Development of a Computer Simulation Tool to Study the Attitude Determination and Control of CubeSats

by

Annalisa Wailand

Submitted in partial fulfilment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
December 2020

Table of Contents

List of Tables v

List of Figures vi

Abstract xi

List of Symbols Used xii

Chapter 1: Introduction 1

Chapter 2: Literature Review 4

- 2.1 Launched CubeSats 5
- 2.2 ISS Orbit 8
- 2.3 Attitude Determination 9
- 2.4 Attitude Determination Algorithms 11
  - 2.4.1 TRIAD and q-Method/QUEST 12
  - 2.4.2 Extended and Unscented Kalman Filter 12
- 2.5 Attitude Controllers 14
  - 2.5.1 B-dot Control 14
  - 2.5.2 Proportional-Derivative Control 15
- 2.6 Pointing Accuracy 16
- 2.7 Literature Review - Summary 18

Chapter 3: Simulator Development 20

- 3.1 Orbit Definition 25
  - 3.1.1 Two-Line Elements 25
- 3.2 Orbit Dynamics 31
  - 3.2.1 Simscape Coordinate Frame Definition 32
  - 3.2.2 Satellite Model 33
- 3.3 Space Environment 34
  - 3.3.1 Time and Date 34
  - 3.3.2 Geomagnetic Field 35
  - 3.3.3 Sun Reference Vector 35
  - 3.3.4 Eclipse 37
  - 3.3.5 Disturbance Torques 38

Chapter 4: Attitude Determination 51

- 4.1 Sun Sensors 51
  - 4.1.1 Sensor Placement 53

4.1.2 Sun Vector Determination 54
4.2 Gyroscope Model 55
4.3 Magnetometer Model 57
4.4 Attitude Determination - q-Method Algorithm 59
Chapter 5: Magnetic Detumbling Control 61
5.1.1 Magnetorquer Model 61
5.2 B-Dot Algorithm 68
5.2.1 Gain Tuning 82
5.3 B-Dot Algorithm with Derivative 84
5.4 B-dot Final Results 86
Chapter 6: Magnetic Pointing Control 91
6.1 PD Controller 91
6.1.1 Gain Tuning 93
6.2 Extended Kalman Filter – Magnetic Attitude Control 96
6.2.1 q-Method Problems and EKF Solutions 99
6.2.2 EKF Results 103
6.2.3 Problems with Magnetic PD Control 105
Chapter 7: Reaction Wheel Pointing Control 108
7.1 Reaction Wheel Design 108
7.2 Reaction Wheel PD Control 109
7.3 Extended Kalman Filter – Reaction Wheel Attitude Control 111
7.4 Gain Tuning 114
7.5 Threshold Solution 117
7.6 Wheel Performance Results 122
Chapter 8: Control Approach 125
8.1 Attitude Determination Results 126
8.2 PD Control Performance Results 129
Chapter 9: Conclusions and Recommendations 134
References 136
Appendix A: Coordinate Frame Definition 146
A.1 Coordinate Frames 146
A.1.1 Earth Frames 146
Appendix B: Coordinate Frame Transformations 151

B.1 Coordinate Transformations 151

- B.1.1 Principal Rotations 152
- B.1.2 Perifocal to ECI 153
- B.1.3 ECI to ECEF 155
- B.1.4 ECEF to LLA 157
- B.1.5 LLA to ECEF 158
- B.1.6 Latitude Conversion 158
- B.1.7 ECEF to NED 159
- B.1.8 Perifocal to Nadir-Pointing 161
- B.1.9 Nadir-Pointing to Body-Fixed 162

List of Tables

Table 1: Launched CubeSat Missions with Active Magnetic Actuation – Part 1 6
Table 2: Launched CubeSat Missions with Active Magnetic Actuation – Part 2 7
Table 3: Simulator Guide 23
Table 4: Two-Line Element Set [96] 25
Table 5: Two-Line Element Set Entry Definitions [96] [97, p. 60] 26
Table 6: Parameters for Satellite Geometries 34
Table 7: Sun Reference Vector Error Calculations 37
Table 8: Parameter Comparison for Aerodynamic Drag Torque Validation 44
Table 9: Rewired Magnetorquer Parameters 62
Table 10: Motor Driver Interface Table 64
Table 11: IMU Magnetometer Interface Table 64
Table 12: Reaction Wheel Design Parameters 109
Table 13: Statistics for the Three Worst-Performing Orbits 131

List of Figures

Figure 1: LORIS CubeSat CAD Rendering [1] 1

Figure 2: Histogram of Number of Sun Sensors Used in Surveyed Papers 10

Figure 3: Histogram of Determination Algorithms Used in Surveyed Papers 11

Figure 4: Simulator High-Level Subsystem Block Diagram 22

Figure 5: RAAN, Argument of Perigee, and Inclination in ECI 26

Figure 6: True Anomaly, Eccentric Anomaly, and Mean Anomaly for an Arbitrary Point in an Elliptical Orbit 30

Figure 7: “1U” Geometry 33

Figure 8: “2U” Geometry 33

Figure 9: “Final” Geometry [1] 33

Figure 10: Eclipse Region for a Near-Circular LEO Orbit 38

Figure 11: Inertially-Fixed Attitude for Disturbance Torque Validation 39

Figure 12: Nadir-Pointing Attitude for Disturbance Torque Validation 39

Figure 13: Gravity Gradient Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude 40

Figure 14: Gravity Gradient Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude 40

Figure 15: Aerodynamic Drag Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Inertially-Fixed Attitudeig 42

Figure 16: Aerodynamic Drag Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude 43

Figure 17: Nominal Projected Area (+X Projection) where $nA = [100]$ and $A =$ 0.02208 m2 45

Figure 18: Worst-Case Projected Area (Trimetric Projection) where $nA =$ 0.750.450.47 and $A = 0.05721$ m2 45

Figure 19: Solar Radiation Pressure Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Nadir-Pointing Attitude 47

Figure 20: Solar Radiation Pressure Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Nadir-Pointing Attitude 47

Figure 21: Residual Dipole Moment Disturbance Torque Magnitude Over One Orbit for 2U Prism Model 49

Figure 22: Residual Dipole Moment Disturbance Torque Components Over One Orbit for 2U Prism Model 49

Figure 23: Comparison between SFH 2430-Z Relative Spectral Sensitivity and $\cos\varphi$ versus Light Incident Angle Showing 0.26% Agreement [111] 52

Figure 24: Comparison between Experimentally Obtained Photodiode Noise 53

Figure 25: SolidWorks Sun Sensor Assembly [112] 54

Figure 26: Satellite Orientation Showing Effect of Shadowing 54

Figure 27: Comparison between Experimentally Obtained BNO055 Gyroscope Noise 56

Figure 28: Gyroscope Drift 57

Figure 29: Comparison between Experimentally Obtained MPU-9250 Magnetometer Noise 58

Figure 30: Comparison between Experimentally Obtained BNO055 Magnetometer Noise 59

Figure 31: Magnetorquer Rod Key Dimensions 61

Figure 32: Coil Winder Apparatus 62

Figure 33: Magnetorquer Experimental Arduino Circuit Diagram 64

Figure 34: Magnetometer Experimental Arduino Circuit Diagram 65

Figure 35: Individual Channel Voltages and Voltage Over Magnetorquer for Clockwise Motor Driver Command 66

Figure 36: Start Up Current Over Magnetorquer Compared with Exponential Fit 67

Figure 37: Shutdown Current Over Magnetorquer Compared with Exponential Fit 67

Figure 38: X and Y Component Comparison of B and B in the Body-Fixed Frame, Slow Angular Rate 70

Figure 39: X and Y Component Comparison of B and B in the Body-Fixed Frame, Fast Angular Rate 71

Figure 40: Constant Magnetic Field in ECI Z Direction 72

Figure 41: Angular Rate Vector for Second B-Dot Test Case 73

Figure 42: Body Angular Rates for $\omega ini = 0.11 - 0.120.13T$ rad/s, Field Along ECI Z Axis 74

Figure 43: Magnetic Dipole Moment and Control Torque for Figure 42 Case 74

Figure 44: Equatorial Orbit Ground Track [118] 76

Figure 45: ECI Magnetic Field Vector Directions Over One Equatorial Orbit 76

Figure 46: ECI Magnetic Field Strength Over One Equatorial Orbit 77

Figure 47: Body Angular Rates Over One Equatorial Orbit with B-dot Control 78

Figure 48: Magnetic Dipole Moment and Control Torques Over One Equatorial Orbit with B-dot Control 78

Figure 49: Inclined ISS Orbit Ground Track [118] 79

Figure 50: ECI Magnetic Field Vector Directions Over One Inclined ISS Orbit 80

Figure 51: ECI Magnetic Field Strength Over One Inclined ISS Orbit 80

Figure 52: Body Angular Rates Over One Inclined ISS Orbit with B-dot Control 81

Figure 53: Magnetic Dipole Moment and Control Torques Over One Inclined ISS Orbit with B-dot Control 82

Figure 54: 1U Geometry B-dot Settling Times for Different Gains K [119] 83

Figure 55: Comparison of B-dot Settling Times with Different Gains K for 1U and 2U Geometries [119] 84

Figure 56: Comparison of Magnetic Field Derivatives Over Portion of Two Orbits 86

Figure 57: New B-dot Settling Times for Different Gains K 87

Figure 58: Best Case B-Dot Gain Detumbling from 0.1 rad/s about All Axes 88

Figure 59: Best Case B-Dot Gain Detumbling from 0.35 rad/s about All Axes 88

Figure 60: Best Case B-Dot Gain Detumbling from $\boldsymbol{\omega} i = -0.120.20.005$ rad/s 89

Figure 61: Comparison of Both B-dot Controllers for the Final Satellite Geometry with Noisy Input and Disturbance Torques 90

Figure 62: Attainable (Actual) PD Control Torque versus Desired PD Control Torque 92

Figure 63: PD Settling Time for 1U CubeSat With Constant $Kp = 1 \times 10 - 9$ Nm and Different Values of Kd [119] 94

Figure 64: PD Gain Comparison for 1U and 2U CubeSat Geometries with Constant $Kd = 2 \times 10 - 5$ Nms/rad and Different Values of Kp [119] 95

Figure 65: Figure 66. PD settling time for 2U CubeSat with constant $Kp = 7 \times 10 - 9$ Nm and different values of Kd [119] 96

Figure 67: q-Method Issues – Availability of Two Distinct Measurement Input Vectors 100

Figure 68: q-Method Issues – Quaternion Duality 101

Figure 69: Quaternion Duality Logic Condition 102

Figure 70: q-Method Issues – Parallel Inputs 103

Figure 71: Comparison between Gyroscope Body Angular Rates and EKF Body Angular Rates Over One Orbit 104

Figure 72: Comparison between q-Method Attitude Quaternion and EKF Attitude Quaternion Over One Orbit 105

Figure 73: Magnetic PD Control Results with Best-Case Gains, Noisy Sensor Data, and No External Disturbances 106

Figure 74: Magnetic PD Control Results with Best-Case Gains, Noisy Sensor Data, and Worst-Case External Disturbances 107

Figure 75: Reaction Wheel 5 V Step Response 110

Figure 76: Reaction Wheel Torque from 5 V Step Response 110

Figure 77: Reaction Wheel Torque-Speed Curve 111

Figure 78: Reaction Wheel Configuration Assumed for Dynamic System Model with Reaction Wheel Dynamics 112

Figure 79: Actual Reaction Wheel Configuration for Dynamic System Model used in Simulation 112

Figure 80: PD Gain Analysis with Constant $Kp = 50/3$ V and Different Values of Kd – Settling from Stationary Initial Body Angular Rates and 71° Offset from Nadir 115

Figure 81: PD Gain Analysis with Constant $Kd = 90$ Vs/rad and Different Values of Kp – Settling from Stationary Initial Body Angular Rates and 71° Offset from Nadir 116

Figure 82: Rate and Quaternion Errors for Best Case Reaction Wheel PD Gains 116

Figure 83: EKF Quaternion State Prediction Versus Simscape Quaternion without No q-Method Attitude Available 118

Figure 84: EKF Quaternion State Prediction Versus Simscape Quaternion with Gap Attitude Availability Signal 119

Figure 85: Increasing Reaction Wheel Speed and Corresponding Worsening of Attitude Error 120

Figure 86: EKF Attitude Quaternion versus Simscape Output with Control Switching 121

Figure 87: Body Angular Rates with Control Switching 122

Figure 88: Reaction Wheel Speeds over 100 Orbits 123

Figure 89: Reaction Wheel Torques over 100 Orbits 124

Figure 90: Control Switch Logic Flowchart 125

Figure 91: EKF, q-Method, and “Actual” Quaternion Profile Comparison 127

Figure 92: Average Attitude Determination Error per Orbit between q-Method Quaternion Estimate and “Actual” Simscape Quaternion – 100 Orbit Duration 128

Figure 93: Average Attitude Determination Error per Orbit between EKF Quaternion Estimate and “Actual” Simscape Quaternion – 100 Orbit Duration 128

Figure 94: Viewing Time Per Orbit – 100 Orbit Duration 130

Figure 95: Percent Loss in Viewing Time Per Orbit – 100 Orbit Duration 130

Figure 96: Average Euler Angle Pointing Error Per Orbit for 100 Orbits with Standard Deviations Noted 132

Figure 97: ECI Reference Frame 147

Figure 98: ECEF Reference Frame 147

Figure 99: LLA Reference Frame Showing Geocentric Latitude λ, Geodetic Latitude μ, and Geodetic Altitude h (Longitude not Shown) 148

Figure 100: Perifocal Reference Frame 149

Figure 101: NP Reference Frame 149

Figure 102: BF Reference Frame 150

Figure 103: NED Reference Frame 151

Figure 104: Sequence of Coordinate Transformations 152

Figure 105: ECI to Perifocal Transformation for a Generic Inclined Elliptical Orbit 154

Figure 106: Perifocal to ECI Transformation Steps 155
Figure 107: Greenwich Sidereal Time Between ECI and ECEF 156
Figure 108: Geocentric and Geodetic Latitudes 159
Figure 109: ECEF and NED Coordinate Frames for an Inclined Orbit 160
Figure 110: Geodetic and Geocentric Latitude Difference for Case 4 – Inclined Orbit 160
Figure 111: Nadir-Pointing and Perifocal Coordinate Frames 161
Figure 112: Roll, Pitch, and Yaw Directions in Orbit Relative to Body-Fixed Axes 162

Abstract

This thesis documents the development and validation of a MATLAB/Simulink simulation tool for the Attitude Determination and Control System (ADCS) of the LORIS CubeSat in development at Dalhousie University for the Canadian CubeSat Project. The simulation tool comprises an approximation of the space environment characteristic of the International Space Station's orbit, a Simscape-based satellite dynamics model, two attitude controllers, models of the sensors onboard the LORIS CubeSat and the attitude determination algorithms that use them. Use of magnetic attitude control is assessed in simulation for both detumbling and pointing the satellite, while the latter forms the basis for the design of a reaction wheel-based controller capable of meeting LORIS mission requirements. For the conditions used in the simulations, the proposed ADCS control scheme exceeds mission accuracy requirements under many worst-case assumptions.

List of Symbols Used

Symbol	Unit	Description
$\mathcal{F}$	/	Coordinate frame
C_{ij}	/	Rotation matrix (i to j)
$R_{\oplus}$	[m]	Equatorial radius of Earth
$G_{\oplus}$	$[\mathrm{m}^3/(\mathrm{kg}\cdot\mathrm{s}^2)]$	Gravitational constant of Earth
$m_{\oplus}$	[kg]	Mass of Earth
$\mu_{\oplus}$	$[\mathrm{m}^3/s^2]$	Gravitational parameter of Earth
$\omega_{\oplus}$	[rad/s]	Angular velocity of Earth
f	/	Earth flattening factor (WGS-84)
ρ	$[\mathrm{kg}/\mathrm{m}^3]$	Atmospheric density
m	[kg]	Spacecraft total mass
J	$[\mathrm{kg}\cdot\mathrm{m}^2]$	Moment of inertia
h	[m]	Altitude
$\boldsymbol{r}$	[m]	Orbital radius vector
$\boldsymbol{v}$	[m/s]	Orbital velocity vector
$\boldsymbol{a}$	$[\mathrm{m}/\mathrm{s}^2]$	Orbital acceleration vector
$\boldsymbol{h}$	$[\mathrm{m}^2/\mathrm{s}]$	Orbital angular momentum vector
M	[°]	Mean anomaly
E	[°]	Eccentric anomaly
θ	[°]	True anomaly
a	[m]	Semimajor axis
p	[m]	Semilatus rectum
T	[s]	Orbital period
e	/	Eccentricity
i	[°]	Inclination
Ω	[°]	Right ascension of the ascending node
ω	[°]	Argument of perigee
u	[°]	Argument of latitude
μ	[°]	Geodetic latitude
λ	[°]	Geocentric latitude
l	[°]	Geodetic longitude
$\boldsymbol{B}$	[T]	Magnetic field strength
ϕ	[rad]	Euler X (roll)
θ	[rad]	Euler Y (pitch)
ψ	[rad]	Euler Z (yaw)
q	/	Quaternion
$J_{D(h,m,s)}$	[day]	Julian day number
T_{JD}	[century]	Time in Julian centuries
θ_G	[°]	Greenwich sidereal time

t_s	[sec]	Sample time
$\phi_{\odot}$	[deg]	Sun mean longitude
$M_{\odot}$	[deg]	Sun mean anomaly
$\phi_{ecliptic}$	[deg]	Ecliptic longitude
ϵ	[deg]	Obliquity
$\boldsymbol{S}$	[m]	Sun vector
T	[Nm]	Applied torque
C_D	/	Drag coefficient
$\boldsymbol{s}_{cp}$	[m]	Centre of pressure to centre of mass vector
c_p	[m]	Centre of pressure
c_g	[m]	Centre of gravity
q	/	Reflectance factor
A	$[\mathrm{m}^2]$	Spacecraft exposed surface area
c	m/s	Speed of light
l	rad	Angle of incidence of sunlight
F_s	$[\mathrm{W/m}^2]$	Solar constant
$\boldsymbol{D}$	$[\mathrm{Am}^2]$	Residual dipole moment
NP	/	Noise power
σ	/	Variance
$\boldsymbol{n}$	/	Sun sensor normal unit vector
$\boldsymbol{H}$	/	Sun sensor normal unit vector matrix
I	/	Sun sensor illumination
$\boldsymbol{y}$	/	Sun sensor illumination matrix
$\boldsymbol{s}$	/	Unit sun vector
I	[A]	Current
V	[V]	Voltage
R	$[\Omega]$	Resistance
m	$[\mathrm{Am}^2]$	Dipole moment
$\dot{\boldsymbol{B}}$	[T/s]	Magnetic field time derivative
K	$[\mathrm{A}^2\mathrm{s}^3\mathrm{m}^2/\mathrm{rad}\cdot\mathrm{kg}]$	B-dot control gain
K_p	[Nm]	PD proportional control gain (magnetic)
K_d	$[\mathrm{Nm}\cdot\mathrm{s/rad}]$	PD derivative control gain (magnetic)
K_p	[V]	PD proportional control gain (reaction wheel)
K_d	$[\mathrm{V}\cdot\mathrm{s/rad}]$	PD proportional control gain (reaction wheel)
$\boldsymbol{K}$	/	Kalman gain matrix
$\boldsymbol{Q}$	/	Kalman process noise covariance matrix
$\boldsymbol{R}$	/	Kalman measurement noise covariance matrix
K_{ss}	$[\mathrm{rad/V}\cdot\mathrm{s}]$	DC motor model steady-state gain
τ	[s]	DC motor model time constant

Chapter 1: Introduction

The Dalhousie University Low Earth Orbit Reconnaissance Imagery Satellite (LORIS) CubeSat is a satellite in development for the Canadian Space Agency's Canadian CubeSat Project, set to launch in 2021. This thesis describes the design and development of an Attitude Determination and Control System (ADCS) simulation tool for the LORIS project built in MATLAB/Simulink. Attitude is defined as the orientation of a satellite in orbit. The determination component of the ADCS includes the sensors used to estimate this orientation, while the control component of the ADCS includes the actuators and algorithms that orient the satellite and maintain it in its desired orientation throughout its mission life. The simulator aims to 1) propagate the LORIS orbit in a realistic environment with models of the external disturbances experienced by low-Earth orbiting (LEO) satellites, and 2) model the satellite attitude dynamics to aid in the design and validation of attitude determination and control algorithms for its mission.

CubeSats are sized in terms of units (U), where 1U is a cube with dimensions $10 \times 10 \times 10$ cm. The LORIS CubeSat is 2U – having fixed dimensions of $10 \times 10 \times 20$ cm – with four deployable solar panels extending outwards from the top. Its deployed configuration is depicted in the CAD rendering shown in Figure 1.

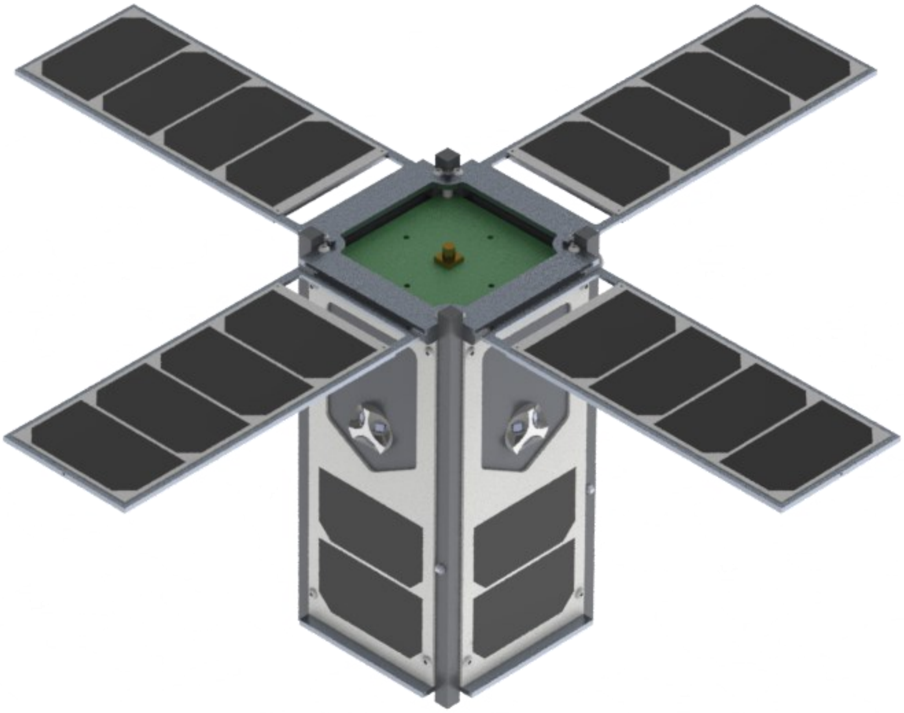


Figure 1: LORIS CubeSat CAD Rendering [1]

The satellite payload features two cameras on the Earth-pointing face of the satellite to photograph Earth's surface. Two controllers – one for detumbling and one for pointing – are designed and validated with respect to the LORIS mission requirements dictated by its camera payload. "Detumbling" refers to the operational phase wherein the high angular rates of a spinning satellite are reduced. "Pointing" refers to the nominal operational phase where the payload is active, and the satellite enters and maintains a stable orientation in orbit with its cameras facing Earth. Relevant mission requirements are reproduced below:

1. The ADCS should detumble the spacecraft in under 15 hours after the 30-minute waiting period post-deployment from the International Space Station (ISS). Launch requirements disallow the CubeSat to be powered on until 30 minutes have elapsed after its release from the ISS, whereinafter it is free to begin its detumbling phase as soon as enough power has been provided to the system.
2. The LORIS ADCS shall sample attitude determination telemetry (data from its onboard Sun, magnetic field, and angular rate sensors) at a rate of at least 1 Hz.
3. The ADCS shall have an attitude determination accuracy of at least $\pm 10°$. The estimate of the satellite's attitude calculated from its sensor measurements should be within $\pm 10°$ of the actual orientation of the satellite during its pointing phase.
4. The ADCS subsystem shall have an attitude pointing accuracy of at least $\pm 10°$. This accuracy ensures that the camera payload can photograph its intended target.
5. The LORIS ADCS shall implement a Proportional Derivative (PD) controller for nominal operations. "Nominal operations" means pointing at Earth during portions of the orbit where Sun is not eclipsed by Earth.
6. The ADCS shall operate on a supply voltage of 5 V and 3.3 V during nominal operation. This requirement drives the modelling of the ADCS actuators in simulation.

In the interest of keeping costs for the project and complexity of the system low, this thesis will first examine the feasibility of solely magnetic attitude control for both the detumbling and nominal (pointing) mission phases of LORIS. Whether or not magnetic pointing control is deemed a viable choice, magnetic actuation is an inevitable inclusion to the ADCS. Actuators such as reaction wheels typically have desaturation periods where magnetic detumbling control is assumed until the actuators are able to resume operation, and the detumbling phase of the satellite's mission necessitates magnetic detumbling control.

The objectives of this thesis are, therefore, to:

1. create a modular and realistic simulation tool for the LORIS CubeSat in which different ADCS approaches can be developed, validated, and assessed
2. explore the feasibility of magnetic detumbling and magnetic pointing control using the simulation tool
3. develop an ADCS solution to meet the LORIS mission requirements

To achieve these objectives, this thesis is organized as follows: Chapter 2 carries out a literature review that forms a basis for the assessment of the feasibility of developing a solely magnetic control scheme for the LORIS CubeSat using the proposed onboard suite of sensors including a three-axis gyroscope for angular rate measurement, magnetometers for measurement of the local magnetic field, and an arrangement of 18 photodiode Sun sensors on its outer surfaces. Chapter 3 describes the overall structure of the simulator and the flow of data between its elements and develops a model of the space environment that can predict the disturbance torques experienced by the satellite in its orbit. Chapter 4 covers attitude determination – the implementation of realistic sensor models and the acquisition of a satellite attitude estimate from their measurements. The analysis and assessment of magnetic attitude control begins in Chapter 5, detailing first the model of the magnetic actuators implemented in the simulation followed by the development, validation, and results of a magnetic detumbling controller. Chapter 6 follows with design, validation, and assessment of a magnetic pointing controller, outlining where it succeeds and where it fails. Chapter 7 then proposes an alternate control solution using actuation with reaction wheels, detailing the actuator design, modelling, and corresponding results. Chapter 8 outlines the proposed control approach that was implemented in the simulation tool for conditionally switching between the detumbling and pointing controllers. The final pointing results are then presented and compared against relevant mission requirements. Chapter 9 then draws conclusions and makes recommendations.

Chapter 2: Literature Review

The number of CubeSats being launched has dramatically increased from three per year in 2005 to about 288 per year in 2017] [2]. With about 1150 CubeSats launched to date, this chapter focuses on CubeSats described in the literature that exhibit solely magnetic active attitude control: actuation where controlling torques are provided solely by magnetic actuators (magnetorquer rods and/or coils) [3]. These actuators are desirable on the CubeSat scale for their relatively low cost, reliability, simplicity, and their significantly lower power requirement than that of a spinning wheel. Fully magnetic attitude control is defined as an underactuated system due to its inability to command control over all three axes instantaneously [4]. Control torques can only be generated in the plane normal to direction of the local magnetic field vector [5], [6]. This constraint necessitates a variation in the direction of Earth's magnetic field vector throughout the orbit to ensure torques can be generated along all three axes [7]. Although this constraint redefines the control problem as a time-variant periodic system with non-linear characteristics, the periodicity and scale of the problem allow for a linear approximation of its dynamics [8]. Problems therein arise from the weakness in control torques generated due in part to this directional restriction and a resultant slower reorientation than is achievable by other, more complex means [9].

CubeSat ADCS often contain a set of orthogonal reaction or momentum wheels and three orthogonal magnetorquers, the latter used exclusively for detumbling and momentum dumping. CubeSats employing reaction or momentum wheels are not considered within the scope of this review as the magnetorquers used in these CubeSats are not the primary active component of the satellite's nominal attitude control scheme. Types of fully active magnetically-actuated control schemes where the magnetorquers alone control the satellite's attitude include spin-stabilisation and three-axis stabilisation, both of which typically require a secondary controller to detumble the satellite after launch.

All surveyed satellites travel in a circular low-Earth orbit (LEO). Earth's magnetic field becomes weaker at higher altitudes, and thus LEO is optimal for magnetic control schemes where attitude depends on the behaviour of local magnetic field vectors throughout its mission. Focus is placed on surveying papers about small satellites only, ranging from 60 kg class small satellites to CubeSats and picosats. Larger satellites are not considered, though it is likely that magnetic control schemes that work on a larger scale could be adapted to and miniaturised for smaller satellites.

First, a tabulated summary of magnetically actuated CubeSats quantifies those named in the surveyed papers that have been successfully launched. Satellites and paper analyses for satellites that orbit in the same ISS orbit proposed for LORIS are then briefly explored, with an assessment of the common uses and constraints on magnetometers, Sun sensors, and gyroscopes following. Common attitude determination algorithms making use of measurements from these sensors are then analysed for theoretical and launched satellite cases, including an analysis of Kalman filtering techniques in the control context. The controller design section follows, focusing on proportional-derivative (PD) and B-dot control in line with the controller selections for LORIS. Lastly, conclusions about the most applicable ADCS components to the Dalhousie University CubeSat are then drawn.

2.1 Launched CubeSats

Table 1 (and its continuation in Table 2) on the following pages compile information found in the literature about launched CubeSat missions with fully active magnetic attitude control schemes onboard. These satellites are either spin-stabilised or three-axis stabilised – the former referring to establishing constant satellite rotation about a single fixed axis, the latter referring to pointing the satellite at a target according to mission payload objectives – and most utilise similar sensor information in their determination and control algorithms. Satellites that have not been launched, were cancelled, or that exist in theoretical name, such as the UYS-1 and AraMis-C1 CubeSats [4] [10] are not included in the table. Launch dates and some supplementary information in the table were taken from [3].

Table 1: Launched CubeSat Missions with Active Magnetic Actuation – Part 1

Satellite	Launch	Class	Sun Sensor	Gyroscope	Magnetometer	Magnetorquers	Ref.
AAU CubeSat	2003	1U	6	/	1 (3-axis)	3 coils	[11]
AAUSAT3	2013	1U	6	1 (2-axis) 1 (1-axis)	1 (3-axis)	3 coils	[12] [13]
AeroCube 6A/6B	2014	0.5U	3	1 MEMS	2	3 rods	[14] [15]
AntelSat	2014	2U	6	3 (1-axis)	1 (3-axis)	3 coils	[16] [17]
Chasqui I	2014	1U	Yes	Yes	1 (3-axis)	3	[18] [19]
CINEMA 1	2012	3U	2	/	1 (3-axis)	2 coils	[20]
COMPASS-1	2008	1U	5	/	1 (3-axis)	3 coils	[21]
CSTB-1	2007	1U	16	/	5 (2-axis)	3 coils	[22]
CXBN	2012	2U	1	2 (3-axis)	2 (3-axis)	3 coils	[23] [24]
DICE	2011	1.5U	1	/	1 (3-axis)	3 coils	[25]
Dove-2	2013	3U	Yes	Yes	Yes	Yes	[26]
ESTCube-1	2013	1U	6	4 (3-axis)	2 (3-axis)	3 coils	[27] [28]
GomX-1	2013	2U	Yes	Yes	Yes	3 coils	[29]
HIT-Sat	2006	1U	1	1 (1-axis)	1 (3-axis)	3 rods	[30]
MaSat-1	2012	1U	Infrared	(3-axis)	(3-axis)	2 electromagnets	[31]
MOVE-II	2018	1U	5	5 (3-axis)	5 (3-axis)	5 coils	[32]
OPUSAT	2014	1U	4	3	1 (3-axis)	2 coils	[33]
ParkinsonSat	2015	1.5U	Yes	/	Yes	3 coils	[34]
SamSat-QB50	2017	2U	Yes	1 (3-axis)	1 (3-axis)	3 coils	[35]
QBUS 1,2,4	2017	2U	Unknown	Unknown	Unknown	3 rods	[36]
UNSW-EC0	2017	2U	1	1 (3-axis)	1 (3-axis)	2 rods, 1 coil	[37]
VZLUSAT-1	2017	2U	/	/	1 (3-axis)	4 coils	[38]
RAIKO	2012	2U	6	/	1 (3-axis)	3 coils	[39]
SOMP	2013	1U	12	/	3 (1-axis)	3 coils	[40]
STUDSAT-1	2010	1U	/	3 (1-axis)	1 (3-axis)	3 coils	[41]
SwissCube	2009	1U	6	3 (1-axis)	1 (3-axis)	3 coils	[42]
TigriSat	2014	3U	Yes	1 (3-axis)	1 (3-axis)	3 coils	[43]
UKube-1	2014	3U	8	Yes	Yes	6 coils	[44] [45]
VELOX-PII	2013	1U	6	2 (3-axis)	2 (3-axis)	3 coils	[46] [47]

Table 2: Launched CubeSat Missions with Active Magnetic Actuation – Part 2

Satellite	Other ADCS	Controller (Active)	Pointing Error	Ref
AAU CubeSat	/	B-dot/Periodic constant gain LQR	8°	[11]
AAUSAT3	Permanent magnet	B-dot/Linearised state-space MPC	5°	[12] [13]
AeroCube 6A/6B	Earth sensor	Spin-stabilised	30°	[14] [15]
AntelSat	/	B-dot/LQR	10°	[16] [17]
Chasqui I	2 permanent magnets, 2 hysteresis foils	B-dot/continuous sliding mode	3°/20°	[18] [19]
CINEMA 1	/	B-dot/Spin stabilisation	10°	[20]
COMPASS-1	/	B-dot/Constant gain LQR	12°	[21]
CSTB-1	Unknown	Unknown	Unknown	[22]
CXBN	Pipper (star sensor)	B-dot/Spin-stabilised/PD	2°	[23] [24]
DICE	/	Spin-stabilised	5°	[25]
Dove-2	Hysteresis rods	B-dot	Unknown	[26]
ESTCube-1	/	Spin-stabilised	3°	[27] [28]
GomX-1	/	Unknown	10°	[29]
HIT-Sat	/	Spin-stabilised	10°	[30]
MaSat-1	Permanent magnets/hysteresis	Passive/Unknown	Unknown	[31]
MOVE-II	/	B-dot/LQR/Spin-stabilised	10°	[32]
OPUSAT	/	Spin-stabilised	10°	[33]
ParkinsonSat	/	Spin-stabilised	23°	[34]
SamSat-QB50	Hysteresis rods	Passive	20°	[35]
QBUS 1,2,4	Air drag panels	MPC LQR	20°	[36]
UNSW-EC0	Earth horizon CMOS	B-dot/LQR	20°	[37]
VZLUSAT-1	/	Custom, B-vector rotation-based	20°	[38]
RAIKO	/	Detumbling	N/A	[39]
SOMP	Permanent magnet	Detumbling	5°	[40]
STUDSAT-1	/	B-dot/Constant gain LQR	8°	[41]
SwissCube	/	B-dot	7.5°	[42]
TigriSat	/	Modified B-dot/proportional	Unknown	[43]
UKube-1	Permanent magnet	B-dot/ Sun tracking	5°	[44] [45]
VELOX-PII	/	Unknown (3-axis stabilised)	Unknown	[46] [47]

2.2 ISS Orbit

The LORIS CubeSat's orbit is to be identical to that of the International Space Station (ISS). The ISS orbit is near circular with an eccentricity on the order of 10^{-4} and an inclination of 51.6° relative to the equator [36]. Launched magnetically-actuated CubeSats from Table 1 and 2 that utilise this orbit include AAUSAT3 [13], Chasqui 1 [18], RAIKO [39], the three QBUS QB50 CubeSats [36] and UNSW-EC0 [37] – all of which employ linear detumbling and pointing control in an attempt to satisfy coarse pointing accuracy requirements of 10°-20°. Theoretical analyses and simulation cases wherein the ISS orbit is assumed include Sugimura *et al.* in [48], Santiago *et al.* in [49], Vedant and Ghosh in [50], Ivanov, *et al.* in [51], and Walker *et al.* in [52]. It is found that the ISS orbit provides less pointing accuracy and slower attitude convergence than a Sun-synchronous orbit, likely due to the extent of variability in local magnetic field direction between the two [48]. A Sun-synchronous orbit is a near-polar circular retrograde orbit with an inclination around 100°. The orbiting satellite experiences consistent illumination by Sun throughout its mission life and sees the same areas of Earth at the same time every day. Its high inclination subjects the satellite to significant directional variation in the local magnetic field vector which is a desirable feature for magnetic control.

Many papers – such as [6] – suggest the magnetic field is periodic with period $2\pi/n$, where n is the orbital rate. A higher orbit inclination implies a higher periodicity of the local magnetic field vector, and thus a higher degree of controllability [53]. An inclined orbit is necessary to avoid a constant-direction magnetic field vector (as experienced in equatorial and polar orbits). Inclination ensures the possibility of three-axis attitude control given the magnetorquers' inability to produce torques outside of the plane perpendicular to Earth's magnetic field vector.

Though Sun-synchronous and near-polar orbits are the more popular choice, ISS orbits are generally preferred for university-level satellite projects like Dalhousie University's LORIS CubeSat, likely given the simplicity of maintaining the launch orbit from the ISS without need for major correction and alteration. Changing a satellite's orbit require propellant and thrusters which are an additional expense that occupies significant space on the small satellite body.

2.3 Attitude Determination

This section briefly covers typical sensors and their applications onboard the satellites described in the surveyed papers, be it in simulation or in hardware development and flight. The sensors covered are magnetometers, Sun sensors, and gyroscopes. Having more than one sensor type is preferable to decrease attitude determination errors and to enable use of higher-fidelity attitude determination algorithms. Any attitude errors will propagate into control algorithms and degrade the pointing accuracy of the satellite.

Magnetometers measure the strength of the local magnetic field and provide an actual three-dimensional vector to compare against the expected or desired vector. An accurate onboard model of Earth's magnetic field is necessary for use as a reference in the most common attitude determination algorithms. The possibility of interference between the magnetometers and magnetorquers is high, and thus determination and control must be cycled. For example, the magnetometer onboard the launched AntelSat operates on a cycle in tandem with the magnetorquers; of the 500 ms controller cycle, the magnetometers run for 50 ms [16]. An 88% duty cycle was proposed for use onboard the AAUSAT3 [12]. In Ivanov *et al.*'s [54] simulation case, one second of the five second control phase is allotted to attitude determination, with magnetometers sampled at a rate of 1 Hz. The picosatellite simulated in [55] cycles the magnetometer operation at a 90% duty cycle, 1 second measuring for every 9 seconds of torqueing. With an ADCS cycle frequency of 3 Hz, the determination portion in Slavinskis *et al.*'s simulation takes only 0.013 seconds [27]. The LORIS CubeSat plans to use a sample rate of 1 Hz with a 90% duty cycle such that the magnetometers operate for 0.1s and the magnetorquers operate for 0.9s.

Sun sensors measure light intensity and are used to determine the position of Sun relative to the satellite body – ideal for use onboard CubeSats as a low-cost and well-documented choice for reliable attitude determination. At least three sensors must be illuminated for the satellite-to-Sun vector to be determined [16]. The Sun vectors measured are compared to theoretically-calculated reference Sun vectors (Earth to Sun in J2000), attainable within 0.01° accuracy [56]. The histogram in Figure 2 quantifies the number of Sun sensors onboard the satellites in the surveyed literature, with six sensors (one per each face) being the most common configuration. The legend of this and subsequent histograms distinguishes between "Launched" and "Simulated" CubeSats. "Launched" refers to those catalogued in Tables 1 and 2, while "Simulated" refers to theoretical simulations presented in the papers included in the scope of this chapter. Papers pertaining to satellites yet to be launched are categorised as "Simulated."

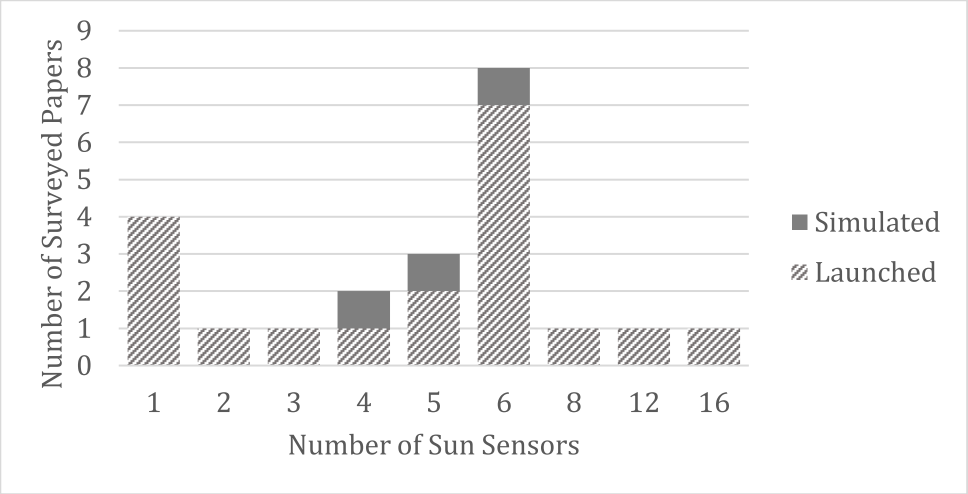


Figure 2: Histogram of Number of Sun Sensors Used in Surveyed Papers

Photodiodes arranged over all faces (occasionally excluding the Earth pointing face) of the launched satellites catalogued in Table 1 are present for the AAU CubeSat [11], AAUSAT3 [12], AntelSat [16], CSTB-1 [22], MOVE-II [32], OPUSAT [33], RAIKO [39], SOMP [40], UKube-1 [45], and VELOX-PII [46] missions. Sun sensors are assumed to be photodiodes unless otherwise stated, while "dual" or "two-axis" Sun sensors comprise two photodiodes arranged with a known angle between them or a fixed "mask" acting as an intensity threshold placed above them. Two photodiodes comprise the functional portion of the "dual Sun sensor" used onboard the launched CINEMA 1 [20], CXBN [24] and VELOX-PII [46] missions. In the literature, nonspecific Sun sensors are noted onboard the simulation cases presented in the work of Colagrossi *et al.* [57], Kinger *et al.* [58], Ovchinnikov *et al.* [59], Lovera and Silani [60], and for AraMiS-C1 in [10]. The simulated UYS-1 nanosatellite low-cost ADCS contains four "light transducers" affixed to its four body-length solar panels, creating a "line-of-sight" Sun sensor [4]. The launched ESTCube-1 contains six two-axis Sun sensors placed on each side of the satellite [27], while the UNSW-EC0 features a CMOS Sun sensor with a 150° field of view on one face of the satellite [61]. Microelectromechanical/ micro-optoelectromechanical systems (MEMS/MOEMS) Sun sensors are used onboard the launched AeroCube-6 [15], COMPASS-1 [21], DTUSat [62] and SwissCube [42] missions – a technology well-suited to the scale of a CubeSat. Photodiodes are sensitive to reflected light, termed "albedo", from other celestial bodies and the satellite itself. AAUSAT3 considers the effects of Earth's albedo as a reflected solar radiation disturbance and only considers the three faces of the satellite that see the highest current to determine the Sun vector [12]. In the ADCS for the launched AntelSat albedo is ignored as "direct solar illumination is dominant" [17]. The same is true for launched COMPASS-1 as per its associated thesis [21]. Given the potential for Sun sensors to be in the shade from its deployed solar arrays, the LORIS CubeSat plans to use 18 photodiodes – three per side facing different directions – along with a "mask" for each sensor to apply a lower bound on the measured intensity by limiting its field of view.

Gyroscopes (often contained within an inertial measurement unit (IMU) alongside a magnetometer and accelerometer) measure the angular rates of the satellite. For three-axis stabilised systems, the gyroscope may saturate post-launch and become unusable [63], or may be difficult to obtain data from in a sufficient resolution to satisfy mission pointing requirements [64]. As such, onboard models of the satellite's attitude dynamics are often necessary. Gyroscopes appear less frequently than magnetometers and Sun sensors amongst the satellites in Table 1 but do appear onboard both three-axis stabilised and spin-stabilised satellites. The LORIS CubeSat plans to use a three-axis gyroscope to measure the satellite's angular rates.

2.4 Attitude Determination Algorithms

Sensors that detect the attitude of the satellite in a useful form – quaternions or Euler angles – do not exist, thus satellite attitude must be calculated based on values read from the various sensors onboard. The use of three common and oft-utilised determination algorithms amongst the papers in the scope of this review are catalogued in this section. These algorithms include TRIAD, quaternion estimation (QUEST) and Kalman filtering including both Extended Kalman Filters (EKF) and Unscented Kalman Filters (UKF). As shown in Figure 3, the extended Kalman filter is the favoured method of attitude determination amongst the surveyed papers.

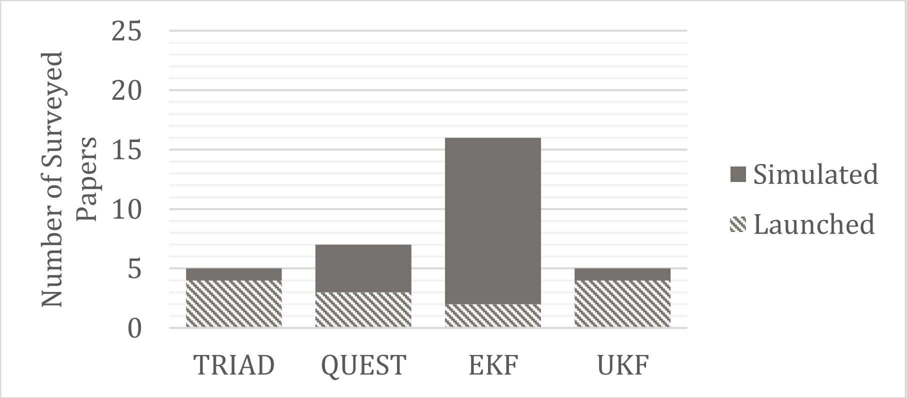


Figure 3: Histogram of Determination Algorithms Used in Surveyed Papers

2.4.1 TRIAD and q-Method/QUEST

The TRIAD algorithm for attitude determination estimates an attitude matrix between two frames of reference – one inertial and one fixed to the rotating satellite's principal inertial axes – from two different reference vectors in the former and their corresponding sensor measurements in the latter. Most commonly, a calculable standard Sun vector from Earth to Sun is used alongside the Sun vector from satellite to Sun obtained from Sun sensor measurements as the first vector pair. The second pair typically comprises a local magnetic field vector calculated according to the International Geomagnetic Reference Field (IGRF) or World Magnetic Model (WMM) geomagnetic field models alongside the body magnetic field vector measured by onboard magnetometers. The TRIAD algorithm is used with Sun sensor and magnetometer readings onboard the launched DICE spacecraft [25], Boeing's CSTB-1 [22], VELOX-PII [47], and exists as one of the options for the H-I-L simulation of the generic IlliniSat-2 bus [65]. The launched UNSW-EC0 CubeSat uses data from its Sun sensors, Earth sensor, and/or magnetometers in its algorithm [61].

The q-Method and quaternion estimator (QUEST) algorithms produce least-squares estimates of satellite attitude using the same vector concept as in the TRIAD algorithm with improved accuracy [66]. Typically, as with the TRIAD algorithm, QUEST estimates the attitude quaternion based on measured magnetic field and Sun vectors from magnetometers and Sun sensors in the satellite's body frame and modelled reference field and Sun position vectors in an inertial or orbital reference frame. In the literature, "q-Method" and "QUEST" are, on occasion, used interchangeably to refer to the method of quaternion attitude determination via eigen decomposition, though the papers included in this section all refer explicitly to their respective determination algorithms as "QUEST". QUEST is demonstrated in simulation by Kamal *et al.* in [67] and Kinger *et al.* in [58]. The launched COMPASS-1 [21] uses the QUEST algorithm with magnetometer and Sun sensor readings, and the same is developed (but not implemented) as a mechanism of attitude determination for the launched AntelSat and AAU CubeSat [17] [66].

2.4.2 Extended and Unscented Kalman Filter

Kalman filtering is a common means by which attitude and angular rate are estimated onboard satellites, utilising discrete measurements of sensor data, models of their expected values, and models of expected bias and noise for each. Its predictive functionality allows attitude to be predicted in the event of the loss of a sensor and can compensate for the worsening of onboard sensor noise and bias with time [67]. Lovera and Silani [60] note the necessity of a Kalman filter to attain unbiased state vector estimation when direct measurement is otherwise unavailable, as it is in the case of satellite attitude. Its most used forms are the nonlinear Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF). The EKF linearises a nonlinear system model with respect to current estimates of the system states [68]. The UKF is deterministic and models the system states as a probability distribution [69].

Searcy and Pernicka [64] develop an algorithm in simulation with two EKFs using only magnetometer measurements for spinning satellites capable of determining attitude within 1° of accuracy. Si Mohammed *et al.* discuss the design of an EKF combining magnetometer and Sun sensor readings with modelled values in [70] and compare the results with those of a UKF in [71], the former preferable for its computational simplicity. Ivanov *et al.* develop an EKF using a linearized angular rate dynamics model, quaternion attitude, and measurements of induced EMF in the magnetorquers [51]. These authors use the same Kalman filter for the small satellite simulation study in [54] and [72]. A similar EKF using only magnetometer data was implemented for the generic IllinSat-2 bus H-I-L simulation described in [65]. Tortora *et al.* [73] developed an EKF applicable to tumbling spacecraft, while Sugimura *et al.* [48] developed a determination algorithm using two EKFs. Pointing the launched UNSW-EC0 CubeSat involves an EKF with readings from all onboard attitude sensors listed in Table 1 [61], as does the launched StudSat-1 [41]. Habib presents a simulation case in [74] for the small EGYPTSAT-2 satellite wherein an EKF is developed based on magnetometer, GPS, and gyro measurements. A "standard" Kalman filter is used onboard the UKube-1 for gyroscope bias correction using readings from the satellite's Earth horizon sensor and magnetometers [45]. The launched AAU CubeSat uses an EKF with Sun sensor and magnetometer data as one of two determination methods onboard [66]. Lastly, Li *et al.* [63] present a "gyro-less" rate filtering algorithm with an EKF.

Launched AAUSAT3 implements a UKF with measured Sun vector, magnetic field vector, and angular rates in the body frame, and predicted eclipse status, control torque, Sun and magnetic field vectors in the standard Earth-centred-inertial reference frame [12]. A simulation case for the ESTCube-1 mission is studied in [27] wherein a UKF is used for both attitude and bias estimation. The UKF developed for the AntelSat uses Sun, magnetic field, and rate sensor readings and includes quaternion axial error among its state variables [17]. The launched GOMX-1 mission contains a similar UKF that uses Sun, magnetic field, and angular rate as inputs [75]. de Oliveira *et al.* [4] design an "unscented quaternion estimator" for the attitude determination component of the UYS-1 nanosatellite ADCS.

The LORIS CubeSat plans to use the q-Method algorithm analogous to the QUEST algorithm first proposed by Shuster in 1981 [76] in combination with a dynamic model integrated into an EKF based on Yang [77] to estimate the attitude and angular rates of the satellite.

2.5 Attitude Controllers

This review focuses on the use of B-dot detumbling control and proportional-derivative pointing control; however, other commonly used magnetically actuated controllers include the linear quadratic regulator (LQR) – another type of state feedback control – and model predictive control (MPC), which, as the name suggests, predicts "future" values from discrete measurements over a finite horizon [12].

2.5.1 B-dot Control

The B-dot algorithm is a proportional controller typically used during a satellite's post-launch detumbling phase and can be modified to act as a coarse nominal attitude controller. Its simplicity can cause instability and poor controllability in the presence of external disturbances, and it tends to best suit inclined orbits with significant variation in their magnetic field direction [78]. Typical B-dot controllers are used for coarse attitude and detumbling control onboard the launched AAU CubeSat [11], AAUSAT3 [12], AntelSat [16], ARC-1 [79], Chasqui 1 [19], CINEMA 1 [20], COMPASS-1 [21], CXBN [24], Dove-2 [26], DTUSat [62], MOVE-II [80], UNSW-EC0 [37], STUDSAT-1 [41], SwissCube [42], and UKube-1 [45] missions. The generic IlliniSat02 bus H-I-L simulation implements the typical algorithm as a control option [65].

The B-dot algorithm is often modified to better suit specific mission requirements. Ovchinnikov *et al.* [59] propose a B-dot control algorithm for a simulated axisymmetric satellite to reorient its axis of symmetry/spin axis and damp satellite nutation using only the magnetorquer along its spin axis. Roldugin and Testani [81] simulate the B-dot algorithm to detumble the satellite using the same nutation-damping modification (and reorientation controllers). The module for active debris removal proposed and simulated in [78] attaches itself via tether to an "uncooperative" target to detumble it via a modified B-dot control scheme eliminating the "speed term" in the magnetic field time derivative, stabilising faster than the unmodified law while consuming less power. The B-dot control scheme designed for the proposed nanosatellite IITMSAT by Kumar *et al.* [82] adds a magnetic moment offset component. The offset aligns the satellite with the local magnetic field along a chosen satellite axis as opposed to an arbitrary axis as in the unmodified algorithm [82]. In the low-cost attitude simulator proposed by Santiago and Velasco [49], the time derivative of the magnetic field is estimated from subsequent samples and their sampling interval, and the current required to power the magnetorquer is solved from the calculated magnetic dipole moment vector. Li *et al.* [63] propose an improved B-dot method unaffected by magnetometer measurement noise to work in tandem with an extended Kalman filter angular rate estimation scheme. Lastly, the launched TigriSat uses a modified B-dot controller that adds a proportional term to the dipole moment acting on the pointing error [43].

2.5.2 Proportional-Derivative Control

Proportional-derivative (PD) controllers are a common choice for attitude control wherein the control torque is calculated relative to attitude quaternion error and satellite angular rate error [48]. These errors are defined as the difference between the orbital reference frame (oriented in the satellite's desired pointing orientation) and the body reference frame fixed to the satellite. Controller gains can be designed in any number of ways, including pole-placement techniques or through trial and error. The same restrictions on torque apply as in the B-dot case: the achievable magnetic moment is determined from the pseudoinverse of the local magnetic field vector (or some variation thereof [48]) to constrain the torque normal to its direction. The references for the controller are a rate error of zero and a quaternion error of $q = [1 \quad 0 \quad 0 \quad 0]^T$ (unity scalar part) as demonstrated, for example, in the simulated design by Lovera and Astolfi [83]. Theoretical PD controllers operating under these conditions are proposed by Celani [84] who simplifies Lovera and Astolfi's design to comprise solely the attitude quaternion component, by Sugimura *et al.* [48] with a singularity robust (SR) inverse matrix for a nadir-pointing microsatellite, by Ivanov *et al.* [54] based on the authors' previous work in [72], by Si Mohammed *et al.* [71] with gains designed by selecting a desirable damping ratio and undamped natural frequency, and by Torcyznski *et al.* in [85]. Torcyznski *et al.* [85] report that the PD controller performs better in simulation than an LQR under the same conditions, settling 700 seconds faster within an equal degree of accuracy. In [6], Celani designs a PD control law based on his previous work in [84], where exponential stability of the control law is proven at length [6]. The case study for a near-polar circular orbit converges within five orbits – a common result with papers [84] [86], and [6]. Giri *et al.* [86] note in the design of their PD controller that issues may arise under high initial angular rate conditions with generating sufficient magnetic moments, though the periodicity of the magnetic field renders the system "controllable on average" [86]. The simulated magnetically-actuated tethered nanosatellite in descent studied by Kamal *et al.* in [67] is controlled by a simple PD controller with Euler angle and angular rate feedback [67]. Della Rossa, Lovera, and Dercole [87] propose an adaptive PD-like control law for an Earth pointing satellite. Lovera and Astolfi's previous work in [88] and [83] form the control law's basis and echo its design utilising averaging theory [87]. The launched CXBN CubeSat utilises a "simple" PD controller onboard for pointing [24]. Lastly, Inamori *et al.* [89] propose a PD controller to perform a series of sequential rotations to attain simulated three-axis control.

2.6 Pointing Accuracy

In this section, magnetic attitude control pointing accuracies are considered to hold only for simulation cases, as on-orbit accuracies could not be numerically verified for the majority of satellites successfully launched.

Among the spin-stabilised solely-magnetically-actuated CubeSats listed in Table 1, the ParkinsonSat mission requires an on-orbit accuracy of $\pm 23°$ in pointing its side panels towards Sun for a slow spin about its Z-axis [34]. Similarly, the Z-axis spin-stabilised AeroCube 6A/6B mission requires a pointing accuracy in the vicinity of 30° [14], having achieved an average steady-state error of 20° during its differential drag profile phase and 6° during its spin control phase in simulation [15]. A 20° pointing accuracy was specified as a mission requirement for the Chasqui 1 mission to maintain ground communication, though its failure to communicate post-launch means its success cannot be assessed [18]. The proposed SamSat-QB50 [35] and other launched QB50 CubeSats VZLUSat-1 [38], X-CubeSat [90], and QBUS satellites [36] are designed to meet a 20° pointing accuracy requirement for the QB50 mission, and mission requirement for the pointing accuracy of the QB50 UNSW QB50 EC0 CubeSat was set at 10°, with a determination accuracy of 2° [37]. Specific pointing accuracies achieved for the QB50 satellites post-launch are not noted. The best QB50 accuracy in simulation was obtained for the Aalto-2 at 15° [91].

Ivanov *et al.* [51] obtain a pointing accuracy of 13° for their simulated ADCS that uses electromotive force measurements, an EKF, and a PD control law. A mean pointing error of 12° was achieved in simulation in the design of the ADCS for the COMPASS-1 CubeSat, though its on-orbit success cannot be assessed due to bugs in its software resulting in an inability to estimate its attitude in orbit [21]. AntelSat obtains a pointing accuracy of under 10° with its B-dot control law and nominal linear PD-like control law in simulation only [16]. The simulation case studied by Ovchinnikov *et al.* [54] achieves a worst-case pointing accuracy of 10° with its PD controller and EKF, worsening in the presence of external disturbances [54]. Ovchinnikov *et al.* [72] also achieve a simulated pointing accuracy of 10° in the presence of modelled disturbances and measurement noise and a simulated pointing accuracy of 20° at low orbit inclinations for a PD controller. In line with typical picosat pointing requirements, Colagrossi and Lavagna [57] achieve a pointing accuracy of 10° to 15° for their simulation test-case 3U CubeSat. The genetically tuned LQR controller designed in [92] obtained a pointing accuracy in simulation within the 10° range, at the cost of a complex and time-consuming controller tuning process. Wiśniewski obtains a pointing accuracy in line with the required 10° in pitch and roll and 20° in yaw for the small Ørsted satellite infinite horizon LQR simulation case studied in [93]. In [94], Ørsted requirements are altered to be 10° about each axis, obtained via the PD control law and stability assessment simulations within. For the CubeSat simulation case in [52], a 10° pointing accuracy is achieved via the genetic control algorithm with fuzzy logic. GomX-1 [29], HIT-Sat [30], OPUSAT [33], and CINEMA 1 [20] CubeSats have pointing accuracy mission requirements of 10° specified, though their respective on-orbit results cannot be verified. For the MOVE-II in [80], it is noted that increasing the spin rate of the satellite can increase its pointing accuracy beyond the 10° obtained in simulation.

Accuracies obtained in the range of 2° – 8° were common amongst the surveyed papers in simulation only. The AAU CubeSat [11], STUDSAT-1 [41], and SwissCube [42] had pointing accuracy mission requirements of 8°, though it is not apparent if this requirement was satisfied for the missions. IITMSAT [80] is designed to meet a pointing requirement of 5° maintained for the majority of the orbit but drops to around 20° at the poles for the proposed control algorithm. The LQR and PD controllers simulated in Torczynski *et al.*'s analysis in [85] obtain maximum steady-state errors of 5.5° and 2° respectively, both in the range of the required 5° about each axis. The LQR controller designed and simulated in [95] meets its pointing requirement accuracy of 5° within 5 orbits. Accuracies of 5° were required of the SOMP [40], UKube-1 [44], DICE [25], and AAUSAT3 [12] missions, though results were not numerically reported. The simulation of UniSat-5 [81] (excluding the reaction wheels later added to the mission) obtains a pointing accuracy of 4° for its spin-stabilised magnetic attitude control system. Pointing accuracies of 3° in each axis were obtained in simulation for the low-cost ADCS system on the proposed UYS-1 nanosatellite [4]. The proposed alternate high-precision ADCS for the UYS-1 nanosatellite with a suite of reaction wheels and a star tracker could obtain accuracies better than 0.5° in simulation [4].

The most accurate of the relevant simulated cases include the simulation case considered by Sugimura *et al.* [48] where an orbit average pointing accuracy of 1.56° was obtained for the Sun-pointing orbit and of 2.64° for the ISS orbit. The finest accuracies catalogued in Table 2 were for the CXBN and ESTCube-1 missions. The pointing accuracy for the slowly spinning CXBN mission [24] was proposed to be around 2°, though numerical on-orbit results were not reported. Some difficulty was experienced in initially obtaining the desired photographs from the ESTCube-1 [28] (with an estimated pointing accuracy of 3°) due in part to the necessity of recalibrating the ADCS after launch, though eventually it achieved its mission objective. These papers suggest that accuracies better than 5° are theoretically achievable, more so for magnetic spin stabilisation, but sources of noise and robustness in the presence of high initial spin rates should be carefully considered in the ADCS design. Obtainable pointing accuracies for magnetic control vary greatly in simulation depending primarily on the magnitude of disturbances assumed to be acting upon the satellite.

2.7 Literature Review - Summary

This survey offers an in-depth look at the field of solely-magnetic attitude control for CubeSats, focusing on those with particular relevance to the proposed LORIS satellite ADCS design. Solely-magnetic three-axis stabilisation is yet to obtain high degrees of pointing accuracy validated in orbit. As shown in the simulated results of many of the papers, satellite pointing in its steady state adopts a slowly exponentially decreasing sinusoidal profile, indicating constant side-to-side motion of the satellite about all three axes. Some examples of this sinusoidal motion are in Sugimura *et al.* [48] with an approximate period of 1.5 oscillations/hour and one oscillation per orbit for both the PD control law and LQR control law in Torczynski *et al.*'s simulations [85].

Based on this review, magnetometers, gyroscopes, and Sun sensors, all of which are planned for use onboard LORIS, are the most commonly documented and utilised attitude determination sensors for CubeSats – preferable for their low cost and relative ease of implementation. Focus will be placed in this thesis on developing models for these sensors and incorporating them into a LORIS CubeSat simulation tool. An attitude determination system capable of estimating the LORIS CubeSat's orientation and angular rates using q-Method (analogous to the QUEST algorithm first proposed by Shuster in 1981 [76]) and extended Kalman filtering will then be incorporated into the simulator. The performance of

B-dot and magnetically-actuated PD control will initially be investigated using the simulator, after which the performance of reaction wheels will be investigated.

Therein lies the primary goal for the simulation tool designed in this thesis: the creation of a modular and realistic environment in which different attitude controllers and determination algorithms can be developed, validated, and assessed in regards to the LORIS mission requirements.

Chapter 3: Simulator Development

Appendices A and B contain the coordinate frame definitions and transformations between them upon which all subsequent simulator development and validation are based.

The LORIS simulation environment was built in MATLAB/Simulink, with satellite dynamics modelled using Simulink's Simscape multibody toolbox. As shown in Figure 4, the simulator is organised into "blocks" functionally equivalent to Simulink subsystems, sorted into five overarching categories: orbit, environment, determination, control, and satellite. Orbit blocks contain the algorithms that define the orbit geometrically and propagate the satellite's orbital motion in time. Environment blocks model the space environment and external disturbances experienced by the satellite throughout its mission. Determination and control blocks model the hardware and software components of the ADCS, producing control torques which are then carried to the satellite blocks containing the Simscape model of the satellite.

In reference to Figure 4, the "Orbit Propagator" block propagates the satellite's orbit from which the "Orbital Elements" that geometrically define the orbit and the satellite's position therein and the corresponding "Coordinate Transformations" from Appendix B are calculated. The location of the CubeSat within its orbit alongside readings from Earth's "Local Magnetic Field" model for a given "Time and Date" are used to determine the "LEO Disturbance Torques" the CubeSat experiences from the environment. The CubeSat's location and given "Time and Date" is also used to locate Sun via a "Sun Reference" vector and to determine if the CubeSat is transiting through "Eclipse". The "q-Method Attitude" block calculates the quaternion describing the orientation of the CubeSat based on noisy measurements from both the "Sun Sensor Subsystem" and "Magnetometer Models" in combination with the calculated "Sun Reference" and "Local Magnetic field" reference vectors. The resulting attitude quaternion from "q-Method Attitude" and noisy angular rate measurements from the "Gyroscope Model" are combined with an internal attitude dynamics model in an "Extended Kalman Filter" to provide the best estimate of the CubeSat's orientation and angular rates. A "Control Switch and Duty Cycle" block controls which controller is being used to control the CubeSat's attitude: "Magnetic B-dot Control", "Magnetic PD Control", or the "Reaction Wheel Subsystem". The block labelled "Simscape" simulates the attitude dynamics of the CubeSat, and encompasses the "Satellite Model", "Sun Sensor Subsystem", and the "Reaction Wheel Subsystem". Table 3 provides a detailed description of each of the blocks in Figure 4 and includes the inputs and outputs to each block. In the table, inputs from the MATLAB workspace are indicated with an asterisk. Workspace variables are compiled into two bus structures. The first, “Satellite”, contains the mass, inertia matrix, and other relevant geometric properties of the Simscape satellite model for use outside of Simscape in the simulator. The second, “Earth”, contains the planetary constant parameters for Earth, defining its shape and gravitation. The set of initial conditions (IC) are grouped together as input “IC” and are described subsequently in Section 3.1.1.

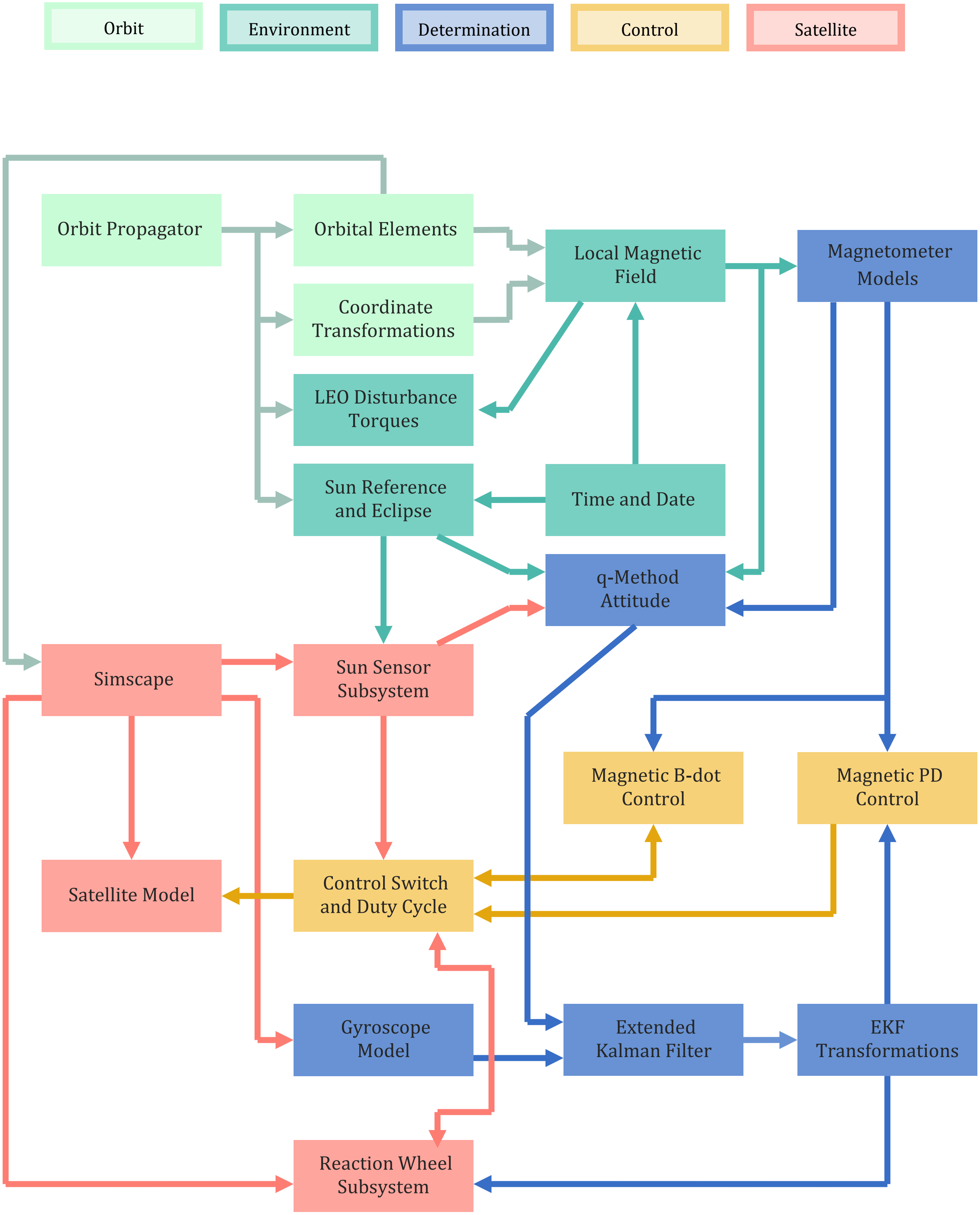


Figure 4: Simulator High-Level Subsystem Block Diagram

Table 3: Simulator Guide

	Block	Inputs	Outputs	Description
Orbit	Orbit Propagator	Earth* IC*	Orbital radius Orbital velocity	This block contains a simple acceleration-based feedback loop that propagates the orbit based on the initial conditions described in Section 3.1.1
	Orbital Elements	Earth* Orbital radius Orbital velocity	Orbital elements True anomaly (∞)	This block calculates the classical orbital elements at each timestep, and contains a function allowing true anomaly to increase infinitely
	Coordinate Transformations	Earth* Orbital radius GMT	Altitude Latitude Longitude	This block contains MATLAB function blocks for each of the coordinate transformations described in Appendix B.
Environment	Time and Date	IC Simulation time Vector date*	GMT Vector date Decimal date Julian date	This block uses simulation time to increase the time and date based on the epoch value assigned in the driving routine. Its outputs include the current date in three formats and the local sidereal time
	Sun Reference and Eclipse	Earth* Orbital radius Julian date Sun vector estimate	Sun reference Eclipse	This block calculates the relative position between Earth and Sun, and uses this value to determine when the satellite is in eclipse
	Local Magnetic Field	Altitude Latitude Longitude Decimal date GMT Orbital elements	Magnetic field	This block contains the local magnetic field model and coordinate transformations thereof
	LEO Disturbance Torques	Earth* Satellite* Orbital radius Orbital velocity Magnetic field	Disturbance torques	This block outputs the summation of the four modelled external disturbance torques
Determination	Magnetometer Models	Magnetic field Noise parameters*	Magnetic field (noisy)	This block models sensor noise for two magnetometers, and outputs BF magnetic field vectors that emulate noisy sensor readings from each
	Gyroscope Model	Actual angular rate Noise parameters*	Angular rate (noisy)	This block models sensor noise and drift for the gyroscope and outputs a BF rate vector that emulates its noisy sensor readings

	Block	Inputs	Outputs	Description
Determination	q-Method Attitude	Sun reference Sun vector Magnetic field Magnetic field (noisy)	Attitude estimate	This block estimates the satellite's attitude quaternion based on modelled sensor readings and their associated reference vectors
	Extended Kalman Filter	Satellite* EKF tuning* Angular rate (noisy) Attitude estimate	EKF attitude EKF angular rate	This block contains the extended Kalman filter, whose attitude quaternion and angular rate vector estimate outputs improve upon those based on raw sensor readings
	EKF Transformations	EKF attitude EKF angular rate	EKF attitude error EKF rate error	This block transforms the EKF attitude quaternion and angular rate estimates from ECI to BF/NP, redefining each as the error between the desired attitude/rate and the current attitude/rate
Control	Magnetic B-Dot Control	Magnetic field (noisy) B-dot tuning*	B-dot control torque	This block contains the B-dot detumbling controller
	Magnetic PD Control	EKF attitude error EKF rate error Magnetic PD tuning*	PD control torque	This block contains the magnetic PD controller, presently unused
	Control Switch and Duty Cycle	B-dot control torque Simulation time Sun vector Angular rate	Control torque (cycle) Control switch	This block applies a 90% duty cycle to the B-dot control torque signal. The control switch signal that switches between the detumbling and pointing controllers is computed separately
Satellite	Simscape	True anomaly (∞) Disturbance torques Magnetic control torques Control torque (cycle) Sun reference EKF attitude error EKF rate error Control Switch	Actual attitude Actual angular rate Actual attitude error Actual rate error Actual Euler angles Actual wheel speeds	This block contains all Simscape components for simulating the satellite's body dynamics. Its "actual" outputs are gleaned from multibody transform sensors and are used as a basis of comparison for their matching estimated values
	Satellite Model	Satellite*	Satellite model Sun sensor locations	This block contains the Simscape model to be actuated, with reference frames defined along its principal inertial axes and on each of its surfaces to locate the Sun sensor assembly

	Block	Inputs	Outputs	Description
Satellite	Sun Sensor Subsystem	Noise parameters* Sun reference Sun sensor locations	Sun vector Sun sensor status	This block computes the Sun vector estimate between satellite and Sun with added photodiode sensor noise and geometric interference from the satellite model. The use status of each Sun sensor is output to the workspace to track the frequency of shadowing from the satellite's deployed solar panels
	Reaction Wheel Subsystem	EKF attitude error EKF rate error Control switch DC motor model*	Wheel torques Wheel speeds	This block contains all reaction wheel components, including PD controllers for their individual voltages and separate DC motor models for each

3.1 Orbit Definition

Within the simulator, parameters defining the geometry and orientation of the satellite's orbit are obtained from the North American Aerospace Defense Command (NORAD) two-line element (TLE) set for the ISS Zarya module, effectively duplicating its orbit for the simulated CubeSat. A single TLE set is used to obtain the initial parameters for the CubeSat orbit including its epoch time and date which define the point at which orbit propagation is set to begin. The initial orbital position is calculated for the satellite at epoch and the orbit is propagated from this point. Equations in this section are defined relative to the Earth-centred inertial (ECI) coordinate frame defined in Appendix A.

3.1.1 Two-Line Elements

The TLE defining the simulator's initial conditions is shown in Table 4 with the corresponding definition of each TLE entry "A" through "R" shown in Table 5. Of the eighteen TLE entries, only the following six are needed in the simulator: the epoch ordinal date "E", mean anomaly "Q", inclination "M", right ascension of the ascending node "N", eccentricity "O", and argument of perigee "P". The data in Table 4 was obtained in the early afternoon of 26 April 2019.

Table 4: Two-Line Element Set [96]

ISS (ZARYA)									
1	25544U	98067A	19116.54834	.00001183	00000-0	26373-4	0	9995
2	25544	51.6413	257.8729	0001068	231.7821	251.6112	15.5259257	

Table 5: Two-Line Element Set Entry Definitions [96] [97, p. 60]

A								
B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	

A	Satellite name
B	Line number
C	NORAD satellite number (U = unclassified)
D	International designator (launch year | launch number of the year | object number of launch)
E	Epoch year | day number of the year.current fraction of day
F	First derivative of mean motion (rate of change of number of orbit revolutions in rev/day)
G	Second derivative of mean motion (terminal orbit decay in rev/day^3)
H	BSTAR drag term and its associated power of 10 in $R_{\oplus}^{-1}$
I	Ephemeris/orbital propagation model (SGP4 model assumed, value is always zero)
J	Checksum/total number of TLE generated for spacecraft
K	Line number
L	NORAD satellite number
M	Inclination (deg)
N	Right ascension of the ascending node (deg)
O	Orbit eccentricity (assume 0.xxxxxxxx)
P	Argument of perigee (deg)
Q	Mean anomaly (deg)
R	Mean motion (rev/day) | orbit number at ascending node epoch | checksum

The right ascension of the ascending node, argument of perigee and inclination obtained from the TLE are illustrated Figure 5, where $\boldsymbol{r}_{p,I}$ is the radius of perigee in the ECI frame, and $\boldsymbol{N}$ is the line of nodes.

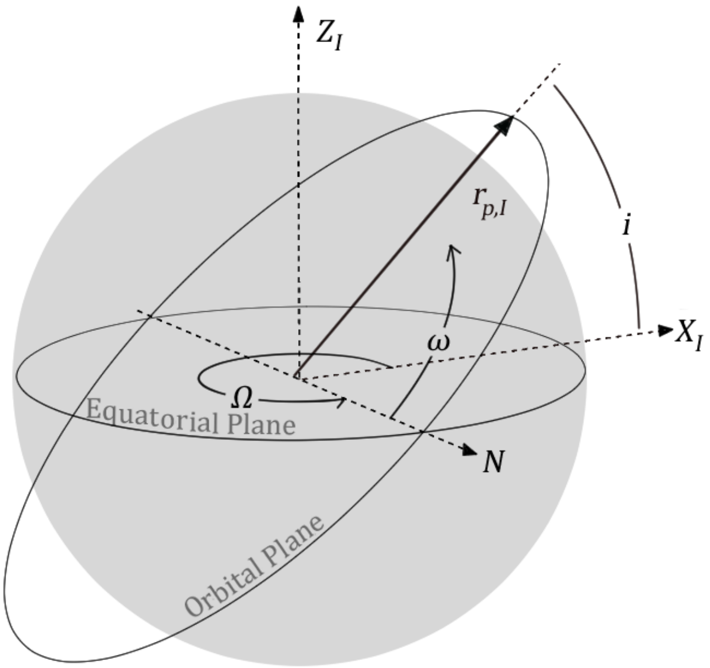


Figure 5: RAAN, Argument of Perigee, and Inclination in ECI

The right ascension of the ascending node (RAAN) (alternatively termed the "argument of the ascending node") defines the angle between the ECI X-axis and the line of nodes $\boldsymbol{N}$ directed along the intersection between the orbital and equatorial planes [97, p. 54]. It is calculated in the simulator via the following [97] [98]:

$$\sin\Omega = \frac{\boldsymbol{h}_I \cdot \boldsymbol{I}_x}{h_I \sin i} \tag{1}$$

$$\cos\Omega = \frac{-\boldsymbol{h}_I \cdot \boldsymbol{I}_y}{h_I \sin i} \tag{2}$$

where $\boldsymbol{I}_x$ and $\boldsymbol{I}_y$ are unit vectors in ECI defined as $[1 \quad 0 \quad 0]^T$ and $[0 \quad 1 \quad 0]^T$, respectively, and the orbital angular momentum is given by the ECI radius and velocity via [97, p. 54]:

$$\boldsymbol{h}_I = \boldsymbol{r}_I \times \boldsymbol{v}_I \tag{3}$$

Then,

$$\Omega = \operatorname{atan2}(\sin\Omega, \cos\Omega) \tag{4}$$

Orbit inclination defines the angle between the orbital and equatorial planes. The initial value for orbit inclination i is obtained from the TLE; it is assumed that this value is maintained throughout an unperturbed orbit. Inclination can be calculated from the ECI orbital angular momentum vector $\boldsymbol{h}_I$ via the following relation [97, p. 54]:

$$i = \cos^{-1}\left(\frac{\boldsymbol{h}_I \cdot \boldsymbol{I}_Z}{h_I}\right) \tag{5}$$

Within the scope of this simulator, a two-body orbit propagator is used which neglects the gravitational effects of Moon, Sun, and Earth's oblateness. As a result of this assumption, Ω, ω, and i should not change with time. To validate the orbit propagation within the simulator, each of these three angles are calculated at each time step using Equations 1 through 5 to ensure they remain constant with respect to the initial TLE values.

These angles constitute three of six classical orbital elements which can be used to describe a satellite's orbit. The remaining three classical orbital elements are: orbit eccentricity, orbit semimajor axis, and the true anomaly. Orbit eccentricity and orbit semimajor axis should remain constant when using a two-body propagator, while the true anomaly can be calculated from the mean anomaly obtained from the TLE.

Eccentricity defines the oblateness of the orbit. The eccentricity vector is directed along the semimajor axis and is considered positive in the perigee direction. It is calculated as follows [97, p. 53]:

$$\boldsymbol{e} = \frac{\boldsymbol{v}_I \times \boldsymbol{h}_I}{\mu_\oplus} - \frac{\boldsymbol{r}_I}{r_I} \tag{6}$$

where $\boldsymbol{h}_I$ is the orbital angular momentum, defined in ECI as [97, p. 53]:

$$\boldsymbol{h}_I = \boldsymbol{r}_I \times \boldsymbol{v}_I \tag{7}$$

The scalar value for eccentricity is found via the norm of its vector:

$$e = \|\boldsymbol{e}\|$$

Eccentricity should maintain a constant value in the absence of external perturbations, equivalent to that initially obtained from the TLE. An eccentricity of zero corresponds to a circular orbit while elliptical orbits have values ranging between 0 and 1.

The argument of perigee angle ω is drawn between the line of nodes $\boldsymbol{N}$ and the perigee of the orbit $\boldsymbol{r}_{p,I}$. The line of nodes is defined by the following [97, p. 54]:

$$\boldsymbol{N} = \boldsymbol{I}_z \times \boldsymbol{h}_I \tag{8}$$

As the perigee radius coincides with the eccentricity vector, the argument of perigee can be found as follows [97, p. 54]:

$$\omega = \cos^{-1}\left(\frac{\boldsymbol{N} \cdot \boldsymbol{e}}{Ne}\right) \tag{9}$$

The quadrant condition is as follows [97, p. 54]:

$$0 \leq \omega \leq 180° \quad \text{if} \quad \boldsymbol{e} \cdot \boldsymbol{I}_Z \geq 0$$

$$180° < \omega < 360° \quad \text{if} \quad \boldsymbol{e} \cdot \boldsymbol{I}_Z < 0$$

The argument of perigee taken from the TLE appears in the first quadrant and thus no logic is required to place it in the correct quadrant after the initial calculation.

With the line of nodes direction defined using the RAAN as [98]:

$$\boldsymbol{M} = \begin{bmatrix} \cos\Omega \\ \sin\Omega \\ 0 \end{bmatrix}$$

and the orbit normal defined as [98]:

$$\boldsymbol{n} = \frac{\boldsymbol{h}_I}{h_I}$$

the argument of perigee can then be defined relative to the true anomaly [98]:

$$\sin(\omega + \theta) = \frac{\boldsymbol{n}^T(\boldsymbol{M} \times \boldsymbol{r}_I)}{r_I}$$

$$\cos(\omega + \theta) = \frac{\boldsymbol{r}_I^T(\boldsymbol{M})}{r_I}$$

$$\omega = \text{atan2}(\sin(\omega + \theta), \cos(\omega + \theta)) - \theta \tag{10}$$

This approach is currently implemented in the simulator. To maintain the argument of perigee within the range of 0° to 360°, 2π is added to the radian value if the difference in the above equation results in a negative value for ω.

The initial value for the semimajor axis a of the orbit is defined from the radius at perigee and the TLE eccentricity. Perigee radius r_p in the simulator was obtained from perigee altitude h_p from the TLE data compiled on [99]. The semimajor axis is then obtained as [97, p. 39]:

$$a = \frac{r_p}{(1 - e)} \tag{11}$$

The true anomaly defines the satellite's angular position in orbit with respect to the orbit's perigee. Its initial value is obtained from the mean anomaly given in the TLE, where the mean anomaly M is defined by the mean motion n – the number of orbit revolutions per day as reported in the TLE – and the time since perigee passage for the dataset [100, p. 88]. To obtain the true anomaly from this reported angle, it is necessary to define an eccentric anomaly which arises from the definition of the orbit ellipse in polar form [100, p. 84]. The TLE epoch time as given by the ordinal date entry defines the start time for the simulator. As such, the true anomaly given by the mean anomaly is nonzero (i.e. simulations do not start at orbit perigee). The mean anomaly is given in degrees and must be converted to radians for use in the equations that follow.

Figure 6 illustrates the true anomaly, eccentric anomaly, and mean anomaly for a highly elliptical orbit at an arbitrary orbital radius drawn in red. The radius of the outer grey circle is equal to the semimajor axis of the elliptical orbit and thus both orbits have equal values of mean motion n. At any position r occurring at time t with true anomaly θ and known time of perigee passage t_p, the expression $M = n(t - t_p)$ determines the mean anomaly in terms of Kepler's third law: the area of the circular orbit "swept" during $(t - t_p)$ at M is equivalent to that area "swept" during $(t - t_p)$ in the elliptical orbit at θ [97, p. 46]. These areas are denoted on the diagram as A and B, respectively.

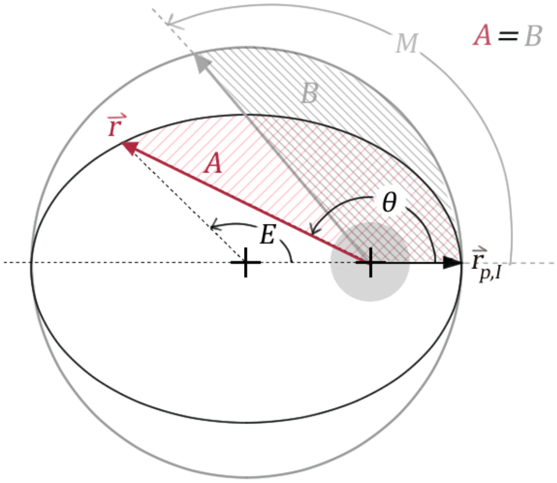


Figure 6: True Anomaly, Eccentric Anomaly, and Mean Anomaly for an Arbitrary Point in an Elliptical Orbit

The eccentric anomaly can be obtained from the mean anomaly via the following relation [97, p. 47]:

$$M = E - e \sin E \tag{12}$$

The true anomaly is then obtained from the eccentric anomaly by solving [97, p. 47]:

$$\tan\frac{\theta}{2} = \sqrt{\frac{1+e}{1-e}}\tan\frac{E}{2} \tag{13}$$

To obtain the true anomaly at any subsequent point in the orbit, the following relation is used [98]:

$$\sin\theta = \frac{p}{her_I}\boldsymbol{r}_I{}^T\boldsymbol{v}_I$$

$$\cos\theta = \frac{p - r_I}{er_I}$$

$$\theta = \operatorname{atan2}(\sin\theta, \cos\theta) \tag{14}$$

The true anomaly varies between 0° to 360° with 0° defined at perigee and is positive in the satellite's direction of travel. A correction is made in the simulator's code to round calculated values of $\cos\theta > 1$ down to 1. This correction is the result of numerical integration errors during orbit propagation which, at worst, are on the order of 10^{-6} when recalculating the orbital elements at each time step.

To prevent anomalous and discontinuous results that would otherwise arise in the simulator at the point where the true anomaly resets from 360° back to 0°, the value of the true anomaly is allowed to monotonically increase with time beyond 360°. This continuity compensation is handled in the simulator using a custom MATLAB function block with a discretized input true anomaly value (between 0° to 360°), calculated via Equation 14 at a sample time of one second. The calculated true anomaly value is compared to the value from the previous timestep; if the current value is less than the previous value, an appropriate integer multiple of 2π is added to the current true anomaly value to ensure continuity of the true anomaly being calculated.

3.2 Orbit Dynamics

The final equations needed to define the orbit dynamics within the simulator are those required for orbit propagation: orbital radius, orbital velocity, and orbital acceleration. These equations are defined relative to the Earth-centred inertial (ECI) and perifocal reference frames described in Appendix A.

In the simulator, initial vectors are obtained for the orbital radius and velocity in the ECI frame. These values are used to calculate the orbital acceleration at each time step, and subsequent values are obtained by integrating this acceleration term.

An initial value for the ECI radius at epoch is obtained in the perifocal frame of reference using initial values obtained from the TLE. The orbital radius can be expressed in the perifocal frame as follows [97, p. 51]:

$$\boldsymbol{r}_P = \begin{pmatrix} r\cos\theta \\ r\sin\theta \\ 0 \end{pmatrix} \tag{15}$$

where θ is the initial value of the true anomaly at epoch and r is the radius at epoch, calculated via the orbit equation [97, p. 51]:

$$r = \frac{a(1-e^2)}{1+e\cos\theta} \tag{16}$$

The initial orbital radius in ECI is then obtained via the perifocal to ECI rotation matrix:

$$\boldsymbol{r}_I = \boldsymbol{C}_{IP}\boldsymbol{r}_P$$

Angle data used in the rotation matrix $\boldsymbol{C}_{IP}$ is obtained from the epoch TLE. This $\boldsymbol{r}_I$ vector is used as an initial condition in the corresponding integrator block in the simulator.

Similarly, the initial velocity is obtained in the perifocal frame as follows [97, p. 52]:

$$\boldsymbol{v}_P = \begin{bmatrix} -\sqrt{\frac{\mu_\oplus}{p}}\sin\theta \\ \sqrt{\frac{\mu_\oplus}{p}}(e+\cos\theta) \\ 0 \end{bmatrix} \tag{17}$$

where values for the eccentricity e and semilatus rectum p are obtained from the same TLE as the epoch. The ECI initial velocity used as the initial condition in its corresponding integrator block is obtained as follows for the same rotation matrix as used for the orbital radius:

$$\boldsymbol{v}_I = \boldsymbol{C}_{IP}\boldsymbol{v}_P$$

The method of orbit propagation used in the simulator centres around the calculation of the Keplerian instantaneous acceleration at each timestep, and the integration of this value to obtain its associated orbital velocity and radius. The instantaneous acceleration is given by [97, p. 72]:

$$\boldsymbol{a}_I = -\frac{\mu_\oplus}{r_I^3}\boldsymbol{r}_I + \boldsymbol{f}_p \tag{18}$$

where $\boldsymbol{f}_p$ is the sum of all perturbing accelerations acting upon the satellite. For the scope of this research, $\boldsymbol{f}_p$ it is assumed to be zero. It is also assumed that the location of the CubeSat within its orbit can be precisely determined at any given time. This assumption eliminates the potential for errors in determining the reference attitude of the satellite.

3.2.1 Simscape Coordinate Frame Definition

Appendix A defines the coordinate frames used throughout simulator development, four of which are of relevance to this section. The ECI and perifocal frames define the location of the centre of the Earth and position of orbital plane relative to it, while the Nadir-Pointing (NP) and Body-Fixed (BF) frames defined the desired nadir-pointing attitude and the orientation of the rotating satellite relative to the satellite's centre of mass. The transformations between these frames are detailed in Appendix B.

Three main coordinate frames with respect to which satellite dynamics are defined are used within Simscape. The Simscape multibody "World Frame" block represents the ECI frame described in Appendix A. The NP frame is defined from the ECI frame via a three-step sequence combining the transformations drawn in Appendix B. The ECI to perifocal transformation matrix $\boldsymbol{C}_{IP}$ is set constant in a rigid transform block, which is connected as the “base” frame to a multibody revolute joint with an input motion signal taking in the infinitely increasing true anomaly value. The “follower” frame from this joint is connected to another rigid transform block containing the last component of the perifocal to NP transformation, the transpose of the product $\boldsymbol{C}_3(\pi/2)\boldsymbol{C}_2(-\pi/2)$:

$$\boldsymbol{C}_{PNP} = \begin{pmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}$$

Lastly, the BF frame is defined as a reference frame in the satellite model multibody solid block, with+ Z oriented along the proper axis according to the model being used.

3.2.2 Satellite Model

Simulator development, initial analyses and algorithm validation were based on two simplified geometries – a 1U cube and a 2U prism – while final results were obtained for geometry based on an interim model of the LORIS CubeSat. The three geometries along with the orientation of the body-fixed (BF) axes are illustrated in Figure 7, Figure 8, and Figure 9, respectively. The body-fixed frame is aligned with the satellite’s principal inertial axes with its origin at the satellite’s centre of mass such that its +Z axis points nadir (aligned with the orbital radius) and its +X axis lies in the direction of spacecraft propagation (aligned with the body-fixed spacecraft velocity vector).

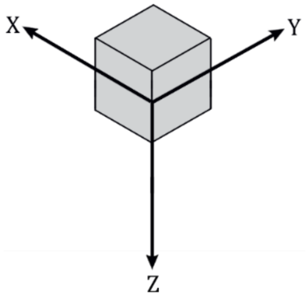


Figure 7: “1U” Geometry

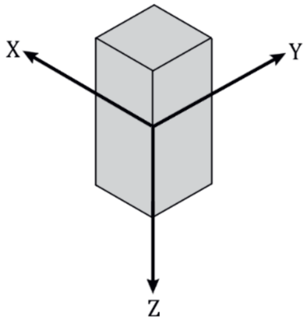


Figure 8: “2U” Geometry

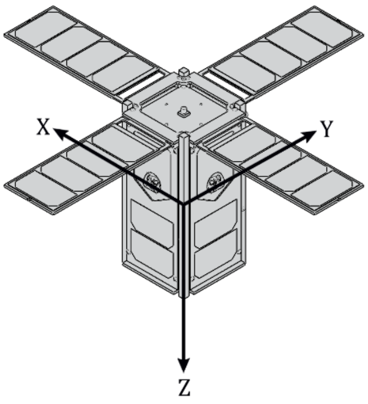


Figure 9: “Final” Geometry [1]

Relevant mass properties for the three geometries are detailed in Table 6. For the 1U and 2U models, the mass and inertia matrices were calculated based on a default density value of 1000 kg/m^3 assigned in their respective Simscape multibody solid blocks. The properties and dimensions for the final geometry were obtained from the LORIS satellite SolidWorks assembly’s mass properties.

Table 6: Parameters for Satellite Geometries

Model	Parameter	Value
1U		
	Dimensions	$0.1 \times 0.1 \times 0.1$ m
	Mass	$m = 1$ kg
	Principal Moments of Inertia	$J_x = 0.001667\ \text{kgm}^2$ $J_y = 0.001667\ \text{kgm}^2$ $J_z = 0.001667\ \text{kgm}^2$
	Centre of Mass	$\text{CoM} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\ \text{m}$
Model	Parameter	Value
2U		
	Dimensions	$0.1 \times 0.1 \times 0.2$ m
	Mass	$m = 2$ kg
	Principal Moments of Inertia	$J_x = 0.00833\ \text{kgm}^2$ $J_y = 0.008333\ \text{kgm}^2$ $J_z = 0.003333\ \text{kgm}^2$
	Centre of Mass	$\text{CoM} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\ \text{m}$
Final		
	Mass	$m = 1.7856$ kg
	Principal Moments of Inertia	$J_x = 0.01580\ \text{kgm}^2$ $J_y = 0.01581\ \text{kgm}^2$ $J_z = 0.01591\ \text{kgm}^2$
	Centre of Mass	$\text{CoM} = \begin{bmatrix} 0.01971 \\ -0.02466 \\ 0.07851 \end{bmatrix}\ \text{m}$

3.3 Space Environment

Elements of the space environment both benefit and detriment the ability of the ADCS to acquire and maintain its attitude. Thus, assessing control performance and robustness of the proposed system requires a realistic representation of the environment that the satellite will be subjected to throughout its mission life. This section details the equations and theory behind the aspects of the space environment modelled in the simulator.

3.3.1 Time and Date

Decimal time is tracked in the simulator in units of hours. Simulation time output by Simulink's clock is added to the starting UTC time to propagate the simulation from the epoch assigned in the driving routine. Calendar date is tracked with persistent year, month, and day variables in a MATLAB function block and carried through the simulator as a YMD vector. The day integer increases when the decimal time passes exactly 24 hours, while month and year integers increase according to the day integer. Leap year recognition has been hardcoded into the date and time subsystem until 2024.

3.3.2 Geomagnetic Field

Earth's LEO geomagnetic field is modelled using the International Association of Geomagnetism and Aeronomy (IAGA)'s International Geomagnetic Reference Field 12 (IGRF-12) model block from the Simulink Aerospace Blockset [101]. The model's magnetic field estimates are considered accurate until 2020 [101]. Inputs of altitude in meters, geodetic latitude and longitude in degrees, and decimal year are fed into the block at one-second intervals to obtain the local magnetic field vector in nanotesla relative to the North-East-Down (NED) reference frame described in Appendix A. This field is transformed to the BF reference frame to emulate both readings from the magnetometers and the "actual" magnetic field reference vector that interacts with outputs from the magnetic attitude controllers to produce Simscape torque inputs.

3.3.3 Sun Reference Vector

The second reference vector used in most attitude determination algorithms is the vector distance between Earth and Sun, which is calculated based on Seidelmann's 1992 "low precision ephemeris" algorithm [102, p. 485]. This vector lies in the ecliptic – the plane formed by the apparent orbit of the Sun as viewed from Earth. Definition of the vector requires knowledge of the time in Julian centuries T_{J_D} [103, p. 261]:

$$T_{JD} = \frac{J_{D(h,m,s)} - 2451545}{36525} \tag{19}$$

for the accurate-to-time Julian date defined below, with year Y, month M, day D, hour h, minute m, and second s [104, p. 33]:

$$J_{D(h,m,s)} = 1721013.5 + 367Y - \mathrm{fix}\left\{\frac{7}{4}\left[Y + \mathrm{fix}\left(\frac{M+9}{12}\right)\right]\right\} + \mathrm{fix}\left(\frac{275M}{9}\right) + D + \frac{60h + m + s/60}{1440} \tag{20}$$

The quantity T_{J_D} is then used to calculate four "solar arguments" [102, p. 485]. The first two, the mean longitude $\phi_\odot$ and mean anomaly $M_\odot$ for the Sun in degrees, are defined below [104, p. 420]:

$$\phi_\odot = 280.460 + 36000.771\, T_{J_D} \tag{21}$$

$$M_\odot = 357.5277233 + 35999.05034\, T_{J_D} \tag{22}$$

Quantities $\phi_{\odot}$ and $M_{\odot}$ are constrained within the bounds 0° - 360° using MATLAB's modulo function in the simulator. These quantities are then used to calculate the remaining two solar arguments: the ecliptic longitude $\phi_{ecliptic}$ and the angle between Earth's equatorial plane and the ecliptic termed the obliquity ϵ [104, p. 421].

$$\phi_{ecliptic} = \phi_{\odot} + 1.914666471 \sin M_{\odot} + 0.019994643 \sin(2M_{\odot}) \tag{23}$$

$$\epsilon = 23.439291 - 0.0130042 T_{J_D} \tag{24}$$

The magnitude of the distance between Earth and Sun is then found relative to the mean anomaly [104, p. 421]

$$r_{\oplus\odot} = 1.000140612 - 0.016708617 \cos M_{\odot} - 0.000139589 \cos 2M_{\odot} \tag{25}$$

which is multiplied by a unit vector to obtain the distance vector [104, p. 421]

$$\boldsymbol{r}_{\oplus\odot_I} = \boldsymbol{S}_I = r_{\oplus\odot} \begin{bmatrix} \cos \phi_{ecliptic} \\ \cos \epsilon \sin \phi_{ecliptic} \\ \sin \epsilon \sin \phi_{ecliptic} \end{bmatrix} \tag{26}$$

in Astronomical Units (AU). One AU is equivalent to the average distance between Earth and Sun, namely:

$$1 \text{ AU} = 149597870700 \text{ m}$$

For April 26th 2019 at 13:09:36 UTC, the following Sun vector was obtained from the above series of equations implemented in the simulator:

$$\boldsymbol{S}_I = \begin{bmatrix} 0.813805 \\ 0.542968 \\ 0.235377 \end{bmatrix} \text{Au} = \begin{bmatrix} 1.217435 \times 10^8 \\ 0.812269 \times 10^8 \\ 0.352119 \times 10^8 \end{bmatrix} \text{km}$$

$$S_I = \|\boldsymbol{S}_I\| = 1.505296 \times 10^8 \text{ km}$$

This Sun vector was validated using the NASA JPL Horizons System (accessed through telnet) from which the following ephemeris data was obtained from a generated vector table with the target body as Sun's centre (code @Sun) and the coordinate centre as Earth's "geocenter" (code 399) in the ICRF/J2000 reference frame "frame" (with respect to "Earth mean equator and equinox of reference epoch"). For 13:09:00 UTC (Julian time 2458600.047916667):

$$\boldsymbol{S}_I = \begin{bmatrix} 0.816526 \\ 0.539517 \\ 0.233877 \end{bmatrix} \text{Au} = \begin{bmatrix} 1.221505 \times 10^8 \\ 0.807106 \times 10^8 \\ 0.349875 \times 10^8 \end{bmatrix} \text{km}$$

$$S_I = \|\boldsymbol{S}_I\| = 1.505293 \times 10^8 \text{ km}$$

There exists a magnitude error of 273.07 km (1.814×10^{-4} %) and an error in angle of

$$\theta_{err} = \cos^{-1}\left(\frac{\vec{a} \cdot \vec{b}}{ab}\right) = 0.2644^{\circ}$$

between the calculated and actual vectors. Additional vector calculations and their actual values over a wide range of times and dates are compiled in Table 7. Absolute errors in angle between the calculated and actual vectors never exceed 1° and thereby fall within the desired range of attitude determination accuracy

Table 7: Sun Reference Vector Error Calculations

Date and Time	Calculated S_I (km)	Actual S_I (km)	Magnitude Error	Angle Error
2018-Sep-10 18:45:00 2458372.28125	$\begin{bmatrix} -1.473442 \times 10^8 \\ 0.287086 \times 10^8 \\ 0.124453 \times 10^8 \end{bmatrix}$	$\begin{bmatrix} -1.472059 \times 10^8 \\ 0.293146 \times 10^8 \\ 0.127089 \times 10^8 \end{bmatrix}$	3497.91 km	0.257°
2008-Nov-04 23:27:00 2454775.47708	$\begin{bmatrix} -1.086163 \times 10^8 \\ -0.926718 \times 10^8 \\ -0.401759 \times 10^8 \end{bmatrix}$	$\begin{bmatrix} -1.088292 \times 10^8 \\ -0.924625 \times 10^8 \\ -0.400852 \times 10^8 \end{bmatrix}$	918.52 km	0.121°
1993-Aug-08 10:24:00 2449207.93333	$\begin{bmatrix} -1.088910 \times 10^8 \\ 0.968786 \times 10^8 \\ 0.420037 \times 10^8 \end{bmatrix}$	$\begin{bmatrix} -1.090539 \times 10^8 \\ 0.967157 \times 10^8 \\ 0.419332 \times 10^8 \end{bmatrix}$	6400.97 km	0.091°
1981-May-27 03:24:00 2444751.64167	$\begin{bmatrix} 0.622251 \times 10^8 \\ 1.268146 \times 10^8 \\ 0.549872 \times 10^8 \end{bmatrix}$	$\begin{bmatrix} 0.616012 \times 10^8 \\ 1.270752 \times 10^8 \\ 0.551003 \times 10^8 \end{bmatrix}$	4440.92 km	0.259°
1970-Oct-22 18:08:00 2440882.25556	$\begin{bmatrix} -1.302823 \times 10^8 \\ -0.660491 \times 10^8 \\ -0.286410 \times 10^8 \end{bmatrix}$	$\begin{bmatrix} -1.297611 \times 10^8 \\ -0.669025 \times 10^8 \\ -0.290116 \times 10^8 \end{bmatrix}$	2371.52 km	0.411°
		Average:	3525.99 km	0.228°

3.3.4 Eclipse

The orbiting satellite is considered to be “in eclipse” while it is travelling through Earth’s shadow, a period wherein its attitude cannot be determined due to inadequate Sun sensor data. Figure 10 illustrates the eclipse region of an arbitrary elliptical orbit as the solid portion of the orbit trace.

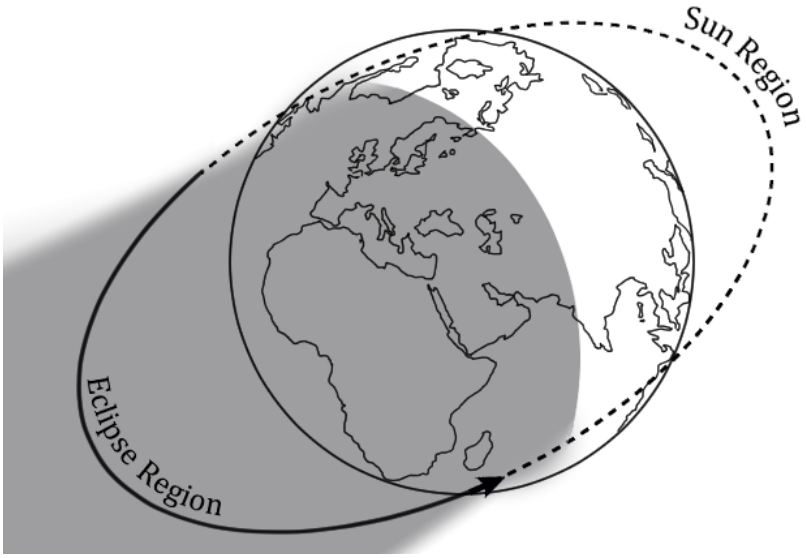


Figure 10: Eclipse Region for a Near-Circular LEO Orbit

The eclipse condition is determined in the simulator from the orbital radius in ECI $\boldsymbol{r}_I$ and the vector from Earth to Sun $\boldsymbol{r}_{\oplus\odot_I}$ via the following relation [104, p. 422]:

$$\boldsymbol{r}_I \cdot r_{\oplus\odot_I} < -\sqrt{r_I^2 - R_\oplus^2} \tag{27}$$

where $R_\oplus$ is Earth's equatorial radius. When this condition is satisfied, the satellite is in Earth's shadow and attitude is considered inestimable. The satellite spends 2111 s or about 38% of each orbit in eclipse.

3.3.5 Disturbance Torques

In LEO, gravity gradient, atmospheric drag, and solar radiation pressure are the primary external torque disturbances perturbing a satellite's attitude throughout its orbit. The torque that arises from the satellite's residual magnetic dipole is the sole internal torque disturbance. These four torque disturbances are implemented in the simulator and can be summed together with the calculated control torque from the magnetorquers as an input to the Simscape model where all torques are resolved in the BF frame. To validate the simulated disturbance torques, the values reported in subsequent sections were calculated using a simplified 2U prism model with the inertia tensor reported in Table 6.

Two attitude cases are considered: inertially-fixed and nadir-pointing. These attitudes are illustrated at four points in the orbit in Figure 11 and Figure 12, respectively.

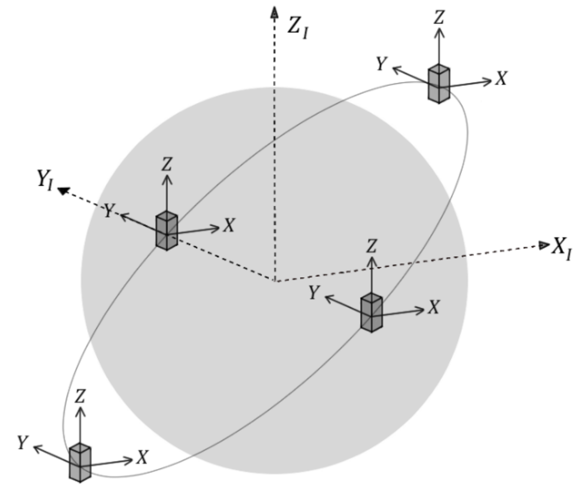


Figure 11: Inertially-Fixed Attitude for Disturbance Torque Validation

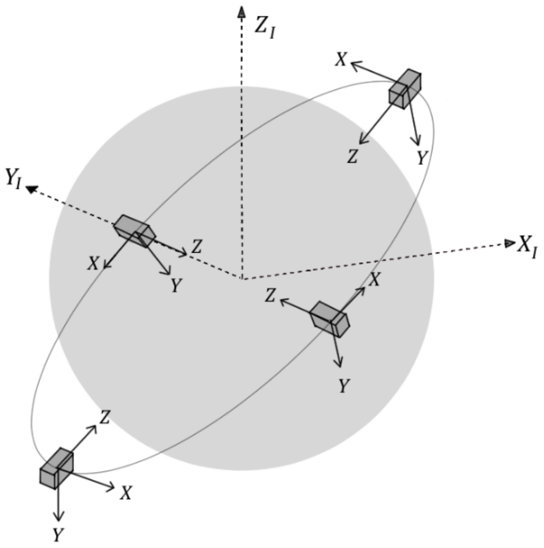


Figure 12: Nadir-Pointing Attitude for Disturbance Torque Validation

3.3.5.1 Gravity Gradient

The vector representation of the gravity gradient (GG) disturbance torque $\vec{T}_{GG}$ is defined in BF coordinates as [105, p. 324]:

$$\vec{T}_{GG} = \frac{3\mu}{r_B^3} \vec{u}_e \times (\boldsymbol{J} \cdot \vec{u}_e) \tag{28}$$

where $\mu = 3.9859 \times 10^{14}$ m$^3/s^2$ is the gravitational parameter of Earth, r_B is the BF vector distance between Earth's centre and the spacecraft's CoM in m, $\vec{u}_e$ is the BF unit vector towards nadir which emanates from the spacecraft CoM, and $\boldsymbol{J}$ is the spacecraft inertia tensor about the CoM. The dot product of the inertia tensor and the nadir direction unit vector can be represented as:

$$(J \cdot \vec{u}_e) = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \cdot \vec{u}_e = \begin{bmatrix} [J_x & 0 & 0]^T \cdot \vec{u}_e \\ [0 & J_y & 0]^T \cdot \vec{u}_e \\ [0 & 0 & J_z]^T \cdot \vec{u}_e \end{bmatrix}$$

Figure 13 plots the magnitude of the GG torque disturbance as a function of time over one orbit for the inertially-fixed 2U Prism Model illustrated in Figure 8, while Figure 14 plots the individual torque components about each of the BF axes. The variation in the unit vector towards nadir relative to the satellite's principal axes for the inertially-fixed case results in a cyclic profile for the BF X and Y GG torque components and zero for the Z GG torque component. The torque norm reaches zero at the two points in the orbit where the satellite crosses the equatorial plane.

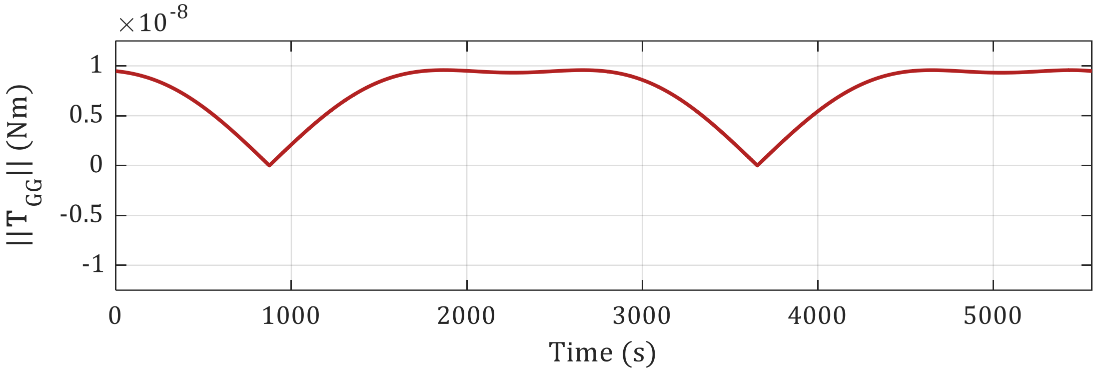


Figure 13: Gravity Gradient Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude

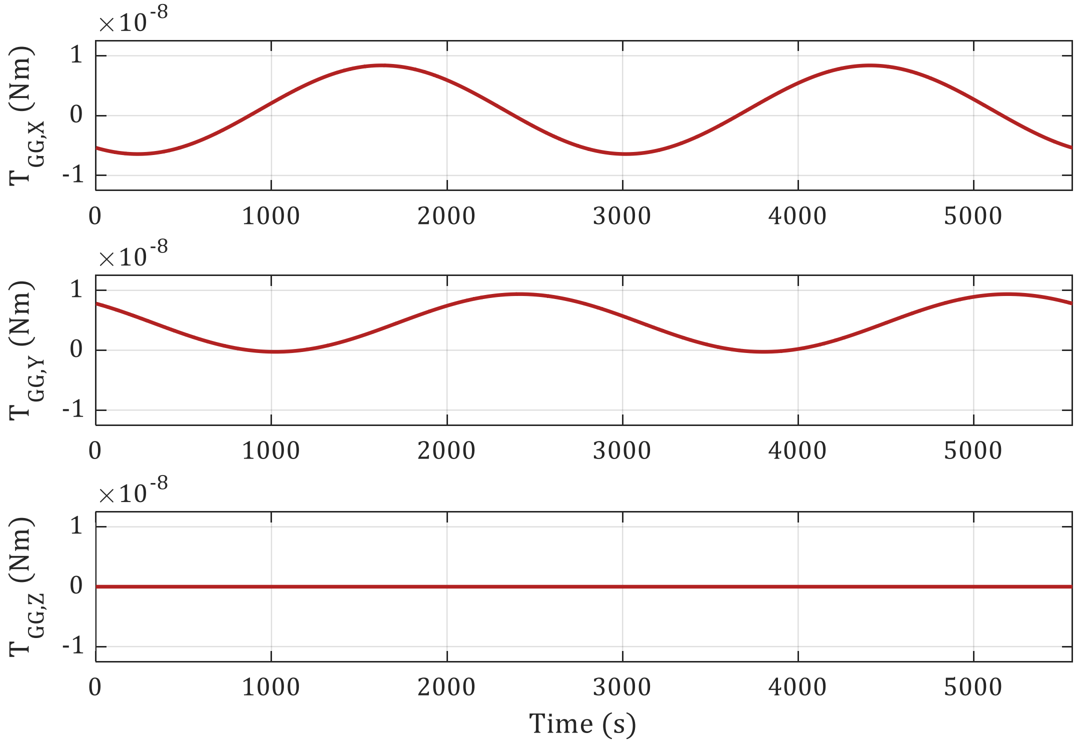


Figure 14: Gravity Gradient Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude

The maximum GG torque for the inertially fixed attitude case is:

$$\boldsymbol{T}_{GG} = \begin{bmatrix} 7.31 \times 10^{-9} \\ 6.18 \times 10^{-9} \\ 0 \end{bmatrix} \text{ Nm}$$

$$T_{GG} = \|\boldsymbol{T}_{GG}\| = 9.57 \times 10^{-9} \text{ Nm}$$

For the nadir-pointing prism satellite illustrated in Figure 8, the GG torques generated demonstrated virtually no variation in magnitude, coinciding with the prediction of a constant profile for circular orbits stated in [105, p. 366]. The maximum GG torque was obtained for the nadir pointing case as follows, notably one order of magnitude lower than that for the inertially-pointing case.

$$\boldsymbol{T}_{GG} = \begin{bmatrix} 4.01 \times 10^{-11} \\ 2.66 \times 10^{-10} \\ 0 \end{bmatrix} \text{ Nm}$$

$$T_{GG} = \|\boldsymbol{T}_{GG}\| = 2.69 \times 10^{-10} \text{ Nm}$$

To validate the simulation results, the worst-case scalar formulation of the gravity gradient disturbance torque defined in BF coordinates was evaluated as follows [105, p. 366]:

$$T_{GG} = \frac{3\mu}{2r_B^3} \left|J_z - J_y\right| \sin(2\theta) \tag{29}$$

where θ is the "maximum deviation of the Z-axis from the local vertical in radians." This equation can be assumed maximised when $\sin(2\theta) = 1$ and r_B is taken at perigee:

$$r_{B,p} = 6.78 \times 10^6 \text{ m}$$

$$T_{GG} = 9.57 \times 10^{-9} \text{ Nm}$$

This worst-case value is equivalent to the maximum magnitude obtained for the inertially-pointing case, suggesting a valid implementation of the vector equation in the simulator. To further validate the simulation results, the worst-case gravity gradient torque value of 9.57×10^{-9} Nm was compared to that predicted for the 2U LEO CubeSat AntelSat reported at 3.9×10^{-9} Nm [106]. The discrepancy between the values can be ascribed to the difference in moments of inertia; For the AntelSat, $\left|J_z - J_y\right|$ was reported as 0.002524, i.e. less than that for the 2U prism by a factor of 0.5048 [106]. The AntelSat also calculates the torque at an orbital radius of 7000 km, a factor of 0.9103 larger than that for the 2U prism [106]. Taking the product of these factors implies that the GG torque obtained for the AntelSat should be a factor of 0.4031 less than that obtained using Equation (29) for the 2U prism:

$$(9.57 \times 10^{-9})(0.4031) = 3.86 \times 10^{-9}$$

The adjusted value is approximately equivalent to the reported AntelSat value of 3.9×10^{-9} which further validates the vector implementation of the GG torque within the simulator.

3.3.5.2 Aerodynamic Drag

The aerodynamic drag vector torque can be calculated via the following relation [105, p. 324]:

$$\boldsymbol{T}_D = \frac{1}{2}\rho v_B^2 C_D A\left(\boldsymbol{u}_v \times \boldsymbol{s}_{cp}\right) \tag{30}$$

where $\rho = 2.72 \times 10^{-12}$ kg/m^3 is the atmospheric density at 400 km altitude (assumed constant in the simulator) [105, p. 502], v_B is the magnitude of the orbital velocity vector in BF, C_D is the drag coefficient, A is the projected area of the spacecraft perpendicular to the orbital velocity vector, $\boldsymbol{u}_v$ is the unit velocity vector in BF, and $\boldsymbol{s}_{cp}$ is the vector distance from the centre of pressure to the satellite's centre of mass. The value of the drag coefficient C_D typically falls between 2 and 2.5, with the median value 2.25 selected for use in the simulator according to [105, p. 324] [105, p. 366]. The projected area is assumed to be constant for the 2U prism at a nominal value of 0.02 m^2, while $\boldsymbol{s}_{cp}$ is assumed to be fixed along the +X direction at 0.02 m – a value selected to facilitate comparison with the aerodynamic drag reported for two existing CubeSats – AntelSat and COMPASS-1 [106] [107]. Figure 15 and Figure 16 plot the magnitude and vector components of the aerodynamic drag torque over one orbit for the inertially-fixed 2U prism attitude. The cyclic nature of the magnitude in Figure 17 is the result of fixed axes travelling in a circular orbit – the torque at each timestep is proportional to the cross product of a fixed vector in +X and a vector that rotates about it within the inclined orbital plane.

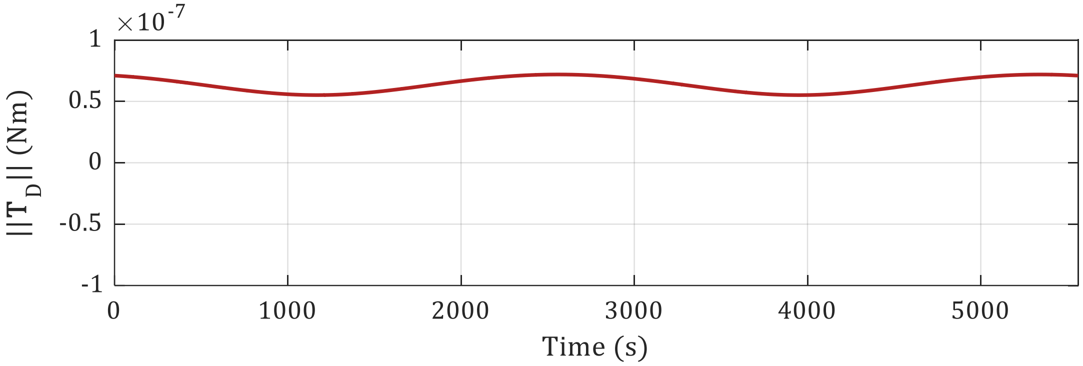


Figure 15: Aerodynamic Drag Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude

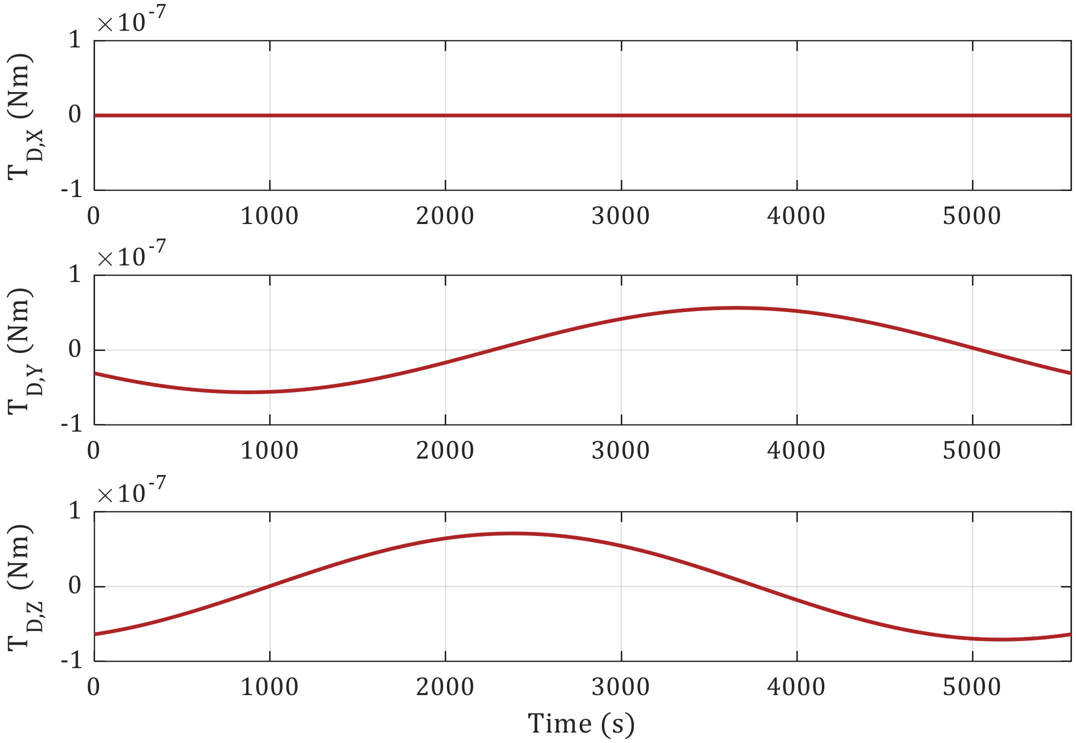


Figure 16: Aerodynamic Drag Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Inertially-Fixed Attitude

The maximum aerodynamic drag torque obtained for the inertially-fixed case was found to be:

$$\boldsymbol{T}_D = \begin{bmatrix} 0 \\ 1.84 \times 10^{-8} \\ 6.95 \times 10^{-8} \end{bmatrix} \text{ Nm}$$

$$T_D = \|\boldsymbol{T}_D\| = 7.19 \times 10^{-8} \text{ Nm}$$

In the nadir-pointing orientation, the value of the torque is near-constant across the orbit as expected for a circular orbit [105, p. 324]. Torques in the nadir-pointing orientation are an order of magnitude lower than those for the inertially-fixed case. When the satellite is oriented nadir, the unit velocity vector maintains its direction throughout the orbit when viewed in the BF frame under the assumption of an unchanging centre of pressure vector. When $\boldsymbol{s}_{cp}$ is fixed, therefore, the magnitude of the cross product in the vector definition will vary little. This invariance is not the case when the satellite is inertially-fixed, and as a result the magnitude of the cross product varies cyclically as the velocity vector rotates about the fixed $\boldsymbol{s}_{cp}$. The maximum aerodynamic drag torque for the nadir-pointing case was calculated in the simulator to be:

$$\boldsymbol{T}_D = \begin{bmatrix} 0 \\ -9.96 \times 10^{-9} \\ -9.56 \times 10^{-9} \end{bmatrix} \text{ Nm}$$

$$T_D = \|\boldsymbol{T}_D\| = 1.38 \times 10^{-9} \text{ Nm}$$

For validation, the worst-case aerodynamic drag scalar torque can be calculated via the following relation [105, p. 366]:

$$T_D = \frac{1}{2}\rho v_B^2 C_D A\left(c_p - c_g\right) \tag{31}$$

where $\left(c_p - c_g\right)$ is the scalar distance between the centre of aerodynamic pressure and the centre of gravity [105, p. 366]. Assuming a maximum distance of 0.02 (as with the vector case in Equation (30)), and assuming the equation is maximised when orbital velocity is maximised with

$$v_B = 7.67 \times 10^3 \text{ m/s}$$

then the maximum aerodynamic torque value is obtained from Equation (31) as:

$$T_D = 7.19 \times 10^{-8} \text{ Nm}$$

This scalar value is equal in magnitude to the value of 7.19×10^{-8} Nm obtained from the inertially-fixed vector case, implying the vector equation as implemented in the simulator to be valid. To further validate the aerodynamic drag calculations within the simulator, the predicted maximum aerodynamic value of 7.19×10^{-8} Nm was compared with the maximum aerodynamic drag torque was reported for the 2U, ISS-orbiting AntelSat as 2.60×10^{-8} Nm and as 1.34×10^{-7} for the 1U LEO COMPASS-1 [106] [107]. Table 8 compares parameter values used in their respective worst-case torque computations using Equation (31) to those used in the torque computation for the 2U prism in the simulator.

Table 8: Parameter Comparison for Aerodynamic Drag Torque Validation

Parameter	2U Prism	AntelSat [106]	COMPASS-1 [107]
C_D	2.25	2.2	2.2
v (m/s)	7665	7558	7558
ρ (kg/m^3)	2.72×10^{-12}	1×10^{-12}	6.2×10^{-12}
A (m^2)	0.02	0.02041	0.1
$\left(c_p - c_g\right)$ (m)	0.02	0.02	0.02

The worst-case torque obtained for AntelSat is of the same order of magnitude as that of the 2U prism, which is expected given the similarity between their parameter values. The worst-case projected area of COMPASS-1 reported as 0.1 m^2 is one order of magnitude higher than both the 2U prism and AntelSat, and it follows that its aerodynamic drag torque is one order of magnitude higher as well. Thus, the implementation of the torque equation within the simulator can be considered valid.

For the actual model of the satellite, the simplified geometries for the nominal projected area (area perpendicular to the satellite's direction of travel when the satellite is nadir-pointing) and the worst-case projected area are shown in Figure 17 and Figure 18. The simulator assumes a constant projected area, and the user can select between these two values in the simulator's driving routine depending on the desired strength of the aerodynamic drag torque applied in the simulation. Results shown in this thesis assume the Figure 18 case.

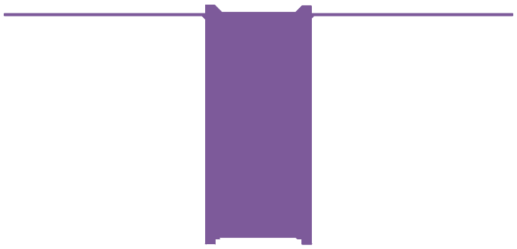


Figure 17: Nominal Projected Area (+X Projection) where $\vec{n}_A = [1 \quad 0 \quad 0]$ and $A = 0.02208\ \text{m}^2$

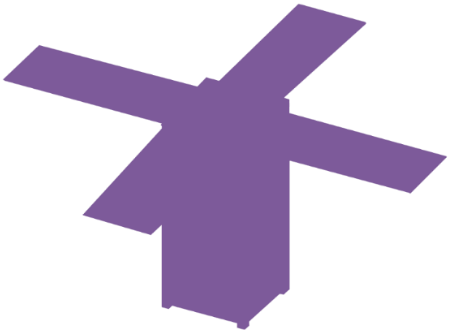


Figure 18: Worst-Case Projected Area (Trimetric Projection) where $\vec{n}_A = [0.75 \quad 0.45 \quad 0.47]$ and $A = 0.05721\ \text{m}^2$

3.3.5.3 Solar Radiation Pressure

The solar radiation pressure (SRP) vector and scalar torques T_{SRP} are calculated via the following expression [105, p. 366]:

$$T_{SRP} = \frac{F_s}{c} A(1+q)\cos(l)\,(c_{ps} - c_g) \tag{32}$$

where $F_s = 1367\ \text{W/m}^2$ is the solar constant (at neither solar maximum nor minimum), $c = 3 \times 10^8\ \text{m/s}$ is the speed of light, A is the exposed area of the satellite, $q = 0.6$ is the reflectance factor (typically 0.6-2.2, with 0.6 selected for the simulator [105, p. 366]), l is the angle of incidence of Sunlight, and $(c_{ps} - c_g)$ is the distance between the centre of solar pressure and the centre of gravity of the satellite. To utilise the body frame Sun vector in lieu of the angle of incidence of Sunlight (eliminating the need for an additional calculation), the SRP torque can be rewritten using the cross product of $(c_{ps} - c_g)$ and the unit vector from the satellite to Sun in BF $\boldsymbol{u}_{r_B}$:

$$\boldsymbol{T}_{SRP} = \frac{F_s}{c} A_s(1+q)\left[\left(\boldsymbol{c}_{ps} - \boldsymbol{c}_g\right) \times \boldsymbol{u}_{r_B}\right] \tag{33}$$

Unit vector $\boldsymbol{u}_{r_B}$ is determined from the reference vector from Earth to Sun $\boldsymbol{r}_{\oplus\odot_I}$ and the orbital radius in ECI $\vec{r}_I$, rotated into BF by q_{BI}:

$$\boldsymbol{u}_{r_B} = q_{BFI} * \frac{\boldsymbol{r}_{\oplus\odot_I} - \boldsymbol{r}_I}{\left\|\boldsymbol{r}_{\oplus\odot_I} - \boldsymbol{r}_I\right\|}$$

The greatest distance between the satellite's centre of mass and centre of solar pressure $\left(c_{ps} - c_g\right)$ is 0.1 m directed along the BF +Z axis; this assumes c_{ps} is centred on the top face of the satellite at the maximum surface distance possible given its geometry.

The magnitude and vector components of the SRP torque over one orbit for an inertially-fixed 2U prism model are found to be near constant. This result is expected given that the distance between Earth and Sun changes little over the duration of an orbit, and the relative magnitude of the orbital radius in comparison to its magnitude results in little variation in the direction of $\boldsymbol{u}_{r_B}$ over a single orbit. Therefore, the torque calculated from the cross product of this marginally changing vector with a fixed $\left(c_{ps} - c_g\right)$ vector should remain near-constant. The maximum SRP torque obtained for the inertially-fixed case is:

$$\boldsymbol{T}_{SRP} = \begin{bmatrix} 7.88 \times 10^{-9} \\ -1.18 \times 10^{-8} \\ 0 \end{bmatrix} \text{Nm}$$

$$T_{SRP} = \|\boldsymbol{T}_{SRP}\| = 1.42 \times 10^{-8}\ \text{Nm}$$

Figure 19 plots the magnitude of the SRP torque as a function of time over one orbit for a nadir-pointing 2U prism while Figure 20 plots its vector components in the BF frame. Though the same fixed $\left(c_{ps} - c_g\right)$ vector was defined, the torque adopts a cyclic profile as the BF axes rotate in orbit. The direction and magnitude of the torque itself do not change significantly versus the inertial case attesting again to the influence of the magnitude of the Earth-Sun vector on the SRP torque's profile.

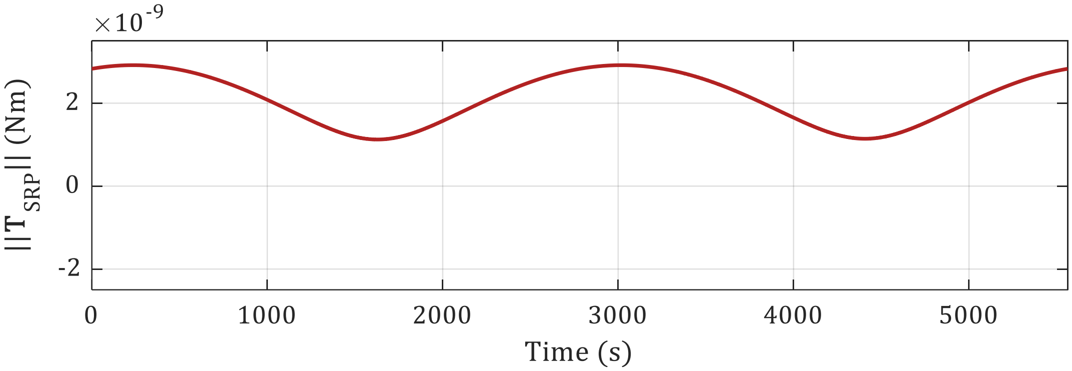


Figure 19: Solar Radiation Pressure Disturbance Torque Magnitude Over One Orbit for 2U Prism Model – Nadir-Pointing Attitude

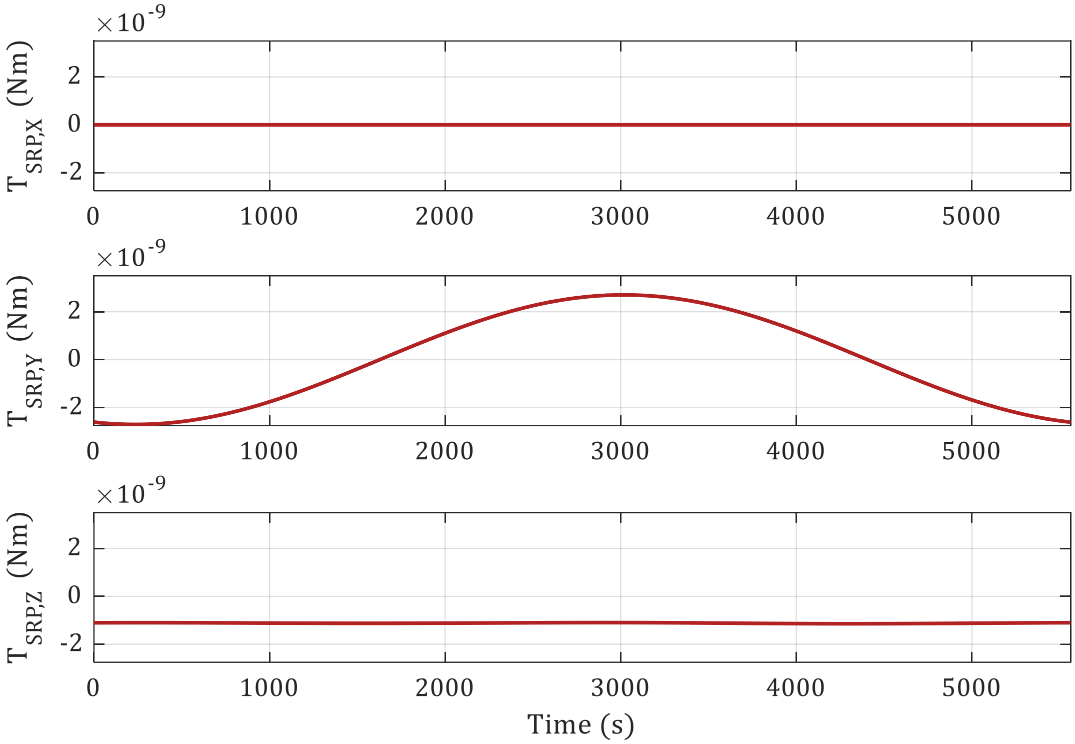


Figure 20: Solar Radiation Pressure Disturbance Torque Components in BF Over One Orbit for 2U Prism Model – Nadir-Pointing Attitude

For the nadir-pointing case, the maximum SRP torque was obtained as follows and is close in magnitude to that of the inertial case:

$$\boldsymbol{T}_{SRP} = \begin{bmatrix} 5.71 \times 10^{-9} \\ -1.34 \times 10^{-8} \\ 0 \end{bmatrix} \text{Nm}$$

$$T_{SRP} = \|\boldsymbol{T}_{SRP}\| = 1.46 \times 10^{-8}\ \text{Nm}$$

For the scalar case, assuming the equation is maximised when $\cos(l) = 1$ (Sun is directly incident on the satellite) and $(c_{ps} - c_g) = 0.1$ m, the SRP torque is calculated to be equivalent to that obtained for the nadir-pointing case:

$$T_{SRP} = 1.46 \times 10^{-8} \text{ Nm}$$

The difference in maxima between the inertial and nadir-pointing cases is attributable their respective orbits – the nadir-pointing data was obtained 5 orbits after the inertial case, resulting in a slight difference in the direction of the Earth-Sun vector increasing the magnitude of its cross product with the constant $(c_{ps} - c_g)$ vector.

For the 3U CINEMA CubeSat, the SRP torque was obtained as 2.48×10^{-9} Nm using equivalent values of F_s and q under the same assumption that the maximum occurs when the Sun's angle of incidence is zero [108, p. 24]. A projected area of 0.034 m^2 and a $(c_{ps} - c_g)$ distance of 0.01 m were chosen as its worst-case parameters [108, p. 24]. The area is larger than that used for 2U prism by a factor of 1.7 while the $(c_{ps} - c_g)$ distance was less by a factor of 0.1, implying that the CINEMA torque should be approximately 0.17 times smaller than that obtained for the 2U prism – which was shown to hold true:

$$(1.46 \times 10^{-8})(0.17) = 2.48 \times 10^{-9} \text{ Nm}$$

and thus, the implementation of the vector SRP torque can be considered valid.

3.3.5.4 Residual Magnetic Dipole

The vector residual magnetic dipole moment torque is defined [12, p. 149]:

$$\boldsymbol{T_M} = \boldsymbol{D} \times \boldsymbol{B} \tag{34}$$

where D is the residual dipole moment of the satellite and B is Earth's magnetic field expressed in BF in units of tesla. The torque is calculated assuming a typical dipole moment value of 0.01 Am^2 presumed to lay along the +Z direction. The maximum torque value is shown below, which is expected to occur when the residual magnetic field vector is perpendicular to the local magnetic field vector.

$$\boldsymbol{T_M} = \begin{bmatrix} 3.15 \times 10^{-7} \\ -3.35 \times 10^{-7} \\ 0 \end{bmatrix} \text{ Nm}$$

The scalar residual magnetic dipole moment torque is defined as [105, p. 366]:

$$T_M = DB \tag{35}$$

The magnitude over one orbit for the residual dipole moment torque is plotted in Figure 21, with its components plotted individually in Figure 22.

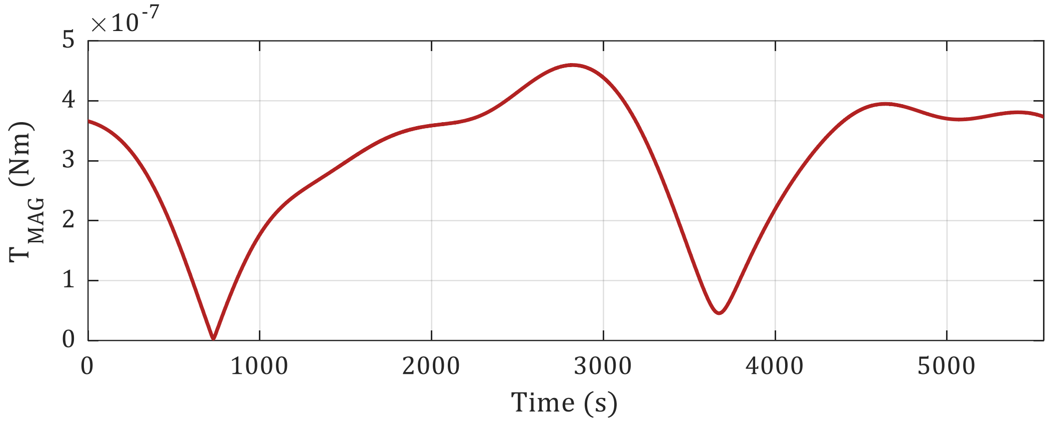


Figure 21: Residual Dipole Moment Disturbance Torque Magnitude Over One Orbit for 2U Prism Model

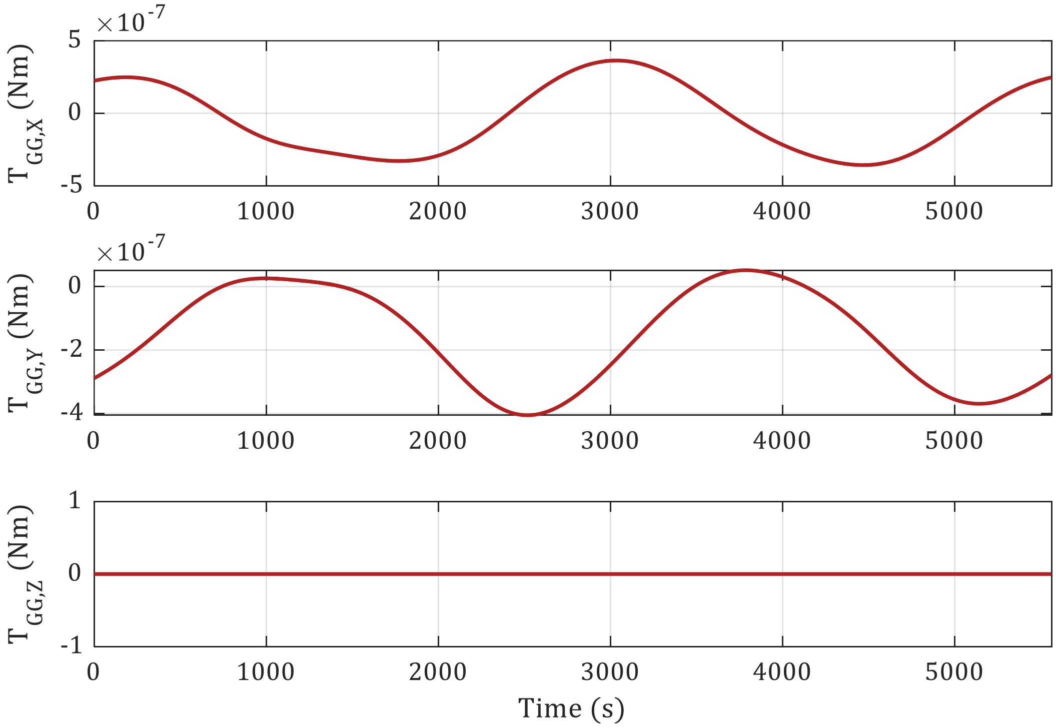


Figure 22: Residual Dipole Moment Disturbance Torque Components Over One Orbit for 2U Prism Model

The torque is obtained from the product of the maximum magnitude magnetic field for the current orbit and the scalar dipole moment value 0.01 Am^2, equivalent to the norm of the above maximum torque vector:

$$T_M = 4.60 \times 10^{-7} \text{ Nm}$$

The value of residual dipole torque depends solely upon the value of the satellite's residual magnetic dipole moment (in units of Am^2) and the value of the local magnetic field (in units of T). Worst-case values reported for the CADRE and COMPASS-1 satellites were identical in magnitude and near-identical in value at 4×10^{-7} Nm and 4.587×10^{-7} Nm, respectively [109, p. 17] [107, p. 43]. The COMPASS-1 value is obtained using the relation $2M/R^3$ (where $M = 7.96 \times 10^{15}\ \mathrm{T/m^3}$ is Earth's magnetic moment and R is the orbit radius) to calculate Earth's magnetic field – an approximation of scalar field strength as an "ideal dipole" [107, p. 43]. For CADRE, a similarly simplified model is used with an additional factor accounting for magnetic latitude [109, p. 17]. Geomagnetic field values, model simplicity notwithstanding, fall on the order of 10^{-5} T, and the resultant order of torque maxima depends solely upon the order of magnitude of the residual dipole moment.

The following chapter develops the sensor models and attitude determination algorithms that compare their output against the reference values described in this chapter.

Chapter 4: Attitude Determination

The attitude determination system of the LORIS ADCS comprises eighteen photodiode Sun sensors and one inertial measurement unit (IMU) including one three-axis gyroscope and one three-axis digital magnetometer. Quaternion attitude is estimated from these sensor measurements via the q-Method, and the estimate and its sensor measurements are improved using an extended Kalman filter whose output provides the basis for the satellite's pointing controller. Sensor models are based on experimentally obtained noise parameters to obtain as accurate an assessment as possible of obtainable satellite attitude and of the performance of the pointing controller.

Noise is generated in the simulator using Simulink Band-Limited White Noise blocks, each of which requires a noise power input to be specified. Each sensor has a sample rate t_s of 1 second via the following relation where NP is noise power and σ^2 is noise variance. In accordance with the approximation within the Band-Limited White Noise block, it is assumed that the required noise power input is equal to the noise variance for each sensor case [110].

$$NP = \sigma^2 t_s \tag{36}$$

4.1 Sun Sensors

Sun sensor photodiodes measure incident sunlight as a current intensity, with intensity presumed directed along the sensor normal. With each normal fixed relative to the BF coordinate frame, the vector between the satellite and Sun can be obtained using the relative intensity readings of multiple Sun sensors. The simulated measure of intensity was based on the OSRAM SFH 2430-Z photodiode selected for use onboard the LORIS satellite [111].

A threshold value for incident light intensity was sought to improve the accuracy of the Sun vector estimation. Setting a limit on the intensity that constitutes a sensor as being in view of Sun serves to eliminate secondary sources of light – primarily Earth and Moon albedo – thereby ensuring that only the most illuminated sensors are used to estimate the location of the Sun vector. Assuming the angle of incident light φ is measured relative to the sensor normal, Figure 23 compares the directional characteristic plot from the SFH 2430-Z datasheet with the approximation $\cos\varphi$ used in the simulator to relate intensity and angle of incidence, with a root-mean-square (RMS) error of 0.0026 or 0.26% between the two [111]. For light incident at 60°, the spectral sensitivity of the sensor drops to 50%. As Earth's albedo commonly reaches a maximum of 30-40%, a threshold angle of 60°

corresponding to a spectral sensitivity of 0.5 was deemed sufficient to ensure the sensors could distinguish between Earth albedo and Sunlight [12, p. 154].

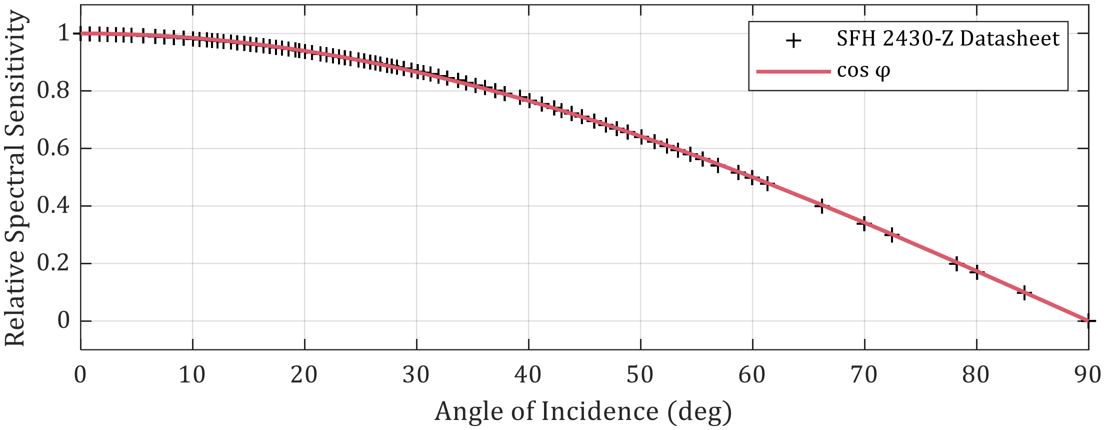


Figure 23: Comparison between SFH 2430-Z Relative Spectral Sensitivity and $\cos\varphi$ versus Light Incident Angle Showing 0.26% Agreement [111]

Photodiode noise characteristics were obtained experimentally. To account for nonlinearities resulting from inconsistencies in ambient light during the experiment, only a small, near-linear portion of the experimental data was used to identify variance and noise power. This experimental data is shown in the first subplot of Figure 24. The equivalent white Gaussian noise power for the photodiodes NP_{PD} was identified from the variance ${\sigma_{PD}}^2$ of the recorded rates in rad/s for a sample rate of 1 s according to Equation 36.

$$NP_{PD} = {\sigma_{PD}}^2 = 1.13 \times 10^{-4}\ \mathrm{V}^2$$

The quantisation interval for the noise (placed into a Quantizer Simulink block) was experimentally identified as 0.0098 V. Subplots 2 and 3 in Figure 24 compare the generated white Gaussian noise and quantised white Gaussian (WG) noise with the experimental data. The variance of the resulting quantised data is $1.23 \times 10^{-4}\ \mathrm{V}^2$; a 9% error between it and the experimental data.

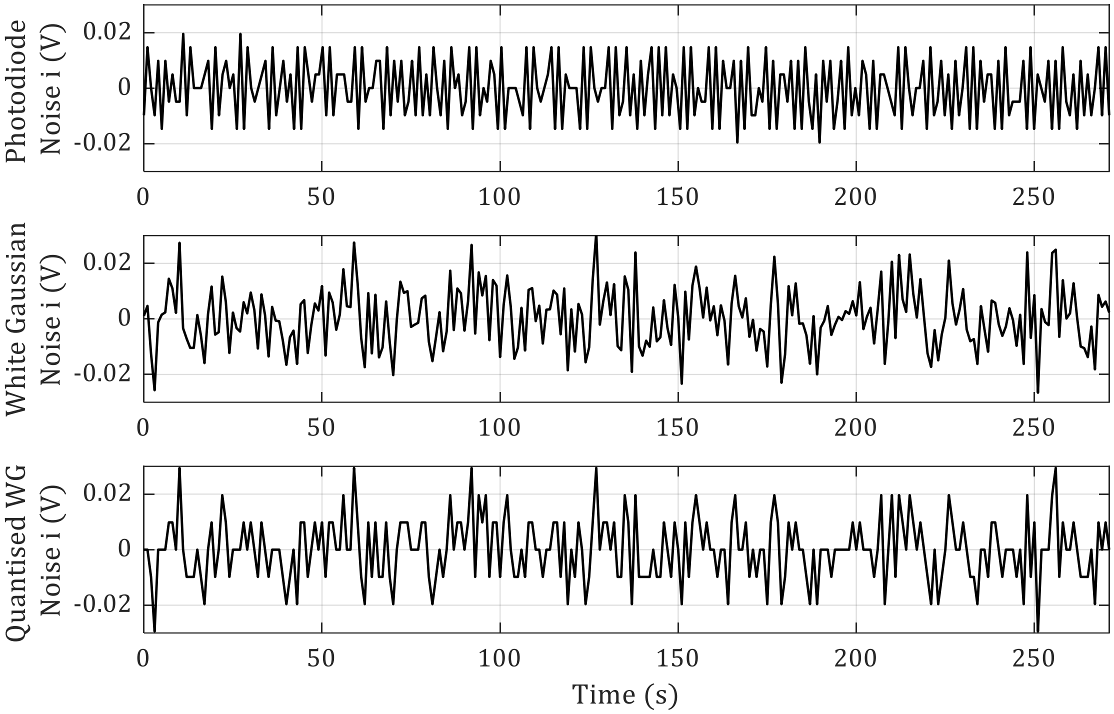


Figure 24: Comparison between Experimentally Obtained Photodiode Noise and Generated White Gaussian Noise

4.1.1 Sensor Placement

To further maximise Sun vector availability, three photodiodes are arranged in a triad on each face of the satellite illustrated in Figure 25, tilted 60 degrees from the face itself with 120 degrees separating the three in the set. Each sensor has a conic field of view of 120° imposed by an aluminium "mask" to accommodate the aforementioned 60° intensity threshold. In addition to limiting the field of view of the Sun sensors to the desired 120°, the mask acts as a mounting fixture that sets the photodiodes at the correct angles and offers some protection to the sensors during CubeSat assembly, testing and launch when the solar panels are undeployed.

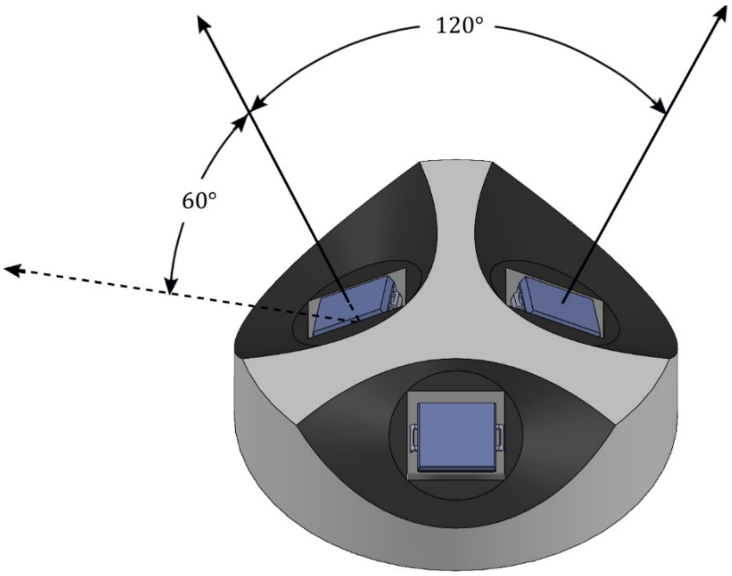


Figure 25: SolidWorks Sun Sensor Assembly [112]

4.1.2 Sun Vector Determination

For a Sun vector to be estimated, at least three Sun sensors must be illuminated by Sun – limiting the availability of the Sun vector to periods of the orbit outside eclipse. Additionally, in certain orientations, the Sun vector may not be estimable due to shadows cast on the sensors by the solar panels. Figure 26 demonstrates the shade effect of the solar panels with an arbitrary, nearby light source. For the sample orientation and Sun location shown in the figure, nine sensors (depicted as circles on the top and two sides of the CubeSat) have the potential to see Sun if not for the presence of the solar panels. Of the nine sensors, only one is completely in shade due to solar panel shade – denoted in red – leaving eight sensors from which the Sun vector can be determined. The remaining sides of the satellite are completely in shadow, and any residual illumination detected in their sensors can be assumed to be Earth's albedo.

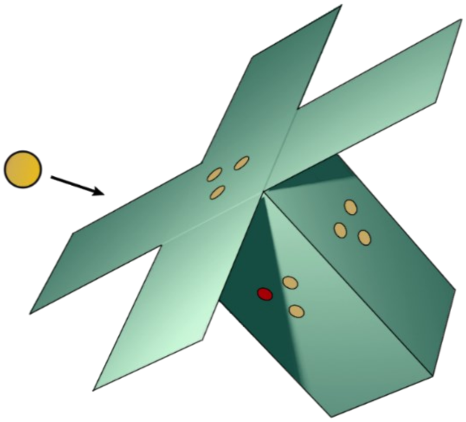


Figure 26: Satellite Orientation Showing Effect of Shadowing

The normal directions for each of the eighteen Sun sensors are concatenated into an 18×3 matrix $\boldsymbol{H}$ and remain constant in the BF frame. In the simulator, the illumination I of each sensor (indices $i = [1 \dots 18]$) is equivalent to the dot product of the sensor normal $\boldsymbol{n}$ with the incoming reference Sun vector $\boldsymbol{S}$ defined in Section 3.3.3, transformed into BF:

$$I_i = \boldsymbol{n_i} \cdot \boldsymbol{S}_{BF}$$

The eighteen illumination readings, each ranging between 0 and 1, are concatenated into the 18×1 matrix $\boldsymbol{y}$. Before adding the experimentally determined quantised sensor noise to each illumination value, the noise was scaled by 5 V (to correctly scale the sensor noise; the sensor full-scale reading is 5 V, while the illumination readings I_i used in the simulator have a full-scale reading of 1). A threshold value of 0.5 is applied to each sensor to account for Earth's albedo – assuming any nonzero illumination reading less than 0.5 in magnitude to be light reflected off Earth. Sensors whose illumination values in the simulator pass this threshold are flagged as "potentially" seeing Sun and are further checked against a line-plane intersection algorithm to determine if they lie in the shade of the solar panels. Sensors that cannot see Sun are removed from both the $\boldsymbol{H}$ and $\boldsymbol{y}$ matrices, and the following relation is used to solve for the Sun vector $\boldsymbol{s}$, making use of the pseudoinverse of $\boldsymbol{H}$ [113, p. 59]:

$$\boldsymbol{y} = \boldsymbol{Hs}$$

$$\boldsymbol{s} = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{y} \tag{37}$$

If the Sun vector cannot be determined, it is set to a zero value of $\boldsymbol{s} = [0 \quad 0 \quad 0]^T$. Over 100 orbits excluding time spent in eclipse (342,866 seconds), only one sensor could see Sun 0.10% (351 seconds) of the total duration, only two sensors could see Sun 0.32% (1,056 seconds) of the total duration, and only three sensors could see Sun 4.84% (16,591 seconds) of the total duration – implying that shadowing presents issues for Sun vector determination less than 1% of the total time the satellite is in orbit.

4.2 Gyroscope Model

The IMU gyroscope measures the satellite's body angular rates in units of rad/s. The first row of subplots in Figure 27 shows experimental rate vector data obtained for a BNO055 IMU gyroscope. Gyroscope noise powers $\boldsymbol{NP}_G$ were identified from the variance of these rates ${\boldsymbol{\sigma}_G}^2$ in rad/s for a sample time of 1 s, according to Equation 36:

$$\boldsymbol{NP}_G = {\boldsymbol{\sigma}_G}^2 = [2.18 \times 10^{-6} \quad 2.99 \times 10^{-6} \quad 1.30 \times 10^{-6}]\ (\text{rad/s})^2$$

with an experimentally determined quantisation interval of 0.0011 rad/s. Placing these values into their respective Band-Limited White Gaussian Noise and quantizer Simulink blocks produced the results in the remaining Figure 27 subplots. Variances of the resulting quantised noise data are listed below, with errors of 6%, 4% and 9% relative to the experimental data.

$$\boldsymbol{\sigma^2} = [2.32 \times 10^{-6} \;\; 3.09 \times 10^{-6} \;\; 1.42 \times 10^{-6}]\ (\mathrm{rad/s})^2$$

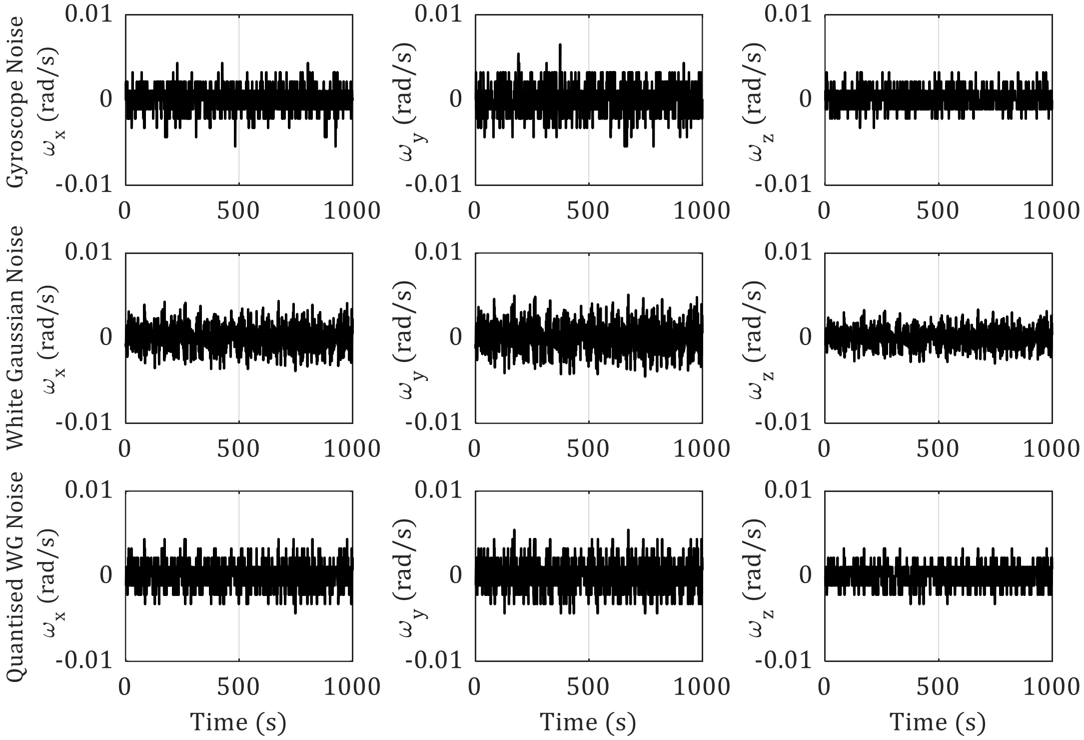


Figure 27: Comparison between Experimentally Obtained BNO055 Gyroscope Noise and Generated White Gaussian Noise

A "random walk" model of gyroscope drift as used by Yang [77] was then added to this noisy sensor data to simulate the zero-rate offset and temperature/voltage relative angular velocity drift specified for the BNO055 sensor [114]. Because the experimentally collected data did not exhibit any noticeable drift over a several hour sampling period, a bias value of 1×10^{-3} was empirically selected. This value was supplied as a noise power to a Band Limited White Noise block whose output is integrated to produce a noisy drifting signal. This drifting value is summed with the above sensor noise to the Simscape angular rate output to emulate a gyroscope reading. Figure 28 shows this added drift isolated over 100 seconds. This estimate likely exceeds a realistic value of drift – the BNO055 datasheet suggests $\pm$ 3 deg/s maximum at near-zero angular rates – but can be considered a worst-case assumption.

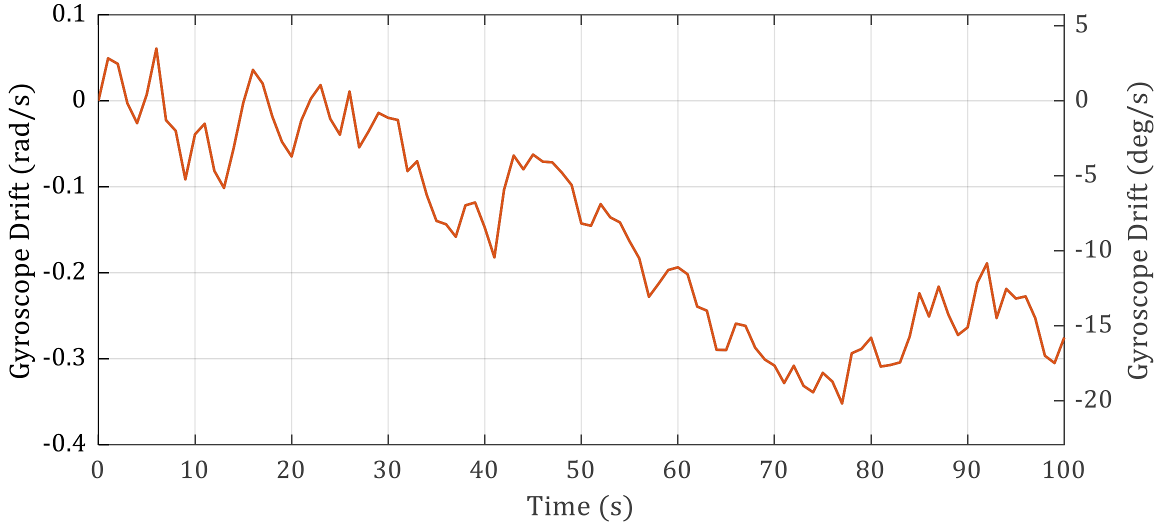


Figure 28: Gyroscope Drift

4.3 Magnetometer Model

The three-axis magnetometer measures the strength of the local magnetic field (in units of microtesla) with respect to the BF frame axes. Two magnetometers are modelled in the simulator – one from an MPU-9250 IMU and one from the same BNO055 IMU described in Section 4.2. Though the LORIS ADCS initially included two separate magnetometers for redundancy, the hardware plan for the system has since been revised to only include one. Regardless, both magnetometers are modelled in the simulator and either can be used at will. In the simulator, both noise power values are scaled by a factor of 10^6 to obtain units of tesla.

The experimental data in units of microtesla obtained for the MPU-9250 magnetometer (shown in the first row of subplots in Figure 29) was collected with a sample time of 0.0080 seconds. Variances of the recorded magnetic fields were determined as:

$$\boldsymbol{\sigma}_{MPU}{}^2 = [0.56 \quad 0.59 \quad 0.56]\ \mu\text{T}^2$$

Noise powers were then identified according to Equation 36 as:

$$\boldsymbol{NP}_{MPU} = [0.0045 \quad 0.0047 \quad 0.0044]\ \mu\text{T}^2$$

with an experimentally determined quantisation interval of 0.7242 µT. Variances were obtained for the quantised white Gaussian noise shown in the Figure 29 bottom row of subplots as follows, with errors of 9%, 10% and 9% versus the experimental data:

$$\boldsymbol{\sigma}^2 = [0.61 \quad 0.65 \quad 0.61]\ \mu\text{T}^2$$

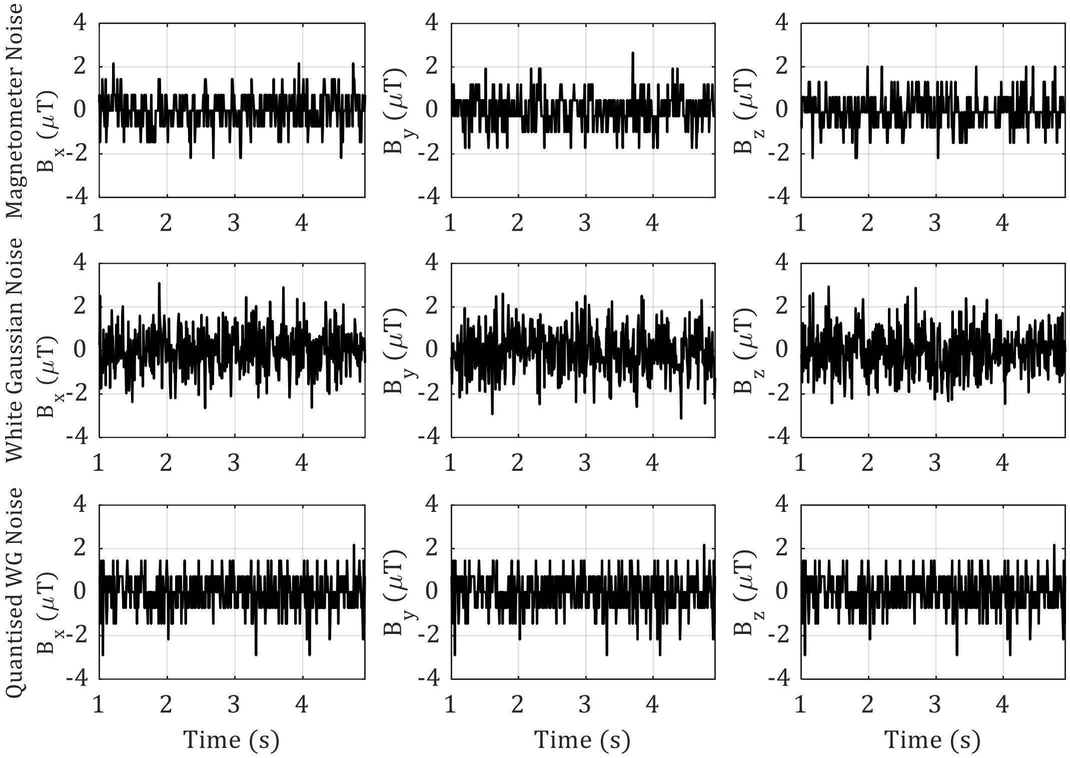


Figure 29: Comparison between Experimentally Obtained MPU-9250 Magnetometer Noise and Generated White Gaussian Noise

Via Equation 36 for the BNO055 magnetometer, Noise powers were identified from the variance of the recorded field in microtesla with a sample time of 1 second:

$$\boldsymbol{NP}_{BNO} = \boldsymbol{\sigma}_{BNO}{}^2 = [0.39 \quad 0.21 \quad 0.37]\ \mu\text{T}^2$$

with a quantisation interval of 0.44 μT. The quantised white Gaussian noise in Figure 30 has the following variances, with errors of 5%, 9%, and 6% in comparison to the experimental data:

$$\boldsymbol{\sigma^2} = [0.41 \;\; 0.23 \;\; 0.39]\ \mu\text{T}^2$$

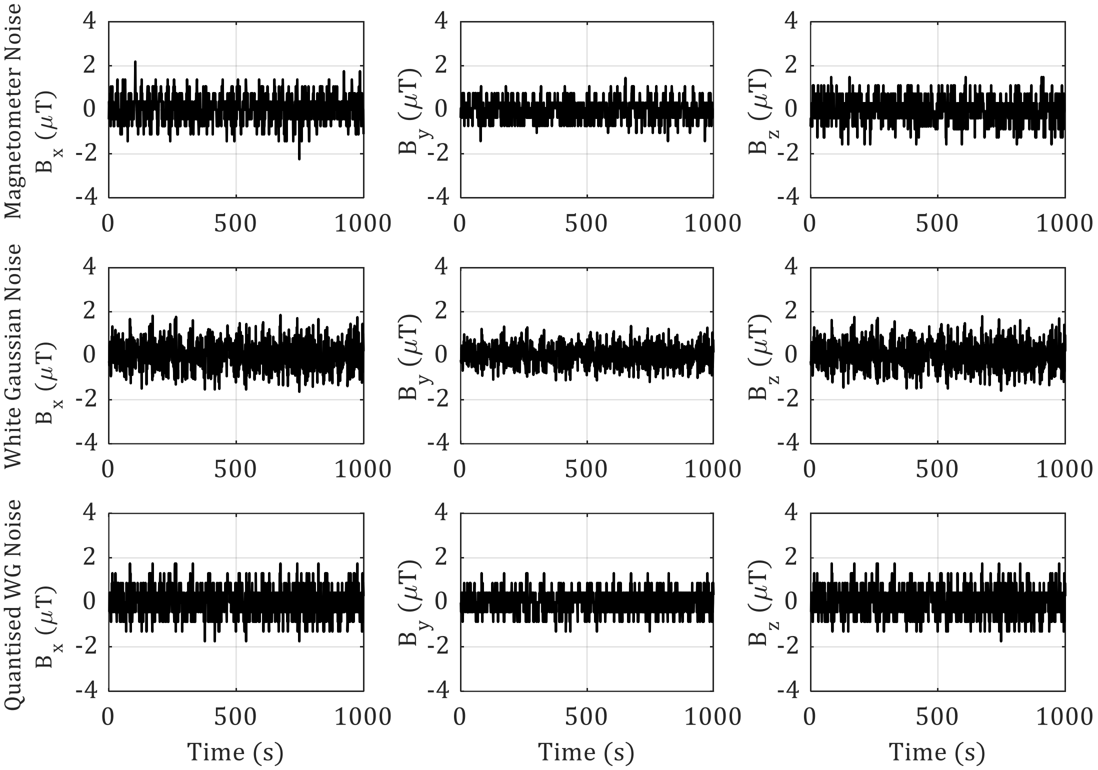


Figure 30: Comparison between Experimentally Obtained BNO055 Magnetometer Noise and Generated White Gaussian Noise

4.4 Attitude Determination - q-Method Algorithm

The q-Method for attitude determination implemented in the simulator is analogous to the QUEST (QUaternion ESTimation) algorithm first proposed by Shuster in 1981 [76]. The algorithm compares sets of nonparallel unit vectors defined in two reference frames – “reference” vectors $\boldsymbol{V}_i$ and “observation” vectors (from sensor readings) $\boldsymbol{W}_i$ [76, p. 71]. Shuster’s QUEST algorithm seeks an “optimal attitude matrix” $\boldsymbol{A}_{opt}$ to maximise the gain function $g(\boldsymbol{A})$ [76, p. 73]:

$$g(\boldsymbol{A}_{opt}) = \sum_{i=1}^{n} a_i \boldsymbol{W}_i^T \boldsymbol{A} \boldsymbol{V}_i \tag{38}$$

An “attitude profile matrix” $\boldsymbol{B}$ is defined below, where a_i are positive (unity) weights [76, p. 73]:

$$\boldsymbol{B} = \sum_{i=1}^{n} a_i \boldsymbol{W}_i \boldsymbol{V}_i^T \tag{39}$$

Based on $\boldsymbol{B}$, Shuster defines the quantities $\boldsymbol{\sigma}$, $\boldsymbol{S}$, and $\boldsymbol{Z}$ [76, p. 73]

$$\boldsymbol{\sigma} = \mathrm{tr}(\boldsymbol{B}) \tag{40}$$

$$\boldsymbol{S} = \boldsymbol{B} + \boldsymbol{B}^T \tag{41}$$

$$\boldsymbol{Z} = \sum_{i=1}^{n} a_i(\boldsymbol{W}_i \times \boldsymbol{V}_i) \tag{42}$$

as components in the coefficient matrix $\boldsymbol{K}$ for the "bilinear form" of the gain function, obtained by redefining Equation 38 in quaternion terms [76, p. 73]:

$$g(\bar{q}) = \bar{q}^T \boldsymbol{K} \bar{q} \tag{43}$$

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{S} - \boldsymbol{\sigma} \boldsymbol{I}_3 & \boldsymbol{Z} \\ \boldsymbol{Z}^T & \boldsymbol{\sigma} \end{bmatrix} \tag{44}$$

For the quaternion to satisfy a constraint $\bar{q}^T\bar{q} = 1$, the derivative of Equation 43 can be written according to the method of Lagrange multipliers [76, p. 73] as

$$g'(\bar{q}) = \bar{q}^T \boldsymbol{K} \bar{q} - \lambda \bar{q}^T \bar{q} \tag{45}$$

from which a stationary value is obtained when [76, p. 73]

$$\boldsymbol{K} \bar{q} = \lambda \bar{q} \tag{46}$$

The optimal attitude quaternion $\bar{q}$ that maximises Equation 43 is derived from the eigen decomposition of $\boldsymbol{K}$ as the eigenvector corresponding to the maximum eigenvalue.

The eigenvector takes the form of vector|scalar, and thus the output of the q-Method block is reordered to match the scalar|vector quaternion format used elsewhere in the simulator. Two vector sets are used in the simulator: the estimated Sun vector from Sun sensor readings in BF $\boldsymbol{S}_{BF}$, the reference Sun vector in ECI $\boldsymbol{S}_I$, the measured local magnetic field vector $\boldsymbol{B}_{BF}$ and the reference IGRF-12 ECI magnetic field vector $\boldsymbol{B}_I$.

This chapter outlined the components of the attitude determination system and the algorithms used to obtain attitude estimates from them. The next chapter uses these sensors to devise a B-dot detumbling controller capable of reducing the high angular rates of the satellite down to a range from which the pointing controller can be used within.

Chapter 5: Magnetic Detumbling Control

When the satellite is first launched into orbit, it will be unable to power on until its batteries are sufficiently charged, meaning it will orbit Earth uncontrolled for some duration of time. Throughout this period, the satellite is typically "tumbling" – spinning at an unknown potentially-high residual angular rate from launch. Once it becomes operational, the satellite will enter directly into its detumbling phase to reduce these residual launch body angular rates before it can be pointed to carry out its mission objective. Throughout its mission life, it may start to spin up again and require additional detumbling. This detumbling phase is handled entirely by a proportional B-dot controller acting on sensor inputs. Torqueing using three orthogonal magnetorquers and sensing using three-axis magnetometers must be cycled to reduce the influence of the magnetorquers on the magnetometers. This section covers the experimental determination of a feasible sensing/actuation duty cycle to fit within the desired 1 second sample time of the sensors, and then outlines the design and validation process of the B-dot detumbling controller in the simulator.

5.1.1 Magnetorquer Model

The LORIS satellite ADCS includes three NewSpace Systems NCTR-M002 magnetorquer rods arranged orthogonally along its BF axes. Each magnetorquer rod consists of a magnetically permeable metallic core of diameter 0.9 cm wound with 2000 turns of 34-36 AWG copper magnet wire along 6 cm of its total length.

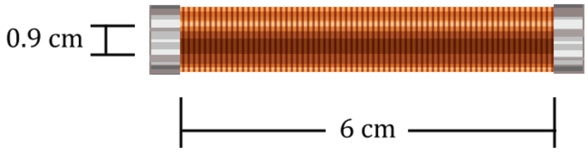


Figure 31: Magnetorquer Rod Key Dimensions

These torque rods are capable of generating a nominal magnetic dipole moment of 0.2 Am^2 for a current of 31 mA, up to a saturation magnetic dipole moment of 1.4 Am^2 for a current of 215 mA [115, p. 9]. The wire on the purchased magnetorquers was damaged, and thus each rod was rewired using the coil winder apparatus in Figure 32 with the same number of turns of a higher-gauge appropriately-insulated copper wire. Properties of the new wire and final magnetorquer assembly are compiled in Table 9.

Table 9: Rewired Magnetorquer Parameters

Parameter	Value
Material	Enamelled copper, polyester with polyamideimide overcoat
Wire Gauge	32 AWG
Wire Diameter	0.23622 mm (nominal)
Number of Turns	2000
Length of Wire (Approx.)	66.44 m
Rod Diameter	9.24 mm
Rod Cross-Sectional Area	6.71×10^{-5} m^2

Figure 32: Coil Winder Apparatus

For a typical solenoid electromagnetic coil – essentially analogous to a magnetorquer without a core – the scalar magnetic moment m is calculated via the following relation, where N is the number of turns of wire in the solenoid, I is the supplied current, and S is the cross-sectional area of the solenoid [116]:

$$m = NIS \tag{47}$$

The current passing through the magnetorquer was measured by connecting it to a 5 V power supply carried through a TB6612FNG motor driver. The voltage over the magnetorquer was measured at 5.07 V with a measured resistance value of 24.7 Ω and these values were used to estimate the maximum (steady state) current across the magnetorquer via Ohm's law:

$$I = \frac{V}{R}$$

$$I = 0.2053 \text{ A} = 205.3 \text{ mA}$$

Based on this current estimate falling within 10 mA of the datasheet saturation current of 215 mA, it can be assumed that the new wire generates magnetic dipole moments essentially equivalent to the $\pm 1.4 \text{ Am}^2$ magnetic dipole moment of the original. A saturation limit of $\pm 1.4 \text{ Am}^2$ is, therefore, imposed upon any magnetic moment signal in the simulator.

5.1.1.1 Time Constant Identification

To identify a feasible duty cycle to avoid interference between torqueing and sensing the local magnetic field, time constants for the start up and shutdown of the rewired magnetorquers were determined using an Arduino-based experimental setup with two components: a circuit driving the magnetorquer and a circuit for measuring its generated magnetic field with an IMU magnetometer.

The magnetorquer is powered by a constant 5 V source provided by the 5 V Arduino pin. Current travels from the Arduino through a TB6612FNG motor driver used to change the direction of the field the magnetorquer generates. These three components are wired as shown in Figure 33, and descriptions of all wire connections are compiled in Table 10. Specifying the motor driver to run "clockwise" defines terminal A01 as the voltage input, while "counter-clockwise" defines A02 as the voltage input. The corresponding wires on the magnetorquer are undifferentiated, and field direction depends entirely upon how the user connects the magnetorquer to the motor driver and subsequently orients the magnetorquer in the experiment. The setup is controlled using a MATLAB-created UI with functions from the MATLAB Support Package for Arduino Hardware used to write values to the digital pins and read values from the analogue pins.

Table 10: Motor Driver Interface Table

Motor Driver	Arduino	Description
VM	Vin	Motor power from barrel jack – unused
VCC	5V	Logic voltage and main 5 V source from regulated USB connection
GND	GND	Common ground
A01	A1	Motor A output 1, connected to one magnetorquer wire
A02	A2	Motor A output 2, connected to other magnetorquer wire
PWMA	D11	PWM speed control for A I/O channels (square wave output)
AI2	D13	Direction input to A channel
AI1	D12	Direction input to A channel
STBY	D8	Internal h-bridge standby input

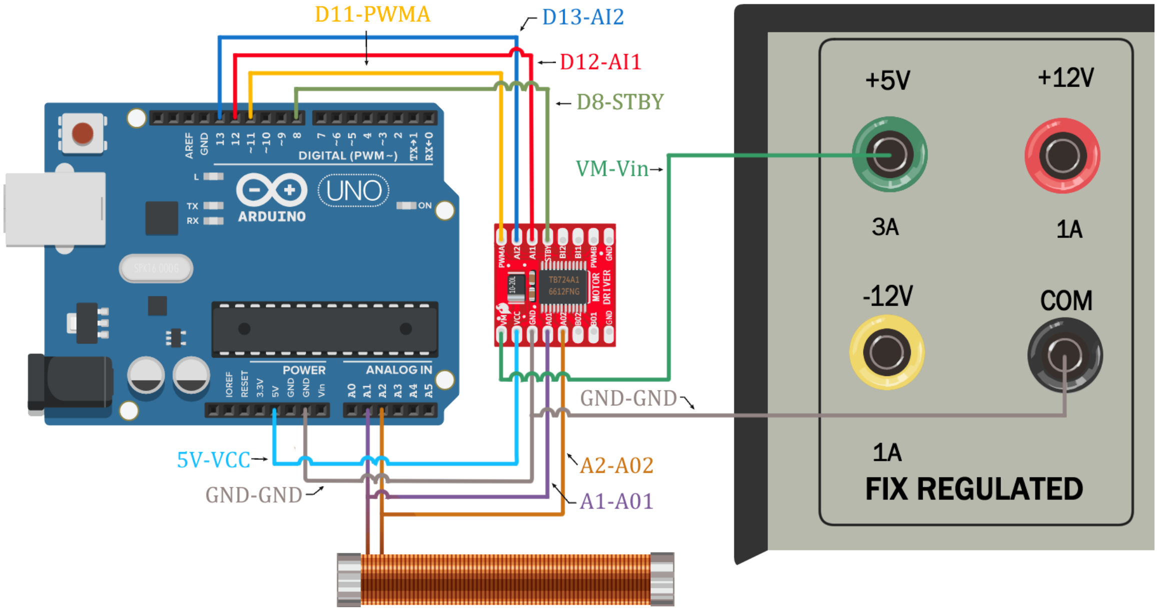


Figure 33: Magnetorquer Experimental Arduino Circuit Diagram

For the magnetometer setup, an MPU-9250 IMU breakout board is used to acquire magnetometer data with a maximum sample rate of 200 Hz and measurement range of ±4800 μT at its I2C interface. Table 11 compiles the wire connections between the IMU and Arduino shown in Figure 34.

Table 11: IMU Magnetometer Interface Table

IMU	Arduino	Description
SCL	A5	12C serial clock
SDA	A4	12C serial data output
VDD	3.3V	Main 3.3 V source from regulated USB connection
GND	GND	Common ground

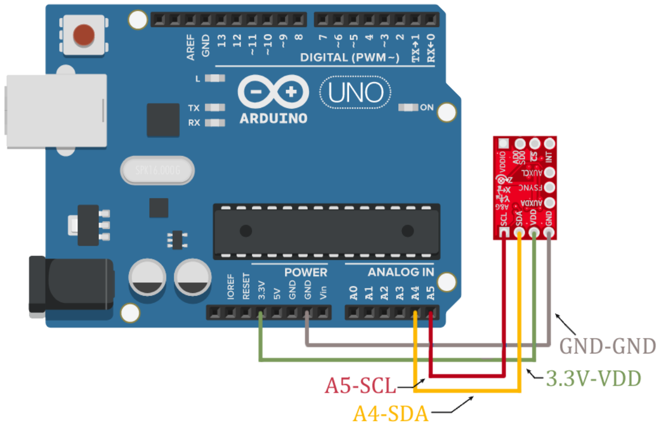


Figure 34: Magnetometer Experimental Arduino Circuit Diagram

The time constants for the start up and shutdown of the magnetorquer were determined using voltages read from an oscilloscope. To “Start” the torquers, the motor driver was commanded to run CW at a 100% duty cycle with current flowing through the magnetorquer from A01 to A02. To “Stop” the torquer, the motor driver was powered off to 0% duty cycle.

Raw oscilloscope voltage data is plotted in Figure 35 when the motor driver is commanded to start and stop. The torquer powers up quickly and maintains a constant A01 voltage of 5.25 V with A02 close to zero V. When the torquer is powered off, the input voltage from A01 is severed manifesting as a brief sign inversion on A02 when read by the oscilloscope.

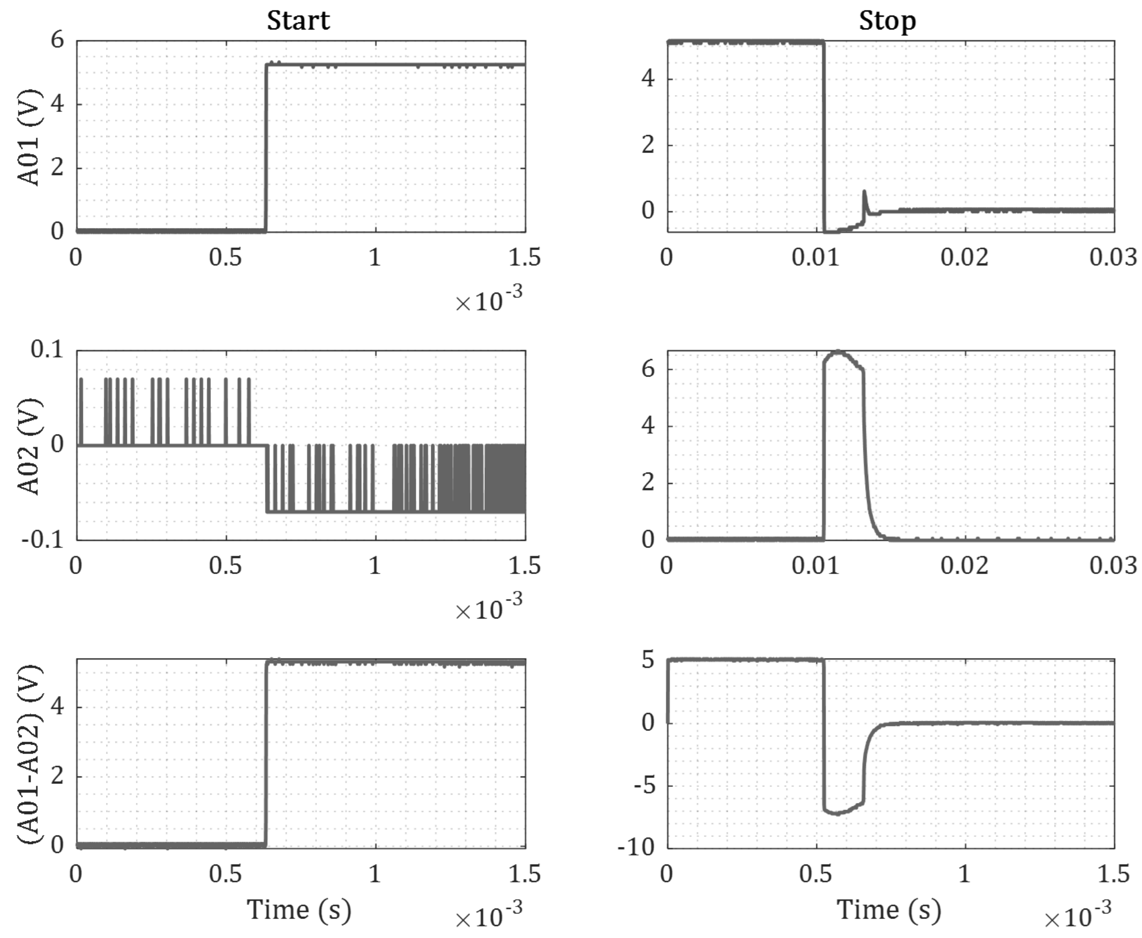


Figure 35: Individual Channel Voltages and Voltage Over Magnetorquer for Clockwise Motor Driver Command

For both start up and shutdown, the current over the magnetorquer was calculated via the following formula based on the resistance value (24.7 Ω) of the magnetorquer:

$$i = \frac{V_{A01} - V_{A02}}{24.7} \tag{48}$$

Exponential curves were fit to the start up and shutdown data according to the following relation, with a different gain K and time constant τ identified for each.

$$-Ke^{-t/\tau} \tag{49}$$

Figure 36 plots the experimental start up data against an exponential curve with empirically identified gain and time constant listed below. Zero seconds on the time-axis of the plot corresponds to the point when the magnetorquer was turned on.

$$K = 2.15 \times 10^{-1}\ \mathrm{A}$$

$$\tau = 7.00 \times 10^{-7}\ \mathrm{s}$$

$$-2.15 \times 10^{-1} e^{-t/7.00 \times 10^{-7}}$$

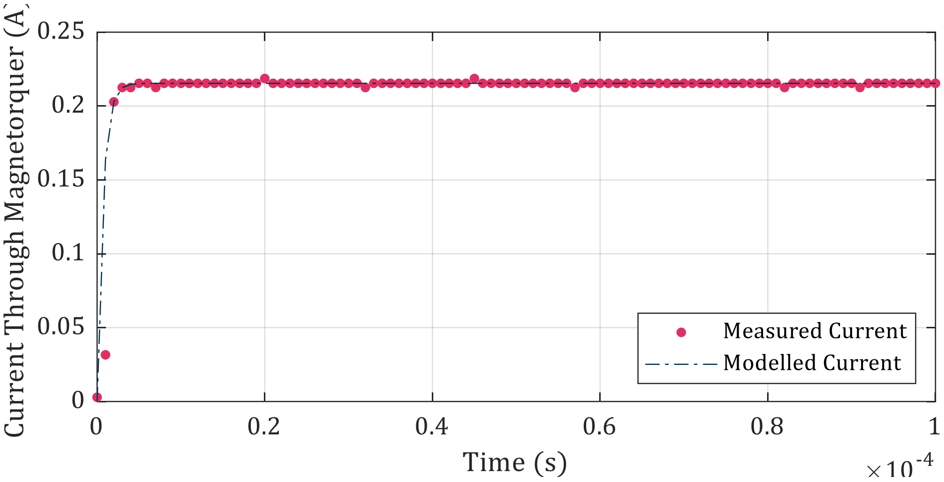


Figure 36: Start Up Current Over Magnetorquer Compared with Exponential Fit

The steady-state current maintained an average value of 2.15×10^{-1} A, with an exponential settling time (3τ criterion) of approximately 2.1×10^{-6} s.

Figure 37 plots the same for the experimental shutdown data with zero seconds on the time axis of the plot corresponding to the point when the magnetorquer was turned off. The gain and time constant were identified empirically for the portion of the plot after the initial dip in current as:

$$K = 2.63 \times 10^{-1}\ \mathrm{A}$$

$$\tau = 3.00 \times 10^{-4} \mathrm{s}$$

$$-2.63 \times 10^{-1} e^{-t/3.00 \times 10^{-4}}$$

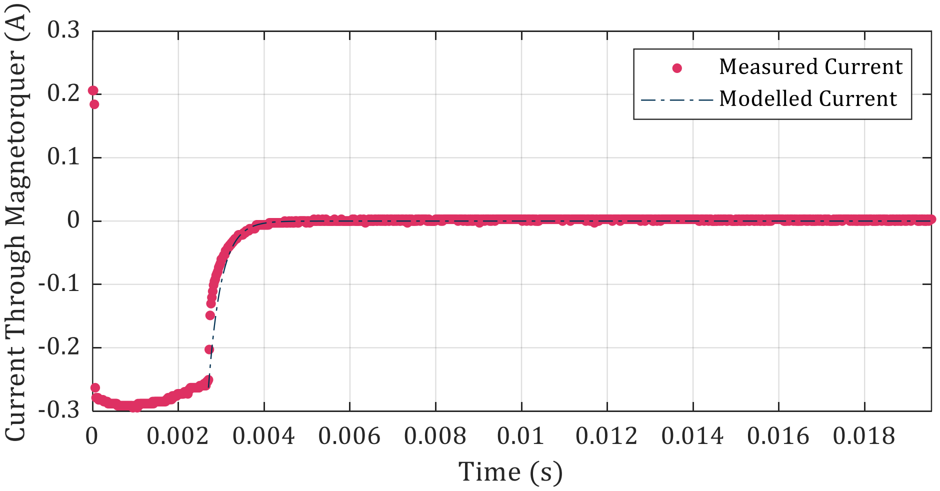


Figure 37: Shutdown Current Over Magnetorquer Compared with Exponential Fit

The settling time of the current (3τ criterion) is approximately 3.50×10^{-3} s including the initial negative spike, with a settling time of the exponential portion alone (3τ criterion) of 9.00×10^{-4} s.

The experimental start up and shutdown behaviour defines a 99.70% duty cycle for the magnetorquers based on a 1 second magnetometer sample time. This exact behaviour is not modelled in the simulator, and instead a conservative duty cycle of 90% is applied as a square wave to any magnetic input torque signals in Simulink. This estimate allows time for any residual magnetic field and consequent residual dipole moment to dissipate before another magnetic field measurement is taken.

5.2 B-Dot Algorithm

The B-dot algorithm describes a proportional controller that acts on the time-derivative (or rate of change) of the local magnetic field expressed in the body-fixed frame: B-dot or $\dot{\boldsymbol{B}}$ [117, p. 79]. This term is defined as the cross-product of the local magnetic field in tesla and the satellite's rotation rate in rad/s [117, p. 80]:

$$\dot{\boldsymbol{B}} = \boldsymbol{B} \times \boldsymbol{\omega} \tag{50}$$

This $\dot{\boldsymbol{B}}$ value is then utilised in a proportional control law to determine the desired magnetic dipole moment m in units Am^2 commanded to the magnetorquers. The scalar gain K is negative to apply a restoring moment about the desired axis to the rotating spacecraft [117, p. 79]:

$$\boldsymbol{m} = -K(\boldsymbol{B} \times \boldsymbol{\omega}) = -K\dot{\boldsymbol{B}} \tag{51}$$

Current is supplied to the magnetorquer to meet this desired moment. The resulting magnetic dipole moment from the magnetorquers generates an external control torque on the satellite according to the following equation in units of Nm [117, p. 79]:

$$\boldsymbol{T}_c = \boldsymbol{m} \times \boldsymbol{B} \tag{52}$$

All vectors in subsequent equations are expressed in the BF frame. B-dot control cannot be used in its generic form for accurate three-axis pointing control. Most typically, the B-dot algorithm is used during the initial detumbling phase of the satellite's orbit before control is switched over to a three-axis attitude controller.

As a first step towards validating the implementation of the B-dot algorithm in the simulator, the $\dot{\boldsymbol{B}}$ values obtained using Equation (50) were checked for the case when the satellite is rotating at a constant angular rate about its body-fixed Z-axis in an equatorial orbit.

The initial angular rates of the satellite were defined to be equivalent to one rotation about the Z axis per orbital period as follows:

$$\boldsymbol{\omega}_{ini} = \begin{bmatrix} 0 \\ 0 \\ \frac{2\pi}{T} \end{bmatrix} \text{rad/s}$$

Using Equation (50), the resulting expression $\dot{\boldsymbol{B}} = \boldsymbol{B} \times \boldsymbol{\omega}$ yields:

$$\dot{\boldsymbol{B}} = \boldsymbol{B} \times \boldsymbol{\omega} = \boldsymbol{B}^x \boldsymbol{\omega} = \begin{bmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ -B_y & B_x & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$\dot{\boldsymbol{B}} = \begin{bmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ -B_y & B_x & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \frac{2\pi}{T} \end{bmatrix}$$

$$\dot{\boldsymbol{B}} = \begin{bmatrix} \frac{2\pi}{T} B_y \\ -\frac{2\pi}{T} B_x \\ 0 \end{bmatrix}$$

The resulting relation shows that the X component of the $\dot{\boldsymbol{B}}$ vector should be equivalent to the Y component of the local magnetic field in BF scaled by the constant angular rate, while the Y component of the $\dot{\boldsymbol{B}}$ vector should be equal in magnitude but opposite in direction to the X component of the body-fixed local magnetic field vector scaled by the same angular rate. The simulation was run for 5562 seconds corresponding to one orbit. Figure 38 plots the simulation results of the calculated X and Y components of $\dot{\boldsymbol{B}}$ in nT/s as a function of time and superimposes the corresponding $\frac{2\pi}{T} B_y$ and $-\frac{2\pi}{T} B_x$ components in nT. As expected, there is excellent agreement between these values with an average error on the order of 10^{-17}.

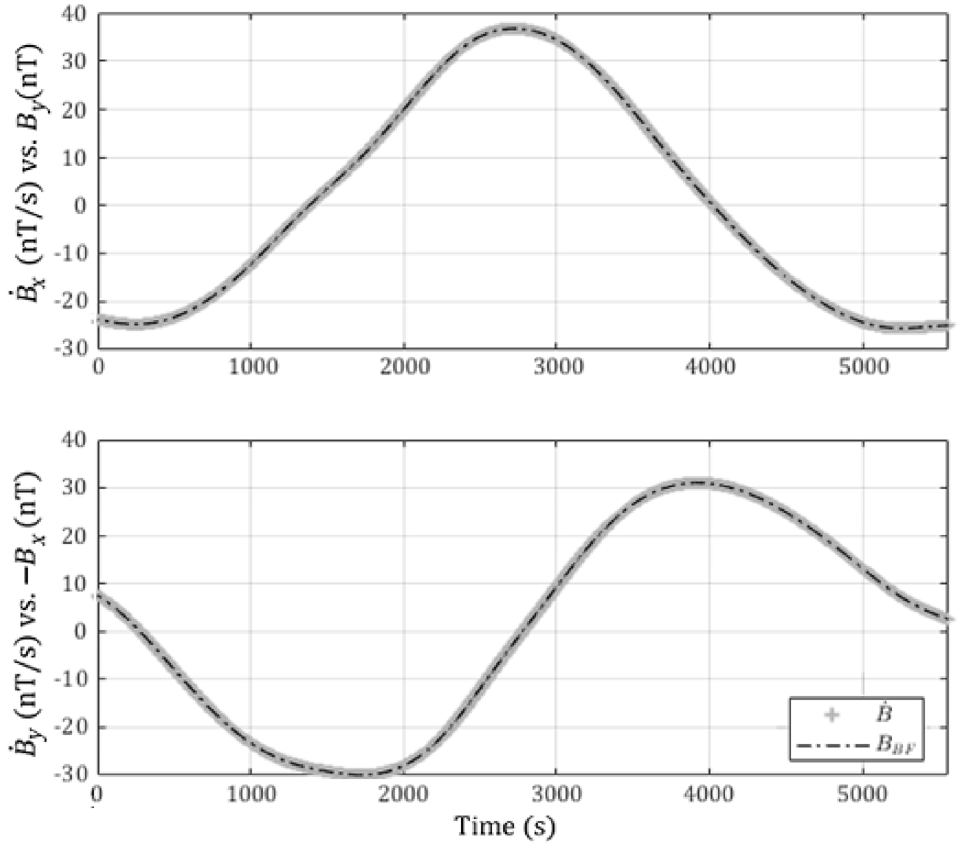


Figure 38: X and Y Component Comparison of $\dot{B}$ and B in the Body-Fixed Frame, Slow Angular Rate

The relation for $\dot{\boldsymbol{B}}$ was then validated for a higher initial constant satellite angular rate.

$$\omega_{ini} = \begin{bmatrix} 0 \\ 0 \\ 0.1 \end{bmatrix} \text{rad/s}$$

$$\dot{B} = \begin{bmatrix} 0.1B_y \\ -0.1B_x \\ 0 \end{bmatrix}$$

The simulation was again run for one orbit (5562 seconds) and the results are shown in Figure 39. Due to the density of the plot resulting from the higher angular rate only 1000 seconds of the simulation are shown. The average errors for the obtained data were on the order of 10^{-15}. Thus, the calculation of $\dot{\boldsymbol{B}}$ as implemented in the simulator can be considered valid.

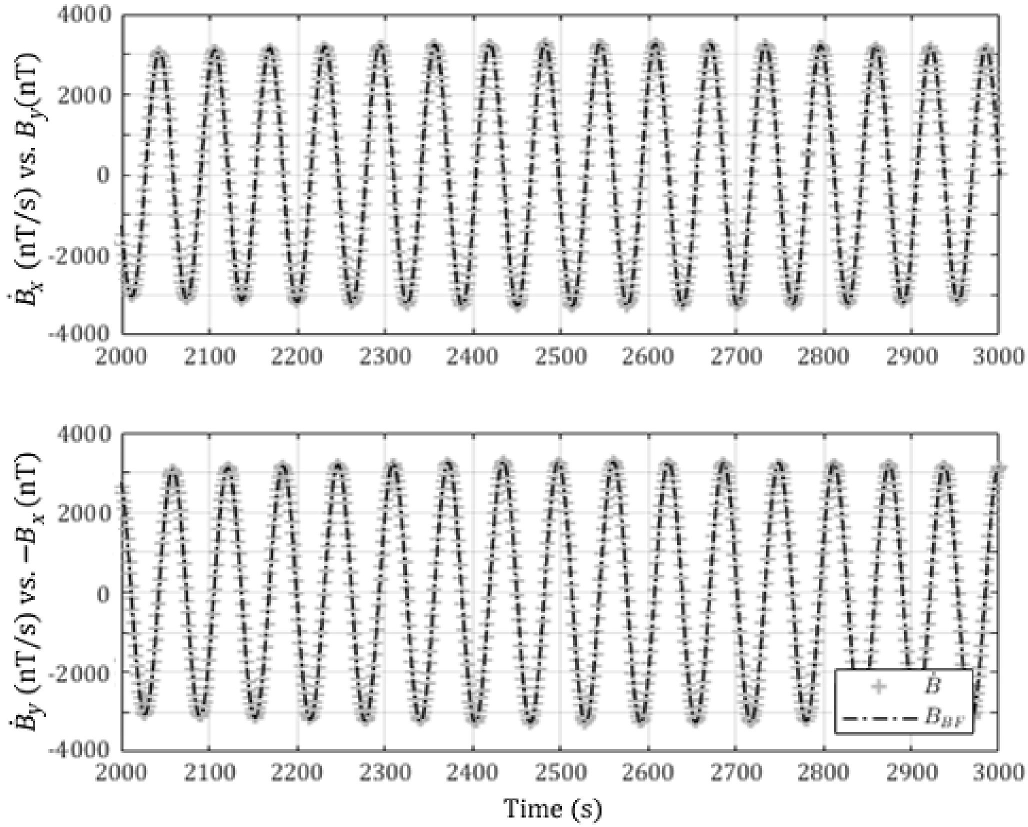


Figure 39: X and Y Component Comparison of $\dot{B}$ and B in the Body-Fixed Frame, Fast Angular Rate

The inability to attain three-axis B-dot control over a satellite in an equatorial orbit due to the invariability in the direction of the local magnetic field vector is well-documented. Without significant changes in local magnetic field direction, control torques can only be applied in one plane and one degree of freedom of control is lost. To validate the implementation of the B-dot control algorithm, a simplified constant-direction geomagnetic field case for a circular equatorial orbit was considered. The magnetic field shown in Figure 40 is defined as constant with magnitude 4000 nT in the ECI Z direction. Within the B-dot algorithm itself, an arbitrary constant gain was selected:

$$K = -1 \times 10^6 \text{ A}^2 s^3 \text{m}^2/\text{rad-kg}$$

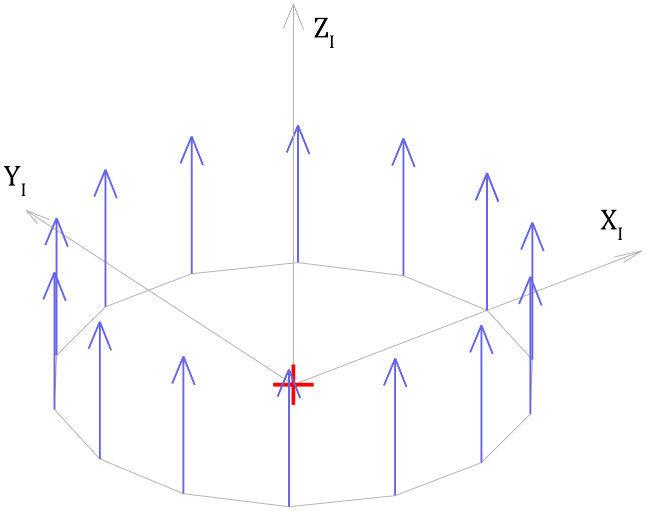
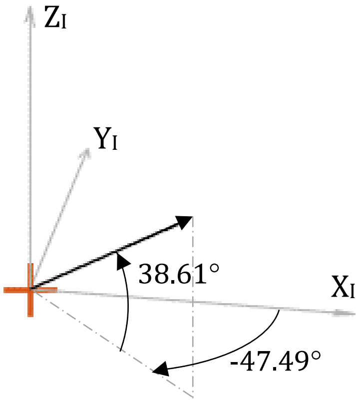


Figure 40: Constant Magnetic Field in ECI Z Direction

For the first test case, an initial velocity of 0.1 rad/s was applied to the satellite along the body-fixed Z-axis with the B-dot algorithm applied over the full orbit. The resulting simulations show that the angular rates in both the ECI and BF frames remain unchanged across the entire orbit and the resulting applied control torquers are zero. One can understand the reason for this observation by examining Equation (50) which, for this case, would result in $\dot{\boldsymbol{B}}$ being zero. From Equation (51), the dipole moment m induced by the magnetorquers would then be zero and, from Equation (52), the resulting applied control torque $\boldsymbol{T}_c$ on the satellite would be zero. Thus, this first validation case ensures that the satellite cannot be detumbled when its angular rate vector is directed along the local magnetic field axis, as is the characteristic limitation of the B-dot algorithm.

For the constant magnetic field along the ECI Z direction, one would expect that an initial spin about all three BF axes would result in the rates about X and Y to tend towards zero while the rate about Z would remain unchanged. To illustrate this concept, the angular rate vector in Figure 41 with distinct components in X, Y, and Z was applied to the satellite.



$$\boldsymbol{\omega}_{ini} = \begin{bmatrix} 0.11 \\ -0.12 \\ 0.13 \end{bmatrix} \text{rad/s}$$

Figure 41: Angular Rate Vector for Second B-Dot Test Case

Figure 42 plots the angular rate components and magnitudes over one orbit for the case where an initial angular rate is applied about all three body axes. The initial 0.13 rad/s component of the angular velocity is shown to remain constant in the ECI frame over one orbit – demonstrated by the datacursor in subplot 6 – as it is unaffected by B-dot control while the remaining rate components in ECI tend exponentially towards zero. Corresponding components of the magnetic dipole moments and control torques are plotted in Figure 43, with datacursors on the latter near zero across the majority of the orbital period indicating that no further control is possible in this case.

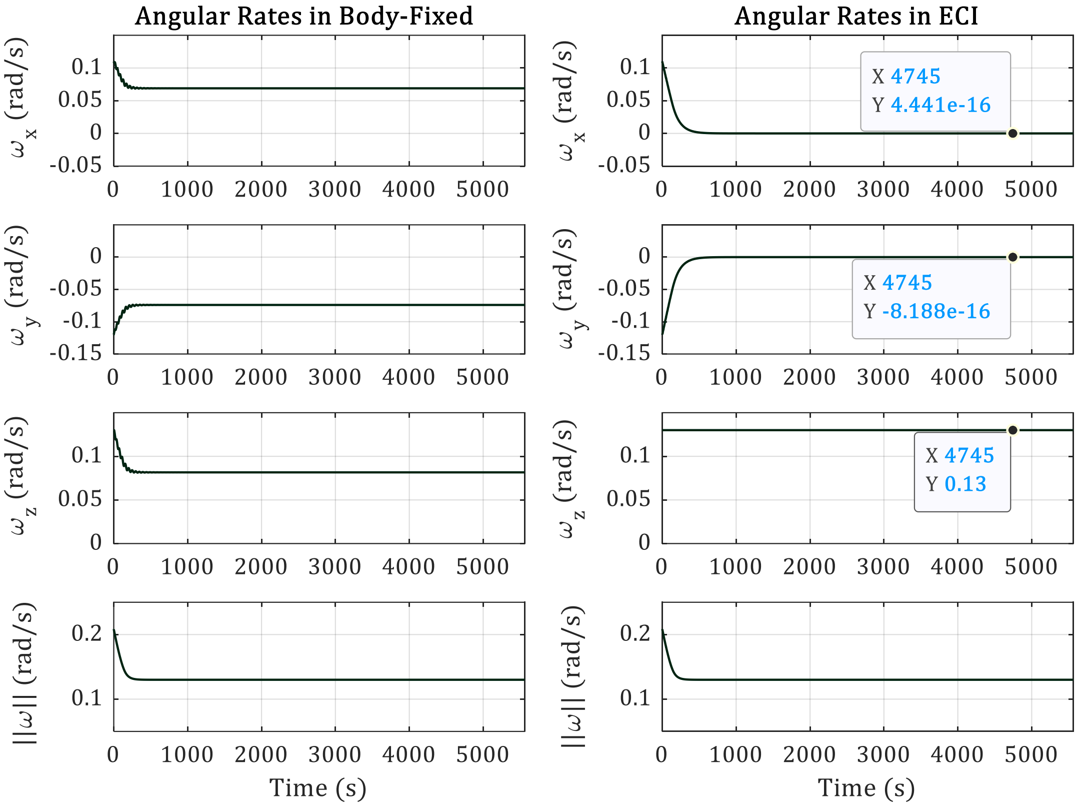


Figure 42: Body Angular Rates for $\omega_{ini} = [0.11 \quad -0.12 \quad 0.13]^T$ rad/s, Field Along ECI Z Axis

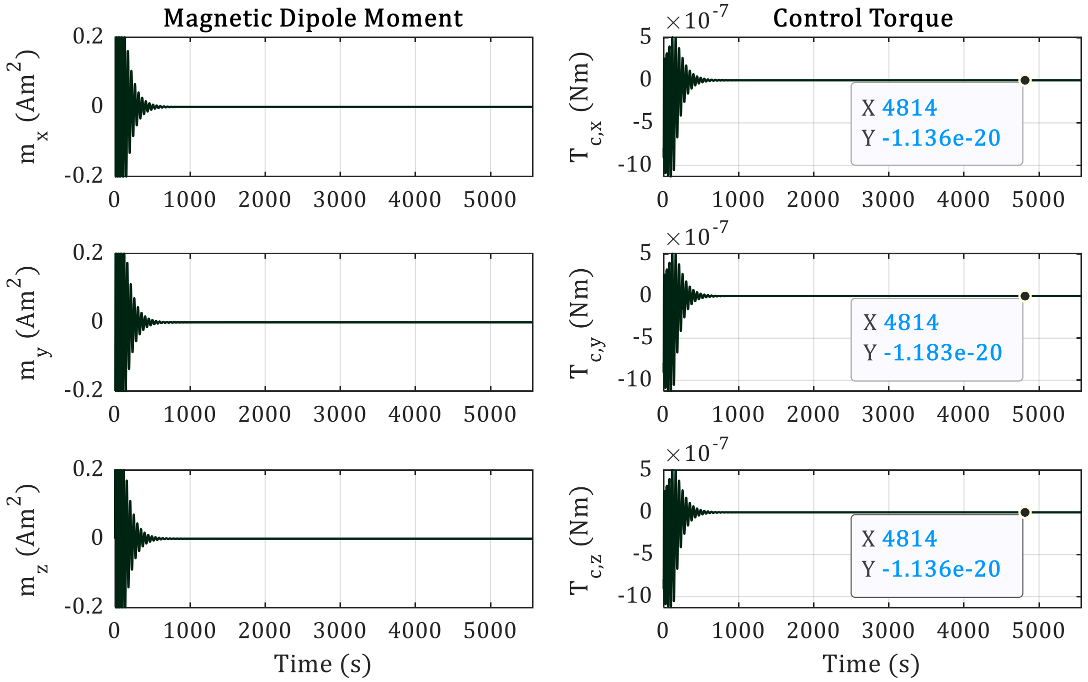


Figure 43: Magnetic Dipole Moment and Control Torque for Figure 42 Case

These two cases prove the proper implementation of the B-dot algorithm in the simulator and the significance of its limitations, with results exhibiting no change in angular rate when rate and field vectors are aligned and clear exponential profiles in the reduction of angular rate when they are not. Furthermore, Figure 42 demonstrates that the profile of the change in angular rates of the satellite when B-dot control is applied is exponential, as documented under ideal conditions [117, p. 80]. Additional cases (not shown) produced similar expected results for all possible combinations of rate and field directions.

The final step in validating the implementation of the B-dot algorithm in the simulator is to test its functionality with more realistic magnetic fields obtained from the IGRF-12 model. Two orbit cases are examined – an equatorial orbit where the magnetic field is oriented predominantly in the ECI Z direction and the ISS orbit defined in Section 3.1.1. The constant B-dot gain value was reduced in magnitude from the previous cases to avoid potential over-saturation of the magnetorquers in the presence of varying magnetic field strength. This gain was selected as:

$$K = -1 \times 10^4 \ \mathrm{A}^2 s^3 \mathrm{m}^2/\text{rad-kg}$$

In both cases, the initial orbital radius was selected as $r_{ini} = 6785$ km and the initial angular rate vector of the satellite was specified as:

$$\boldsymbol{\omega}_{ini} = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} \text{ rad/s}$$

The equatorial orbit case is defined by the following parameters with its ground track shown in Figure 44.

$$e = 0 \qquad T = 5562 \text{ s}$$

$$i = 0.0001 \text{ rad} \qquad \theta_{ini} = 0$$

$$\omega = 0.0001 \text{ rad} \qquad t_p = 0$$

$$\Omega = 0.0001 \text{ rad} \qquad t_{ini} = 13{:}09{:}36 \text{ UTC}$$

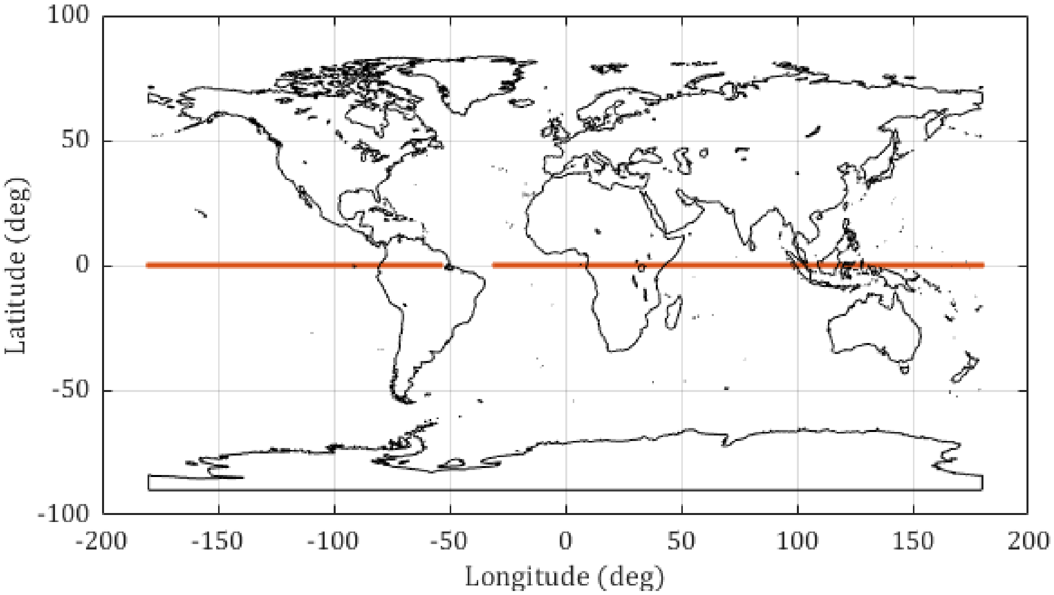


Figure 44: Equatorial Orbit Ground Track [118]

Magnetic field vectors in the ECI frame around the equator are shown in Figure 45, with the approximate 23° offset due to Earth's obliquity between each orbit illustrated in Figure 44 excluded.

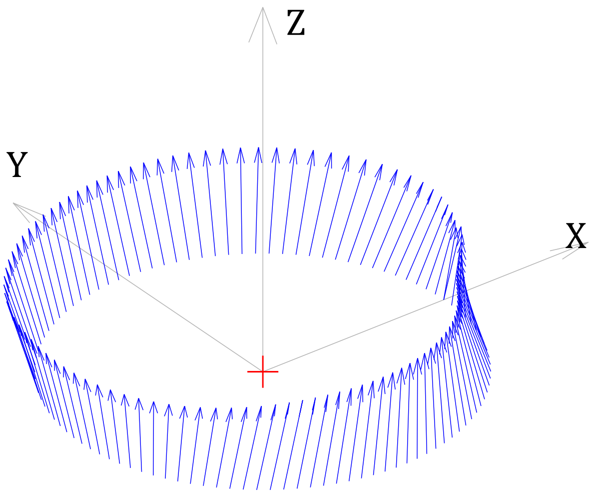


Figure 45: ECI Magnetic Field Vector Directions Over One Equatorial Orbit

From this figure, the geomagnetic field direction appears to be oriented predominantly in the Z direction with very little variation in the X and Y directions. This observation is supported by the field strength vector components over one orbit plotted in Figure 46, showing Z to be greatest in magnitude throughout the orbit.

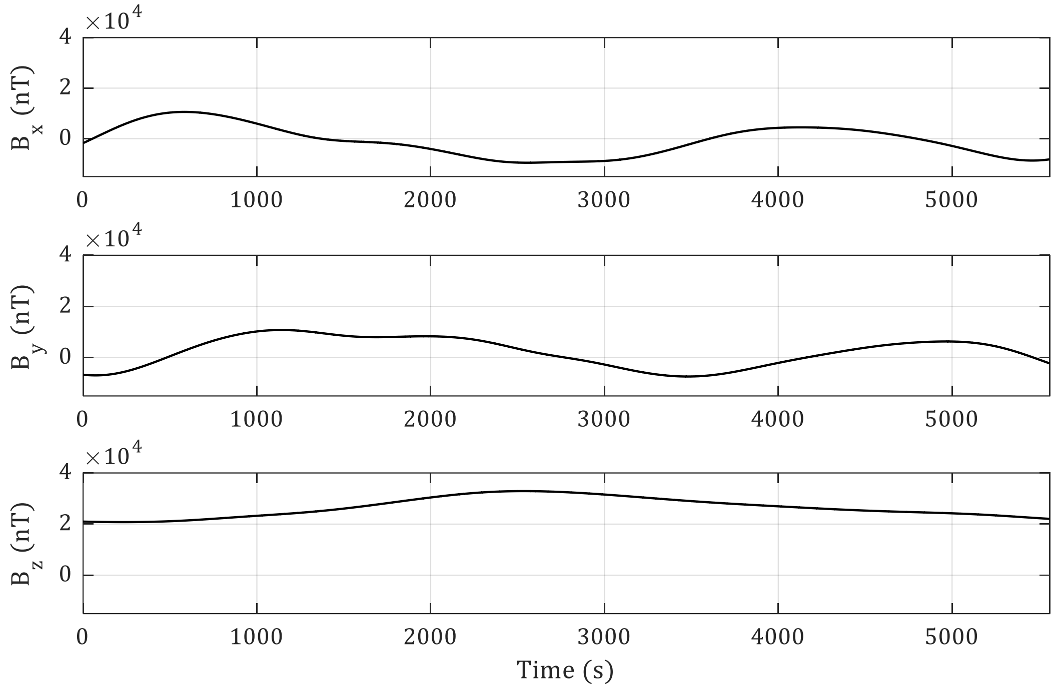


Figure 46: ECI Magnetic Field Strength Over One Equatorial Orbit

Body angular rates are shown in Figure 47 for one orbit in the BF and ECI frames. The rates tend towards zero but do not reach it in a single orbit, as expected given the general Z-axial trend in geomagnetic field direction for an equatorial orbit shown in Figure 45. Corresponding magnetic dipole moments and control torques are plotted in Figure 48 as a function of time. It was expected that the equatorial orbit will be unfavourable for B-dot control given the lack of variation in magnetic field strength and direction across a single orbit, which has proven to hold true – the satellite will need to orbit significantly longer to reduce the body angular rates to zero.

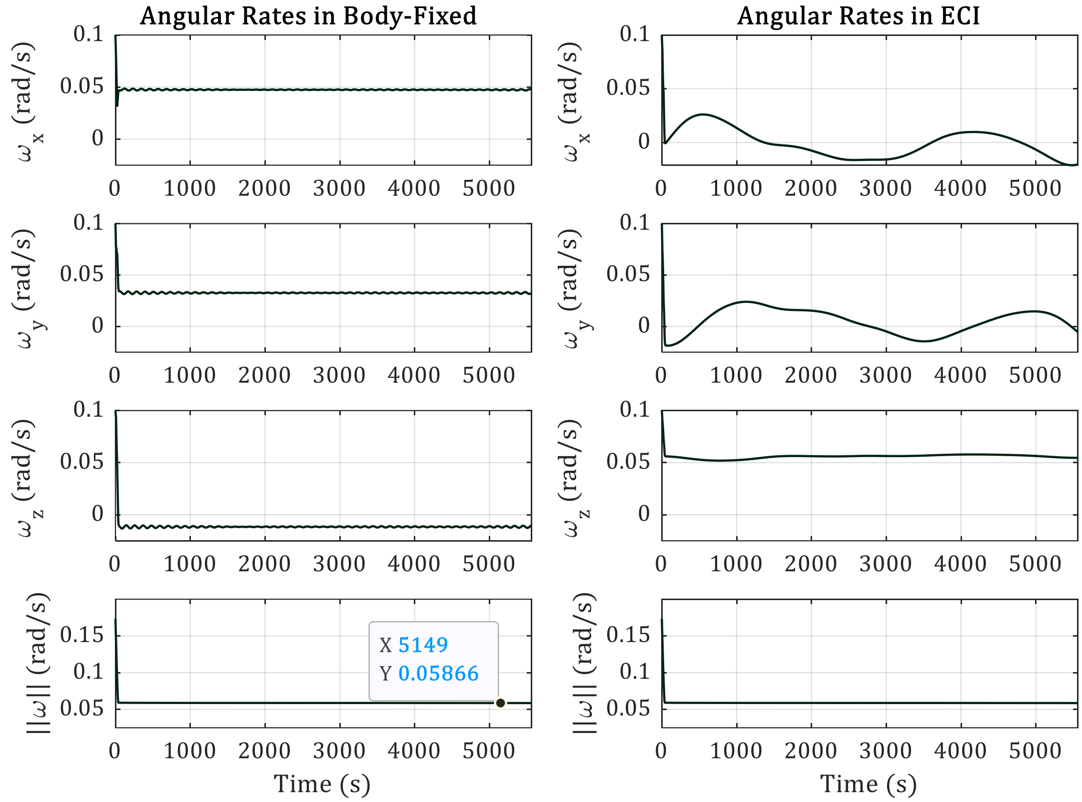


Figure 47: Body Angular Rates Over One Equatorial Orbit with B-dot Control

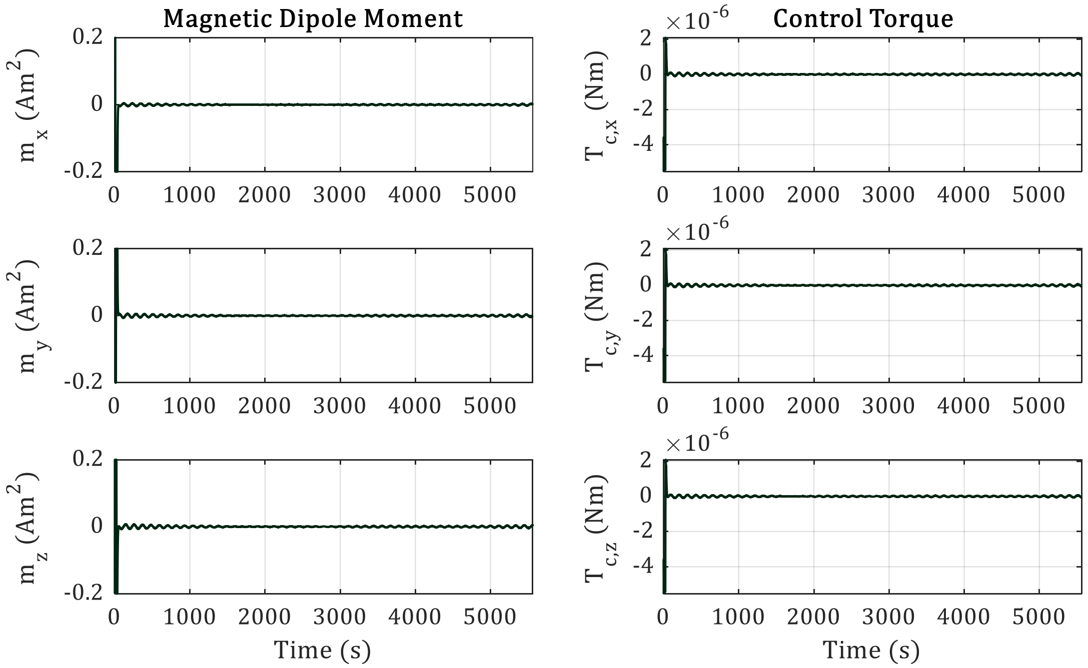


Figure 48: Magnetic Dipole Moment and Control Torques Over One Equatorial Orbit with B-dot Control

The previously defined ISS orbit is used as the second realistic geomagnetic field test case; its orbital elements are reproduced below with its ground track following in Figure 49.

$e = 0.0001068$

$T = 5562\text{ s}$

$i = 51.6413°$

$\theta_{ini} = 251.5996°$

$\omega = 231.7821°$

$t_p = 1673\text{ s}$

$\Omega = 257.8729°$

$t_{ini} = 13{:}09{:}36\text{ UTC}$

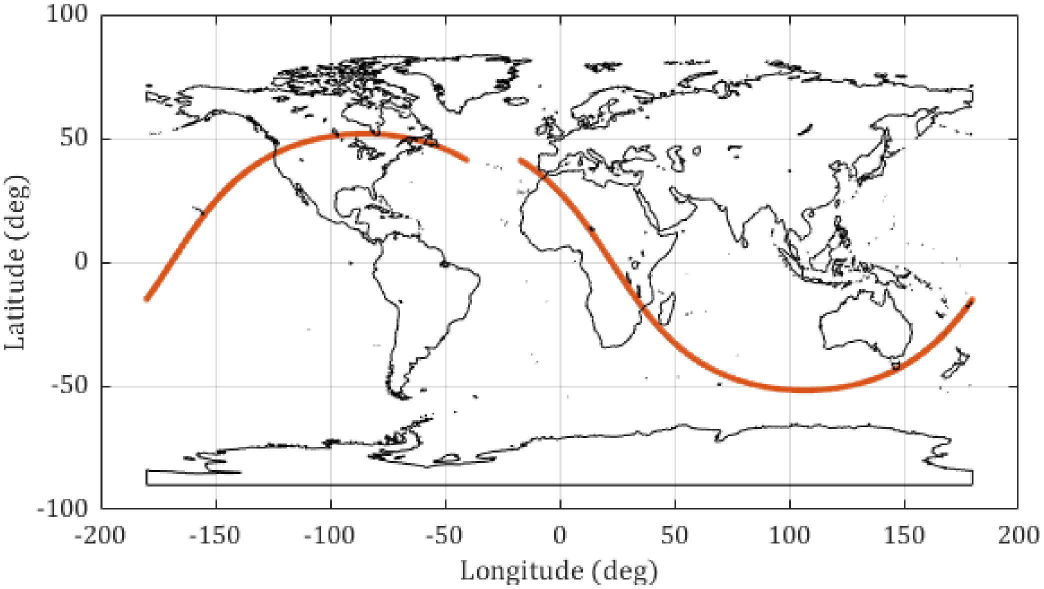


Figure 49: Inclined ISS Orbit Ground Track [118]

The local magnetic field direction plotted in Figure 50 is shown to vary about all three axes across one orbit. Figure 51 plots the individual components of the field over one orbit, showing greater variations in magnitude and direction than in the equatorial orbit case.

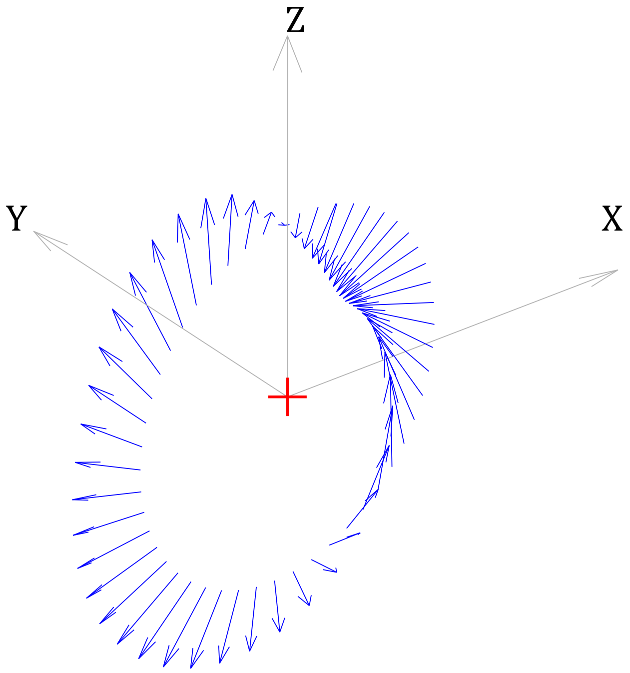


Figure 50: ECI Magnetic Field Vector Directions Over One Inclined ISS Orbit

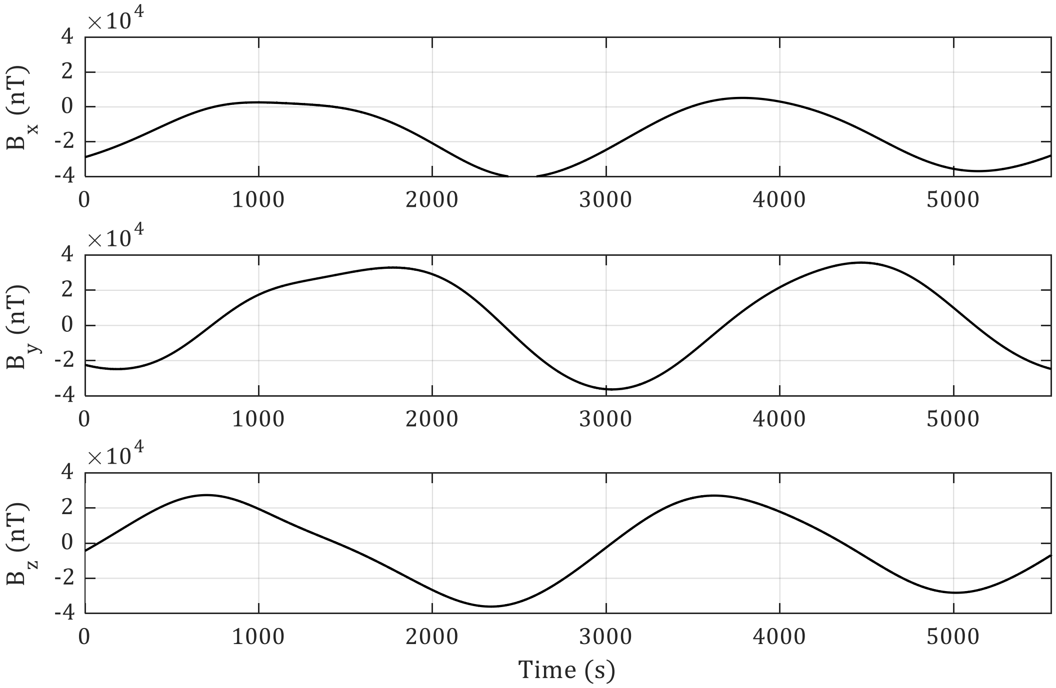


Figure 51: ECI Magnetic Field Strength Over One Inclined ISS Orbit

Controlled body angular rates over one orbit follow in Figure 52 with associated magnetic dipole moments and control torques plotted in Figure 53. The angular rates approach zero and magnetic dipole moments and corresponding torques approaching virtually zero. As demonstrated by comparing geomagnetic field directions in Figure 45 and Figure 50, inclined orbits experience far greater directional variation and are hence favoured for use with magnetically controlled LEO satellites for their ability to reduce angular rate more significantly within the same timeframe.

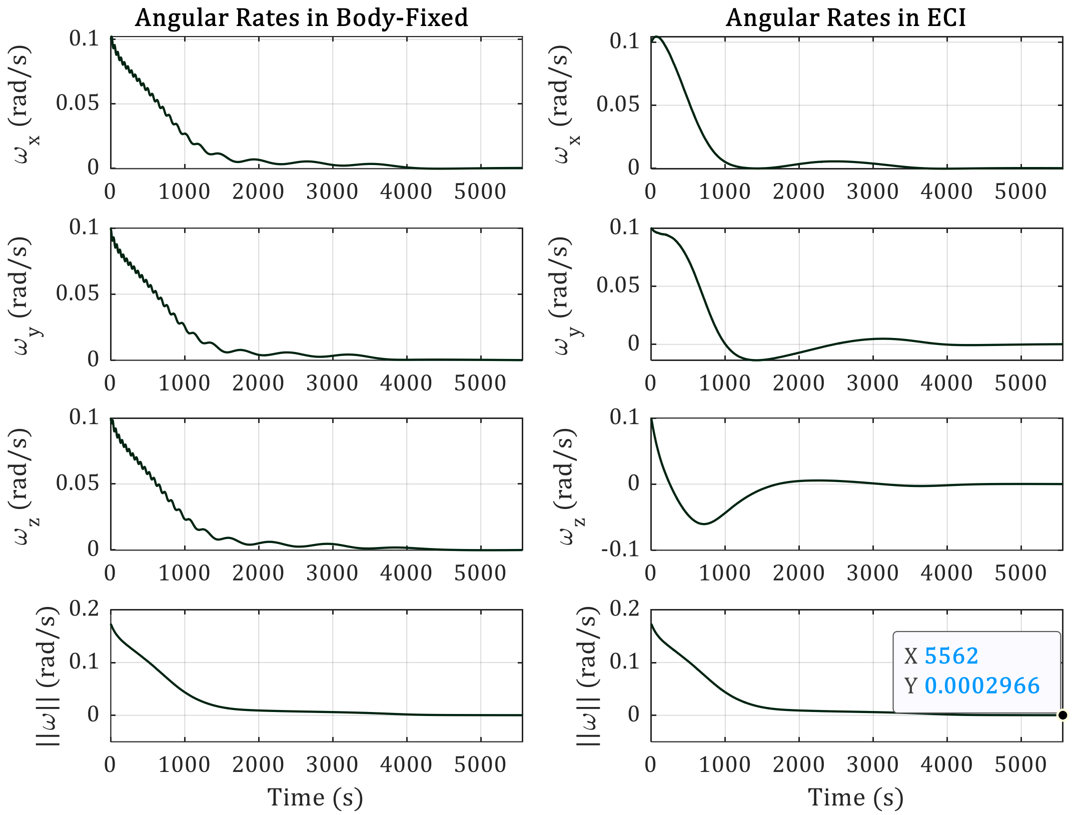


Figure 52: Body Angular Rates Over One Inclined ISS Orbit with B-dot Control

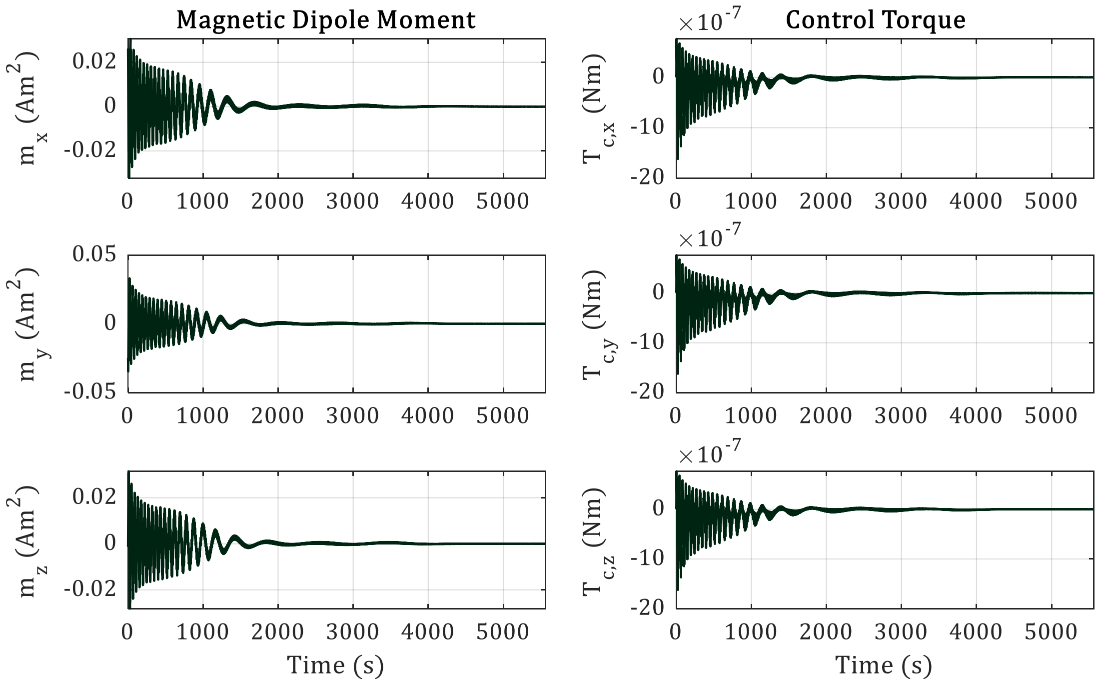


Figure 53: Magnetic Dipole Moment and Control Torques Over One Inclined ISS Orbit with B-dot Control

5.2.1 Gain Tuning

The B-dot control algorithm is considered optimised for the lowest possible settling time in accordance with the mission requirement of detumbling in under 15 hours (around 10 orbits). The designed B-dot controller makes use of a constant proportional gain in the interest of reducing the potential for controller error in flight. To select a suitable gain, different values were tested under ideal conditions with no external disturbance torques and perfect knowledge of the attitude, local magnetic field, and angular rate. The absence of external disturbances implies that the satellite should remain motionless once detumbled, and thus is considered settled once the angular rate reaches 10^{-6} rad/s. To determine the significance of changes in CubeSat geometry on the ability of the controller to function as designed, the simplified 1U and 2U geometries shown in Table 6 were analysed.

The initial angular rate vector was chosen as in previous validation cases, approximately equivalent to a 10 deg/s tumble:

$$\boldsymbol{\omega}_{ini} = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix} \text{ rad/s}$$

Figure 54 plots settling times for the 1U geometry as a function of B-dot gains K in the absence of external disturbances and assuming perfect attitude, local magnetic field, and angular rate knowledge. The inset plot shows the settling times for additional gains between 6,000 and 10,000 $\mathrm{A^2 s^3 m^2/rad \cdot kg}$ – the best performing region in the dataset. Under these conditions, a gain of $K = 7900\ \mathrm{A^2 s^3 m^2/rad \cdot kg}$ yielded the lowest settling time of 4224 seconds (less than one orbital period) for the 1U geometry.

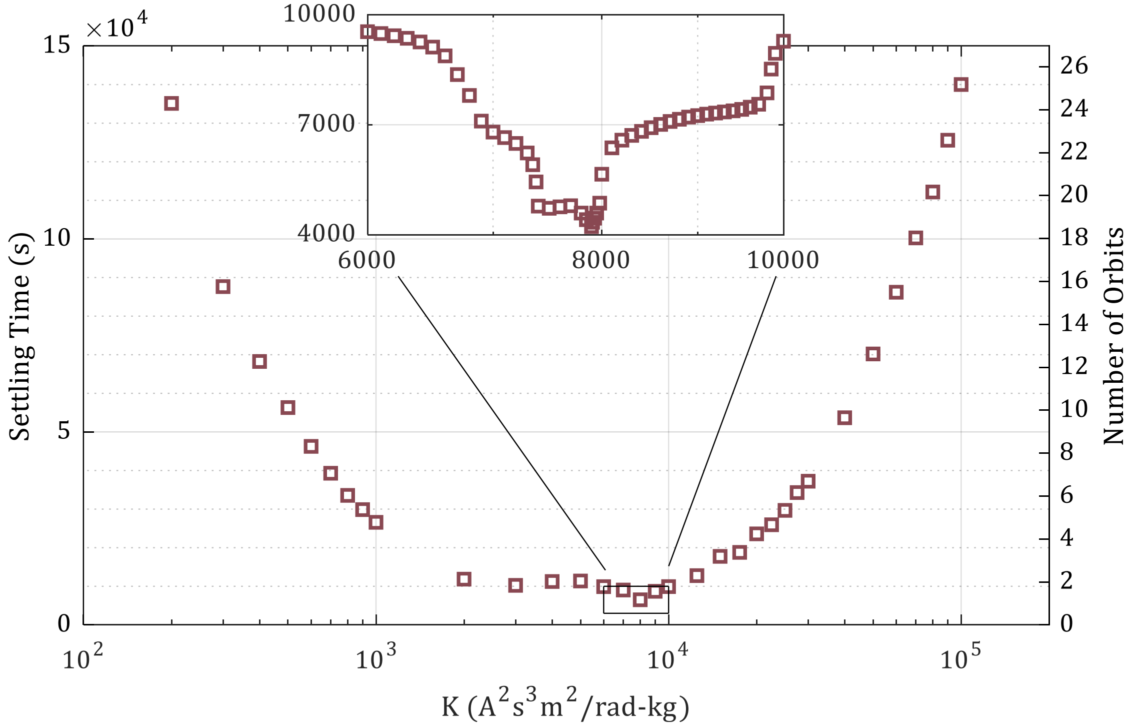


Figure 54: 1U Geometry B-dot Settling Times for Different Gains K [119]

A similar range of B-dot gains were then simulated for the 2U geometry. Figure 55 compares the resulting settling times for both geometries, the 1U denoted by cubes and the 2U by diamonds. A gain of $K = 40{,}000\ \mathrm{A^2 s^3 m^2/rad \cdot kg}$ yielded the lowest settling time at 7258 seconds (~18 minutes more than one orbital period) for the 2U geometry, circled in red in Figure 55. While the lowest settling times for the 1U and 2U CubeSats are similar, the 2U geometry requires a B-dot gain approximately 5 times higher than that for the 1U geometry. Figure 55 further suggests there exists a relatively wide range of B-dot gain values that enable both 1U and 2U satellites to settle in 10 orbits or fewer, thus easily meeting the relevant LORIS system requirement. Should the satellite geometry or its mass distribution characteristics change in orbit, there is potential for the B-dot controller to retain its base functionality to detumble the satellite within a reasonable timeframe with its launch parameters.

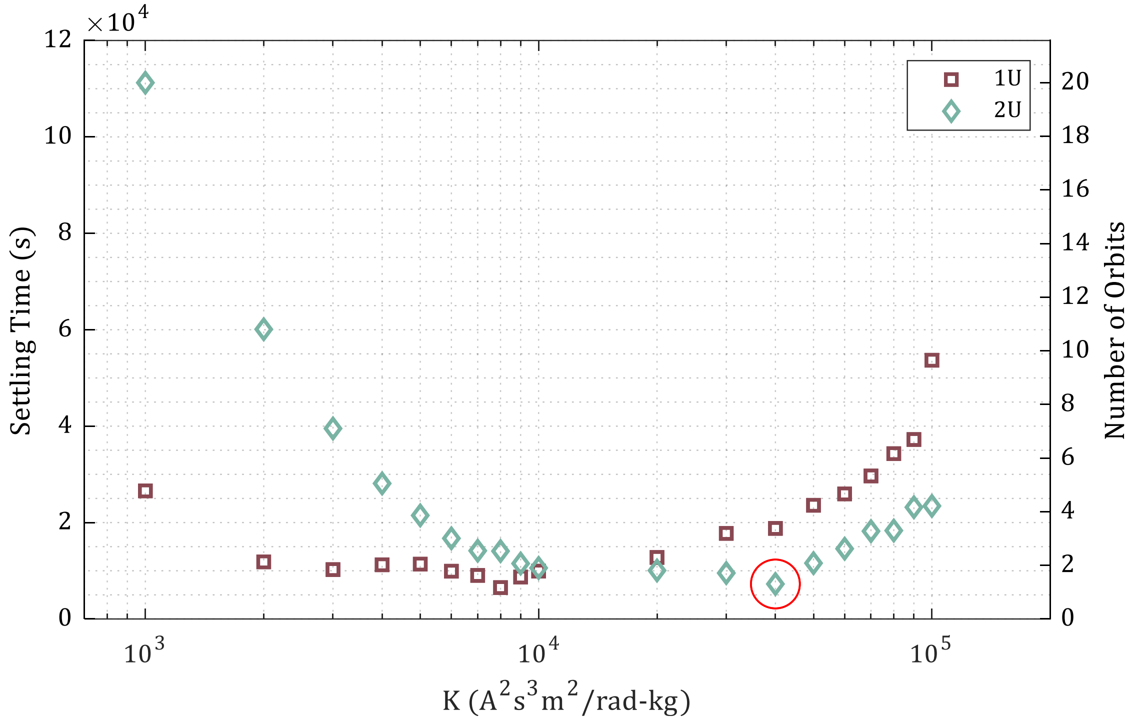


Figure 55: Comparison of B-dot Settling Times with Different Gains *K* for 1U and 2U Geometries [119]

These results are presented for idealised geometry, environment, and sensor data; the settling time can be expected to worsen as more realistic conditions are added. These results can be taken as a baseline to assess final controller performance against. The following section relates to the last point – instead of assuming the angular rate to always be known, it is assumed that the angular rate cannot be measured and can only be approximated from existing sensor data.

5.3 B-Dot Algorithm with Derivative

Reliance on a gyroscope to provide accurate readings of Earth's magnetic field becomes problematic when the satellite's rates are reduced beyond the typical gyroscope's operational range. An overcompensating detumbling algorithm may further destabilise the satellite without an accurate read on the satellite's body rates and may prevent the rates from ever being reduced to an appropriate range for use of the three-axis pointing controller. An alternative to using a gyroscope to determine the satellite's angular rates is to instead use the change in magnetic field direction with time [120].

The expression for magnetic dipole moment $\boldsymbol{m}$ in Section 5.2 may be redefined with $\dot{\boldsymbol{B}}$ equal to the time derivative of the local magnetic field:

$$\boldsymbol{m} = -K\dot{\boldsymbol{B}} = -K\frac{d\boldsymbol{B}}{dt} \tag{53}$$

For a sample time t_s of 1 second, the time derivative can be calculated at a timestep k from the difference of its current and previous $(k-1)$ measurements:

$$\dot{\boldsymbol{B}}_k = \frac{\boldsymbol{B}_k - \boldsymbol{B}_{k-1}}{t_s} \tag{54}$$

Measured magnetic field data is noisy, and therefore an infinite impulse response (IIR) low-pass filter is applied to the calculated derivative to produce a cleaner derivative estimate according to [121]:

$$\dot{\boldsymbol{B}}_k = \alpha\dot{\boldsymbol{B}}_k + (1-\alpha)\dot{\boldsymbol{B}}_{k-1} \tag{55}$$

where smoothing factor α was selected empirically to produce the most consistent results as:

$$\alpha = 0.03$$

Figure 56 compares the low-pass-filtered derivative of the noisy magnetic field to the derivative of the noiseless angular rate in the same reference frame over a 3500 second period. The MPU-9250 magnetometer noise described in Section 4.3 is used for the former. The filtered derivative approximates the amplitude of higher-frequency regions less accurately than lower-frequency regions, though any detriment this causes to the overall performance of the controller quickly resolves itself once the rates are attenuated within a certain frequency.

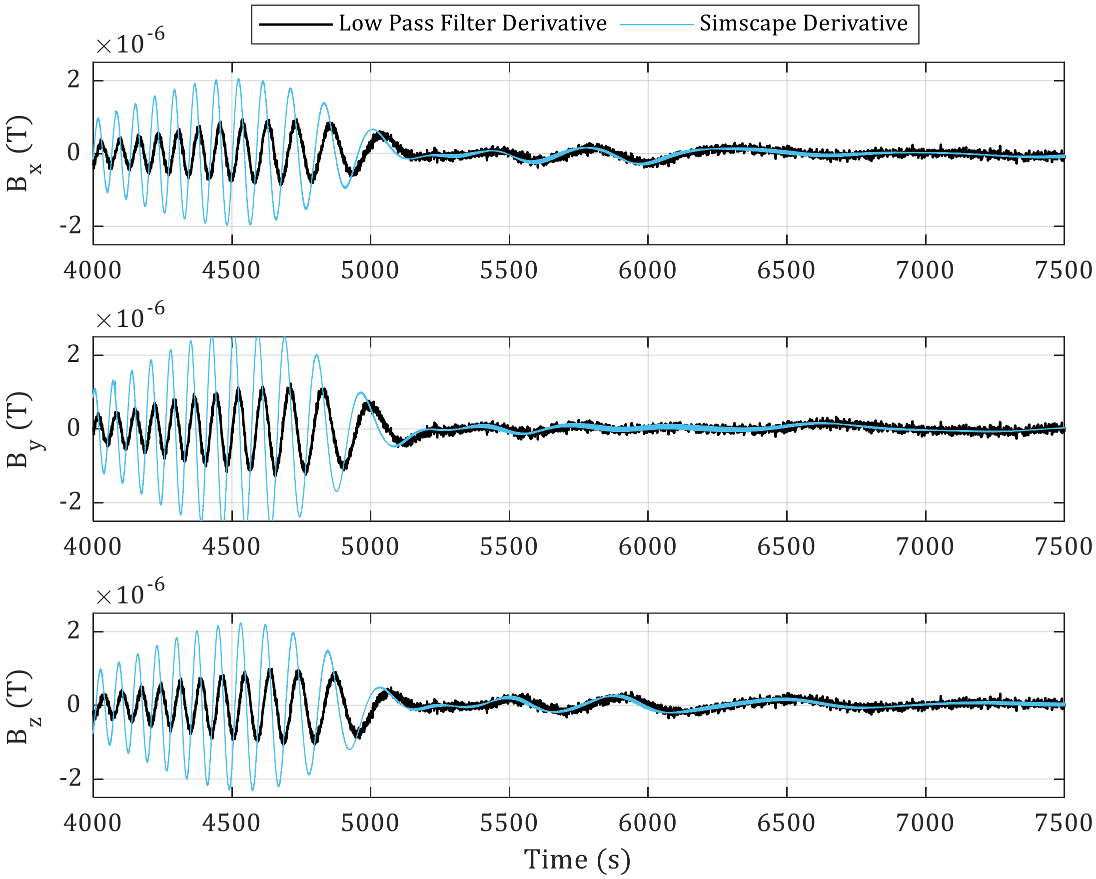


Figure 56: Comparison of Magnetic Field Derivatives Over Portion of Two Orbits

The next section compares the performance of this form of the B-dot controller to that derived in Section 5.2 for the final geometry of the satellite.

5.4 B-dot Final Results

With the form of the B-dot controller finalised, another iteration of gain tuning was completed for the final geometry of the satellite. With noisy differentiated inputs, body angular rates will never truly reach zero. Thus, the settling time criteria was redefined for this tuning iteration – the satellite is considered settled once its angular rates remain within the threshold of ± 0.005 rad/s for 500 seconds with the settling time taken once the rates about all axes enter and remain within this range. At higher gains, it was found that the satellite would occasionally leave this threshold after some duration of time. For a proper comparative basis with the results presented in Section 5.2.1, settling times greater than 150000 seconds (27 orbits) were excluded from the analyses and initial angular rates were set to 0.1 rad/s about every axis. Environmental disturbance torques are summed to the B-dot control torques before being applied to the satellite in Simscape.

Figure 57 plots different B-dot gains versus settling time over a similar range as in Figure 55. Once again, the settling times formed a parabolic profile with respect to the logarithmic gain axis. Gains on the highest and lowest ends of the spectrum exhibited more noise in their rate behaviour than those in the vicinity of the vertex of the parabola and would occasionally spike outside of the 0.005 rad/s threshold for durations of less than 100 seconds. The fastest settling time is circled on the plot for a gain $K = 7 \times 10^4\ \mathrm{A^2 s^3\ m^2/rad\ kg}$ at 6716 seconds, or approximately 1.2 orbits.

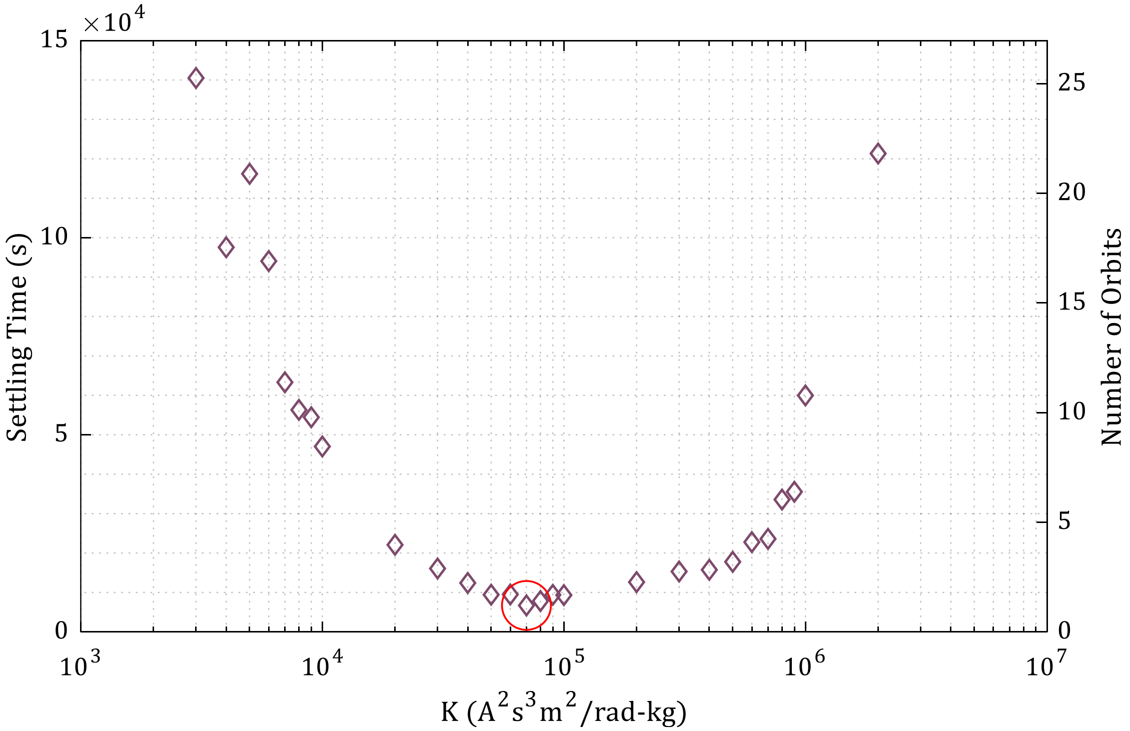


Figure 57: New B-dot Settling Times for Different Gains *K*

The following three plots present different detumbling case studies for the final low-pass filtered derivative B-dot controller with disturbance torques and sensor noise present for the final satellite geometry. The two grey lines imposed on each plot show the detumbling threshold of ±0.005 rad/s. For all cases, the body angular rates are maintained within this threshold once the satellite is detumbled.

For the first case study, Figure 58 plots the body angular rates for the tuning case with an initial angular rate of 0.1 rad/s about each axis (a total initial rate magnitude of 0.1732 rad/s (approximately 10 deg/s)). The satellite settles as stated in approximately 6716 seconds or 1.2 orbits.

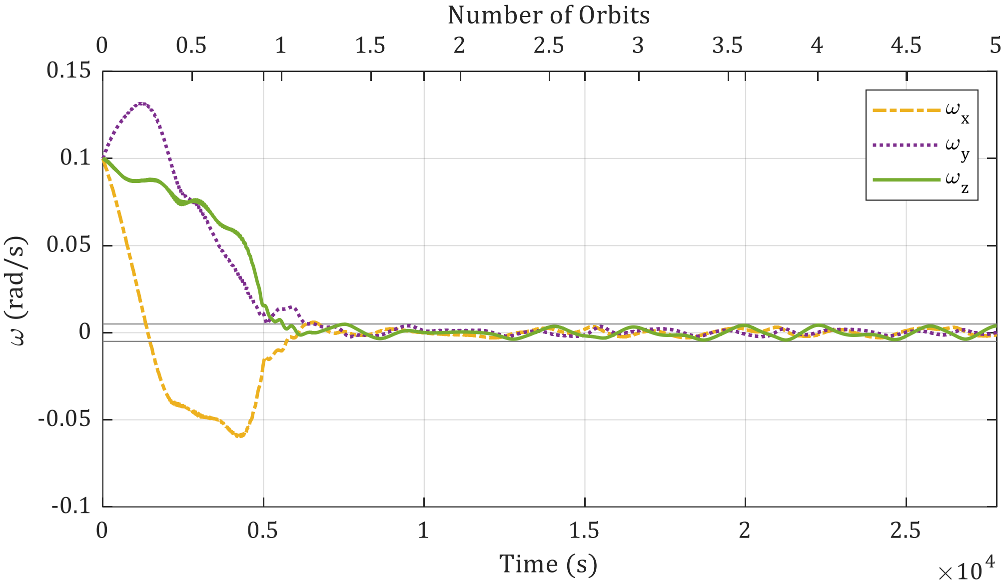


Figure 58: Best Case B-Dot Gain Detumbling from 0.1 rad/s about All Axes

Figure 59 shows the results for the second case study where the satellite started from a much higher initial angular rate of 0.35 rad/s (20 deg/s) about each axis, with an absolute magnitude of 0.6061 rad/s (approximately 34 deg/s). Though detumbling takes approximately 10 times as long to settle versus the previous case at 12 orbits, the body angular rates are still reduced and maintained within the threshold.

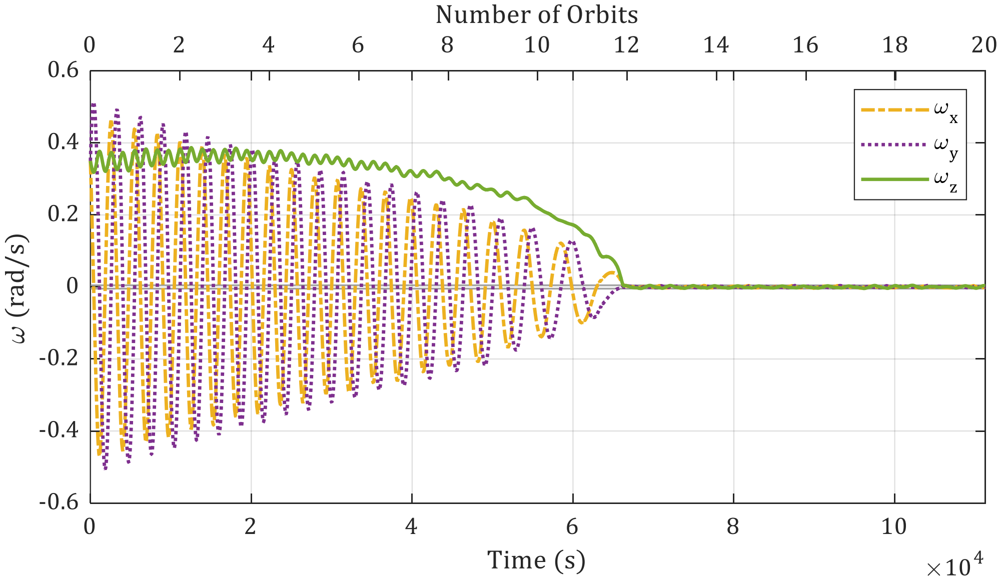


Figure 59: Best Case B-Dot Gain Detumbling from 0.35 rad/s about All Axes

For the final case study, detumbling a random initial tumble rate with different magnitudes and directions about each body axis is shown in Figure 60. The initial rate vector is shown below, having a magnitude of 0.2333 rad/s (approximately 13 deg/s).

$$\boldsymbol{\omega} = \begin{bmatrix} -0.12 \\ 0.2 \\ 0.005 \end{bmatrix}$$

The satellite detumbles in approximately 2.5 orbits, with the highest initial rate in Y settling last within the threshold.

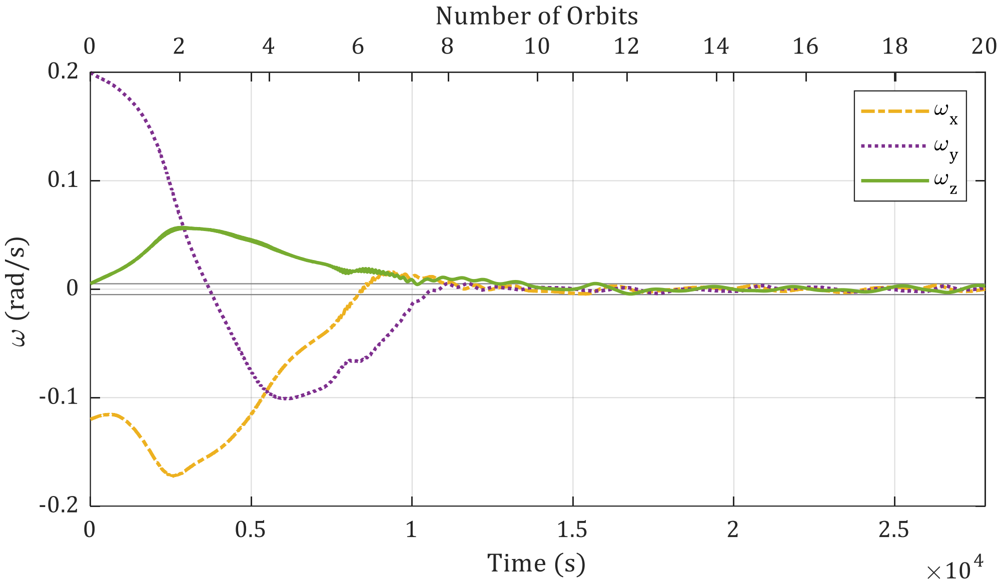


Figure 60: Best Case B-Dot Gain Detumbling from $\boldsymbol{\omega}_i = [-0.12 \quad 0.2 \quad 0.005]$ rad/s

Figure 61 reproduces Figure 58 in its bottom subplot to compare the detumbling profile directly with the performance of the original B-dot controller derived in Section 5.2. The top subplot uses the original controller with the best-case gains identified in 5.2.1 with the final satellite geometry, noisy gyroscope and magnetometer readings, and disturbance torques summed to its control signal output. Though the original controller exhibits the expected exponential profile characteristic of the B-dot algorithm, it does not settle in significantly less time than the low-pass filtered derivative controller – 0.8 orbits versus 1.2 orbits. Neither destabilise once their rates have entered the detumbling threshold. Thus, the new B-dot simplifies the ADCS with little effect on overall control performance.

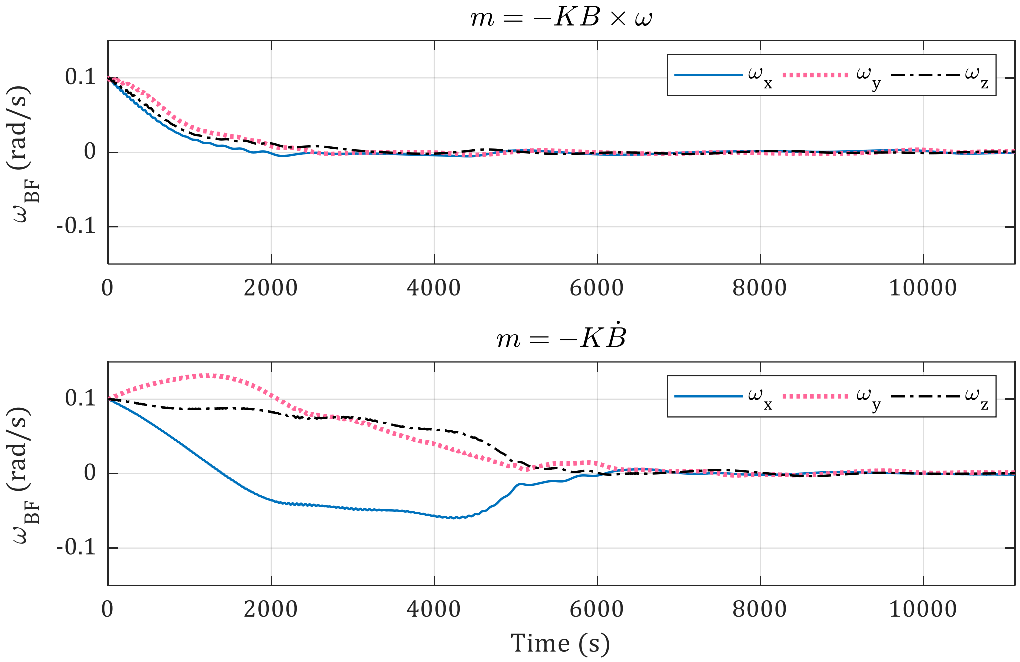


Figure 61: Comparison of Both B-dot Controllers for the Final Satellite Geometry with Noisy Input and Disturbance Torques

With the detumbling controller finalised, the implementation of the first of the two operational phases of the LORIS CubeSat in the simulator has been completed. The controller is stable in the presence of worst-case disturbance torques and is capable of detumbling the satellite over a wide range of angular rates to the order of magnitude at which the pointing controller can operate, easing the transition between operational phases. The next chapter outlines the design of and problems encountered for a magnetically-actuated PD pointing controller.

Chapter 6: Magnetic Pointing Control

The second of the LORIS CubeSat's two main ADCS operational phases is the three-axis pointing phase, wherein the satellite tracks a desired attitude that enables its payload to complete its Earth-pointing objective using a three-axis stabilised control algorithm. This chapter investigates the feasibility of using only magnetic actuators with a proportional-derivative (PD) attitude controller to achieve the desired pointing accuracy within a reasonable timeframe. Improvements to the attitude estimate and raw sensor data used as inputs to the pointing controller are assessed, and problems encountered with magnetic actuation are presented.

6.1 PD Controller

Three-axis magnetic attitude control for the satellite can be accomplished using a PD controller defined as follows [48, p. 5]:

$$\boldsymbol{T}_{c_{desired}} = -K_p\hat{q} - K_d\hat{\boldsymbol{\omega}} \tag{56}$$

where $\boldsymbol{T}_{c_{desired}}$ is the desired control torque expressed in the BF frame, $\hat{q}$ is the vector part of the quaternion error, $\hat{\boldsymbol{\omega}}$ is the body angular rate error, and K_p and K_d are the controller proportional and derivative gains, respectively. The quaternion error $\hat{q}$ and body angular rate error $\hat{\boldsymbol{\omega}}$ are defined as the attitude quaternion and body angular rates in the body-fixed frame expressed relative to the nadir-pointing frame. Both proportional and derivative terms are negative; taking the negative of the quaternion vector elements defines the opposite-direction axis, thus representing the "error" with the nadir-pointing attitude as the desired orientation [122, p. 9]. Angular rate error is taken directly from Simscape (with the optional addition of noise). Noiseless quaternion error is taken directly from Simscape for initial validation cases, while subsequent simulation testing calculates the error quaternion using the following relation:

$$\hat{q} = \ q_{BFI}{}_{estimated}^{-1} q_{NPI} \equiv q_{BFNP}$$

where $q_{BFI_{estimated}}$ is the estimated attitude quaternion from the q-Method representing the orientation of BF frame with respect to the ECI frame, and q_{NPI} represents the orientation of the NP frame with respect to ECI frame. The quaternion inverse is calculated as follows, where $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ are unit vectors along the BF X, Y, and Z axes, respectively [123]:

$$q^{-1} = \frac{q_s - \boldsymbol{i}q_x - \boldsymbol{j}q_y - \boldsymbol{k}q_z}{q_s^2 + q_x^2 + q_y^2 + q_z^2} \tag{57}$$

The desired outcome of magnetic PD control is aligning the CubeSat's BF frame with the NP frame such that it rotates at its same relative rate with respect to the ECI frame. The corresponding desired vector of angular rates $\omega_{desired}$ and attitude quaternion $q_{desired}$ are as follows:

$$\omega_{desired} = \begin{bmatrix} 0 \\ -\frac{2\pi}{T} \\ 0 \end{bmatrix} \text{ rad/s}$$

$$q_{desired} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where the first element of $q_{desired}$ is the scalar component of the quaternion.

The desired control torque $\boldsymbol{T}_{c_{desired}}$ obtained from Equation 56 is not necessarily attainable given the nature of the local magnetic field. To calculate the achievable (actual) control torque $\boldsymbol{T}_{c_{actual}}$ the magnetic moment $\boldsymbol{M}_{BF}$ is first calculated using $\boldsymbol{T}_{cdesired}$ and the local magnetic field strength $\boldsymbol{B}_{BF}$ as follows:

$$\boldsymbol{M}_{BF} = \boldsymbol{B}_{BF}^{-1}\, \boldsymbol{T}_{c_{desired}} \tag{58}$$

The actual control torque $\boldsymbol{T}_{c_{actual}}$ that can be applied is then calculated from the desired as follows [48, p. 5]:

$$\boldsymbol{T}_{c_{actual}} = \boldsymbol{M}_{BF} \times \boldsymbol{B}_{BF} \tag{59}$$

where all vectors in these equations are expressed in the BF frame.

The difference between the desired and actual control torques is illustrated in Figure 62; the actual control torque is the component of the desired control torque that lies in the plane normal to the local magnetic field direction.

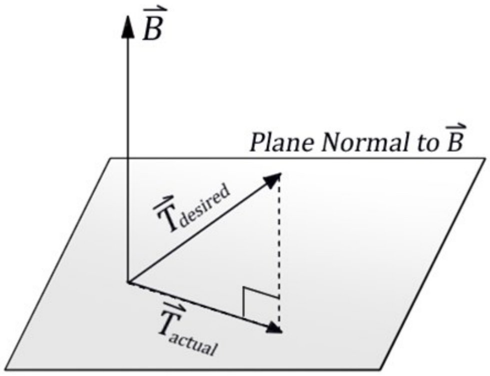


Figure 62: Attainable (Actual) PD Control Torque versus Desired PD Control Torque

6.1.1 Gain Tuning

For initial investigation of the effect of different gains on magnetic PD control, the simplified 1U and 2U CubeSat geometries in Table 6 were used. The CubeSat was given an initial rotation rate of 10^{-3} rad/s about all three body axes and was simulated under ideal conditions assuming no environmental disturbance torques, perfect attitude knowledge, and an absence of sensor noise. The amount of time for the satellite to settle to within 5% of the desired orbital NP rate of 1.1297×10^{-3} rad/s was then determined. The corresponding orientation of the satellite at this settling time was also checked to ensure that the satellite's attitude matched the desired NP frame: if the attitude error was less than 10 degrees, then the satellite was considered to have "settled".

When attempting to select suitable proportional and derivative gains for the magnetic PD control of a 1U CubeSat, it was observed that many gain combinations did not enable the CubeSat to settle within 500 orbits and some gain combinations could not attenuate the errors. For the simulation conditions used in this research, PD controller gains of $K_p = 1 \times 10^{-9}$ Nm and $K_d = 1 \times 10^{-7}$ Nms/rad were found to successfully enable the 1U CubeSat to track the desired NP frame within ~500 orbits (about 32 days.)

Using $K_p = 1 \times 10^{-9}$ Nm as a starting point, equally spaced values of K_d ranging between $10^{-7} \leq K_d \leq 10^{-3}$ Nms/rad (i.e. $1 \times 10^{-7}, 2 \times 10^{-7}, 3 \times 10^{-7}, \dots$) were simulated. Figure 63 plots the corresponding settling times as a function of K_d for gain values that successfully enabled the 1U CubeSat to "settle". Although this figure shows a local minimum when $K_d = 1 \times 10^{-5}$ Nms/rad, it was observed that a gain of $K_d = 2 \times 10^{-5}$ Nms/rad exhibited a less oscillatory response at steady-state which would be preferable for an Earth-pointing satellite. Therefore, $K_d = 2 \times 10^{-5}$ Nms/rad was selected as the derivative gain for the next iterative stage of tuning.

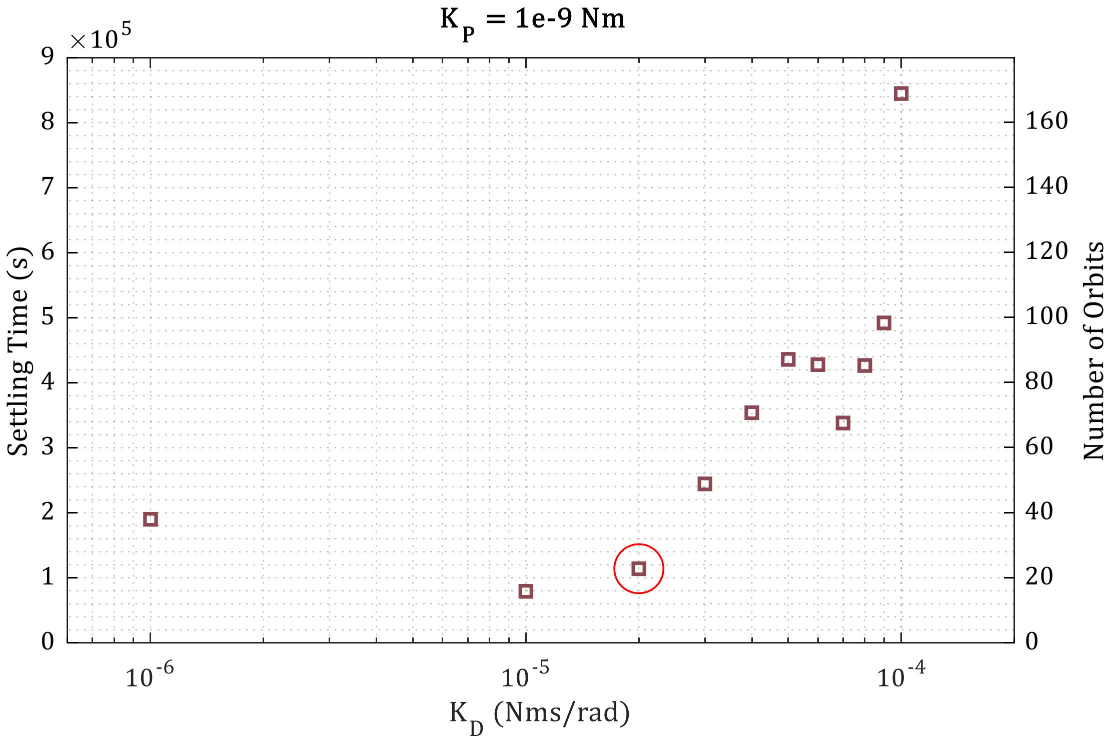


Figure 63: PD Settling Time for 1U CubeSat With Constant $K_p = 1 \times 10^{-9}$ Nm and Different Values of K_d [119]

Holding $K_d = 2 \times 10^{-5}$ Nms/rad fixed, the value of K_p was then varied for the 1U CubeSat from 1×10^{-9} to 1×10^{-8} Nm. For comparison, these gains were also used to simulate the 2U CubeSat and the resulting settling times as a function of K_p are superimposed for both geometries in Figure 64. Note that the two values absent for the 1U CubeSat ($K_p = 9 \times 10^{-9}$ and $K_p = 1 \times 10^{-8}$ Nm) did not successfully settle within 500 orbits and are, therefore, excluded from the plot to maintain a reasonable scale.

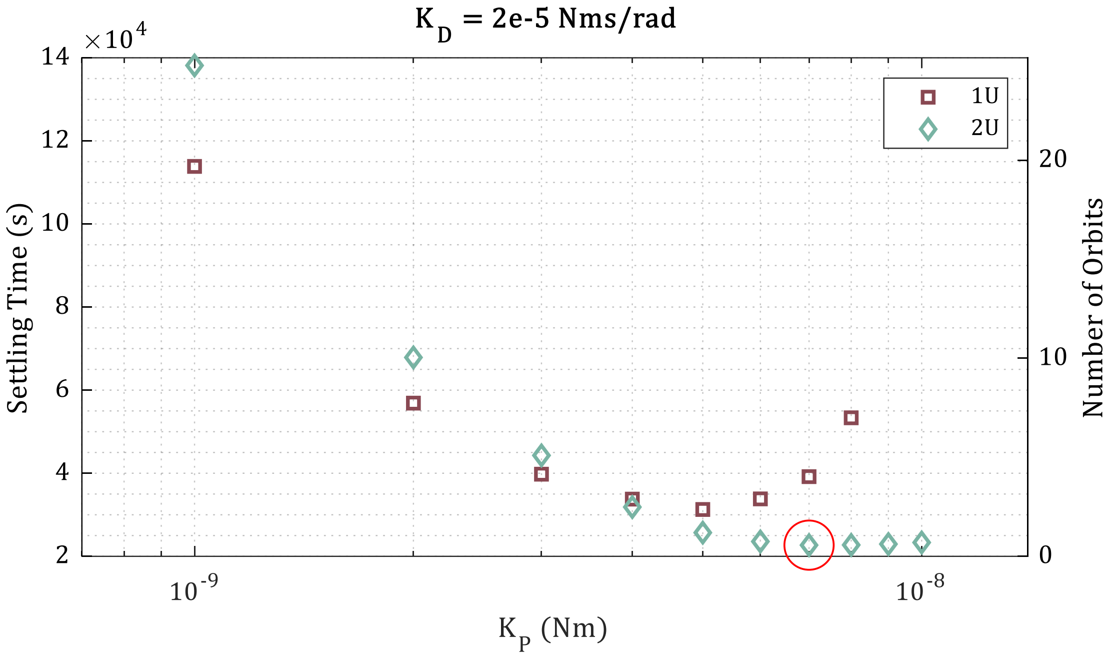


Figure 64: PD Gain Comparison for 1U and 2U CubeSat Geometries with Constant $K_d = 2 \times 10^{-5}$ Nms/rad and Different Values of K_p [119]

This iterative gain tuning process can be repeated as desired. For example, if one were to use Figure 64 and continue to tune the gains for the 2U CubeSat, then one would select the lowest settling time (corresponding to $K_p = 7 \times 10^{-9}$ Nm) and, holding this gain fixed, vary the K_d values. As an example, Figure 65 plots the corresponding settling times for the 2U CubeSat as a function of K_d showing that the lowest settling time of 13,707 seconds (2.5 orbits) corresponds to $K_d = 9 \times 10^{-6}$ Nms/rad.

Tuning was stopped at this stage for the scope of this thesis, although the iterative process may be continued if desired. For the simulation conditions used, it can be concluded that the selection of magnetic PD attitude controller gains is far more sensitive to both changes in satellite geometry and combinations of control gains when compared to a magnetic B-dot detumbling controller. Improper PD control gain selection could result in significant settling times or an ineffective attitude control system.

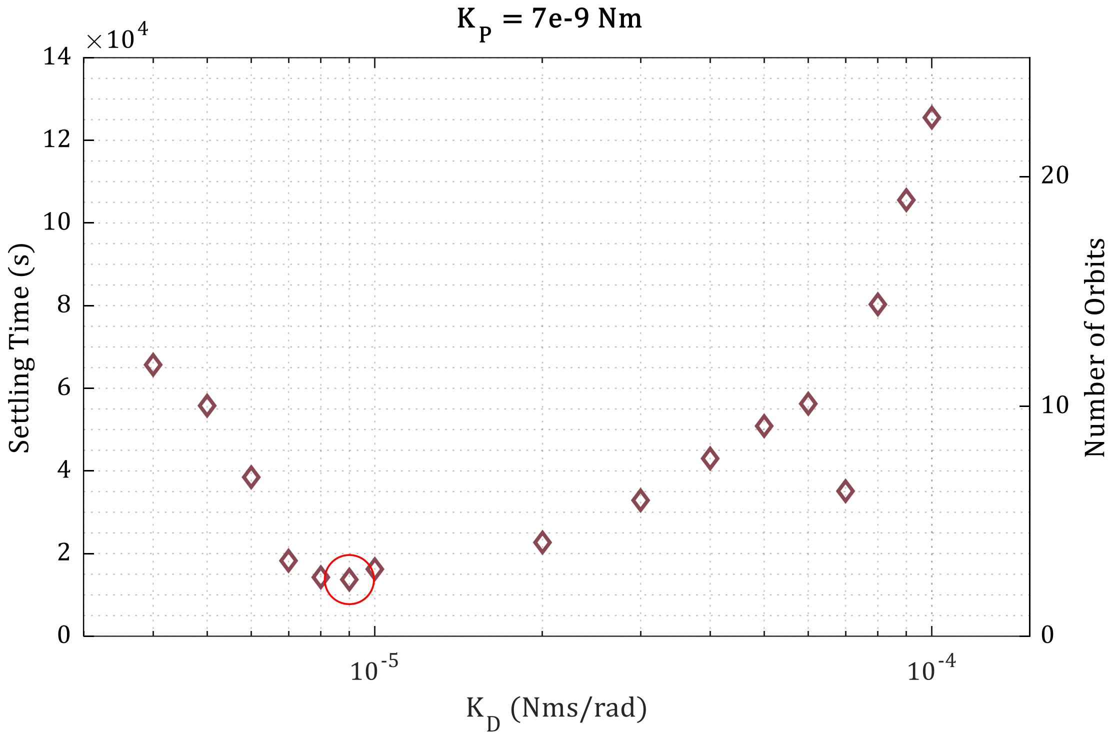


Figure 65: Figure 66. PD settling time for 2U CubeSat with constant $K_p = 7 \times 10^{-9}$ Nm and different values of K_d [119]

This section assumed no disturbance torques and perfect attitude knowledge with no sensor noise in its analyses. More realistically, the controller will be acting on noisy inputs gleaned from the q-Method attitude determination algorithm and the gyroscope. As magnetic PD control performance notably degrades in the presence of noise, next section proposes a method to improve the performance of this control under non-ideal conditions.

6.2 Extended Kalman Filter – Magnetic Attitude Control

Improving magnetic PD controller performance and stability necessitates measures taken to reduce the impact of sensor noise. In lieu of directly using raw gyroscope sensor data and raw output from the q-Method algorithm in the PD controller, this raw data was applied to an extended Kalman filter (EKF) which was implemented in the simulator with the express purpose of producing cleaner input for use with the PD controller. Use of a Kalman filter to improve attitude estimates from QUEST/q-Method has been documented in the literature for CubeSats such as AntelSat [16], AAUSAT3 [12], InnoSat [124], and ITASAT [125].

The EKF and its internal nonlinear state-space dynamics model of the satellite are based on the “reduced quaternion model” proposed by Yang, expanded to include all four elements of the attitude quaternion [77, p. 90]. The EKF operates at a sample rate of 1 second, with a 10% model uncertainty added to the dynamic model by multiplying the satellite inertia matrix $\boldsymbol{J}$ by a factor of 1.10. In this section, identity matrices are denoted $\boldsymbol{I}_i$ and zero matrices are denoted $\emptyset_{m\times n}$.

The 10×1 state matrix $\boldsymbol{x}$ comprises the three components of the BF satellite angular rate $\boldsymbol{\omega}$ (from the gyroscope) in rad/s, the four components of the BF/ECI attitude quaternion $\mathbf{q}$ (from the q-Method), and the three components of the gyroscope bias vector $\boldsymbol{\beta}$:

$$\boldsymbol{x} = [\omega_x \quad \omega_y \quad \omega_z \quad q_x \quad q_y \quad q_z \quad q_s \quad \beta_x \quad \beta_y \quad \beta_z]^T \tag{60}$$

The execution of the EKF consists of a prediction phase and an update phase. For the former, an internal discrete-time state-space dynamics model of the satellite is used to estimate the current state of the system $\boldsymbol{x}_{k|k-1}$ from the previous estimate of the state $\boldsymbol{x}_{k-1|k-1}$ and previous control input $\boldsymbol{u}_{k-1}$ as follows [77, p. 91]:

$$\boldsymbol{x}_{k|k-1} = \boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_{k-1}) \tag{61}$$

where the subscript notation $k|k-1$ indicates that the current estimate at time step k is made given the previous measurement from the sensors at time step $k-1$. The function $\boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_{k-1})$ in Equation 84 is given by [77, p. 91]:

$$\boldsymbol{F}(\boldsymbol{x}_{k-1|k-1}, \boldsymbol{u}_{k-1}) = x_{k-1|k-1} + \begin{bmatrix} -\boldsymbol{J}^{-1}\boldsymbol{\omega}^{\mathrm{x}}_{k-1|k-1}(\boldsymbol{J}\boldsymbol{\omega}_{k-1|k-1}) + \boldsymbol{J}^{-1}\boldsymbol{u}_{k-1} \\ -\frac{1}{2}\boldsymbol{\omega}^{\mathrm{x}}_{k-1|k-1}\boldsymbol{q}_{k-1|k-1} + \frac{1}{2}q_{0_{k-1|k-1}}\boldsymbol{\omega}_{k-1|k-1} \\ -\frac{1}{2}\boldsymbol{\omega}^{T}_{k-1|k-1}\boldsymbol{q}_{k-1|k-1} \\ \emptyset_{3\times 3} \end{bmatrix} \mathrm{dt} \tag{62}$$

where $\boldsymbol{\omega}_{k-1|k-1}$, $\boldsymbol{q}_k$ and q_{0_k} are the angular velocity and quaternion elements of $\boldsymbol{x}_{k-1|k-1}$ and $\boldsymbol{u}_{k-1}$ is the applied control torque from the previous time step $k-1$. The superscript x in $\boldsymbol{\omega}^{\mathrm{x}}_{k-1|k-1}$ corresponds to the skew-symmetric matrix given by:

$$\boldsymbol{\omega}^{x} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{63}$$

The corresponding 7×1 output vector $\boldsymbol{y}_{k|k-1}$ is defined based on the estimated state $\boldsymbol{x}_{k|k-1}$ which includes only the angular rate and quaternion estimates [77, p. 92]:

$$\boldsymbol{y}_{k|k-1} = \boldsymbol{H}\boldsymbol{x}_{k|k-1} \tag{64}$$

where $\boldsymbol{H}$ is the observation (or measurement) matrix:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{I}_3 & \emptyset_{3\times3} & \emptyset_{3\times1} & \boldsymbol{I}_3 \\ \emptyset_{3\times3} & \boldsymbol{I}_3 & \emptyset_{3\times1} & \emptyset_{3\times3} \\ \emptyset_{1\times3} & \emptyset_{1\times3} & 1 & \emptyset_{1\times3} \end{bmatrix} \tag{65}$$

To obtain the covariance prediction matrix $\boldsymbol{P}_{k|k-1}$ containing the variance and correlation between states, the state transition Jacobian $\boldsymbol{F}_{k-1}$ is defined as [77, p. 93]:

$$\boldsymbol{F}_{k-1} = \begin{bmatrix} \boldsymbol{I}_3 - \boldsymbol{J}^{-1}\left(\boldsymbol{\omega}_{k-1|k-1}^{\mathrm{x}}\boldsymbol{J} - \left(\boldsymbol{J}\boldsymbol{\omega}_{k-1|k-1}\right)^{x}\right)\mathrm{dt} & \emptyset_{3\times3} & \emptyset_{3\times1} & \emptyset_{3\times3} \\ \frac{1}{2}\left(\boldsymbol{q}_{k-1|k-1}^{\mathrm{x}} + q_{0k-1|k-1}\boldsymbol{I}_3\right)\mathrm{dt} & \boldsymbol{I}_3 - \frac{1}{2}\boldsymbol{\omega}_{k-1|k-1}^{\mathrm{x}}\mathrm{dt} & \frac{1}{2}\boldsymbol{\omega}_{k-1|k-1}\mathrm{dt} & \emptyset_{3\times3} \\ -\frac{1}{2}\boldsymbol{q}_{k-1|k-1}^{T}\mathrm{dt} & -\frac{1}{2}\boldsymbol{\omega}_{k-1|k-1}{}^{T}\mathrm{dt} & 1 & \emptyset_{1\times3} \\ \emptyset_{3\times3} & \emptyset_{3\times3} & \emptyset_{3\times1} & \boldsymbol{I}_3 \end{bmatrix} \tag{66}$$

such that $\boldsymbol{P}_{k|k-1}$ can be calculated as [77, p. 93]:

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_{k-1}^{T} + \boldsymbol{L}\boldsymbol{Q}\boldsymbol{L}^{T} \tag{67}$$

where $\boldsymbol{P}_{k-1|k-1}$ is the covariance prediction matrix from the previous state estimate, $\boldsymbol{Q}$ is the process noise covariance matrix, and $\boldsymbol{L}$ is the control input matrix, defined for a sample time $t_s = 1$ as:

$$\boldsymbol{L} = \boldsymbol{I}_{10}t_s \tag{68}$$

As a result of the prediction phase, the EKF produces a state prediction $\boldsymbol{x}_{k|k-1}$and covariance prediction $\boldsymbol{P}_{k|k-1}$ for the current time step k using only the measurement of the states from previous time step $k-1$. The next step in the EKF is the update phase, which compares these predictions against measurements of the angular rates from the gyroscope and attitude quaternion from the q-Method $\boldsymbol{y}_k$, noting that gyroscope bias is not a measurable quantity. The output prediction error (or "innovation") $\widetilde{\boldsymbol{y}}_k$ is defined as [77, p. 93]:

$$\widetilde{\boldsymbol{y}}_k = \boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_{k|k-1} \tag{69}$$

The innovation covariance is then defined as follows [77, p. 93]:

$$\boldsymbol{S}_k = \boldsymbol{H}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^{T} + \boldsymbol{R} \tag{70}$$

where $\boldsymbol{R}$ is the measurement noise covariance matrix. The computation of the Kalman gain $\boldsymbol{K}_k$ is dictated by the invertibility of the innovation covariance. If $\boldsymbol{S}_k$ is invertible then [77, p. 93]:

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}^{T}\boldsymbol{S}_k^{-1} \tag{71}$$

else, $\boldsymbol{K}_k = \emptyset_{10\times7}$. Finally, the updated state estimate $\boldsymbol{x}_{k|k}$ and updated covariance prediction matrix $\boldsymbol{P}_{k|k}$ for time k are computed as follows only if a nonzero attitude estimate measurement from the q-Method algorithm is available at the current timestep.

$$\begin{aligned} \boldsymbol{x}_{k|k} &= \boldsymbol{x}_{k|k-1} + \boldsymbol{K}_k \widetilde{\boldsymbol{y}}_k \\ \boldsymbol{P}_{k|k} &= (\boldsymbol{I}_{10} - \boldsymbol{K}_k \boldsymbol{H})\boldsymbol{P}_{k|k-1} \end{aligned} \tag{72}$$

If an attitude estimate is not available as a measured state, the state prediction is not updated and is output as computed at the end of the prediction phase:

$$\begin{aligned} \boldsymbol{x}_{k|k} &= \boldsymbol{x}_{k|k-1} \\ \boldsymbol{P}_{k|k} &= \boldsymbol{P}_{k|k-1} \end{aligned} \tag{73}$$

The values in the measurement noise covariance matrix $\boldsymbol{R}$ are based on the variances obtained through experiment for each state's associated sensor. The values in the process noise covariance matrix $\boldsymbol{Q}$ were empirically set to be an order of 10 larger than R to improve EKF performance:

$$\boldsymbol{Q} = \mathrm{diag}\begin{bmatrix} 0.00003 \\ 0.00003 \\ 0.00002 \\ 0.02 \\ 0.02 \\ 0.10 \\ 0.04 \\ 1 \\ 1 \\ 1 \end{bmatrix} \times 10^{-3} \qquad \boldsymbol{R} = \begin{bmatrix} 0.000003 \\ 0.000003 \\ 0.000002 \\ 0.002 \\ 0.002 \\ 0.01 \\ 0.004 \end{bmatrix}$$

6.2.1 q-Method Problems and EKF Solutions

The attitude estimate provided by the q-Method algorithm becomes predictably worse with the introduction of sensor noise into the system. Further to that, three limitations that can cause major issues with control were discovered during simulator development. The first issue concerns the availability of sensor measurements and how attitude is handled in the simulator when it cannot be estimated, the second issue relates to the duality property of the quaternion and the resultant control instability, and the third issue involves compensating for geometric limitations within the q-Method algorithm itself. All three solutions proposed in this thesis involve use of the EKF attitude prediction in tandem with that from the q-Method.

6.2.1.1 Attitude Availability

The first issue relates to the requirement that at least three Sun sensor measurements be available at each 1 second sample rate for the q-Method algorithm to calculate the current orientation quaternion. When the Sun vector is not available due to eclipse or shadowing of more than 15 of the 18 sensors at once, the q-Method does not have enough information to calculate the quaternion and produces Not a Number (NaN) values. Predictably, propagating a NaN through to the controllers causes countless numerical errors throughout the simulator. To avoid such errors, the NaN quaternion is overwritten with an invalid quaternion where all elements are set to zero to flag the EKF downstream to omit the update phase. Figure 67 illustrates noiseless q-Method output showing an example of periods where the attitude cannot be determined and is therefore overwritten with the zero quaternion for a freely tumbling satellite, where q_s corresponds to the scalar portion of the quaternion and q_x, q_y and q_z correspond to the vector portion of the quaternion.

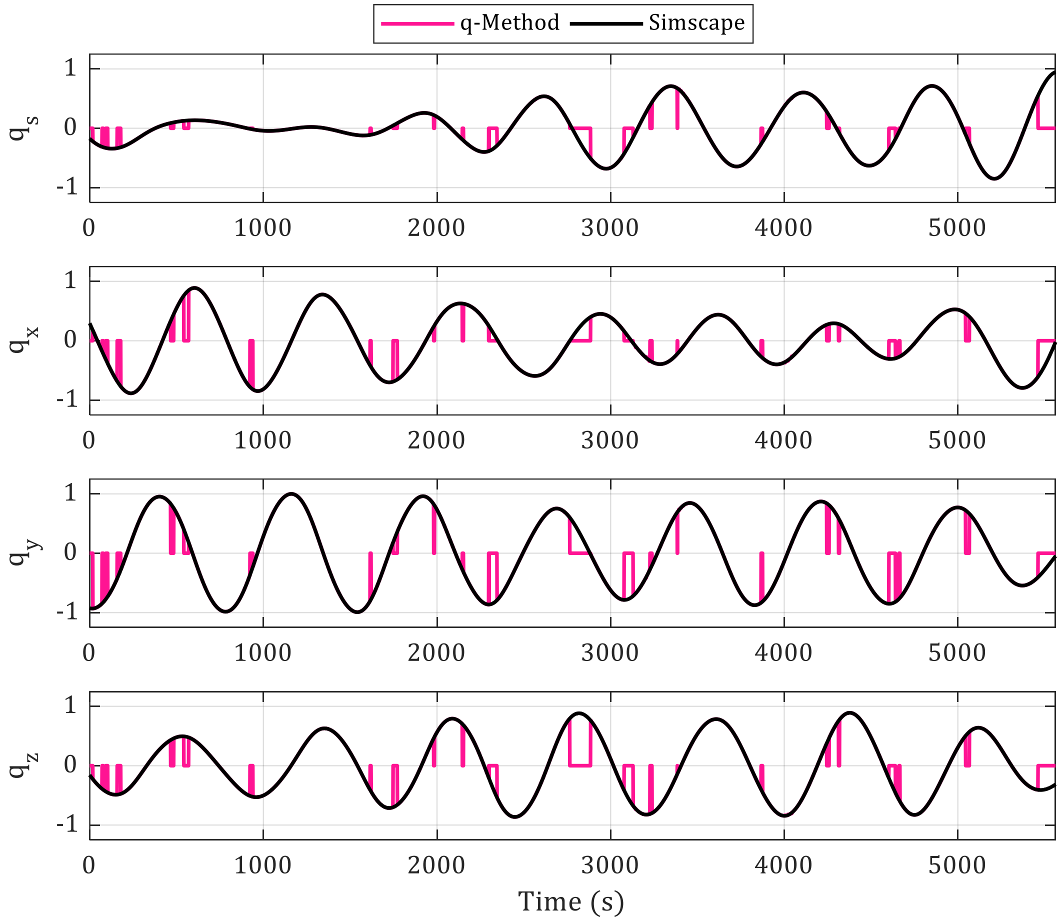


Figure 67: q-Method Issues – Availability of Two Distinct Measurement Input Vectors

This missing data can be further improved upon by using the Kalman filter prediction. When all four incoming attitude states are zero, the prediction state and covariance matrices are output from the EKF without being updated.

6.2.1.2 Quaternion Duality

The second issue with the q-Method estimate is the issue of duality. Quaternions are not unique representations of a given attitude; a quaternion and its negative counterpart represent equivalent orientations, one along the positive direction of its axis with a counter-clockwise rotation angle and one along the negative direction of its axis with a clockwise rotation angle. This duality is shown in Figure 68 which compares the q-Method quaternion estimate to Simscape's quaternion output. The q-Method quaternion appears to "flip" somewhat indiscriminately, though its clear the profile of the flipped portions is the negative of the non-flipped portions. Rapid sign changes will negatively affect the controller and the directions of the torques it produces, potentially destabilising the satellite through over-actuation under the assumption that the attitude error between timesteps is much greater than it is.

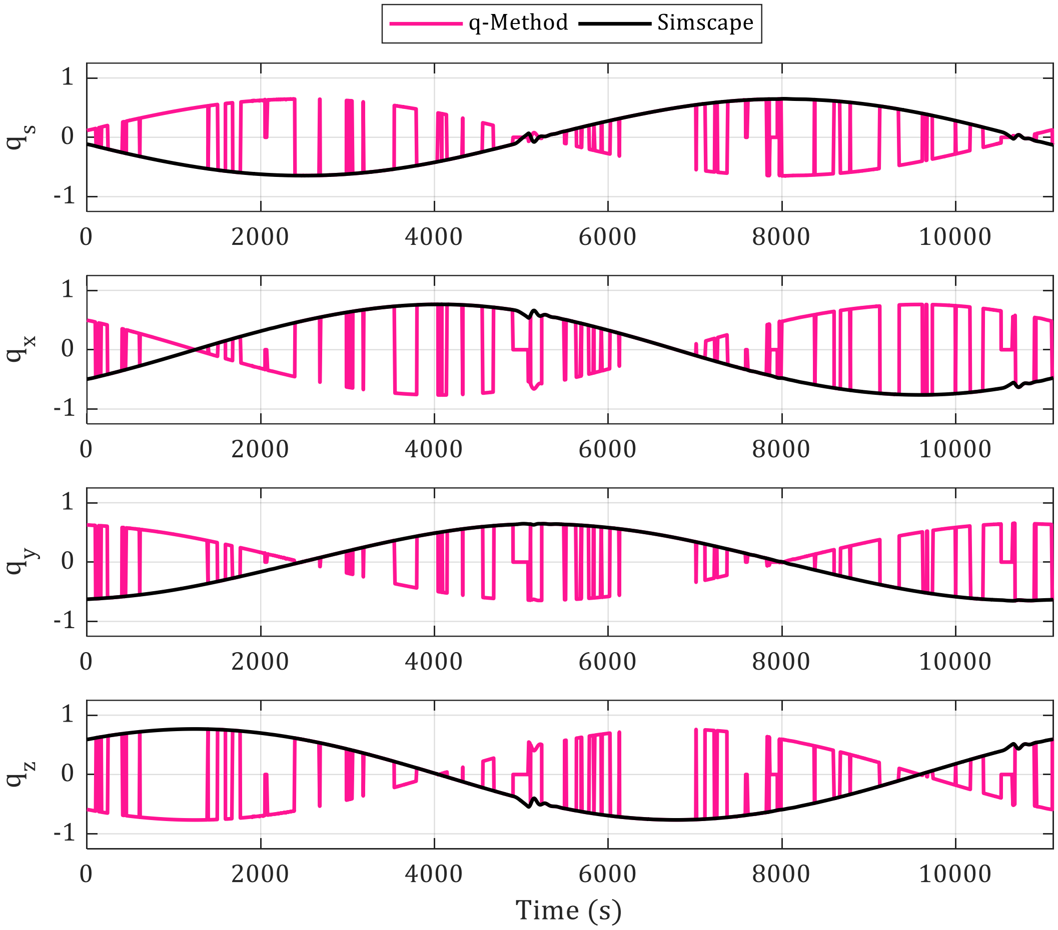


Figure 68: q-Method Issues – Quaternion Duality

This problem is expounded for the QUEST algorithm by authors Campos and Furtado [126]. Their analysis proposes a solution based on use of the second derivative of the quaternion scalar element to identify discontinuities in the quaternion output signal: if the second derivative of the scalar is greater than some limit value, then a “point filter” inverts the output until the next discontinuity is detected [126]. Their solution, however, falters when noise is added to the system, with results noting a minimum signal-to-noise ratio of 160 dB required for an accurately corrected quaternion to be obtained [126]. A more robust solution was therefore sought that did not require taking the derivative of the noisy signal.

A logic condition is therefore proposed in this thesis that compares the q-Method output against the attitude estimate from the EKF. If the 2-norm of the difference between the q-Method quaternion and EKF prediction is greater than the 2-norm of the difference between the negative q-Method quaternion and EKF prediction, the q-Method quaternion is flipped. Figure 69 demonstrates this condition. In Case 1, the difference between the q-Method and EKF quaternions is smaller than the difference between the negative q-Method and EKF quaternions, and thus the q-Method estimate would not be flipped. For Case 2, the difference between the q-Method and EKF quaternions is larger than the difference between the negative q-Method and EKF quaternions, and thus the q-Method quaternion would be flipped.

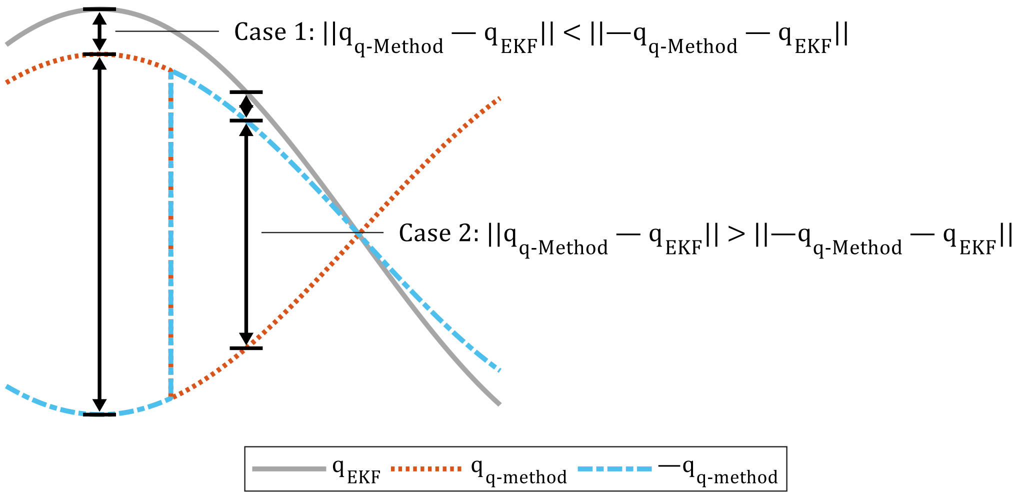


Figure 69: Quaternion Duality Logic Condition

6.2.1.3 Parallel Inputs

The third issue with the q-Method estimate relates to the relative orientation of the pairs of its input vectors corresponding to the measurement and reference Sun vectors as well as the measurement and reference Earth magnetic field vectors. When either the two measurement vectors or two reference vectors approach parallel, the attitude is rendered not uniquely determinable. For the noiseless q-Method results shown in Figure 70 the satellite was held at a zero-error pointing attitude for two orbits. The top plot shows the angle between the Sun reference vector and IGRF-12 vector – both expressed in ECI – approaching 180 degrees around the end of the first orbit, and the quaternion scalar and vector components in the plots below worsening as a result. Peak Euler angle errors are $[4.5° \quad 3° \quad 1°]$

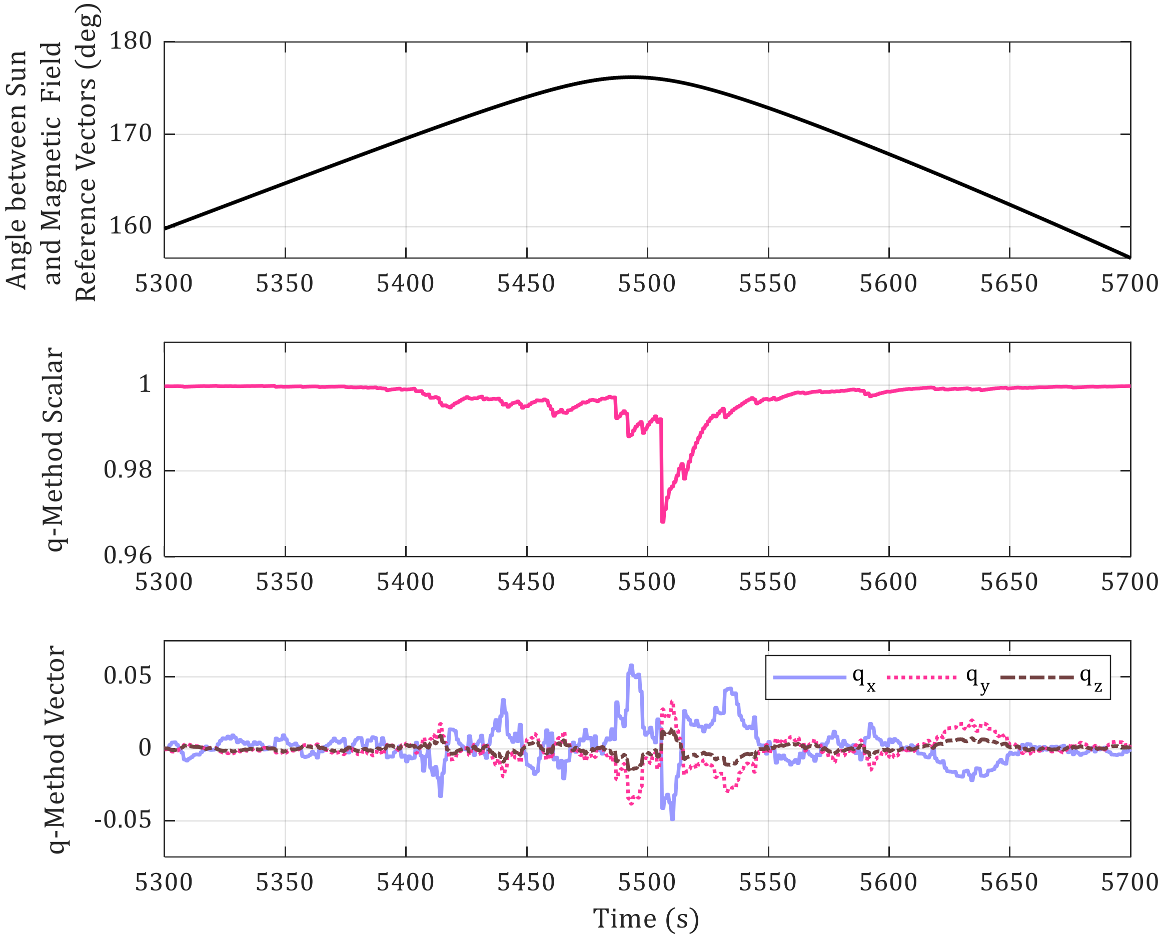


Figure 70: q-Method Issues – Parallel Inputs

To overcome this issue, this thesis proposes to incorporate the following logic: When the input vectors are directed within 10° of parallel to one another (i.e. $> 170°$ or $< 10°$), the q-Method outputs a zero quaternion which, similar to the first issue, flags the downstream EKF to omit the update phase.

6.2.2 EKF Results

With the three solutions to the issues with attitude availability implemented, the top subplots of Figure 71 and Figure 72 show the raw noisy gyroscope angular rates and q-

Method attitude estimates, respectively, while the bottom subplots of these figures illustrate the improvements achieved with the EKF. Results are shown for one orbit excluding eclipse.

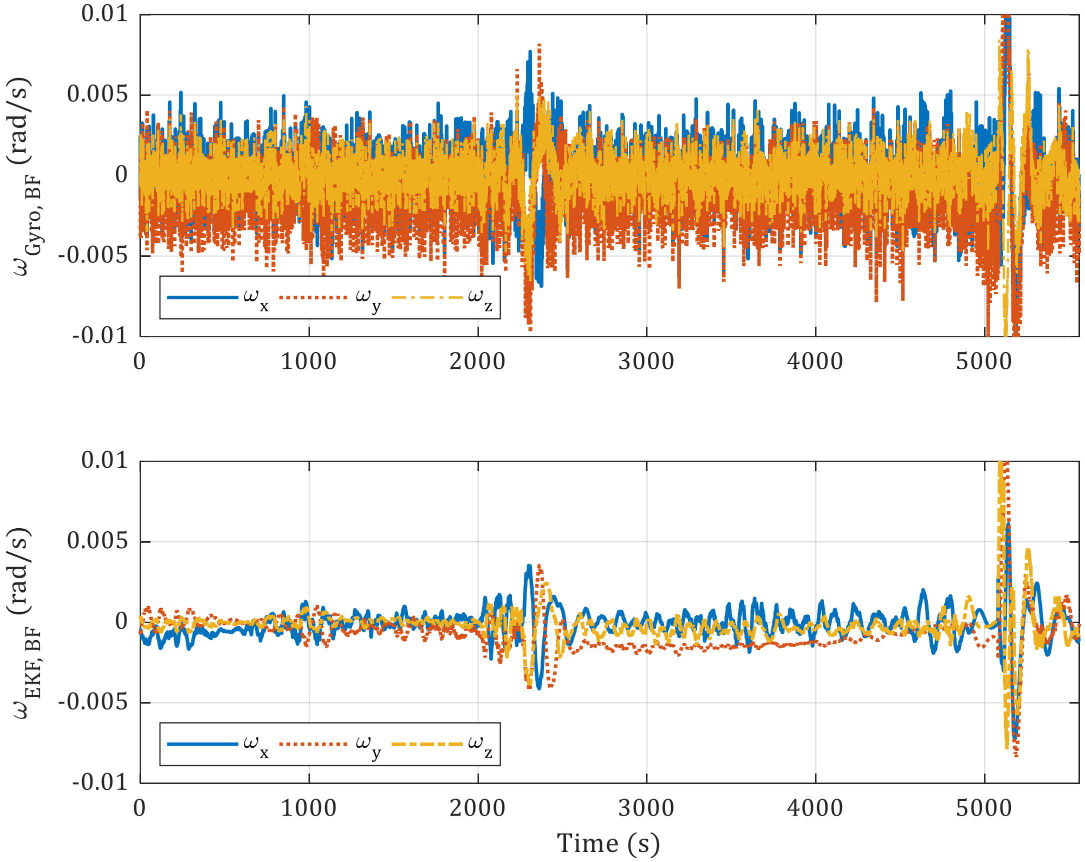


Figure 71: Comparison between Gyroscope Body Angular Rates and EKF Body Angular Rates Over One Orbit

The significant magnitude of noise in the upper plot of Figure 71 is reduced to approximately the same level as the orbital rate (0.001130 rad/s) while still following the same general profile. A lower noise amplitude will help improve the precision of the magnetic PD control torques and reduce settling time overall by eliminating the potential issue of overcompensating for noise masquerading as a higher rate. Similarly, the results shown in Figure 72 comparing the noisy q-Method output with the three issues fixed as outlined greatly reduce the sensor noise and predict the attitude within a fair degree of accuracy when it is unavailable.

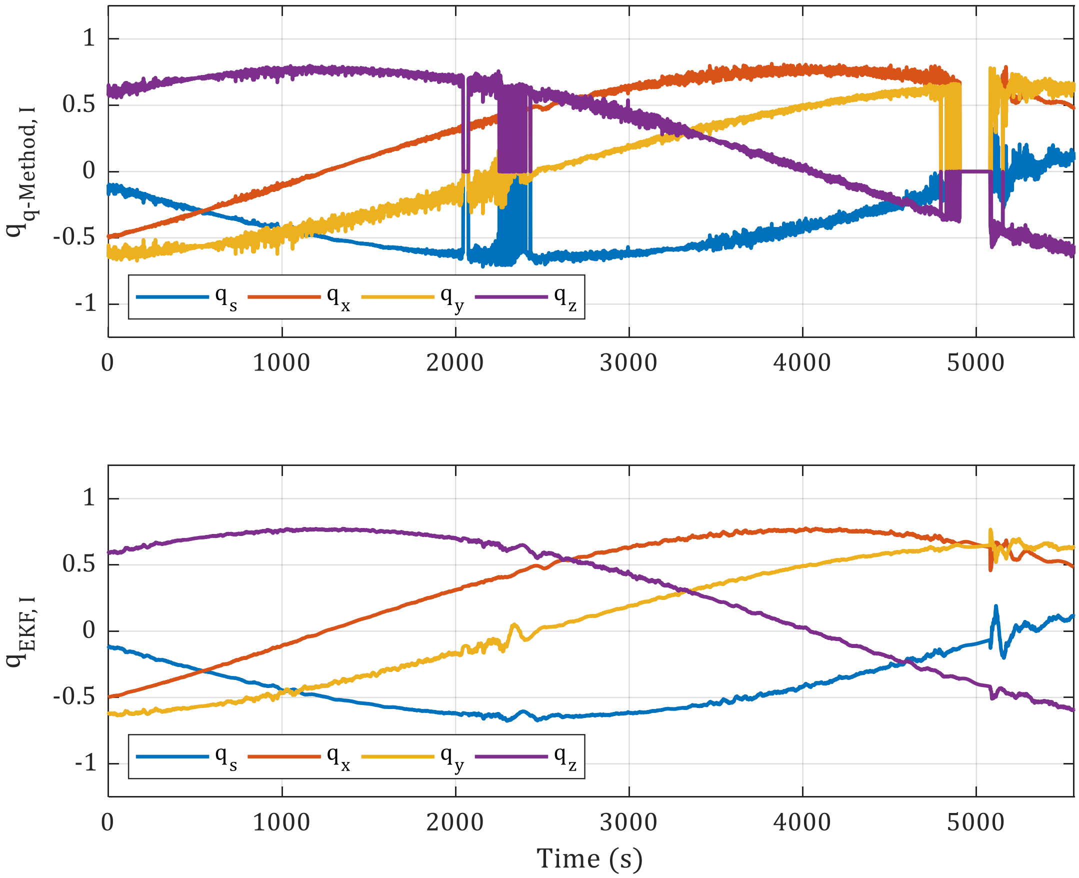


Figure 72: Comparison between q-Method Attitude Quaternion and EKF Attitude Quaternion Over One Orbit

Development and validation of the PD controller and its associated attitude determination algorithms has heretofore assumed the absence of external disturbance torques. The next section outlines the issues that arise when disturbance torques are introduced into the simulation.

6.2.3 Problems with Magnetic PD Control

For a typical magnetic PD simulation for the 2U prism geometry using the derived best-case gains, no environmental disturbance torques, noisy sensors, EKF error inputs to the controller, and a 90% magnetorquer duty cycle, the results in terms of attitude and rate errors are produced as shown in Figure 73 for an initial angular rate condition of 0.1 rad/s about all axes. In the beginning oscillatory region, the maximum magnitude of applied magnetic moment is approximately 6×10^{-3} Am^2, resulting in a maximum applied magnetic PD control torque magnitude of approximately 1.5×10^{-7} Nm. The steady-state region beginning around 3.5 orbits has an average pointing error of $9.1888 \pm 2.4591°$, falling generally within the 10° pointing accuracy requirement for LORIS.

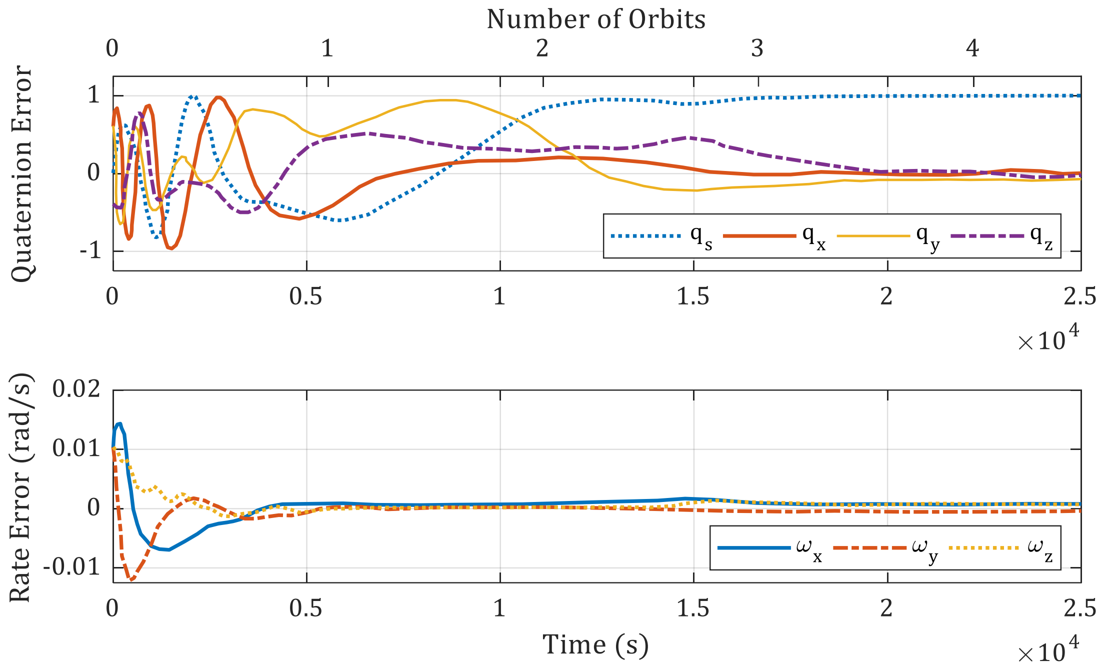


Figure 73: Magnetic PD Control Results with Best-Case Gains, Noisy Sensor Data, and No External Disturbances

Until this point, results have been gradually worsening as the simulation becomes less idealised with the additions of sensor noise and imperfect estimations of attitude and rate. However, even with such modifications to the idealised case, settling time has not increased significantly versus previous analyses. The addition of disturbance torques is the next logical step in making the simulation more realistic.

The estimated magnitude of the worst-case disturbance torques from Section 3.3.5 sums, at its peak, to a value of 7.16×10^{-7} Nm. The order of magnitude of the peak magnetic control torque generated in the above simulation was 1.5×10^{-7} Nm – implying that the nominal magnetic control torque is weaker than the nominal external disturbance. Applying a sinusoidal worst-case disturbance torque one order of magnitude lower than the maximum (amplitude 1×10^{-8}Nm) to the satellite with the same simulation conditions as the previous case produced the results shown in Figure 74. Though the rate error in the bottom plot attenuates, neither it nor the attitude error reach a steady-state value. The satellite instead tumbles at a low rate, with no indication that it may eventually settle to its desired value. The maximum magnetic moment applied is unchanged at 6×10^{-3} Am^2, as is the maximum magnetic control torque of approximately 1.5×10^{-7} Nm.

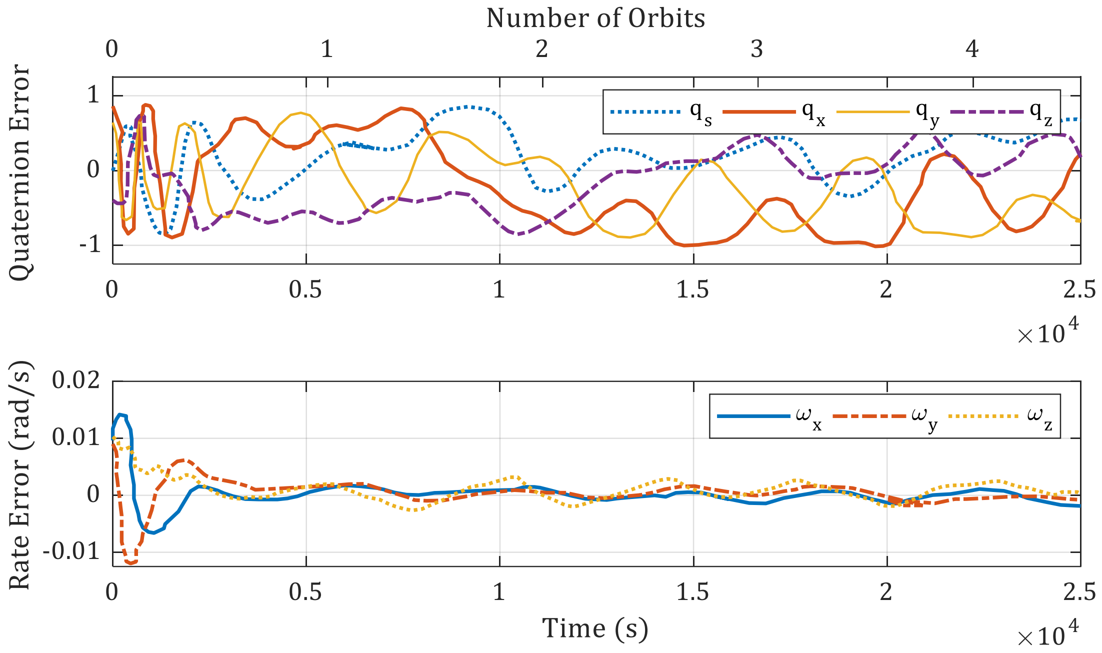


Figure 74: Magnetic PD Control Results with Best-Case Gains, Noisy Sensor Data, and Worst-Case External Disturbances

Attempting to counter the disturbance torques by increasing the gains only serves to further destabilise the satellite; magnetically torqueing too strongly oscillates the satellite back and forth too quickly, and with the inability to apply the exact calculated desired torque, the satellite merely ends up oscillating instead of settling to its desired steady state. Thus, for the conditions used in this research, the magnetic PD controller cannot be improved by simply increasing the gains and moving the controller's magnetic moments closer to their saturation range. To obtain behaviour similar to that in Figure 73, the lowest possible sinusoidal disturbance torque that can be added to the simulation is of magnitude 4×10^{-9} Nm: two orders of magnitude smaller than the estimated maximum.

At the preliminary design review for the LORIS CubeSat, the Canadian Space Agency suggested that accurate pointing control using only magnetorquers may not be attainable under realistic environmental conditions. One potential solution is to incorporate reaction wheels into the satellite ADCS design. The next chapter, therefore, details the reaction wheel design process for a three-axis stabilised PD control scheme capable of overcoming these environmental disturbances.

Chapter 7: Reaction Wheel Pointing Control

One of the most common solutions to the three-axis control problem is the implementation of a flywheel-based actuator system. Reaction wheels are flywheels attached to DC motors whose individual speeds are controlled to produce torques about their respective axes of rotation. Unlike magnetic torque rods, the control torque strength of reaction wheels is independent of external factors like magnetic field direction. This chapter presents the design of a reaction wheel system consisting of three orthogonally-oriented reaction wheels (one for each BF axis) capable of countering the worst-case disturbance torques presented in Section 3.3.5. Limitations on the use of the reaction wheels are then discussed, and operational results are presented.

7.1 Reaction Wheel Design

The reaction wheels are designed based principally on rejection of the estimated worst-case disturbance torques. Assuming a worst-case disturbance torque magnitude of $T_{WCD} = 7.16 \times 10^{-7}$ Nm from Section 3.3.5, the maximum countering reaction wheel torque required with a design margin of 10 is determined as [105, p. 357]:

$$T_1 = 10T_{WCD} = 7.16 \times 10^{-6}\text{ Nm} \tag{74}$$

The momentum storage required to reject this worst-case torque over a fraction $x = 0.5$ of the 5,562 s orbital period P is then defined as [105, p. 357]:

$$H = 0.637T_1Px = 0.013\text{ Nms} \tag{75}$$

where 0.637 is the root mean square average value of a sinusoidal function and setting $x = 0.5$ assumes that the sum of all disturbance torques is cyclic over half an orbit with period $P = 5{,}562$ s. This momentum storage is then used to obtain the desired moment of inertia of the reaction wheel capable of reaching a maximum wheel speed of $\omega_{max} = 10{,}000$ RPM (1047 rad/s) as follows [105, p. 357]:

$$I_{RW} = \frac{H}{\omega_{max}} = 1.21 \times 10^{-5}\text{ kgm}^2 \tag{76}$$

The flywheel radius was chosen empirically as $r_{RW} = 0.016$ m, to resolve a solid cylindrical design wider than it is tall. Using the common density of brass of $\rho_{RW} = 8{,}730$ kg/m^2, a mass and wheel height are obtained as follows:

$$m_{RW} = \frac{2I_{RW}}{r_{RW}^2} = 0.095\text{ kg} \tag{77}$$

$$h_{RW} = \frac{m_{RW}}{\pi r_{RW}^2 \rho_{RW}} = 0.013\text{ m} \tag{78}$$

The final flywheel design is summarised in Table 12.

Table 12: Reaction Wheel Design Parameters

Parameter	Variable	Value
Maximum Torque with Design Margin	T_1	7.16×10^{-6} Nm
Momentum Storage Requirement	H	0.013 Nms
Flywheel Moment of Inertia	I_{RW}	1.21×10^{-5} kgm^2
Flywheel Radius	r_{RW}	0.016 m
Flywheel Mass	m_{RW}	0.095 kg
Flywheel Height	h_{RW}	0.013 m

These design parameters are used to determine a simplified first-order DC motor model in the next section, which outlines the PD control scheme for the reaction wheel system.

7.2 Reaction Wheel PD Control

The magnetic PD controller defined in Section 6.1 – with the proportional gain acting on the vector portion of the EKF quaternion attitude error and the derivative gain acting on the EKF angular rate error – was reappropriated for use with the reaction wheels. The new controller, defined for each wheel via the following equations, computes a scalar reaction wheel motor voltage V_c which saturates at $\pm$ 5 V.

$$V_{c,x} = -K_p \hat{q}_x - K_d \widehat{\omega}_x \tag{79}$$

$$V_{c,y} = -K_p \hat{q}_y - K_d \widehat{\omega}_y \tag{80}$$

$$V_{c,z} = -K_p \hat{q}_z - K_d \widehat{\omega}_z \tag{81}$$

These voltages are set to change once per second to match the sample rate of the EKF inputs. The voltages are input to a simplified first-order DC motor model, with steady-state gain K_{ss} and time constant τ defined with respect to the maximum reaction wheel speed ω_{max} and orbital period P:

$$K_{ss} = \frac{\omega_{max}}{5\ \mathrm{V}} = 209.4\ \frac{\mathrm{rad}}{\mathrm{V \cdot s}} \tag{82}$$

$$\tau = \frac{P/2}{5} = 556.2\ \mathrm{s} \tag{83}$$

The steady-state gain and time constant define the following first-order transfer function:

$$G(s) = \frac{209.4}{556.2s + 1}$$

Figure 75 plots the resulting 5 V simulated step response for the reaction wheel system, showing a rise time of 1,221 seconds and a settling time of 2,190 seconds (2% settling time criterion).

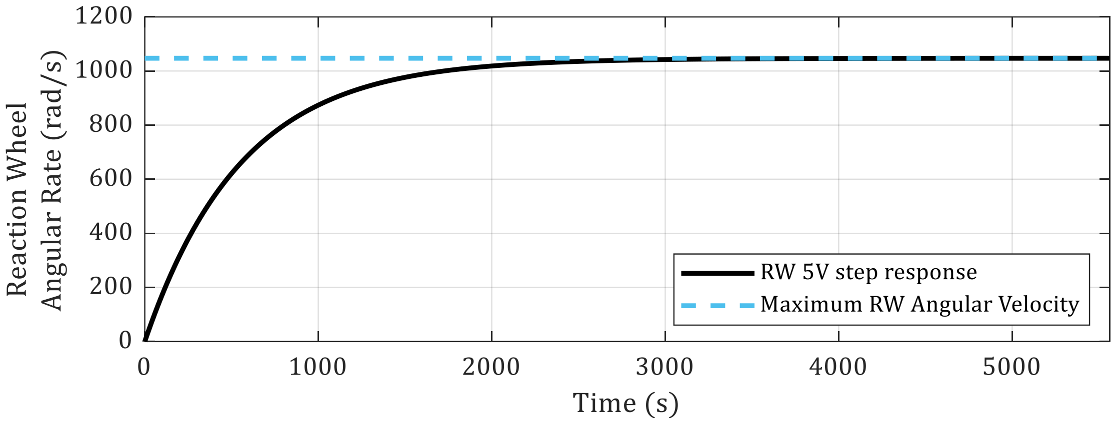


Figure 75: Reaction Wheel 5 V Step Response

The corresponding required motor torque as a function of time to achieve the angular rates in Figure 98 is plotted in Figure 76 as follows. This value falls well within the range of the rated torque for the FAULHABER Series 2610 B reaction wheel motors selected for LORIS at 2.85 mNm [127]:

$$T_{max} = 2.2805 \times 10^{-5} \text{ Nm}$$

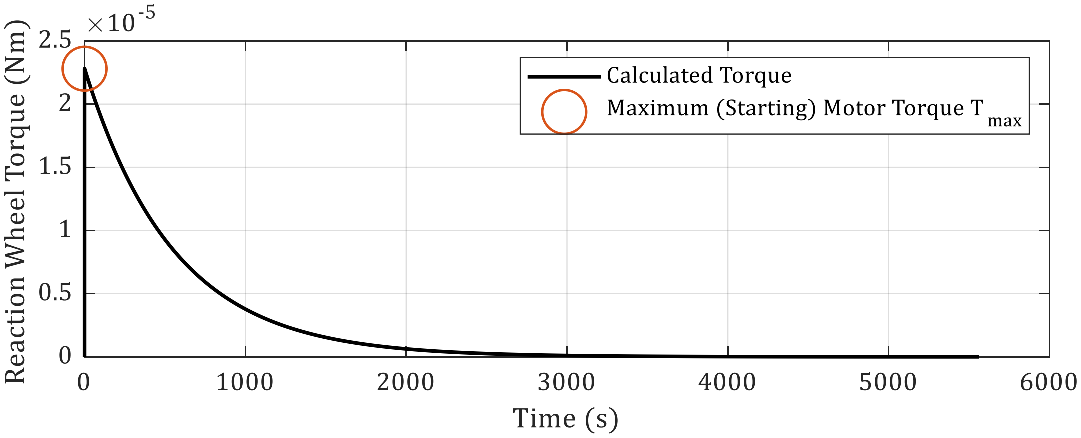


Figure 76: Reaction Wheel Torque from 5 V Step Response

Lastly, Figure 77 plots the torque-speed curve for the motors. Speeds up to 600 rad/s (5730 RPM) correspond to torques that exceed the maximum external disturbance torque by two orders of magnitude – offering a wide range of moderate wheel speeds guaranteed to produce torques that can overcome the disturbances.

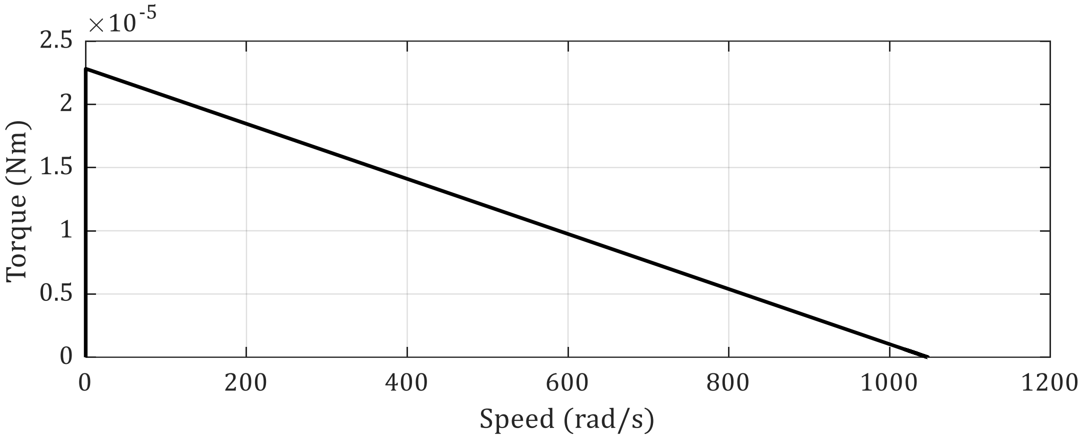


Figure 77: Reaction Wheel Torque-Speed Curve

7.3 Extended Kalman Filter – Reaction Wheel Attitude Control

The addition of reaction wheels to the ADCS introduces additional inertias and dynamics into the system, requiring revision of the state-space system model and its associated EKF first presented in Section 6.2. The revised EKF is based on Yang's "reaction wheel desaturation control" expansion to his reduced quaternion dynamics model EKF [77, p. 151]. Beyond expanding the state-space model to include reaction wheel dynamics, the formulation of the EKF is identical to that presented in Section 6.2.

A second inertia quantity $\boldsymbol{J}_{RW}$ representing the inertia matrix of the three reaction wheels aligned with the satellite BF axes is introduced [77, p. 152]. Using the following formula for the transverse moment of inertia of a cylinder, the reaction wheel inertia matrix is defined:

$$J_{RW} = \frac{1}{12} m(3r_{RW}^2 + h_{RW}^2) \tag{84}$$

$$\boldsymbol{J}_{RW} = \begin{bmatrix} 1.4977 \times 10^{-5} & 0 & 0 \\ 0 & 1.4977 \times 10^{-5} & 0 \\ 0 & 0 & 1.4977 \times 10^{-5} \end{bmatrix} \text{kgm}^2$$

The formulation of this matrix assumes that the three reaction wheels coincide at the satellite's centre of mass, as illustrated in Figure 78. The configuration of the reaction wheels in the simulator is shown in the figure adjacent, with the three wheels offset from the geometry's centre of mass by -0.03 m in their corresponding BF directions. Superimposed reaction wheels are a physical impossibility; thus, the dynamic model contains inherent inaccuracies propagating within the EKF.

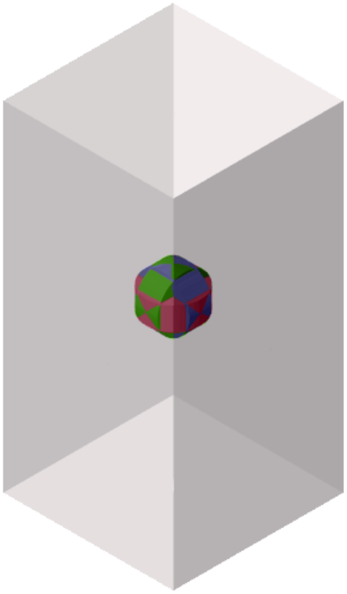


Figure 78: Reaction Wheel Configuration Assumed for Dynamic System Model with Reaction Wheel Dynamics

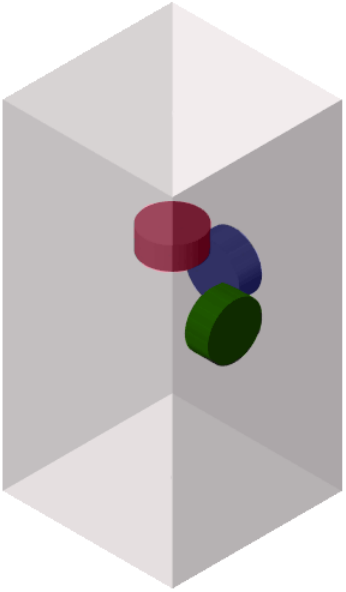


Figure 79: Actual Reaction Wheel Configuration for Dynamic System Model used in Simulation

Both the satellite inertia matrix $\boldsymbol{J}$ and the reaction wheel inertia matrix $\boldsymbol{J}_{RW}$ are assumed diagonal. A 10% model uncertainty is added to the EKF by multiplying both inertia matrices by a factor of 1.10.

Three additional states are added to the original state matrix from Equation 60 – the BF X, Y, and Z reaction wheel speeds Ω in rad/s – to define the new 13 × 1 state matrix $\boldsymbol{x}$:

$$\boldsymbol{x} = [\Omega_x \quad \Omega_y \quad \Omega_z \quad \omega_x \quad \omega_y \quad \omega_z \quad q_x \quad q_y \quad q_z \quad q_s \quad \beta_x \quad \beta_y \quad \beta_z]^T \tag{85}$$

The current state of the system $\boldsymbol{x}_{k|k-1}$ is estimated via the new dynamic model, where the first equation defines the reaction wheel speeds with respect to their generated torques and the second defines the overall satellite angular rates to incorporate both control torques from the magnetorquers $\boldsymbol{u}_{k-1}$ (set to zero) and from the reaction wheels $\boldsymbol{T}_{c,RWk-1}$.

$$\boldsymbol{x}_{k|k-1} = \boldsymbol{x}_{k-1|k-1} + \begin{bmatrix} \boldsymbol{J}_{RW}\boldsymbol{T}_{c,RWk-1} \\ -\boldsymbol{J}^{-1}\boldsymbol{\omega}^{\mathrm{x}}_{k-1|k-1}\left(\boldsymbol{J}\boldsymbol{\omega}_{k-1|k-1} + \boldsymbol{J}_{RW}\boldsymbol{\Omega}_{k-1|k-1}\right) + \boldsymbol{J}^{-1}\boldsymbol{u}_{k-1} - \boldsymbol{J}^{-1}\boldsymbol{T}_{c,RWk-1} \\ -\frac{1}{2}\boldsymbol{\omega}^{\mathrm{x}}_{k-1|k-1}\boldsymbol{q}_{k-1|k-1} + \frac{1}{2}q_{0_{k-1|k-1}}\boldsymbol{\omega}_{k-1|k-1} \\ -\frac{1}{2}\boldsymbol{\omega}^{T}_{k-1|k-1}\boldsymbol{q}_{k-1|k-1} \\ \emptyset_{3\times 3} \end{bmatrix} \mathrm{dt} \tag{86}$$

The 10×1 output vector $\boldsymbol{y}_{k|k-1}$ is defined as before with the addition of the reaction wheel angular rates:

$$\boldsymbol{y}_{k|k-1} = \boldsymbol{H}\boldsymbol{x}_{k|k-1} \tag{87}$$

where observation matrix $\boldsymbol{H}$ is defined:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{I}_3 & \emptyset_{3\times3} & \emptyset_{3\times3} & \emptyset_{3\times1} & \emptyset_{3\times3} \\ \emptyset_{3\times3} & \boldsymbol{I}_3 & \emptyset_{3\times3} & \emptyset_{3\times1} & \boldsymbol{I}_3 \\ \emptyset_{3\times3} & \emptyset_{3\times3} & \boldsymbol{I}_3 & \emptyset_{3\times1} & \emptyset_{3\times3} \\ \emptyset_{1\times3} & \emptyset_{1\times3} & \emptyset_{1\times3} & 1 & \emptyset_{1\times3} \end{bmatrix} \tag{88}$$

Calculating the covariance prediction matrix $\boldsymbol{P}_{k|k-1}$ requires the definition of the new state transition Jacobian $\boldsymbol{F}_{k-1}$:

$$\boldsymbol{F}_{k-1} = \cdots$$
$$\begin{bmatrix} \boldsymbol{I}_3 & \emptyset_{3\times3} & \emptyset_{3\times3} & \emptyset_{3\times1} & \emptyset_{3\times3} \\ -\boldsymbol{J}^{-1}\boldsymbol{\omega}_{k-1|k-1}^{x}\boldsymbol{J}_{RW}\mathrm{dt} & \boldsymbol{I}_3 - \boldsymbol{J}^{-1}(\boldsymbol{\omega}_{k-1|k-1}^{x}\boldsymbol{J} - \left(\boldsymbol{J}\boldsymbol{\omega}_{k-1|k-1} + \boldsymbol{J}_{RW}\boldsymbol{\Omega}_{k-1|k-1}\right)^{x})\mathrm{dt} & \emptyset_{3\times3} & \emptyset_{3\times1} & \emptyset_{3\times3} \\ \emptyset_{3\times3} & \frac{1}{2}\left(\boldsymbol{q}_{k-1|k-1}^{x} + q_{0_{k-1|k-1}}\boldsymbol{I}_3\right)\mathrm{dt} & \boldsymbol{I}_3 - \frac{1}{2}\boldsymbol{\omega}_{k-1|k-1}^{x}\mathrm{dt} & \frac{1}{2}\boldsymbol{\omega}_k\mathrm{dt} & \emptyset_{3\times3} \\ \emptyset_{1\times3} & -\frac{1}{2}\boldsymbol{q}_{k-1|k-1}^{T}\mathrm{dt} & -\frac{1}{2}\boldsymbol{\omega}_{k-1|k-1}^{T}\mathrm{dt} & 1 & \emptyset_{1\times3} \\ \emptyset_{3\times3} & \emptyset_{3\times3} & \emptyset_{3\times3} & \emptyset_{3\times1} & \boldsymbol{I}_3 \end{bmatrix} \tag{89}$$

The covariance predication matrix $\boldsymbol{P}_{k|k-1}$ is then computed as follows:

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}_{k-1}^{T} + \boldsymbol{L}\boldsymbol{Q}\boldsymbol{L}^{T} \tag{90}$$

with the process noise covariance matrix $\boldsymbol{Q}$ and $\boldsymbol{L}$ defined for a sample time $t_s = 1$ as:

$$\boldsymbol{L} = I_{13}t_s \tag{91}$$

The innovation $\widetilde{\boldsymbol{y}}_k$ is defined as

$$\widetilde{\boldsymbol{y}}_k = \boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_{k|k-1} \tag{92}$$

with innovation covariance $\boldsymbol{S}_k$

$$\boldsymbol{S}_k = \boldsymbol{H}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^{T} + \boldsymbol{R} \tag{93}$$

where $\boldsymbol{R}$ is the measurement noise covariance matrix. If $\boldsymbol{S}_k$ is invertible, the Kalman gain is calculated via

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}^{T}\boldsymbol{S}_k^{-1} \tag{94}$$

otherwise, $\boldsymbol{K}_k = \emptyset_{13\times10}$. The updated state estimate $\boldsymbol{x}_{k|k}$ and covariance prediction matrix $\boldsymbol{P}_{k|k}$ are computed as follows if and only if a nonzero attitude estimate is available at the current timestep.

$$\begin{aligned}\boldsymbol{x}_{k|k} &= \boldsymbol{x}_{k|k-1} + \boldsymbol{K}_k\widetilde{\boldsymbol{y}}_k \\ \boldsymbol{P}_{k|k} &= (\boldsymbol{I}_{13} - \boldsymbol{K}_k\boldsymbol{H})\boldsymbol{P}_{k|k-1}\end{aligned} \tag{95}$$

If attitude is unavailable, nether prediction is updated:

$$\begin{aligned}\boldsymbol{x}_{k|k} &= \boldsymbol{x}_{k|k-1} \\ \boldsymbol{P}_{k|k} &= \boldsymbol{P}_{k|k-1}\end{aligned} \tag{96}$$

Values of the measurement noise covariance matrix $\boldsymbol{R}$ are again based on the variances obtained through experiment for each state's associated sensor, while the values for $\boldsymbol{Q}$ were tuned to improve EKF performance:

$$\boldsymbol{Q} = \text{diag}\begin{bmatrix}0.0003\\0.0003\\0.0002\\0.00003\\0.00003\\0.00002\\0.02\\0.02\\0.10\\0.04\\1\\1\\1\end{bmatrix}\times10^{-3} \qquad \boldsymbol{R} = \begin{bmatrix}0.000003\\0.000003\\0.000002\\0.000003\\0.000003\\0.000002\\0.002\\0.002\\0.01\\0.004\end{bmatrix}$$

7.4 Gain Tuning

Using the attitude and angular rate estimates from the EKF, the reaction wheel PD controller gains were tuned via the same process presented for the magnetic PD controller in Section 6.1.1. These gains were tuned for the final geometry in Figure 9 with sensor noise applied throughout and worst-case external disturbances present. It was assumed that pointing must be completed within the non-eclipse regions of the orbit and the satellite was considered settled and pointing once all three Euler angles crossed the $\pm10°$ accuracy threshold. Gains that settled with steady-state errors exceeding 5° were rejected with the preferred PD gain combination retaining stability (i.e. exhibiting non-oscillatory behaviour once settled), having a low steady-state error, and settling as fast as possible to leave the majority of the Sun period available for pointing.

Settling times were considered from a stationary initial condition with a 71° offset from Nadir. Figure 80 fixes an initial value of $K_p = 50/3 = 16.7$ V and varies K_d over two orders of magnitude. The settling times notably do not exhibit a parabolic profile as they did for the magnetic PD controller under ideal conditions, and instead appear to be randomly distributed. K_d values less than 20 V · s/rad and greater than 200 V · s/rad exhibited steady-state errors exceeding 5° and were deemed insufficient at attaining the desired attitude behaviour. The best-case derivative gain was selected as $K_d = 90$ V · s/rad with a settling time of 141 seconds within 10°. This gain further settled and maintained a pointing error within 5° in 263 seconds.

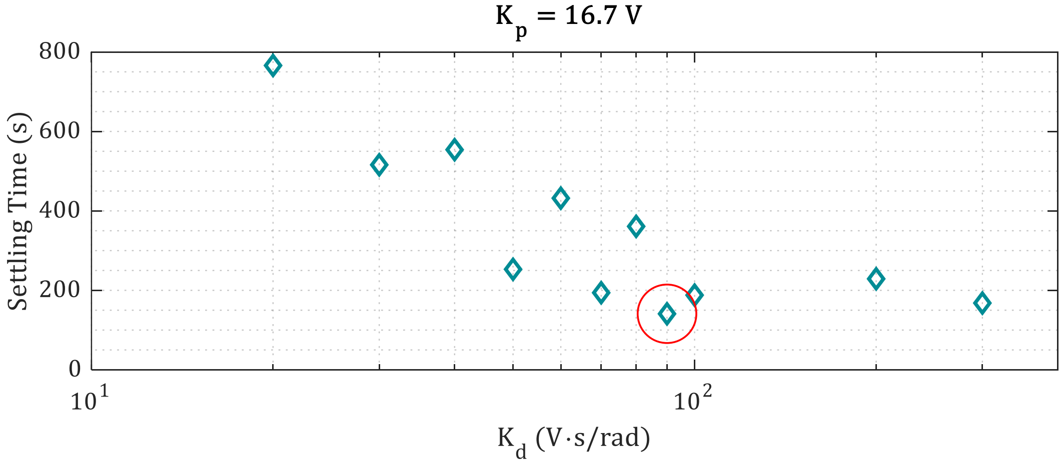


Figure 80: PD Gain Analysis with Constant $K_p = 50/3$ V and Different Values of K_d – Settling from Stationary Initial Body Angular Rates and 71° Offset from Nadir

Holding $K_d = 90$ V · s/rad constant, K_p values were then similarly analysed and plotted against their respective settling times in Figure 81. Again, limitations were observed for K_p values less than 11 V and greater than 20 V as they exhibited steady-state errors in excess of 5° and, in the latter case, displayed oscillations in their settled behaviour. The best-case gain remains unchanged from the original value of $K_p = 50/3 = 16.7$ V, settling within 10° in 141 seconds and within 5° in 263 seconds as previously reported. Thus, the best-case PD gains for the reaction wheel voltage controller are:

$$K_p = 50/3 \text{ V}$$

$$K_d = 90 \text{ V} \cdot \text{s/rad}$$

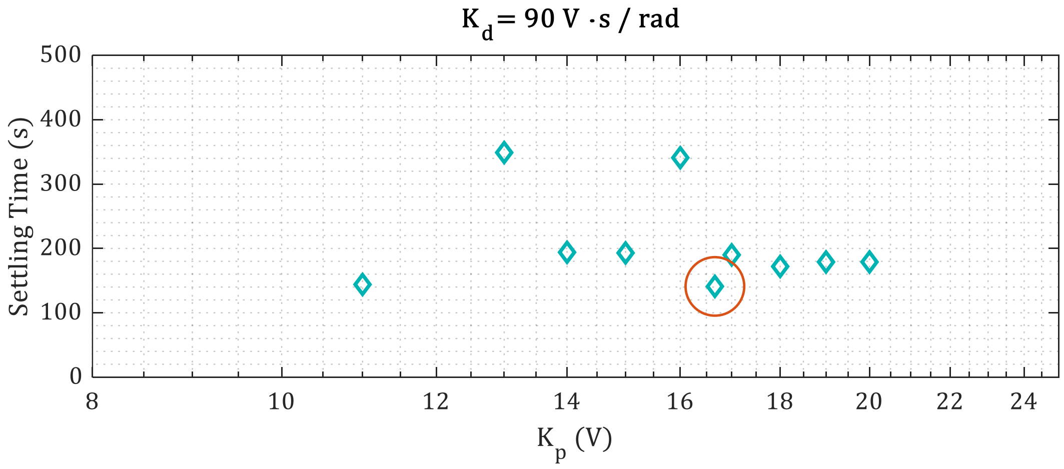


Figure 81: PD Gain Analysis with Constant $K_d = 90$ Vs/rad and Different Values of K_p – Settling from Stationary Initial Body Angular Rates and 71° Offset from Nadir

These plots show a wide range of derivative gains for a narrow range of proportional gains with little variation in performance beyond a worsening steady-state error. The majority of settling times fall in the 400 second range corresponding to one tenth of the available time the satellite sees Sun in a typical orbit.

For these best-case gains, from the same stationary starting conditions as above, Figure 82 shows the settling behaviour in terms of quaternion error and rate error. This figure shows a marked improvement in settling time versus the best-case magnetic PD controller with zero disturbance torques shown in Figure 73, allowing for the majority of the time the satellite spends in Sun during a typical orbit to be spent pointing.

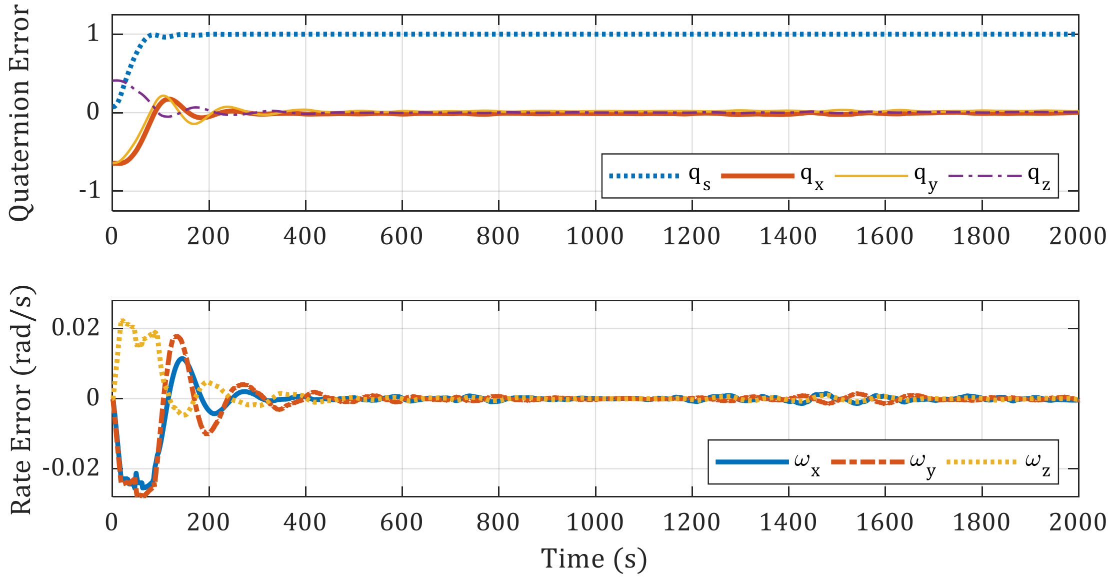


Figure 82: Rate and Quaternion Errors for Best Case Reaction Wheel PD Gains

The proposed PD reaction wheel controller requires satellite attitude as an input which is unavailable for a minimum of 38% of each orbit due to eclipse. A control scheme is proposed in the following sections using the EKF attitude prediction to enable the reaction wheels to obtain a sufficient pointing accuracy while not in eclipse.

7.5 Threshold Solution

Though the EKF was designed to improve upon noisy sensor data and provide a cleaner, more accurate estimate of both attitude and rate errors, its internal model cannot be expected to produce consistently accurate state predictions after extended periods of time without receiving measurement updates. This section proposes a control switch based on this "threshold" of time wherein the EKF can be used to confidently provide accurate satellite attitude and rate estimations.

To determine the appropriate threshold of time (after which the EKF is unable to accurately predict the satellite's attitude in the absence of Sun sensor measurements) Figure 83 plots the BF/ECI quaternion result from the EKF against the Simscape ("actual") quaternion attitude for a stationary satellite initial condition, 71° offset from nadir. Attitude measurements are initially provided to the satellite for 1,500 seconds to allow a consistent pointing orientation to be established (attitude availability = 1). At 1,500 seconds, the q-Method attitude input to the EKF is zeroed (attitude availability = 0), resulting in the EKF output propagating based on its internal model for the remainder of the simulation without any sensor updates. It can be seen from Figure 83 that the EKF predictions begin to deviate from the actual Simscape Quaternion after approximately 100-300 seconds depending on the quaternion element.

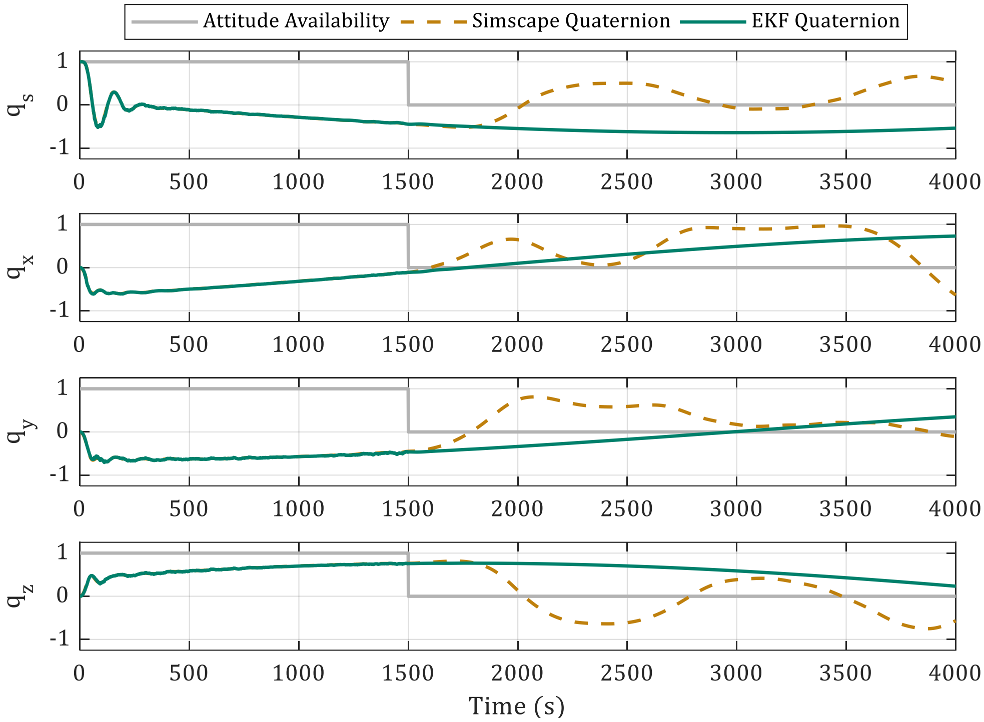


Figure 83: EKF Quaternion State Prediction Versus Simscape Quaternion without No q-Method Attitude Available

The threshold value was selected to be 300s – a value that should suffice in filling gaps in attitude availability. After 300 seconds, the EKF will need to be shut off entirely as it no longer can accurately predict the attitude in the absence of measurements (i.e. all states are set to zero except the quaternion scalar set to 1) and the satellite tumbles freely until the attitude becomes available again.

To test this threshold value, a series of "gaps" in attitude were sent in lieu of the standard eclipse signal to the control system. To better illustrate the effects of switching the EKF on and off, it was assumed that the satellite could always see Sun except during intentional gap periods. Five gaps comprise the series for gradually increasing durations of 100 seconds, 300 seconds, 500 seconds, 1,000 seconds, and 2,781 seconds (half of an orbit), with 1,000 seconds between the gaps to allow the satellite to regain attitude as needed.

Figure 84 plots the elements of the EKF quaternion against the elements of the Simscape (actual) quaternion for this gap signal with the satellite starting from a stationary initial condition, 71° offset from Nadir. As expected, the 100 and 300 second gaps are successfully predicted by the EKF without issue. As the satellite exits the three longer gaps, the satellite quickly regains attitude with no regions of destabilisation.

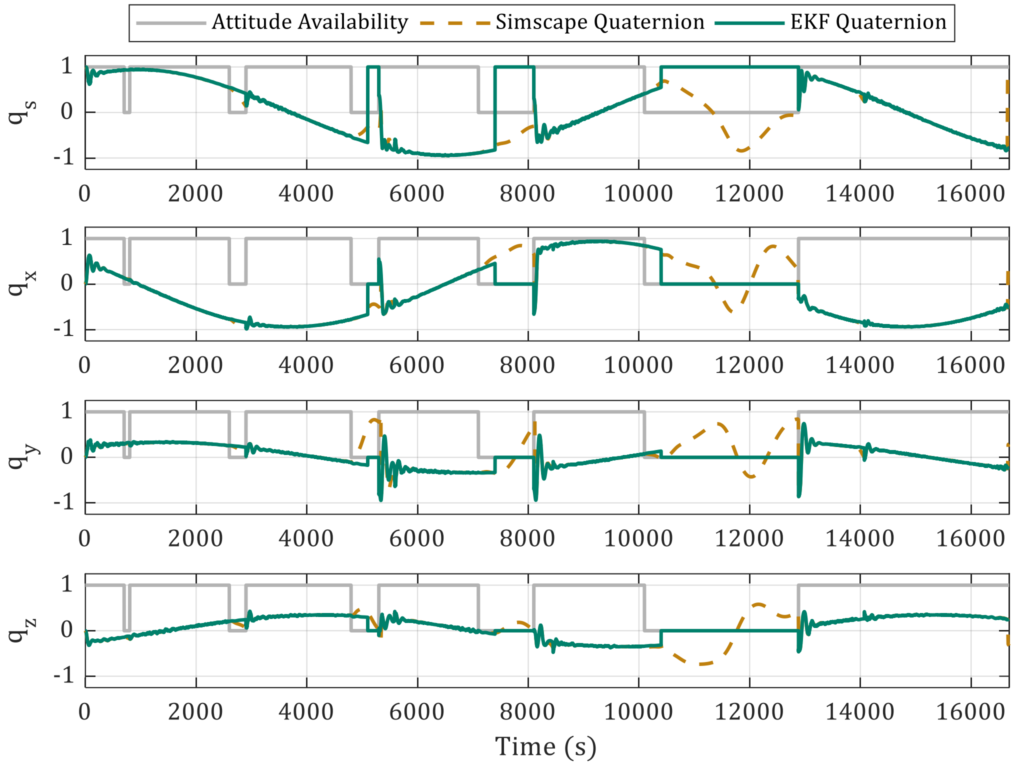


Figure 84: EKF Quaternion State Prediction Versus Simscape Quaternion with Gap Attitude Availability Signal

Another issue with control performance was observed in relation to the reaction wheel speeds. Figure 85 plots sample simulation results for reaction wheel speed and Euler angle attitude with no eclipse, disturbance torques and sensor noise present, and perfect attitude knowledge where the reaction wheels were allowed to run for the entire duration of 4.5 orbits. It can be seen in the top subplot of Figure 85 that, over time, the reaction wheel speed and attitude errors gradually increase. This gradual increase in attitude error with increase in reaction wheel speed is likely because a spinning flywheel induces a gyroscopic stiffness effect which creates a degree of rigidity or resistance to disturbance torques – similar to the way a spinning top appears to defy gravity as it resists gravitation torques. The faster the rotational speed, the stronger the gyroscopic stiffness.

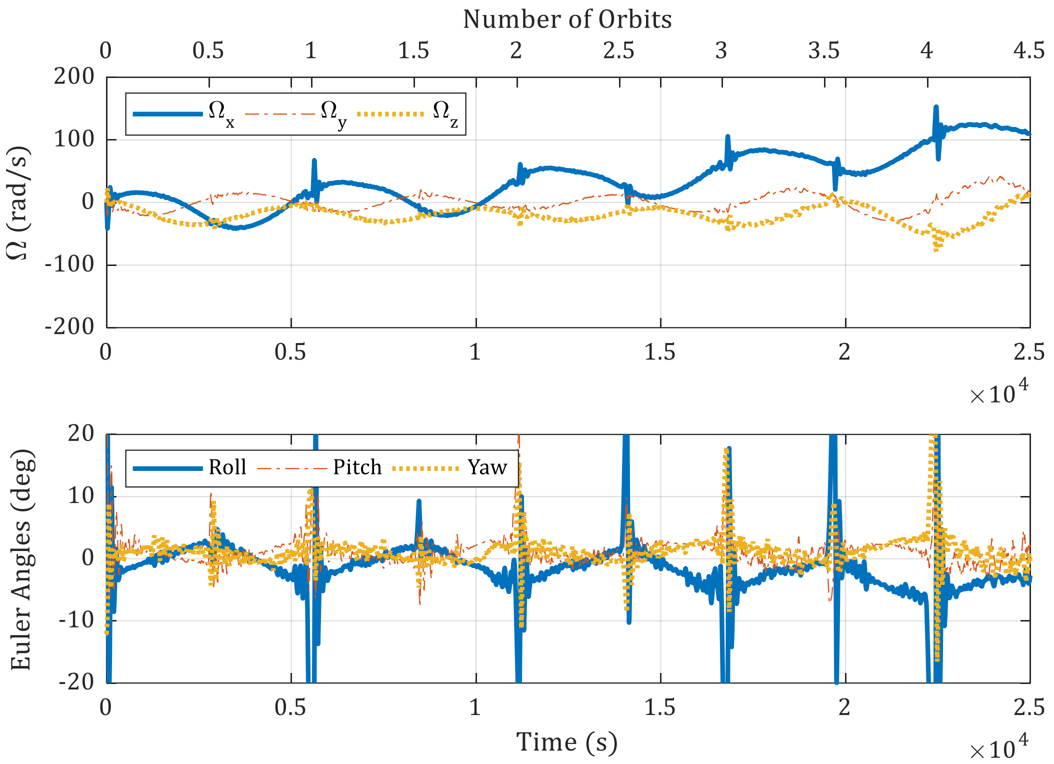


Figure 85: Increasing Reaction Wheel Speed and Corresponding Worsening of Attitude Error

It was observed that PD controller performance was reduced as the reaction wheel speed increased and, therefore, a solution was proposed to maintain the wheel speeds below a certain value by detumbling the satellite during eclipse using magnetic B-dot control while zeroing the reaction wheel speeds and only reactivating the reaction wheels when the magnitude of the satellite's body angular rate is less than 0.04 rad/s (approximately 0.02 rad/s about each axis).

To implement this proposed solution, logic control switches were incorporated into the simulator for three conditions. The first: if an attitude estimate is unavailable for less than 300 seconds, replace the EKF output with the EKF prediction (without the update step) and continue using reaction wheels. The second: if an attitude estimate is unavailable for more than 300 seconds, turn off the EKF, zero the reaction wheel control and switch to detumbling the satellite. The third: if the body angular rate magnitude exceeds 0.04 rad/s at any point in the simulation, switch to detumbling the satellite. To illustrate the proposed control switches, Figure 86 and Figure 87 plot the EKF quaternion output versus the actual quaternion taken from Simscape and the associated body angular rates, respectively, for the same gap signal used for Figure 84. The satellite starts from an initial angular rate of 0.02 rad/s about every axis. The 100 and 300 second gaps are again filled without issue. The discontinuity after the 500 second gap occurs due to a sign change in the attitude quaternion between where the prediction ends and where attitude is regained. The final value of the scalar part of the predicted quaternion approaches -1 before it is overridden with +1 due to the 300 second threshold being exceeded. The light tumble resulting from the disturbance torque/no pointing controller combination during the gap coupled with the sudden command to the wheels to orient the satellite back towards the actual attitude causes the satellite's angular rates at that instant to exceed the rate threshold. The detumbling controller is switched on until the angular rates are reduced to within the angular rate threshold (around the 9000 second mark in Figure 86). The EKF regains attitude after the half orbit gap without issue, likely due to the less drastic difference in sign between the overridden attitude and the actual attitude.

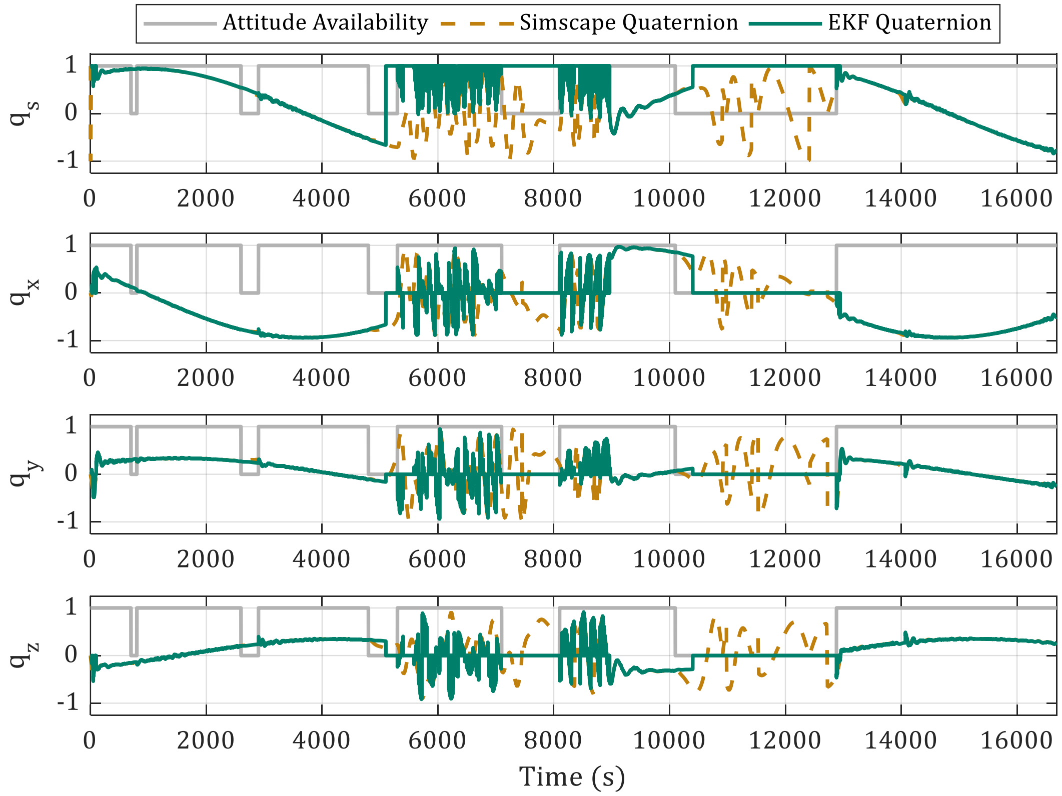


Figure 86: EKF Attitude Quaternion versus Simscape Output with Control Switching

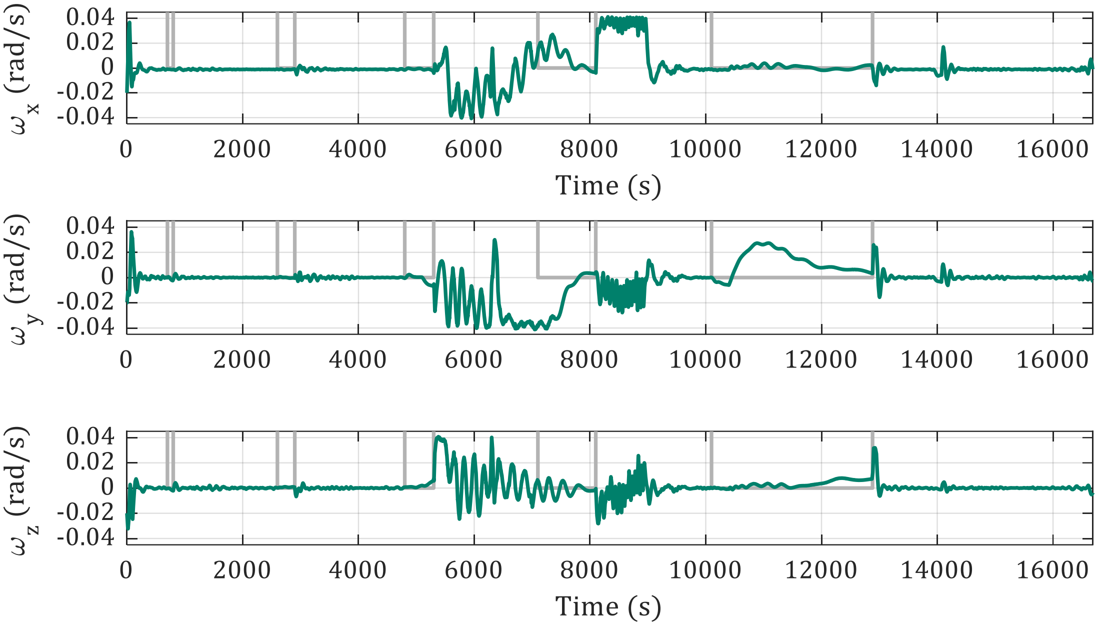


Figure 87: Body Angular Rates with Control Switching

7.6 Wheel Performance Results

Noting the degradation in PD controller pointing accuracy observed in the previous section, it becomes necessary to assess wheel speed and controller performance over a longer time frame. With the angular rate threshold solution proposed, each reaction wheel is given the chance to desaturate and unload its momentum during the eclipse period of every orbit. If the wheel speeds are maintained within a consistently low range, excess momentum should not increase over time. Figure 88 plots the reaction wheel speeds over 100 orbits. A linearly increasing trend is not present in the speeds about any axis, and the profile of the switch between reaction wheel and B-dot control follows the same general profile orbit to orbit. Though speeds increase sharply at orbit 89 – an orbit where the Sun vector is unavailable for a greater percentage of its duration than usual – the speeds decrease back to the typical range in under 5 orbits as desired.

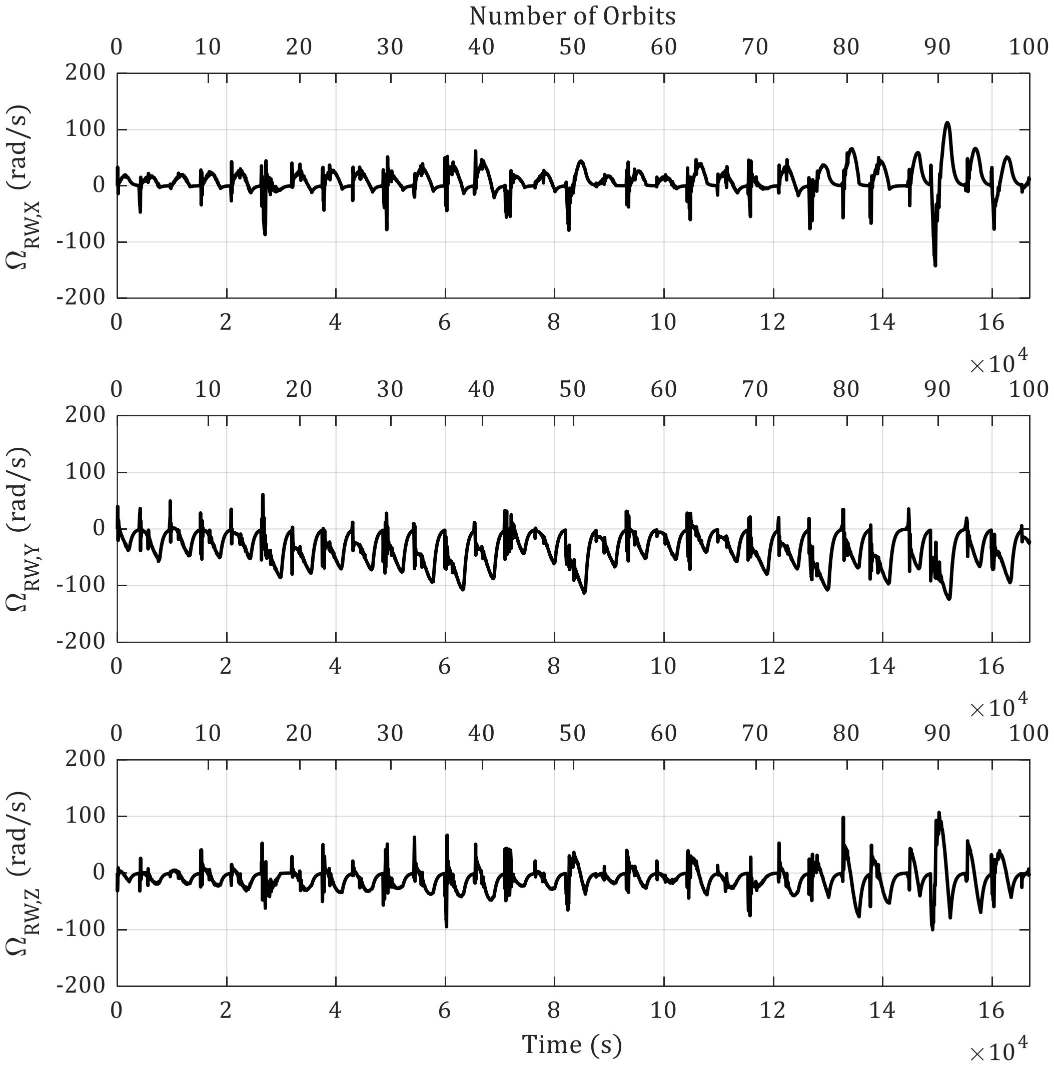


Figure 88: Reaction Wheel Speeds over 100 Orbits

Similarly, reaction wheel torques are plotted for the 100-orbit duration in Figure 89. The magnitude of the spikes corresponds to T_{max} defined for the DC motor model in Section 7.2, while the nominal torques rarely exceed 5×10^{-6} Nm. Again, a consistent pattern is followed orbit-to-orbit with no evidence of momentum buildup.

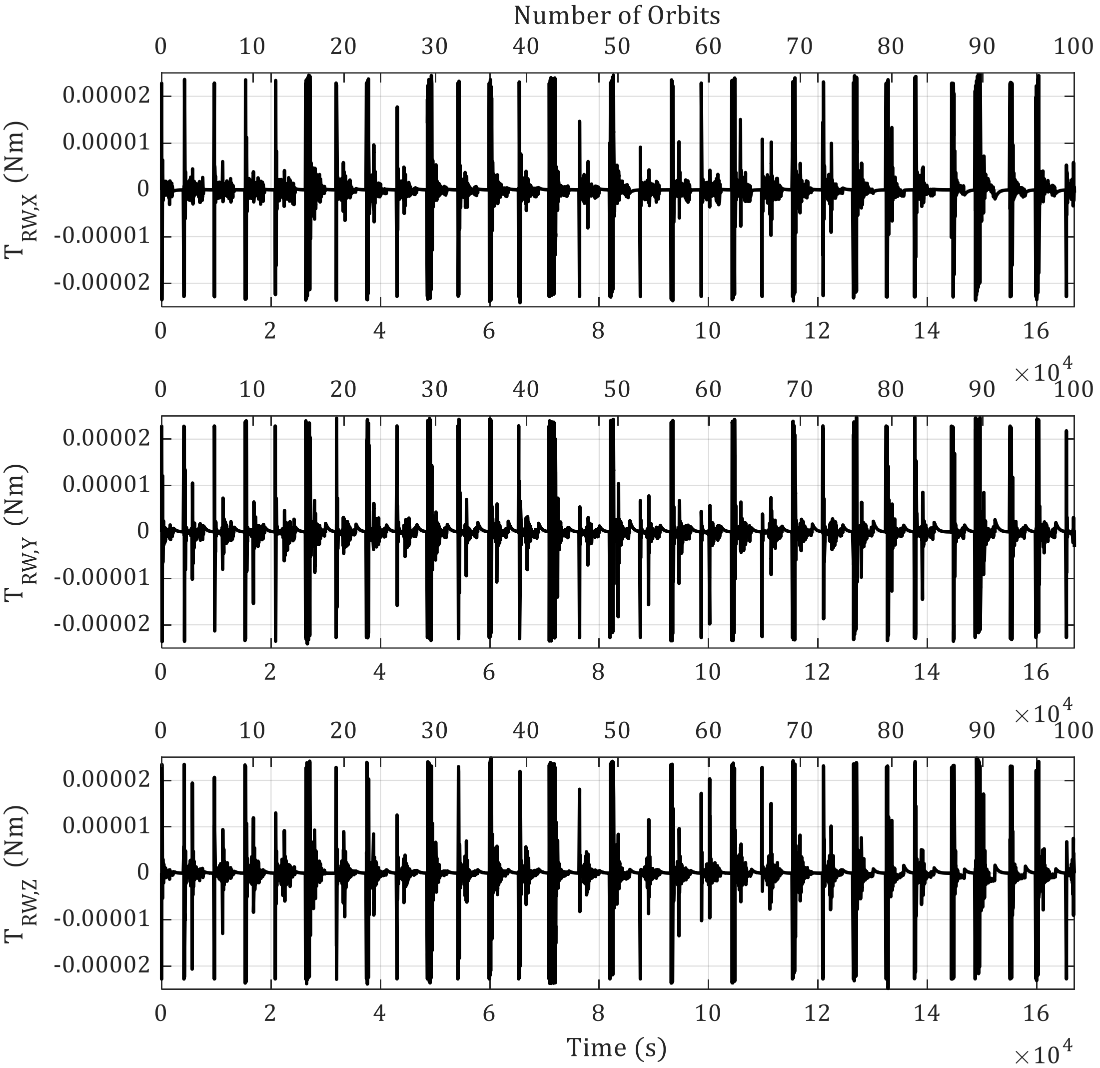


Figure 89: Reaction Wheel Torques over 100 Orbits

With the reaction wheel system operating as intended and successfully desaturating over the set interval, the control approach can be finalised. The following chapter summarises when and where each controller is used throughout the mission life and presents the final PD control performance results in the context of the LORIS mission requirements.

Chapter 8: Control Approach

LORIS operates in two configurations. Post-launch its four deployable solar panels are closed until initial detumbling is complete. Once LORIS is detumbled, its solar panels and antennae are deployed and remain open for the remainder of its mission life.

The control approach while the solar panels are deployed alternate between detumbling and pointing, as expounded in previous section. The progression through the control scheme appears in Figure 90, and all control switching is handled in an aptly named "control switch" MATLAB function block containing the logic illustrated by the flowchart.

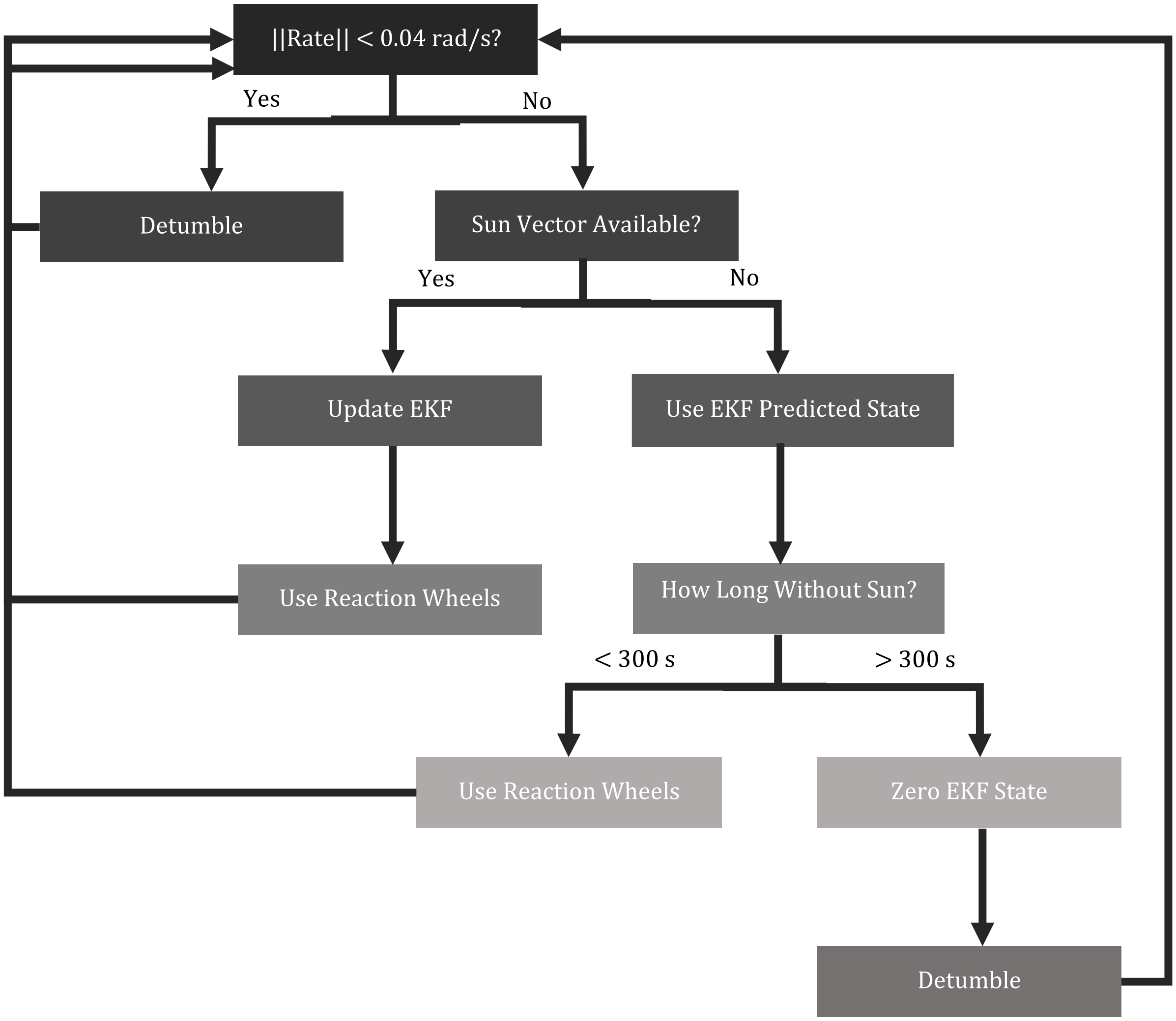


Figure 90: Control Switch Logic Flowchart

Referring to Figure 90, if the body angular rates exceed a magnitude of 0.04 rad/s at any time during the simulation, the reaction wheels are shut off and the satellite is detumbled until the rates fall below this threshold. Once they do, the availability of the Sun vector is checked. If the Sun vector is available, then the satellite attitude can be estimated by the q-Method algorithm and the EKF can enter and remain in its update phase. The updated EKF state estimates are carried through to the reaction wheel PD controller, and the wheels maintain the satellite in its desired pointing attitude. Once the Sun vector becomes unavailable, the EKF output switches to its prediction phase and continues to carry its predicted state output through to the reaction wheels until the “gap” duration threshold of 300 seconds has been reached. Once 300 seconds have passed without a Sun vector, the EKF outputs and reaction wheel inputs are zeroed and the detumbling controller is switched on, whereupon the journey through the logic tree resets. In the simulator, the control switch is set to “1” when the reaction wheels are on and “0” when the detumbling controller is on.

As shown in Section 7.6, the reaction wheels desaturate frequently enough to not require an official “wheel desaturation” mode of operation. If the ADCS is turned back on after a period where the satellite has been switched to safe mode or due to some fault, it will automatically detumble until the residual rates from this period are reduced back into the control range, then automatically regain its pointing attitude when Sun is in view. The control scheme detumbles the satellite for all but 300 seconds of every eclipse period, leaving the majority of the period the satellite is in Sun available for pointing. This solution also fits with the normal operation of the payload cameras, as daytime photographs are preferred.

8.1 Attitude Determination Results

The ability of the satellite’s ADCS to maintain pointing attitude throughout its mission life hinges upon its ability to obtain an accurate estimate of the satellite’s attitude. As such, this section focuses on assessing the accuracy of the attitude determination system – the ability of the EKF to predict the satellite’s attitude accurately and consistently. Results in this section are obtained from regions where the reaction wheels are actively maintaining pointing attitude, thus excluding eclipse periods and the settling periods thereafter. The satellite is given a conservative bound of 800 seconds post-eclipse before attitude error calculations begin.

Figure 91 compares the q-Method, EKF and “actual” Simscape quaternions over a settled 1900 second portion of one orbit to show the difference in the three quaternion profiles. The q-Method estimation is notably the noisiest, though both the q-Method and EKF follow the general profile of the actual attitude within a reasonable degree of accuracy.

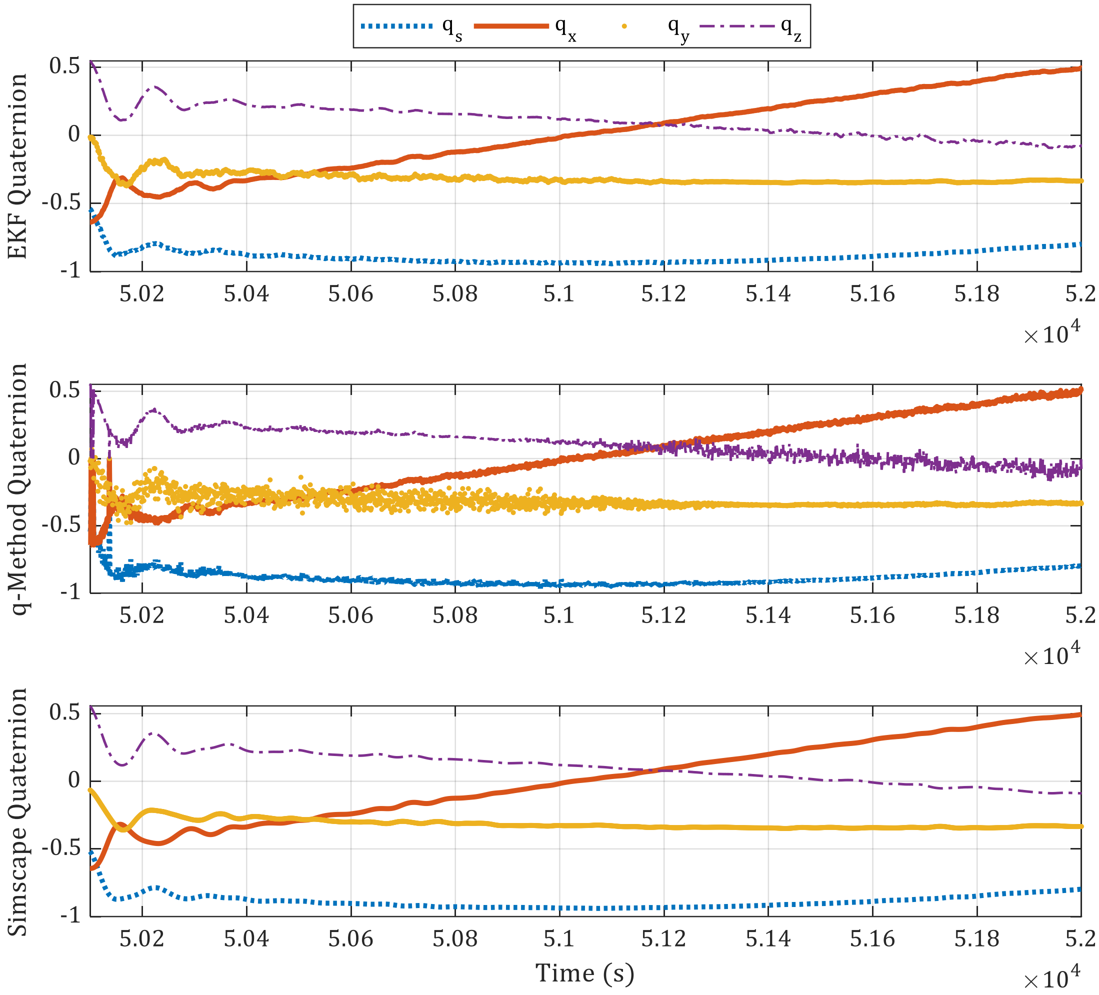


Figure 91: EKF, q-Method, and "Actual" Quaternion Profile Comparison

Attitude determination error is defined as q_{err}, the quaternion representing the quaternion transformation between the estimated and "actual" quaternions, represented by the relation below. Two pairs of quaternions are considered: EKF and Simscape and q-Method and Simscape.

$$q_{err} = q_{estimated}^{-1} q_{Simscape} \tag{97}$$

This transformation quaternion is converted into axis-angle representation via the MATLAB function quat2axang, and the attitude determination error is taken as its angle component.

The following analyses consider the average attitude determination accuracy per orbit. Outliers with errors greater than 60 degrees – the result of occasional instantaneous Simscape quaternion inversion and gap periods of Sun unavailability that were not caught in the error calculation logic - are excluded from average calculations. Based on the data, for regions where the EKF filled gaps, attitude determination error reached a maximum of approximately 25°. Figure 92 plots the average q-Method-Simscape error per orbit for 100 orbits.

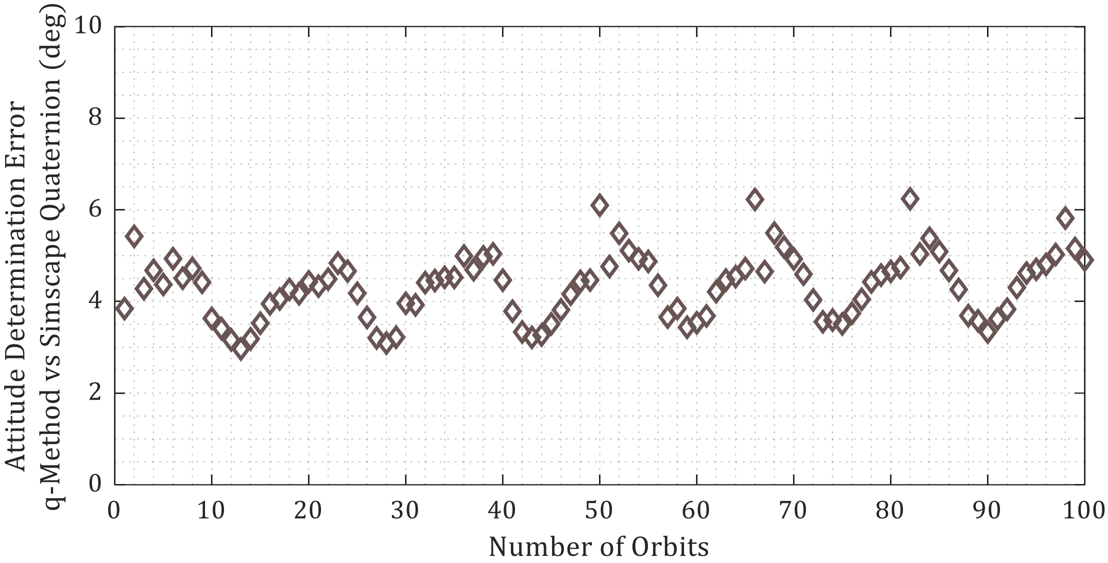


Figure 92: Average Attitude Determination Error per Orbit between q-Method Quaternion Estimate and “Actual” Simscape Quaternion – 100 Orbit Duration

This data has an average and standard deviation of:

$$4.33 \pm 0.72°$$

Despite the noisier estimation, the q-Method is still capable of obtaining an accurate estimate of the satellite’s attitude that fits well within the LORIS requirement of ±10°. Figure 93 plots the average attitude determination error per orbit over 100 orbits for the EKF-Simscape pair.

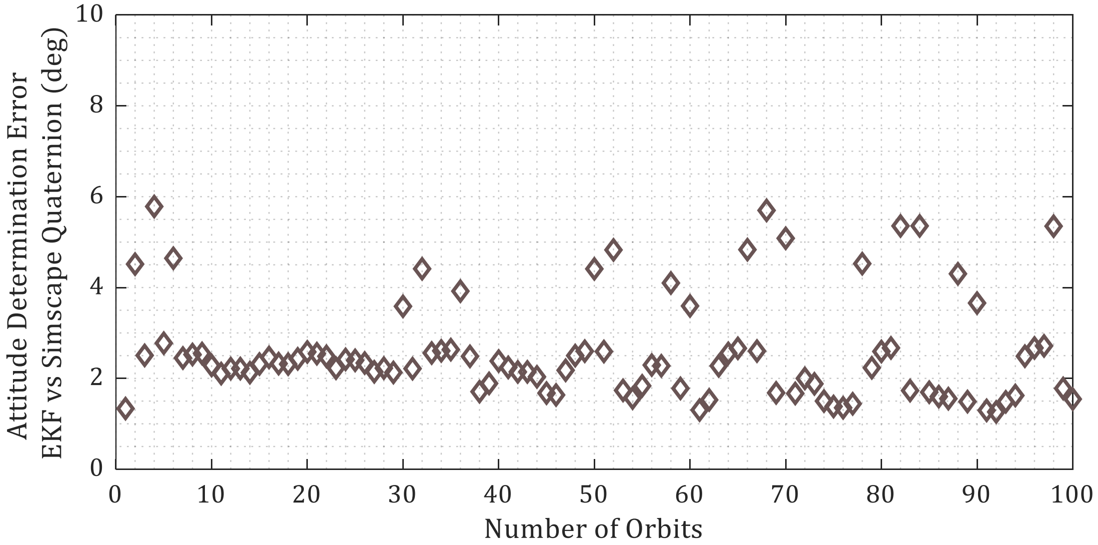


Figure 93: Average Attitude Determination Error per Orbit between EKF Quaternion Estimate and “Actual” Simscape Quaternion – 100 Orbit Duration

The attitude determination error with the EKF quaternion estimate in Figure 93 is generally lower than the attitude determination error with the q-Method in Figure 92. Outliers in Figure 93 typically occur for periods where the satellite is detumbled. The satellite experiences higher initial angular rates pre-settling and the averages may include some of this pre-settled data. The final attitude determination accuracy for the ADCS is taken from the mean data in Figure 93 as:

$$2.6 \pm 1.1°$$

which lies well within the attitude determination requirement of $\pm 10°$.

8.2 PD Control Performance Results

With the hard limit on CubeSat angular rates determining when the reaction wheels may operate and in reference to the control switch logic in Figure 90, the satellite's angular rates may still be above the 0.04 rad/s body angular rate threshold when it comes out of eclipse. The satellite will then need to continue detumbling until the angular rates drop below the threshold before nadir pointing can resume. As a result, even when not in eclipse, some time may be lost per orbit where the satellite will not be able to point at its target. Under the simplifying assumption that the satellite is "pointing" and "viewing" its target any time the wheels are active, Figure 94 plots viewing time as a percentage of each orbit, while Figure 95 plots the percent loss in viewing time per orbit, calculated as the difference between the maximum viewing time and the percent viewing time from Figure 94. The reaction wheels can point the satellite for a maximum of about 62% of an orbit, the period when the satellite is in Sun. This maximum is marked in Figure 94 with a dotted line.

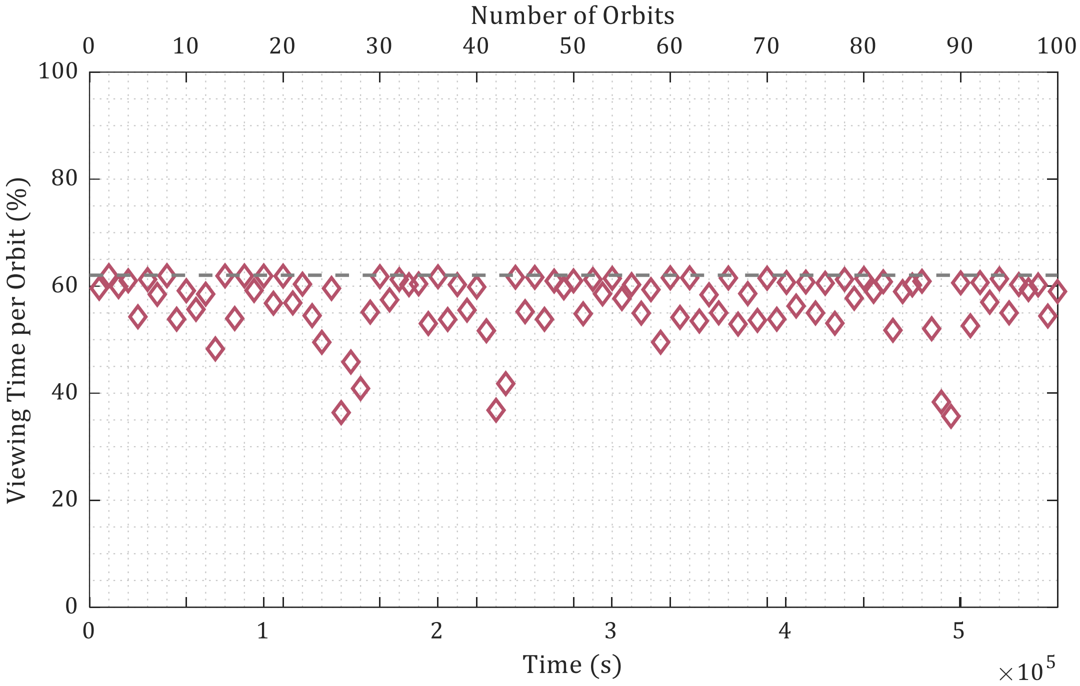


Figure 94: Viewing Time Per Orbit – 100 Orbit Duration

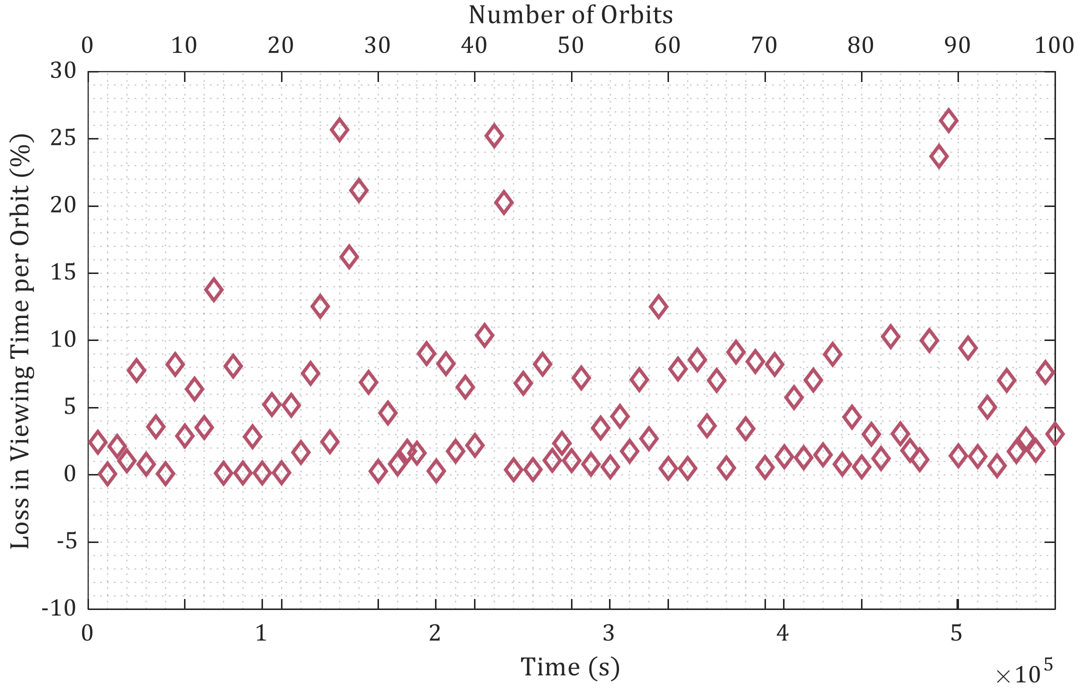


Figure 95: Percent Loss in Viewing Time Per Orbit – 100 Orbit Duration

From the data in Figure 94, the average viewing time per orbit over 100 orbits is 57%, with a maximum of 62% reached during orbit 2, a minimum of 36% reached during orbit 89, and a standard deviation of about 6%. Only 10 orbits had less than 50% of their duration available for viewing.

Pointing error is assessed over the same 100 orbits in terms of the Euler angles [roll pitch yaw] in Figure 96. The satellites are considered to be pointing after 800 seconds have elapsed post-eclipse, based on the range of settling times observed during PD gain tuning analyses. Figure 96 plots the average errors in roll, pitch, and yaw for each of the 100 orbits. Three orbits fall outside the range of the plot: orbit 27 with an average error [40° 16° 33°], orbit 43 with an average error of [47° 21° 41°], Orbit 89 with an average error of [67° 26° 58°]. The orbits with the worst pointing performance correspond to those with the greatest reduction in viewing time, which themselves correspond to orbits where the Sun vector is unavailable for the greatest period of time or that exceed the control switching body angular rate threshold – conditions that often occur in tandem. Statistics for the three orbits with the highest pointing errors noted in the preface for Figure 96 are compiled into Table 13 to illustrate this correlation.

Table 13: Statistics for the Three Worst-Performing Orbits

Orbit Number	Viewing Time	Roll	Pitch	Yaw	Period Sun Unavailable
89	36%	67°	26°	58°	21%
27	46%	40°	16°	33°	16%
43	42%	47°	21°	41°	26%

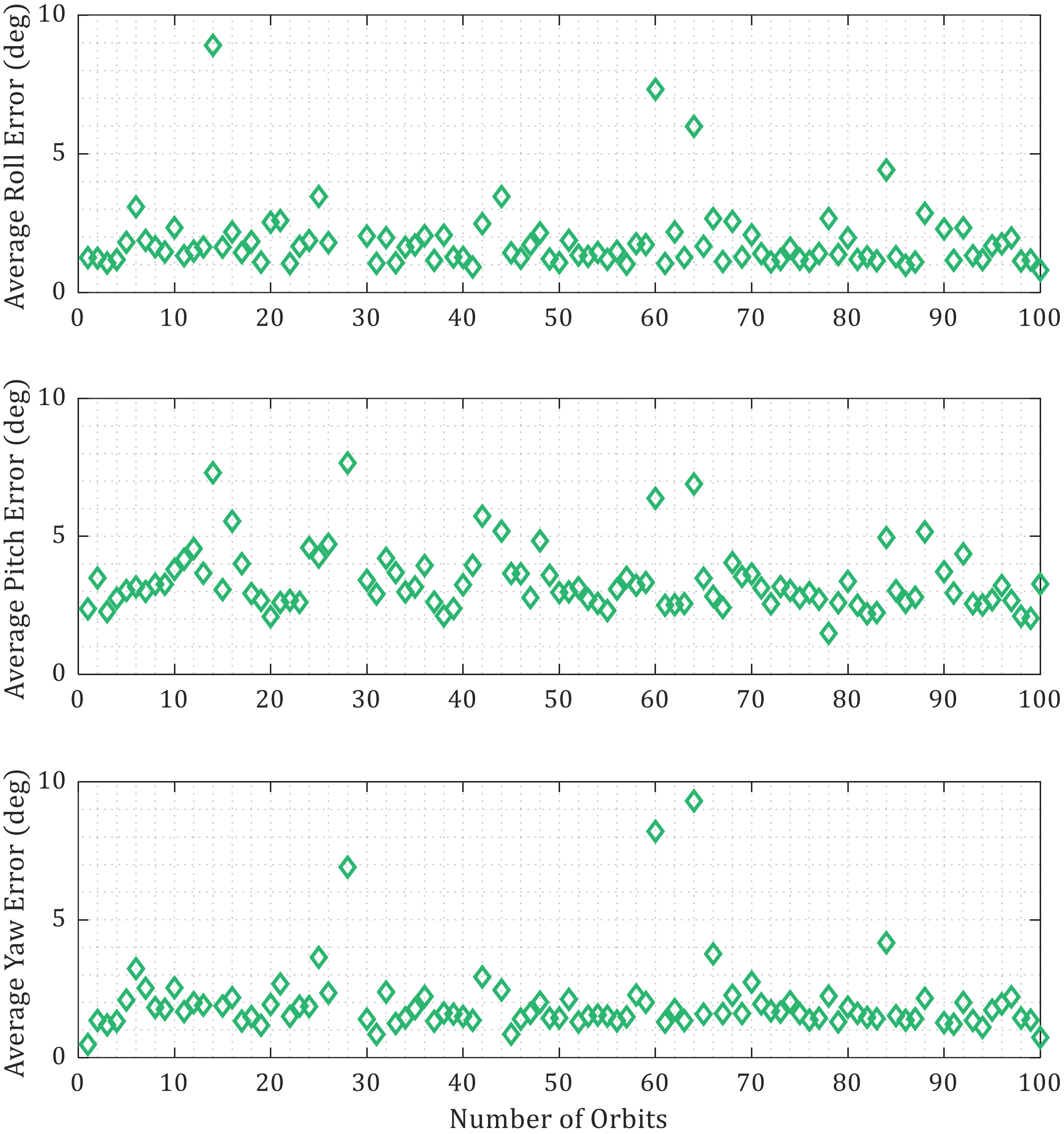


Figure 96: Average Euler Angle Pointing Error Per Orbit for 100 Orbits with Standard Deviations Noted

Standard deviations of the pointing errors are calculated excluding any outlier orbits with average pointing errors greater than 15 degrees about any axis. This condition eliminates only 6% of the simulated orbits. Standard deviations for each Euler angle (where R = roll, P = pitch, and Y = yaw) are indicated as follows:

$$\sigma_R = 1.5^\circ$$

$$\sigma_P = 1.1^\circ$$

$$\sigma_Y = 1.8^\circ$$

The average errors over 100 orbits are calculated excluding the same 6 outliers as:

$$R = 1.9^\circ$$

$$P = 3.4^\circ$$

$$Y = 2.1^\circ$$

The absolute pointing error can be determined by the angle component of the axis-angle representation of attitude by converting the set of Euler angles to a quaternion and converting that quaternion to an axis-angle rotation via the associated MATLAB function quat2axang. The final absolute pointing error was obtained from the average errors per orbit excluding outliers as:

$$4.4 \pm 2.7^\circ$$

This absolute pointing error satisfies the LORIS mission requirement of an attitude pointing accuracy of at least $\pm 10^\circ$, achieving nearly double the required degree of accuracy on average.

Chapter 9: Conclusions and Recommendations

This thesis details the development of a robust simulation tool for the LORIS CubeSat ADCS. A simplified orbit propagator that propagates the ISS orbit ad infinitum based on a set of TLE elements in an environment containing accurate models of Earth's geomagnetic field, Sun and eclipse, and prominent LEO worst-case disturbance torques was designed. Realistic sensor noise and magnetorquer behaviour based on experimentally-obtained data was modelled, and a quaternion-based attitude determination scheme based on the q-Method algorithm and an extended Kalman filter capable of producing an accurate estimate of satellite attitude was designed and validated. For the simulation conditions used in this research, the attitude determination scheme was found to satisfy and perform almost five times better than the $\pm 10°$ LORIS mission requirement at an average accuracy of $2.6 \pm 1.1°$. A B-dot detumbling controller that operates solely based on filtered differentiated geomagnetic field readings was designed and validated and was found to detumble well within the required range of 15 hours (10 orbits) for a variety of initial angular rate conditions. The design and validation of a simple PD pointing control scheme based around three orthogonal reaction wheels capable of attaining the required pointing accuracy of $10°$ with an average pointing error was completed, with a final pointing accuracy of $4.4 \pm 2.7°$ obtained for the simulation conditions used in this research.

The contributions of this research are:

1. the development of a simulation tool for the LORIS CubeSat ADCS
2. the use of the simulation tool to validate that the proposed sensor suite, actuators, and control algorithms for LORIS meets its mission requirements
3. the completion of gain sensitivity analyses for B-Dot and magnetic PD control, and the assessment of the impact of different levels of sensor noise on the performance of each controller. The results of these analyses were published in [119]
4. the development of an EKF-based correction scheme for the "duality" problem inherent to q-Method attitude determination that performs well even in the presence of sensor noise.
5. the proposed control scheme for cycling between pointing the CubeSat during portions of the orbit where Sun is available to the satellite's Sun sensors for attitude determination, and simultaneous detumbling of the satellite and desaturation of its reaction wheels during eclipse.

Future work includes the implementation of a high-fidelity orbit propagator that incorporates accurate orbital perturbations, such as the standard Simplified General Perturbations (SGP4) model. This propagator is best suited to long running simulations where the effects of drag will have a significant effect on orbit shape and altitude, thereby changing key characteristics of the satellite's environment (namely magnetic field strength and eclipse period). It follows that quotidian disturbance torques (rather than worst-case) be implemented to assess how the satellite fares during typical operation. The addition of an orbit determination algorithm further based on periodic GPS measurements would remove the current simulator assumption of perfect orbit knowledge and allow the investigation of the effect of satellite positioning errors on the proposed control algorithms. Further improving the realism of the simulator would also require the implementation of an EKF with improved reaction wheel dynamics and no simplifying reaction wheel location assumptions, preferably such that the location of the wheels on LORIS can be accurately represented in the dynamic model. As well, reaction wheel and satellite geometry and properties should be continually updated as the project design continues to change and approach its final version.

References

[1] Dalhousie Space Systems Lab Chassis Subsystem, *CAD Assembly V3.7,* 2020.

[2] T. Villela, C. A. Costa, A. M. Brandão, F. T. Bueno and R. Leonardi, "Towards the Thousandth CubeSat: A Statistical Overview," *International Journal of Aerospace Engineering,* vol. 2019, pp. 1-13, 2019.

[3] E. Kulu, "Nanosatellite & Cubesat Database," 10 June 2019. [Online]. Available: https://www.nanosats.eu/database. [Accessed 24 October 2019].

[4] G. D. de Oliveira, J. Y. Ishihara, R. A. Borges, H. C. Ferreira, A. M. Kulabukhov, V. A. Larin and V. V. Belikov, "A Low-Cost Attitude Determination and Control System for the UYS-1 nanosatellite," in *2013 IEEE Aerospace Conference*, Big Sky, MT, 2013.

[5] J. Cubas, A. Farrahi and S. Pindado, "Magnetic Attitude Control for Satellites in Polar or Sun-Synchronous Orbits," *Journal of Guidance, Control, and Dynamics,* vol. 38, no. 10, pp. 1-12, October 2015.

[6] F. Celani, "Spacecraft Attitude Stabilization using Magnetorquers with Separation between Measurement and Actuation," in *AIAA Guidance, Navigation, and Control Conference*, San Diego, CA, 2016.

[7] C. J. Damaren and J. R. Forbes, "Linear Time-Varying Passivity-Based Attitude Control Employing Magnetic and Mechanical Actuation," *Journal of Guidance, Control, and Dynamics,* vol. 34, no. 5, pp. 1363-1372, 2011.

[8] M. Lovera, "Optimal Magnetic Momentum Control for Inertially Pointing Spacecraft," *European Journal of Control,* vol. 7, no. 1, pp. 30-39, 2001.

[9] K. Roh, S. Park, K. Choi and S. Lee, "Minimum-Time Attitude Reorientations of Three-Axis Stabilized Spacecraft Using Only Magnetic Torquers," *International Journal of Aeronautical and Space Sciences,* vol. 8, no. 2, pp. 17-27, November 2007.

[10] A. Ali, M. R. Mughal, H. Ali, L. M. Reyneri and M. N. Aman, "Design, implementation, and thermal modeling of embedded reconfigurable magnetorquer system for nanosatellites," *IEEE Transactions on Aerospace and Electronic Systems ,* vol. 51, no. 4, pp. 2669-2679, October 2015.

[11] T. Graversen, M. K. Frederiksen, S. V. Vedstesen, K. Krogh and E. Schreder, "AAU CubeSat: Aalborg University's Studentsatellite," Aalborg University, 6 June 2002. [Online]. Available: http://www.space.aau.dk/cubesat/. [Accessed 24 October 2019].

[12] K. F. Jensen and K. Vinther, “Attitude Determination and Control System for AAUSAT3,” M. S. Thesis, Department of Electronic Systems, Aalborg University, Aalborg, Denmark, 2009.

[13] B. G. Thomsen and J. Nielsen, "CubeSat Sliding Mode Attitude Control," Aalborg University, Aalborg, Denmark, 2016.

[14] J. Gangestad, D. Rowen, B. Hardy, C. Coffman and P. O'Brien, "Flight Results from AeroCube-6: A Radiation Dosimeter Mission in the 0.5U Form Factor," The Aerospace Corporation, San Luis Obispo, CA, 2015.

[15] A. D. Schutte, D. P. Arndt, P. T. Doyle, D. W. Rowen and R. M. Dolphus, "Generalized Magnetic Attitude Control of Spacecraft with Application to a 0.5U CubeSat Platform," in *65th International Astronautical Congress*, Toronto, Canada, 2014.

[16] M. Tassano, P. Monzon and J. Pechiar, "Attitude Determination and Control System of the Uruguayan CubeSat, AntelSat," in *2013 16th International Conference on Advanced Robotics*, Montevideo, Uruguay, 2013.

[17] M. Tassano Ferrés, "Design and Implementation of an Attitude Determination and Control System for the AntelSat," Universidad de la República, Montevideo, Uruguay, 2015.

[18] "Chasqui I," Wikipedia: The Free Encyclopedia, 14 November 2017. [Online]. Available: https://en.wikipedia.org/wiki/Chasqui_I#System_Identification_and_Attitude_Control_%E2%80%93_SDCA. [Accessed 22 October 2019].

[19] R. Miyagusuku, K. R. Arias and E. R. Villota, "Hybrid Magnetic Attitude Control System Under CubeSat Standards," in *2012 IEEE Aerospace Conference*, Big Sky, MT, 2012.

[20] K. P. Vega, "Attitude Control System for CubeSat for Ions, Neutrals, Electrons and MAGnetic Field (CINEMA)," University of California, Berkeley, Berkeley, CA, 2009.

[21] J. Gießelmann, "Development of an Active Magnetic Attitude Determination and Control System for Picosatellites on highly inclined circular Low Earth Orbits," RMIT University, Melbourne, Australia, 2006.

[22] "CSTB1 (CubeSat TestBed 1)," eoPortal, [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/c-missions/cstb1. [Accessed 30 October 2019].

[23] K. Z. Brown, T. G. Rose, B. K. Malphrus, J. A. Kruth and E. T. Thomas, "The Cosmic X-Ray Background NanoSat (CXBN): Measuring the Cosmic X-Ray Background using the CubeSat Form Factor," in *26th Annual AIAA/USU Conference on Small Satellites*, Logan, UT, 2012.

[24] P. Testani, "Development of Sun-Pointing Magnetic Attitude Control System for CXBN CubeSat," in *63rd International Astronautical Congress*, Naples, Italy, 2012.

[25] C. Fish, C. Swenson, T. Neilsen, B. Bingham, J. Gunther, E. Stromberg, S. Burr, R. Burt and M. Whitely, "DICE Mission Design, Development, and Implementation: Success and Challenges," in *26th Annual AIAA/USU Conference on Small Satellites*, Logan, UT, 2012.

[26] Planet Labs, Inc., "Planet Labs' Remote Sensing Satellite System," in *CubeSat Developers Workshop*, Logan, UT, 2013.

[27] A. Slavinskis, U. Kvell, E. Kulu, I. Sünter, H. Kuuste, S. Lätt, K. Voormansik and M. Noorma, "High spin rate magnetic controller for nanosatellites," *Acta Astronautica,* vol. 95, pp. 218-226, March 2014.

[28] A. Slavinskis, M. Pajusalu, H. Kuuste, H. Ehrpais, K. Zalite, R. Rosta and T. Kalvas, "ESTCube-1 In-Orbit Experience and Lessons Learned," *IEEE A&E Systems Magazine,* pp. 12-22, 2015.

[29] "GOMX-1 (GomeSpace Express-1)," eoPortal, [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/g/gomx-1. [Accessed 24 October 2019].

[30] "HIT-Sat (Hokkaido Institute of Technology Satellite)," eoPortal, [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/h/hit-sat. [Accessed 2 November 2019].

[31] "Masat-1," Budapest University of Technology and Economics CubeSat, 2012. [Online]. Available: https://www.masat.space/en/projektek/masat-1/. [Accessed 2 November 2019].

[32] M. Langer, F. Schummer, N. Appel, T. Gruebler, K. Janzer, J. Kiesbye and L. Krempel, "MOVE-II Munich Orbital Verification Experiment II," in *4th IAA Conference on University Satellite Missions and CubeSat Workshop*, Rome, Italy, 2017.

[33] Small Spacecraft Systems Research Center, "OPUSAT," Osaka Prefecture University, 4 May 2012. [Online]. Available: https://web.archive.org/web/20120513223707/http://www.sssrc.aero.osakafu-u.ac.jp/projects/OPUSAT/shisei.html. [Accessed 1 November 2019].

[34] B. Bruninga, "PSAT - APRS and a new PSK31 Approach," US Naval Academy Satellite Lab, 18 January 2014. [Online]. Available: http://aprs.org/psat.html. [Accessed 22 October 2019].

[35] E. Shakhmatov, I. Belokonov, I. Timbai, E. Ustiugov, A. Nikitin and S. Shafran, "SSAU Project of the nanosatellite SamSat-QB50 for monitoring the Earth's thermosphere parameters," *Procedia Engineering,* vol. 104, pp. 139-146, 2015.

[36] R. Sutherland, I. Kolmanovsky and A. R. Girard, "Attitude Control of a 2U Cubesat by Magnetic and Air Drag Torques," *IEEE Transactions on Control Systems Technology,* vol. 27, no. 3, pp. 1047-1059, 2019.

[37] L. Qaio, M. Zhang, B. P. Osborne and A. Dempster, "Attitude Determination and Control System (ADCS) of UNSW QB50 project UNSW EC0," in *Proceedings of the 13th Australian Space Science Conference*, Sydney, Australia, 2013.

[38] M. Urban, O. Nentvich, V. Stehlikova, T. Baca, V. Daniel and R. Hudec, "VZLUSAT-1: Nanosatellite with miniature lobster eye X-ray telescope and qualification of the radiation shielding composite for space application," *Acta Astronautica,* vol. 140, pp. 96-104, 2017.

[39] Y. Sakamoto, Y. Tanabe, H. Yagisawa, N. Sugimura, K. Yoshida, M. Nishio, T. Nakajo and H. Akiyama, "Operation Results of Cubesat RAIKO Released from International Space Station," *Transactions of the Japan Society for Aeronautical And Space Sciences, Aerospace Technology Japan,* vol. 12, no. 29, pp. 7-12, 2014.

[40] "SOMP (Students' Oxygen Measurement Project)," eoPortal, [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/s/somp. [Accessed 2 November 2019].

[41] C. Angadi, "STUDSAT: India's first student Pico-satellite project," in *2011 Aerospace Conference*, Big Sky, MT, 2011.

[42] M. Noca, F. Jordan, N. Steiner, T. Choueiri, F. George, G. Roethlisberger, N. Scheidegger, H. Peter-Contesse, M. Borgeaud, R. Krpoun and H. Shea, "Lessons Learned from the First Swiss Pico-Satellite: SwissCube," in *23rd Annual AIAA/USU Conference on Small Satellites*, Logan, UT, 2009.

[43] H. A. A. Alarazah, "Theoretical and Numerical Analysis of the Attitude Control of a 3U CubeSat - Iraqi Satellite (TIGRISAT)," *Association of Arab Universities Journal of Engineering Sciences,* vol. 25, no. 4, pp. 236-256, 2018.

[44] "UKube-1 (United Kingdom Universal Bus Experiment 1)," eoPortal, [Online]. Available: https://directory.eoportal.org/web/eoportal/satellite-missions/u/ukube-1. [Accessed 1 November 2019].

[45] J. Kim and K. Worrall, "Sun tracking controller for UKube-1 using magnetic torquer only," in *19th IFAC Symposium on Automatic Control in Aerospace*, Würzburg, Germany, 2013.

[46] "VELOX-P Pico-satellite," Nanyang Technological University, 3 March 2014. [Online]. Available: https://web.archive.org/web/20140312035244/http://www.sarc.eee.ntu.edu.sg/Project/Pages/VELOX-P.aspx. [Accessed 2 November 2019].

[47] S. Goh, J. Chia, S. Chin, K. Low and L. Lim, "Performance Analysis on NORAD TLE Pre-Process Method for Low Cost ADCS MCU," in *30th International Symposium on Space Technology and Science, 34th International Electrical Propulsion Conference & 6th Nano-Satellite Symposium*, Kobe, Japan, 2015.

[48] N. Sugimura, T. Kuwahara and K. Yoshida, "Attitude Determination and Control System for Nadir Pointing Using Magnetorquer and Magnetometer," in *2016 IEEE Aerospace Conference*, Big Sky, MT, 2016.

[49] M. C. Santiago, L. S. Velasco and A. R. Aranda, "Low cost magnetic simulator for attitude control," in *2018 XIII Technologies Applied to Electronics Teaching Conference*, La Laguna, Spain, 2018.

[50] V. Vedant, "Dynamic programming based attitude trajectories for underactuated control systems," in *41st Annual AAS Rocky Mountain Section Guidance and Control Conference*, Breckenridge, CO, 2018.

[51] D. S. Ivanov, M. Y. Ovchinnikov and D. S. Roldugin, "Three-Axis Attitude Determination Using Magnetorquers," *Journal of Guidance, Control, and Dynamics,* vol. 41, no. 11, November 2018.

[52] A. R. Walker, P. T. Putman and K. Cohen, "Solely Magnetic Genetic/Fuzzy-Attitude-Control Algorithm for a CubeSat," *Journal of Spacecraft and Rockets,* vol. 52, no. 6, November 2015.

[53] M. Lovera, "Periodic Attitude Control for Satellites with Magnetic Actuators: An Overview," *IFAC Proceedings Volumes,* vol. 34, no. 12, pp. 113-118, August 2001.

[54] D. S. Ivanov, M. Y. Ovchinnikov, V. I. Penkov, D. S. Roldugin, D. M. Doronin and A. V. Ovchinnikov, "Advanced numerical study of the three-axis magnetic attitude control and determination with uncertainties," *Acta Astronautica,* vol. 132, pp. 103-110, March 2017.

[55] M. Reyhanoglu, C. Ton and S. Drakunov, "Attitude Stabilization of a Nadir-Pointing Small Satellite Using Only Magnetic Actuators," *IFAC Proceedings Volumes,* vol. 42, no. 19, pp. 292-297, 2009.

[56] D. A. Vallado, Fundamentals of Astrodynamics and Applications, 2nd ed., El Segundo, CA: Microcosm Press, 2004.

[57] A. Colagrossi and M. Lavagna, "Fully magnetic attitude control subsystem for picosat platforms," *Advances in Space Research,* vol. 62, no. 12, pp. 3383-3397, 15 December 2018.

[58] K. Kinger, R. Agarwal, C. Nagarajan, B. Shahi, V. Kashyap and N. Gupta, "Using xPC Target to test the control system of a nano satellite," in *2016 IEEE Aerospace Conference*, Big Sky, MT, 2016.

[59] M. Y. Ovchinnikov, D. S. Roldugin and V. I. Penkov, "Asymptotic study of a complete magnetic attitude control cycle providing a single-axis orientation," *Acta Astronautica,* vol. 77, pp. 48-60, September 2012.

[60] M. Lovera and E. Silani, "Predictive Attitude Control Techniques for Satellites With Magnetic Actuators," *IFAC Proceedings Volumes,* vol. 35, no. 1, pp. 199-204, 2002.

[61] J. W. Cheong, B. J. Southwell, J. Bultitude, C. Lam, W. Andrew, S. Green, B. Osborne, A. G. Dempster, E. Aboutanios and W. Crowe, "Design and Development of the UNSW QB50 CubeSat - EC0," in *Proceedings of the International Astronautical Congress*, Guadalajara, Mexico, 2016.

[62] "Attitude Control," Technical University of Denmark, 25 August 2013. [Online]. Available: https://web.archive.org/web/20130825072303/http://dtusat1.dtusat.dtu.dk/. [Accessed 2 November 2019].

[63] L. Li, M. Li, L. Jiang, D. Wang, F. Zhan and T. Sheng, "Angular rate estimation and damping control of satellite with magnetometer data," *Optik,* vol. 180, pp. 1049-1055, February 2019.

[64] J. D. Searcy and H. J. Pernicka, "Magnetometer-Only Attitude Determination Using Novel Two-Step Kalman Filter Approach," *Journal of Guidance, Control, and Dynamics,* vol. 35, no. 6, pp. 1693-1701, November 2012.

[65] V. Vedant, P. Haddox and A. R. Ghosh, "Hardware in loop simulation for attitude determination and control of Illinisat-2 bus," in *26th AAS/AIAA Space Flight Mechanics Meeting*, Napa, CA, 2016.

[66] K. Krogh and E. Schreder, "Attitude Determination for AAU CubeSat," Aalborg University, Aalborg, Denmark, 2002.

[67] S. Kamal, K. Potty, C. Nagarajan, S. Mayya and A. Boratkar, "Descent Modeling and Attitude Control of a Tethered Nano-Satellite," in *2014 IEEE Aerospace Conference*, Big Sky, MT, 2014.

[68] Y. Kim and H. Bang, "Introduction to Kalman Filter and Its Applications," in *Introduction and Implementations of the Kalman Filter*, F. Govaers, Ed., London, United Kingdom, IntechOpen, 2019.

[69] J. Soh and X. Wu, "A Scalable, FPGA-Based Implementation of the Unscented Kalman Filter," in *Introduction and Implementations of the Kalman Filter*, F. Govaers, Ed., London, United Kingdom, IntechOpen, 2019.

[70] A. M. Si Mohammed, A. Boudjemai, S. Chouraqui and M. Benyettou, "Magnetorquer Control for Orbital Manoeuvre of Low Earth Orbit Microsatellite," in *Proceedings of the 5th WSEAS International Conference on Applied Computer Science*, Hangzhou, China, 2006.

[71] A. M. Si Mohammed, A. Bellar, Y. Bentoutou, A. Boudjemai, R. Roubache and N. Taleb, "Extended and unscented Kalman filtering for attitude estimation application to the active control system for microsatellite - A comparative study," in *2014 IEEE 27th Canadian Conference on Electrical and Computer Engineering*, Toronto, Canada, 2014.

[72] M. Y. Ovchinnikov, D. S. Roldugin, D. S. Ivanov and V. I. Penkov, "Choosing control parameters for three axis magnetic stabilization in orbital frame," *Acta Astronautica,* vol. 116, pp. 74-77, December 2015.

[73] P. Tortora, Y. Oshman and F. Santoni, "Spacecraft Angular Rate Estimation from Magnetometer Data Only Using an Analytic Predictor," *Journal of Guidance, Control, and Dynamics,* vol. 27, no. 3, pp. 365-373, 2004.

[74] T. M. Habib, "Combined spacecraft orbit and attitude control through extended Kalman filtering of magnetometer, gyro, and GPS measurements," *The Egyptian Journal of Remote Sensing and Space Science,* vol. 17, no. 1, pp. 87-94, June 2014.

[75] L. K. Alminde, J. Christiansen, K. K. Laursen and A. Midtgaard, "GomX-1: A Nano-satellite Mission to Demonstrate Improved Situational Awareness for Air Traffic Control," in *26th Annual AIAA/USU Conference on Small Satellites*, Logan, UT, 2012.

[76] M. D. Shuster and S. D. Oh, "Three-Axis Attitude Determination from Vector Observations," *Journal of Guidance and Control,* vol. 4, no. 1, pp. 70-77, 1981.

[77] Y. Yang, Spacecraft Modeling, Attitude Determination, and Control: Quaternion-based Approach, 1st ed., Boca Raton, FL: CRC Press, 2019.

[78] A. Caubet and J. D. Biggs, "Design of an Attitude Stabilization Electromagnetic module for detumbling uncooperative targets," in *2014 IEEE Aerospace Conference*, Big Sky, MT, 2014.

[79] "Alaska Research CubeSat Attitude Control and Determination System flight code," 3 March 2015. [Online]. Available: https://github.com/AlaskaResearchCubeSat/ACDS. [Accessed 30 October 2019].

[80] D. Messmann, F. Coelho, P. Niermeyer, M. Langer, H. Huang and U. Walter, "Magnetic Attitude Control for the MOVE-II Mission," in *7th European Conference for Aeronautics and Aerospace Sciences*, Milan, Italy, 2017.

[81] D. S. Roldugin and P. Testani, "Spin-stabilized satellite magnetic attitude control scheme without initial detumbling," *Acta Astronautica,* vol. 94, no. 1, pp. 446-454, February 2014.

[82] B. P. Kumar, B. S. Kumar, C. Sujatha and S. Bandyopadhyay, "B-DOT Moment Offset Control Algorithm for Geo-Magnetic Field Pointing IITMSAT," in *Second IAA Conference on Dynamics and Control of Space Systems*, Rome, Italy, 2015.

[83] M. Lovera and A. Astolfi, "Global Magnetic Attitude Control of Inertially Pointing Spacecraft," *Journal of Guidance, Control, and Dynamics,* vol. 28, no. 5, September 2005.

[84] F. Celani, "Robust three-axis attitude stabilization for inertial pointing spacecraft using magnetorquers," *Acta Astronautica,* vol. 107, pp. 87-96, March 2015.

[85] D. M. Torczynski, R. Amini and P. Massioni, "Magnetorquer Based Attitude Control for a Nanosatellite Testplatform," in *AIAA Infotech@Aerospace*, Atlanta, GA, 2010.

[86] D. Giri, B. Mukherjee and M. Sinha, "Three-Axis Global Magnetic Attitude Control of Earth-Pointing Satellites in Circular Orbit," *Asian Journal of Control,* vol. 19, no. 6, pp. 2028-2041, November 2017.

[87] F. Della Rossa, F. Dercole and M. Lovera, "Attitude stability analysis for an Earth pointing, magnetically controlled spacecraft," *IFAC Proceedings Volumes,* vol. 46, no. 19, pp. 518-523, 2013.

[88] M. Lovera and A. Astolfi, "Global Magnetic Attitude Control of Spacecraft in the Presence of Gravity Gradient," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 42, no. 3, pp. 796-805, 20 November 2006.

[89] T. Inamori, K. Otsuki, Y. Sugawara, P. Saisutjarit and S. Nakasuka, "Three-axis attitude control by two-step rotations using only magnetic torquers in a low Earth orbit near the magnetic equator," *Acta Astronautica,* vol. 128, pp. 696-706, 2016.

[90] N. Lequette, L. Solovyeva and A. Tavant, "X-CubeSat data analysis and mission results," in *8th Interplanetary CubeSat Workshop*, Milan, Italy, 2019.

[91] N. Jovanović, "Aalto-2 satellite attitude control system," Aalto University, Espoo, Finalnd, 2014.

[92] S. Kukreti, A. Walker, P. Putman and K. Cohen, "Genetic Algorithm Based LQR for Attitude Control of a Magnetically Actuated CubeSat," in *AIAA Infotech @ Aerospace*, Kissimmee, FL, 2015.

[93] R. Wiśniewski, "Linear Time-Varying Approach to Satellite Attitude Control Using Only Electromagnetic Actuation," *Journal of Guidance, Control, and Dynamics,* vol. 23, no. 4, pp. 640-650, July 2000.

[94] R. Wiśniewski and M. Blanke, "Fully magnetic attitude control for spacecraft subject to gravity gradient," *Automatica,* vol. 35, no. 7, pp. 1201-1214, July 1999.

[95] M. Abdelrahman, I. Chang and S. Park, "Magnetic torque attitude control of a satellite using the state-dependent Riccati equation technique," *International Journal of Non-Linear Mechanics,* vol. 46, no. 5, pp. 758-771, June 2011.

[96] T. S. Kelso, "FAQS: Two-Line Element Set Format," CelesTrak, 19 June 2018. [Online]. Available: http://celestrak.com/columns/v04n03/. [Accessed 5 May 2019].

[97] S. Ulrich. Orbital Mechanics. (2016, Fall). AERO 3240. Ottawa, ON: Carleton University.

[98] R. Bauer, Class Lecture, Topic: “Chapter 8 – Orbital Elements.” MECH 6560, Dalhousie University, Halifax, NS, 2018.

[99] C. Peat, "ISS – Orbit," Heavens Above, 2019. [Online]. Available: https://heavens-above.com/orbit.aspx?satid=25544. [Accessed 5 May 2019].

[100] A. de Ruiter, C. J. Damaren and J. R. Forbes, Spacecraft Dynamics and Control: An Introduction, Chichester, United Kingdom: John Wiley & Sons Ltd., 2013.

[101] MathWorks, "International Geomagnetic Reference Field 12," MATLAB Aerospace Blockset Documentation, 2020. [Online]. Available: https://www.mathworks.com/help/aeroblks/internationalgeomagneticreferencefield12.html. [Accessed 5 November 2020].

[102] P. K. Seidelmann, Ed., Explanatory Supplement to the Astronomical Almanac, Mill Valley, CA: University Science Books, 1992.

[103] H. D. Curtis, Orbital Mechanics for Engineering Students, 3rd ed., Oxford, United Kingdom: Butterworth-Heinemann, 2014.

[104] F. L. Markley and J. L. Crassidis, Fundamentals of Spacecraft Attitude Determination and Control, New York, NY: Springer Science + Business Media, 2014.

[105] W. J. Larson and J. R. Wertz, Space Mission Analysis and Design, 3rd ed., El Segundo, CA: Microcosm Press, 2005.

[106] M. T. Ferrés, "Design and Implementation of an Attitude Determination and Control System for the AntelSat," M. S. Thesis, Department of Engineering, Universidad de la República, Montevideo, Uruguay, 2015.

[107] J. Gießelmann, "Development of an Active Magnetic Attitude Determination and Control System for Picosatellites on highly inclined circular Low Earth Orbits," M. S. Thesis, School of Aerospace, Mechanical and Manufacturing Engineering,, RMIT University, Melbourne, Australia, 2006.

[108] K. P. Vega, "Attitude Control System for CubeSat for Ions, Neutrals, Electrons and MAGnetic Field (CINEMA)," M. S. Thesis, Department of Mechanical Engineering, University of California, Berkeley, CA, 2009.

[109] D. Miller, "Design optimization of the CADRE Magnetorquers," University of Michigan, Ann Arbor, MI, 2013.

[110] MathWorks, "Band-Limited White Noise," Simulink Documentation, 2020. [Online]. Available: https://www.mathworks.com/help/simulink/slref/bandlimitedwhitenoise.html. [Accessed 14 11 2020].

[111] OSRAM, "DIL SMT Ambient Light Sensor: Silicon Photodiode with Vλ Characteristic," SFH 2430 Z datasheet, May 2018..

[112] Dalhousie Space Systems Lab ADCS Subsystem, *CAD Assembly V3.7,* 2020.

[113] J. C. Springmann, "Satellite Attitude Determination with Low-Cost Sensors," PhD Thesis, Department of Engineering, University of Michigan, Ann Arbor, MI, 2013.

[114] Bosch Sensortec, "Intelligent 9-axis absolute orientation sensor," BNO055 datasheet, Nov. 2014..

[115] NewSpace Systems, "NCTR-M002 Interface Control Document," NCTR-M002 datasheet, 2018..

[116] K. Spendier, Class Lecture, Topic: "A Current-Carrying Coil as a Magnetic Dipole." PES1120, University of Colorado Colorado Springs, Colorado Springs, CO, 2014.

[117] T. W. Flatley, W. Morgenstern, A. Reth and F. Bauer, "A B-Dot Acquisition Controller for the RADARSAT Spacecraft," in *Flight Mechanics Symposium*, Greenbelt, MD, 1997.

[118] MathWorks, "Display map latitude and longitude data," MATLAB Mapping Toolbox Documentation, 2020. [Online]. Available: https://www.mathworks.com/help/map/ref/geoshow.html. [Accessed 4 12 2020].

[119] A. Wailand and R. Bauer, "Investigation of gain tuning and sensor noise for CubeSat B-dot detumbling and 3-axis PD magnetic attitude control," in *Proceedings of the*

Canadian Society for Mechanical Engineering International Congress 2020, Charlottetown, PE, 2020.

[120] A. C. Stickler and K. T. Alfriend, "Elementary Magnetic Attitude Control System," *Journal of Spacecraft and Rockets,* vol. 13, no. 5, pp. 282-287, 1976.

[121] M. N. Nounou and B. R. Bakshi, "Multiscale Methods for Denoising and Compression," in *Data Handling in Science and Technology Volume 22*, Amsterdam, The Netherlands, Elsevier Science Publishers B.V., 2000, pp. 119-150.

[122] K. Großekatthöfer and Z. Yoon, Class Lecture, Topic: "Introduction into quaternions for spacecraft attitude representation." Department of Astronautics and Aeronautics, Technical University of Berlin, Berlin, Germany, 31 May 2012.

[123] MathWorks, "Calculate inverse of quaternion," MATLAB Aerospace Blockset Documentation, 2020. [Online]. Available: https://www.mathworks.com/help/aerotbx/ug/quatinv.html. [Accessed 15 11 2020].

[124] M. Fadly, O. Sidek, M. Said, H. Djojodihardjo and A. Ain, "Deterministic and Recursive Approach in Attitude," *TELKOMNIKA,* vol. 9, no. 3, pp. 583-594, 2011.

[125] V. Carrara, R. B. Januzi, D. H. Makita, L. Santos and L. S. Sato, "The ITASAT CubeSat Development and Design," *Journal of Aerospace Technology and Management,* vol. 9, no. 2, pp. 147-156, 2017.

[126] L. J. E. Campos and E. C. Furtado, "Analysis of Quaternions Components in QUEST Algorithm - The Duality Problem," in *Proceedings of the XXXVIII Iberian Latin-American Congress on Computational Methods in Engineering*, Florianópolis, Brazil, 2017.

[127] FAULHABER, "Brushless DC-Flat Motors," Series 2610 ... B datasheet, Feb. 2018.

[128] MathWorks, "About Aerospace Coordinate Systems," MATLAB Aerospace Blockset Documentation, 2019. [Online]. Available: https://www.mathworks.com/help/aeroblks/about-aerospace-coordinate-systems.html. [Accessed 5 May 2019].

[129] MathWorks, "ECEF Position to LLA," MATLAB Aerospace Blockset Documentation, 2019. [Online]. Available: https://www.mathworks.com/help/aeroblks/ecefpositiontolla.html?s_tid=doc_ta. [Accessed 5 May 2019].

[130] R. Burtch, "A Comparison of Methods Used in Rectangular to Geodetic Coordinate Transformations," in *ACSM Ann. Conf. and Technology Exhibition*, Orlando, FL, 2006.

[131] MathWorks, "LLA to ECEF Position," MATLAB Aerospace Blockset Documentation, 2019. [Online]. Available: https://www.mathworks.com/help/aeroblks/llatoecefposition.html?s_tid=doc_ta. [Accessed 5 May 2019].

[132] R. G. Madonna, Orbital Mechanics, Malabar, FL: Krieger Publishing Company, 1997.

[133] MathWorks, "Geodetic to Geocentric Latitude," MATLAB Aerospace Blockset Documentation, 2019. [Online]. Available:

https://www.mathworks.com/help/aeroblks/geodetictogeocentriclatitude.html?s_tid=doc_ta. [Accessed 5 May 2019].

[134] MathWorks, "Direction Cosine Matrix ECEF to NED," MATLAB Aerospace Blockset Documentation, 2020. [Online]. Available: https://www.mathworks.com/help/aeroblks/directioncosinematrixeceftoned.html. [Accessed 5 11 2020].

[135] MathWorks, "Rotate Vector by Quaternion," MATLAB Aerospace Blockset Documentation, 2019. [Online]. Available: https://www.mathworks.com/help/aerotbx/ug/quatrotate.html. [Accessed 14 February 2020].

[136] R. Wiśniewski, "Three-Axis Satellite Attitude Control Based on Magnetic Torquing - Linear Optimal Approach," *IFAC Proceedings Volumes,* vol. 29, no. 8, pp. 91-96, December 1996.

[137] S. Lee, S. Cho, B. Lee and J. Kim, "Design, Implementation, and Validation of KOMPSAT-2 Software Simulator," *ETRI Journal,* vol. 27, no. 2, pp. 140-152, April 2005.

[138] K. Zhou, H. Huang, X. Wang and L. Sun, "Magnetic attitude control for Earth-pointing satellites in the presence of gravity gradient," *Aerospace Science and Technology,* vol. 60, pp. 115-123, January 2017.

Appendix A: Coordinate Frame Definition

This appendix summarizes the coordinate frames used to develop the CubeSat simulation tool. The coordinate frames place the satellite and its orbit relative to Earth and Sun in space throughout its mission life, drawn by the orbit propagator.

A.1 Coordinate Frames

This section defines the reference coordinate systems to which orbit geometry and orbital dynamics are referenced. Coordinate frames are considered as belonging to one of two categories. Earth frames reference geodesy and the orientation of Earth within its heliocentric orbit and Satellite frames which depend upon the geometry of the satellite and its geocentric orbit.

A.1.1 Earth Frames

Three Earth reference frames are defined: a fixed Earth-centred inertial, a rotating Earth-centred-Earth-fixed, and a geodetic latitude, longitude, altitude frame. Simulation results are chiefly expressed in a combination of these three frames.

A.1.1.1 Earth-Centered Inertial (ECI)

The Earth-centred inertial (ECI) frame – denoted $\mathcal{F}_I$ – is illustrated in Figure 97. This non-rotating reference frame has an origin that is fixed at the centre of Earth and translates with Earth. The +X axis points in the direction of the vernal equinox (and lies within Earth's equatorial plane). The +Z axis points towards the North Pole aligned with Earth's polar axis [128], and the +Y axis completes the triad. The vernal equinox direction is assumed to be constant throughout Earth's heliocentric orbit, and neither precession nor nutation of the polar axis are considered in this definition. Over multiple orbits, the Satellite’s unperturbed orbit trace should remain unchanged when viewed in the ECI frame and appear as a single, slightly elliptical path.

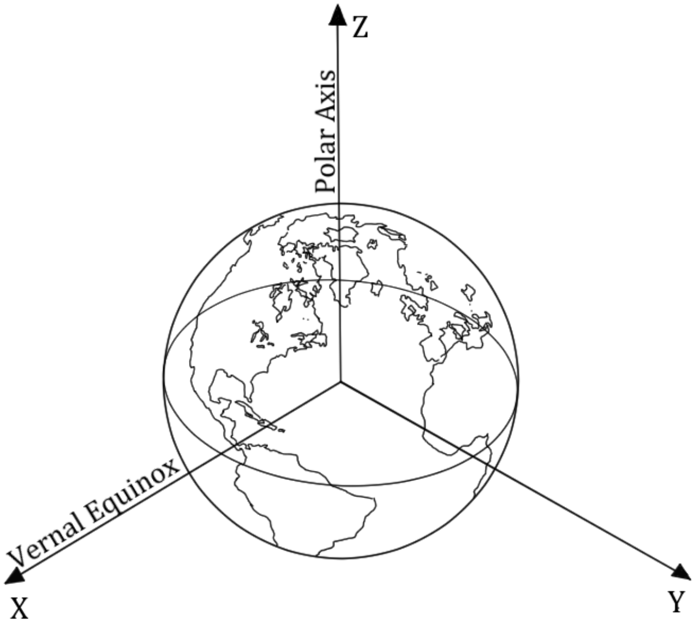


Figure 97: ECI Reference Frame

A.1.1.2 Earth-Centred Earth-Fixed (ECEF)

The Earth-centred Earth-fixed (ECEF) reference frame $\mathcal{F}_F$ rotates with Earth and has its origin at Earth's centre. As shown in Figure 98, the frame's +X axis points along the prime meridian (line of zero longitude) and its +Z axis points towards the North Pole aligned with Earth's axis of rotation, coincident with the +Z axis of the ECI frame [97, p. 30]. The ECEF frame is essential for obtaining the satellite's position in geodetic terms, as it tracks the position of the orbiting body with respect to a rotating Earth.

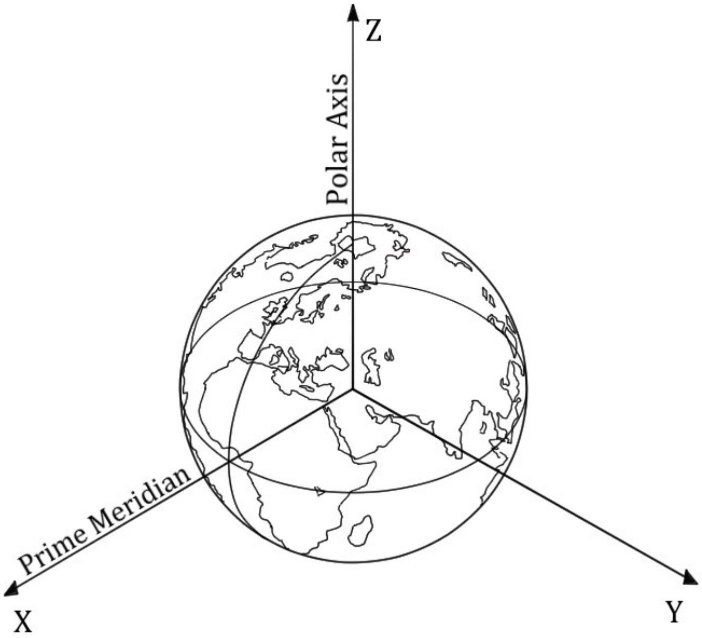


Figure 98: ECEF Reference Frame

A.1.1.3 Latitude, Longitude, Altitude (LLA)

The Latitude, Longitude, Altitude (LLA) reference frame – denoted $\mathcal{F}_{LLA}$ – is illustrated in Figure 99 and defines the orbiting satellite's position in terms of geodetic longitude l, geodetic latitude μ, and altitude h with respect to the surface of an oblate-spheroidal model of Earth. Geodetic longitude is measured relative to the prime meridian at 0°, with East longitude considered positive up to $+180°$ and West longitude considered negative up to $-180°$. Geodetic latitude ranges from $-90°$ at the South pole to $+90°$ at the North pole. Figure 99 shows geodetic latitude μ, altitude h and compares μ with geocentric latitude λ. Earth's oblateness is exaggerated in this figure to better illustrate the difference between geocentric and geodetic latitudes. The satellite shown is in a circular orbit at an exaggeratedly low latitude for illustrative purposes.

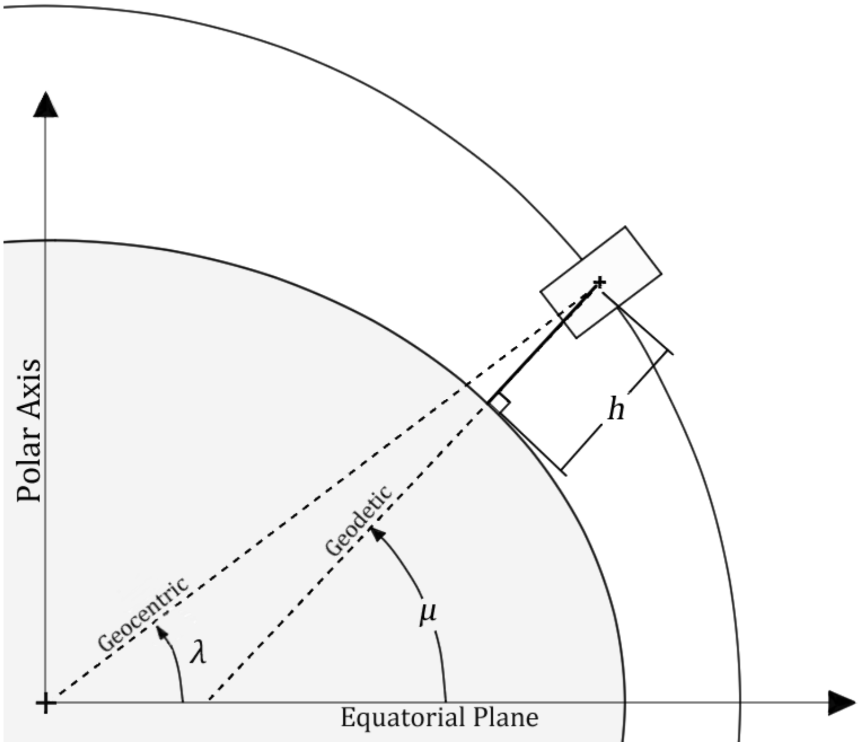


Figure 99: LLA Reference Frame Showing Geocentric Latitude λ, Geodetic Latitude μ, and Geodetic Altitude h (Longitude not Shown)

A.1.1.4 Satellite Frames

Four satellite reference frames are defined within the simulator: the perifocal frame, the Nadir-pointing frame, the body-fixed frame, and the North-East-Down frame.

A.1.1.5 Perifocal (P)

The perifocal reference frame – denoted $\mathcal{F}_P$ – is shown in Figure 100. Its origin is located at Earth's centre and its axes are oriented with respect to the orbital plane. The +X axis points in the direction of the perigee of the orbit $\overrightarrow{r_p}$ while the +Z axis is normal to the orbital plane pointing in the direction of orbital angular momentum [97, p. 51].

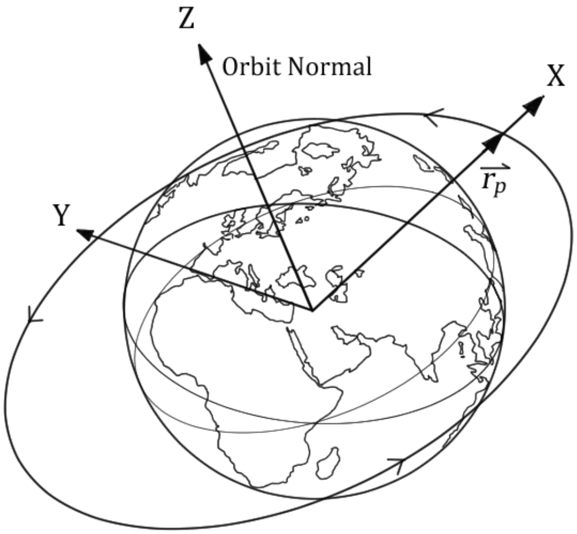


Figure 100: Perifocal Reference Frame

A.1.1.6 Nadir-Pointing (NP)

The nadir-pointing frame shown in Figure 101– denoted $\mathcal{F}_{NP}$ – has its origin at the satellite's centre of mass. It serves as a reference for spacecraft attitude defining the desired Earth-pointing orientation of the satellite. The +Z axis points nadir towards the centre of Earth and the +X axis points tangent to the orbit in the direction of travel. This reference frame is illustrated in Figure 101 for the same elliptical orbit as in Figure 100.

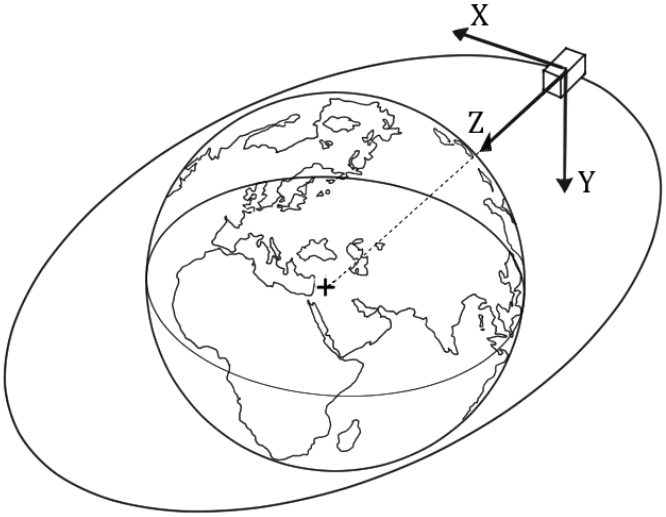


Figure 101: NP Reference Frame

A.1.1.7 Body-Fixed (BF)

The body-fixed frame – denoted $\mathcal{F}_B$ – is aligned with the satellite's principal inertial axes with its origin at the satellite's centre of mass. The coordinate system shown in Figure 102 is oriented such that its +Z axis points nadir (aligned with the orbital radius) and its +X axis lies in the direction of spacecraft propagation (aligned with the body-fixed spacecraft velocity vector).

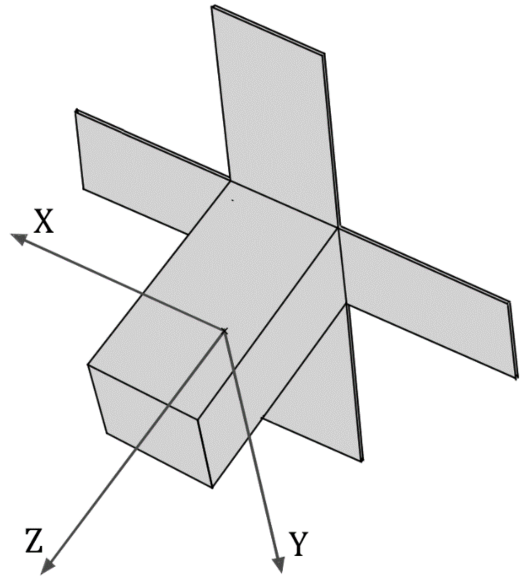


Figure 102: BF Reference Frame

A.1.1.8 North-East-Down (NED)

The North-East-Down coordinate frame shown in Figure 103 – denoted $\mathcal{F}_{NED}$- is fixed at the satellite's centre of mass. "North" points parallel to the Earth's surface towards the polar North, assigned as +X [128]. "East" points parallel to the lines of latitude, assigned as +Y, and "Down" points antiparallel to Earth's surface normal, assigned as +Z [128]. It is necessary to note that – unlike all other satellite frames described in this section – the NED frame Z axis does not point towards the centre of Earth. Earth's oblateness is exaggerated in Figure 103 to emphasise the difference between the intersection of the surface normal with the equatorial plane and the centre of Earth.

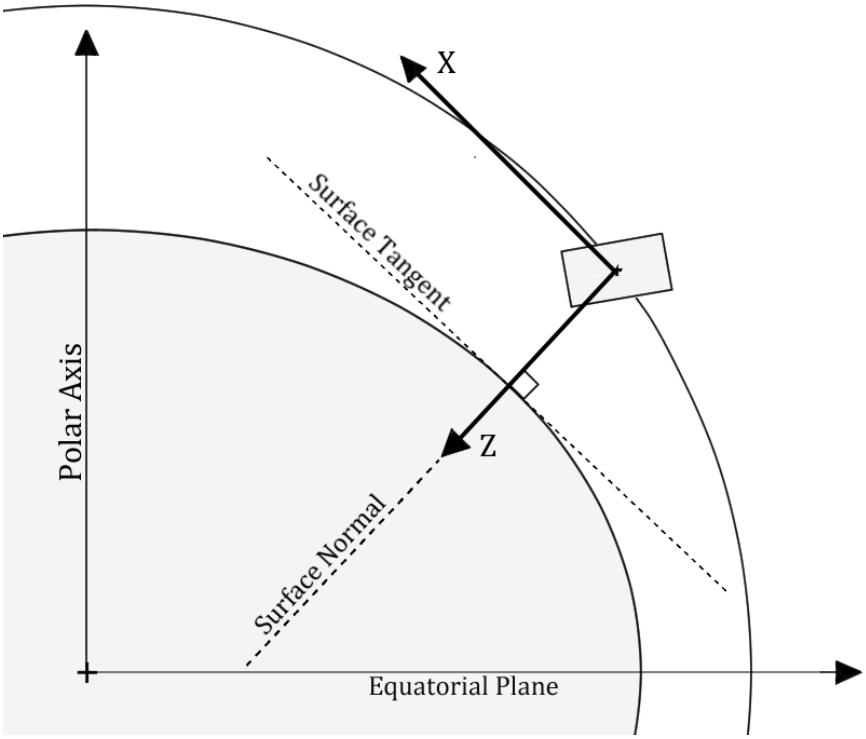


Figure 103: NED Reference Frame

The built-in MATLAB and Simulink functions for both the International Geomagnetic Reference Field (IGRF) and World Magnetic Model (WMM) express magnetic field vectors in the NED frame relative to geodetic latitude, longitude, and altitude.

Appendix B: Coordinate Frame Transformations

This appendix summarizes the coordinate frame transformations used to develop the CubeSat simulation tool.

B.1 Coordinate Transformations

Certain orbital and simulator parameters are inherently expressed within certain frames of reference, such as MATLAB's IGRF block producing magnetic field vectors in NED and data from the TLE reported in ECI. Coordinate system transformations become a necessity when reporting results and combining data with one another, e.g. translating the magnetic field strength into usable attitude control torques in the body-fixed frame or plotting the ground track of the satellite from its ECI radii over one period.

Transformations between each of the previously described coordinate systems follow the sequence shown in Figure 104. For example, a vector expressed in the ECI frame of reference can be expressed as the same vector relative to the body-fixed frame of reference by multiplying together the direction cosine matrices for the ECI to perifocal,

perifocal to nadir-pointing, and nadir-pointing to body-fixed transformations to produce the transformation matrix between two frames, and vice versa. This transformation matrix then multiplies the components of a vector expressed in the ECI frame to obtain the components of the same vector expressed in the body-fixed frame.

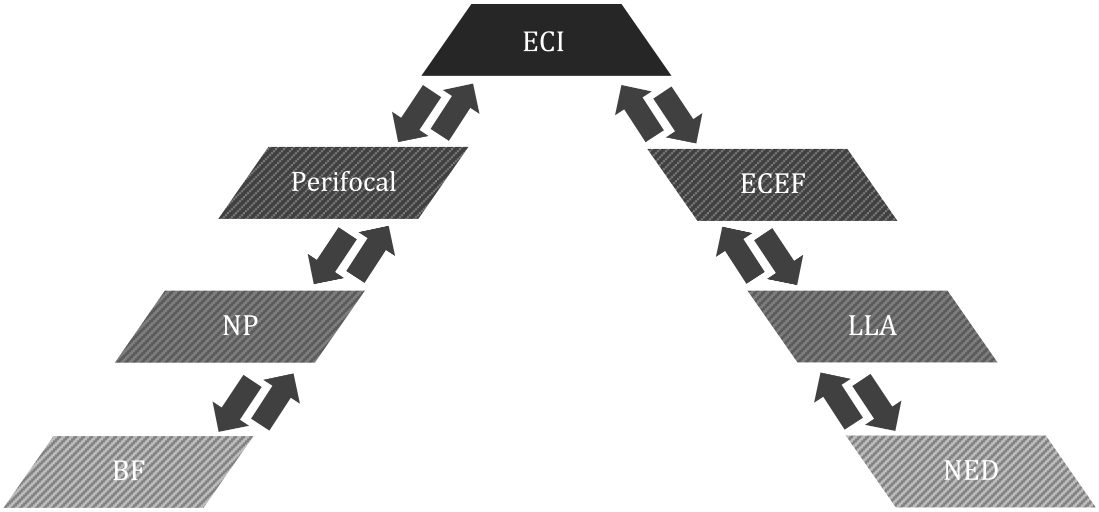


Figure 104: Sequence of Coordinate Transformations

B.1.1 Principal Rotations

The simplest form of a direction cosine matrix is the principal rotation matrix, which rotates a vector by some angle Θ about one of the three principal axes. Three principal rotation matrices are thus defined about the three principal axes X, Y, and Z denoted $\boldsymbol{C}_1, \boldsymbol{C}_2,$ and $\boldsymbol{C}_3$, respectively. The three principal rotation matrices are defined as follows for a rotation through an angle Θ as [100, p. 15]:

$$\boldsymbol{C}_1(\Theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Theta & \sin\Theta \\ 0 & -\sin\Theta & \cos\Theta \end{pmatrix} \tag{98}$$

$$\boldsymbol{C}_2(\Theta) = \begin{pmatrix} \cos\Theta & 0 & -\sin\Theta \\ 0 & 1 & 0 \\ \sin\Theta & 0 & \cos\Theta \end{pmatrix} \tag{99}$$

$$\boldsymbol{C}_3(\Theta) = \begin{pmatrix} \cos\Theta & \sin\Theta & 0 \\ -\sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{100}$$

For a rotation sequence between two arbitrary frames A and B, the principal rotation matrices are multiplied in their opposite order of appearance within the sequence.

For example, a 1-2-3 sequence implies first a rotation about the initial X axis (1) of frame A, followed by a rotation about an intermediate Y axis (2), and a final rotation about an intermediate Z axis (3) to yield frame B. Assuming the rotations are through angles defined as Θ_1, Θ_2 and Θ_3 corresponding to their respective rotation axes, the 1-2-3 rotation matrix $\boldsymbol{C}_{123}$ is defined by the product of the principal rotation matrices in their reverse order of appearance (3-2-1):

$$\boldsymbol{C}_{BA} = \boldsymbol{C}_{123} = \boldsymbol{C}_3(\Theta_3)\boldsymbol{C}_2(\Theta_2)\boldsymbol{C}_1(\Theta_1)$$

Reducing this matrix produces the result:

$$\boldsymbol{C}_{BA} = \boldsymbol{C}_{123} = \begin{pmatrix} \cos\Theta_3 & \sin\Theta_3 & 0 \\ -\sin\Theta_3 & \cos\Theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\Theta_2 & 0 & -\sin\Theta_2 \\ 0 & 1 & 0 \\ \sin\Theta_2 & 0 & \Theta_2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Theta_1 & \sin\Theta_1 \\ 0 & -\sin\Theta_1 & \cos\Theta_1 \end{pmatrix}$$

$$\boldsymbol{C}_{BA} = \boldsymbol{C}_{123}$$
$$= \begin{pmatrix} \cos\Theta_2\cos\Theta_3 & \cos\Theta_1\sin\Theta_3 + \cos\Theta_3\sin\Theta_1\sin\Theta_2 & \sin\Theta_1\sin\Theta_3 - \cos\Theta_1\cos\Theta_3\sin\Theta_2 \\ -\cos\Theta_2\sin\Theta_3 & \cos\Theta_1\cos\Theta_3 - \sin\Theta_1\sin\Theta_2\sin\Theta_3 & \cos\Theta_3\sin\Theta_1 + \cos\Theta_1\sin\Theta_2\sin\Theta_3 \\ \sin\Theta_2 & -\cos\Theta_2\sin\Theta_1 & \cos\Theta_1\cos\Theta_2 \end{pmatrix}$$

To transform a vector $\vec{v}$ with components expressed in frame A $\boldsymbol{v}_A$ to the same vector with components expressed in frame B $\boldsymbol{v}_B$, the following relation is used, where $\boldsymbol{C}_{BA}$ contains the transformation sequence between the two frames of reference:

$$\boldsymbol{v}_B = \boldsymbol{C}_{BA}\boldsymbol{v}_A \tag{101}$$

The leading letter, number, or initialism in the rotation matrix subscript indicates the resulting coordinate frame of the transform while the last indicates the base. In the following sections, this notation is used in accordance with the letters and initialisms for the different reference frames defined in Section A.1.

B.1.2 Perifocal to ECI

The perifocal reference frame is obtained from the ECI reference frame via a 3-1-3 rotation sequence through the classical orbital elements RAAN Ω, inclination i, and argument of perigee ω, respectively [97, p. 52]. The transformation sequence angles are illustrated for an elliptical orbit alongside the two coordinate system axes in Figure 105. The Y axes are omitted from the diagram for ease of visualisation.

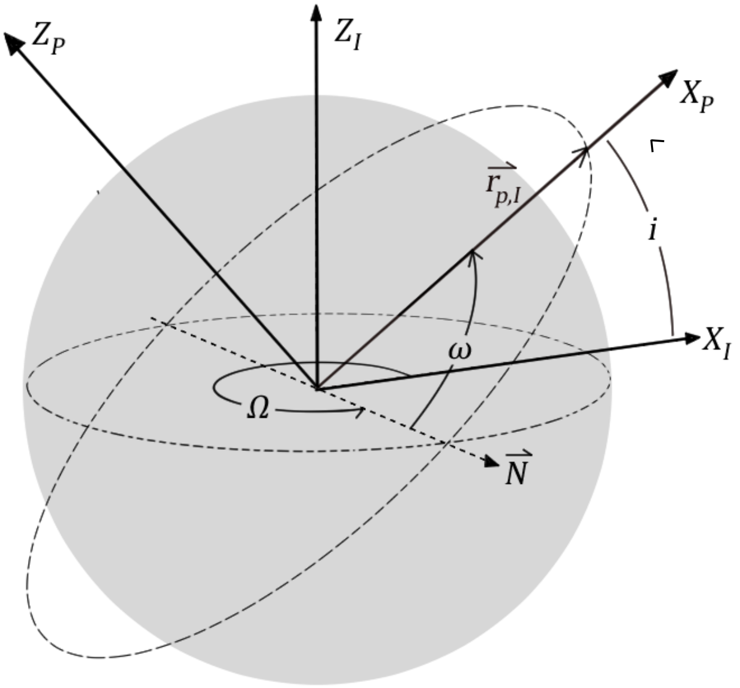


Figure 105: ECI to Perifocal Transformation for a Generic Inclined Elliptical Orbit

In reference to Figure 105, first, a rotation about the ECI Z axis by Ω aligns the intermediate X-axis with the orbit line of nodes $\boldsymbol{N}$. Then, the orbital plane is rotated by the inclination i about this intermediate X-axis making the intermediate Z-axis normal to the orbital plane. Lastly, the orbit is rotated about this intermediate Z-axis from $\boldsymbol{N}$ by the argument of perigee ω to align the intermediate X-axis with the eccentricity vector of the orbit (and, by extension, the radii at apogee and perigee). The transformation matrix for the above described sequence is shown below [97, p. 52]:

$$\boldsymbol{C}_{PI} = \boldsymbol{C}_3(\omega)\boldsymbol{C}_1(i)\boldsymbol{C}_3(\Omega) \tag{102}$$

$$\boldsymbol{C}_{PI} = \begin{pmatrix} \cos\Omega\cos\omega - \sin\Omega\cos i\sin\omega & \sin\Omega\cos\omega + \cos\Omega\cos i\sin\omega & \sin i\sin\omega \\ -\cos\Omega\sin\omega - \sin\Omega\cos i\cos\omega & \cos\Omega\cos i\cos\omega - \sin\Omega\sin\omega & \cos\omega\sin i \\ \sin\Omega\sin i & -\cos\Omega\sin i & \cos i \end{pmatrix}$$

To transform the components of a vector $\vec{r}$ expressed in the ECI frame into the same vector with components expressed in the perifocal frame, the following relation is used:

$$\boldsymbol{r}_P = \boldsymbol{C}_{PI}\boldsymbol{r}_I$$

To transform from perifocal to ECI, the above sequence is reversed:

$$\boldsymbol{C}_{IP} = \boldsymbol{C}_{PI}^{-1}$$

$$\boldsymbol{r}_I = \boldsymbol{C}_{IP}\boldsymbol{r}_P$$

The perifocal to ECI rotation sequence $\boldsymbol{C}_{IP}$ is illustrated in Figure 106. The leftmost diagram – representing $\boldsymbol{C}_3(\Omega)$ – illustrates the rotation about the ECI Z axis by Ω where the X-axis of the perifocal frame is shown as the perigee radius in ECI $\boldsymbol{r}_{p,I}$. The second diagram shows the rotation $\boldsymbol{C}_1(i)$ about the ECI X-axis, aligning the orbital and equatorial planes. Lastly, the rightmost diagram shows $\boldsymbol{C}_3(\omega)$ about the shared ECI/perifocal Z-axis, with the equatorial-plane-coincident orbit trace omitted for visualisation purposes.

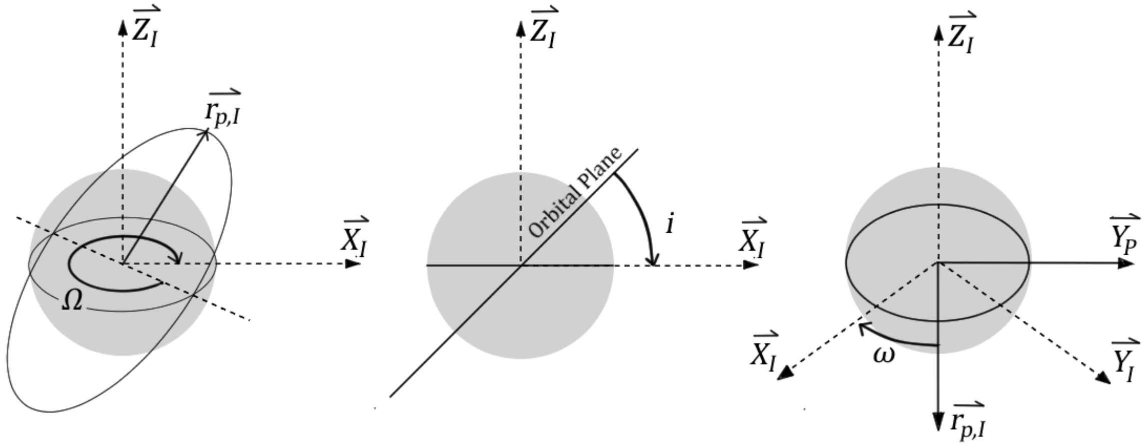


Figure 106: Perifocal to ECI Transformation Steps

B.1.3 ECI to ECEF

As shown in Figure 107, the ECEF frame is obtained from the ECI frame via a single rotation about their shared Z axis through θ_G, the Greenwich sidereal time. The simplified method to determine this angle proposed in [103, pp. 258-262] is implemented in the simulator.

The Julian day number J_0 for the given start date at 0 UTC is determined from the following formula, where Y denotes year, M denotes month, D denotes day, and “fix” denotes a function which rounds towards zero [103, p. 259]:

$$J_0 = 367Y - \mathrm{fix}\left(\frac{7\left(Y - \mathrm{fix}\left(\frac{M+9}{12}\right)\right)}{4}\right) + \mathrm{fix}\left(\frac{275\mathrm{M}}{9}\right) + D + 1721013.5 \tag{103}$$

The Julian day is then used to determine the Julian day number at epoch J_D, where UT is the current universal time in hours [103, p. 259].

$$J_D = J_0 + \frac{\mathrm{UT}}{24}$$

With respect to the J2000 convention – the Julian epoch defined at 12:00:00 UTC 1 January 2000 – the current time in terms of Julian centuries T_0 is found [103, p. 261]:

$$T_0 = \frac{J_0 - 2451545}{36525} \tag{104}$$

This term is then used to calculate the Greenwich sidereal time at 00:00:00 UTC in degrees via the following [103, p. 261]:

$$\begin{aligned}\theta_{G0} = 100.4606184 + 36000.77004T_0 + 0.000387933T_0^2 \\ - 2.583 \times 10^{-8}T_0^3\end{aligned} \tag{105}$$

With respect to this angle, the Greenwich sidereal time at the given epoch (and at any time thereinafter) is determined as follows [103, p. 261]:

$$\theta_G = \theta_{G0} + 360.98564724\frac{\mathrm{UT}}{24} \tag{106}$$

Within the simulator, the simulation time (in seconds) is added (in hours) to the starting time (in hours) such that θ_G is recalculated for each time step. Figure 107 shows θ_G drawn between the two coordinate systems.

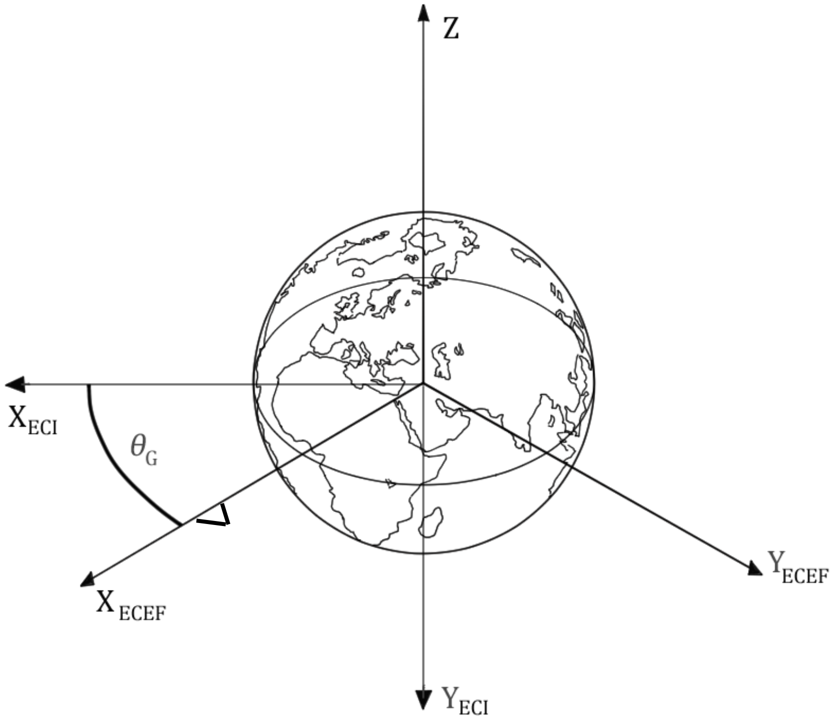


Figure 107: Greenwich Sidereal Time Between ECI and ECEF

With θ_G defined, the rotation matrix to transform from the ECEF reference frame to the ECI reference frame is defined via the principal rotation matrix $\boldsymbol{C}_3$ as [97, p. 30]:

$$\boldsymbol{C}_{FI} = \begin{pmatrix} \cos\theta_G & \sin\theta_G & 0 \\ -\sin\theta_G & \cos\theta_G & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{107}$$

To transform the components of a vector $\vec{r}$ expressed in the ECEF reference frame into the same vector with components expressed in the ECI reference frame, the following relation is used:

$$\boldsymbol{r}_I = \boldsymbol{C}_{IF}\boldsymbol{r}_F$$

The rotation matrix $\boldsymbol{C}_{FI}$ from ECI to ECEF is obtained via the inverse of the above:

$$\boldsymbol{C}_{IF} = \boldsymbol{C}_{FI}^{-1}$$

B.1.4 ECEF to LLA

The ECEF to LLA transformation is implemented computationally according to the method outlined in MATLAB's Aerospace Blockset documentation [129]. The directional components of the orbital position vector with components expressed in ECEF are given as:

$$\boldsymbol{r}_F = \begin{bmatrix} r_{x,F} \\ r_{y,F} \\ r_{z,F} \end{bmatrix}$$

Geodetic latitude is solved via Bowring's iterative method summarised in [130]. The following four parameters are defined for use in the iterative equations, where s is the distance between the polar axis and the point in orbit of interest, f the flattening factor of Earth, $R_\oplus$ Earth's equatorial radius, and e^2 the square of first eccentricity [130] [129]. Values are obtained for each as [129]:

$$s = \sqrt{r_{x,F}^2 + r_{y,F}^2}$$

$$f = \frac{1}{298.257223563}$$

$$R_\oplus = 6378.137 \text{ km}$$

$$e^2 = 1 - (1 - f)^2$$

where the two Earth parameters f and R are assigned according to the World Geodetic System (WGS84) model in MATLAB. Initial guesses for the reduced latitude $\bar{\beta}$and geodetic latitude $\bar{\mu}$ are computed via [129]:

$$\bar{\beta} = \tan^{-1}\left(\frac{r_{z,F}}{(1-f)s}\right)$$

$$\bar{\mu} = \tan^{-1}\left(\frac{r_{z,F} + \dfrac{e^2(1-f)}{(1-e^2)} R_{\oplus} \sin^3(\beta)}{s - e^2 R_{\oplus} \cos^3(\beta)}\right) \tag{108}$$

where β for each subsequent iteration is computed as [129]:

$$\beta = \tan^{-1}\left(\frac{(1-f)\sin\mu}{\cos\mu}\right)$$

The solution of μ is considered converged when the error reaches 10^{-5} or after 50 iterations, chosen in the interest of simulation speed.

The geodetic longitude l is calculated directly from the components of $\overrightarrow{r_F}$ as [129]:

$$l = \tan^{-1}\left(\frac{r_{y,F}}{r_{x,F}}\right) \tag{109}$$

The altitude is found by [129]:

$$h = s\cos\mu + \left(r_{z,F} + e^2 N \sin\mu\right)\sin\mu - N \tag{110}$$

where N is the radius of curvature in the vertical prime [129]:

$$N = \frac{R_{\oplus}}{\sqrt{1 - e^2 \sin^2\mu}}$$

B.1.5 LLA to ECEF

The inverse of the preceding transform – LLA to ECEF – is obtained computationally as follows from the geodetic latitude μ, longitude l, and altitude h. Defining the geocentric latitude at mean sea level λ_s and the radius at a surface point r_s as [131]:

$$\lambda_s = \tan^{-1}((1-f)^2 \tan\mu)$$

$$r_s = \sqrt{\frac{R^2}{1 + ((1-f)^2 - 1)^{-1} \sin^2\lambda_s}}$$

The ECEF position vector is obtained [131]:

$$\boldsymbol{r}_F = \begin{pmatrix} r_s\cos\lambda_s\cos l + h\cos\mu\cos l \\ r_s\cos\lambda_s\sin l + h\cos\mu\sin l \\ r_s\sin\lambda_s + h\sin\mu \end{pmatrix} \tag{111}$$

B.1.6 Latitude Conversion

Latitude can be expressed in geodetic or geocentric terms, the former measuring the angle between the intersection of the surface normal at the point of interest in orbit and the equatorial plane, while the latter measures the angle between the line through Earth's centre from the point of interest in orbit and the equatorial plane. Both latitudes are illustrated in Figure 108.

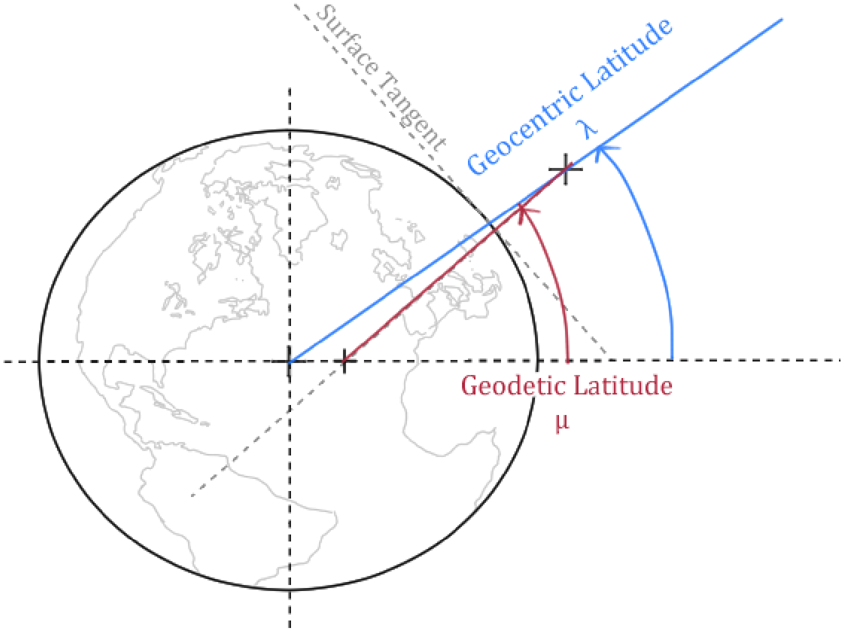


Figure 108: Geocentric and Geodetic Latitudes

To obtain geocentric latitude λ given geodetic latitude μ, the following relation is used [132, p. 25]:

$$\lambda = \tan^{-1}((1-f)^2 \tan\mu) \tag{112}$$

To obtain geodetic latitude μ given geocentric latitude λ and mean sea-level altitude h, the following relation is used, where the geocentric latitude at Earth's surface λ_s is defined as above [129] [133]:

$$\lambda_s = \tan^{-1}((1-f)^2 \tan\mu)$$
$$\lambda = \tan^{-1}\left(\frac{h\sin\mu + r_s \sin\lambda_s}{h\cos\mu + r_s\cos\lambda_s}\right) \tag{113}$$

B.1.7 ECEF to NED

The ECEF to NED transformation is described by a 3-2-2 rotation sequence through geodetic longitude l, negative geodetic latitude μ and −90° [134]. The two coordinate systems are illustrated in the following figure, with the transformation sequence following.

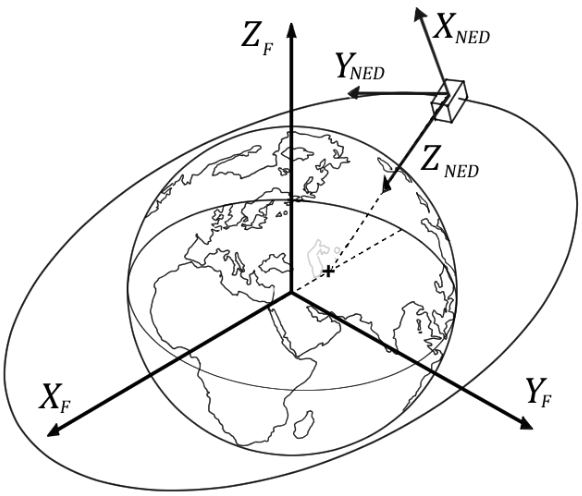


Figure 109: ECEF and NED Coordinate Frames for an Inclined Orbit

$$\boldsymbol{C}_{NEDF} = \boldsymbol{C}_2(-90)\boldsymbol{C}_2(-\mu)\boldsymbol{C}_3(l) \tag{114}$$

$$\boldsymbol{C}_{NEDF} = \begin{pmatrix} -\sin\mu\cos l & -\sin\mu\sin l & \cos\mu \\ -\sin l & \cos l & 0 \\ -\cos\mu\cos l & -\cos\mu\sin l & -\sin\mu \end{pmatrix}$$

To transform from the NED to the ECEF frame, the inverse of the direction cosine matrix is used:

$$\boldsymbol{C}_{FNED} = \boldsymbol{C}_{NEDF}^{-1}$$

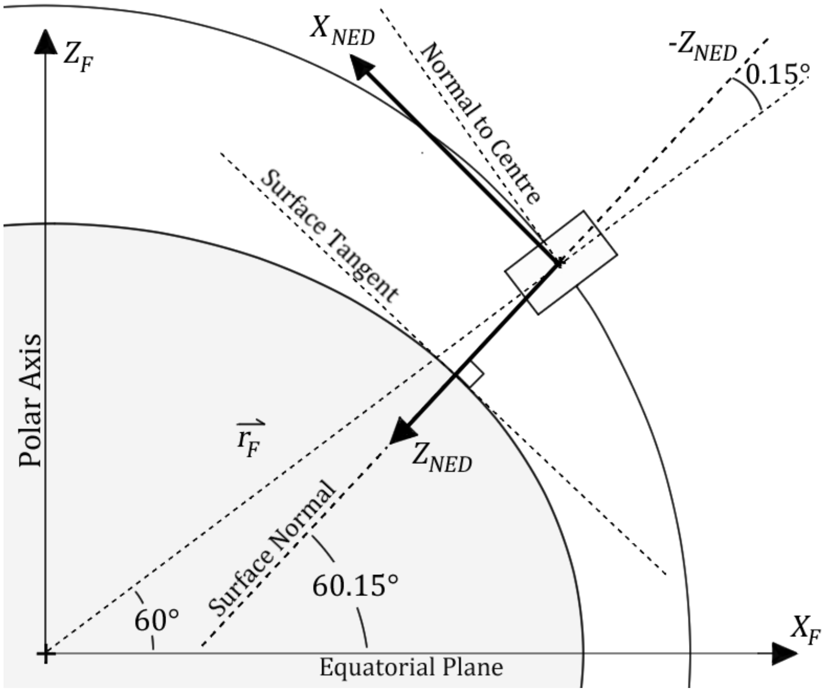


Figure 110: Geodetic and Geocentric Latitude Difference for Case 4 – Inclined Orbit

B.1.8 Perifocal to Nadir-Pointing

The perifocal to nadir-pointing transformation takes place entirely in the orbital plane. Both coordinate frames are shown in Figure 111, where the satellite's position is not coincident with the perigee of the orbit and is offset by some non-zero value of true anomaly θ.

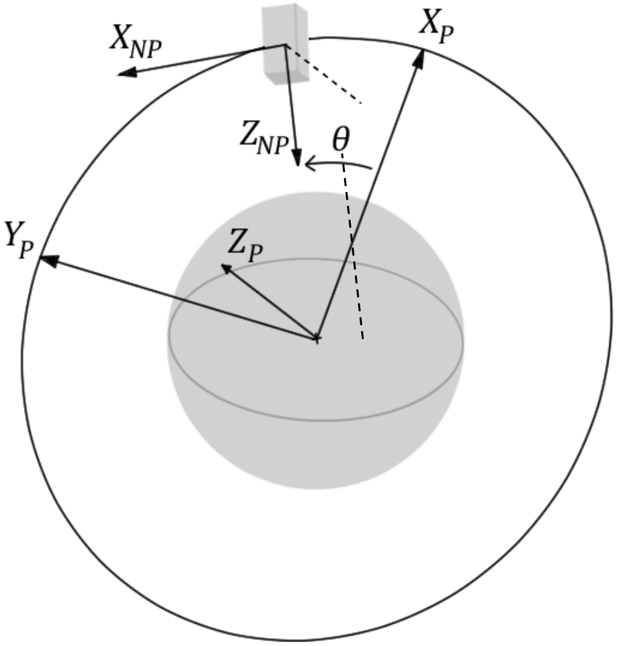


Figure 111: Nadir-Pointing and Perifocal Coordinate Frames

The transformation between the two frames follows a 3-2-3 rotation sequence through true anomaly θ, -90°, and +90°. This sequence is calculated as:

$$\boldsymbol{C}_{NPP} = \boldsymbol{C}_3(90)\boldsymbol{C}_2(-90)\boldsymbol{C}_3(\theta) \tag{115}$$

$$\boldsymbol{C}_{NPP} = \begin{pmatrix} \cos^2 90 \sin\theta - \sin 90 \sin\theta & \cos^2 90 \sin\theta + \cos\theta \sin 90 & \cos 90 \sin 90 \\ -\cos 90 \sin\theta - \cos 90 \sin 90 \cos\theta & \cos 90 \cos\theta - \cos 90 \sin 90 \sin\theta & -\sin^2 90 \\ -\cos\theta \sin 90 & -\sin 90 \sin\theta & \cos 90 \end{pmatrix}$$

which reduces to:

$$\boldsymbol{C}_{NPP} = \begin{pmatrix} -\sin\theta & \cos\theta & 0 \\ 0 & 0 & -1 \\ -\cos\theta & -\sin\theta & 0 \end{pmatrix} \tag{116}$$

The inverse coordinate transformation from the nadir-pointing frame to the perifocal frame is:

$$\boldsymbol{C}_{PNP} = \boldsymbol{C}_{NPP}^{-1}$$

B.1.9 Nadir-Pointing to Body-Fixed

Satellite attitude is commonly described in terms of Euler angles drawn between the desired and actual attitudes of the orbiting satellite, represented respectively by the nadir-pointing and body-fixed reference frames. The Euler angles roll ϕ, pitch θ, and yaw ψ define the rotation of the satellite about its centre of mass and correspond to angular rotation about the X, Y, and Z axes, each illustrated in Figure 112, showing respective positive rotation directions.

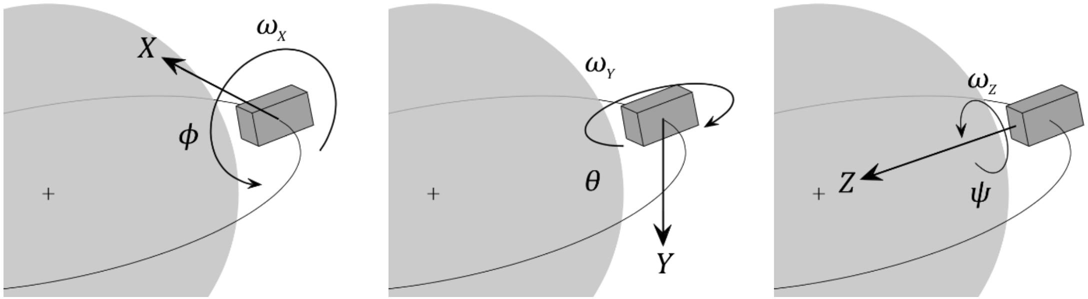


Figure 112: Roll, Pitch, and Yaw Directions in Orbit Relative to Body-Fixed Axes

To describe how the body-fixed reference frame has been rotated relative to the nadir-pointing reference frame, a 3-2-1 rotation sequence is used to align the nadir-pointing and body-fixed coordinate axes at any point in orbit. First, a rotation by yaw ψ about the NP Z-axis, followed by a rotation by pitch θ about the current Y-axis, and a rotation by roll ϕ about the current X-axis. The transformation matrix from body-fixed to nadir-pointing is defined:

$$\boldsymbol{C}_{NPBF} = \boldsymbol{C}_1(\phi)\boldsymbol{C}_2(\theta)\boldsymbol{C}_3(\psi) \tag{117}$$

$$\boldsymbol{C}_{NPBF} = \begin{pmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{pmatrix}$$

The inverse transformation matrix from the nadir-pointing frame to the body-fixed is calculated as:

$$\boldsymbol{C}_{BFNP} = \boldsymbol{C}_{NPBF}^{-1}$$

This direction cosine matrix uses Euler angles which vary continuously throughout the simulation over a range of 0 to π radians. The issue of gimbal lock arises in the typical 3-2-1 sequence, causing a “singularity” when $\theta = \pm 90° = \pi/2$. For the matrix as presented:

$$\boldsymbol{C}_{NPBF} = \begin{pmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{pmatrix}$$

Let $\theta = 90° = \pi/2$. The matrix reduces to:

$$\boldsymbol{C}_{NPBF} = \begin{pmatrix} 0 & 0 & -1 \\ \sin\phi\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\psi + \cos\phi\cos\psi & 0 \\ \cos\phi\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\psi - \sin\phi\cos\psi & 0 \end{pmatrix}$$

which can be further reduced via the sum and difference trigonometric identities:

$$\boldsymbol{C}_{NPBF} = \begin{pmatrix} 0 & 0 & -1 \\ \sin(\phi - \psi) & \cos(\phi - \psi) & 0 \\ \cos(\phi - \psi) & \sin(\psi - \phi) & 0 \end{pmatrix}$$

Because the expression for $\boldsymbol{C}_{NPBF}$ contains only the difference between ϕ and ψ, these angles cannot be uniquely determined from the rotation matrix and one degree of freedom describing the satellite's orientation is instantaneously lost in the system. To avoid this singularity, quaternions are instead used to describe how the body-fixed frame is rotated relative to the nadir-pointing frame. An arbitrary three-dimensional vector $\vec{v}$ expressed in the nadir-pointing frame $\boldsymbol{v}_{NP}$ can be placed in the body-fixed frame $\boldsymbol{v}_{BF}$ by arranging the four elements of the quaternion into a 3x3 rotation matrix. Quaternion rotation is accomplished in the simulator using the MATLAB function "quatrotate" whose transform syntax is reproduced as follows [135]:

$$\boldsymbol{v}_{BF} = q_{BFNP}\boldsymbol{v}_{NP}q_{BFNP}^{-1}$$

$$\boldsymbol{v}_{BF} = \begin{bmatrix} (1 - q_y^2 - 2q_z^2) & 2(q_xq_y + q_sq_z) & 2(q_xq_z - q_sq_y) \\ 2(q_xq_y - q_sq_z) & (1 - 2q_x^2 - 2q_z^2) & 2(q_yq_z + q_sq_x) \\ 2(q_xq_z + q_sq_y) & 2(q_yq_z - q_sq_x) & (1 - 2q_x^2 - 2q_y^2) \end{bmatrix} \begin{bmatrix} v_{NP,x} \\ v_{NP,y} \\ v_{NP,z} \end{bmatrix}$$