5.18 illustrates the RUL distribution for the system based on  $k_t$  reaching its threshold. In this figure,  $\mu$  is the mean of the data and  $\tilde{x}$  is the median of the data, and also in lognormal distribution Mode is introduced as another important parameter.

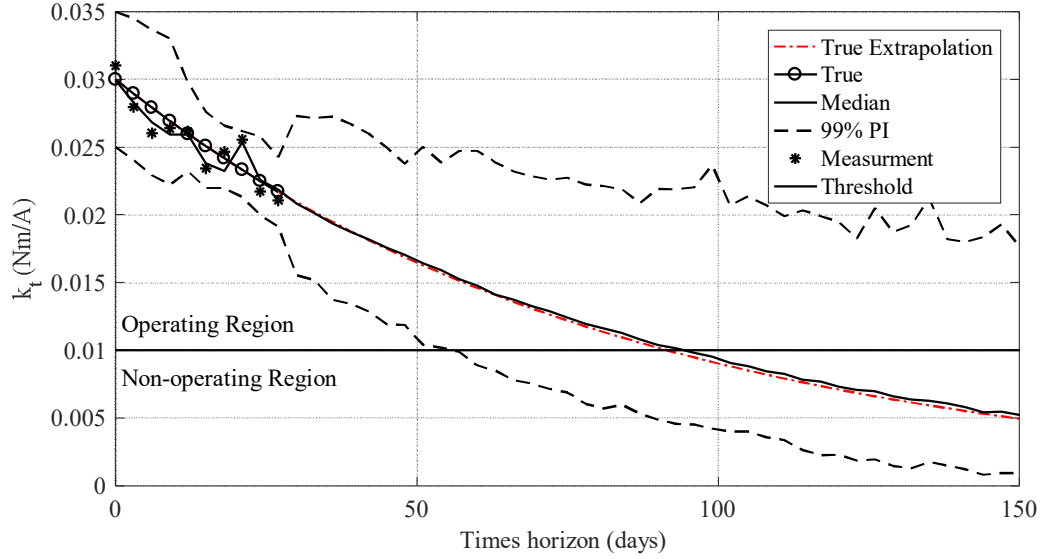


Figure 5.17 Fault prognosis for  $k_t$  under lognormal noise

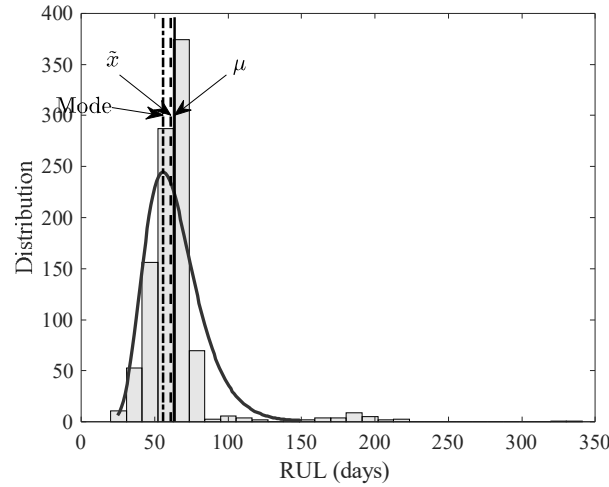
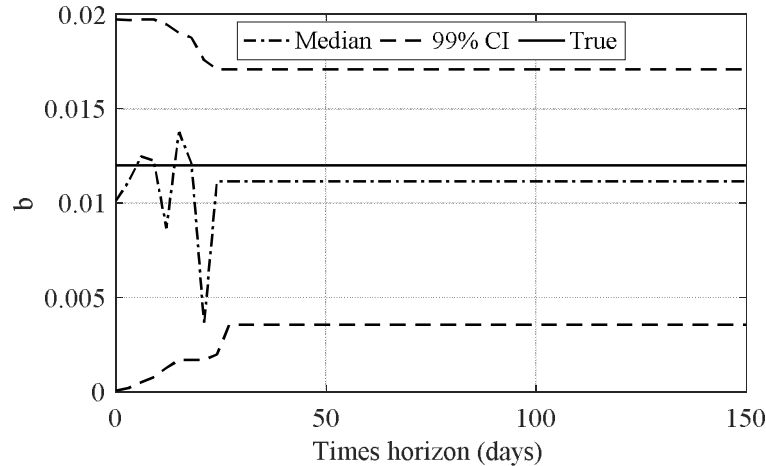
In order to obtain the mean and median, a lognormal distribution curve (normal distribution curve) was fitted to the estimated data spread (bars), and the mean was obtained from that fit. Median could be obtained from the estimated data spread (bars) directly or from the fitted distribution.

Figure 5.19 and Figure 5.20 show the fault prognosis results obtained for the damage model parameter  $b$  and measurement noise  $\sigma$ . Detailed results for the prognosis of the parameter degradation are provided in Table 5.7 for simulation with 500 particles under a log-normally distributed samples, using inverse CDF resampling method. For the results provided in Table 5.7, out of 100 simulation attempts, 99 were successful. Total computation time for this set was 53.68 sec. From the results provided in Table 5.7, median ( $\tilde{x}$ ) is considered as the representation of the RUL. The estimated RUL is close to the true RUL by 5.27 in value or  $\sim 8\%$  error. Table 5.8 presents results for the prognosis for same system in Table 5.7 with 100 particles instead of 500. This simulation took 16.5 sec to complete and 99 simulations were successful out of 100.



**Table 5.7** Fault prognosis for a log-normally distributed sample set with 500 particles

Metric	RUL (days)						
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	$t$ (sec)
Min	64.56	34.78	33	6	69	0.09	0.06
Mean	64.56	68.86	59.29	18.21	338.64	0.44	0.44
Max	64.56	215.10	120	45	2589	1.01	6.89

**Figure 5.18** RUL distribution based on  $k_t$  damage model for lognormal noise**Figure 5.19** Fault prognosis for  $b$  under lognormal noise

It is clear that with a smaller number of particles the accuracy has decreased to 8.65 in value or  $\sim 13\%$  error. This comes with the advantage of faster computation compared to 500 particles case. Table 5.9 presents the results for the system in Table 5.7 and Table 5.8 with the same conditions except for the number of particles. In Table 5.9, 1000 particles were employed. The whole simulation took 245.9 sec to complete, and 85 attempts were successful out of the total 100. It can be clearly observed that increasing the number particles has improved the accuracy of the estimated by decreasing the error to 4.67 in value or  $\sim 7\%$ . However, the trade-off is the increased computation time.

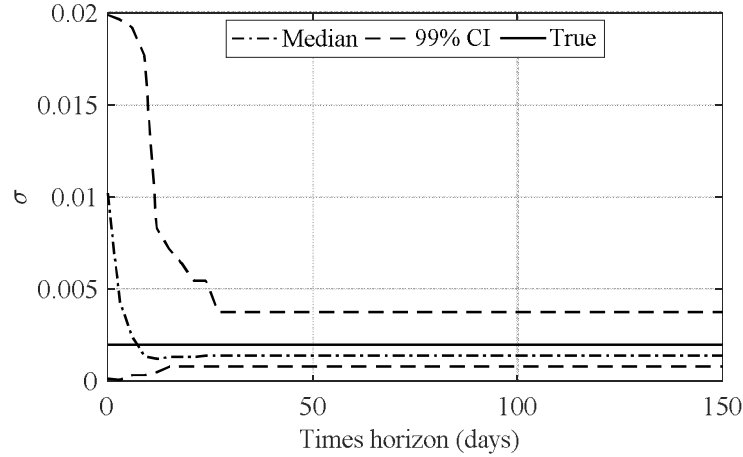
Figure 5.20 Fault prognosis for  $\sigma$  under lognormal noise

Table 5.8 Fault prognosis for a log-normally distributed sample set with 100 particles

Metric	RUL (days)						$t$ (sec)
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	
Min	64.56	26.53	21	3	51	0.10	0.02
Mean	64.56	63.88	55.91	19.45	181.42	0.42	0.06
Max	64.56	239.25	222	159	1002	1.05	0.35

Table 5.9 Fault prognosis for a log-normally distributed sample set with 1000 particles

Metric	RUL (days)						$t$ (sec)
	True	$\mu$	$\tilde{x}$	0.5%	99.5%	$\sigma$	
Min	64.56	35.18	30	3	63	0.13	0.12
Mean	64.56	71.30	59.89	17.54	437.65	0.48	1.12
Max	64.56	244.27	189	37.50	2359	0.99	7.36

## 5.6 Conclusions

In this chapter, a new 2-step approach was proposed for estimating the remaining useful life a RW onboard satellite. Adaptive unscented Kalman filter was employed to estimate non-measurable system parameters  $k_t$  and  $v_{bus}$ , of which only  $k_t$  was fed to PF as a HI. When inception of a fault was detected by the FDI algorithm, an exponential failure damage model was used along with a Particle Filter to adapt the deterioration trend of the system to measurements provided by the CAUKF. Next, the identified damage model was used to extrapolate the data and determine the RUL of the faulty RW. The proposed scheme was evaluated using two types of measurement noise: normal Gaussian and lognormal. Results showed successful estimation of the RUL with errors in estimating ranging from 1.5% to 13% in both cases. Therefore, it can be concluded that the proposed scheme can be a feasible approach for such application.

## CHAPTER 6

# Conclusions and Future Work

---

This thesis dealt with the problem of fault detection, isolation, and prognosis in actuators of the in-orbit satellite. The main modules of a CBM/PHM framework were discussed as fault detection module, fault isolation/identification module, and fault prognosis module. In fault detection module, the occurrence of a fault is detected followed by fault isolation/identification where location and severity of the fault are determined. Finally, fault prognosis provides an estimate for the RUL of the faulty component, which can help plan future maintenance or remedial actions in advance.

In light of the importance associated with CBM/PHM frameworks for today's complex systems, in this thesis, three major modules of this framework were investigated, and enhancement techniques were proposed and evaluated to address some of the limitations observed in the literature. Most importantly, the combination of these methods can provide a complete CBM/PHM flow to enhance reliability and availability of engineering system where expert knowledge and experienced staff are not available. Furthermore, the proposed framework can be extended from single satellites to multi-agent systems, using similar approach to the proposed algorithms, given the dynamics equations of the multi-agent system as well as proper implementation of the schemes.

### 6.1 Highlights

In Chapter 3, a new 2-step fault detection scheme was proposed to address the agility and accuracy limitations in other related literature. The proposed scheme was designed based on adjusting the covariance matrix for the adaptive unscented Kalman filter, which estimated two non-measurable parameters of a high-fidelity RW on-line. The process was a two-step scheme where first a fault was annunciated and second filter was adjusted. The performance of the proposed scheme was evaluated on all possible fault scenarios including abrupt, transient, intermittent and incipient. it should be noted that incipient faults are not investigated in many of the open literature due to their ongoing variable behavior. In addition, mild and severe noise levels were considered for the measurement noise to evaluate the performance of the proposed scheme with regards to noise. Results showed that the proposed scheme outperformed all other methods it was compared to in terms of accuracy and agility in detecting faults and adjusting estimates.

In Chapter 4, a new 2-step fault isolation scheme was proposed where the applicability of the technique was broadened, and implementation of it was made easier.

The technique relies on a multiple model approach where multiple version of the system run in parallel to the nominally behaving unit. At each time, when a fault is annunciated, parallel models, each pertaining to a particular faulty mode, run in parallel to the nominal unit and outputs are compared using residuals to create a likelihood distribution of the faulty cases and when a confidence level is achieved, faulty mode is determined and current system mode is switched to the determined faulty mode to be able to investigate future faults. After switching is done, parallel models stop running and fault detection scheme monitors system behavior for anomalies. The benefits of this approach can be mentioned as (1) since no simplification or linearization is considered for the model, range of operation in this approach is broader compared to other multi-model approaches where, due to the estimator used (e.g. KF or EKF), linearization is required, and inaccuracies can occur. (2) Any initial distribution can be assumed for the faulty modes, and no uniform or Gaussian distribution assumption is necessary. (3) Since all models are not running at all times and parallel models only run during the isolation process, computational resources can be allocated to other necessary sections onboard or remain idle to save power consumption. (4) Hierarchical architecture of the scheme allows for demanding certain information and/or resources at each level, saving the need to require unnecessary information or resources at all times. The proposed scheme was tested on an RWA with four RW in two different configurations (standard four-wheel and pyramid) onboard a satellite in orbit and under various disturbance torques. Results showed more than 99% success is properly isolating the faulty units in case of faulty behavior, uniform and normal distributions were evaluated for the initial distribution of the faulty models to ensure applicability of the scheme to various scenarios. Results showed successful isolation regardless of the initial distribution considered for faulty modes. In addition, concurrent and consequent faults were investigated to evaluate the performance of the proposed scheme in case of faults occurring at the same time or in sequence, respectively. Results showed the successful isolation of the faults in both cases.

In Chapter 5, a new hybrid fault prognosis scheme using UKF and PF was proposed. First, based on the literature pertaining to RW failures onboard satellites, key components and measurable and non-measurable parameters were identified. Next, a fault scenario was introduced based on the data collected from the literature, and simulated data were fed to the fault detection scheme, which then provided simulated measurements to the prognosis scheme. The prognosis scheme used a degradation model and the measurements provided by the estimator to adjust the degradation model to the current situation of the system. After adjusting the degradation model, its trend was extrapolated to find its intersection with the threshold for the critical parameter representing the component under wear and tear, to find the RUL of the faulty unit. Using the true data and estimated RUL, results were compared to evaluate the performance of the proposed scheme. Results showed that

the proposed scheme was able to estimate the RUL of the system with errors as low as 1.5%. In addition, multiple scenarios for the measurement noise added to the estimator's estimates were considered including normal and lognormal distributions. Results were satisfactory for both cases, and RUL was estimated successfully.

## 6.2 Summary of Contributions

As mentioned earlier the main focus of this thesis was to develop a framework for CBM/PHM of a complex engineering system, particularly satellite's actuators. In the process, three major modules were investigated, and enhancement proposals were made to improve accuracy and reliability of each module. All numerical simulations were conducted on a high fidelity '*ITHACO Type A*' RW model. A summary of the contributions in each module can be provided as follows:

**Fault Detection:** A new 2-step fault detection approach was proposed that (1) considers a high-fidelity model of an RW system, (2) considers states error covariance matrix when abrupt changes occur to ensure agile tracking of non-measurable system parameters, (3) is applicable to all fault scenarios including abrupt, transient, intermittent, and incipient, (4) performs well under various noise levels.

The proposed method was also implemented and analysed through numerical simulations for a CMG assembly onboard satellite where results showed superior performance of the proposed method compared to other similar available techniques in the literature.

**Fault Isolation:** A new 2-step fault isolation approach was proposed based on a hierarchical MM scheme. The fault is detected using residual sequence followed by the first level of isolation in the formation level using residual windowing, second level of isolation using Bayes' probability theorem to locate the faulty unit and the third level of isolation using parameter estimation to locate the faulty parameter with limited *a priori* information. This approach (1) considers a high-fidelity model for each RW in the RWA mounted on an in-orbit satellite with external disturbances, (2) does not require linearization of the system model, (3) due to hierarchical architecture only requires certain information at certain levels, (4) performs well regardless of the initial distribution considered for faulty modes.

A variation of the proposed method was also implemented and analyzed through numerical simulations for a CMG assembly onboard satellite where results showed satisfactory results comparable to the RWA case.

**Fault Prognosis:** A new hybrid fault prognosis approach was proposed for RW onboard satellite using a mathematical model of the system, degradation model, and synthetic data to investigate the feasibility of employing PF for such application. The proposed approach (1) employs particle filter for system identification on degradation model, (2) performs well under different noise distributions including normal and lognormal, (3) does

not require linearization of the system model, (4) can estimate RUL with errors as low as 1.5%.

### 6.3 Future Work

A number of future works can be envisioned for this thesis considering the extensively growing requirements and applicability of fault diagnosis, prognosis, and health management (DPHM) technologies across science and engineering fields:

**Investigating the proposed framework on other Systems:** Although in this thesis, substantial simulations were provided for the proposed framework on an RW system, it is interesting to investigate the performance of the proposed framework on other complex systems.

**Investigating integrity of the proposed framework:** Although each module of the proposed framework was evaluated individually or in conjunction with another module, all three modules were not investigated together. It would be valuable to evaluate the performance of the proposed framework when all modules are integrated, and outputs of one module are fed to the following modules for a complete DPHM process.

**Investigating the proposed framework in a complete close-loop system:** Another aspect that was not focused on in this thesis was the controller aspect of the ACS. It is understood that the outcomes of the CBM/PHM framework can greatly help accommodate unforeseen circumstances in complex systems. However, the controller unit is responsible for considering such changes in the system and applying alternations/reconfigurations to maintain system performance or at least partially accomplish its expected mission. Therefore, implementing this framework along with a control unit to evaluate the overall system performance can be a valuable asset.

**Investigating the proposed framework in other architectures/topologies for multi-agent systems:** In this thesis, single satellite system and two different architectures of formation flying were considered, and satellite's ACS actuators were investigated for fault diagnosis and prognosis. However, the proposed framework can be extended to other multi-agent systems and architectures where the difference would be in the proper derivation of the multi-agent dynamics equations and proper implementation of the proposed framework.

**Investigating hybrid approaches:** Due to lack of experimental data, this thesis was mostly based on simulated and synthetic data. It would be interesting to evaluate the performance of the proposed framework when it is combined with data-driven approaches built on experimental data. Such approach could expand the applicability of the framework to cases and systems where the mathematical model of the system is not available. It can also compensate for inaccuracies or inconsistencies between mathematical models and hardware.

# References

- 
- [1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, 2nd ed. Springer, 2006.
  - [2] G. Vachtsevanos *et al.*, *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*, 1st ed. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2006.
  - [3] N. Tahri, C. Hamrouni, and A. M. Alimi, “Study of Current Femto-Satellite Approches and Services,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 5, pp. 148–153, 2013.
  - [4] G. Konecny and Institute, “Small Satellites – A Tool for Earth Observation? Gottfried,” in *XXth ISPRS Congress-Commision*, 2004, pp. 1–3.
  - [5] Godard, K. D. Kumar, and A.-M. Zou, “A novel single thruster control strategy for spacecraft attitude stabilization,” *Acta Astronaut.*, vol. 86, pp. 55–67, May 2013.
  - [6] H. Kayal, F. Baumann, K. Briess, and S. Montenegro, “BEESAT: A Pico Satellite for the On Orbit Verification of Micro Wheels,” in *2007 3rd International Conference on Recent Advances in Space Technologies*, 2007, pp. 497–502.
  - [7] S. Trowitzsch, F. Baumann, and K. Brieß, “BEESAT-2: A Picosatellite Demonstrating Three-Axis Attitude Control Using Reaction Wheels,” in *65th International Astronautical Congress*, 2014.
  - [8] M. Tafazoli, “A study of on-orbit spacecraft failures,” *Acta Astronaut.*, vol. 64, no. 2–3, pp. 195–205, Jan. 2009.
  - [9] E. Buchen and D. DePasquale, “2014 Nano/Microsatellite Market Assessment,” Atlanta, GA, 2014.
  - [10] B. Doncaster, C. Williams, and J. Shulman, “2017 Nano/Microsatellite Market Assessment,” Atlanta, GA, 2017.
  - [11] J. D. Harrington, “NASA Selects Explorer Investigations for Formulation,” *NASA press*, Washington, DC, DC, 2016.
  - [12] R. Cowen, “The wheels come off Kepler,” *Nature*, vol. 497, no. 7450, pp. 417–418, May 2013.
  - [13] A. Pasetti, “Background on Attitude and Orbit Control Systems (AOCS),” 2002. [Online]. Available: <https://www.pnp-software.com/AocsFramework/AocsBackground.html>. [Accessed: 11-May-2017].
  - [14] N. Abreu, “Fault Diagnosis with Adaptive Kalman Filters and CMG Design for Picosatellite ACS,” Ryerson University, 2010.
  - [15] S. Murugesan and P. S. Goel, “Fault-tolerant spacecraft attitude control system,” *Sadhana*, vol. 11, no. 1–2, pp. 233–261, Oct. 1987.
  - [16] E. Sobhani-Tehrani and K. Khorasani, “Identification For Nonlinear Systems Using Hybrid Approach,” *Master’s Thesis*. p. 12–13,15,18,37,92, 2008.
  - [17] J. T. Cheung and G. Stephanopoulos, “Representation of process trends - Part 1. A formal representation framework,” vol. 14, no. 4/5, pp. 495–510, 1990.

- 
- [18] Godard, “Fault Tolerant Control of Spacecraft,” Ryerson University, 2010.
  - [19] J. J. Gertler, “Survey of Model-based Failure Detection and Isolation in Complex Plants,” *IEEE Control Syst. Mag.*, vol. 8, no. 6, pp. 3–11, Dec. 1988.
  - [20] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, “Model-Based Fault Diagnosis for Aerospace Systems: A Survey,” *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, vol. 226, no. 10, pp. 1329–1360, Jan. 2012.
  - [21] 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 Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.
  - [22] Z. Gao, C. Cecati, and S. Ding, “A Survey of Fault Diagnosis and Fault-Tolerant Techniques Part II: Fault Diagnosis with Knowledge-Based and Hybrid/Active Approaches,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3768–3774, 2015.
  - [23] P. Manikandan and M. Geetha, “Takagi Sugeno Fuzzy Expert Model-Based Soft Fault Diagnosis for Two Tank Interacting System,” *Arch. Control Sci.*, vol. 24, no. 3, Jan. 2014.
  - [24] H. A. a Talebi, K. Khorasani, and S. Tafazoli, “A recurrent neural-network-based sensor and actuator fault detection and isolation for nonlinear systems with application to the satellite’s attitude control subsystem,” *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 45–60, Jan. 2009.
  - [25] I. Samy, I. Postlethwaite, and D.-W. Gu, “Sensor Fault Detection and Accommodation Using Neural Networks with Application to A Non-linear Unmanned Air Vehicle Model,” *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, vol. 224, no. 4, pp. 437–447, Jan. 2010.
  - [26] Z. Liu and H. He, “Model-based Sensor Fault Diagnosis of a Lithium-ion Battery in Electric Vehicles,” *Energies*, vol. 8, no. 7, pp. 6509–6527, Jun. 2015.
  - [27] W. Yin, W. Zhang, and X. Sun, “A SVM- Based Multiple Faults Classification Scheme Design in Flight Control FDI System,” in *Second International Conference on Innovative Computing, Informatio and Control (ICICIC 2007)*, 2007, pp. 187–187.
  - [28] R. T. Rausch, K. F. Goebel, N. H. Eklund, and B. J. Brunell, “Integrated in-Flight Fault Detection and Accommodation: A Model-Based Study,” *J. Eng. Gas Turbines Power*, vol. 129, no. 4, p. 962, 2007.
  - [29] L. Ren, W. Lv, S. Jiang, and Y. Xiao, “Fault Diagnosis Using a Joint Model Based on Sparse Representation and SVM,” *IEEE Trans. Instrum. Meas.*, vol. 65, no. 10, pp. 2313–2320, Oct. 2016.
  - [30] R. Fujimaki, T. Yairi, and K. Machida, “An Approach to Spacecraft Anomaly Detection Problem Using Kernel Feature Space,” in *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD ’05*, 2005, p. 401.
  - [31] S. Borguet and O. Léonard, “Coupling Principal Component Analysis and Kalman Filtering Algorithms for On-line Aircraft Engine Diagnostics,” *Control Eng. Pract.*, vol. 17, no. 4, pp. 494–502, Apr. 2009.
  - [32] I. Trendafilova, M. P. Cartmell, and W. Ostachowicz, “Vibration-based Damage Detection in an Aircraft Wing Scaled Model Using Principal Component Analysis and Pattern Recognition,” *J. Sound Vib.*, vol. 313, no. 3–5, pp. 560–566, Jun. 2008.



- 
- [33] H. Yu, F. Khan, and V. Garaniya, "A Sparse PCA for Nonlinear Fault Diagnosis and Robust Feature Discovery of Industrial Processes," *AIChE J.*, vol. 62, no. 5, pp. 1494–1513, May 2016.
- [34] T. Jiang, K. Khorasani, and S. Tafazoli, "Parameter Estimation-Based Fault Detection, Isolation and Recovery for Nonlinear Satellite Models," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 4, pp. 799–808, Jul. 2008.
- [35] H. Ye, W. Wang, and S. Zhai, "Fault Diagnosis Based on Parameter Estimation in Closed-Loop Systems," *IET Control Theory Appl.*, vol. 9, no. 7, pp. 1146–1153, Apr. 2015.
- [36] B. Pourbabaei, N. Meskin, and K. Khorasani, "Sensor Fault Detection, Isolation, and Identification Using Multiple-Model-Based Hybrid Kalman Filter for Gas Turbine Engines," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1184–1200, Jul. 2016.
- [37] A. Sidhu, A. Izadian, and S. Anwar, "Adaptive Nonlinear Model-Based Fault Diagnosis of Li-Ion Batteries," *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1002–1011, Feb. 2015.
- [38] S. Nadarajan, S. K. Panda, B. Bhangu, and A. K. Gupta, "Online Model-Based Condition Monitoring for Brushless Wound-Field Synchronous Generator to Detect and Diagnose Stator Windings Turn-to-Turn Shorts Using Extended Kalman Filter," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3228–3241, May 2016.
- [39] K. Xiong *et al.*, "Detection of satellite attitude sensor faults using the UKF," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 2, pp. 480–491, Apr. 2007.
- [40] H. S. Kim, S. K. Park, Y. Kim, and C. G. Park, "Hybrid Fault Detection and Isolation Method for UAV Inertial Sensor Redundancy Management System," *IFAC Proc. Vol.*, vol. 38, no. 1, pp. 265–270, 2005.
- [41] A. Rahimi, K. D. Kumar, and H. Alighanbari, "Enhanced Adaptive Unscented Kalman Filter for Reaction Wheels," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 2, pp. 1568–1575, Apr. 2015.
- [42] H. Liu, D. Liu, C. Lu, and X. Wang, "Fault Diagnosis of Hydraulic Servo System Using the Unscented Kalman Filter," *Asian J. Control*, vol. 16, no. 6, pp. 1713–1725, Nov. 2014.
- [43] P. V. S. Nag, G. K. Silla, V. H. V. Gummadi, C. B. Harishankar, V. K. Ray, and C. S. Kumar, "Model Based Fault Diagnosis of Low Earth Orbiting (LEO) Satellite using Spherical Unscented Kalman Filter," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 635–638, 2016.
- [44] I. Arasaratnam and S. Haykin, "Cubature Kalman Filters," *IEEE Trans. Automat. Contr.*, vol. 54, no. 6, pp. 1254–1269, Jun. 2009.
- [45] K. Tidriri, N. Chatti, S. Verron, and T. Tiplica, "Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges," *Annu. Rev. Control*, vol. 42, pp. 63–81, 2016.
- [46] M. Vale, R. Martins, and A. Maitelli, "Hybrid Method for Fault Detection and Identification Based on State Observers and Decision Trees," *J. Eur. des Systèmes Autom.*, vol. 48, no. 1–3, pp. 185–210, May 2014.
- [47] J. D. Boskovic, S. M. Li, and R. K. Mehra, "Intelligent control of spacecraft in the presence of actuator failures," in *38th IEEE Conference on Decision and Control*, 1999, vol. 5, pp.

- 4472–4477.
- [48] N. Tudoroiu, K. Khorasani, and K. K. N. Tudoroiu, “Satellite fault diagnosis using a bank of interacting Kalman filters,” *Aerosp. Electron. Syst. IEEE Trans.*, vol. 43, no. 4, pp. 1334–1350, Oct. 2007.
  - [49] H. Azarnoush and K. Khorasani, “Fault detection in spacecraft attitude control system,” in *IEEE International Conference on Systems, Man and Cybernetics*, 2007, pp. 726–733.
  - [50] W. Chen and M. Saif, “Observer-Based Fault Diagnosis of Satellite Systems Subject to Time-Varying Thruster Faults,” *J. Dyn. Syst. Meas. Control*, vol. 129, no. 3, pp. 352–356, May 2007.
  - [51] L. M. Z. Q. Li, K. Khorasani, Z. Q. Li, and L. Ma, “A Dynamic Neural Network-based Reaction Wheel Fault Diagnosis for Satellites,” in *2006 International Joint Conference on Neural Networks*, 2006, pp. 3714–3721.
  - [52] E. Sobhani-Tehrani and K. Khorasani, *Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach*, vol. 383. Springer, 2009.
  - [53] V. Venkatasubramanian and R. Vaidyanathan, “Diagnosing noisy process data using neural networks,” *IFAC/IMACS Symp. Fault Detect.*, no. 6, pp. 547–552, 1991.
  - [54] J. Wong, K. McDonald, and A. Palazoglu, “Classification of process trends based on fuzzified symbolic representation and hidden Markov models,” *J. Process Control*, vol. 8, no. 5, pp. 395–408, 1998.
  - [55] J. B. MacQueen, “Some methods for classification and analysis of multivariate observations,” in *Berkeley Symposium on Mathematical Statistics and Probability*, 1967, vol. 1, pp. 281–297.
  - [56] V. Venkatasubramanian, R. Rengaswamy, K. Yin, S. N. Kavuri, and K. Yin, “A review of process fault detection and diagnosis,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 293–311, Mar. 2003.
  - [57] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, “A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 313–326, 2003.
  - [58] S. Mousavi and K. Khorasani, “Fault detection of reaction wheels in attitude control subsystem of formation flying satellites: A dynamic neural network-based approach,” *Int. J. Intell. Unmanned Syst.*, vol. 2, no. 1, pp. 2–26, 2014.
  - [59] A. Barua and K. Khorasani, “Hierarchical Fault Diagnosis and Fuzzy Rule-Based Reasoning for Satellites Formation Flight,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 4, pp. 2435–2456, 2011.
  - [60] Y.-C. Choi, J.-H. Son, and H.-S. Ahn, “Fault detection and isolation for a small CMG-based satellite: A fuzzy Q-learning approach,” *Aerosp. Sci. Technol.*, vol. 47, pp. 340–355, Dec. 2015.
  - [61] R. Wang, X. Gong, M. Xu, and Y. Li, “Fault detection of flywheel system based on clustering and principal component analysis,” *Chinese J. Aeronaut.*, vol. 28, no. 6, pp. 1676–1688, 2015.
  - [62] D. Hu, A. Sarosh, and Y.-F. F. Dong, “A novel KFCM based fault diagnosis method for

- unknown faults in satellite reaction wheels,” *ISA Trans.*, vol. 51, no. 2, pp. 309–316, Mar. 2012.
- [63] A. Barua and K. Khorasani, “Verification and Validation of Hierarchical Fault Diagnosis in Satellites Formation Flight,” *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.)*, vol. 42, no. 6, pp. 1384–1399, Nov. 2012.
- [64] A. Joshi, V. Gavriloiu, A. Barua, A. Garabedian, P. Sinha, and K. Khorasani, “Intelligent and learning-based approaches for health monitoring and fault diagnosis of RADARSAT-1 attitude control system,” in *2007 IEEE International Conference on Systems, Man and Cybernetics*, 2007, pp. 3177–3183.
- [65] X. Li, Q. Zhang, and H. Su, “An adaptive observer for joint estimation of states and parameters in both state and output equations,” *Int. J. Adapt. Control Signal Process.*, vol. 25, no. 9, pp. 831–842, Sep. 2011.
- [66] R. Zhan and J. Wan, “Iterated Unscented Kalman Filter for Passive Target Tracking,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 3, pp. 1155–1163, Jul. 2007.
- [67] F. Campillo and V. Rossi, “Convolution Particle Filter for Parameter Estimation in General State-Space Models,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 1063–1072, Jul. 2009.
- [68] S. Yin and X. Zhu, “Intelligent Particle Filter and Its Application on Fault Detection of Nonlinear System,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 1–1, 2015.
- [69] N. Meskin and K. Khorasani, “Fault Detection and Isolation of discrete-time Markovian jump linear systems with application to a network of multi-agent systems having imperfect communication channels,” *Automatica*, vol. 45, no. 9, pp. 2032–2040, Sep. 2009.
- [70] N. B. Hoang and H. J. Kang, “A model-based fault diagnosis scheme for wheeled mobile robots,” *Int. J. Control. Autom. Syst.*, vol. 12, no. 3, pp. 637–651, 2014.
- [71] K. Kolcio and L. Fesq, “Model-based off-nominal state isolation and detection system for autonomous fault management,” *IEEE Aerosp. Conf. Proc.*, vol. 2016–June, 2016.
- [72] A. Rosich, R. Sarrate, and F. Nejjari, “On-line model-based fault detection and isolation for PEM fuel cell stack systems,” *Appl. Math. Model.*, vol. 38, no. 11–12, pp. 2744–2757, 2014.
- [73] Q. N. Xu, K. M. Lee, H. Zhou, and H. Y. Yang, “Model-Based Fault Detection and Isolation Scheme for a Rudder Servo System,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2384–2396, 2015.
- [74] Y. Cheng, R. Wang, and M. Xu, “A Combined Model-Based and Intelligent Method for Small Fault Detection and Isolation of Actuators,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2403–2413, 2016.
- [75] Tao Jiang and K. Khorasani, “A fault detection, isolation and reconstruction strategy for a satellite’s attitude control subsystem with redundant reaction wheels,” in *2007 IEEE International Conference on Systems, Man and Cybernetics*, 2007, pp. 3146–3152.
- [76] B. Bialke, “High Fidelity Mathematical Modeling of Reaction Wheel Performance,” in *1998 Annual AAS Rocky Mountain Guidance and Control Conference, Advances in the Astronautical Sciences*, 1998, pp. 483–496.

- 
- [77] N. Meskin and K. Khorasani, "Fault detection and isolation in a redundant reaction wheels configuration of a satellite," *Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern.*, no. 1, pp. 3153–3158, 2007.
  - [78] N. Meskin, E. Naderi, and K. Khorasani, "A Multiple Model-Based Approach for Fault Diagnosis of Jet Engines," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 1, pp. 254–262, Jan. 2013.
  - [79] S. M. Azizi and K. Khorasani, "A distributed Kalman filter for actuator fault estimation of deep space formation flying satellites," in *2009 3rd Annual IEEE Systems Conference*, 2009, pp. 354–359.
  - [80] N. Meskin and K. Khorasani, "Fault Detection and Isolation of Actuator Faults in Spacecraft Formation Flight," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 1159–1164.
  - [81] S. M. Azizi and K. Khorasani, "A Hierarchical Architecture for Cooperative Actuator Fault Estimation and Accommodation of Formation Flying Satellites in Deep Space," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 2, pp. 1428–1450, 2012.
  - [82] S. Ghasemi and K. Khorasani, "Fault detection and isolation of the attitude control subsystem of spacecraft formation flying using extended Kalman filters," *Int. J. Control*, vol. 88, no. 10, pp. 2154–2179, Oct. 2015.
  - [83] I. Shames, A. M. H. Teixeira, H. Sandberg, and K. H. Johansson, "Distributed fault detection for interconnected second-order systems," *Automatica*, vol. 47, no. 12, pp. 2757–2764, Dec. 2011.
  - [84] K. Zhang, G. Liu, and B. Jiang, "Robust Unknown Input Observer-Based Fault Estimation of Leader–Follower Linear Multi-agent Systems," *Circuits, Syst. Signal Process.*, vol. 36, no. 2, pp. 525–542, Feb. 2017.
  - [85] M. Chadli, M. Davoodi, and N. Meskin, "Distributed state estimation, fault detection and isolation filter design for heterogeneous multi-agent linear parameter-varying systems," *IET Control Theory Appl.*, vol. 11, no. 2, pp. 254–262, Jan. 2017.
  - [86] D. An, N. H. Kim, and J. H. Choi, "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews," *Reliab. Eng. Syst. Saf.*, vol. 133, pp. 223–236, 2015.
  - [87] L. Liao and F. Kottig, "Review of Hybrid Prognostics Approaches for Remaining Useful Life Prediction of Engineered Systems, and an Application to Battery Life Prediction," *IEEE Trans. Reliab.*, vol. 63, no. 1, pp. 191–207, Mar. 2014.
  - [88] L. Liao and F. Köttig, "A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction," *Appl. Soft Comput.*, vol. 44, pp. 191–199, Jul. 2016.
  - [89] B. Bole, C. S. Kulkarni, and M. Daigle, "Adaptation of an Electrochemistry-based Li-Ion Battery Model to Account for Deterioration Observed Under Randomized Use," *Annu. Conf. Progn. Heal. Manag. Soc.*, pp. 1–9, 2014.
  - [90] L. Datong, P. Jingyue, Z. Jianbao, and P. Yu, "Data-driven prognostics for lithium-ion battery based on Gaussian Process Regression," *Progn. Syst. Heal. Manag. (PHM)*, 2012

- IEEE Conf.*, pp. 1–5, 2012.
- [91] D. Liu, Y. Luo, Y. Peng, X. Peng, and M. Pecht, “Lithium-ion Battery Remaining Useful Life Estimation Based on Nonlinear AR Model Combined with Degradation Feature,” *Annu. Conf. Progn. Heal. Manag. Soc. 2012*, pp. 1–7, 2012.
  - [92] B. E. Olivares, M. A. Cerda Munoz, M. E. Orchard, and J. F. Silva, “Particle-Filtering-Based Prognosis Framework for Energy Storage Devices With a Statistical Characterization of State-of-Health Regeneration Phenomena,” *IEEE Trans. Instrum. Meas.*, vol. 62, no. 2, pp. 364–376, Feb. 2013.
  - [93] B. Saha, K. Goebel, and J. Christophersen, “Comparison of prognostic algorithms for estimating remaining useful life of batteries,” *Trans. Inst. Meas. Control*, vol. 31, no. 3–4, pp. 293–308, 2009.
  - [94] B. Saha and K. Goebel, “Modeling Li-ion battery capacity depletion in a particle filtering framework,” *Proc. Annu. Conf. Progn. Heal. Manag. Soc.*, pp. 2909–2924, 2009.
  - [95] J. Zhou, D. Liu, Y. Peng, and X. Peng, “Dynamic battery remaining useful life estimation: An on-line data-driven approach,” *2012 IEEE I2MTC - Int. Instrum. Meas. Technol. Conf. Proc.*, pp. 2196–2199, 2012.
  - [96] T. Van Tung, B.-S. Yang, V. T. Tran, and B.-S. Yang, “Machine Fault Diagnosis and Prognosis: The State of The Art,” *Int. J. Fluid Mach. Syst.*, vol. 2, no. 1, pp. 61–71, Mar. 2009.
  - [97] C. Ciandrini, M. Gallieri, A. Giantomassi, G. Ippoliti, and S. Longhi, “Fault detection and prognosis methods for a monitoring system of rotating electrical machines,” in *2010 IEEE International Symposium on Industrial Electronics*, 2010, pp. 2085–2090.
  - [98] G. Jin, Q. Liu, J. Zhou, and Z. Zhou, “RePofe: Reliability physics of failure estimation based on stochastic performance degradation for the momentum wheel,” *Eng. Fail. Anal.*, vol. 22, pp. 50–63, 2012.
  - [99] M. Rocchi *et al.*, “Fault prognosis for rotating electrical machines monitoring using recursive least square,” *EDERC 2014 - Proc. 6th Eur. Embed. Des. Educ. Res. Conf.*, pp. 269–273, 2014.
  - [100] C. Lu, Z. Y. Wang, W. L. Qin, and J. Ma, “Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification,” *Signal Processing*, vol. 130, pp. 377–388, 2017.
  - [101] W. Abed, S. Sharma, R. Sutton, and A. Motwani, “A Robust Bearing Fault Detection and Diagnosis Technique for Brushless DC Motors Under Non-stationary Operating Conditions,” *J. Control. Autom. Electr. Syst.*, vol. 26, no. 3, pp. 241–254, 2015.
  - [102] W. Ahmad, S. Ali Khan, and J.-M. Kim, “Estimating the remaining useful life of bearings using a neuro-local linear estimator-based method,” *J. Acoust. Soc. Am.*, vol. 141, no. 5, p. EL452-EL457, 2017.
  - [103] W. Jingfu, H. Zheng, Q. Li, H. Wu, and B. Zhou, “Prognostic for On-orbit Satellite Momentum Wheel Based on the Similitude Method,” 2015.
  - [104] E. Zio and F. Di Maio, “A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system,” *Reliab. Eng. Syst. Saf.*, vol. 95, no.

- 1, pp. 49–57, 2010.
- [105] T. Wang, J. Yu, D. Siegel, and J. Lee, “A similarity-based prognostics approach for remaining useful life estimation of engineered systems,” *2008 Int. Conf. Progn. Heal. Manag. PHM 2008*, no. November, 2008.
- [106] H. Li, G. Jin, J. Zhou, and B. Zhou, “Bayesian Network Reliability Modeling and Assessment of Momentum Wheel on Orbit,” *Acta Aeronaut. Astronaut. Sin.*, vol. 30, no. 6, pp. 1084–1089, 2009.
- [107] X. Hao, “Life prediction of satellite key components based on RVM-PF,” National Defense University of Science and Technology, 2011.
- [108] Y. W. Fan, X. L. Yang, C. W. Li, and T. Qing, “Experimental research on the momentum wheel failure physical model,” *Chinese J. Sp. Sci.*, vol. 29, no. 1, pp. 78–86, 2009.
- [109] S. S. Bisht and M. P. Singh, “An Adaptive Unscented Kalman Filter for Tracking Sudden Stiffness Changes,” *Mech. Syst. Signal Process.*, vol. 49, no. 1–2, pp. 181–195, 2014.
- [110] L. Losik and D. Ph, “Using the Prognostic Health Management Program on the Air Force Next Generation Reusable Launch Vehicle,” pp. 1–20, 2012.
- [111] “Research and Technology Goals and Objectives for Integrated Vehicle Health Management (IVHM),” Washington, DC, 1992.
- [112] M. Schwabacher, J. Samuels, and L. Brownston, “The NASA integrated vehicle health management technology experiment for X-37,” *Proc. SPIE - Int. Soc. Opt. Eng.*, vol. 4733, pp. 49–60, 2002.
- [113] P. Johnson, S.B., Gormley, T., Kessler, S., Mott, C., Patterson-Hine, A., Reichard, K. and Scandura Jr, *System health management: with aerospace applications*. John Wiley & Sons, 2011.
- [114] S. Li, M. Zhong, and Y. Zhao, “Estimation and compensation of unknown disturbance in three-axis gyro-stabilized camera mount,” *Trans. Inst. Meas. Control*, vol. 37, no. 6, pp. 732–745, Jul. 2015.
- [115] H. Gui, L. Jin, S. Xu, and J. Zhang, “On the attitude stabilization of a rigid spacecraft using two skew control moment gyros,” *Nonlinear Dyn.*, vol. 79, no. 3, pp. 2079–2097, Feb. 2015.
- [116] F. Zhang, L. Jin, and S. Xu, “Fault Tolerant Attitude Control for spacecraft with SGCMGs under actuator partial failure and actuator saturation,” *Acta Astronaut.*, vol. 132, no. September 2016, pp. 303–311, 2017.
- [117] A. Noumi, H. Misuzu, T. Kanzawa, and M. Takahashi, “Fault-Tolerant Attitude Control System for a Spacecraft with Control Moment Gyros Using Multi-Objective Optimization,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no. 1, pp. 40–48, 2015.
- [118] L. Jin and S. Xu, “Fault Tolerant Attitude Control for Small Satellites Using Single Gimbal Control Moment Gyros and Magnetic Torquers,” *J. Aerosp. Eng.*, vol. 28, no. 3, p. 4014079, May 2015.
- [119] Y. Yoshimura, “Optimal Fault-Tolerant Configurations of Control Moment Gyros,” *J. Guid. Control. Dyn.*, vol. 38, no. 12, pp. 2460–2467, Dec. 2015.
- [120] B. Wie, D. Bailey, and C. Heiberg, “Singularity robust steering logic for redundant single-

- gimbal control moment gyros,” *J. Guid. Control. Dyn.*, vol. 24, no. 5, pp. 865–872, 2001.
- [121] H. Chang, W. Jiao, Q. Fu, J. Xie, and W. Yuan, “Design and Simulation of a MEMS Control Moment Gyroscope for the Sub-Kilogram Spacecraft,” *Sensors*, vol. 10, no. 4, pp. 4130–4144, Apr. 2010.
- [122] S. Beck, J. Kim, and H. Oh, “Identification of Control Moment Gyro Faults on Noisy Sensor Data,” *Recent Adv. Syst. Signals, Control. Commun. Comput. Identif.*, no. 2, pp. 298–303.
- [123] H. Oh and J. Kim, “The Detection and Identification of Control Moment Gyro Faults,” in *Second IAA Conference on Dynamics and Control of Space Systems 2014, DyCoSS’2014*, 2015, pp. 1–10.
- [124] J. Li and K. D. Kumar, “Decentralized fuzzy fault tolerant control for multiple satellites attitude synchronization,” in *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, 2011, pp. 1836–1843.
- [125] D. Fragopoulos and M. Innocenti, “Stability considerations in quaternion attitude control using discontinuous Lyapunov functions,” *IEE Proc. - Control Theory Appl.*, vol. 151, no. 3, pp. 253–258, May 2004.
- [126] P. Tsiotras, H. Shen, and C. Hall, “Satellite Attitude Control and Power Tracking with Energy/Momentum Wheels,” *J. Guid. Control. Dyn.*, vol. 24, no. 1, pp. 23–34, Jan. 2001.
- [127] P. C. Hughes, *Spacecraft Attitude Dynamics*. Mineola, New York: Dover Publication Inc., 2004.
- [128] A. H. J. De Ruiter, C. J. Damaren, and J. R. Forbes, *Spacecraft Dynamics and Control An Introduction*, 1st ed. John Wiley & Sons Ltd., 2013.
- [129] H. Schaub and J. L. Junkins, “Analytical Mechanics of Space Systems,” *AIAA Educ. Ser.*, p. 11,34,37,112,113, 2003.
- [130] W. Cai, X. Liao, and D. Y. Song, “Indirect Robust Adaptive Fault -Tolerant Control for Attitude Tracking of Spacecraft,” *J. Guid. Control. Dyn.*, vol. 31, no. 5, pp. 1456–1463, Sep. 2008.
- [131] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, “A survey of spacecraft formation flying guidance and control. Part II: control,” *Proc. 2004 Am. Control Conf.*, vol. 4, no. Part 11, pp. 2976–2985, 2004.
- [132] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, “A Survey of Spacecraft Formation Flying Guidance and Control (Part I): Guidance,” *Proc. 2003 Am. Control Conf.*, no. Part I, pp. 1733–1739, 2003.
- [133] D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, “A Survey of Spacecraft Formation Flying Guidance and Control (part I): Guidance,” in *American Control Conference*, 2013, pp. 1733–1739.
- [134] Wei Ren and R. W. Beard, “A decentralized scheme for spacecraft formation flying via the virtual structure approach,” in *Proceedings of the 2003 American Control Conference, 2003.*, 2003, vol. 2, no. 1, pp. 1746–1751.
- [135] C. H. Bong Wie David Bailey, “Singularity Robust Steering Logic for Redundant Single-Gimbal Control Moment Gyros,” *J. Guid. Control*, vol. 24, no. 5, pp. 865–872, 2001.

- 
- [136] R. Van der Merwe and E. Wan, "The Square-Root Unscented Kalman Filter for State and Parameter-Estimation," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, 2001, vol. 6, pp. 3461–3464.
  - [137] J. H. Gove and D. Y. Hollinger, "Application of a Dual Unscented Kalman Filter for Simultaneous State and Parameter Estimation in Problems of Surface-Atmosphere Exchange," *J. Geophys. Res.*, vol. 111, no. D8, p. D08S07, 2006.
  - [138] J. Zhou, S. Knedlik, and O. Loffeld, "INS/GPS Tightly-Coupled Integration Using Adaptive Unscented Particle Filter," *J. Navig.*, vol. 63, no. 3, pp. 491–511, Jul. 2010.
  - [139] A. Doucet, N. Freitas, and N. Gordon, "An Introduction to Sequential Monte Carlo Methods," in *Sequential Monte Carlo Methods in Practice*, New York, NY: Springer New York, 2001, pp. 3–14.
  - [140] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comput.*, vol. 10, no. 3, pp. 197–208, 2000.
  - [141] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc. F Radar Signal Process.*, vol. 140, no. 2, p. 107, 1993.
  - [142] J. MacCormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999, pp. 572–578 vol.1.
  - [143] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved particle filter for nonlinear problems," *IEE Proc. - Radar, Sonar Navig.*, vol. 146, no. 1, p. 2, 1999.
  - [144] D. Crisan, P. Del Moral, and T. Lyons, "Discrete filtering using branching and interacting particle systems," *Markov Process. Relat. Fields*, vol. 5, no. 3, pp. 293–319, 1999.
  - [145] K. Kanazawa, D. Koller, and S. Russell, "Stochastic simulation algorithms for dynamic probabilistic networks," *Proc. Elev. Annu. Conf. Uncertain. AI*, pp. 346–351, 1995.
  - [146] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, 2002.
  - [147] N. Bergman, "Recursive Bayesian Estimation: Navigation and Tracking Applications," Linköping University, 1999.
  - [148] J. S. Liu and R. Chen, "Sequential Monte Carlo Methods for Dynamic Systems," *J. Am. Stat. Assoc.*, vol. 93, no. 443, pp. 1032–1044, Sep. 1998.
  - [149] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *J. Comput. Graph. Stat.*, vol. 5, no. 1, pp. 1–25, 1996.
  - [150] W. R. Gilks and C. Berzuini, "Following a moving target-Monte Carlo inference for dynamic Bayesian models," *J. R. Stat. Soc. Ser. B (Statistical Methodol.)*, vol. 63, no. 1, pp. 127–146, Feb. 2001.
  - [151] F. Martinerie and P. Forster, "Data association and tracking using hidden Markov models and dynamic programming," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1992, pp. 449–452 vol.2.
  - [152] S. Godsill and T. Clapp, "Improvement Strategies for Monte Carlo Particle Filters," in



- Sequential Monte Carlo Methods in Practice*, New York, NY: Springer New York, 2001, pp. 139–158.
- [153] N. Oudjane and C. Musso, “Progressive correction for regularized particle filters,” in *Proceedings of the Third International Conference on Information Fusion*, 2000, p. THB2/10-THB2/17 vol.2.
- [154] M. K. Pitt and N. Shephard, “Filtering via simulation: Auxiliary particle filters,” *J. Am. Stat. Assoc.*, vol. 94, no. 446, pp. 590–599, 1999.
- [155] C. Musso, N. Oudjane, and F. Gland, “Improving Regularised Particle Filters,” in *Sequential Monte Carlo Methods in Practice*, New York, NY: Springer New York, 2001, pp. 247–271.
- [156] S. V. Stehman, “Selecting and interpreting measures of thematic classification accuracy,” *Remote Sens. Environ.*, vol. 62, no. 1, pp. 77–89, Oct. 1997.
- [157] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.
- [158] E. W. Grafarend, *Linear and Nonlinear Models: Fixed Effects, Random Effects, and Mixed Models*. Berlin: Walter de Gruyter, 2006.
- [159] K. Xiong, H. Y. Zhang, and C. W. Chan, “Performance evaluation of UKF-based nonlinear filtering,” *Automatica*, vol. 42, no. 2, pp. 261–270, Feb. 2006.
- [160] K. Reif and R. Unbehauen, “The extended Kalman filter as an exponential observer for nonlinear systems,” *IEEE Trans. Signal Process.*, vol. 47, no. 8, pp. 2324–2328, 1999.
- [161] A. Farina, B. Ristic, and D. Benvenuti, “Tracking a ballistic target: comparison of several nonlinear filters,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 3, pp. 854–867, Jul. 2002.
- [162] M. B. Rhudy, “Sensitivity and Stability Analysis of Nonlinear Kalman Filters with Application to Aircraft Attitude Estimation,” West Virginia, 2013.
- [163] C. F. Van Loan, *Introduction to Scientific Computing: A Matrix-Vector Approach Using Matlab*, 2nd ed. Upper Saddle River, New Jersey: Prentice-Hall, 2000.
- [164] M. D. Griffin and J. R. French, *Space Vehicle Design*, 2nd ed. Blacksburg, Virginia, US: American Institute of Aeronautics and Astronautics, 2004.
- [165] M. M. Carrasco and A. L. da Silva, “Attitude Determination for Low Cost IMU and Processor Board using the Methods of TRIAD , Kalman Filter and Allan Variance,” *BRAZILIAN J. Sci. Initiat.*, vol. 3, no. 2, pp. 26–41, 2016.
- [166] M. Staroswiecki and N. E. Wu, Eds., *Fault Detection, Supervision and Safety of Technical Processes 2003 (SAFEPROCESS 2003): A Proceedings Volume from the 5th IFAC Symposium, Washington, D.C., USA, 9-11 June 2003, Volume 1*. Washington, DC: Elsevier, 2004.
- [167] V. Kumar and H. B. Hablani, “Autonomous Formation Keeping of Geostationary Satellites with Regional Navigation Satellites and Dynamics,” *J. Guid. Control. Dyn.*, vol. 40, no. 3, pp. 563–583, Mar. 2017.
- [168] H. B. Hablani, “Imaging Sensors Pointing and Tracking Controller Insensitive to Inertial Sensors Misalignments,” *J. Guid. Control. Dyn.*, vol. 31, no. 4, pp. 980–990, Jul. 2008.

- 
- [169] A. Rahimi, K. D. Kumar, and H. Alighanbari, "Fault estimation of satellite reaction wheels using covariance based adaptive unscented Kalman filter," *Acta Astronaut.*, vol. 134, pp. 159–169, May 2017.
- [170] C. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006.
- [171] E. Naderi, "Fault Diagnosis and Estimation of Dynamical Systems with Application to Gas Turbines," Concordia University, 2016.
- [172] K. E. Arkinson, *An Introduction to Numerical Analysis*, 2nd ed. New York, NY, USA: John Wiley & Sons, 1989.
- [173] V. Nguyen, J. Seshadrinath, D. Wang, S. Nadarajan, and V. Vaiyapuri, "Model-Based Diagnosis and RUL Estimation of Induction Machines Under Interturn Fault," *IEEE Trans. Ind. Appl.*, vol. 53, no. 3, pp. 2690–2701, May 2017.
- [174] V. Carrara and H. K. Kuga, "Estimating Friction Parameters in Reaction Wheels for Attitude Control," *Math. Probl. Eng.*, vol. 2013, pp. 1–8, 2013.
- [175] A. Y. Lee and E. K. Wang, "In-Flight Performance of Cassini Reaction Wheel Bearing Drag in 1997–2013," *J. Spacecr. Rockets*, vol. 52, no. 2, pp. 470–480, Mar. 2015.
- [176] J. M. Hacker, J. Ying, and P. C. Lai, "Reaction Wheel Friction Telemetry Data Processing Methodology and On-Orbit Experience," *J. Astronaut. Sci.*, vol. 62, no. 3, pp. 254–269, 2015.
- [177] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *J. Basic Eng.*, vol. 82, no. 1, p. 35, 1960.
- [178] M. Jouin, R. Gouriveau, D. Hissel, M.-C. Péra, and N. Zerhouni, "Particle filter-based prognostics: Review, discussion and perspectives," *Mech. Syst. Signal Process.*, vol. 72–73, pp. 2–31, May 2016.
- [179] D. An, J.-H. Choi, T. L. Schmitz, and N. H. Kim, "In situ monitoring and prediction of progressive joint wear using Bayesian statistics," *Wear*, vol. 270, no. 11–12, pp. 828–838, May 2011.
- [180] D. An, J.-H. Choi, and N. H. Kim, "Identification of correlated damage parameters under noise and bias using Bayesian inference," *Struct. Heal. Monit.*, vol. 11, no. 3, pp. 293–303, May 2012.
- [181] S. J. Payne, "A Bayesian approach for the estimation of model parameters from noisy data sets," *IEEE Signal Process. Lett.*, vol. 12, no. 8, pp. 553–556, Aug. 2005.
- [182] J. M. Hacker, J. Ying, and P. C. Lai, "Reaction Wheel Friction Telemetry Data Processing Methodology and On-Orbit Experience," *J. Astronaut. Sci.*, vol. 62, no. 3, pp. 254–269, Sep. 2015.
- [183] D. An, J. H. Choi, and N. H. Kim, "Prognostics 101: A tutorial for particle filter-based prognostics algorithm using Matlab," *Reliab. Eng. Syst. Saf.*, vol. 115, no. July, pp. 161–169, 2013.
- [184] D. Crisan and A. Doucet, "A survey of convergence results on particle filtering methods for practitioners," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 736–746, Mar. 2002.
- [185] R. A. Fisher, *Statistical methods for research workers*, Fifth., no. V. Edinburgh, UK: Oliver

- and Boyd, 1925.
- [186] J. Neyman, E. S. Pearson, and G. U. Yule, “The testing of statistical hypotheses in relation to probabilities a priori,” *Math. Proc. Cambridge Philos. Soc.*, vol. 29, no. 4, p. 492, Oct. 1933.